JOURNAL OF AERONAUTICAL MATERIALS
ISSN: 1005-5053

Vol. 44, Issue-01, 2024
pp. 140-146

MESSAGE INTEGRITY AND SIMULATION OF CYCLIC REDUNDANCY CHECK USING MATLAB

Mallikarjun Mudda*, Syed Jahangir Badashah*, T Ramaswamy*, Shruti Bhargava Choubey*, Abhishek Choubey *, Ambika M#, D Geervani1, PVNSV Rohit [2], Shivani A[2]
*Professor, Sreenidhi Institute of Science and Technology, Hyderabad.
#Assistant Professor, Nalla Malla Reddy Engineering College, Hyderabad, India.
#B. tech Scholars, Department of ECE, Sreenidhi Institute of Science and Technology, Hyderabad, India.

ABSTRACT: In digital communication and information theory, error detection has great practical importance in maintaining information integrity across noisy channels. Cyclic Redundancy Check (CRC) is a widely used error detection method in data communication and storage systems. This project presents an implementation of CRC in MATLAB (Matrix Laboratory), aiming to enhance the reliability and integrity of data transmission or storage. The primary objective is to design and implement a computationally efficient, adaptable, and standards- compliant CRC algorithm that can be seamlessly integrated into MATLAB environments. The MATLAB implementation is designed to be versatile, accommodating various data types, and allowing users to customize CRC parameters. The code is optimized for efficiency, considering real-time applications and computational resources. Through simulations and testing, the proposed CRC implementation demonstrates its effectiveness in detecting errors, ensuring data integrity, and adhering to industry standards. This project contributes a valuable tool for MATLAB users seeking a robust error detection mechanism for their data communication and storage applications. The CRC implementation in MATLAB serves as a reliable and efficient solution for enhancing the integrity of transmitted or stored data in diverse scenarios. Keywords: Cyclic redundancy check, Data integrity, Matrix labarotary

I. INTRODUCTION

One of the most important signal processing techniques, channel coding allows for the efficient transmission of digital information (bit data) over the channel. The primary objectives at the receiver are error detection and correction; channel coding regulates the increase in the number of symbols in the source encoded message to support these tasks. Its purpose is to lessen the amount of interference and noise in electronic media. Unpredictable changes in the transmission medium and the lack of a guarantee of accurate reception are inherent problems with data transmission over networks. Consequently, error control is the responsibility of the Data Link layer in the OSI model. The integration of error detection and correction is known as error control. This data link layer provides error detection and flow control mechanisms to ensure that there is reliable transfer of the data between nodes which are being connected by the physical layer.

The Data Link layer is in charge of detecting and fixing errors between routers as data is transmitted over the network. Error is any unwanted change which reduces the usefulness of original data. Error is a condition that when the receiver's information does not match with the sender's information. During the transmission the digital signals can get affected by different noise that can introduce the error in the binary bits, between the receiver and sender. That means a 0 bit may change to 1 bit or 1 bit may change to 0 bit. To prevent such errors, error-detection codes are added as extra data to digital messages. This helps in detecting any errors that may have occurred during message transmission.

To detect this errors a common technique is to introduce redundancy bits that provide the additional information. There are various techniques for error detection methods, one of them is Cyclic Redundancy Check (CRC) which have the advantages that, increased the data realiabilty and improved network performance.

II. PROPOSED SOLUTION:

The proposed solution outlines encompassing approach to designing and implementing a CRC algorithm using the MATLAB programming language.

MATLAB- MATrix LABoratory. It was developed by Mathworks, and it is used for multipurpose programming language. It can be used for the analysis and design as such as the control systems. MATLAB is generally perform the various operations like signal processing , optimization of function and also for Machine learning and Deep learning. The approach involves a step-by-step design process, ensuring clarity and transparency in the implementation. By breaking down the CRC algorithm into modular components, the solution aims to enhance readability, maintainability, and facilitate easy integration into various applications. The solution emphasizes the importance of understanding CRC principles during the implementation process. In this implementation of CRC we can specify the number of data bits as per our requirements. The MATLAB code of CRC algorithm can provide the check value which is additional information to our data. The application focus of the proposed solution extends to digital data verification in communication systems and storage. MATLAB is suitable for data analysis makes it an ideal platform for developing the CRC algorithm capable of solving the challenges the proposed by the noisy communication channels and data corruption during transmission.

III. LITERATURE SURVEY:

In the paper A cyclic Redundancy Check Architecture for High speed networks, they have studied that in their work they have shown us the value of the CRC and how the CRC has been revolutionizing the present trends in the given digital communication and how the CRC would be an effective to to be used for the encoding and decoding process.

In the paper Efficient implementation of CRC for the embedded systems it has been researched by them that the CRC is also an important and could be an important part for the embedded systems also as it also plays an crucial role over there, here they have taken some of the information and have seen that the CRC could also be an revolutionizing work for the embedded systems.

In the paper of the design and implementation fo the Highspeed CRC circuit for the Aerospace communication system, they have done an detailed analysis where the authors have taken the CRC for the implementation in the Aerospace and they have also provided an insight on how

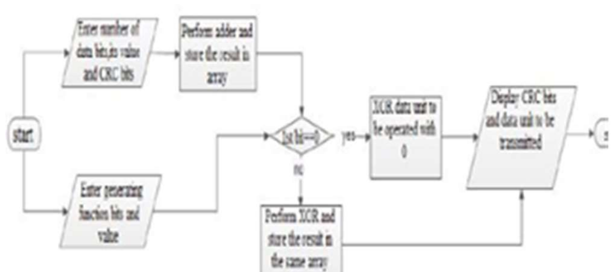
the fast revolutionizing technologies have been significantly using the CRC in their work which indirectly indicates how efficient the CRC is and showing us the importance of the CRC.
In the paper A New method CRC Generation for Data Integrity, the authors of the paper have been clear where they have shown us that the CRC is playing an key role and efficient role in the present generation of the digital communication for both the encoding and the decoding process which is an important and key factor and they have even researched and have given the theoretical and practical values on how the work and implementation could be done to get the desired results and output.

In the paper Cyclic Redundancy codes: Study and Implementation by Sukirty Jain and Siddarth Singh Chouhan, in this they have show how CRC can be part of the reliable communication and the system with the CRC would have an enriched error detection thus, they have used CRC to achieve an accuracy in data communication.

Emily R. Blem, Suman Mamidi, and Michael J. Schulte's "Instruction Set Extensions for Cyclic Redundancy Check on a Multithreaded Processor" are three authors. Several CRC instruction set extensions for the Sand bridge multithreaded CPU were examined in this research. The processor is specifically designed to use SIMD vector operations and compound instructions to efficiently run multimedia programs and wireless communication. With the aid of computer programming intrinsic instructions, CRC methods were evaluated and improved. The instructions are incorporated into the Sand bridge compiler and speed-optimized to guarantee effective scheduling with other code elements. When the authors compared different CRC algorithms, they found that one technique—the 8-bit table look-up—was approximately seven times faster than another—the bit-by-bit CRC. They also looked into the use of CRC algorithms in hardware with the specialized programming language Verilog.

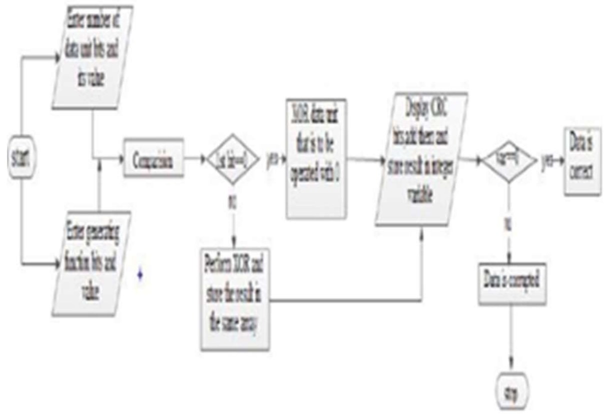
IV. METHODOLOGY:

One of the most dynamic error detection techniques based on the mechanism of binary division is Cyclic Redundancy Check (CRC), which we will be utilizing here. Here in the CRC, we basically use a sequence of bits and then append to the data unit, this is the overview process of the CRC which we are going to use. Further going deep into the CRC, there are two important steps which are being followed in the CRC techniques, they are the CRC generator and the CRC checker.

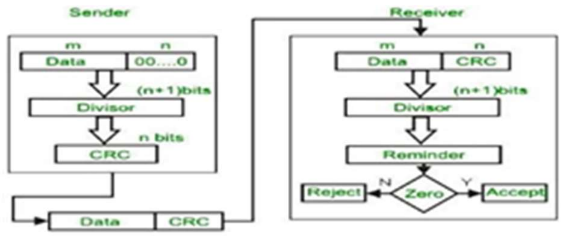


CRC Generator

So, in the CRC generator we would be calculating the CRC bits at the sender side by generating a function such that the given data unit and the generator function are operated using the bitwise XOR operation where the generating function is of the n+1 bits and the given CRC is of the n bits. The data is then being operated by appending with the n 0’s which together referred as data unit to be transmitted to the receiver end.



Then comes the CRC checker part, there in this part we would be calculating the CRC bits at the receiver side to check on whether there is an equivalent to 0 or not in the received signal. It also uses the similar bitwise XOR operation on the message which is being received from the sender and then generates a function according to the input it received. If all the CRC bits are being 0, then there are not errors and it accepts it, otherwise we have to change accordingly to get no error output. So, here in the CRC for the function generation we would be using the polynomial equation for the function generation the polynomial function must not be divisible by x and it should be divisible by x+1. The polynomial equation must not be divisible by x as it guarantees that it can detect the burst error with length equal to highest degree of polynomial, then it must be divisible by x+1 so that it guarantees that the burst error affecting odd bits are detected. So, this mathematical equation is used to generate the generative function which we are going to use in our CRC. There are some conditions to be satisfied here in the CRC, They are: The first condition ensures that CRC can effectively detect burst errors of a length equal to the highest degree of the polynomial used for computation. This means that CRC is capable of identifying consecutive errors within the message up to the length determined by the degree of the polynomial. The second condition is critical as it guarantees the detection of all burst errors affecting an odd number of bits. By selecting appropriate polynomial lengths and configurations, CRC can detect errors that affect an odd number of bits within the message. This ensures robust error detection capabilities, particularly for burst errors that may occur in communication channels.The choice of generating polynomial is indeed a crucial aspect of CRC error detection technique. The polynomial determines the properties of the CRC checksum and influences its error detection capabilities. Careful selection of the generating polynomial is essential to ensure that CRC can effectively detect errors within the transmitted data while minimizing false positives and negatives. So, in the CRC we have to take care that the whole generative function remains constant throughout the sending and receiving process in order to have integrity in our system and no issues.



Overview Block Diagram of CRC

E.g: Division Algorithm of CRC

V. RESULT:

The below are the results of the CRC which we have performed in order to show the efficiency of the system. Here for the result we are going to transmit the data and then perform the CRC on it and here we would be displaying the overall output of the system which indicates whether there is an error in the given code and how effectively the error is being detected using the CRC.

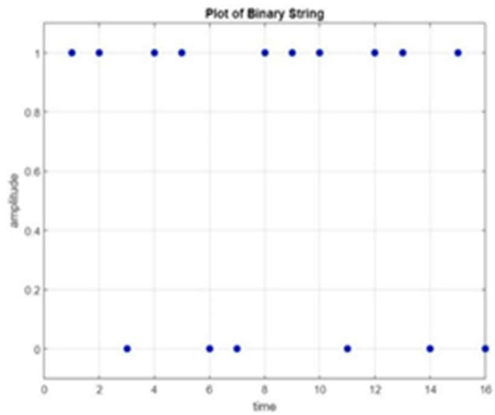
```
crc checkvalue
101
encoded message
1101100111011010101
message contains errors
>>
```

The below is the output and results for a message to which the errors are introduced, there is an detailed output which shows that there is an error and graphical representation for this error where the graph represents the binary string, the divisor and the output error we have obtained.

```
crc checkvalue
101
encoded message
1101100111011010101
message is error free
>>
```

The below is when the data which is being sent comes without any error in the channel transmission, the image below shows that there has been CRC applied on the data and it shows us that there is no error being detected in the received code at the receiver which means the transmitted code is received without errors.

VI. GRAPHS:



CONCLUSION

Cyclic Redundancy Check (CRC) emerges as the optimal choice for error detection in data communication due to its robust mathematical foundation and efficient algorithms, enabling it to detect a wide range of errors, including single-bit and burst errors, with high accuracy. Moreover, CRC's effectiveness extends to its ability to adapt to various communication environments, accommodating different data lengths and transmission speeds without compromising its error detection capabilities. Its versatility makes it an integral component in diverse systems, ranging from high-speed networks to embedded devices. Furthermore, CRC is resistant to typical transmission impairments like noise and interference because of its built-in error-detection capabilities. Its dependability and trustworthiness are further demonstrated by its extensive use in mission-critical applications like data storage and telecommunications. Because of its unrivaled performance and assurance of data integrity in a wide variety of communication scenarios, CRC is still the go-to method for error detection.

Thus, performing the experiment with the MATLAB provides us with an detailed analysis of how the CRC is going to function and what is the error detection it is going to do and it helps us to get better insights and better idea on how the CRC is proceeding and MATLAB plays an major role in the analysis and is very helpful.

FUTURE SCOPE

The CRC is one of the most efficient and effective ways which can be used by us for the error detection but there is a small defect in this CRC that is that though this CRC is very efficient and effective in the error detection for any time of the data, it lags behind in the error correction part which is a small drawback for our CRC as it cannot correct the error though it but it can detect any number of error. So, here we are going to come across some of the other coding techniques which could act as a better part for the error correction of the data. We can use the coding techniques such as the reed Solomon codes here in the reed Solomon codes, we would code in an algebraic way and use combination of the symbols and mathematical techniques which can we used for both error detection and the error correction. They encode blocks of data into longer codewords by adding redundant symbols and allow for the detection and the correction. This Reed Solomon code is an effective way for the error correction. So, what we can do is combine both of them that is the CRC and the Reed Solomon codes and the encoding part of the Reed Solomon code is complex we can use the CRC part for the error detection and as the CRC has only error detection part we can use the Reed Solomon code for the error correction as the Reed Solomon code is an effective code for the correction so now by combining both of these we can form an new technique which could be very effective for us and help in the communication process in an great and effective way and even can be an path breaking technique for the future uses.

REFERENCES

[1] Okechukwu, Michael & Osuagwu, Henry & Ahaneku, Mamilus. (2020). Performance Analysis of Cyclic Redundancy Check (CRC) Error detection Technique in the Wireless Sensor Network.

[2] Sooch, S.K., Gupta, M. and Kumar, R., 2020. Implementing Cyclic Redundancy Check asError Correction Technique in HDLC.

[3] Murade, R.T., Mujahid, M.M. and Sabir, M.A.M., 2013. The Design and Implementation of a Programmable Cyclic Redundancy Check (CRC) Computation Circuit Architecture Using FPGA. International Journal of Science and Modern Engineering (IJISME).

[4] C. Condo, M. Martina, G. Piccinini and G. Masera, "Variable Parallelism Cyclic Redundancy Check Circuit for 3GPP-LTE/LTE-Advanced," in IEEE Signal Processing Letters, vol. 21, no. 11, pp. 1380-1384, Nov. 2014, doi: 10.1109/LSP.2014.2334393.

[5] P. Kishore, B. A. Pal, L. Nanda Kishore and C. V. Revathi, "Implementation of Table-Based Cyclic Redundancy Check (CRC-32) for Gigabit Ethernet Applications," 2023 4th International Conference for Emerging Technology (INCET), Belgaum, India, 2023, pp. 1-4, doi: 10.1109/INCET57972.2023.10170336.

[6] Koopman, P., Driscoll, K. and Hall, B., 2015. Selection of cyclic redundancy code and checksum algorithms to ensure critical data integrity.

[7] N. Sridevi, K. Jamal and K. Mannem, "Implementation of Cyclic Redundancy Check in Data Recovery," 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2021, pp. 17-24, doi: 10.1109/ICESC51422.2021.9532802.

[8] J. Van Waes, J. Lannoo, A. Degraeve, D. Vanoost, D. Pissoort and J. Boydens, "Effectiveness of Cyclic Redundancy Checks under harsh electromagnetic disturbances," 2017 International Symposium on Electromagnetic Compatibility - EMC EUROPE, Angers, France, 2017, pp. 1-6, doi: 10.1109/EMCEurope.2017.8094763.

[9] M. Khosraviani, A. E. Pourzandi and M. R. Abbassian, "A Flexible Cyclic Redundancy Codes' Selection to Increase Fault-Tolerance in Smart Buildings," 2022 27th International Computer Conference, Computer Society of Iran (CSICC), Tehran, Iran, Islamic Republic of, 2022, pp. 1-6, doi: 10.1109/CSICC55295.2022.9780482