International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 2, Issue 4, April 2015

Burst Error Correction Using CRC

[1]Remya Maria Mathew,[2]Shilpa Cherian,[3]Shyno Mary Alexander,[4]Surumy Salim,[5]Agi Joseph George
[1][2][3][4]UG Scholar,Assistant Professor[5]
Amal Jyothi College of Engineering,Koovappally,Kanjirappally

Abstract: **Networks must be able to transfer data from one device to another with complete accuracy. Data can be corrupted during transmission. For secure communication, errors must be detected and corrected.CRC is an efficient method to detect and correct the errors. In this paper we have introduced a new approach for the burst error detection and correction using non-lookup table technique.**

Index Terms **CRC,Single bit error correction,non-lookup table,burst error**

I. INTRODUCTION

The need to transmit and store large amount of data reliably and without error is important in communication system. Error correcting and detecting codes play a vital role in reducing the error in the received data due to noise and interference. The received data may not be the same as the transmitted data because of noise and interference, which leads to errors during the process of data transmission and storage. Hence to achieve better quality data transmission coding and detection methods are employed.CRC is an easy and efficient method to detect and correct the errors.

In this paper we have introduced a new approach for the error detection and correction using non-lookup table technique. This method is memory efficient and operates at high frequency. Since most of the methods currently used are based on lookup table technique, they consume more space and also there will be a reduction in speed because of the overhead required in accessing the ROM which stores the lookup table.

I. PRINCIPLE

A cyclic redundancy check (CRC) is an error-detecting code commonly used in digital networks and storage devices to detect accidental changes to raw data. Blocks of data entering these systems get a short check value attached, based on the remainder of a polynomial division of their contents. On retrieval the calculation is repeated, and corrective action can be taken against presumed data corruption if the check values do not match.

Specification of a CRC code requires a generator polynomial. This polynomial act as the divisor in the polynomial long division, in which the message is the dividend, the quotient is discarded and the remainder becomes the result. The important point is that the polynomial coefficients are calculated according to the arithmetic of a finite field, so the addition operation can always be performed bitwise-parallel (there is no carry between digits). The length of the result can be determined by evaluating the length of the remainder which is always less than the length of the generator polynomial. Most commonly used CRCs employ the Galois field GF (2) which usually consist of two elements 0 and 1, suitable for computer architecture.

Before transmission, check bits are calculated which is then cascaded with data bits. These check bits which are redundant are used for error detection and correction at the receiver. Checksum bits are calculated using a fixed generator polynomial. The data bits are then divided by the generator polynomial and the resultant bits are the checksum bits. The division of the data by the generator polynomial is performed using the circuit which is shown in Figure 1.

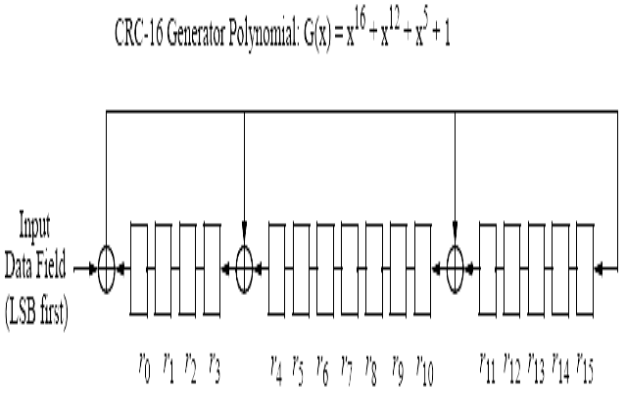


Fig 1.Polynomial Divider

Modulo-2 addition is performed to calculate the checksum bits using the above shift register. Initially all the shift registers are initialized to zero. Then the data bits are shifted

into the circuit with the LSB of the data is entering into the circuit first. Now depending upon the generator polynomial the data is just shifted right or it is added with the feedback value. Once all the data bits are shifted into the registers, the remainder registers indicate the checksum bits. These checksum bits are cascaded with the data bits before the data is transmitted.

The codeword C(x) can be calculated using the Equation

$$C(x) = x^{n-k} D(x) / G(x) + D(x) \quad (1)$$

The first part of right hand side of Equation 1 represents the checksum bits. Here 'n' represents the frame width and 'k' represents the data width. As an example take n=24 and k=8 the calculation of checksum bits is shown below.
Similarly the codeword for CRC-16 can be calculated. The CRC Encoder is designed to generate the codeword in a single clock cycle with high speed and efficiency. [1]

$$\begin{array}{l} \quad\quad x^5+x^4+x^2+x \\ x^{16}+x^{12}+x^5+1 \;\big|\; x^{21}+x^{20}+x^{18}+x^{16} \\ \quad\quad\quad\; x^{21}+x^{17}+x^{10}+x^5 \\ \hline \quad\quad\quad\quad x^{20}+x^{18}+x^{17}+x^{16}+x^{10}+x^5 \\ \quad\quad\quad\quad x^{20}+x^{16}+x^9+x^4 \\ \hline \quad\quad\quad\quad\quad x^{18}+x^{17}+x^{10}+x^9+x^5+x^4 \\ \quad\quad\quad\quad\quad x^{18}+x^{14}+x^7+x^2 \\ \hline \quad\quad\quad\quad\quad\quad x^{17}+x^{14}+x^{10}+x^9+x^7+x^5+x^4+x^2 \\ \quad\quad\quad\quad\quad\quad x^{17}+x^{13}+x^6+x \\ \hline \quad\quad\quad\quad\quad\quad\quad x^{14}+x^{13}+x^{10}+x^9+x^7+x^6+x^5+x^4+x^2+x \end{array}$$

Here:
Data, D(x):00010011
Check bits:00110011011110110
Codeword, C(x):00110011011110110__00010011

II. METHOD FOR ERROR DETECTION AND CORRECTION

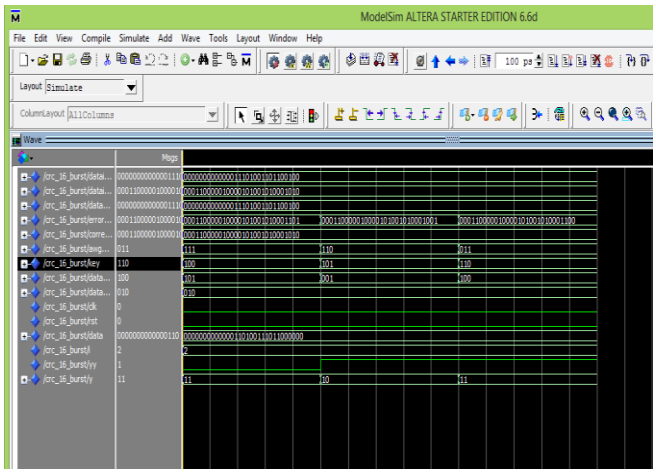
The term burst error means that two or more bits in the data unit have changed from 1 to 0 or from 0 to 1. Burst errors do not necessarily mean that the errors occur in consecutive bits, the length of the burst is measured from the first corrupted bit to the last corrupted bit. Some bits in between may not have been corrupted. Burst error is most likely to happen in serial transmission since the duration of noise is normally longer than the duration of a bit. The number of bits affected depends on the data rate and duration of noise. In this paper, we will be representing a unique way of implementing multiple bit error detection and correction using CRC for a frame width of 3 bits and 27 bit data.The data is then interleaved. At the receiver end, error detection is done by checking if the received data is equal to the interleaved data otherwise there is an error and the data needs to be corrected.Let Ft be the frame transmitted in which the checksum is appended after 27 bits of data .We can express Ft as shown in Equation 2. Ft=Dt & Ct (2) Where, & -Concatenation operator Dt-Transmitted 8bit data Ct-Transmitted 16 bit checksum

At Receiver side, let Frr is the received frame with Drr data bits and Crr checksum bits

Frr=Drr & Crr (3)

If some error has occurred during transmission the received data will not match the interleaved data. In such case error need to be detected and corrected. The CRC is generated using the circuit in figure 1.The data is then interleaved and transmitted serially. At the receiver end the received data is checked if it is same as the transmitted interleaved data.If not an error has occurred and the error bit is inverted.The data is then deinterleaved to get back the original data. This method can be followed for any number of data bits and CRC polynomial.

SIMULATION RESULT



CONCLUSION

CRC method can detect the error in transferring data between two points.In this project a new method based on CRC has been introduced for burst error correction.Usual components like shift registers ,XOR and NOR gates are used.So the entire circuit can be easily designed.

REFERENCES

[1] Pramod S P, Rajagopal A, Akshay S Kotain,"FPGA Implementation of Single bit error correction using CRC" International Journal of Computer Applications (0975 – 887) [2] Shukla S, Bergmann N W. "Single bit error correction implementation in CRC-16 on FPGA" In: IEEE International Conference on Field-programmable Technology. Brisbane, Australia, 2004: 319-322. [3] W.W Peterson, E. J. Weldon. "Cyclic Codes for Error Detection". Second Edition Published by MIT Press, 1972 ISBN 0262160390, 9780262160391. [4] K. Sam Shanmugam, John Wiley, "Digital and analog communication systems", 1996. [5] Peterson, W. W. and Brown, D. T. "Cyclic Codes for Error Detection "Proceedings of the IRE 49: 228. doi: 10.1109/JRPROC.1961.287814, January 1961

[6] J.K. Wolf and D. Chun, "The single burst error detection performance of binary cyclic codes," IEEE Trans. Commun., vol. 42,pp. 11-13, Jan. 1994