# BY USING VHDL IMPLEMENTATION OF REED SOLOMON ENCODER & DECODER FOR WIRELESS NETWORK 802.16

[1]**Priyanka J. Jambhulkar**
[1]YTIET College of Engineering, Mumbai (MS), India

[2]**P. A. Salunkhe**
[2]YTIET College of Engineering, Mumbai (MS), India

***Abstract- The reed Solomon codes can detect & correct error within blocks of data. Channel coding for error detection & correction help the communication system designers to reduce the effect of a noisy transmission channel. The aim of this paper implementation of reed Solomon error detecting & correcting code by using VHDL language Berlekamp-Massey Algorithm are the main components. This operation based on Field Programmable gate array (FPGA)***

***Keywords — Reed-Solomon codes; code generator polynomials; syndrome; Berlekamp Massey; Chien; IEEE 802.16; VHDL;FPGA***

## I. INTRODUCTION

Reed-Solomon codes are block-based error correcting codes with a wide range of applications in digital communications and storage such as

- Storage device (hard disks, compact disks, DVD).
- Wireless communication (mobile phone, microwave links).
- Digital television.
- Satellite communication.
- Broadband modems (ADSL).

Reed Solomon codes work by adding extra "redundant bits" to the original data. The encoded data can then be transmitted or Stored. During transmission error may happen for a number of reasons e.g.(scratches on CD, radio frequency interference with mobile phone reception, noise etc.) At the receiving side, the decoder detects and corrects a limited predetermined number of errors occurred during transmission.

## II. REED SOLOMON CODE

RS codes are linear block codes and a subset of BCH codes. An RS code is based on finite fields, often called Galois fields. The number and types of errors that can be corrected depends on the characteristics of the Reed-Solomon code. RS (n, k) codes parameters are described as follows.

1. m - Number of bits per symbol
2. k- Uncoded message length in symbols
3. n- Codeword length in symbols
4. (n-k)- Number of parity check symbols
5. t -Number of correctable symbol errors and 2t = =n-k

## III. REED SOLOMON ENCODING AND DECODING

The most notable error correcting codes are Reed Solomon codes. In real world communication, errors are introduced in messages sent from one point to another. Reed Solomon is an error-correcting coding system that was devised to address the issue of correcting multiple errors - especially burst-type errors. In order for the transmitted data to be corrected in the event that it aquires errors, it has to be encoded .The receiver uses the appended encoded bits to determine and correct the errors upon reception of the transmitted signal. The number and type of errors that are correctable depend on the specific Reed Solomon coding scheme used.

### 2.1 Reed-Solomon Encoder

Reed-Solomon code is represented by RS (n,k) with m-bit symbols. Two types of encoders are available systematic and non-systematic. Systematic encoder is used for Reed-Solomon encoder in which data remains unaltered. This requires that data symbols must be shifted from power level of (n-1) down to (n-k) and remaining positions from power (n-k-1) to 0 be filled with zeros. RS encoder performs two main operations, shifting and division. Both operations are implemented by using linear-feedback shift registers [5]. Let a message or data is represented in the polynomial form as:

$$M(x) = m_{k-1}\, X^{k-1} + m_{k-2}\, X^{k-2} + \ldots\ldots\ldots\ldots + m_1 X + m_0 \qquad (5)$$

4F

And the codeword is represented in polynomial as:

$$C(X) = C_{n-1}\, X^{n-1} + C_{n-2}\, X^{n-2} + \ldots\ldots\ldots\ldots + C_1\, X + C_0 \qquad (6)$$

And the generator polynomial is shown below:

$$g(X) = g_0 + g_1\, X + g_2 X^2 + \ldots\ldots + g_{2t-1}\, X^{2t-1} + X^{2t} \qquad (7)$$

And parity-check is given by:

$$Ck(X) = X^{n-k}\, M(X) \bmod g(X) \qquad (8)$$

And codeword is given by:

$$C(X) = X^{n-k}\, M(X) + CK(X) \qquad (9)$$

Here, for encoding Reed-Solomon code (255,239) for IEEE 802.16 network the code generator polynomial is used as given below [6]:

$$g(X) = X^{16} + 76X^{15} + 34X^{16} + 67X^{13} + 1FX^{12} + 68X^{11} + 7EX^{10} + BBX^{9} + E8X^{8} + 11X^{7} + 38X^{6} + B7X^{5} + 31\,X^{4} + 64X^{3} + 51X^{2} + 2CX + 4F \qquad (10)$$

There are 2t= n-k parity symbols that are 16 in this case. The error-correcting capability for this code is t= $\frac{(n-k)}{2}$ that is 8 symbols. In digital hardware, it is represented by using linear feedback shift re(LFSR) [7]. It is represented in the figure 2

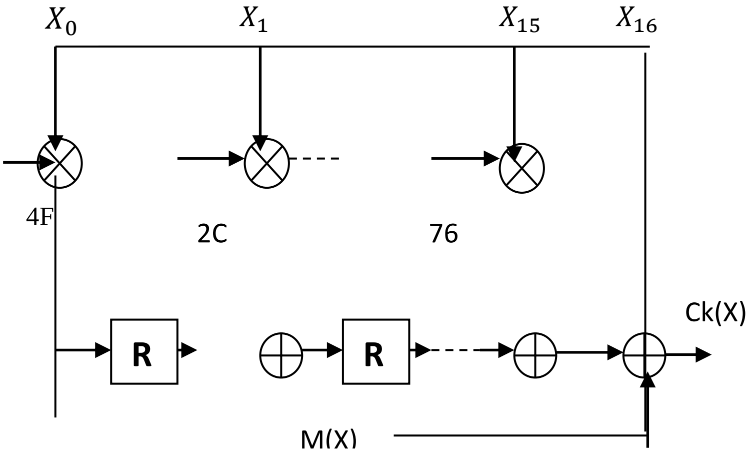


Fig 2: LFSR encoder implementation

### 2.2 Reed-Solomon Decoder

Decoding processes are cumbersome, difficult to understand and to implement in the hardware than encoding process. Reed-Solomon decoder works in two parts error detection and correction of that detected errors. The capability of correct error is depend on the code uses. It is denoted by t and calculated by the formula t= (n-k)/2 in which n is the length of codeword and k is the length of source data and n-k is the parity symbols. The main blocks of reed-Solomon decoder are the syndrome calculation, Berlekamp-Massey algorithm, and Chien search and error correction [8]. Syndrome means symptoms and it detect whether the error is present or not. If the result of syndrome calculator is zero then it means there is no error and if the result is non-zero then error will be there. The other blocks are used to correct the error if syndrome detects the error. All the algorithms are discussed below:

#### 2.2.1 Syndrome calculation

At the receiver, syndrome is calculated to check the presence of errors. If the error correction capability of code is 8 then syndromes will be 16 that is 2t. The syndrome polynomial contains the location and magnitude of up to t errors in an invalid codeword. A valid codeword generates a syndrome polynomial with all zero coefficients.

$$S(X) = \sum_{i=1}^{2t} Si\, X^{(i-1)}\ , \quad Si = R\,(\alpha^{i}) \quad (11)$$

Where,

$$R\,(X) = R_0 + R_1\,X + R_2\,X^2 + .......+R_{n-1}\,X^{n-1}$$

is the received polynomial and $R_{n-1}$ is the first received symbol.

#### 2.2.2 Berkelamp-Massey algorithm

R(X)

The main component of Reed-Solomon decoder is the key equation solver. This equation involves 2t linearly dependent equations. It generates the key equations [9]:

$$\mu(X)\,S(X) = \Omega\,(X) \bmod (X^{2t}) \qquad (12)$$

Where μ(X) is the error locator polynomial and $\Omega$(X)is the error evaluator polynomial. The error locator polynomial contains information about location of bad symbols and error evaluator polynomial contains information about the error magnitude of the bad symbols. There are different techniques for solving key equation are berkelamp algorithm [10] and Euclidean algorithm [11]. The technique used in this paper is the berkelamp-massey algorithm because this algorithm having less hardware complexity.

#### 2.2.3 Chien search and error-correction

Chien search is used to find the roots of polynomial in reed-solomon decoder. Chien search performs the function of finding the error locations, and correcting the data, using Forney's algorithm, when the Chien sum is zero. The evaluation of the polynomials needed for error correction is pipelined. Chien search uses all possible input values and then checks to see if the outputs are zero. In the reed-solomon decoder, the delay block is the necessary to adjust for the delays in the syndrome block and startup delay in the Chien search block.

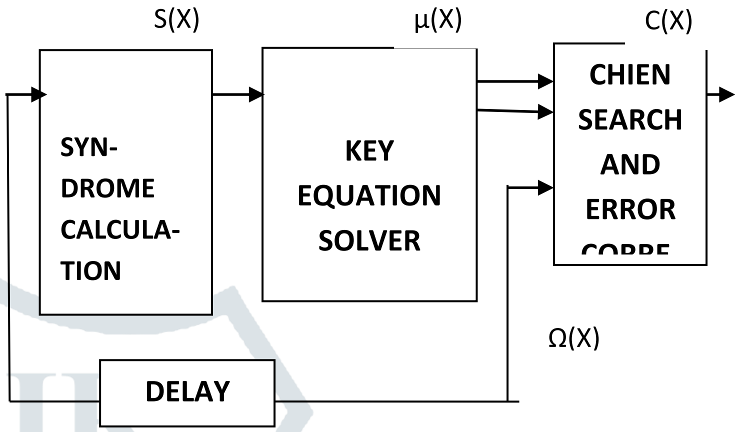


Fig 3: Reed-Solomon Decoder

### PROBLEM STATEMENT

It has been noticed that the present Reed-Solomon Decoder are very large in area and their speed is not too fast. Since Galois field inverters are very complex, several times more so than a Galois field multiplier, a key solver that eliminates the need for such an inverter can yield savings in decoder latency. Such an algorithm is the inversion less massey-berlekamp algorithm by Reed, Shih and Truong [6]. However, the drawback to this algorithm is that it only computes the error – locator polynomial.In the inversion less massey-berlekamp algorithm, the error- evaluator polynomial is computed then by multiplying the error – locator polynomial by the syndrome polynomial. This results in extra latency of the order of the error-locator polynomial, and as a result an extra number of gates and registers are needed.

### V. PROPOSED WORK

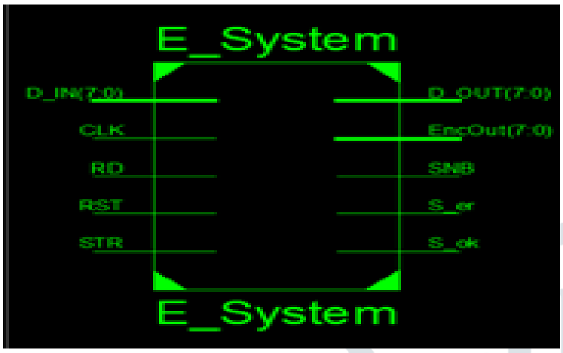
In this paper, the extended inversion less Massey-Berlekamp algorithm is used. The extended inversion less Massey-Berlekamp algorithm overcomes the extra latency by computing both the error - locator polynomial and the error – evaluator polynomial at the same time. In this thesis, (255,239) reed Solomon codes have been designed. A pipelined RS decoders is proposed with the aim of reducing the hardware complexity and improving the clock frequency in the RS decoders. This design also use the pipelined GF multiplier in the syndrome computation block, KES block, Forney block, Chien search block and error correction block for the enhanced clock frequency and provides low complexity as compared to the conventional ME algorithm architectures.

**Table 1 – Used available utilization design summary for RS(255,239) for 802.16 wireless network**

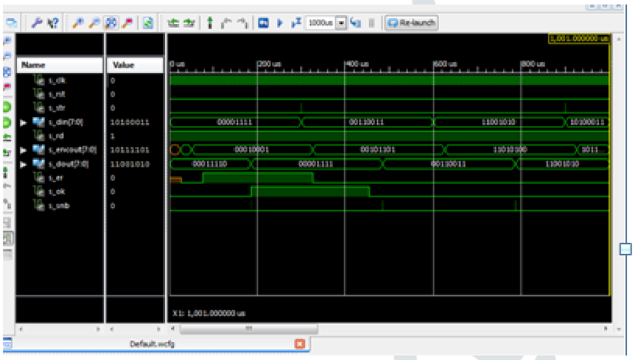
| LOGIC UTILIZATION | USED | AVAILABLE | UTILIZATION |
|---|---|---|---|
| Number of Slice Registers | 4866 | 30064 | 16% |
| Number of Slice LUTs | 4382 | 15032 | 29% |

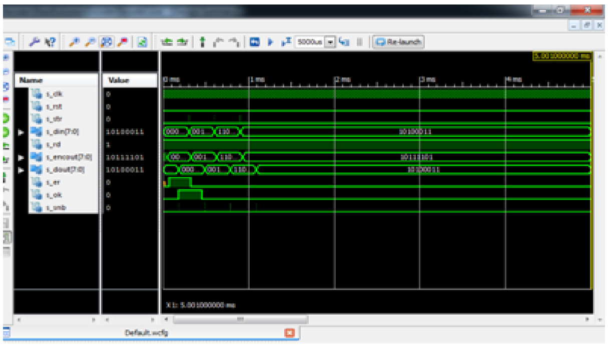
| | | | |
|---|---|---|---|
| Number of Fully used LUT-FF pairs | 2716 | 6532 | 41% |
| Number of bonded IOBs | 12 | 190 | 6% |
| Number of Block RAM/FIFO | 2 | 52 | 3% |
| Number of BUFG/BUFGCTRLs | 2 | 16 | 12% |

**Decoder Implementation –**



**Simulation Result of RS Encoder:**





**REFRENCE: -**

**[1]** J.-I. Park, H. Lee ―Area-efficient truncated Berlekamp-Massey architecture for Reed- Solomon decoder‖ 17th February 2011.[2] Mostafa El-Kham et.al. ―Iterative Algebraic Soft-Decision List Decoding of Reed–Solomon Codes‖ IEEE Vol. 24, No. 3, March 2006.

**[2]** ShadiaHijazie, ― High-level synthesis and its application in the design of Reed-Solomon decoders‖, Master's thesis, Concordia University, July 1997.

**[3]** Lee H., ―A high speed, low complexity Reed-Solomon decoder for optical communications, ‖ IEEE Transactions on Circuits and Systems II, PP. 46 [6] Kenny Chung ChungWai, Dr. Shanchieh Jay Yang‖ Field Programmable Gate Array Implementation of Reed-Solomon Code, RS(255,239)‖In proceeding of 9th Annual Military and Aerospace programmable logic Device International Conference,2 september 2011.

**[4]** Lin Shu, ―An Introduction To Error-Correcting Codes‖, Prentice Hall, Inc., 1970.

**[5]** Joaquin Garcia, Rene Cumplido Department of Computer Science , ―On the design of an FPGA-Based OFDM modulator for IEEE 802.16-2004‖ INAOE, Puebla, Mexico 2005 International Conference on Reconfigurable Computing and FPGAs (ReConFig 2005).

**[6]** K. Harikrishna, T. Rama Rao, and Vladimir A. Labay, ―FPGA Implementation of FFT Algorithm for OFDM Based IEEE 802.16d(Fixed WiMAX) Communications‖ Journal Of Electronic Science And Technology, Vol. 8, No. 3, September 2010.