Implementation of effective transistor level Hamming code circuit

P.V.S.R.Bharadwaja
Assistant Professor,
S V E C ,
A.Rangampet , Andhra Pradesh , India

Neelima Koppala
Assistant Professor
S V E C,
A.Rangampet , Andhra Pradesh , India

M.Venkata Naresh
Assistant professor,
S V E C,
A.Rangampet, Andhra Pradesh, India

Abstract **- Hamming codes are very simple error detecting codes. EXOR and AND gates are required to implement hamming codes. Two input EXOR gate requires 12 transistors and two input AND gate requires 6 transistors. Increase in the number of bits in the implementation increases the number of transistors which enhances the delay and area. This is an abstemious issue in the practical implementation. To overcome this problem we reduce the utilization of number of transistors. This paper mainly focuses in reduction of number of transistors. The effective model in this paper is compared with the previous structures of hamming code by using several methods. Adiabatic logic is the low power technique is used to reduce the power. Hence an effective circuit is designed for generation and correction of hamming code with fewer transistors. In this paper we compared Power Delay Product (PDP) of several implementations with our implementation. This work was made by using HSPICE**.

Keywords—Adiabatic Logic; Body Biasing; Error detection and Correction; Hamming Codes; SPICE.

I. INTRODUCTION

With the demand of large storage and processing units, the necessity of error detection and correction circuits are gaining popularity. There exist several causes of errors[1][2][3][4]. The defect may be a permanent deformation in IC layer which may but does not have to result in fault. The fault is functional misbehaviour like IC malfunctioning. There are several types of faults caused in ICs like Stuck at 0, stuck at 1, bridge faults, stuck open fault, etc.

The faults are modelled in several ways like Intermittent Fault, Line-delay Fault, Logical Fault (often Stuck-at), Memory Fault (SA0/1, pattern sensitive, cell coupling faults), Multiple Fault, Non-classical Fault (not stuck-at, stuck-open or stuck-on for CMOS)[5][6][7][8][9], Oscillation Fault (or star-faults, bridging faults in combo logic), Parametric Fault (changes the values of electrical parameters), Path-delay Fault, Pattern Sensitive Fault, Permanent Fault, Physical Fault, Pin Fault (SA faults on the signal pins of all modules in the circuit), PLA Fault, Potentially Detectable Fault (a subset of the initialization faults), IDDQ Fault, Race Fault, Redundant Fault, Segment-delay Fault, Structural Fault, Stuck-at Fault, Stuck-open and Stuck-short Fault, Transistor Fault (Stuck-open and Stuck-short faults), Transition Fault, Untestable Fault, Assertion Fault, Behavioural Fault, Branch Fault, Bridging, Bus Fault, Cross-point Fault, Defect-oriented Fault (physical level faults, bridging, stuck-open, IDDQ), Delay Fault (transition, gate-delay, line-delay, segment-delay, path-delay), Functional Fault, Gate-delay Fault, Hyperactive Fault, Initialization Fault, Instruction Fault, etc.

The error is said to be the wrong output signal produced by a defective system. Hence there must be a mechanism to correct errors. As the transistor level faults are more dominant. The design of error detection and correction circuit is carried out at transistor level.

This paper aims at the design of the Hamming Code circuit. Also several low power techniques such as adiabatic logic, body biasing techniques are used for low power design. The optimized transistor level implementation of design especially for EXOR and AND gates is used.

This paper is organized as follows, section II deals with the single bit error detecting and correcting Hamming code circuit design and analysis. Section III deals with the low power techniques like adiabatic logic, body biasing techniques and the optimized transistor level of XOR and AND gates. Section IV details the designs considered for implementation. Section V details the results and the corresponding discussion, followed by conclusion.

II. HAMMING CODE

A. The Concept of hamming code for single bit error detection and correction

The general procedure for writing detection and correction of hamming code is explained here. Let us consider there are three parity bits and four data bits. We will be evident with the fact that based upon the parity bits we can explain weather it is

an even parity or an odd parity. In this paper we will be writing about (7,4) bit hamming code generation and correction. The code with data and parity will be in form of D3D2P4D1P2P1D0. Let us consider an example that the transmitted code is 0100110 and the received code 0110110. We can understand that the error is there near P4. Hence this error needs to be identified and detected. To check for error in even parity we use the formula

P1 = T1 + T3 + T5 => P1 = 1

P2 = T2 + T3+ T6 => P2 = 0

P4 = T4 + T5 + T6 => P4 =

1 Then for error correction

For even Parity

XOR (P1, 0, 1) = XOR (1 , 0 , 1) => C1 = 0

XOR (P2 , 0 , 0) = XOR(1 , 0 , 0) => C2 = 1

XOR (P1 , 0 , 1) = XOR(1 , 1 , 0) => C4 =

0 Now these are taken in as C4C2C1 =>

010

Hence we can tell that the error is there in T2 location.

There for the correct code must be "0100110" instead of "0110110". The T2 must be detected. For the detection and correction, the following analysis is used. But in order to design an circuit we follow the similar analysis.

B. Gate level hamming code Design

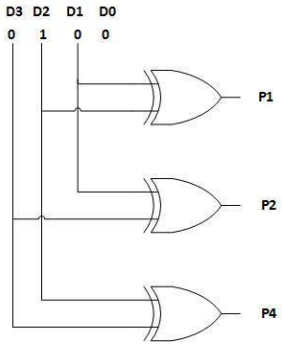

Figure 1: Parity Generation

Here the Di is the data input to the channel or the transmitted data.

T6 T5 T4 T3 T2 T1 T0

0 1 P4 0 P2 P1 0

Hence the parity is generated.

For the even parity of 4 bits we use the EXOR gate combination to find out the error parity.

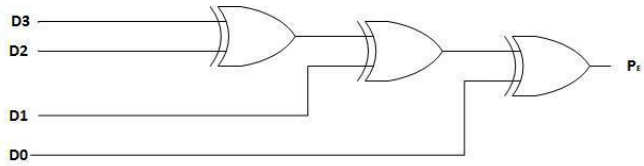

Figure 2: Parity generator for error detection

But the delays in the EXOR gates may lead to undesired outputs. To overcome this problem we go for Parallel Processing [10][11]. The parallel processing of the EXOR gate can be

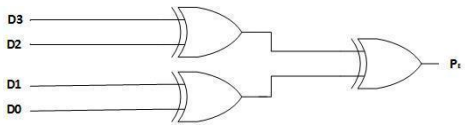

Figure 3: Parallel Processing for detection of Error Parity bit

Hence the desired output can be resulting due to the parallel processing.

As in 8 to 3 encoder D0 is not considered

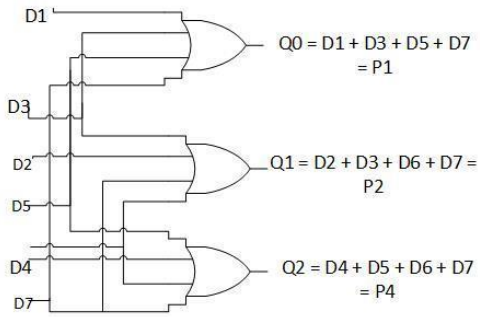

Figure 4: Showing Parity bits with position

Let us consider DT[7:1] as T[6:0]

i..e DT[7:1] = T[6:0]

Hence the detection block and correction block are designed.

Now this block needs to be designed by using the transistors.

III. LOW POWER DESIGN

The power dissipation can be from various sources like the static power from the supply rails, dynamic power dissipation due to the supply rail potential, load capacitance, frequency of the signals and/ or clock signal used. Also the leakage power due to the sub threshold currents in the device[12]. Among these the dynamic power dominates in micrometer technology but as the technology scales down, the leakage power dominates. Hence the various possibilities are devised in this paper. Several techniques can be used to obtain the low power design of a circuit. Among them reducing the number of transistors used can greatly reduce the area and the power dissipation. Here describe the two and three transistor structures of EXOR and AND gates with diagrams.

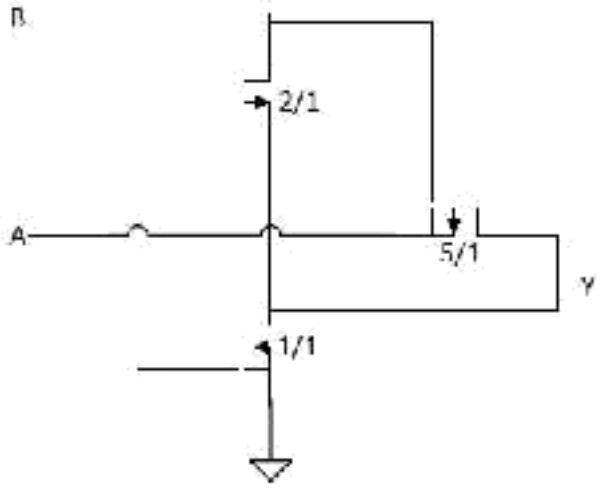

Figure 6: XOR gate using three transistors [13]

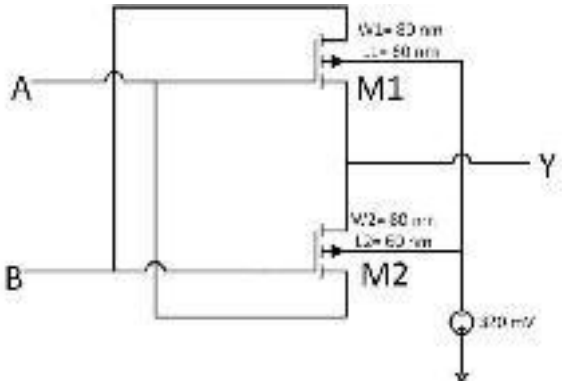

Figure 7: XOR gate using two transistors [13]

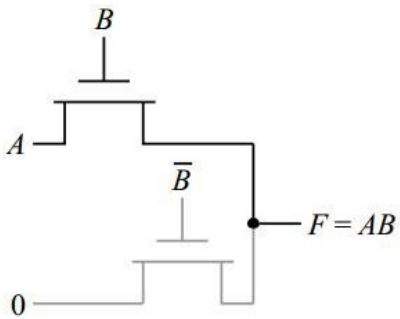

Figure 8: AND Gate using two transistors

Another possible method is to conserve the energy within the circuit by using the adiabatic logic. Here in order to conserve the energy of the circuit ac supply voltages are used[14].

Another possible method is to use the body biasing techniques for varying the threshold voltage of the transistors used in the design [15].

IV. DESIGNS DEVELOPED FOR IMPLEMENTATION

Several types of designs were used for the hamming code implementation. The Eight different designs were used in the implementation of the hamming code. Each of the design will be explained clearly here.

Design 1

In this design we have directly taken three transistors of 2 Input based XOR gate and two transistors of two input AND gates. While doing this design the Output logic level 1 was considered as 5V and Logic 0 level was considered as 0V. The time delay was 1.19E-13 sec. The power dissipation of the circuit was found to be 6.0361E-01mW. No low power technique was used in this circuit.

Design 2

In this design we have taken the second type of EXOR gate as discussed in this paper. This EXOR gate can be having two transistors and now the total design will be changed. While doing this design the output logic level for logic 1 is considered as 4.2V and logic level for logic 0 is considered as 800mV. Interesting observation here was that the power dissipation increased to 4.9553mW and the delay was reduced to 0.044887pS. The decrease in delay and increase in power dissipation is due to the fact that the design is simple and the W/L ratio of the design.

Design 3

The third design was based on the change in the design of the EXOR gate. It again used design of EXOR gate using 2 transistors. The Output logic level for logic 1 is considered as 5V and Logic 0 is considered as 1.2 V. The power dissipation of the design was 5.2769mW and the delay of the design was 4.9235E-13 Sec.

Design 4

The best of the above three designs is the Design 1. The low power technique called adiabatic logic was used here. For the design 1, Adiabatic Logic at 250Hz Sine Wave as bias. We can observe that the power dissipation has reduced to 0.61901mW and the delay was 0.11854pSec.

Design 5

The next design was done by using three transistors of 2-Input XOR gate and two transistors of 2-Input And gates. But the technique used here was the body biasing where gate and bulk are mixed together. The output logic level for Logic 1 was 4V and the Input logic level for Logic 0 was 457mV. The power dissipation for the following circuit is 3.3128 mW and the delay was 0.039388pSec.

Design 6

This design was based on the three transistor based 2-Input EXOR gate and two transistor based 2-Input AND gates with body biasing where source and bulk are mixed together. The output logic level 1 was degraded to 2V and Logic level 0 is 400mV. The power dissipation for the design was 0.43029mW.

Design 7

This design was based on the three transistor based 2-input EXOR gate and Two transistor based 2-Input AND gates with body biasing where drain and bulk are mixed together. The output logic level 1 for the design was 2.5V and Output logic level 0 was 0V. The power dissipation is 2.5mW and delay was 0.034448 pSec.

Design 8

The design was based on the Three transistor based 2-Input EXOR gate and Two transistor based 2-Input AND Gates with body biasing where VB = 140mV and source and bulk are mixed together. The output logic level 1 for the design was 5V and output logic level 0 was 0.8V. The power dissipation of the design was 0.55325mW and the delay was 0.12382pSec.

V. RESULTS AND DISCUSSION

The designs are developed at transistor level and are coded in SPICE i.e., Simulation Program with Integrated Circuit Emphasis. The functional verification of the designs is carried out in HSPICE Tools. The output waveforms are visualized in Avan waves interface. The fig.1 shows the output waveform of the design for the explained example of error detection and correction.

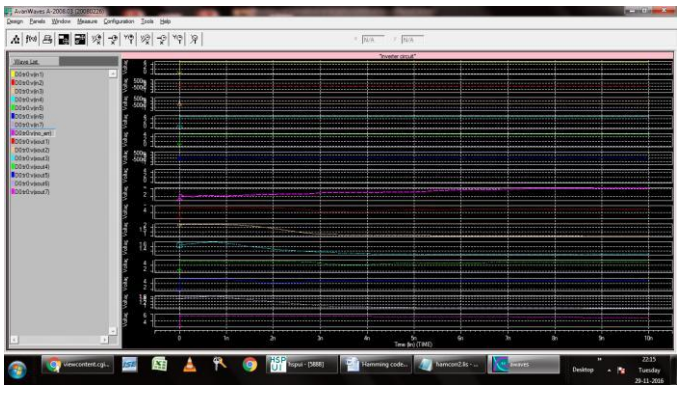

Figure 9 : Simulated output waveforms of the hamming code circuits

The Table 1 shows the comparison of the designs for the parameters like power dissipation, delay and power delay product. As the power delay product here corresponds to the Product of power dissipation along with delay, the least value will be preferred.

TABLE 1: COMPARISON OF PARAMETERS FOR DESIGNS DEVELOPED

Design	Power Dissipation (mW)	Delay (pS)	Power Delay Product (pWS)
Design 1	0.60361	0.11904	0.07185
Design 2	4.9553	0.044887	0.22242
Design 3	5.2769	0.49235	2.598
Design 4	0.61901	0.11854	0.07337
Design 5	3.3128	0.039388	0.1304
Design 6	0.43029	0.007902	0.0034
Design 7	2.5000	0.034448	0.08612
Design 8	0.55325	0.12382	0.0685

From Table 1, Design 6 has the least values of all of the compared parameters and is well suited for practical designs where online test is performed. Also here the scaling of the design is not performed which can be done as per the technology requirement.

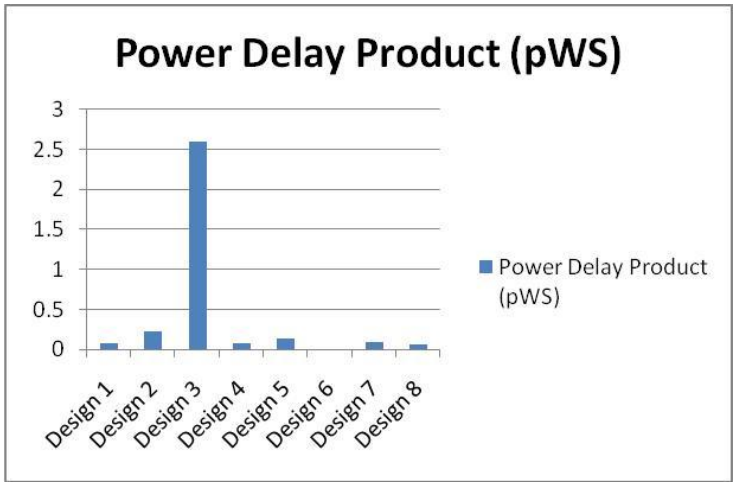

Figure 10: Graph representing PDP's of the designs.

Conclusion

Hamming codes are the simple single bit error-detecting and correcting proven to be effective from years together. The design of (7, 4) Hamming code is implemented using EXOR and AND gates. During practical implementation of transistor level circuits the area and corresponding power dissipation is a sober issue. By using optimized transistorized designs, these problems can be resolved. This paper totally emphasizes on reducing number of transistors. The comparison of the hamming code transistor models by using several methods of Designs are done. The low power technique i.e., adiabatic logic is added to the transistor to reduce the utilization of power. Eight designs are developed and evaluated for performance parameters like delay, power and area by using HSPICE Tool. Among them, Design 6 comprising of three transistor based 2-Input EXOR gate and two transistor based 2-Input AND gates with body biasing where source and bulk are mixed together proves to be a better option as it has corresponding power dissipation as 0.43029W and the least delay as 0.007902pS. Hence it is concluded as the best circuit for implementing the (7, 4) Hamming Code Design.

References

[1] J. Wang, S. Fang, and W. Feng, "New efficient designs for XOR and XNOR functions on the transistor level," IEEE J. Solid-State Circuits, vol. 29, no. 7, Jul. 1994, pp. 780–786.K.

[2] Roy, S. Prasad, "Low-Power CMOS VLSI Circuit Design", *Wiley India*, 2009.

[3] J. Rabaey, "Low Power Design Essentials", *Springer*, 2009.

[4] D. Radhakrishnan, "Low-voltage low-power CMOS full adder," in *Proc. IEEE Circuits Devices Syst.*, vol. 148, Feb. 2001, pp. 19-24.

[5] V. Adler and E. G. Friedman, "Delay and power expressions for a CMOS inverter driving a resistive-capacitive load," *Analog Integrat. Circuits Signal Process,* vol. 14, pp. 29–39, 1997.

[6] H. T. Bui, A. K. Al-Sheraidah, and Y.Wang, "New 4-transistor XOR and XNOR designs," in Proc. 2nd IEEE Asia Pacific Conf. ASICs, 2000, pp.25–28.

[7] A. M. Shams, T. K. Darwish, and M. A. Bayoumi, "Performance analysis of low-power 1-bit CMOS full adder cells," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 10, no. 1, Feb. 2002, pp. 20–29.

[8] Manoj Kumar, Sandeep K. Arya, Sujata Pandey, "Single bit full adder design using 8 transistors with novel 3 transistors XNOR gate," International Journal of VLSI Design & Communication Systems, Vol. 2, pp. 47-59, Dec. 2011.

[9] Yingtao Jiang Al-Sheraidah, A. Yuke Wang Sha, E. and Jin-Gyun Chung, "A novel multiplexer based low-power full adder," IEEE Transactions on Circuits and Systems: Express Briefs, vol. 51, no.7, pp.345-348, Jul. 2004.

[10] S. Goel. A. Kumar, M. A. Bayoumi, "Design of robust, energy –efficient full adders for deep sub-micrometre design using hybrid-CMOS logic style," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.14, no.12, pp.1309-1321, Dec. 2006.

[11] Avisek Sen, Partha Mitra, Debarshi Datta, "Low Power MAC Unit for DSP Processor", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-1, Issue-6, January 2013.

[12] Abdelgawad, A., "Low power multiply accumulate unit (MAC) for future Wireless Sensor Networks," Sensors Applications Symposium (SAS), 2013 IEEE, vol., no., pp.129, 132, 19-21 Feb. 2013.

[13] Shubhajit Roy Chowdhury, Aritra Banerjee, Aniruddha roy, and Hiranmay Saha, "A High Speed 8 Transistor Full Adder Design using Novel 3 Transistor XOR Gates", International Journal of Electronics, Circuits and Systems, WASET Fall, 2008.

[14] Amrita Oza, Poonam Kadam, "Techniques for Sub-threshold Leakage Reduction in Low Power CMOS Circuit Designs", International Journal of Computer Applications (0975 – 8887), Volume 97– No.15, July 2014.

[15] S. Goel. A. Kumar, M. A. Bayoumi, "Design of robust, energy –efficient full adders for deep sub-micrometre design using hybrid-CMOS logic style," IEEE Transactions on Very Large Scale Integration (VLSI)Systems, vol.14,no.12, pp.1309-1321, Dec. 2006.