IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 12, Issue 4, Ver. II (Jul.-Aug. 2017), PP 69-73
www.iosrjournals.org

A VHDL Implementation of UART with Error Coding Algorithms

*Deepika S S[1], Ashwin Kumar[2], Nisha[2]
[1](Electronics &Communication Engineering, Sahyadri College of Engineering& Management, India)
[2](Assistant Professor Electronics & Communication Engineering, Sahyadri College of Engineering &Management, India)
[2](AssistantProfessor Electronics & Communication Engineering,Sahyadri Collegeof Engineering &Management, India)
Corresponding Author: Deepika S S*

Abstract: *Inan industrial working environment employing multiprocessor communication using UART, noise may affect the data and data may be received with errors. The working of the system may be affected due to these kind of errors resulting in an improper control. Several existing UART designs are incorporating error detection logic. In this technique, it requires retransmission of data frames when errors are detected. Thus Linear block codes like hamming code which have forward error correction (FEC) as well as error detection capability can be used to correct data. There are numerous error coding techniques in the field of digital communication. They are utilized to detect/correct single bit or multiple bit errors. In this paper commonly used codes like Hamming, Manchester and Hamming-Manchester codes are compared based on encoding and decoding methods. Hamming codes are single error correcting and double error detecting (SECDED) codes. Manchester coding is used to provide high secured data communication. The whole design is implemented in Xilinx ISE 14.2 simulator. The power, area and delay of the algorithms are computed and compared. Applications include in fields like DRAM memory chips, satellite communications.*

Keywords: *Error coding, SECDED, Hamming code, Manchester code, Xilinx*

Date of Submission: 17-07-2017 Date of acceptance: 28-07-2017

I. Introduction

Environmental interference and physical defects in the communication medium can cause random bit errors during data transmission. Error coding is a method of detecting and correcting these errors to ensure information is transferred intact from its source to its destination. Error coding is used for fault tolerant computing in computer memory, magnetic and optical data storage media, satellite and deep space communications, network communications, cellular telephone networks, and almost any other form of digital data communication. The various codes are used for the error detection and correction in the field of communication. Some are for the single bit error and some are for the multiple bits error. The basic idea for achieving error detection is to add some redundancy to a message which receivers can use to check consistency of the delivered message and to recover data determined to be erroneous. Error-detection schemes can be either systematic or non-systematic: In a systematic scheme the transmitter sends the original data and attaches a fixed number of check bits. That is derived from the data bits by some deterministic algorithm. If only error detection is required a receiver can simply apply the same algorithm to the received data bits and compare its output with the received check bits if the values do not match an error has occurred at some point during the transmission. In a system that uses a non-systematic code the original message is transformed into an encoded message that has at least as many bits as the original message.Therefore error-detecting and correcting codes can be generally distinguished between random-error detecting/ correcting and burst-error-detecting/correcting. Some codes can also be suitable for a mixture of random errors and burst errors. If the channel capacity cannot be determined or is highly varying an error-detection scheme may be combined with a system for retransmissions of erroneous data. This is known as automatic repeat request (ARQ) and is most notably used in the Internet. Error detection is the detection of errors caused by Noise or other impairments during transmission from the transmitter to the receiver. Error correction is the detection of errors and reconstruction of the original error free data or signal.

II. Hamming-Manchester Coding

Hamming-Manchester coding is the combination of two coding techniques namely Hamming and Manchester coding techniques. In Hamming-Manchester coding both Hamming and Manchester coding is performed in series with one another which is implemented to a UART. At the transmitter side, Hamming

encoding is performed first followed by Manchester encoding. At the receiver side, Manchester decoding is performed first followed by hamming decoding.Here Hamming code is used for the error detection and correction of the data and Manchester code is used to encrypt and decrypt the data.The advantage of adding Manchester coding is that it improves the network security of the system. Thus the data received in the UART undergoes both coding methods which ensures transfer of error free data and provides a high data security path.

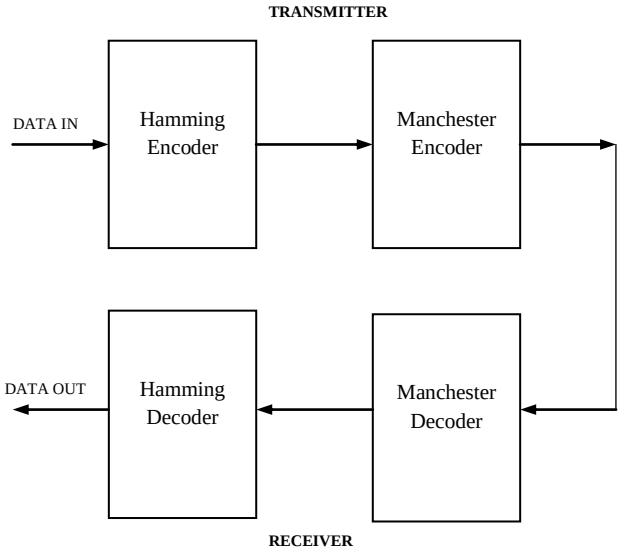

Fig.1 Block diagram of Hamming-Manchester coding

2.1 Hamming-Manchester Simulation Results:

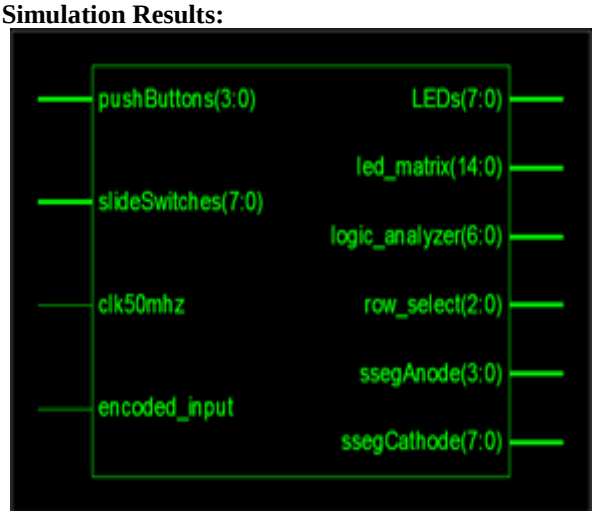

Fig.2 RTL Schematic of Hamming-Manchester coding

III. Hamming Algorithm

In telecommunication, Hamming codes are a family of linear error-correcting codes that generalize the Hamming (7, 4)-code, and were invented by Richard Hamming in 1950. Hamming codes can detect up to two-bit errors or correct one-bit errors without detection of uncorrected errors. By contrast, the simple parity code cannot correct errors, and can detect only an odd number of bits in error. Hamming codes are perfect codes, that is, they achieve the highest possible rate for codes with their block length and minimum distance of three.UART is the peripheral to the microprocessor for asynchronous serial communication. It takes a data byte/wordat a time from the microprocessor and adds start, stop, parity bit etc., to form a frame. This frame is transmitted bit by bit. Synchronization between transmitter and receiver is achieved by the start and stop bits. This protocol thus adds extra bits (overhead) to the data byte. The inclusion of a hamming encoder in the transmitter section and hamming decoder in the receiver section can correct up to one error and detect up to two errors.

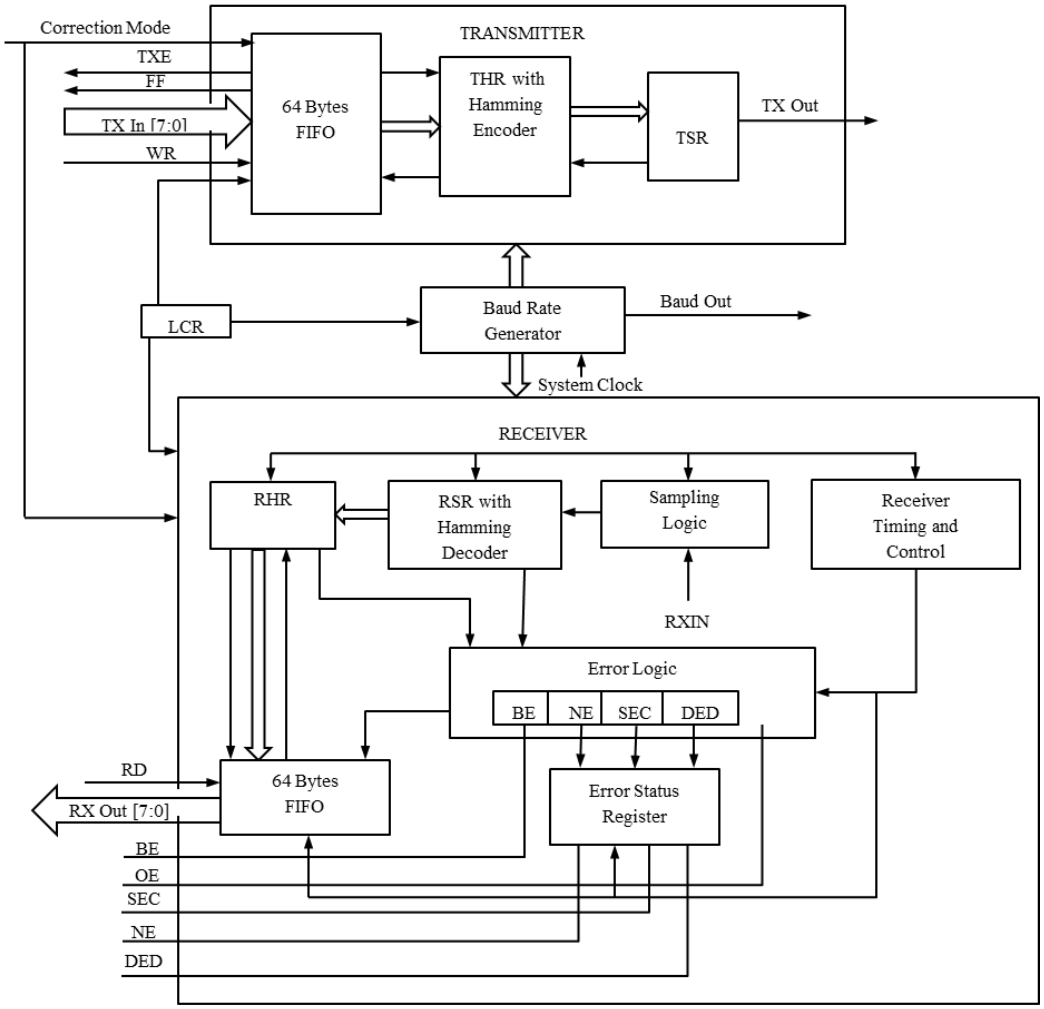

Fig.3 Proposed UART architecture

The proposed architecture is shown in figure 3. The proposed UART has a transmitter, a line control register (LCR), a baud rate generator (BRG), a Transmitter with hamming encoder, and a receiver with status register and hamming decoder. LCR is 8 bit programmable register which is used to determine the number of data bits, baud rate parity, etc. BRG is used to generate required baud rate. The transmitter has three sections: a FIFO (First in First out), THR(Transmit Hold Register) and TSR(Transmit Shift Register). FIFO is a dual port RAM of 64KB. As the name suggests the 8 data is transferred in first in first out manner. THR converts the 8 bit data into 12 bits and it also includes the Hamming encoder. The encoder takes the 4 data bits followed by 4 parity bits and creates syndromes which is transferred along with the data. TSR is a parallel to serial converter of 12 bits. The receiver has 5 main sections: a Sampling stage, RSR (Receiver Shift Register), Error Logic, RHR(Receiver Hold Register) and FIFO. In the Sampling stage, data is received when read enable signal is high and it is sampled thrice. When three samples are same data is moved to next stage that is the RSR. RSR converts the serial data into parallel data and it also includes Hamming decoder. The syndrome is decoded in the decoder and updates various flags based on the presence of error. Error Logic contains various flags like BE (Burst Error), NE(No Error), SEC(Single Error Correction), DED(Double Error Detection)and OE(Overrun Error). The RSR is a 12 bit register which removes the start, stop and any overhead bits. It converts the 12 bit data into 8 bit data which is transferred to the next stage FIFO. Data from this FIFO can be read under the control of system clock.

2.1 Hamming Simulation Results:

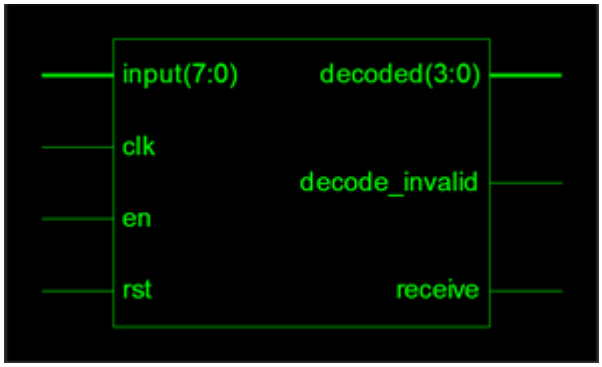

Fig.4 RTL Schematic of Hamming decoder

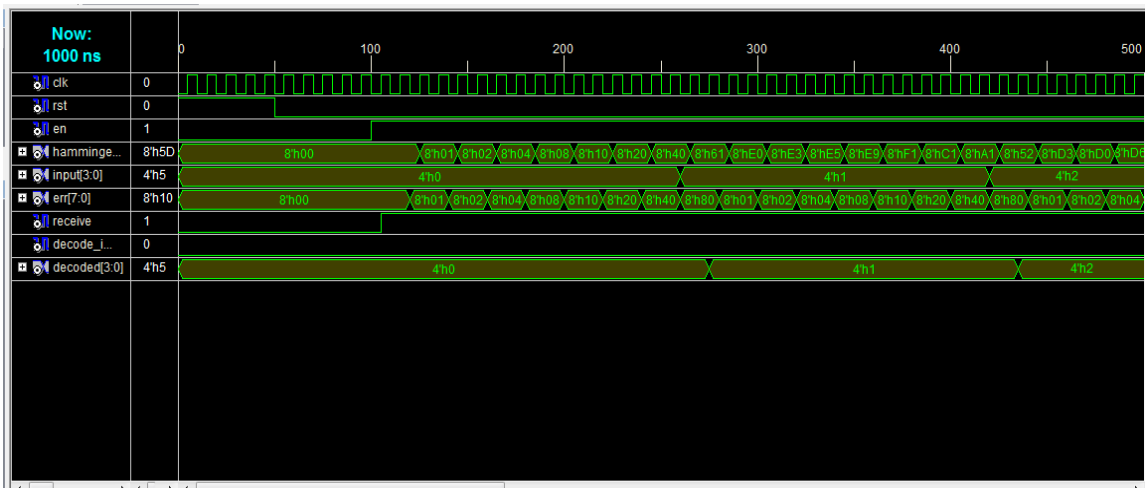

Fig.5 Simulation result of Hamming decoder

IV. Manchester Coding

Manchester coding also known as phase encoding, or PE is a line code in which the encoding of each data bit is either low then high, or high then low, of equal time.It is used in telecommunication and data storage systems.The main advantage of Manchester coding is the fact that the signal synchronizes itself. This minimizes error rate and optimizes reliability. Here the Manchester encoding is performed at the transmitter end and decoding is performed at the receiver end of the UART. The encoding and decoding uses differential Manchester coding method. In this method transition occurs in the middle of each time interval. When the data received is a zero(low) bit then transition occurs at the beginning of the interval and if the data received is one(high) bit then no transition occurs at the beginning of the interval. This method ensures data security of the UART.

3.1 Manchester Simulation Results:

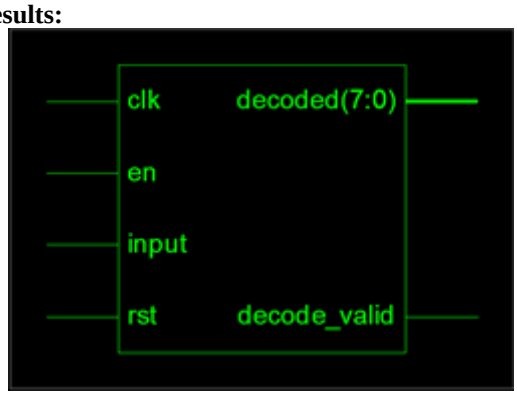

Fig.6 RTL Schematic of Manchester decoder

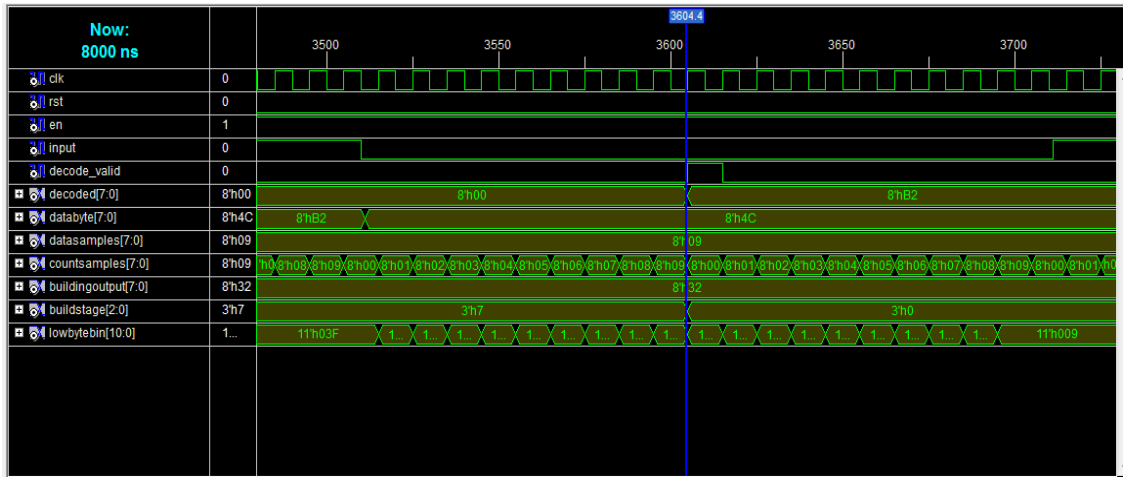

Fig.7 Simulation result of Manchester decoder

V. Performance Result Of Three Algorithms

Algorithms are implemented using Xilinx 14.2 and their delay, power, area are computed using the same.

Table.1 Results

Algorithms	Max Freq(MHz)	Time delay(ns)	LUT's	FF's	Power(W)
HAMMING	359.245	1.57	10	16	1.294
MANCHESTER	399.114	2.506	66	68	1.298
HAMMING-MANCHESTER	327.955	2.807	298	182	1.295

Fig.8 Comparison Graph

VI. Conclusion

This paper proposes an UART with Hamming error correction and detection algorithm along with encryption using Manchester algorithm. The algorithms are implemented separately and then combination of both algorithmsare implemented. The delay, area and power of the three algorithms are compared. When compared it can be seen that delay, gate area and power of Hamming is better than other two algorithms. Thus for data reliability applications only Hamming code can be implemented in UART. The Manchester and Hamming-Manchester algorithm have almost similar area and delays. Thus Hamming-Manchester algorithm can be implemented for applications where both data reliability and secure communication is required.

References

[1] SindhujaMuppalla, Koteshwara Rao Vaddempudi, "A Novel VHDL Implementation of UART with Single Error Correction and Double Error DetectionCapability" SPACES-Dept of ECE, KL UNIVERSITY, 2015.

[2] Naresh Patel, Vatsalkumar Patel and Vikaskumar Patel, "VHDL Implementation of UART with Status Register" International Conference on Communication Systems and Network, May 2012.

[3] R.W. Hamming, "Error detecting and error correcting codes", The Bell System Technical journal Vol. XXIX, American Telephone and Telegraph company,Vol. 2, April 1950.

[4] PallaviUpase and PravinChoudhari "Overview of Error Mitigation Techniques in UART Communication" International Journal of Innovative Research in Computer and Communication Engineering, Vol. 4, Issue 3, March 2016

[5] Anjali.V P. Satishkumar, "Manchester Encoder and Decoder with Clock Recovery Unit and Invalid Detector" Computer Engineering and Intelligent, Vol.6, No.2, 2015

[6] Elmenreich W, Delvai M, "Time-triggered communication withUARTs", 4thIEEE International Workshop on Factory CommunicationSystems, 2002, pp. 97- 104, 2002.

[7] Gallo R, Delvai M, Elmenreich W, Steininger A, "Revision andverification of an enhanced UART", 2004, Proceedings, 2004 IEEEInternational Workshop on Factory Communication Systems, pp. 315-318, 22-24, Sept. 2004.

[8] Norhuzaimin J, Maimun H.H, "The design of high speed UART", Asia-Pacific Conference on AppliedElectromagnetics (APACE), pp. 20- 21,Dec. 2005.

IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) is UGC approved Journal with Sl. No. 5016, Journal no. 49082.

Deepika S S. "UART with Error Coding." IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) 12.4 (2017): 69-73.