ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 1, January 2017

Error Correction and Detection using Cyclic Redundancy Check

Dr. T. Logeswari

Associate Professor, Dept of Computer Science, New Horizon College, Banglore, Karnataka, India

ABSTRACT: In this paper Cyclic Redundancy Check codes are implemented to detect the usefulness of various types of errors that might occur through the transmission of data stream carrying message signal through the internet. The explanation has been prepared in such a way that bit position, number of bits of data word and codeword of the generator polynomial are measured random. This helped to obtain satisfactory result with the proposed polynomial.

KEYWORDS: Cyclic Redundancy Check, Data word, Codeword, Syndrome.

I. INTRODUCTION

When a communication signal is sent from one network to another a mixture of distortion or alter within the shape of that signal occurs. This is due to attenuation, presence of noise, high latency all these factors. Broadcast links and transmission links are the two main types of data transmission technology[1]. Many error-detecting and error-correcting codes are known, but both ends of the link must agree on which one is being used. In addition, the receiver must have some way of telling the sender which communication have been suitably established and which has not. Method of coding can be mainly classified as block coding and convolution coding.

There are also other ways like evaluation of minimum hamming distance, internet checksum, cyclic coding. Among these, cyclic codes provide better performance in detecting different types of errors. Cyclic codes are special linear block codes with one extra property[4]. In a cyclic code, if a codeword is cyclically shifted (rotated), the result is another codeword. Cyclic Redundancy Check (CRCs) codes are so called because the check (data verification) code is a redundancy (it adds zero information) and the algorithm is based on cyclic codes[2]. The central concept in detecting or correcting errors is redundancy. These redundant bits are added by the sender and removed by the receiver. Their occurrence allows the receiver to detect or correct the corrupted bits.

II. TYPES OF ERRORS AND A PROPOSED POLYNOMIAL

Whenever bits flow from one position to another, they are subject to changeable since of interference. This interference can change the shape of the signal. In a single-bit error, a single bit has to be changed 0 is altered to a 1 or a 1 to a 0. In a burst error, multiple bits are changed. A burst error is more likely to occur than a single-bit error. The period of noise is usually longer than the period of 1 bit, which means that when noise affects data, it affects a set of bits. The number of bits pretentious depends on the data rate and interval of noise. Now a better way to know cyclic codes and how they can be implemented is to represent them as polynomials[1]. A pattern of 0s and 1s can be represented as a polynomial with coefficients of 0 and 1. The power of each term shows the position of the bit; the coefficient shows the value of the bit. Here ($x^4 + x^2+x + 1$) this 4^{th} order polynomial is chosen as a standard one ,whose binary pattern is represented as 10111.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 1, January 2017

III. DESIGN OF CRC ENCODER-DECODER

A code is called cyclic if $[x_n x_0 x_1 ... x_{n-1}]$ is a codeword whenever $[x_0 x_1 ... x_{n-1} x_n]$ is also a codeword. CRCs are also classified as block coding. In block coding, the original message is divided into blocks, each of k bits, called datawords, r redundant bits are added to each block to make the length n = k + r. The resulting n-bit blocks are called codewords. Now the extra r bits are chosen or calculated. CRC is used in networks as LANs or WANs. Figure (1) depicts the principle of operation of CRC Encoder and Decoder[5].

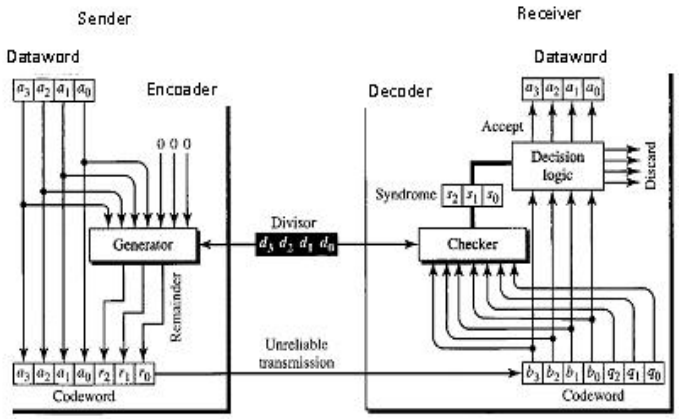


Fig 1: CRC encoder and decoder

In the encoder, the dataword has k bits (8 here); the codeword has n bits (12 here). The size of the dataword is augmented by adding n - k (4 here) 0's to the right-hand side of the word. The n-bit result is fed into the generator. The generator uses a divisor of size n - k + 1 (5 here), predefined and agreed upon. The generator divides the augmented dataword by the divisor (modulo-2 division). The quotient of the division is discarded; the remainder ($r_3 r_2$ $r_1 r_0$) is appended to the dataword to create the codeword[11].

The decoder receives the possibly corrupted codeword. A copy of all n bits is fed to the checker which is a replica of the generator. The remainder formed by the checker is a syndrome of n - k (4 here) bits, which is fed to the decision logic analyzer. The analyzer has a simple function. If the syndrome bits are all as, the 8 leftmost bits of the codeword are accepted as the data word otherwise, the 8 bits are discarded (mistake)[12].

IV. ALGORITHM FOR CRC ERROR DETECTION

CRC_ENCODER (data_word, divisor) k length of the data_word

n length of the code_word c n – k

Augment c 0's to the right hand side of the data_word to create the augmented_data_word

remainder = MOD_2_DIV (augmented_data_word, divisor)
Append the remainder to the original data_word to create the code_word

CRC_DECODER (code_word, divisor) remainder = MOD_2_DIV (code_word, divisor)

If the remainder contains all 0's then the data_word is accepted

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 1, January 2017

Else

the data_word is discarded MOD_2_DIV (augmented_data_word, divisor)

Call the uppermost c bits of the message the remainder Beginning with the most significant bit in the original message and for each bit position that follows, look at the c bit remainder:

If the most significant bit of the remainder is a one
then
Set the appropriate bit in the quotient to a one, and

XOR the remainder with the divisor and store the result back into the remainder
Else

Set the appropriate bit in the quotient to a zero, and

XOR the remainder with zero (no effect) Left-shift the remainder, shifting in the next bit of the message.

Return the remainder

V. CYCLIC CODE ANALYSIS

A cyclic code can be analyzed to find its capabilities by using polynomials. The following is defined where f(x) is a polynomial with binary coefficients[6,7].

		Generator:
Dataword: d(x)	Codeword: c(x)	g(x)
Syndrome: s(x)	Error: e(x)	

If s(x) is not zero, then one or more bits is corrupted. However, if s(x) is zero, bit is not corrupted with any errors. In a cyclic code,
I. If s(x) ≠ 0, one or more bits is corrupted.
2. If s(x) = 0, either

a. No bit is corrupted. or

b. Some bits are corrupted, but the decoder failed to detect them.

In this analysis the intention is to find the criteria that must be imposed on the generator, g(x) to detect the type of error . Now the relationship among the sent codeword, error, received codeword, and the generator is also found as shown below:

Here, received codeword =c(x) + e(x)

In other words, the received codeword is the sum of the sent codeword and the error. The receiver divides the received codeword by g(x) to get the syndrome[8]. This can be written as-

$$\frac{received\ codeword}{g(x)} = \frac{c(x)}{g(x)} + \frac{e(x)}{g(x)}$$

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 1, January 2017

The first term at the right-hand side of the equality does not have a remainder (according to the definition of codeword). So the syndrome is actually the remainder of the second term on the right-hand side. If this term does not have syndrome =0, either e(x) is 0 or e(x) is divisible by g(x). The first case is simple if there is no error; the second case is very important. Those errors that are divisible by g(x) are not caught. In a cyclic code, those e(x) errors that are divisible by g(x) are not caught.

VI. IMPLEMENTATION

Steps followed by Cyclic Redundancy check

Sender Side (Encoder)[10]

1. Convert the message into binary form
2. Add Zeros to the message with divisor values ie 4 bit you add 3 Zeros
3. Perform Modulo 2 division
3. Remainder is consider as redundancy bits

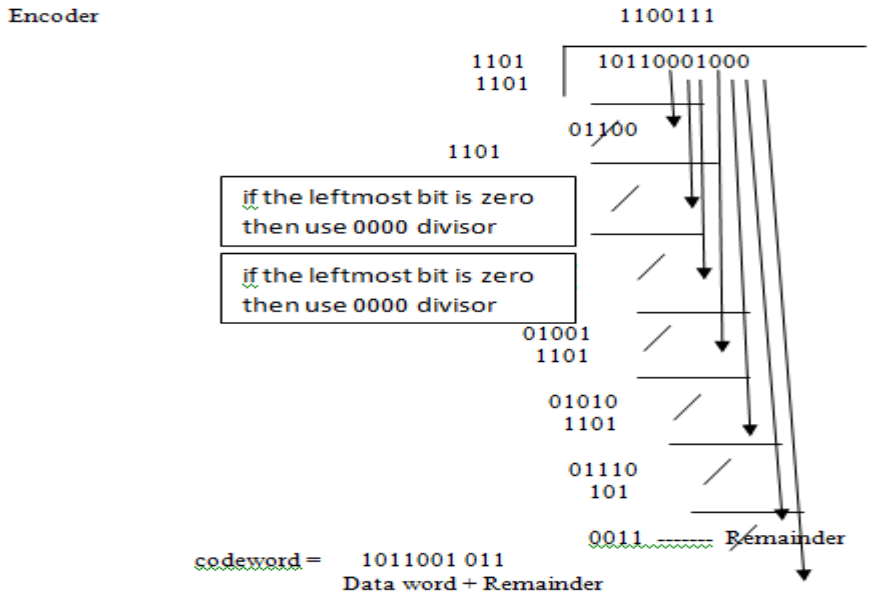
On the other hand, the receiver does not know if no errors have occurred through communication[9],

Receiver Side(Decorder)

1.codeword is transferred to receiver side
2. Perform Module 2 division
3. if remainder is zero message has been sent without any error. if it is non zero then some bits has been corrupted

ENCODER

Divisor - 1101
Dividend - 1011001
if the divisor is 4 bit value then you add three zero to the dividend now you get the value 1011001000



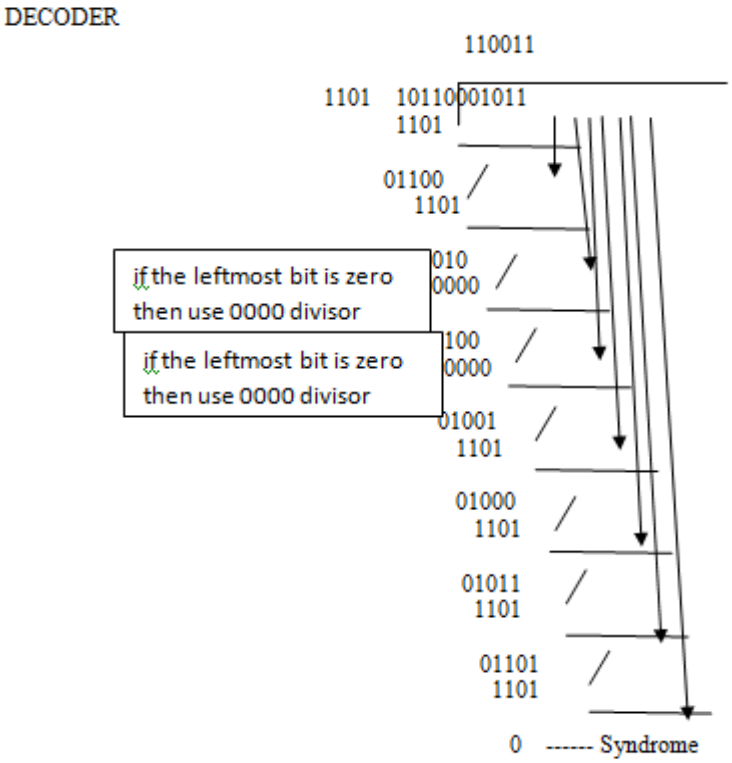
ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 1, January 2017



VII. CONCLUSION

The better way to understand the cyclic codes is polynomial. The properties of cyclic codes suggest a very simple method to encode a message. Cyclic code have a very good performance in system of division process using Modulo-2 arithmetic to detect single bit error and burst error.

REFERENCES

1. Andrew S. Tanenbaum, "Computer Networks"; 4th edition.
2. Behrouz A. Forouzan, "Data Communication and Networking"; 4th edition.
3. R.E. Blahut, "Transform techniques for error control codes", IBM J. Res. Dev. 23 (1979), 299-315.
4. E. R. Berlekamp, "Algebraic Coding Theory", New York McGraw-Hill, 1968
5. 5. X. Chen, I.S. Reed, T. Helleseth & T.K. Truong, "General principles for the algebraic decoding of cyclic codes", IEEE Transactions on Information Theory, vol. 40, N.5, September 94, pp. 1661-63.
6. Sklar, B., 2001. Digital Communications – Fundamentals and Applications. 2nd ed. New Jersey: Prentice Hall
7. C. E. Shannon, "A mathematical theory of communication", Bell Sys. Tech. Jour., vol. 27, pp. 379–423, 623–656, Jul./Oct. 1948.
8. T. Richardson and R. Urbanke, "Modern Coding Theory". Cambridge University Press, 2007.
9. Z. Zhang, V. Anantharam, M. Wainwright, and B. Nikolic, "An efficient 10 GBASE-T ethernet LDPC decoder design with low error floors", IEEE Journal of Solid-State Circuits, vol. 45, no. 4, pp. 843 –855, Apr. 2010.
10. P. Grover, K. Woyach, and A. Sahai, "Towards a communication theoretic understanding of system-level power consumption,‖ IEEE J. Select. Areas Commun., vol. 29, no. 8, pp. 1744 – 1755, Sept. 2011.
11. P. Grover, A. Goldsmith, and A. Sahai, "Fundamental limits on complexity and power consumption in coded communication", paper presented at ISIT‘12, Feb. 2012.
12. P. Grover and A. Sahai, ―Fundamental bounds on the interconnect complexity of decoder implementations,‖ in Proc. of the 45th Annual Conference on Information Sciences and Systems (CISS), March 2011, pp. 1 – 6.