The International Journal Of Engineering And Science (IJES)
|| Volume || 5 || Issue || 4 || Pages || PP -85- 89|| 2016 ||
ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805

15-bit NOVEL Hamming Codec using HSPICE 22nm CMOS Technology based on GDI Technique

[1]Raj Kumar Mistri, [2]Rahul Ranjan ,[3]Manohar Mahto, [4]Manish Jose Minz
[1,2]*Assistant Professor Department of Electronics & Communication Engg., RTCIT, Ranchi, India.*
[3,4]*B.tech scholar Department of Electronics & Communication Engg., RTCIT, Ranchi, India.*

----------**ABSTRACT**----------
GDI(Gate Diffusion Input) technique allows low power consumption, low propagation delay and also minimum number of transistor count (low chip area) for the logic design. in this paper 15-bit NOVEL hamming codec has been proposed. This Novel hamming codec has been simulated with HSPICE using 22nm CMOS technology with various design methodologies like TG technology, pass transistor logic and GDI Technique and designs are compared to 15-bit simple hamming codec with each of various design methodologies respectively. GDI technique provide excellent result in terms power consumption, chip area and propagation delay and also novel hamming codec provide less transistor count over general hamming codec.

Keywords *- GDI, VLSI, CMOS, Pass Transistors, TG.*

Date of Submission: 13 April 2016 Date of Accepted: 28 April 2016

I. INTRODUCTION

When data is transmitted from one location to another there is always the possibility that an error may occur. Hamming code is well known for its single-bit error detection & correction capability. Hamming code is a linear error-correcting code named after its inventor, Richard Hamming. Hamming codes can detect up to two simultaneous bit errors, and correct single-bit errors[1][2].

VLSI technology has played a development of portable digital applications, the applications demand for increasing the speed, implementation of a wide range of complex functions, less power dissipation. Mainly it is focusing on power consumption and area reduction of the digital circuits. Power is the disconcerting factor in portable applications. And another thing is area, it is directly possessions the cost and size of the devices. And also, to improve the performance of logic circuits based on the CMOS technology. The CMOS technology made by the complementary of a pull-up PMOS transistor and Pull-down NMOS transistor. MOSFET networks are the most predictable, but at circuit level an optimized design is desirable having less numbers of transistors, small power consumption and reduces delay [3].

Hamming code: it is error correction and error detection code. At a time it can correct 1-bit error. This code uses following algebraic formula to evaluate redundancy bit

$$2^k - k - 1 \geq n \quad \text{.....1}$$

Where 'n' represents no. of data bits and' r' represents no. of redundancy bit to be added with encoded data. Position of redundancy bit is 2^kth position in encoded bit string, where 'k' ranges from 0 to r-1.
For 15 bit data, No. of redundancy bit 'r' is =5. As 'k' ranges from 0 to r-1, so values of 'k' is 0,1,2,3 & 4 and position of redundancy bits are 2^0(1st), 2^1(2nd), 2^2(4th), 2^3(8th) & 2^4(16th) in encoded bit string.

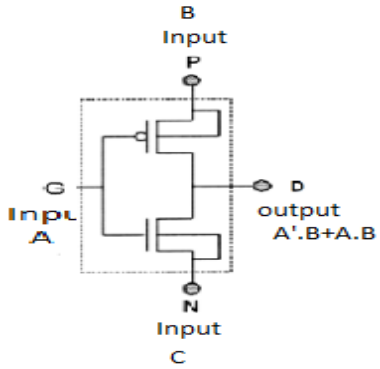
NOVEL hamming code: This is another NOVEL approach to correct 1-bit error at a time. This code uses following algebraic formula to evaluate redundancy bit

$$2^{r-1} - 1 \geq n \quad \text{.....2}$$

Where 'n' represents no. of data bits and' r' represents no. of redundancy bit to be added with encoded data. Position of redundancy bit is 'n+r' th position in encoded bit string, where 'r' ranges from 1 to 'r'.
For 15-bit data "111000111001101", hamming encoder provide "1110P0011100P110P1PP" & NOVEL Hamming encoder provide "PPPPP111000111001101" as encoded bit string, where P represents redundancy bit.

GDI Technique: Gate Diffusion Input, allows to implementation of a wide range of complex logic functions using only two transistors. The inputs are directly diffused into the gates PMOS, NMOS transistors. So, it is

called a gate diffusion input [4][5]. The GDI method is based on the use of a simple cell as shown in Fig. 1. Considering inputs A,B & C then its output function can be Output=A'.B+A.B.



The GDI cell contains four terminals, they are:
1. G: Common gate input of PMOS and NMOS
2. P: Input to the source/ drain of PMOS
3. N: Input to the source/drain of NMOS
4. Bulks of both PMOS and NMOS are connected to N or P.
This method is suitable for fast, the low power circuits reducing to the number transistors as compared to CMOS.

Fig.1 basic GDI cell

II. EVALUATION OF REDUNDANCY BIT

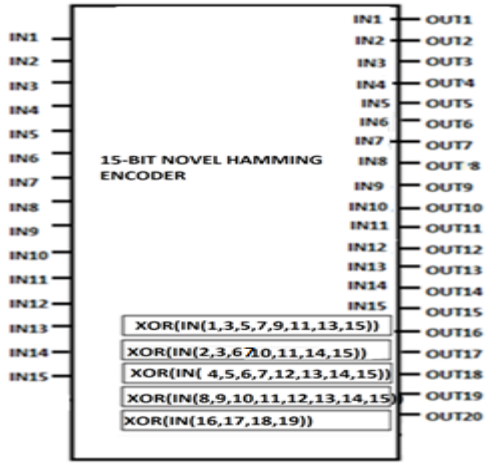
In hamming code Redundancy bit at position 2^k can be evaluated by XORing all data bits whose binary value of position contain bit '1' at its (k+1)th place. For example, for 15-bit data, number of redundancy bit require is 5, hence length of encode bit string should be 20-bit. To evaluate redundancy bit present at $2^3 \rightarrow (8^{th})$ position, following data bits are required whose position in encoded bit string are 9^{th},10^{th},11^{th},12^{th},13^{th},14^{th} and 15^{th} because binary value of all positions mentioned above contain logic '1' at its 4^{th} place. Hence P_8 =XOR[db(9), db(10), db(11), db(12), db(13), db(14), db(15)]. Here db(9) indicates data bit at 9^{th} position.

But in NOVEL hamming code, Redundancy bit present at (n+r)th position can be evaluated by XORing all data bits whose binary equivalent of position at 'r' place is logic '1'. For example, for 15-bit data, number of redundancy bit require is 5, hence length of encode bit string should be 20-bit. To evaluate redundancy bit present at (n+r=2)$\rightarrow$(17^{th}) position, following data bits are required whose position in encoded bit string are 2^{nd} , 3^{rd} ,6^{th} ,7^{th} ,10^{th} ,11^{th} ,14^{th} and 15^{th} because binary value of all positions mentioned above contain logic '1' at its 2^{nd} place. Hence P_2 =XOR[db(2), db(3), db(6), db(7), db(10), db(11), db(14),db(15)]. Here db(11) indicates data bit at 11^{th} position.

III. EVALUATION OF CHECK BIT

Check bit at position 2^k in hamming code can be evaluated by XORing all data bits whose binary value of position contain bit '1' at its (k+1)th place including parity bit. check bit can be evaluated for in above mentioned example at position 4^{th} as p_4=XOR[eb(4), eb(5), eb(6), eb(7), eb(12), eb(13), eb(14), eb(15)], where eb(4) represents encoded bit at 4^{th} position.

Fig.2 is showing bock diagram of 15-bit novel hamming encoder which takes 15-bit data as a input and provide 20-bit encoded bit string by adding 5 redundancy bit to it.



Check bit at position (n+r)th in NOVEL hamming code can be evaluated by XORing all data bits whose binary value of position contain bit '1' at its (r)th place including parity bit. check bit can be evaluated for in above mentioned example at position 4^{th} as p_4=XOR[eb(4), eb(5), eb(6), eb(7), eb(12), eb(13), eb(14), eb(15)], where eb(4) represents encoded bit at 4^{th} position.

Fig.2 Block Diagram15-bit novel hamming encoder

IV. ENCODER

Encoder is a logic design which is used to convert the data bit string into another larger bit string by adding redundancy bit to it. In novel hamming encoder redundancy bits are evaluated by the inequality $2^{r-1} - 1 \geq n$, here for 15-bit input data value of 'n' is 15, so above inequality will satisfy if 'r' is 5. Hence number of redundancy bit will be 5 that's why total output bits provided by novel encoder is 20. Among all encoded bits position of redundancy bits should be 16^{th},17^{th},18^{th},19^{th} and 20^{th}.

V. DECODER

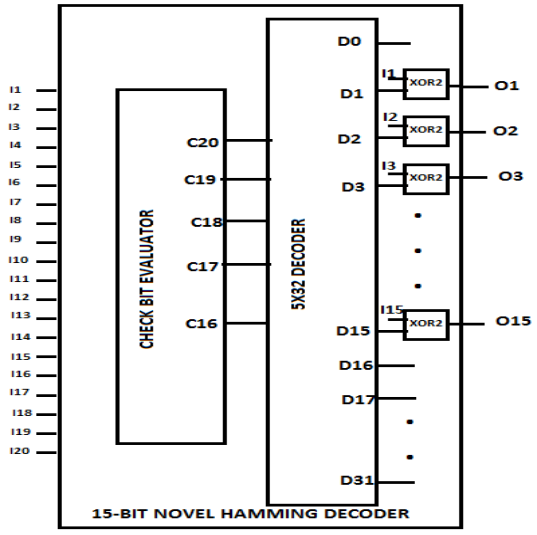


Fig.3 block diagram of 15-bit novel hamming decoder

Fig.3 represents block diagram of 15-bit novel hamming decoder. This novel hamming decoder contains three basic logic circuits. First one is check bit evaluator which contains 4, 8-input XOR gate and 1, 4-input XOR gate. Second one is 5x32 decoder and last one is 15, 2-input XOR gate. As check bits are given as a input to 5x32 decoder due to which it provides logic '1' at the its only one output corresponding to position of error bit and all other output remains logic '0'(if error occurs) otherwise all outputs (from D1 to D15) goes to logic '0'(if no error occurs).

Since all outputs (D1 to D15) connected to one input of XOR gate and other input to encoded data bit,so if no error occurs output of 5x32 decoder(D1 to D15) are to be '0' hence finally output of novel decoder will be all origital data and if error occurs that time one output of decoder (corresponding to the position of error bit) goes to'1' that is why error bit inverts and correction occurs.

VI. SIMULATION RESULT

Input waveform for hamming encoder is shown in Fig.4. In this waveform 4, 15-bit data is given to the input of encoder which are 010101101100100, 100000100100011, 011100101110000 & 101000011010010.

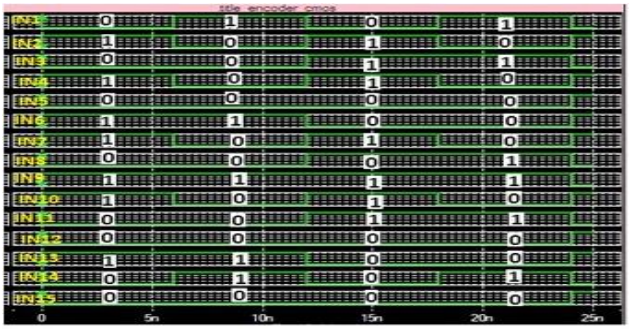


Fig.4 Input waveform of novel hamming Encoder

As we know that for n-input data, number of parity bits added to it is r & its value can be evaluated when inequality $2^{r-1} - 1 \geq n$ will satisfy. For 15-bit input data, at r=5, above inequality satisfy that is why number of parity bit required is 5. Hence the number of bits at the output of encoder is 20. From its output waveform (Fig.5) output bits are 01010110110010010010, 10000010010001110111, 01110010111000000011 and 10100001101001001100.

Fig.5 Output waveform of novel hamming Encoder

Fig.6 show the input waveform of novel hamming decoder. From waveform it is clear that four 20-bit data are given as a input & these are

1101011011001001001 0, 1000001001000111011 1,
01110001111000000011 and 00100001101001001100.

Bits inside the red rectangle indicate the error bit present at input side of decoder. Here Error bits are present at 1st ,7th , 7th & 1st positions respectively

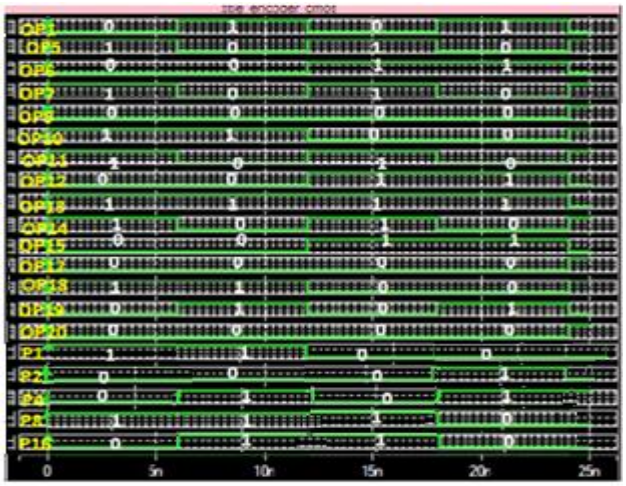


Fig.6 input waveform of novel hamming decode

Fig.7 indicate the output waveform of hamming decoder, in which error bit present at the input to decoder are corrected and data become original data which has given to the input of encoder. The original data from fig.7 are

0101011011001 00, 1000000001000 11, 0111001011 10000
& 1010000110100 10.

Bits inside the green rectangle indicate the corrected bit by hamming decoder at output side of it.

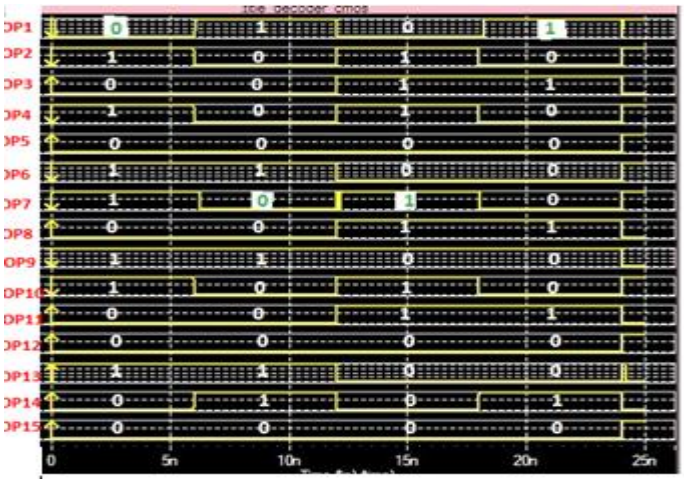


Fig.7 Output waveform of novel hamming decoder

VII. EXPERIMENTAL RESULT

Experimental result provides the information about number of transistor usage by different technologies (Pass Transistor, TG, GDI. In this paper hamming codec and novel hamming codec was designed by various technologies like TG tech, pass transistor and GDI tech and detail of transistor usage has described in table.1.

NO. OF DATA BIT	TECHNOLOGY	HAMMING ENCODER NO. OF TRANSISTOR COUNT	NOVEL HAMMING ENCODER NO. OF TRANSISTOR COUNT
7 bit	TG TECH	104	88
	PASS TRANSISTOR	78	66
	GDI TECH	52	44
15 bit	TG TECH	256	248
	PASS TRANSISTOR	192	186
	GDI TECH	128	124
31 bit	TG TECH	648	632
	PASS TRANSISTOR	486	474
	GDI TECH	324	316
63 bit	TG TECH	1536	1528
	PASS TRANSISTOR	1152	1146
	GDI TECH	768	764

In order to design 15-bit hamming codec by various technologies like TG, pass transistor and GDI tech, it requires 256, 192 and 128 nos. of transistor respectively and 248, 186 and 124 nos. of transistor for novel hamming codec. Hence novel hamming codec saves 3.12% of area over hamming codec in each of design technologies.

Table.1 transistor count by different technologies

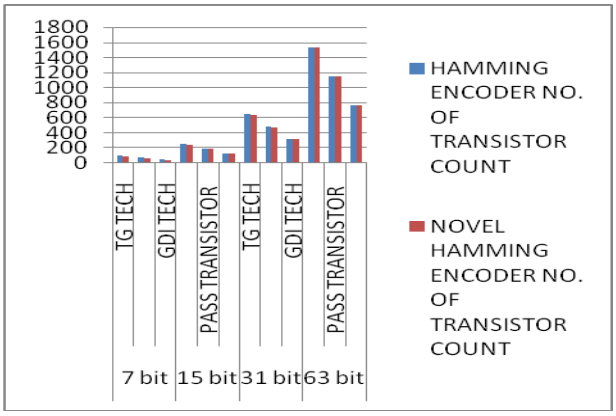


Fig.8 graphical representation of transistor usage by different technologies

REFERENCES

[1] Data communication and networking , Behrouz A.Forouzan , 2nd edition Tata McGraw Hill publication.
[2] Communication Networks, available at: http://www.pragsoft.com/books/CommNetwork.pdf 2001.
[3] Gaddam Naga Durga, D.V.A.N Ravi Kumar, "GDI Technique Based Carry Look Ahead Adder Design ", IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)Volume 4, Issue 6, Ver. I (Nov - Dec. 2014,PP 01-09e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197.
[4]. A.P. Chandrakasan and, S.Sheng, and R.W. Brodersen "Low Power CMOS digital Design," Proc. IEEE J. Solid –State circuits,vol.27, pp. 473-484, Apr.1992.
[5]. W. Al-Assadi, A.P. Jayasumana, and Y.K.Malaiya, " Pass Transistor logic design," Int. J. Electron, vol.70,pp.739 -749,1991