Performance Comparison of New Designs of Chien Search and Syndrome Blocks for Bch and Reed Solomon Codes

Mohamed Elghayyaty[1], Omar Mouhib[1], Azeddine Wahbi[2], Ahmed Amine Barakate[1], Anas El Habti El Idrissi[1], Laamari Hlou[1] and Abdelkader Hadjoudja[1]

[1]Laboratory of Electrical Engineering and Energy System, Faculty of Sciences, University Ibn Tofail, Kenitra, Morocco
[2]Laboratory of Industrial Engineering, data processing and logistic, Faculty of sciences Ain Chock, University Hassan II, Casablanca, Morocco

***Abstract*:** Error correcting codes constitute one of the core technologies in telecommunications field, especially digital communication applications. The objective of this paper is to compare performance among new designs of chien search block on the one hand and syndrome architectures on the other hand in error correcting codes. All comparison of all designs is made by computing the number of logic, bit error rate values and number of iteration in the case of syndrome architectures

Analysis results show that the performances of the new designs based on both second factorization method and Three-Parallel Syndrome architecture are superior to the performances of traditional designs.

***Keywords*:** Error Correcting Codes, Error Locator Polynomial, Chien Search Block, Syndrome Computation Block, Bit Error Rate (BER).

1. Introduction

The quality of a digital transmission depends mainly on the quantity of errors introduced into the transmission channel [1] [2]. Error checking by coding is therefore essential. The use of digital signal processing techniques, and in particular the coding of the information to be transmitted, allows the detection and correction of transmission errors **[**3] [4]. As these techniques make it possible to control the errors induced by the noise of the transmission channel, they are called "channel coding". Among the main techniques used, block coding and convolution coding are predominant. Blocks coding are used in particular in Ethernet networks, in wireless transmission standards such as Bluetooth, and in HDTV (High Definition Television) and DVB-C (Digital Video Broadcasting-Cable) transmission standards. Convolution coding is very common in digital wireless communication systems [5] [6] [7] and other codes like RS (Reed-Solomon), and BCH (Bose, Ray-Chaudhuri) [8].

The Reed Solomon codes are block and not binary codes where the message is divided into blocks to which is added redundant information. The length of the blocks depends on the capacity of this code. For each block on the addition of protection bits or of the additional parity for the old code, a word of n symbols. It is also a systematic code, i.e. the control symbols are added at the end of the information. The coder takes K data symbols and calculates the control information to construct n symbols, which gives n-k control symbols. The decoder can correct at most t symbols, or 2t = n-k. There are more efficient codes for detecting and correcting errors, for example, the RS code (255,233.33) which is used by NASA in space communications and also in a digital terrestrial transmission chain (DVB-T).

The BCH codes (Bose, Ray-Chaudhuri, Hocquenghem), were invented in the 1960, they are cyclic codes and today they are used as a reference for many recent error correction codes. . These are relatively efficient codes, simple to implement and for which there is a set of low complexity algebraic decoding algorithms. The encoder / decoder assembly makes it possible to construct a cyclic code and to correct a number of t errors in a block of n coded symbols transmitted. Understanding the concepts of algebra in the Galois Field (GF) is necessary to better understand these codes. The block length of the BCH code built on GF (2^m) is given by $n = 2^m - 1$. Their uses are in particular in communications by satellite DVB-S2, compact discs CD and DVD.

A number of methods have been proposed to enhance the performance of the chien search block and syndrome architecture [9] [10] [11] [12] [13].

The goal of this work is to present a comparison analysis of the new chien search and new syndrome block with others designs in order to enhance these performances [14] [15] [16] [17].

Finally, the obtained results indicate the proposed designs as having high performance. All designs are evaluated in terms of BER, number of iterations and logic gate resources [18].

The rest of the paper is organized as follows: The Proposed robust factorization methods for RS and BCH decoder are presented in section 2. The syndromes architecture is presented in the session 3, Performance Analysis of the designed circuits are presented in section 3, followed by a conclusion.

2. Proposed Robust Factorization Methods for RS and BCH Decoder

The proposed algorithm is based on a specific and methodic factorization of the error locator polynomial such as {(P (x) = Ax^n + B, where n = 1)} should be depicted in this polynomial. This method allows us to conceive a new circuit of Chien Search Block i.e. we can minimize both the number of components and the response time compared to the basic Chien search block.

2.1 Design using the First factorization method

If we take the case of RS (15, 11, t) the error locator polynomial is:

$$\Lambda(x) = 14X^2+14X+1 \quad (1)$$

It's a polynomial of degree 2 as type:

$$\Lambda(x) = A_2X^2+A_1X+A_0 = (AX+B)(AX+C) \quad (2)$$

The equation (2) can be written as form:

$$\Lambda(x) = (\alpha X+\beta)(\alpha X+\gamma) = \alpha^2X^2 + \alpha X\gamma + \alpha X\beta + \beta\gamma$$
$$= \alpha^2X^{2+}\alpha(\gamma+\beta)X+\beta\gamma \quad (3)$$

Or $A_2 = \alpha^2$, $A_1 = \alpha(\gamma+\beta)$ and $A_0 = \beta\gamma$

For equation (2) the basic circuit corresponding is represented in Figure 1.

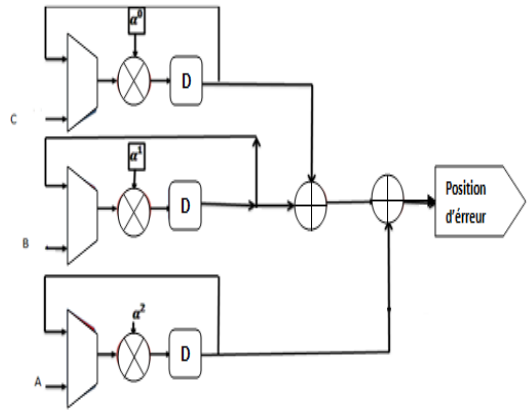

Figure 1. Basic schema of Chien Search.

For the equation (3) the modified circuit by using the factorization method is represented in Figure 2.

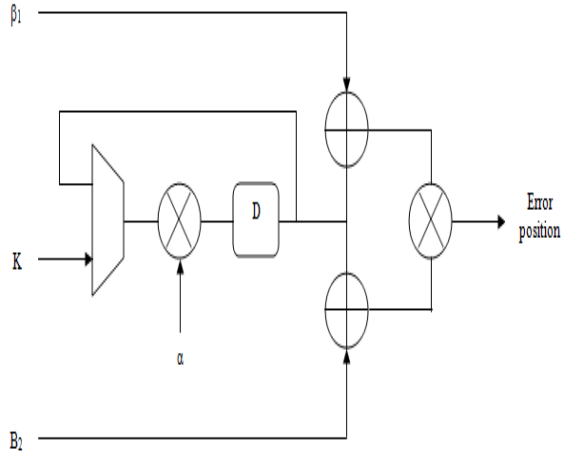

Figure 2. Modified circuit of Chien Search

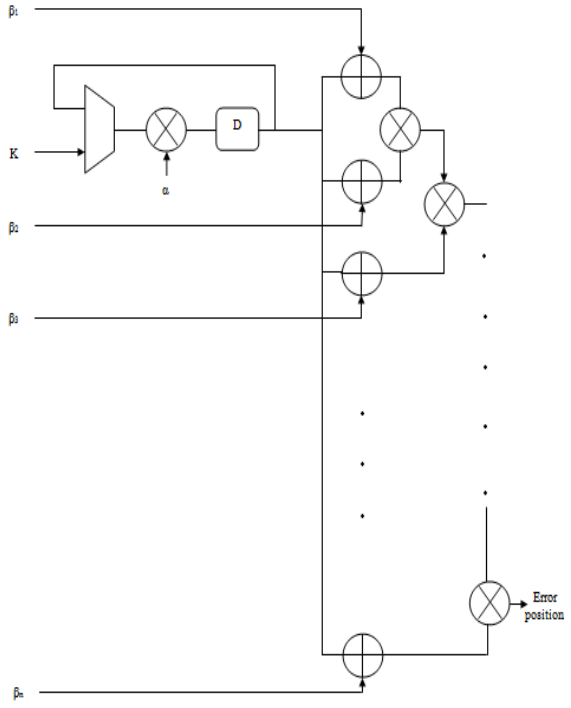

Figure 3. Modified circuit of Chien Search

For the case of a polynomial of degree 3 we have:

$$\Lambda(X) = (\alpha X+\beta)(\alpha X+\gamma)(\alpha X+\lambda)$$
$$= (\alpha^2X^2 + \alpha(\gamma+\beta)X + \beta.\gamma)(\alpha X+\lambda) \quad (4)$$

Generally for:

$$\Lambda(X) = (\alpha X+\beta)(\alpha X+\gamma)\ldots(\alpha X+\nu) \quad (5)$$

The corresponding logic circuit is represented in Figure 3.

2.2 Design using the second factorization method

- **Case for odd polynomial**

If we take the polynomial of degree 3 we have:

$$\Lambda(X) = AX^3+BX^2+CX+D = X^2(Ax+B)+(Cx+D) \quad (6)$$

If we take the polynomial of degree5 we have:

$$\Lambda(X) = AX^5+BX^4+CX^3+DX^2+EX+F$$
$$= X^4(Ax+B)+X^2(Cx+D)+(EX+F) \quad (7)$$

If we take the polynomial of degree7 we have:

$$\Lambda(X) = AX^7+BX^6+CX^5+DX^4+EX^3+FX^2+GX+H$$
$$= X^6(Ax+B)+X^4(Cx+D)$$
$$+X^2(EX+F)+(GX+H) \quad (8)$$

If we take the polynomial of degree 9 we have:

$$\Lambda(X) = AX^9+BX^8+CX^7+DX^6+EX^5+FX^4+GX^3+HX^2+IX+J$$
$$= X^8(AX+B)+X^6(CX+D)+X^4(EX+F)$$
$$+X^2(GX+H)+(IX+J) \quad (9)$$

If we take the polynomial of degree n we have:

$$\Lambda(X) = A_nX^n + A_{n-1}X^{n-1} + \ldots + A_1X^1 + A_0 \quad (10)$$

- **Case for even polynomial**

If we take the polynomial of degree 4we have:

$$\Lambda(X) = AX^4+BX^3+CX^2+DX+E$$
$$= X^3(AX+B)+X(CX+D)+E \quad (11)$$

If we take the polynomial of degree 6 we have:

$$\Lambda(X) = AX^6+BX^5+CX^4+DX^3+EX^2+FX+G = X^5(AX+B)+X^3(CX+D)+X(EX+F)+G \quad (12)$$

If we take the polynomial of degree 8 we have:

$$\Lambda(X) = AX^8+BX^7+CX^6+DX^5+EX^4+FX^3+GX^2+HX+I$$
$$= X^7(AX+B)+X^5(CX+D)+X^3(EX+F)+X(GX+H)+I \quad (13)$$

If we take the polynomial of degree 10 we have:

$$\Lambda(X) = AX^{10}+BX^9+CX^8+DX^7+EX^6+FX^5+GX^4+HX^3+IX^2+JX+K$$
$$= X^9(AX+B)+X^7(CX+D)+X^5(EX+F)+X^3(GX+H)+X(IX+J)+K \quad (14)$$

If we take the polynomial of degree n we have:

$$\Lambda(X) = A_nX^n + A_{n-1}X^{n-1} + \ldots + A_1X^1 + A_0 \quad (15)$$

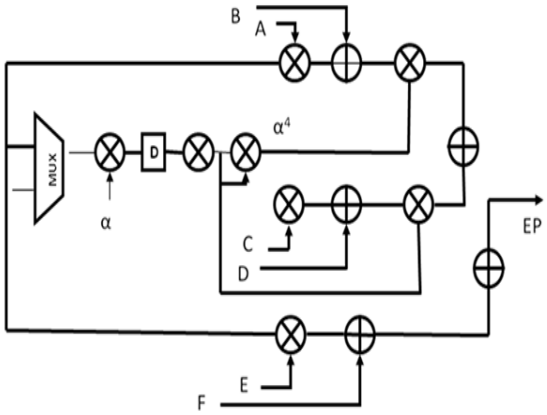

Figure 4. Modified circuit 2 of Chien Search Block polynomial as degree 5

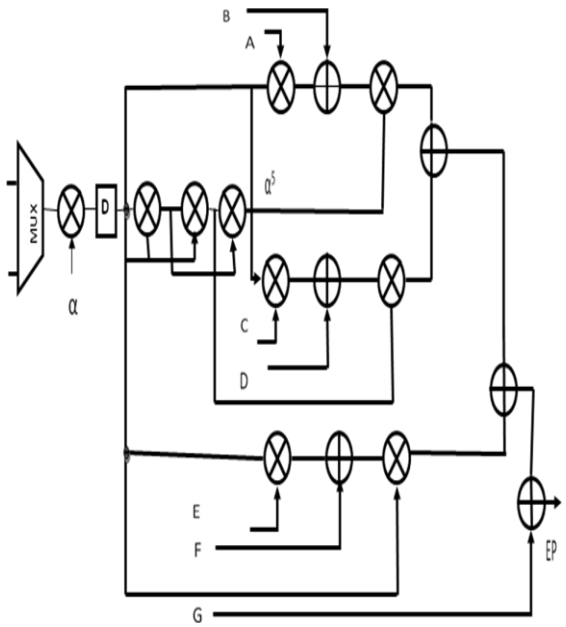

Figure 5. Modified circuit 2 of Chien Search Block polynomial as degree 6

3. Syndromes Architecture

The calculation of the syndrome is defined as the remainder of the division between the received polynomial R(x) and the generator polynomial g (x). The rest indicated the presence of errors. As the division operation is always a complex operation in relation to sums and additions, we are led to look for another method for calculating the syndrome.

The calculation of the syndrome can also be carried out by an iterative process as illustrated in Figure 6. Before being able to calculate the polynomial of the syndrome, we must wait until we have received all the elements of the polynomial R (x).

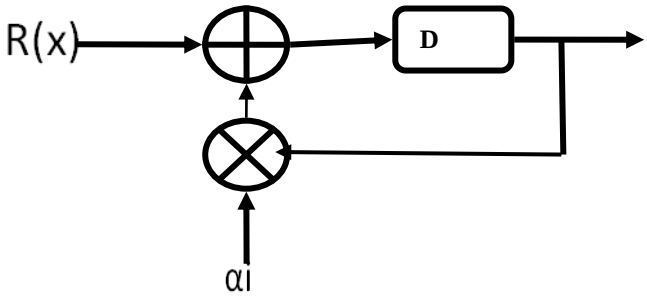

Figure 6. syndrome calculator cell

Given the binary codeword polynomial:

$$C(x) = C_0+C_1x+\cdots+C_{n-1}x^{n-1} \quad (16)$$

The received polynomial $R(x) = R_0+R_1x+\cdots+R_{n-1}x^{n-1}$ is the sum of C (*x*) and E(*x*)

$$R(x) = C(x) + E(x) \quad (17)$$

Where:

$$E(x) = E_0+ E_1x+\cdots+E_{n-1}x^{n-1} \quad (18)$$

We obtain:

$$S_i = R(\alpha^i) = C(\alpha^i) + E(\alpha^i) = E(\alpha^i) \quad (19)$$

Or:

$$M(x).X^{n-k} = g(x).q(x) + X(x) \quad (20)$$

$$M(x).X^{n-k} + X(x) = g(x).q(x) \Rightarrow C(x) = g(\alpha^i).q(\alpha^i) = 0$$

Then:

$$S_i = R(\alpha^i) = E(\alpha^i) = Y_1\alpha^{ie1}+ Y_2\,\alpha^{ie2} + \ldots + Y\nu\,\alpha^{i\,e\nu}$$

$$= Y_1X^i_1 + Y_2X^i_2 + \ldots + Y\nu X^i\nu \quad (21)$$

Where $X_1 = \alpha^{e1}$, $X_2 = \alpha^{e2}$……..$X\nu = \alpha^{e\nu}$

By the equation (21) we can define the different equations which link the polynomial Syndrome errors:

$$S_i = Y_1X^i_1 + Y_2X^i_2 + \ldots + Y\nu X^i\nu$$

$$S_0 = Y_1X^0_1 + Y_2X^0_2 + \ldots + Y\nu X^0\nu$$

$$S_1 = Y_1X^1_1 + Y_2X^1_2 + \ldots + Y\nu X^1\nu$$

$$S_2 = Y_1X^2_1 + Y_2X^2_2 + \ldots + Y\nu X^2\nu$$

The General equations of syndromes are:

$$\begin{bmatrix} s_0 \\ s_1 \\ . \\ . \\ . \\ s_{2t-1} \end{bmatrix} = \begin{bmatrix} X_1^0 & X_2^0 & . & . & . & X_\nu^0 \\ X_1^1 & X_2^1 & . & . & . & X_\nu^1 \\ . & . & . & . & .. & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ X_1^{2t-1} & X_1^0 & . & . & . & X_\nu^{2t-1} \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \\ . \\ . \\ . \\ Y_\nu \end{bmatrix} \quad (22)$$

If the received code r (x) is not affected by errors then all the coefficients of the Syndrome will be zero (r (x) = c (x)).

3.1 Methods for Calculing the syndromes

The syndrome computation block calculates all the syndromes Si ($1 \le i \le 16$) by putting the roots of generator polynomial g(x) into the received codeword polynomial R(x).

For example the code RS (15, 11): 2t=4 syndrome (S0, S1, S2,S3)

$$S(x) = S_3x^3 + S_2x^2 + S_1x^1 + S_0x^0 \quad (23)$$

Direct Method:

The equation for calculing the syndrome with the direct method [19] is:

$$S_i = R(\alpha^i) = R_{n-1}(\alpha^i)^{n-1}+ R_{n-2}(\alpha^i)^{n-2}+\ldots+ R_1(\alpha^1)+R_0 \quad (24)$$

For calculing S_1 we replace x by α_1 in the following equation:

$$S_0 = (\alpha^0)^{14} +2(\alpha^0)^{13} + 3(\alpha^0)^{12} + 4(\alpha^0)^{11} + 5(\alpha^0)^{10} + 11(\alpha^0)^9 + 7(\alpha^0)^8 +8(\alpha^0)^7 + 9(\alpha^0)^6 + 10(\alpha^0)^5 + 11(\alpha^0)^4 + 3(\alpha^0)^3 + (\alpha^0)^2 + 12(\alpha^0) + 12$$

$S_0=15$

$$S_1 = (\alpha^1)^{14} + 2(\alpha^1)^{13} + 3(\alpha^1)^{12} + 4(\alpha^1)^{11} + 5(\alpha^1)^{10} + 11(\alpha^1)^9 + 7(\alpha^1)^8 +8(\alpha^1)^7 + 9(\alpha^1)^6 + 10(\alpha^1)^5 + 11(\alpha^1)^4 + 3(\alpha^1)^3 + (\alpha^1)^2 + 12(\alpha^1) + 12$$

$S_1 = 3$

Horner Method:

The equation for calculing the syndrome with the Horner Method [20] is:

$$S_i = (\ldots\ldots(R_{n-1}\alpha^i + R_{n-2})\alpha^i+\ldots+ R_1)\alpha^1+R_0 \quad (25)$$

$S_2 = ((((((((((((1\ \alpha^2 + 2).\alpha^2 + 3)\ \alpha^2 + 4).\alpha^2 + 5)\ \alpha^2 + 11).\alpha^2 + 7).\alpha^2 + 8)\ \alpha^2 + 9)\ \alpha^2 + 10).\alpha^2\ + 11)$

$S_2 = 4$

$S_3 = ((((((((((((1\ \alpha^3 + 2).\alpha^3 + 3)\ \alpha^3 + 4).\alpha^3 + 5)\ \alpha^3 + 11).\alpha^3 + 7).\alpha^3 + 8)\ \alpha^3 + 9)\ \alpha^3 + 10).\alpha^3\ + 11)$

$S_3 = 12$

$S(x) = S_3x^3 + S_2x^2 + S_1x^1 + S_0x^0$

$S(x) = 12x^3 + 4x^2 + 3x^1 + 15x^0$

The following figure shows the principle of calculation of the four syndromes at the same time:

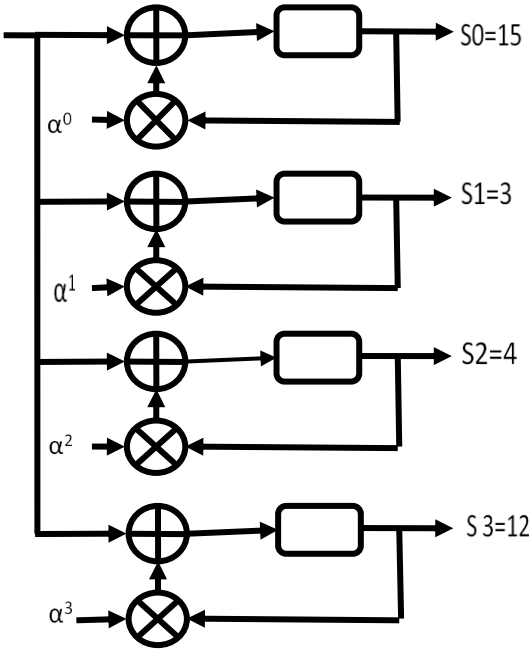

Figure 7. Logic circuit of the RS syndrome block (15, 11)

3.2 Serial Architecture

The serial syndrome computation block is implemented by following equation 4.

$S_i = R(\alpha^i) = r_{254}(\alpha^i)^{254} + r_{253}(\alpha^i)^{253} + \ldots\ldots r_1(\alpha^i) + r_0$ (26)

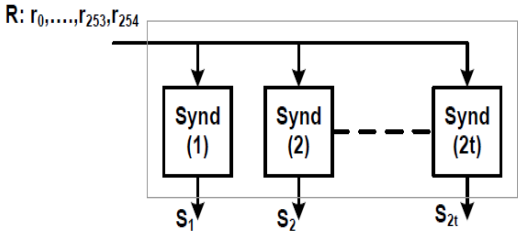

Figure 8. Serial Syndrome Block Diagram

The received vector enters serially to the circuit to compute the syndrome coefficients in parallel in n or (255) clock cycles, as shown in Figure 8.

For the case of RS (255, 239,) where: N = 255, K = 239 we have N-K = t =16=> 16 Syndromes: S1, S2, …, S16.

S1 can be calculated by the direct method and the even syndromes can be calculated using the following relationship $S_{2i} = Si2$.

3.3 Three-Parallel Syndrome

Three -parallel syndrome computation block [ref] can be expressed by the equation presented in equation. 4. Which equivalent to equation.4.1 but in another form.

$R(x) = r_{254}x^{254} + r_{253}x^{253} + \ldots\ldots r_1x + r_0$ (27)

$G(x) = (x-\alpha^0)(x-\alpha^1) \ldots\ldots\ldots (x-\alpha^{14})(x-\alpha^{15})$ (28)

$Si = R(\alpha^i) = r_{254}(\alpha^i)^{254} + r_{253}(\alpha^i)^{253} + \ldots\ldots r_1(\alpha^i) + r_0$ (29)

Where: i = 1, 2, . . ., 2t

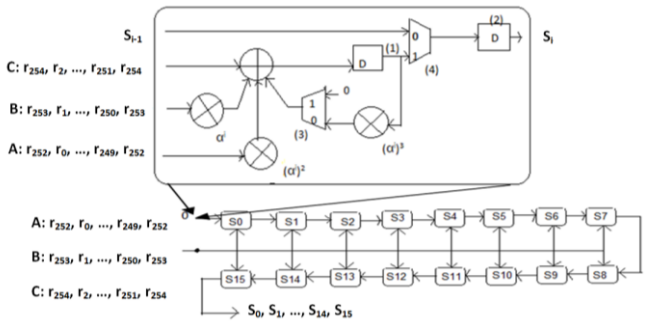

Figure 9. Three parallel syndrome computation block

The Three parallel syndrome computation block calculated by this equation:

$S_i = R(\alpha_i) =$

$((\ldots(r_{254}(\alpha^i)^2 + r_{253}(\alpha^i)^1 + r_{252})(\alpha^i)^3 + r_{251}(\alpha^i)^2 + r_{250}(\alpha^i)^1 + r_{249})$

$(\alpha^i)^3 + \ldots.)(\alpha^i)^3 + r_2(\alpha^i)^2 + r_1(\alpha^i)^1 + r_0)$ (30)

In the first clock, the mot received is (r254, r253, r252) in parallel, the second mot r_{252} is going to adder, r_{523} is multiplied by α^i and added to r_{252},the final r_{254} is multiplied by $(\alpha^i)^2$ and added to($r_{253}*(\alpha^i)^1 + r_{251}$). So the output of latch(1) is : $r_{254}(\alpha^i)^2 r_{253}*(\alpha^i)^1 + r_{251}$.the second clock the output of latch(1) is multiplied by $(\alpha^i)^3$ and then added with $r_{251}(\alpha^i)^2 + r_{250}(\alpha^i)^1 + r_{249}$

4. Performance Analysis of the Designed Circuits

4.1 performance and Comparison of designs based factorization methods

The proposed circuits are compared in terms of BER performance and Number of iterations, and number of minimized logic gates.

Tables 1, 2 and 3, also Figure 10, 11, 12, 13, 14 and 15 show the evaluation of the performance based on the number of the used logic gates and the obtained values of bit error rate BER for all designs by using the modified circuit for different error locator polynomials, especially in the case of second factorization method both odd error locator and even error locator.

Table 1. Computational minimization rate logic gates and BER values of the design based first factorization method

Error locator polynomial	Minimization rate $\left(\frac{Nm}{Nb} X100\right)\%$	BER $\left(\frac{100 - Mr}{100} X10x^{-4}\right)$
$\Lambda(X) = AX^2 + BX + C$	45, 45	0,000054
$\Lambda(X) = AX^3 + BX^2 + CX + D$	46,66	0,000053
$\Lambda(X) = AX^4 + BX^3 + CX^2 + DX + E$	47,36	0,000052
. . .	. . .	. . .
$\Lambda(X) = A_nX^n + \ldots\ldots + A_1X + A_0$	$\left(\frac{2n+1}{3+4n}.100\right)\%$	$\left(\frac{100 - Mr}{100} X10x^{-4}\right)$

Table 2. Computational minimization rate logic gates and BER values of the design based second factorization method with odd error locator polynomial

Error locator polynomial	Minimization rate $\left(\frac{Nm}{Nb}\ X100\right)\%$	BER $\left(\frac{100-Mr}{100}\ X10x^{-4}\right)$
$\Lambda(X) = AX^3+BX^2+CX+D$	33 ,33	0.000066
$\Lambda(X) = AX^5+BX^4+CX^3+DX^2+EX+F$	34,78	0 .000065
$\Lambda(X)=AX^7+BX^6+CX^5+DX^4+EX^3+FX^2+GX+H$	35,48	0.000064
. . .	. . .	. . .
$\Lambda(X)=A_nX^n+\ldots\ldots+A_1X+A_0$	$\left(\frac{2n-(n-1)/2}{3+4n}\ X100\right)$	$\left(\frac{100-Mr}{100}\ X10x^{-4}\right)$

Table 3. Computational minimization rate logic gates and BER values of the design based second factorization method with even error locator polynomial

Error locator polynomial	Minimization rate $\left(\frac{Nm}{Nb}\ X100\right)\%$	BER $\left(\frac{100-Mr}{100}\ X10x^{-4}\right)$
$\Lambda(X)=AX^4+BX^3+CX^2+DX+E$	31,57	0,000068
$\Lambda(X)=AX^6+BX^5+CX^4+DX^3+EX^2+FX+G$	33, 33	0,000066
$\Lambda(X)=AX^8+BX^7+CX^6+DX^5+EX^4+FX^3+GX^2+HX+I$	34 ,28	0,000065
. . .	. . .	. . .
$\Lambda(X)=A_nX+\ldots..A_1X+A_0$	$\left(\frac{3n/2}{3+4n}\ X100\right)$	$\left(\frac{100-Mr}{100}\ X10x^{-4}\right)$

These tables provide the necessary information about the performance of each design in terms of Number of minimized logic gates and the bit error rate (BER).

During testing from all designs, we can see that there is a significant decrease of number of minimized logic gates after enhancement is applied (for both proposed designs). Minimization rate for the second factorization method is over 34% or 35%, and for the first factorization method varies from 45% to 47,36%. Therefore, although for both presented figures 10, 11, 12, 13, 14 and 15, also table 1, 2 and 3 comparisons, we can say that both proposed designs, especially, the one based first factorization are adequate solutions for BCH and RS decoders and give slightly better results.

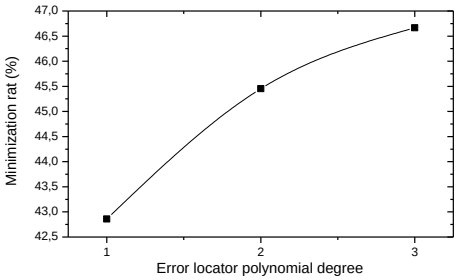

Figure 10. Evolution of the minimization rate depending on the degree of the error locator polynomial for the modified circuit based first factorization method

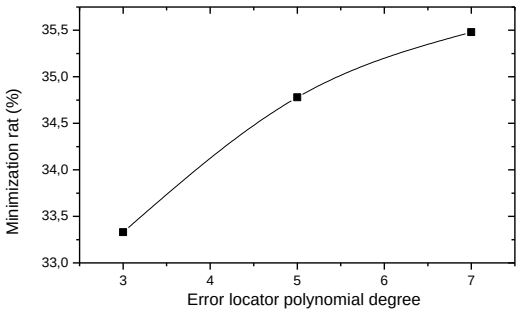

Figure 11. Evolution of the minimization rate depending on the degree of the odd error locator polynomial for the modified circuit based second factorization method

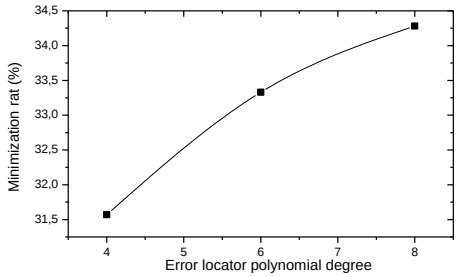

Figure 12. Evolution of the minimization rate depending on the degree of the even error locator polynomial for the modified circuit based second factorization method

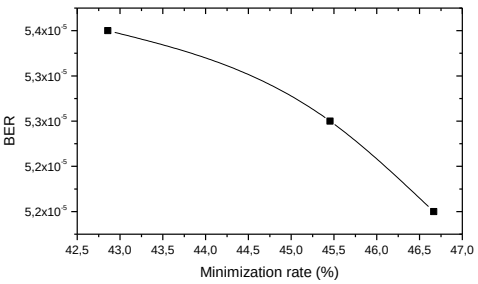

Figure 13. Evolution of the bit error rate BER depending on the minimization rate for the modified circuit based first factorization method

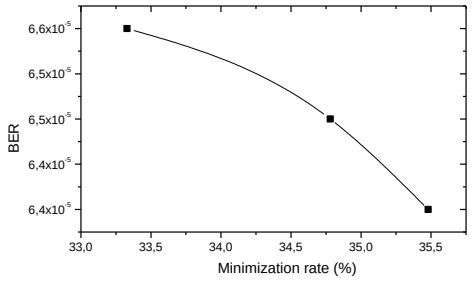

Figure 14. Evolution of the bit error rate BER depending on the minimization rate for the modified circuit based second factorization method with odd error locator polynomial

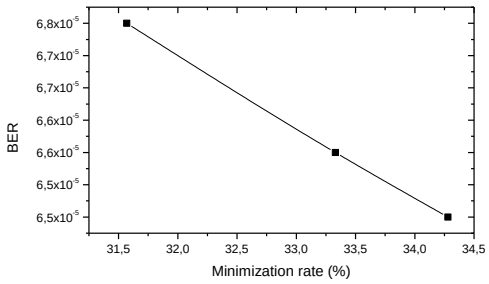

Figure 15. Evolution of the bit error rate BER depending on the minimization rate for the modified circuit based second factorization method with even error locator polynomial

4.2 performance and Comparison of the serial and parallel Circuits for the syndrome Computation architectures

Let's discuss now the performances of different types of Syndrome Computation Architectures by using number of iteration criteria. In this context, Table 5 shows the effect of the number of iteration to the performance of the proposed designs.

In this table 4 shows the number of the gained iterations by using the basic and the modified circuit for different RS and BCH codes.

Table 4. Computational minimization rate of logic gates and number of iterations of the serial and parallel Circuits for the syndrome Computation block

Code N°	Codes RS or BCH	Number of iterations serial Circuit	Number of iterations parallel circuit	Number of gained iterations	Minimization rate
1	(63, 55)	64	22	42	65.62
2	(63, 47)	64	22	42	65.62
3	(63, 39)	64	22	42	65.62
4	(63, 31)	64	22	42	65.62
5	(63, 23)	64	22	42	65.62
6	(63, 15)	64	22	42	65.62
7	(255, 239)	256	86	170	66.40
8	(255, 225)	256	86	170	66.40
9	(255, 205)	256	86	170	66.40
10	(255, 191)	256	86	170	66.40
11	(255, 183)	256	86	170	66.40
12	(255, 175)	256	86	170	66.40
13	(255, 165)	256	86	170	66.40
14	(255, 135)	256	86	170	66.40
15	(3240, 3072)	3241	1081	2160	66.64
n	Code (n, k)	n+1	(n/3) + 1	A - B	$\left(\frac{A-B}{n+1} X100\right)$

As can be seen, we use the following parameter Block Length (n) and the number of information symbols in a code word (k). Thus, from this analysis, we can see that first design based the first factored method gives the better performance than the other design in term number of gained iterations. Because, it needs ((n/3) + 1) iteration instead (n+1) for the second design, however this method gives Minimization rate between 66.40 and 65.62.

This increase in performance, as well as the increase in reliability makes it perfect for BCH and RS decoders. Here the number of iterations are taken are minimize and its summary of the performance is presented in Table 5.

Figure 16 shows the minimization rate performance of the modified Circuit for the syndrome Computation architectures. Performance analysis is carried out by varying the Block Length (n) and number of information symbols in a code word (k) and calculating the corresponding number of iteration and minimization rate percentage. The results obtained are compared to that obtained from the serial circuit.

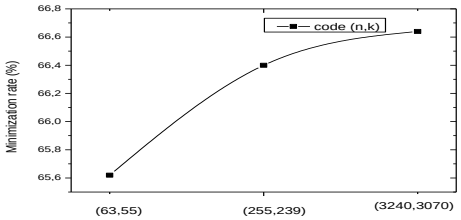

Figure 16. Evolution of the minimization rate depending on the code (n, k)

5. Conclusions

In this work, different designs were analyzed and compared .therefore, several test scenarios concerning the proposed designs have been carried out, in order to validate its effectiveness. Evaluation performances showed that it is effective when the new designs are applied.

The major advantage of the proposed design using second factorization method is its minimized number of logic gates and ease of implementation. That is can decrease the BER to optimal values.

For the second contribution, the new design based Three-Parallel Syndrome architecture shows a high performance in term number of gained iterations. It can be seen that when considering the minimized number of iterations, the Three-Parallel Syndrome architecture is the obvious choice for the real time RS and BCH decoders.

Further work could be done in optimizing designs for RS and BCH codes and developing techniques specifically designed and LPDC codes.

References

[1] Orhan Gazi, "Forward Error Correction via Channel Coding Engineering", Springer, Cham, 2020

[2] P. Salija and B. Yamuna, "Optimum energy efficient error control techniques in wireless systems: a survey", journal of Communications Technology and Electronics, vol. 60, pp. 1257–1263, 2015.

[3] Todd K. Moon, 'Error Correction Coding: Mathematical Methods and Algorithms", 1st Edition, Electronic ISBN: 9780471739210 , 2005.

[4] Robert H. Morelos, "Zaragoza, The Art of Error Correcting Coding", John Wiley & Sons, Electronic, ISBN:9780470847824, 2002.
[5] K. Deergha Rao, "Channel Coding Techniques for Wireless Communication"s, Electronic ISBN: 978-81-322-2292-7, Springer India, 2015.
[6] Trung Q. Duong and Nguyen-Son Vo, "Wireless Communications and Networks for 5G and Beyond", MOBILE NETWORKS AND APPLICATIONS, VOL. 24, PP. 443–446, 2019.
[7] Krishnamoorthy Narasu Raghavan et al., "Performance Analysis of Convolution Code with Variable Constraint Length in Shallow Underwater Acoustic Communication", International Journal of Computer Network and Information Security (IJCNIS), Vol.11, No.2, PP. 13-20, 2019.
[8] Rajashri Khanai et al., "Performance Analysis of Underwater Acoustic Communication using IDMA-OFDM-MIMO with Reed Solomon and Turbo Code", International Journal of Computer Network and Information Security (IJCNIS), Vol.10, No.12, PP.41-46, 2018.
[9] Elghayyaty Mohamed et al., "Minimized Logic Gates Number Of Components In The Chien Search Block For Reed-Solomon (RS)", American Journal of Engineering Research (AJER), Vol.7, No.2, pp. 110-116, 2018.
[10] MelikeYigit, PinarSarisarayBoluk, V. CagriGungor, "A new efficient error control algorithm for wireless sensor networks in smart grid", Computer Standards & Interfaces, Vol. 63, pp. 27-42, 2019.
[11] Vrajkishandugar et al., "A Survey On Hamming Codes For Error Detection", 2nd International Conference On Intelligent Computing, Instrumentation And Control Technologies (ICICICT), pp. 633-638, 2019.
[12] Jagannathsamanta et al., "FPGA Based Area Efficient RS(23, 17) Codec", Microsystem Technologies, Vol. 23, pp. 639-650, 2017.
[13] Vicente Torres et al., "Soft-Decision Low-Complexity Chase Decoders For The Rs(255,239) Code", Electronics, Vol. 9, No. 8, 2019.
[14] Elghayyaty Mohamed et al., "Performance Study of BCH Error Correcting Codes Using the Bit Error Rate Term BER", Int. Journal of Engineering Research and Application, Vol. 7, No. 2, pp.52-54, 2017.
[15] V. Pushpa et al., "BER Analysis Of BPSK for Block Codes and Convolution Codes Over AWGN Channel", International Journal Of Pure And Applied Mathematics, Vol. 114, No. 11, pp. 221-230, 2017.
[16] Alyaa Al-Barrak et al., "Enhancing BER Performance Limit of BCH and RS Codes Using Multipath Diversity", Computers, Vol. 6, No. 21, pp. 1-13, 2017.
[17] Bashar Tahir et al., "BER Comparison Between Convolutional, Turbo, LDPC, and Polar Codes", IEEE 24th International Conference On Telecommunications (Ict), pp. 1-7, 2017.
[18] Yuval Genga et al. , "Improving the Decoding Performance of the PTA Algorithm for RS Codes Through the Extension of the Parity Check Matrix", IEEE Africon, pp. 171-174, 2017.
[19] Richard W. Middlestead, "Digital Communications with Emphasis On Data Modems: Theory, Analysis, Design, Simulation, Testing, And Applications", Wiley Telecom, Electronic ISBN: 9781119011866, JOHN WILEY & SONS, INC., 2017.
[20] Horner, W. G. "A New Method of Solving Numerical Equations of al. Orders by Continuous Approximation." Philos. Trans. Roy. Soc. London 109, pp. 308-335, 1819.