IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 4, Issue 5, May 2016

Error Minimization in BCH Codes

Ashutha K[1], Ankitha K[2]

Assistant Professor, ECE dept., Sahyadri College of Engineering & Management, Mangalore, India [1]

Assistant Professor, CSE dept., Sahyadri College of Engineering & Management, Mangalore, India [2]

Abstract: The error rate or the probability of error is an important parameter in the design of digital communication systems. To get rid of these errors, the error control codes are used. This paper briefs about Field Programmable Gate Array (FPGA) implementation of Bose Chaudhuri Hocquenghem (BCH) code used for error correction in transmission using VHDL. The proposed coding scheme achieves good error-performance and complexity trade-offs as compared to the traditional schemes and is very favourable for implementation. In specific BCH code of block length 15, information bit of 7 and error detection and correction upto2 bits. Simulation is carried out using Xilinx12.1 ISE simulator and verified results for the chosen bits. Finally both encoder and decoder design implemented on Spartan 3 FPGA.

Keywords: Error control, Bose Chaudhuri Hocquenghem code, Error correction, Xilinx.

I. INTRODUCTION

Quick development in web and mobile innovation, exchange of data is a regular practice. Data may be text, sound, video and so on. Transmission of data through a physical medium or remote medium, probability that information get ruined this prompts an error in an irregular just chosen areas of an symbol or the whole symbol. Error correcting codes are utilized to have consistent communication through a communication channel that has a reasonable Bit Error Rate (BER) and High Signal to Noise Ratio (SNR). These codes are introduced in order to identify and correct the preset quantity of errors which may occur during transmission of message over a communication channel [1].

There are diverse sorts of Error correction codes are utilized in present digital communication framework taking into account the kind of channel noise. Few of them are Hamming codes, Bose-Chaudhuri Hocquenghem code (BCH), Low Density Parity Check code (LDPC), Reed-Solomon code, and Turbo code. These codes are unique in relation to one another in their complexity and implementation. BCH codes are generally utilized in the areas like, mobile communication, digital communications, satellite communications, optical and magnetic storage systems, and computer networks etc.

In this paper (15, 7) BCH encoder and decoder is executed on Sparten3 FPGA. For outlining the BCH codes, two coding systems are utilized. They are Systematic codes and Non-Systematic codes. In systematic codes unique message d(x) is as it is in the encoded word c(x).Where as in case of non-systematic code encoded word c(x) is acquired by multiplying message d(x) with generator polynomial G (x) [2].

Henceforth message information won't be same in the encoded code word. At the transmitter side utilize encoder circuit binary digits are encoded by some additional bits with message bits called parity bits. The parity bits and message bits together called as 'Code word'. At the Receiver end 'code word' will be got and error identification and correction procedure is applied. This procedure is known as decoding. If error presents in received data within a correction limits, the error will be revised and unique message is recovered. This enhances the nature of transmitted message to great extent and subsequently decreases the error rate.

II. LITERATURE SURVEY

Cyclic decoding techniques for error corrections in BCH code is discussed in [3]. They actualized binary BCH codes for 5 bit error corrections with a length of 127 bits utilizing Peterson decoding technique. Author demonstrated that burst error correcting codes are not so much speed but rather more complex in equipment contrasted with BCH codes.

Utilization of BCH codes in validation of binary documented pictures is discussed in [4]. They utilized (7, 4) BCH encoder for encoding of a character and inserted in a binary documented picture, every 4 bit in a character is encoded to a 7 bit. They utilized BCH codes to correct one bit error in any position of 7 bit information. Usage of (7, 4) BCH encoder to correct single error in any position of 7 bits is discussed. The circuit outline and simulation was completed utilizing Orcad version 9.1 and executed on FPGA (Xilinx xc4013) [5].

Utilization of BCH code for error detection and correction in memory is presented. They compared both single error correction and double error correction BCH codes using 180nm technology. . The synthesis and simulations were completed utilizing Synopsys Design Compiler, power utilization and range of the circuit was condensed for diverse (n, k) BCH code cases. The outcome demonstrates that power utilization and region required was less contrasted with error correction [6].

Use of BCH codes in issue tolerant technique is given in [7]. They outlined a 32 bit ALU, which is secure against numerous flaws and ready to correct any 5 bit issues in any positions of 32 bits input registers of ALU. The BCH codes were utilized to correct numerous errors. Author

IJIREEICE
ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 4, Issue 5, May 2016

demonstrated that contrasted with TMR (triple secluded repetition) and Residue code, BCH codes were the better choice in estimated area. Non-linear multi error modifying codes in the consistent MLC NAND temporary memories applying BCH and RS codes is analysed [8]. The architectures of encoder and decoder for nonlinear 5 bit error corrections in flash memories can be displayed in Verilog and it is synthesized using RTL design compiler [9].

Proposed paper examines, FPGA execution of (15, 7) Binary BCH Encoder and Decoder for text message utilizing VHDL. This paper aims to correct two mistakes in any position of 15 bit code word. Initially character in a text message is converted into binary information of 7 bits. This 7 bit is encoded into 15 bit code word. By Utilizing (15, 7) BCH Encoder if there is 2 bit error in a code word of 15 bits that will be identified and corrected. The corrected information is converted into an ASCII character. The decoder is implemented utilizing Peterson and Zierler algorithm and chine's search algorithm Simulation was completed by utilizing Xilinx 12.1 ISE simulator, and verified the results for an arbitrarily chosen message n formation [10].

A. Problem Statement

Bose, Chaudhuri, Hocquenghem, created BCH Code. Multiple errors can be recognized and corrected utilizing BCH code. BCH Code is a summarized type of Hamming Code. Reed-soloman (RS) code, Golay code and Shortened cyclic codes and Burst-error correcting codes are good algorithm to check and correct error bits present in the given message or information. The main disadvantage of mentioned code is lack of flexibility, Single error bit correction and low Signal to Noise Ratio (SNR). This entire drawback is overcome by the BCH code.

The BCH codes shape a class of cyclic error remedying codes that are assembled using constrained fields. The key components of BCH codes are that amid code plan, there is an exact control over the quantity of symbol errors correctable by the code. In particular, it is possible to outline binary BCH codes that can correct different bit errors. Another merit of BCH codes is the straight imposition with which they can be decoded, particularly, through a logarithmic technique known as syndrome decoding. This disentangles the outline of the decoder for these codes, using little low-control electronic hardware.

III.SOFTWARE DESIGN

It is implemented in the software tool Xilinx (v 12.2) and the code for the corresponding all error correcting are written in the VHDL mixed language module which efficient for the purpose of writing the codes for the above mentioned coding. Then it stimulated using Modalism

A. Modelsim stimulator

ModelSim is an easy-to-use yet versatile VHDL/ (System) Verilog/System C simulator by Mentor Graphics. It supports behavioural, register transfer level, and gate-level modelling. ModelSim supports all platforms used here at the Institute of Digital and Computer Systems and many others too. On Linux and Solaris platforms ModelSim can be found preinstalled on Department's computers. Windows users, however, must install it by themself [11].

B. Load the Design to modelsim stimulator

- Load the main.v module into the simulator.
- In the Library window, click the '+' sign next to the work library to show the files Contained there.
- Double-click main.v to load the design.

You can also load the design by selecting Simulate > Start Simulation in the menu bar. This opens the Start Simulation dialog. With the Design tab selected, click the'+' sign next to the work library to see the counter and main.v modules. Select the main.v module and click OK

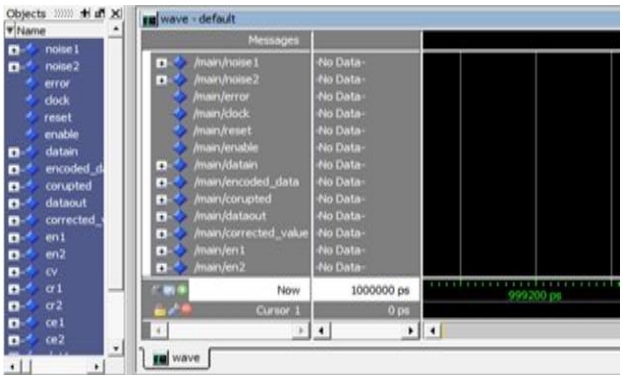

Fig. 1. modelsim stimulator after reset

C. Run the Simulation

We're ready to run the simulation. But before we do, we'll open the Wave window and add Signal's to it [12].

1. Open the Wave window.

- Enter view wave at the command line.

The Wave window opens in the right side of the Main window. Resize it so it is visible. You can also use the View > Wave menu selection to open a Wave window. The wave window is just one of several debugging windows available on the View menu.

2. Add signals to the Wave window.87

- In the Structure (sim) window, right-click main.v to open a popup context menu.
- Select Add > To Wave > All items in region. (All signals in the design are added to the Wave window.)

3. Run the simulation.

- Click the Run icon. (The simulation runs for 100 ns and waves are drawn in the Wave window.)
- Enter run 500 at the VSIM> prompt in thane Trscript window. (The simulation advances another 500 ns for a total of 600 ns)
- Click the Run -All icon on the Main or Wave window toolbar. (The simulation continues running until you execute a break command or it hits a statement in your code (e.g., a Verilog $stop statement) that halts the simulation.
- Click the Break icon to stop the simulation.

IJIREEICE

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 4, Issue 5, May 2016

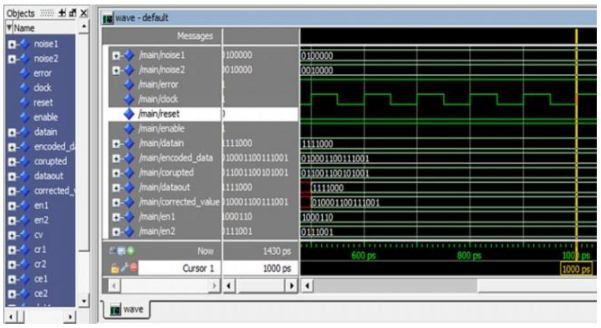

Fig 2. Sample compilation result

IV. RESULTS AND DISCUSSION

In this section simulation and synthesis results of (15, 7) BCH Encoder and Decoder is discussed. The system has been simulated using Xilinx12.1 ISE simulator and functionality of encoder and decoder is verified. Synthesis was carried out by using the cadence RTL compiler, power and area is estimated for 180nm technology.

A. Simulation Results of (15, 7) BCH Encoder and Decoder

The Fig. 3 shows simulation waveform of (15, 7) BCH Encoder. Here "0100110" is considered as a message bits. As seen in fig. 4 when the reset pin is high, the input is loaded to Encoder and all other intermediate signals are set to zero. When enable pin set to high and the reset pin to low, ASCII character converted into 7 bit binary digits. Using these binary digits the parity bits were calculated and these parity bits are appended to the original message bits to obtain a 15 bit encoded data or code word. To indicate the Encoded data in the output terminal hold pin is used. If hold pin is high 15 bit encoded data is obtained at the output port. The same process repeats for other characters as well.

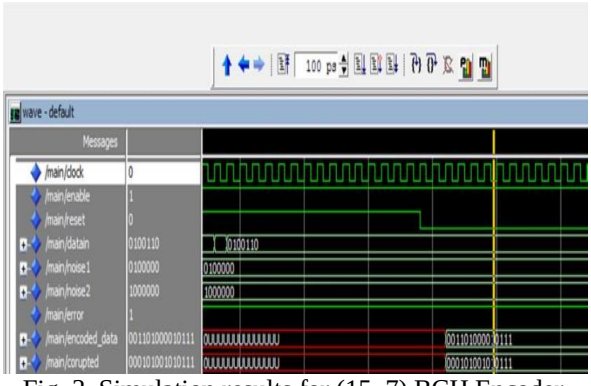

Fig. 3. Simulation results for (15, 7) BCH Encoder

Fig. 5 shows simulation waveform of (15, 7) BCH Decoder with two bit errors. As seen in figure when the reset and load pin is high, encoded data or received vector r(x) is loaded as input to decoder and all other intermediate signals are set to zero. In the decoding process initially, enable pin set to high and the reset pin set to low. During this received 15 bit encoded data is given as input for syndrome computation. If no error in received code word syndrome output is zero. Else syndrome output will not be zero. It indicates error present in the 15 bit received data. Once error in two bits of 15 bit code word, it is detected and corrected using Peterson and Zierler algorithm and Chine's search algorithm as discussed in section 3. To indicate status of error in Message flag bit is used. When the Flag pin is high, it indicates decoded data at the output terminal and it remains low for next corrected data available at the output terminal. This corrected data is converted into an ASCII character. The same procedure is repeated for other characters as well.

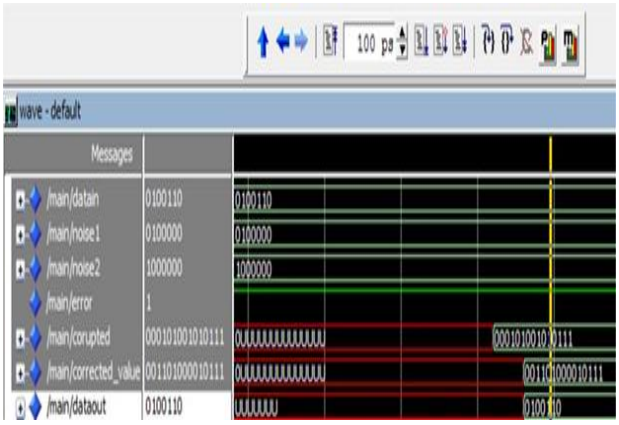

Fig. 4. Simulation results for (15, 7) BCH Decoder

B. Synthesis Results of (15, 7) BCH Encoder and Decoder

The Fig. 5 and Fig. 6 shows RTL schematic of (15, 7) BCH code (encoder and decoder) and Detailed RTL Schematic of (15,7) BCH code respectively generated modalism compiler.

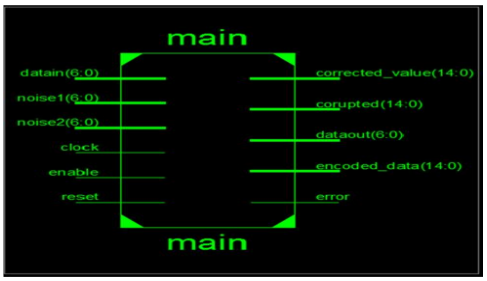

Fig. 5. RTL schematic of (15, 7) BCH

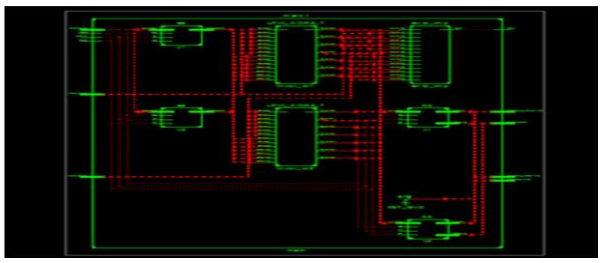

Fig. 6. Detailed RTL schematic of (15,7) BCH Encoder and Decoder

V. ADVANTAGES & APPLICATIONS

The BCH code is a powerful tool used to correct multiple errors present in the message bits. The Encoding and Decoding of message bit very easy and simple.

DOI 10.17148/IJIREEICE.2016.4596

SNR ratio is increased, which helps to get good reception [13].

- Applications like satellite communications and two dimensional bar codes BCH Codes are utilized.
- It is used in Storing devices.
- Wireless or mobile communications
- Digital television
- It is utilized in high speed modems for example ADSL, DSL.

VI. CONCLUSION

The usage of error correcting codes is very important in a modern communication system. In this paper execution of (15, 7) BCH Encoder and Decoder for text message is discussed. Initially each character in a text message is converted into binary data of 7 bits. This 7 bit is encoded into 15 bit code word. If there is any 2 bit error in any location of 15 bit code word, it can be detected and corrected. This corrected data is converted into an ASCII character. The decoder is implemented using the Peterson and Zierler algorithm and chine's search algorithm. Simulation was carried out by using Xilinx 12.1 ISE simulator and verified results for an arbitrarily chosen message data. Also design of both encoder and decoder successfully implemented on Spartan 3E FPGA hardware. Synthesis was successfully done by using the cadence RTL compiler, power and area is estimated for 180nm technology. The power and area obtained for (15, 7) BCH Encoder are 151867.464 nW and 6961µ m2 and for Decoder are 477932.501 nW and 14051 µ m2.

BCH codes have been appeared to be fabulous error rectifying codes among codes of short lengths. They are easy to encoder and moderately easy to decode. The speed and device usage can be enhanced by improving parallel methodology strategies.

REFERENCES

[1] Neubauer, J. Freudenberger and V. Kuhn "Coding Theory Algorithms, Architectures and Applications" John Wiley & Sons, 2007.
[2] T. K. Moon, "Error Correction Coding", John Wiley & Sons, 2005.
[3] A. S. Das, S. Das, and J. Bhaumik "Design of RS (255,251) Encoder and Decoder in FPGA", international journal of soft computing and engineering, Volume- 2, Issue-6, January 2013.
[4] J. G. Proakis,"Digital Communications", Prentice-Hall, 4th edition, 2005.
[5] S. Lin, and D.J. Costello Jr. "Error Control Coding Fundamentals and Applications", Prentice-Hall, New Jersey, 1983.
[6] R. Merha, G. Saini, and S. Singh, "FPGA Based High Speed BCH Encode for Wireless communication Applications", International Conference on Communication System and Network Technologies, 2011.
[7] B. Sklar, "Digital Communications Fundamentals and Applications", Prentice Hall, 2nd edition, 2001.
[8] I. Kuon, R. Tessier and J. Rose. "FPGA Architecture: Survey and Challenges", 2008.
[9] J. P. Deschamps, G. J. A. Bioul and G. D. Sutter,"Synthesis of Arithmetic Circuits FPGA, ASIC and Embedded Systems", John Wiley & Sons, 2006.
[10] A.K. Maini, "Digital Electronics Principles, Devices and Applications", John Wiley & Sons, 2007.
[11] R. Woods, J. McAllister, G. Lightbody and Y. Yi, "FPGAbased Implementation of Signal Processing Systems", John Wiley & Sons, 2008.
[12] S. J. Mohammed, H. F. Abdulsada, "Design and Implementation of 2 BCH Error Correcting Codes using FPGA", Journal of Telecommunicatios, Volume 19, ISSUE 2, APRIL 2013.
[13] S. J. Mohammed, H. F. Abdulsada, "FPGA Implementation of 3 bits BCH Error Correcting Codes", International Journal of Computer Applications, Volume 71– No.7, May 2013.