FPGA IMPLEMENTATION OF 4D-PARITY BASED DATA CODING TECHNIQUE

Vijay Tawar[1], Rajani Gupta[2]

[1]*Student, KNPCST, Hoshangabad Road, Misrod, Bhopal, Pin no.462047*
[2]*Head of Department (EC), KNPCST, Hoshangabad Road, Misrod, Bhopal, Pin no.462047*

Abstract

The scenario of communication systems is changing with the immense development of electronic devices and the increasing demand of data in communication systems. When the communicating data follows the wireless path in its communication channel then the presence of the electromagnetic signals in the communication media can effects the integral value of the signal and may lead to the wrong reception of the data at the receiver side. This wrong reception is further reflected as the wrong processing data to generate the undesired system output. So, the communication systems are required to have some hardware or software resources to detect the effects of the environmental electromagnetic signals, called noise. A number of techniques have been proposed to estimate the effect of noise as well as the actual value of the transmitted data. One of the simplest methods used to identify a change in the transmitted digital data stream is by adding even or odd parity bit with the transmitted data and checking verifying its authenticity at the receiver end after receiving the data. A single bit parity scheme only informs about an odd bit change in the transmitted data. One of the implementation methods of using the similar concept for noise effect identification and correction is by using 4-dimensional parity method. This method involves generation of parity bits of a pre-accumulated data in four directions. Each of the data bit is used to generate 4-parity bits. The involved method can be effectively used to identify and correct 3-bit of error in the transmitted data with the help of the parity bits. In the proposed work the same error detection and correction method is simulated on a field programmable gate array device using Xilinx ISim Tool.

Keywords: *4-D Parity, Dynamic Power, Error Detection and Correction (EDAC) Code, Even Parity, Field Programmable Gate Array (FPGA), Horizontal-Vertical-Diagonal (HVD), Odd Parity, Xilinx.*

----------***----------

1. INTRODUCTION

In the computing systems the complete processing is performed on the binary data. The result and the performance of the system can be effectively utilized only if the data that is processed by the system is correct. So the data authenticity becomes an important issue in a computing system apart from its actual operational hardware. But in the cases when the disturbance in the incoming data bits is random in nature, it is very difficult to select the best estimation of the actual data. Many methods have been suggested, based on different algorithms, to come out as a solution of this error. The main cause of such errors is the electromagnetic radiations that are present in the communication channel or in the environment. These signals being of the same nature, i.e., electromagnetic nature, are interference by each other and the resultant is received by the system. Now the task that is left over the receiver system is to separate the unnecessary electromagnetic signals from the actual data signals. When the received signals are converted into binary signals using the intermittent hardware the effect of the electromagnetic radiations can be diagnosed with the help of binary signal hardware. If the data transmission be initiated with some redundant bits that are generated using the data bits only then a method could be used to detect the reception of correct data using binary signal processing of the received data. This again would be effective only if the number of changes, i.e., amount of noise, introduced by the interference signal in the actual signal can be estimated. The amount of noise can be represented in terms of the number of transmitted signal bits changed by the noise signal. A single bit change is also referred by many scholars as Single Event Upset (SEU) and a multiple bit change as Multiple Bit Upset (MBU). Since the detection of the upset in the data is not only the effective solution of the system issue, so methods are also developed to introduce the correction at the particular upset data bit positions. The realization of such methods is possible at both hardware and software level. In the systems with cost constraint a software approach is used for implementation of such circuits whereas in the systems that require faster performance preference is given to hardware based circuit implementation. In the proposed work, a method is simulated that involves error detection as well as error correction of the data using four directional parities. The data is arranged in the form of a matrix. The parity bits are generated for the data bits in line with the four matrix directions i.e., (i) horizontal, (ii) vertical, (iii) forward-slash diagonal, and, (iv) backward-slash diagonal. The generated parity bits are accumulated with proper separation in the data frame for transmission. The receiver circuit re-generates the parities and performs the comparison of the received and the generated parity for detecting the error and utilizes the error correction hardware to complete its operation. The proposed work is synthesized on Xilinx Field Programmable Gate Array Device [1]. The paper is

organized as: The previous work performed by researchers and scholars is given in section-2. The working of the proposed method of error detection and correction is given in section-3. Simulation and synthesis results are given in section-4. Conclusion and future scope is mentioned in section-5.

2. LITERATURE REVIEW

The basic method of error identification in a pre-defined data stream is parity bit generation and checking. The parity bit can be an odd parity bit or an even parity bit. A single bit addition of parity bit is capable of only identifying an odd number of bit changes in a bit stream. An even number of changes in the data stream appears as correct data using single bit parity. This is the driving force for the development of methods to either use multiple parity bits or switch to any other algorithm of data encoding. Multi-dimensional parity schemes are proposed in [2, 3 and 4] with multi bit error detection and correction capabilities. These schemes can effectively utilize in four to five bit error detection hardware. A 3-bit burst error correction method is proposed in [5]. A software decision based scheme using 4-dimensional parity coding is proposed in [6, 7 and 8]. A hardware and time redundancy based error detection and correcting circuit with reduced number of inputs and outputs in combinational circuit is proposed in [9 and 10]. These designs are implemented for semiconductor memory protection against the erroneous data. A similar scheme is proposed in [11] for satellite applications.

3. PROPOSED HVD ENCODING DECODING SCHEME

The proposed multi-directional parity code is an encoding method that involves parity generation in four directions, viz., horizontal, vertical, forward-slash and backward-slash. The scheme of parity generation in the four directions is shown in Figure-1. In this method the data is first arranged in to the form of a matrix as show in the figure. The matrix arrangement of a fixed length of data ensures such an arrangement that enables the realization of four directions in the data block. Table-1 shows the data bits in a block arrangement of matrix 8X8. Table-2 shows the parity bits corresponding to the four directions.

		BD7	**BD6**	**BD5**	**BD4**	**BD3**	**BD2**	**BD1**	**BD0**			
		BD8										
		BD9		D07	D06	D05	D04	D03	D02	D01	D00	**H0**
		BD10		D17	D16	D15	D14	D13	D12	D11	D10	**H1**
		BD11		D27	D26	D25	D24	D23	D22	D21	D20	**H2**
		BD12		D37	D36	D35	D34	D33	D32	D31	D30	**H3**
		BD13		D47	D46	D45	D44	D43	D42	D41	D40	**H4**
SD0		**BD14**		D57	D56	D55	D54	D53	D52	D51	D50	**H5**
SD1		**BD15**		D67	D66	D65	D64	D63	D62	D61	D60	**H6**
SD2				D77	D76	D76	D74	D73	D72	D71	D70	**H7**
SD3												
SD4												
SD5				**V7**	**V6**	**V5**	**V4**	**V3**	**V2**	**V1**	**V0**	
SD6												
SD7	**SD8**	**SD9**	**SD10**	**SD11**	**SD12**	**SD13**	**SD14**	**SD15**				

Figure 1: A matrix of 8X8 Data Bits for HVD Coding

Table 1: List of the Row-wise data bits in a data block of matrix size 8X8

Row	Data Bits (MSB…LSB)
1	D07-D06-D05-D04-D03-D02-D01-D00
2	D17-D16-D15-D14-D13-D12-D11-D10
3	D27-D26-D25-D24-D23-D22-D21-D20
4	D37-D36-D35-D34-D33-D32-D31-D30
5	D47-D46-D45-D44-D43-D42-D41-D40
6	D57-D56-D55-D54-D53-D52-D51-D50
7	D67-D66-D65-D64-D63-D62-D61-D60
8	D77-D76-D75-D74-D73-D72-D71-D70

Table 2: List of the Horizontal Vertical and Diagonal Parity Bits

List of Parity Bits in Horizontal-Vertical-Diagonal Generation			
Horizontal	**Vertical**	**Slash Diagonal**	**Back-Slash Diagonal**
		SD0	BD0
		SD1	BD1
		SD2	BD2
		SD3	BD3
H0	V0	SD4	BD4
H1	V1	SD5	BD5
H2	V2	SD6	BD6
H3	V3	SD7	BD7
H4	V4	SD8	BD8
H5	V5	SD9	BD9
H6	V6	SD10	BD10
H7	V7	SD11	BD11
		SD12	BD12
		SD13	BD13
		SD14	BD14
		SD15	BD15

The parity generation is performed using logic XOR operation of all the data bits of the group of bits corresponding bits of a particular row or column or diagonal. The encoded data is the combined data that has the initial bits and the parity bits. The parity bits are also called as redundant data. The block diagram of the proposed encoder is shown in Figure-2. The encoder is a synchronous logic that generates parity bits using independent parity generator blocks. The clock synchronization operation and the data transfer are controlled using the transmitter control block. The control block has a master synchronous reset control input to clear data of all registers. The decoder logic receives the data bits as well as the encoded parity bits at the receiver side of the system. The decoder circuit re-generates the parity bits from the received data bits using the same logic that is used by the transmitter block. A synchronous control is used by the decoder circuit to perform this operation.

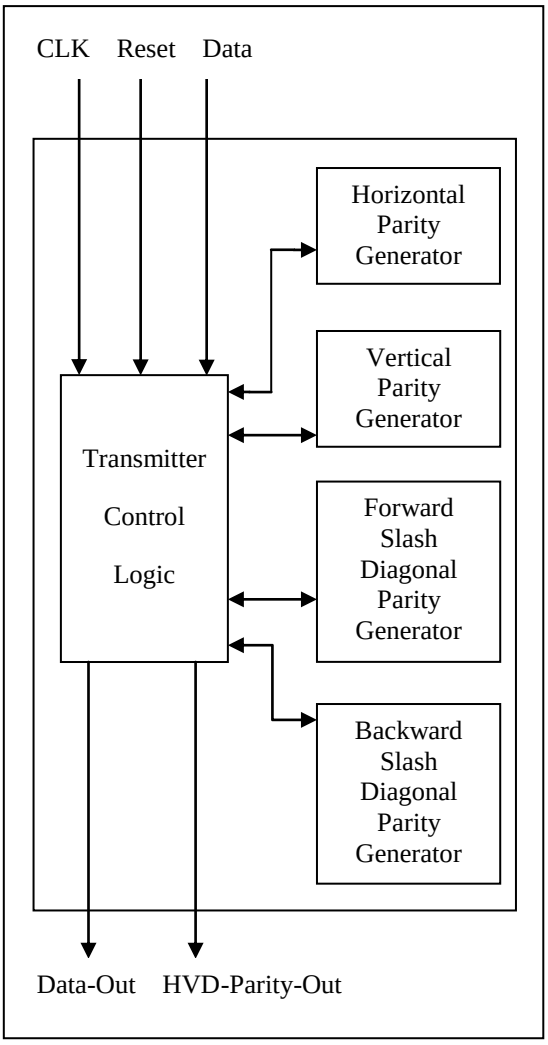

Fig 2: Block Diagram of proposed 4-D HVD Encoder

The block diagram of the proposed decoder scheme is shown in Figure-3. The control block, after re-generating the parity bits from the received data, controls the received parity bits with the re-generated bits. If the two sets of parity bits are equal then the data is not considered for correction and the data is transferred to the block output register. But, in case if the two sets of parity bits are not equal then the data is introduced bit log correction based on the logic values of the received data and the parity bits that are not found equal in the comparator block operation. The three blocks: (i) HVD Parity Generator, (ii) Parity Comparator Logic, and (iii) Parity Correction Logic, are serially synchronized in the decoder with the help of control block. The control block in decoder also has a master synchronous reset input to clear data of all registers. In case when an un-equality in the parity bits is indicated by the parity comparator logic block, an "error indicator" output is set high.

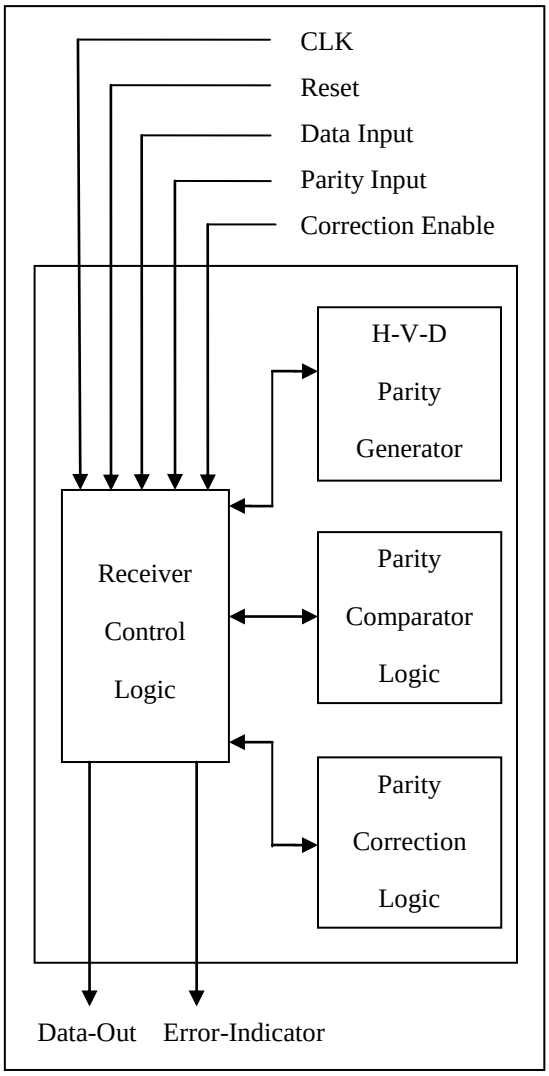

Fig 3: Block Diagram of proposed HVD Decoder

4. SIMULATION AND SYNTHESIS RESULT

The proposed design is simulated on VHDL Language environment using Xilinx ISE Tool. The test-bench waveform result of simulation of the encoder design is shown in Figure-4. The row-wise binary data input of the receiver and the generated parity output of the receiver are as follows:

Row-1 (r1) = 10011011
Row-2 (r2) = 01100110
Row-3 (r3) = 10101010
Row-4 (r4) = 11000111
Row-5 (r5) = 10000001
Row-6 (r6) = 01011010
Row-7 (r7) = 01000011
Row-8 (r8) = 11100100
Horizontal = 01001001
Vertical = 11101100
Forward-slash Diagonal = 110010101010111
Back-slash Diagonal = 010011000001001

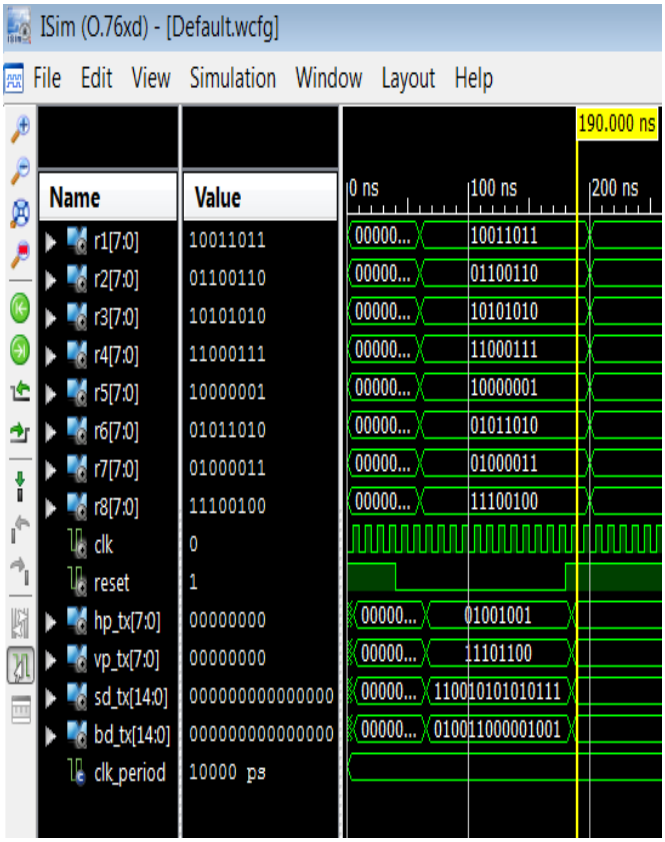

Fig 4: Proposed Encoder Simulation Output

Figure-5 and Figure-6 respectively show the waveform output of the decoder simulation in the cases: (i) "NO-error" in the received data, and (ii) 4-bit error correction by the decoder in the received data. The simulation inputs of Decoder in these conditions are shown in Table-3:

Table 3: Simulation Inputs of Proposed Decoder

Decoder Input	Case (i)	Case (ii)
Row-1 (r1)	10011011	10011011
Row-2 (r2)	01100110	01100110
Row-3 (r3)	10101010	10101010
Row-4 (r4)	11000111	11000111
Row-5 (r5)	10000001	10000001
Row-6 (r6)	01011010	01011010
Row-7 (r7)	01000011	01 **1111** 11
Row-8 (r8)	11100100	11100100

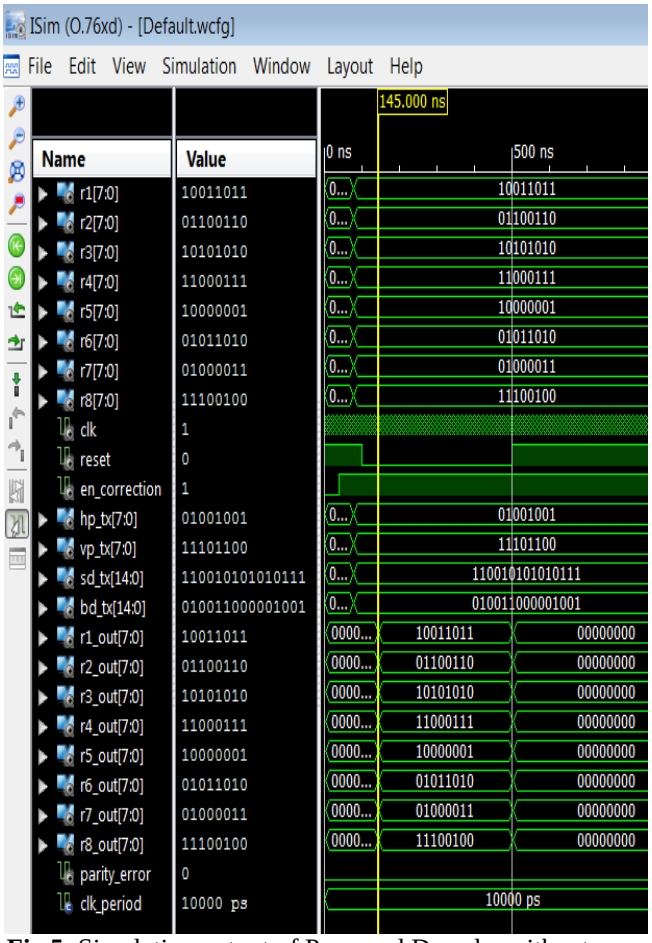

Fig 5: Simulation output of Proposed Decoder without error in received data

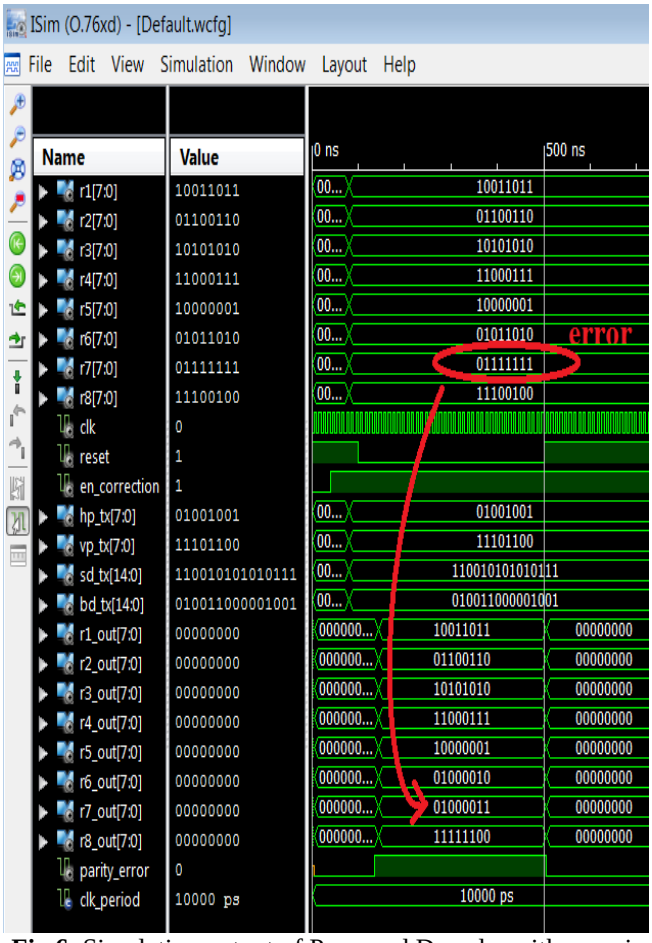

Fig 6: Simulation output of Proposed Decoder with error in received data

The hardware synthesis of the proposed design is performed using Xilinx Synthesis Tool on Xilinx Spartan 3E XC3S500E-4FG320 FPGA device.

Table 4: Hardware Utilization of proposed Encoder Design

Hardware Resource	Total	Encoder	
		Used	%
Slices	4656	47	1
Flipflops	9312	68	1
LUTs	9312	90	1

The hardware utilization summary of the Encoder and Decoder designs are presented in Table-3 and Table-4 respectively.

Table 5: Hardware Utilization of proposed Decoder Design

. Hardware Resource	Total	Decoder	
		Used	%
Slices	4656	227	4
Flipflops	9312	184	1
LUTs	9312	329	3

5. CONCLUSION AND FUTURE WORK

In this paper a method is proposed for error correction and detection. The scheme detects and corrects all the continuous errors up to 3-bits and multiple errors from the data block with length less than 4-bit. Some cases 4-bit errors are also within the correction capability of the proposed method. In the future, the proposed work can be extended by performing simulations for detecting and correcting a higher number of continuous error bits. The proposed work could be used in hybrid with the other proposed techniques of error detection and correction as an improved performance attempt.

ACKNOWLEDGEMENTS

The author wish to thank Mr. Shailesh Raghuwanshi (Principal, KNPCST, RGTU, Bhopal) and Mr. Piyush Jain (Innovative Technology Design and Training Center, Bhopal) for sharing their views in line with the present work.

REFERENCES

[1] www.xilinx.com.

[2] E. Yaakobi and T. Etzion, "High dimensional error-correcting codes", IEEE International Symposium on proceedings in Information Theory, 2010.

[3] T. F. Wong and J.M. Shea, "Multi-dimensional parity check codes for bursty channels", IEEE International Symposium on proceedings in Information Theory, 2001.

[4] M. Rubinoff, "N-dimensional codes for detecting and correcting multiple errors", Communications of the ACM, Volume-4, Number-12, pp. 545-551, 1961.

[5] N.B. Anne, U. Thirunavukkarasu, and S. Latifi, "", International Conference on Proceedings in Information Technology: Coding and Computing (ITCC), 2004.

[6] M. Imran, Z. Al-Ars and G. N. Gaydadjiev, "Improving Soft Error correction capability of 4-Dimensional parity codes", 14th IEEE European Test Symposium, 2009.

[7] M. Kishani, H.R. Zarandi, H. Pedram, A. Tajary, M. Raji and B. Ghavami, "HVD: Horizontal-Vertical-Diagonal error detecting and correcting code to protect against the soft errors", Design Automation for Embedded Systems, Volume-15, No 3-4, 2011.

[8] S. Sharma and P. Vijayakumar, "", International conference on Proceedings in Devices, Circuits and Systems (ICDCS), 2012.

[9] Fernanda Lima, Luigi Carro, Ricardo Reis, "Designing Fault Tolerant Systems into SRAM-based FPGAs", DAC Anaheim, California, June 2-6, 2003.

[10] C. Argyrides, H. R. Zarndi, D K Pradhan, "Multiple upsets tolerance in SRAM Memory", International Symposium on Circuits and Systems, New Orleans, 2007.

[11] Y. Bentoutou, "Program Memories Error Detection and Correction On-board Earth Observation Satellites", World Academy of Science, Engineering and Technology, 2010.