UDC - 519.688:004.42

STUDY OF CIRCUIT PARAMETERS OBTAINED USING ALGORITHMS FOR DETECTING AND CORRECTING ERRORS

N.V. Melikyan, A.M. Danielyan, A.G. Manukyan, A.A. Galstyan
National Polytechnical University of Armenia

Software errors and constantly reducing technologies have paved the way for the creation of software algorithms for detecting and correcting errors in embedded systems. The quality of the algorithm for detecting and correcting errors in such systems depends to a great extent on the behavioral description of this algorithm. Each algorithm for detecting and correcting errors is more efficient than the other. In this paper, two of the most common detector algorithms were studied, the circuit described by these algorithms was synthesized by SAED 14 and SAED 32 nanometer libraries and the parameters were evaluated.

Key words: integrated circuit, device description language, codeword, logical synthesis.

Introduction

Accurate data transmission and storage are of extreme importance nowadays. Hardware and software issues are the cause of data corruption. Unlike hardware problems which are similarly important, software problems may occur due to noise and radiation particles, which can lead to inaccurate data transmission and storage. Since these issues are impossible to avoid, circuits developed using algorithms for detecting and correcting errors which have been already used during data transmission for several decades [1]. The complexity of this task is that the circuits developed by these algorithms should work while the program is running, maintaining the smooth operation of systems such as automation systems, medical equipment, space stations and all systems that use data transmission.

The importance of accurate data transmission in information systems

Data is transmitted between two or more remote points through the transmitting channel, using the digital system. Noise in the transmission channel has a major impact on the quality, reliability, and accuracy of data (Fig. 1). To avoid the following issues, telecommunications systems use algorithms for correcting corrupted data [2]. The main issue is that even in the case of a software error, it is possible to save the data correctly. The data is stored in memory or transferred to another memory. Since these circuits should be used in such areas as automation, medical equipment and space stations, then accurate data transmission is of major importance (Fig. 2). Multiply developed algorithms, such as Hamming, Reed-Solomon, and Bose-Chaudhuri-Hocquenghem algorithms [3-4], are used to transmit and store data accurately. The rationale for this algorithm lies in the fact that when data is transmitted incorrectly, it should be detected and corrected while the program is running. It has been studied that the Reed-Solomon algorithm functions more effectively during data transmission compared to the Hamming one due to its simpler description of hardware. With the results obtained in this paper, we compare the parameters of the circuits obtained by the Reed-Solomon and Bose-Chaudhuri-Hocquenghem algorithms described in the hardware description language [5].

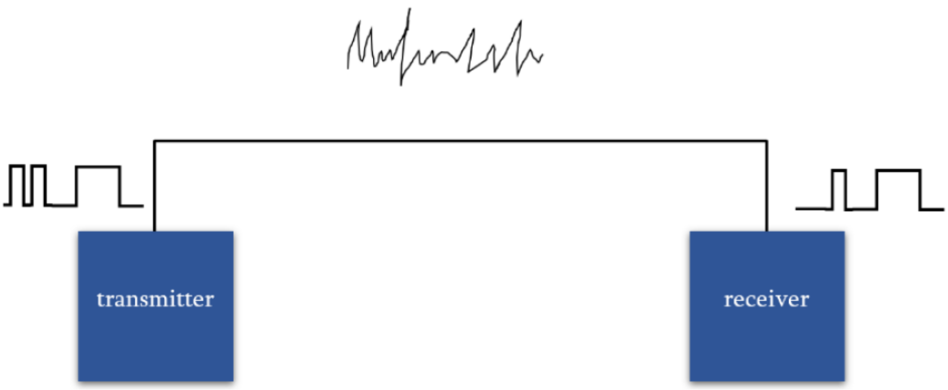

Fig. 1 The impact of noise during data transmission

The Reed-Solomon algorithm

The Reed-Solomon code is an error correction code introduced by Irving Reed and Gustav Solomon in 1960. It is widely used for correcting errors in mass memory systems. The Reed-Solomon code is the main component of the CD. There are two main forms of Reed-Solomon: the original one and the Bose-Chaudhuri-Hocquenghem (BCH) one. The appearance of the BCH is more generalized since decoders function faster and do not require large volumes. There are different procedures for encoding the Reed-Solomon code, that is, different ways to describe the codeword. Each codeword in the original code is represented as a sequence of functions. The codeword is obtained by presenting the message as follows:

$C=(p(a_1) ,p(a_2) , p(a_3), ..., p(a_n))$, where p is a polynomial of a degree less than k.

There are different forms of encoding. In the original X structure, the Reed-Solomon message (1960) is described by the coefficient of the polynomial p, and in the subsequent structure, the message is described by the values $a_1, a_2, \dots, a_k$ of the polynomial and the interpolation of the latter produces the p polynomial. Subsequent procedural encoding, although less efficient, still provides a systematic advantage of encoding in that the message is always contained in subsequent keywords.

Simple encoding procedure

Message as a sequence of coefficients. In the original structure, the Reed-Solomon message (1960) x= $(x_1, x_2, \dots x_k)$ is modified into the polynomial p_x :

$$p_x(a) = \sum_{i=1}^{k} x_i a^{i-1} \tag{1}$$

the codeword is p_x expressing itself in points of the Gaussian distribution $(a_1, \dots a_n)$.

$$C_x = (p_x(a_1), \dots p_x(a_n)) \tag{2}$$

The resulting C function is a linear transformation that satisfies $C_x = x * A$

$$A= \begin{bmatrix} 1 & \dots & 1 & \dots & 1 \\ a_1 & \dots & a_k & \dots & a_n \\ a_1^2 & \dots & a_k^2 & \dots & a_n^2 \\ \vdots & \dots & \vdots & \dots & \vdots \\ a_1^{k-1} & \dots & a_k^{k-1} & \dots & a_n^{k-1} \end{bmatrix} \tag{3}$$

A simple procedure for encoding the generated matrix.

Systematic coding procedure

A message as an initial sequence of values. In this procedure, the format of the polynomial p_x message is performed in a different way. The polynomial p_x is defined as a polynomial less than k.

$$p_x(a_i) = x_i \quad i \in \{1, \dots, k\}$$

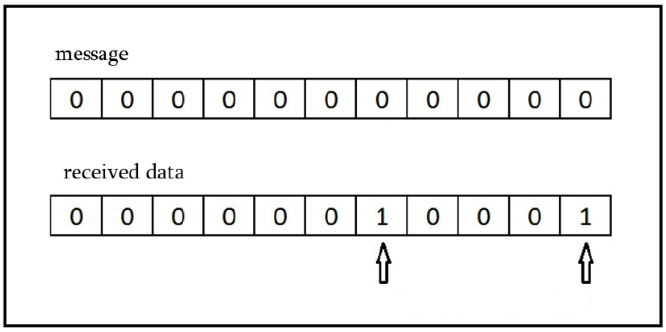

Fig. 2 The simple example of an error message received

Using Lagrange interpolation, we can calculate the polynomial p_x and express it in points of the Gaussian distribution: $a_{k+1}, \dots a_n$ we obtain the function C.

$$C_x = (p_x(a_1), \dots p_x(a_n))$$

Since Lagrange interpolation is a linear transformation, C is also linear C_x = x*G

$$G = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & g_{1,k+1} & \cdots & g_{1,n} \\ 0 & 1 & 0 & \cdots & 0 & g_{2,k+1} & \cdots & g_{2,n} \\ 0 & 0 & 1 & \cdots & 0 & g_{3,k+1} & \cdots & g_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & \cdots & 1 & g_{k,k+1} & \cdots & g_{k,n} \end{bmatrix} \quad (4)$$

G is the generated matrix of the system encoding procedure.

Reed-Solomon decoder

When transmitting, there will be a distortion of the transmitted message, a distortion of the codeword, due to noise.

$$R_x = C_x + E_x \quad (5)$$

C_x is a codeword and E_x is errors in it

$$E_x = e_{n-1}x^{n-1} + e_{n-2}x^{n-2} + \dots + e_1x^1 + e_0 \quad (6)$$

Bose-Chaudhuri-Hocquenghem algorithm

The decoder described by the Bose-Chaudhuri-Hocquenghem algorithm operated in the following sequence of steps: it calculates the distortion from the received codeword, from which the error locator polynomial is obtained, and the latter is used to correct corrupted bits. (Fig. 3).

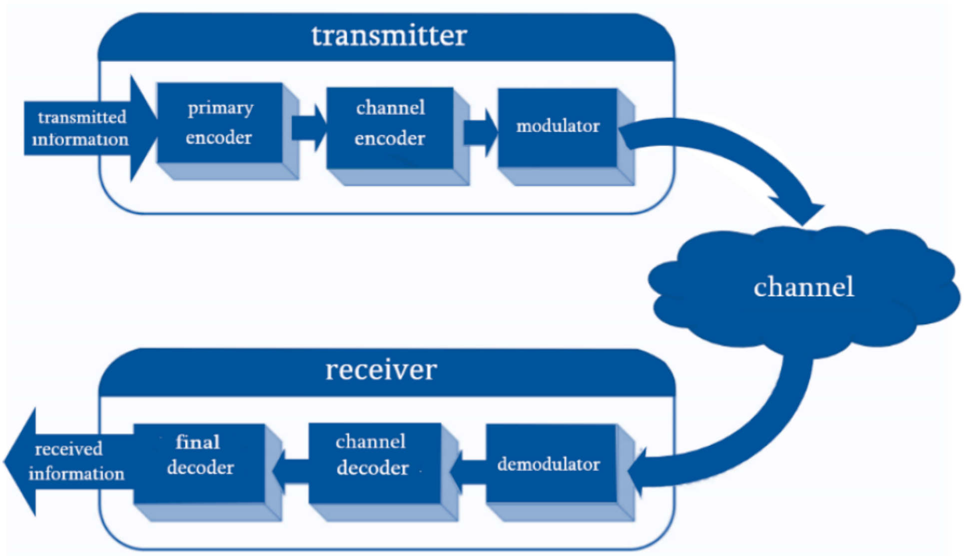

Fig. 3 Route of the Bose-Chaudhuri-Hocquenghem algorithm

Results and Discussion

In the course of the work using the description of algorithms presented in the hardware description language and in the Verilog language, a logical synthesis was performed using ready-made libraries of standard digital cells. Based on the outcomes, algorithms for detecting and correcting errors were evaluated and compared with various input data transmitted by the bitstream. Table 1 shows the expected results on the surface, expressed in μm^2, SAED 14 and 32 nanometers for the process [6-7].

Table 1

Area values of circuits

Bit	SAED 14 nm technology		SAED 32 nm technology	
	Reed-Solomon	Bose-Chaudhuri-Hocquenghem	Reed-Solomon	Bose-Chaudhuri-Hocquenghem
8	1,741.5	602.6	5,466.18	1,995.8
16	3,105.7	1,077.4	9,871.3	3,860.7
32	5,488	1,981.6	18,044.8	7,581.6
64	10,391.8	3,869.8	37,104.8	13,488.5

The table shows that the circuits developed using the Bose-Chaudhuri-Hocquenghem algorithm have approximately 2.7 times the smaller surface area.

Fig. 4 and 5 show the power of circuits synthesized by 14 and 32 nanometer process libraries, respectively which proves that the power consumed when using the Bose-Chaudhuri-Hocquenghem algorithm is 30% more efficient.

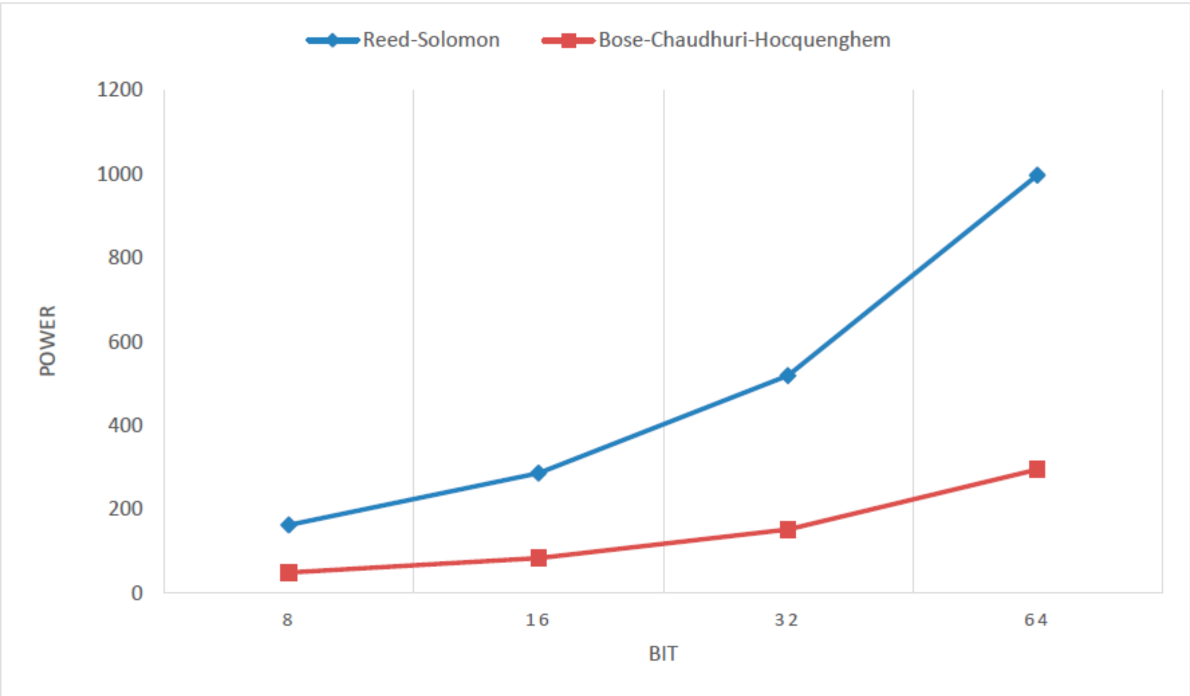

Fig. 4 The power values using 14 nm SAED technology

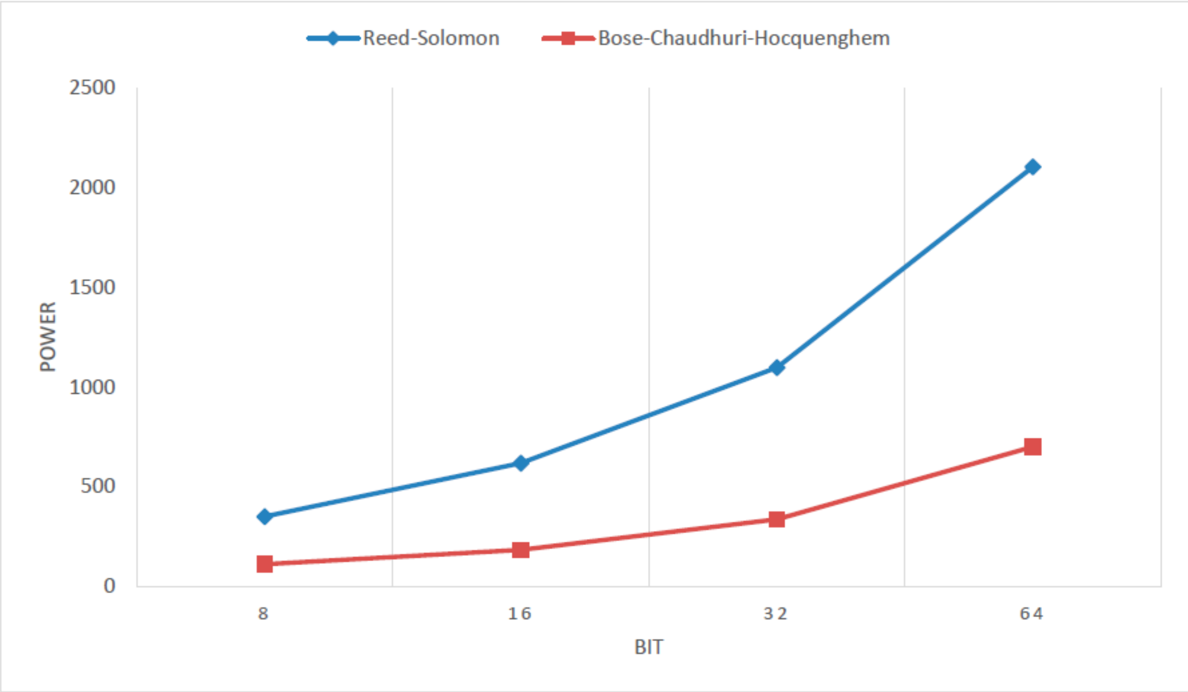

Fig. 5 The power values using 32 nm SAED technology

Conclusion

The purpose of this paper was to study two well-known algorithms for detecting errors, the Reed-Solomon on and the Bose-Chaudhuri-Hocquenghe one as well as to study their application in data transmission systems. The result shows that, in contrast to the Reed-Solomon algorithm, the circuit described by the Bose-Chaudhuri-Hocquenghe algorithm occupies a smaller surface area and consumes less energy due to the data decoder.

References

1. Lee H.P., Chang H.C., Lin T.C., Truong T.K. A Weight Method of Decoding the Binary BCH Code // IEEE - 2008. - P. 545-548.
2. Abildgren R., Moullenc Y., Analyzing and Implementing a Reed-Solomon Decoder for Forward Error Correction in ADSL // Aalborg University - 2007. P. 45-53.
3. Okeke C., Eng M., A Comparative Study between Hamming Code and Reed-Solomon Code in Byte Error Detection and Correction // IJRASET - 2015. P. 34-38.
4. Lone F. R., Puri A., Kumar S. Performance Comparison of Reed Solomon Code and BCH Code over Rayleigh Fading Channel // IEEE - 2013. P. 23-26.
5. Chang H., Cheng C. Reed-Solomon Decoder using Decomposed Euclidean Algorithm // IEEE - 2000. P. 262-265.
6. Melikyan V., Martirosyan M., Melikyan A., Piliposyan G. 14nm Educational Design Kit: Capabilities, Deployment and Future // Small Systems Simulation Symposium Niš, Serbia, - February, 2018. - P. 37-41.
7. Goldman R., Bartleson K., Wood T., Kranen K., Melikyan V., Babayan E. 32/28nm Educational Design Kit // Capabilities, deployment and future, Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia), IEEE, - 2013.

ՀՏԴ - 519.688:004.42

ՍԽԱԼԻ ՀԱՅՏՆԱԲԵՐՄԱՆ ԵՎ ՈՒՂՂՄԱՆ ԱԼԳՈՐԻԹՄՆԵՐՈՎ ՍՏԱՑՎԱԾ ՍԽԵՄԱՆԵՐԻ ՊԱՐԱՄԵՏՐԵՐԻ ՈՒՍՈՒՄՆԱՍԻՐՈՒԹՅՈՒՆ

Ն.Վ. Մելիքյան, Ա.Մ. Դանիելյան, Ա.Գ. Մանուկյան, Ա.Ա. Գալստյան
Հայաստանի ազգային պոլիտեխնիկական համալսարան

Ծրագրային սխալները և անընդհատ փոքրացող տեխնոլոգիաները ճանապարհ են հարթել ծրագրային սխալների հայտնաբերման և ուղղման ալգորիթմների ստեղծմանը ներդրված համակարգերում: Այդպիսի համակարգերում սխալների հայտնաբերման և ուղղման ալգորիթմի բարդության աստիճանը կախված է դրա վարքային նկարագրությունից: Աշխատանքում ուսումնասիրվել է ամենատարածված դետեկտորային ալգորիթմներից երկուսը, կատարվել է այդ ալգորիթմների նկարագրությամբ ՍԱՈՒԴ 14 և 32 նանոմետրանոց տեխնոլոգիական գրադարաններով սինթեզված սխեմաների ուսումնասիրություն, պարամետրերի գնահատում և համեմատում:

Բանալի բառեր. ինտեգրալ սխեմա, սարքակազմի նկարագրման լեզու, ինտերպոլացիա, բառ-կոդ, տրամաբանական սինթեզ:

УДК - 519.688:004.42

ИЗУЧЕНИЕ ПАРАМЕТРОВ СХЕМ ПОЛУЧЕННЫХ С АЛГОРИТМАМИ ОБНАРУЖЕНИЯ И ИСПРАВЛЕНИЯ ОШИБКИ

Н.В. Меликян, А.М. Даниелян, А.Г. Манукян, А.А. Галстян
Национальный политехнический университет Армении

Программные ошибки и постоянно уменьшающиеся технологии проложили путь к созданию алгоритмов обнаружения и исправления программных ошибок во встроенных системах. Степень сложности алгоритма обнаружения и исправления ошибок в таких системах зависит от его поведенческого описания. В данной работе были изучены два из наиболее распространенных алгоритмов детектирования, проведено исследование схем с описанием этих алгоритмов, синтезированных для технологий САУД 14нм и САУД 32нм, оценка и сравнение параметров.

Ключевые слова: интегральные схемы, язык описания аппаратуры, интерполяция, кодовое слово, логический синтез.

Ներկայացվել է՝ 08.03.2020թ.
Գրախոսման է ուղարկվել՝ 15.04.2020թ.
Երաշխավորվել է տպագրության՝ 16.04.2020թ.