International Journal of Engineering and Technology, 2 (4) (2013) 293-301
©Science Publishing Corporation
www.sciencepubco.com/index.php/IJET

Conception and Implementation of a BCH Code on a FPGA Board

R. El gouri[1,2*], W. Ait Ahmed[1], A. Lichioui[3], L. Hlou[1]

[1] *Laboratory of Electrical Engineering & Energy Systems Faculty of Science Ibn Tofail University, Kenitra, Morocco*
[2] *National School of Applied Sciences (ENSA), Ibn Tofail University, Kenitra, Morocco*
[3] *National Society of Radio and Television (SNRT), Rabat, Morocco*
**Corresponding author E-mail: elgouri.rachid@yahoo.fr*

Abstract

In this paper we have designed and implemented a BCH (15, 7, 5) encoder on FPGA using VHDL description language and we implanted it on an FPGA Spartan 3E Starter board. The digital logic implementation of binary encoding of multiple error correcting BCH code of length n=15 is organized into shift register circuits. Multiple characteristics of cyclic codes will be discussed further on. The results of the simulation and implementation using Xilinx ISE.12.1 software and the LCD screen on the FPGA's Board will be shown at last.

Keywords: *Galois fields, Convolutional coding, Turbo coding, DVB-T, DVB-S2, LDPC*

1 Introduction

Nowadays, we live in a world where communications play an important role both in our daily lives and in their involvement in the economic and technological fields. We constantly need to increase the flow of transmission while maintaining and improving their quality. But without a concern of reliability, all improvement efforts would be futile because it would necessarily mean that some data are to be rebroadcast [1, 2], that's why strong data transmission with less errors was needed hence new ways of transmitting data broke through in the world of communications including transmission via satellites or what's called Digital Video Broadcasting Via satellite (2nd generation) DVB-S2.
DVB-S2 represents an evolution of digital broadcasting for television. With the new additions, we have gained in spectral efficiency compared to similar existing standards [3, 4], in addition to new applications introduced.
This contribution by the DVB-S2 standard is due to the changes made to the modulation and channel coding level.[5, 6]
Channel coding is a standard technique in all wireless communication systems. In addition to the typically employed methods like convolutional coding, turbo coding or low density parity check (LDPC) coding, algebraic codes are used in many cases [4,7,8,9]. For example, outer BCH coding is applied in the DVB-S2 standard for satellite TV broadcasting [2, 6, 8] A key operation for BCH and the related Reed- Solomon codes are multiplications in finite fields (Galois Fields) [9, 8,10], where extension fields of prime fields are used. A lot of architectures for multiplications in finite fields have been published over the last decades. This paper examines four different multiplier architectures in detail that offer the potential for very high throughputs. We investigate the implementation performance of these multipliers on FPGA technology in the context of channel coding [5].
This paper will then represent the theoretical body of the BCH error correcting code, how to build it in order to apply the same theory on real BCH code used in the DVB-S2 standard, the only difference is going to be in the parameters of the code , the VHDL codes and the implementation results will be given by the end of the article.

2 Galois field

A finite field is also often known as a Galois field, after the French mathematician Pierre Galois. A Galois field in which the elements can take q different values is referred to as GF (q). The formal properties of a finite field are [9, 8, 10, 11,12]:

There are two defined operations, namely addition and multiplication.
- The result of adding or multiplying two elements from the field is always an element in the field.
- One element of the field is the element zero, such that a + 0 = a for any element a in the field.
- One element of the field is unity, such that a • 1 = a for any element a in the field.
- For every element a in the field, there is an additive inverse element -a, such that a + (- a) = 0. This allows the operation of subtraction to be defined as addition of the inverse.
- For every non-zero element b in the field there is a multiplicative inverse element. This allows the operation of division to be defined as multiplication by the inverse.
- The associative [a + (b + c) = (a + b) + c, a • (b • c) = [(a • b) • c], commutative [a + b = b + a, a • b = b • a], and distributive [a • (b + c) = a • b + a • c] laws apply.

3 BCH code

The Bose, Chaudhuri, and Hocquenghem (BCH) codes form a large class of powerful random error-correcting cyclic codes.
This class of codes is a remarkable generalization of the Hamming codes for multiple-error correction. Binary BCH codes were discovered by Hocquenghem in 1959 and independently by Bose and Chaudhuri in 1960.
For any positive integers m (m ≥ 3) and t ($t<2^m-1$), there exists a binary BCH code with the following parameters [8,13]:

- Block length: $n = 2^m - 1$
- Number of parity check digits : $n - k \leq mt$
- Minimum distance:$d_{min} \geq 2t$

Clearly, this code is capable of correcting any combination of t or fewer errors in a block of $n = 2^m - 1$ digits. We call this code a t-error-correcting BCH code.
The generator polynomial of this code is specified in terms of its roots from the Galois field GF (2^m). Let be a primitive element in GF (2^m) The generator polynomial $g(x)$ of the t-error correcting
BCH code of length $2^m - 1$ is the lowest-degree polynomial over GF (2^m) which has: $\alpha, \alpha^2, \alpha^3, \dots, \alpha^{2t}$ as its roots.
The generator polynomial of a BCH code is the smallest common multiple of the minimum polynomials corresponding to odd powers $\alpha, \alpha^2, \alpha^3, \dots, \alpha^{2t}$of a primitive element of GF (2^m) :

$$g(x) = LCM\{m_1(x), m_2(x), \dots, m_{2t-1}(x)\} \quad (1)$$

Since the degree of each minimal polynomial is m or less, the degree of $g(x)$ is at most mt. That is, the number of parity check digits, $n - k$, of the code is at most equal to mt.
There is no simple formula for enumerating $n - k$, but if t is small, $n - k$ is exactly equal to mt.

3.1 BCH encoding

We often use what's called systematic coding:
It is often used to encode the information that passes through computer networks. The following algorithm shows the steps to follow to achieve this type of coding.[8,13]

- Multiply $m(x)$ by $x^{(n-k)}$.
- Divide $x^{(n-k)}\, m(x)$ by $g(x)$.
- Have the remainder $r(x)$ of the previous division.
- Add $r(x)$ to $x^{(n-k)}\, m(x)$.

The BCH codes are implemented as cyclic codes, that is, the digital logic implementing the encoding and decoding algorithms is organized into shift-register circuits that mimic the cyclic shifts and polynomial arithmetic required in the description of cyclic codes. Using the properties of cyclic codes, the remainder $r(x)$ can be obtained in a linear stage shift register with feedback connections corresponding to the coefficients of the generator polynomial as shown in the following figure:

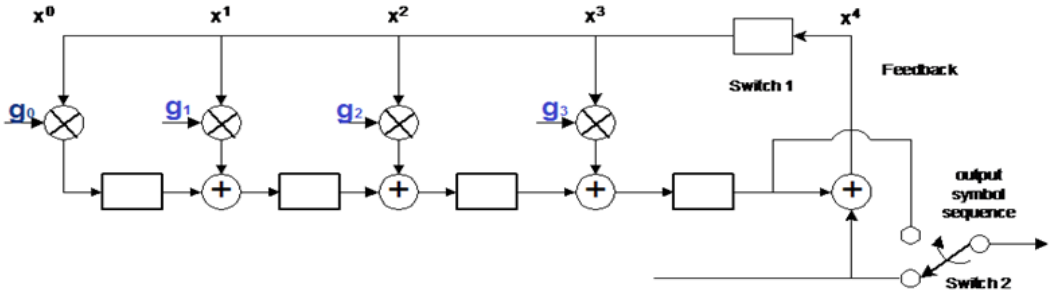

Fig.1: Digital circuit for BCH encoder

3.2 BCH decoding

The basic idea of the BCH code decoder is to detect an erroneous sequence with few words, who summoned the received data, gives rise to a valid code word [8, 13].

Several steps are required for decoding these codes:

- Calculation of syndrome
- Calculation of polynomials error localization and amplitude
- Calculation of roots and evaluation of two polynomials
- Sum of the polynomial consists of the polynomial and to reconstruct the received information
- Start without error.

This can be summed in the upcoming figure for easier conception of the VHDL source-code that we will be using in the conception of our BCH decoder.

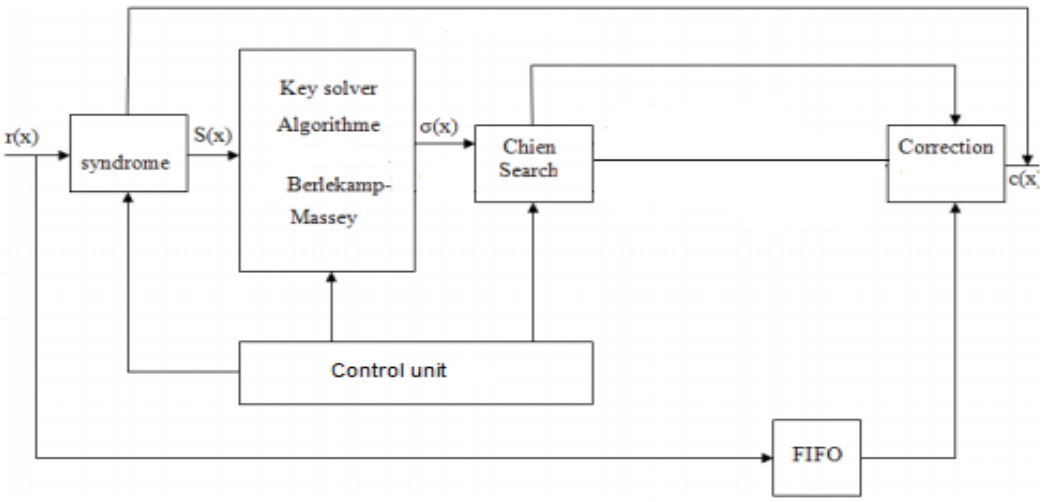

Fig.2: Digital circuit for BCH decoder

Where:

R(x): received code word
S(x): the calculated syndrome
$\sigma(x)$: The eror locating polynomial
C(x): Codeword after decoding

In this paper we used the algorithm of BERKLAMP-MASSEY [14] from the fact that it was specially made for the decoding of this type of codes. The logic chart of this algorithm is:

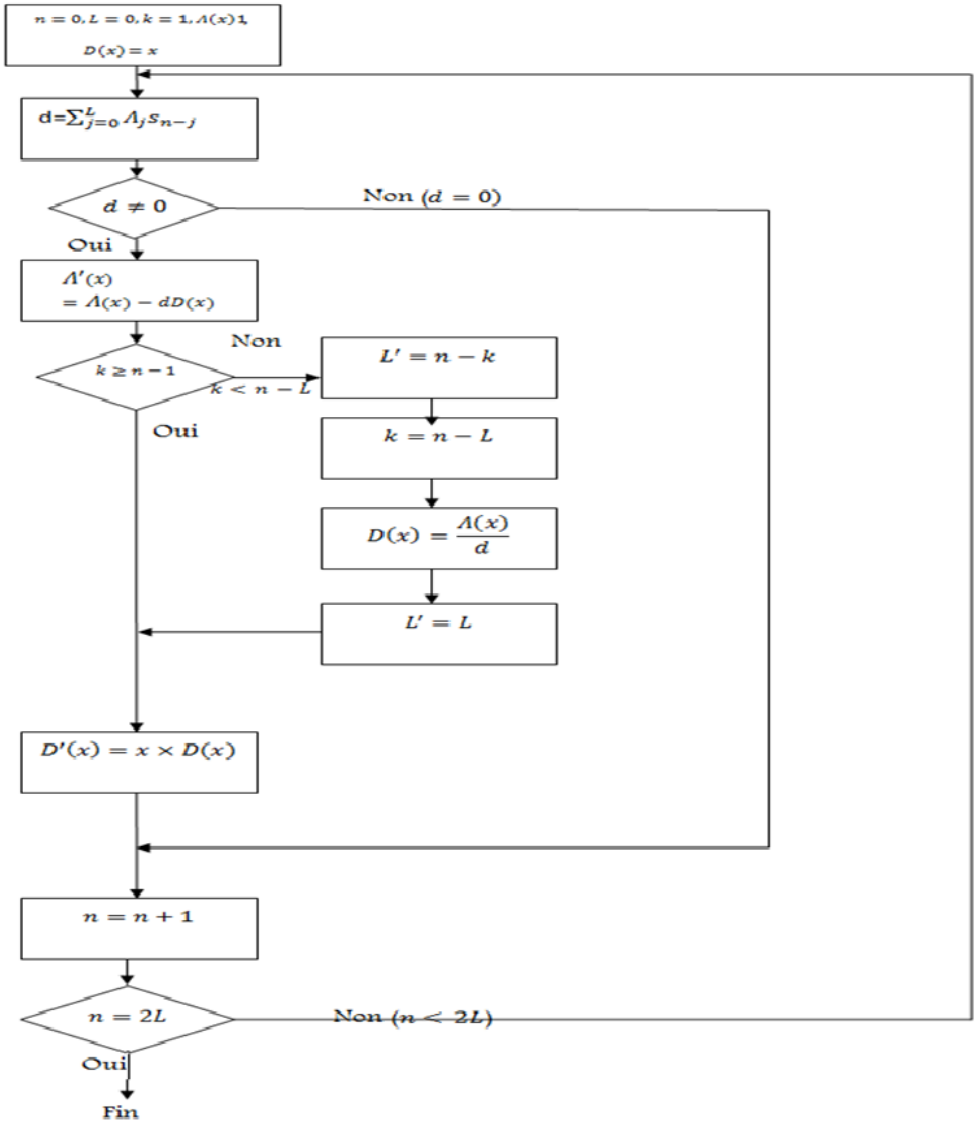

Fig.3: BERKLAMP-MASSEY algorithm [14]

Where:
$\Lambda(x)$: The error locating polynomial
d: Disperency [14]
D(x): disperency polynomial [14]

4 FPGA Target used

We used the FPGA Spartan 3E starter board in order to display the results on the LCD screen that exists on the board. It has many others blocs distend to different areas of use in the world of micro-technology.
The FPGA Spartan 3E starter is shown in the following image [15]:

Fig. 4: FPGA Spartan 3E starter board

This family of boards is a lot easier in manipulating process than others since we implement the programs on the FPGA microchip directly from the development software used to write the VHDL program in the first place unlike the previous families that required other software to realize this task.

5 Implementation of BCH code

We realized the VHDL programs of this code based on the algorithms seen above and it lead to 3 main programs:

- The coder
- The decoder
- The BCH code

As for the LCD display program, it's added at the end to command the characters to be shown after the implementation.

From the Algorithm we generate RTL (Register transfer level) code. This implies that our VHDL code describes how data is transformed as it is passed from register to register. The transforming of the data is performed by the combinational logic that exists between the registers.

Figures 5, 6 and 7 shows respectively the RTL schematics for VHDL coder , decoder and BCH code programs.

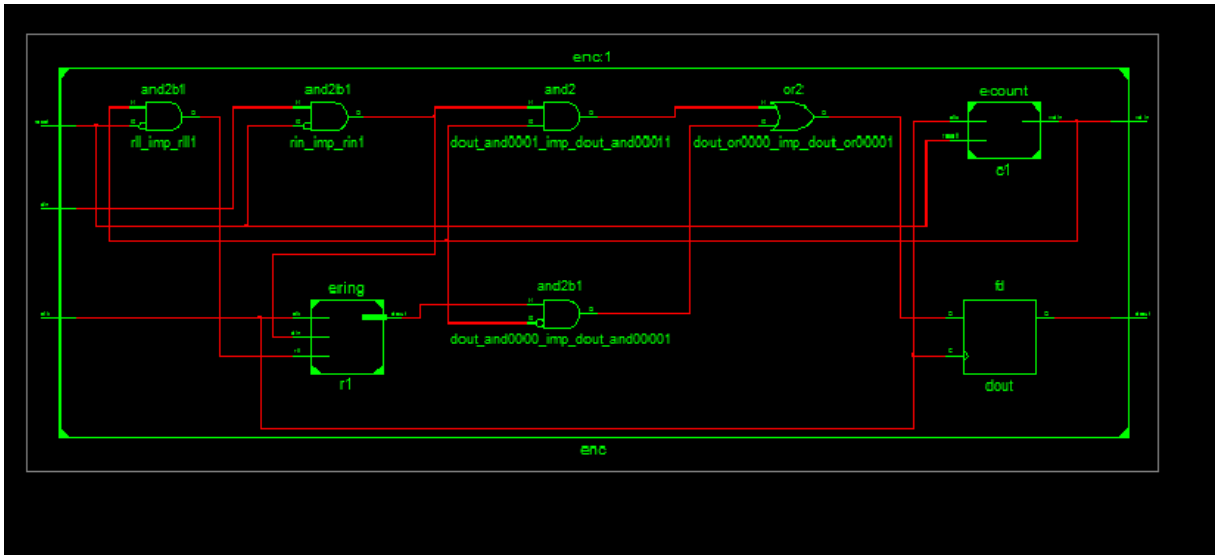

Fig. 5: RTL schematics for VHDL Coder program

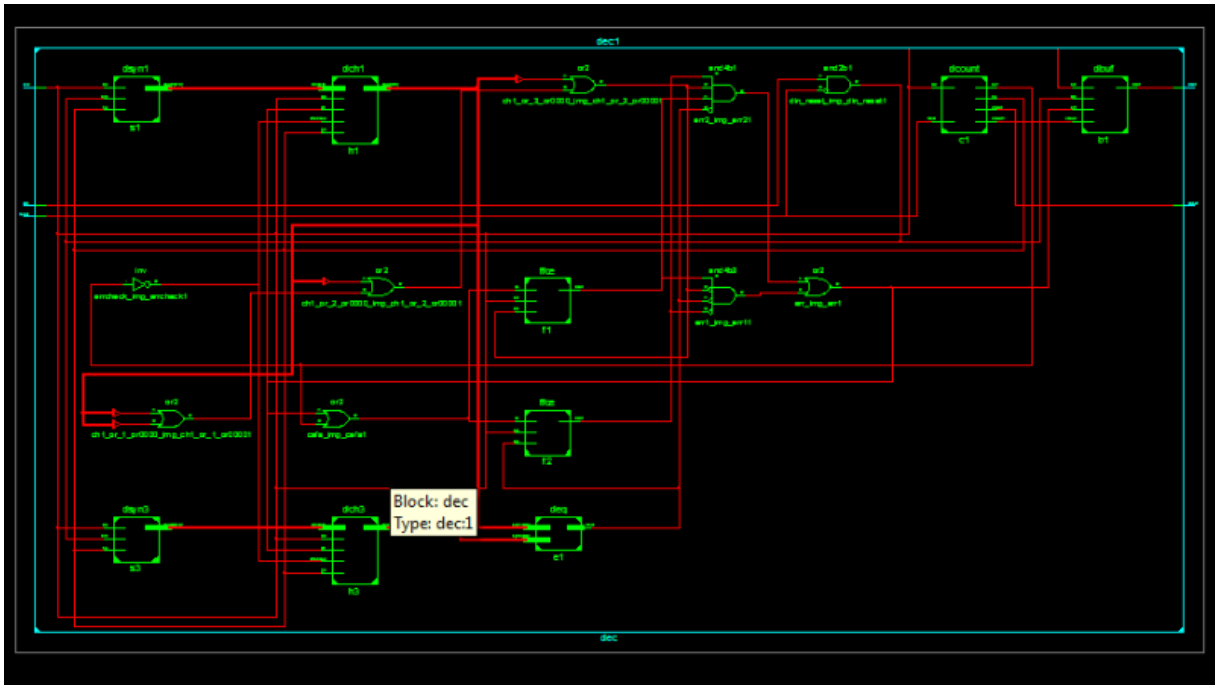

Figure 6: RTL schematics for VHDL decoder program

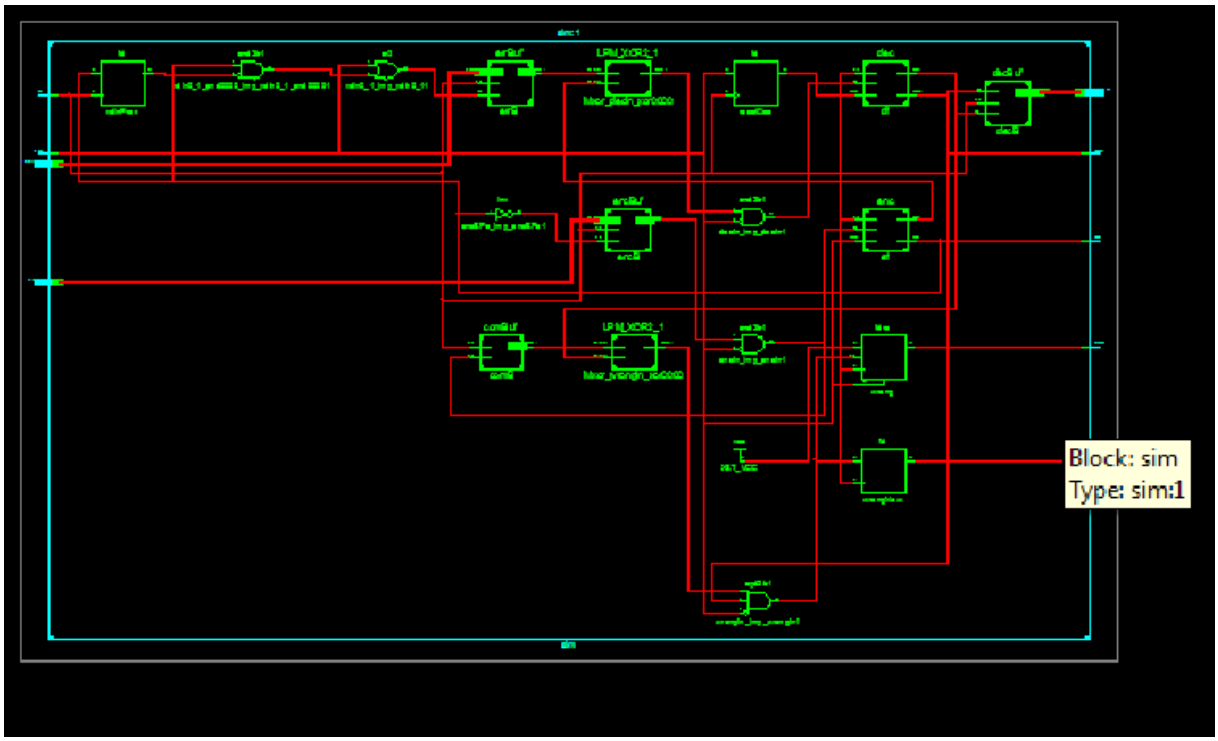

Fig.7: RTL schematics for VHDL program of the BCH code

The design of the encoder/decoder BCH was described in VHDL and simulated on ISIM and its performance has been verified using the software Xilinx 12.1.

As for the simulation results, we decided to do two simulations sequences:

two simulations sequences:

- Sending only one data frame k=1000100 and one error e=011000010000000 (Fig .8)

- Sending two data-frames k= 0100010 and k'=0011101 and two errors e=011000010000000 and e'=100001000100000 (Fig.9).

The results that we got after simulating the blocs are shown in figure 8 and 9. The simulation results shown in the figures are indeed proof of the successful realization of the coding and decoding blocs we implemented a specific frame and got the same one at the exit which was the purpose followed by the algorithm used in this case.

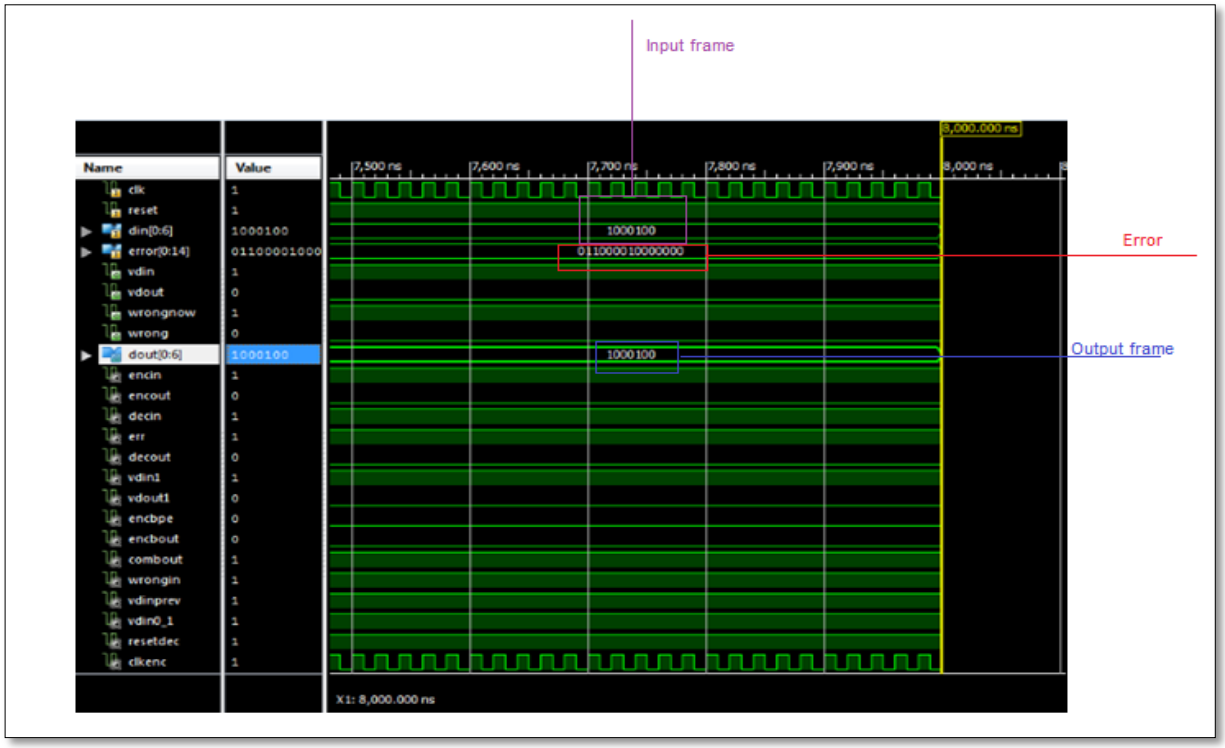

Fig. 8: Simulation for one sequence

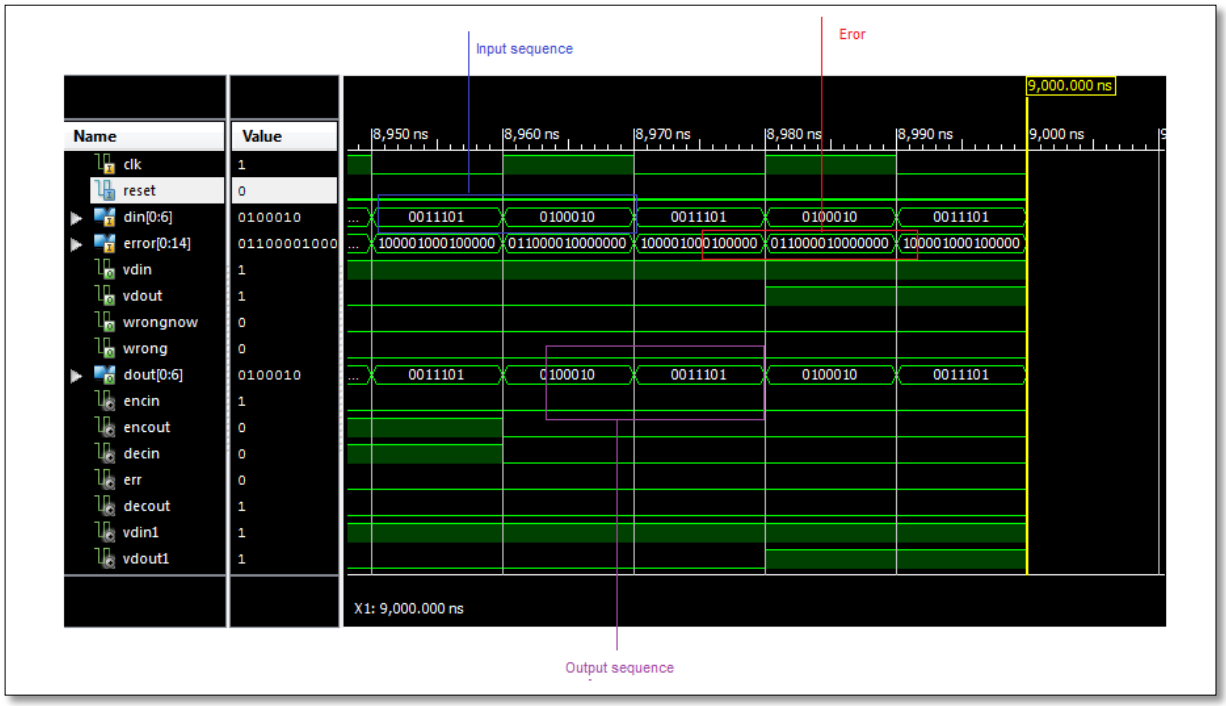

Fig. 9: Simulation for two sequences

The encoder/decoder BCH design was implemented on FPGA Spartan 3E starter, using the software Xilinx ISE12.1. As for the implementation results we can clearly see on the LCD screen the results displayed as the input frame and the output frame (Fig.7). We realized this program as to have the same frame at both sides of the blocs.

Fig.7: Implementation results on LCD screen of the FPGA Spartan 3E starter

The Table 1, shows the performance status of BCH code, like project file, module name, target device, project version, design goal and also it gives utilization summary. Our project uses 73 slice registers among 9312 and also it uses 94 slice LUTs among 9312. It uses 35 bounded IOBs among 238. It uses 15% bounded IO blocks and overall hardware overhead of BCH Code is 22%. These performances were verified by software Xilinx ISE 12.1.

Table 1: Shows the performance status

sim Project Status			
Project File:	Code_BCH.xise	**Parser Errors:**	No Errors
Module Name:	sim	**Implementation State:**	Programming File Generated
Target Device:	xc3s500e-4fg320	**•Errors:**	No Errors
Product Version:	ISE 12.1	**•Warnings:**	3 Warnings (0 new)
Design Goal:	Balanced	**•Routing Results:**	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	**•Timing Constraints:**	All Constraints Met
Environment:	System Settings	**•Final Timing Score:**	0 (Timing Report)

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	73	9,312	1%	
Number of 4 input LUTs	94	9,312	1%	
Number of occupied Slices	68	4,656	1%	
Number of Slices containing only related logic	68	68	100%	
Number of Slices containing unrelated logic	0	68	0%	
Total Number of 4 input LUTs	94	9,312	1%	
Number used as logic	90			
Number used as Shift registers	4			
Number of bonded IOBs	35	232	15%	
Number of BUFGMUXs	1	24	4%	

6 Conclusion

The emergence and deployment of DVB-S2 technology will have a significant impact on the industry of broadcasting and telecommunications satellite. The new standard meets the specifications of DVB-S standards and previous DSNG and significantly improves efficiency of coding and modulation. It addresses the need for a long-awaited higher spectral efficiency and, combined with new video compression technologies such as H.264/AVC, it allows companies to DTH to offer other services and SD TV HD and interactive TV services with the spectral resources. Although the DVB Group does not expect that the DVB-S2 standard replaces the DVB-S standard for television in the near future, due to investment in DVB-S across the globe the effort was considered so beneficial, gains power and so awesome and so universal coding applications, we hear more and more in the middle of the broadcast that "we no longer need another system in our lifetime."

However in this paper we are interested in achieving a VHDL program coding / decoding BCH code destined to the DVB-S2 standard.

To begin, it was necessary to make a point on the digital broadcast standard DVB-S including how the BCH codes are used and the transmit / receive chain to understand the peculiarity of this transmission technique and especially for standard DVB-S2.

It was also a thorough theoretical study for each component used in the encoding and decoding BCH followed by examples for complete understanding.

The design of the encoder/decoder was described in VHDL and validated on FPGA (Spartan 3E starter) using the software Xilinx 12.1.

The results showed that after compilation and simulation, it seemed that the integration of reprogrammable integrated circuits (FPGA) in telecommunications systems facilitates the improvement of the transmission rate and thus increase the transmission speed.

Acknowledgements

This work was supported by the Laboratory of Electrical Engineering and Energy System, Faculty of Science,University Ibn Tofail Kenitra. And the National Society of Radio and Television (SNRT), Rabat.

References

[1] L.Rysdale, P.Bot, S. N. Hulyalkar, ''Digital video broadcasting: Satellite specification'', Philips Journal of Research, Volume 50, Issues 1–2,(1996), Pages 91-104

[2] V.Mignone, M.A.Vazquez-Castro, T.Stockhammer," The Future of Satellite TV: The Wide Range of Applications of the DVB-S2 Standard and Perspectives » Proceedings of the IEEE, Vol.99 , Issue: 11, (2011) , pp: 1905 – 1921

[3] H.Hussien, K.A.Shehata,; M.Khedr, S.Hareth, "Performance study on implementation of DVB-S2 low density parity check codes on additive white Gaussian noise channel and Rayleigh fading channel » Electronics Design, Systems and Applications (ICEDSA) 2012 IEEE International Conference on, (2012) , pp: 179 – 182

[4] J. M.Porras, J.I. Curto, ''Classification convolutional of codes'', Linear Algebra and its Applications, Volume 432, Issue 10, 1 (May 2010), Pages 2701-2725

[5] A.Morello,V.Mignone,"DVB-S2: The Second Generation Standard for Satellite Broad-Band Services ", Proceedings of the IEEE, Vol.94 , Issue: 1, (2006) , pp: 210 - 227 .

[6] ETSI TR 102 376 v1.1.1 (2005-02). Digital Video Broadcasting (DVB); User guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2).

[7] Gabriel Falcao, Vitor Silva, Leonel Sousa, ''Chapter 38 - Parallel LDPC Decoding'', GPU Computing Gems Emerald Edition, (2011), pp: 619-628

[8] R.L. Miller,''Generalized BCH codes'', Information and Control, Vol.40, Issue 1, (January 1979), pp 61-75

[9] G.D. Forney Jr.,''Algebraic Structure of Convolutional Codes and Algebraic System Theory'', Springer-Verlag, Berlin-Heidelberg (1991) pp. 527–557

[10] G.D. Forney Jr., ''Convolutional codes I: algebraic structure'',IEEE Trans. Inform. Theory, 16 (1970), pp:. 720–738.

[11] R. Johannesson, K.S. Zigangirov, ''Fundamentals of convolutional coding'',IEEE Press Series in Digital and Mobile Comm. (1999)

[12] E.H.Anas, ; R.El gouri, L. Hlou « A low power error detection in the Chien Search Block for Reed-Solomon code », IEEE conference publications Complex Systems (ICCS), 2012 International Conference, (2012) , pp: 1 - 3

[13] G.D. Forney Jr,''On decoding BCH codes'', IEEE Trans. Info. Theory, 11 (1965), pp. 549–557

[14] A. Le Glaunec, « Décodage des codes BCH », Cours Supélec, (2001).

[15] www.xilinx.com, « Xilinx Spartan-3E FPGA Starter Kit Board User Guide », (2006).