Module Design Approach of Hamming Code Using Advanced Verilog Concept of FPGA

G. AKHILA[1], V. PRADEEP KUMAR[2]
[1]PG Scholar, Dept of VLSI System Design, Bharat Institute of Engineering and Technology, India,
[2]Assistant Professor, Dept of VLSI System Design, Bharat Institute of Engineering and Technology, India,
.

Abstract: This paper describes Improved Hamming Code, At whatever point data is stored or transmitted, some chance one or more bits will "flip" i.e., will change to an incorrect value. Such incorrect values are called errors; they may be because of a changeless shortcoming (broken hardware) or a transient condition. To neutralize this issue and guarantee dependable operation, error correcting codes (ECC) are utilized. Additional bits are sent or stored close by the data bits to give redundant data. With enough bits of deliberately picked redundant data, we can detect or correct the most likely classes of errors. Hamming code error correction is most generally utilized for computer memories. Hamming code with additional parity/redundancy bit can detect and correct single-bit errors and detect two bit errors. Hamming code is normally utilized for transmission of data with little lengths. Scaling it for bigger data lengths, results in a ton of overhead because of interspersing the redundancy bits and their evacuation later. Improved hamming code strategy is exceptionally adaptable without such overhead. Accordingly it is suitable for transmission of huge size data bit-streams with much lower overhead bits per data bit ratio. The project's objective is to design an error correction core utilizing improved hamming code. Hamming code with extra parity bit can detect and correct single-bit errors and detect two bit errors. The error correction core design endeavored in the paper utilizes improved hamming code error correction strategy. This strategy can detect and correct single-bit errors. In traditional hamming code strategy, extensive quantities of overhead bits are utilized as a part of the procedure of computation of parity/redundancy bits. In improved hamming code system the quantity of overhead bits is significantly decreased. The parity bits are annexed toward the end of data bits. This wipes out the overhead of interspersing the redundancy bits at the sender end and their evacuation at the receiver end. This work is accepted to serve as a decent error correction system for transmission of substantial size data bit-streams the length of there is probability of at the most single-bit error amid transmission.

Keywords*:* Hamming Code, Error Correction.

I. INTRODUCTION

Hamming code is an error correction code that is used in the process of detecting single and 2-bit errors and corrects the single-bit error that may happen when binary data is propagated along the way from one unit into another [2]. Showing the design and development of (11, 7, 1) Hamming code utilizing Verilog hardware description language (HDL). Here, "11" compares to the aggregate number of Hamming code bits in a transmittable unit containing data bits and excess redundancy bits, 7 is the quantity of data bits while "1" signifies the most extreme number of error bits in the transmittable unit. This code fits well into little “field-programmable gate array (FPGAs), application-specific integrated circuits (ASICs) and complex programmable logic devices (CPLDs)” and is preferably suited into communication applications that need error-control [1].

A. The Technique

Utilization of basic parity permits in detecting the single-bit error in a received message. Correcting these obtained errors requires some extra data, since the bit location of the erroneous bit must be recognized in the event that it is to be corrected. (On the off chance that an erroneous bit can be found on the location and it can be re-corrected to original by simply changing inversely or flipping its bit value.) Correction is unrealistic with one parity bit subsequent to any bit error in any bit location creates the very same data, i.e., an error. In the event that more number of bits are incorporated in a source data message, and if that given bits can be organized such that diverse erroneous bits produces distinctive error results, then erroneous bits could be recognized. The forward error correction (FEC) in the digital communication facility systems, especially those utilized as a part of military, need to perform precisely and dependably even in the vicinity of noise and interference. Among numerous conceivable approaches to accomplish this objective, forward error-correction coding is the best and efficient. The forward error correction systemic coding (additionally called 'channel coding') is a kind of digital signal processing so as to handle that enhances dependability of the data a known structure into the data arrangement before transmission. This structure empowers the accepting system that is used to detect and perhaps correct errors brought on by debasement from the provided data channel and the applied receiver at the other end. As this systems name suggests, this employed coding procedure empowers the decoder to correct the found errors by not taking a chance of asking for re-transmission of the original data. Hamming code is a run of the mill illustration of forward error correction. In any event related to the communication system that utilizes forward error correction coding, in the first stage the digital data source sends a message data succession to a systematic data encoder unit. The encoder embeds redundant bits (or parity bits), consequently yielding a more drawn out succession of code bits, which is called a 'code word.' These coded bits in the formation of words can be later propagated to a recipient, which utilizes a proper related decoder to extricate the original message data sequence on the go.

II. METHODOLOGY

The bellow Fig.1 shows the detailed block diagram of Improved Hamming code system. It is composed of 5 stages as shown, such as Transmitter, Encoder, Noise Box, Decoder, and Receiver. The major role for the operation is given to the encoder on the source side and to the decoder on the receiver end. The noise box is introduced only for the demonstration purpose to induce the error assuming it to be in the transmission path. It has a random sequence generation, which in turn determines the position of the bit to be flipped. In improved Hamming code, the redundancy bits are put toward the data's end bits. This significantly decreases considerable measure of overhead because of interspersing the redundancy bits and their evacuation later at the receiver end. Further the quantity of overhead bits included during the process of calculating redundancy bits is less [1]. Therefore this improved hamming code strategy can be used for the transmission of huge size message data bit streams. But the overhead in this strategy is much lower on bits per data bit ratio comparatively.

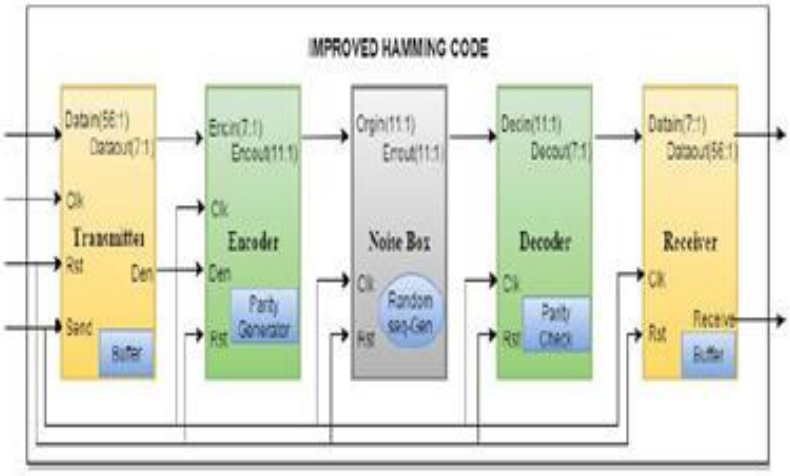

Fig.1. Block diagram of Improved Hamming code system.

III. HAMMING CODE

Hamming code are the linear block code which are invented by Richard. W. Hamming. They are an improvement over simple parity code method. Hamming codes are valid only when the hamming distance between the bits is less then or equal to one. By contrast, the simple parity code cannot correct errors, and can only detect an odd number of errors. They are the type of binary codes. The idea of hamming distance is the central concept in coding the error control. The hamming distance between the two words (of the same size) is the number of differences between the corresponding bits. The hamming distance can easily be found if we apply the X or operation on the two words and count the number of 1s in the result. The hamming distance is a value always greater then zero. If we find the hamming distance between any two words it will be the result of the Xo ring of the two bits. Like the hamming distance between d(000,011) is 2 because 000 xor 011 is 011(two 1s) and the hamming distance between d(10101,11110) is 3 because 10101 xor 11110 is 01011(three 1s).[3][7]Hamming code method works only on two methods(even parity, odd parity) for generating red undancy bits. The number of redundancy bits are generated using a formula. The number of redundancy depends on the number of information data bits. The formula is:

$$2^r = D+r+1 \quad \textbf{(1)}$$

Here, r = number of redundancy bits
D = number of information data bit.

P[0] is having value "1" at bit positions 1, 3, 5, 7 and 9. Consequently P[0] is chosen such that there is even parity at these locations (XXX1 <= 10) [1]. P[1] is chosen such that there is even parity at locations 2, 3, 6, 7 and 10 (XX1X <= 10). P[2] is chosen such that there is even parity at locations 4, 5, 6 and 7 (X1XX <= 10). P[3] is chosen such that there is even parity at locations 8, 9 and 10 (1XXX <= 10). P[4] is chosen such that there is even parity at the bit positions of redundancy bits shown in Fig.2.

Bit position of data	P[3]	P[2]	P[1]	P[0]
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0

Fig.2. Truth table of erroneous bit position.

P[3:0]. P[5] is chosen such that there is even parity in all the bit locations including the redundancy bits P[4:0]. These parity bits are scattered in positions 15, 14, 13, 12, 11 and 10. For the computation of parity bits, even parity checks were per-shaped on 5, 5, 4, 3, 4 & 16 bits respectively. In this way a sum of 37 bits are included during the process of hamming bits computation. The code word format for an example data 10'b1100110011 is appeared in Fig.3. Parity bits are appeared in bold format.

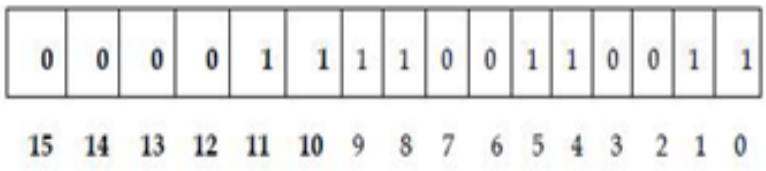

Fig.3. Code word format for sample data.

This code word shown is transmitted or stored in the memory. At the receiving end, the parity bits are uprooted. A parity check is performed between the transmitted parity and parity that is in the received code word. The result of examination decides the way of error. On the off chance that single bit error has been happened, then a mask will be produced and the data will be corrected. The error-detection and correction process in hamming code is as outlined in Fig.4.

Received information including Hamming bits	Status of parity check	Conclusion
0000111100110011	000000	No Error
0000111100110111	100011	Single Error at position 2
0000111000110011	101010	Single Error at position 9

Fig.4. Error detection and correction using improved hamming Code.

IV.VERILOG LANGUAGE

Verilog is one fo the two most common Hardware Description languages(HDL) used by integrated circuits (IC)designers. The other one is VHDL. Verilog can be used to describe designs at four levels of abstraction. They are the algorithmic level, the register transfer level, the gate level and the switch level.[6]

A. Using Verilog For Simulation

The basic structure of Verilog in which all hardware components and test benches are described is called a module. Language constructs, in accordance to Verilog syntax and semantics form the inside of a module. These constructs are designed to facilitate the description of hardware components for simulation, synthesis, and specification of test benches to specify test data and monitor circuit responses. A module that encloses a design's description can be described to test the module under design, in which case it is regarded as the test bench of the design.

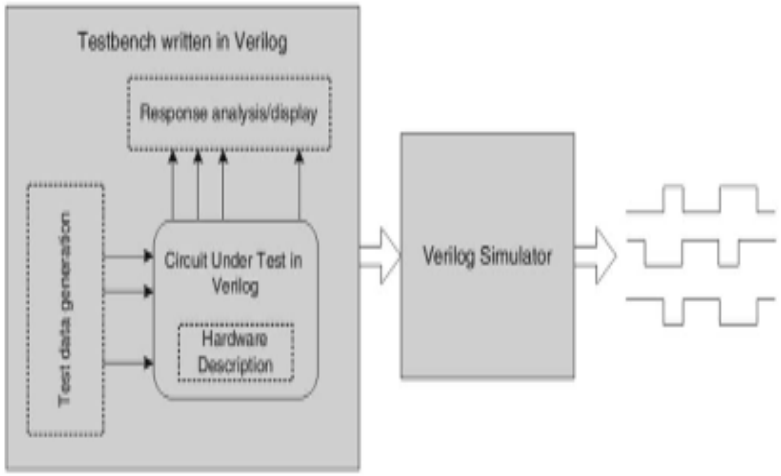

Fig.5. Simulation in verilog.

The above Fig.5 consists of a design with a verilog test bench. Verilog constructs (shown by dotted lines) of the Verilog model being tested are responsible for the description of its hardware, while language constructs used in a test bench are in charge of providing appropriate input data or applying data stored in a text file to the module being tested, and analysis or display of its outputs. Simulation output is generated in the form of a waveform for visual inspection or data files for record or for machine readability. [5][6]

B. Using Verilog For Synthesis

After a design passes basic the functional validations, it must be synthesized into a netlist of components of a target library. The target library is the specification of the hardware that the design is being synthesized to. Verilog constructs used in the Verilog description of a design for its verification or those for timing checks and timing specifications are not synthesizable. A verilog design that is to be synthesized must use language constructs that have a clear hardware correspondence. The Fig.6 below shows a block diagram specifying the synthesis process. Circuit being synthesized and specification of the target library are the inputs of a synthesis tool. The outputs of synthesis are a net list of components of the target library, and timing specification and other physical details of the synthesized design. Often synthesis tools have an option to generate this net list in Verilog.

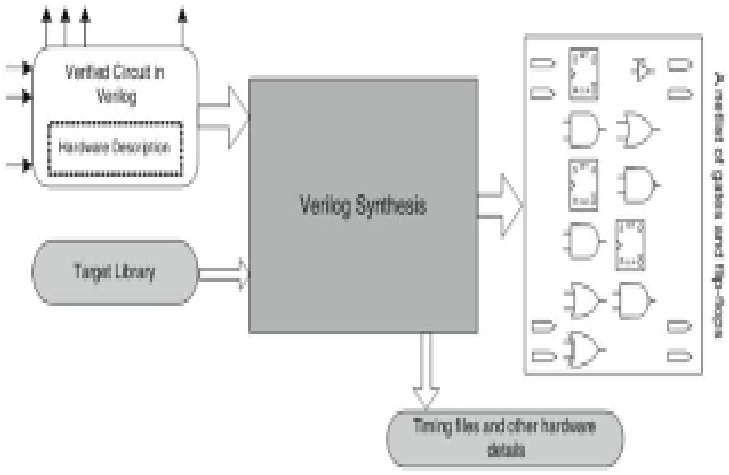

Fig.6. synthesis of a verilog design.

V. RESULTS AND DISCUSSIONS

The following are the results obtained for encoder and decoder using MODELSIM as shown in Fig.7(a) and (b).

Encoder:

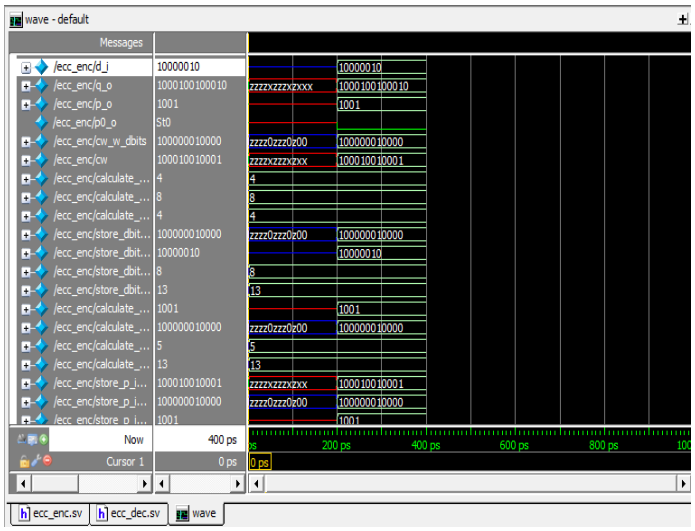

Fig.7(a). Simulation results of encoder.

Decoder:

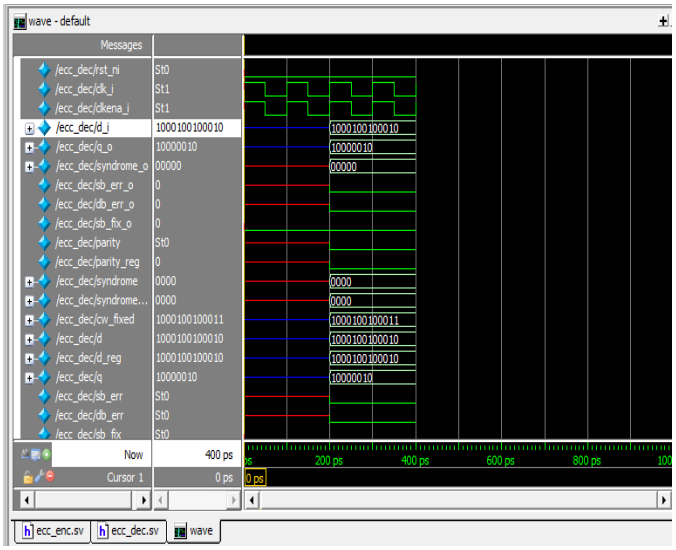

Fig.7(b). Simulation results of decoder.

VI. CONCLUSION

In improved hamming code technique the quantity of overhead bits is incredibly diminished. The parity bits are attached toward the other side of data bits stream. This strategy will wipe out the overhead at the sender end to intersperse the redundancy bits and the redundancy bits elimination at the receiver end. This work is accepted to serve as a decent error correction system for transmission of substantial and variable size data bit-streams, the probability of the length is at the most corrects only single-bit error amid transmission. Further the exertion required in distinguishing the redundancy's values bits is lower in the proposed novel technique. In the traditional Hamming code, if the data lengths are to be scaled for larger data lengths, that results in a considerable measure of overhead because of interspersing the redundancy bits and their evacuation later. In contrast, the improved hamming code proposed strategy [1] can be scaled highly without any of such overhead. Therefore this improved hamming code strategy can be used for the transmission of huge size message data bit-streams. But the overhead in this strategy is much lower on bits per data bit ratio comparatively. This work is accepted to serve as a decent error correction system for transmission of substantial and variable size data bit-streams, the probability of the length is at the most corrects only single-bit error amid transmission.

Future Scope: Since the proposed system is the improvement for the traditional hamming code method, the only drawback which can be improved is the single bit error correction. This proposed system can be further improvised to detect and correct more than one bit error.

VII. REFERENCES

[1] B. Umashankar. "Improved Hamming Code for ErrorDetection and Correction", 2007 2nd International Symposium on Wireless Pervasive Computing, 02/2007
[2] Kythe. "Extended Hamming Codes", Algebraic andStochastic Coding Theory, 2012.
[3] W. W. Peterson and E. J. Weldon, Jr., Error-CorrectingCodes (2nd ed.). Cambridge, MA: MIT Press, 1972.

[4] T. Fujiwara et al., "Error Detecting Capabilities of the Shortened Hamming Codes Adopted for Error Detection in IEEE Standard 802.3," IEEE Trans. Communications, vol. 37,no. 9, pp. 986-989, Sep 1989.
[5] W. Xiong, and D. W. Matolak, "Performance of HammingCodes in Systems Employing Different Code Symbol Energies," IEEE Communications Society, pp. 1055-1058[Wireless and Communications and Networking Conference(WCNC)].
[6] Wyner, "Recent results in the Shannon theory", IEEE Trans.Inf. Theory, vol. 20, pp. 2-10, 1974
[7] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction",Proc. DCC, pp. 158-167
[8] V. Stankovic, A. D. Liveris, Z. Xiong and C. N. Georghiades,"On code design for the Slepian-Wolf problem and lossless multi terminal networks", IEEE Trans. Inf. Theory, vol. 52, no.4, pp. 1495-1507, 2006
[9] S. Pradhan and K. Ramchandran, "Generalized coset codesfor distributed binning", IEEE Trans. Inf. Theory, vol. 51, no.10, pp. 3457-3474, 2005.

Author's Details:

G. AKHILA, student is dept of VLSI System Design in Bharat Institute of Engineering and Technology. garapati.akhila1995@gmail.com

PRADEEP KUMAR, is as a assistant professor is dept of VLSI System Design in Bharat Institute of Engineering and Technology. biet4ru@gmail.com

Prof. G. Kumaraswamy Rao, Director (R&D), BIET (Former Director & O.S. of DLRL Ministry of Defence) Prof. G. Kumaraswamy, former director of Defence Electronics Research Lab an Electronics Systems Engineering Centre, was associated in work with Dr. A. P. J. Abdul Kalam in the "Integrated Guided Missile Program, and was personally responsible for the design and development of a number of missile guidance electronic systems for Trishul, Prithvi and Akash missiles" .He has also designed the ground guidance electronic system for India's first short range surface to air missile. Prof. G. Kumaraswamy is the Director of R&D at Bharat Institute of Engineering and Technology.