

# VLSI Implementation of Low -Complexity Reed Solomon Decoder

G.Lakshmi Priya, A.Raghuram

**Abstract**— In this paper, a low complexity architecture of Reed-Solomon (RS) code is developed to correct errors based on truncated inversion less Berlekamp Massey algorithm. The arithmetic operations which are used in RS code are Galois Fields (GF) addition and multiplication. This paper presents: i) RS encoder modeled using MATLAB with data encoded in the noisy channel for functional verification. ii) RS decoder modeled in Veri log HDL to recover the erroneous data. The Veri log modeled RS (255, 239) decoder has the capability of 8 symbol errors detection and correction. The proposed decoder has been designed and synthesized for the Xilinx Spartan6 series FPGAs xc6Ix16-3. The resource consumption is about 44%, and the data processing rates over 1.3Gbit/s is realized.

**Index Terms**— Berlekamp-Massey algorithm; syndrome; ReedSolomon codes; key equation solver;

## I. INTRODUCTION

Towadays, error-correcting codes are used in various digital systems in order to improve reliability. Out of many error correction codes, the Reed-Solomon (RS) codes have great power and utility, and are found in many applications from compact disc players to deep-space application. RS codes are very effective in correcting burst errors aswell as random errors. The Reed-Solomon code is defined in the Galois field, which contains a finite set of element where any arithmetic operations on elements of that set will result in an element belonging to the same set.

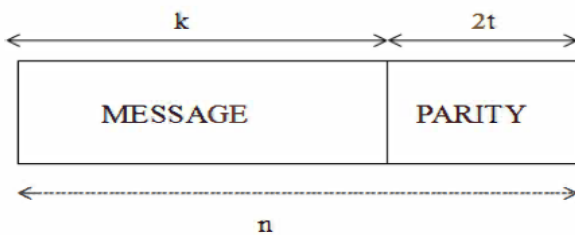


Fig. 1. The Structure of a RS Codeword

Every element of field, except zero, can be expressed as a power of a primitive element,  $\alpha$ , of the field. The non-zero field elements form a cyclic group defined based on a binary primitive polynomial. Let an  $(n, k, t)$  RS code defines over  $GF(2^m)$ , where  $n$  is block length of  $m$  bits wide symbol,  $k$  is message length of the  $m$  bits wide symbol and  $t = [n-k]/2$  is the maximum number of error correcting capability. RS decoder performs detection and correction of information (message) symbols in a codeword. The RS encoded data is processed to determine whether any errors have occurred during

G.Lakshmi Priya, Dept.Of Electronics And Communication Engineering, Indira Institute Of Technology&Science:Markapur.

A.Raghuram, Dept.Of Electronics And Communication Engineering, Indira Institute Of Technology&Science:Markapur

transmission by channel noise. After determining errors, the decoder corrects the errors in the received data.

Many researchers have tried to implement increasingly efficient RS decoder. They aim at offering of high speed and low complexity. One such RS decoder is high speed parallel RS decoder using ME architecture. ME architecture is regular systolic architecture but required cost is high due to its degree computation circuit . Furthermore, to reduce hardware complexity of RS decoder the just-in-time folding Euclidean algorithm (JIT-FMEA) architecture and the recursive degree computation less modified Euclidean (rDCME) architecture were proposed Recently to provide high speed and low complexity of RS decoder using PRiBM architecture proposed.

In Section II, Reed Solomon (RS) encoder theory introduced. Section III presents the proposed architecture for the RS decoder using the TiBM algorithm [6]. In Section, IV and V simulation result and conclusions are present.

## II. REED SOLOMON ENCODER

RS encoder works by adding parity check symbols to the input data before the data transmitted. The encoded data that consist of errors are decoded to recover the error-free data. The parity check symbols are added to allow the RS decoder to detect the locations of the corrupted data and to correct the errors arise in the data during transmission. The number of errors can be corrected by the RS code is depended on the number of parity check symbols added. The transmission codeword is systematically encoded and defined in (1) as a function of the transmitted message polynomial  $m(x)$ , the generator polynomial  $g(x)$  and the number of parity symbols  $2t$ .

$$c(x) = m(x)X^{2t} + m(x) \text{ mod } g(x) \text{-----}(1)$$

Where  $g(x)$  is the generator polynomial given by

$$g(x) = \prod_{i=1}^{2t} (x + \alpha^i) \text{-----} (2)$$

So generator polynomial of RS (255, 239) code is given as.

$$g(x) = X^{16} + 118x^{15} + 52x^{14} + 103x^{13} + 31x^{12} + 104x^{11} + 126x^{10} + 187x^9 + 232x^8 + 17x^7 + 56x^6 + 183x^5 + 49x^4 + 100x^3 + 81x^2 + 44x + 79 \text{-----} (3)$$

The variable  $\alpha$  is a root of the primitive polynomial of degree  $t$ .

In  $GF(2^8)$  the primitive polynomial is defined as  $x^8 + X^4 + x^3 + x^2 + 1$ .

III. REED SOLOMON DECODER

Let  $C(x)$  and  $R(x)$  are the transmitted codeword polynomial and the received codeword polynomial, respectively. The transmitted codeword polynomial can be corrupted by channel noise during the transmission. Therefore, the received codeword polynomial can be described as  $R(x) = C(x) + E(x) = R_{n-1}x^{n-1} + \dots + R_1x + R_0$  where  $E(x)$  is the error polynomial.

A. Syndrome Calculation Block :

The first step in the decoding received symbol is to calculate set of syndromes  $S_j$ ,  $0 \leq j \leq 2t-1$  to correct correctable errors. Any nonzero value of  $S_j$ ,  $0 \leq j \leq 2t-1$  indicates the presence of errors. The syndrome polynomial  $S(x)$  is defined as equation

$$S(x) = S_{15}x^{15} + S_{14}x^{14} + \dots + S_1x^1 + S_0 \dots (4)$$

$$S_j = R(\alpha^j) = R_{254}\alpha^{254j} + R_{253}\alpha^{253j} + \dots + R_1\alpha^j + R_0 \quad (i = 0, 1, 2, \dots, 15) \dots (5)$$

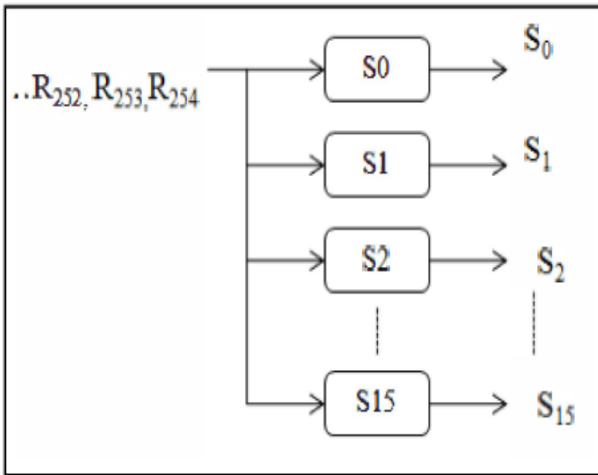


Fig. 2 .Syndrome calculation block

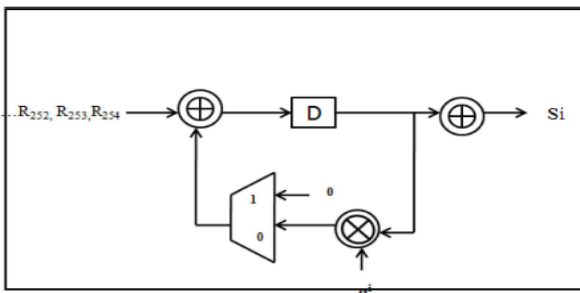


Fig. 3:Syndrome cell

The syndrome calculation block is consisted of 16 parallelsyndrome cells as shown in Fig.2 .Syndrome cell computes  $S_i$  value during 255 clock cycles.

B. KES Block :

The key equation  $s(x) \cdot (x) = a(x) \text{ mod } x^{2t}$  is generally solved by BerJekamp-Massey (BM) algorithm or the modified Euclidean (ME) algorithm. The conventional ME algorithm

are regular, but the hardware cost is high due to the required degree computation and comparison circuit. In the RS decoder, the KES block is the largest block compared with SC and CSEE blocks. The number of PEs in RiBM architecture is  $3t+ 1$  [7] and TiBM architecture is  $2t+ 2$  [6]. Therefore, to reduce the area of the KES block, TiBM architecture used by truncating  $t-1$  PEs. So, reducing the area of KES block means area of RS decoder can be reduce significantly. Also, as the more error correcting capability ( $t$ ) increases, the hardware complexity can be reduced more.

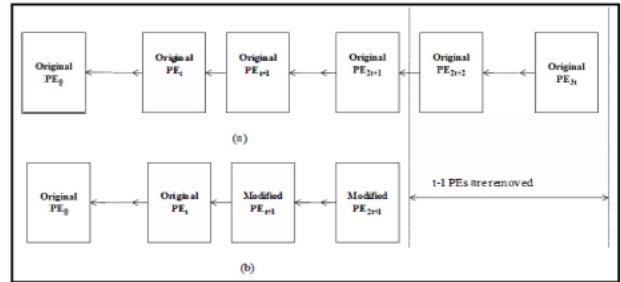


Fig. 4. Number of PEs for (a) RiBM architecture , and (b) TiBM architecture .

C. Chien Search Block, Forney Block and Error Correction Block:

After getting the error locator polynomial  $A(X)$  and error evaluator polynomial  $c(x)$  from KES block are then fed into the Chien search block and Forney algorithm block, respectively. Chien search block calculates the roots of the error locator polynomial. The Forney algorithms block which works in parallel with the Chien search block to calculate the magnitude of the error symbol at each error location. Let error locator polynomial over  $GF(2^m)$  is  $\lambda(X) = x^t + \lambda_{t-1}x^{t-1} + \dots + \lambda_0$ . Then chien search algorithm is used to find roots of error locator polynomial of degree  $t$ , which present inverse of error location. In the  $[m]$ th stage, Forney algorithm is used to calculate the value of error.

The error values corresponding to error locations are calculated according to equation (6). For division in eq. (6), inverse of an element of a divisor is stored in  $256 \times 8$  ROM, and it is then multiplied with an element of dividend.

$$Y_i = x_j \frac{\sigma(x_j^{-1})}{\lambda'(x_j^{-1})} \dots (6)$$

After getting the error locations and error values, finally can form the error polynomial  $E(X)$  and correct the received polynomial  $R(X)$  just by adding (with XOR operation) these two polynomials together

IV. SIMULATION RESULTS

Reed Solomon encoder and decoder are successfully modelled in MATLAB and Verilog HDL respectively. Errors are added into the codeword through RS encoder and the error added codeword is used as input in RS decoder. We have implemented the proposed RS (255,239) decoder using.

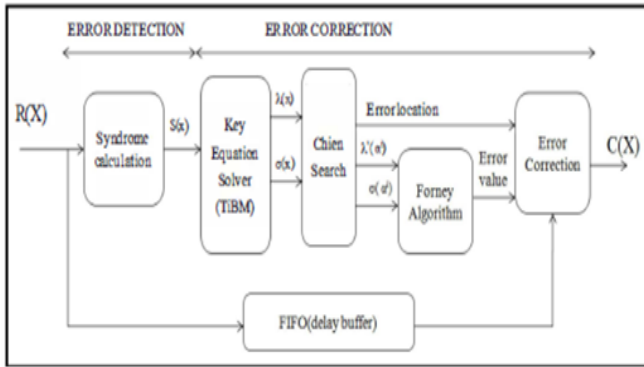


Fig.5.The Structure of Reed-Solomon Decoder

Verilog HDL and performed logic synthesis using ISE3.1 design tool with Xilinx Spartan6 series FPGAs xc6IxI6-3. Maximum frequency for this case is also found to be 162.72 MHz.

Fig.6 and Fig.7 shows simulation result and RTL view of RS decoder respectively.



Fig. 6. The Simulation result of RS decoder

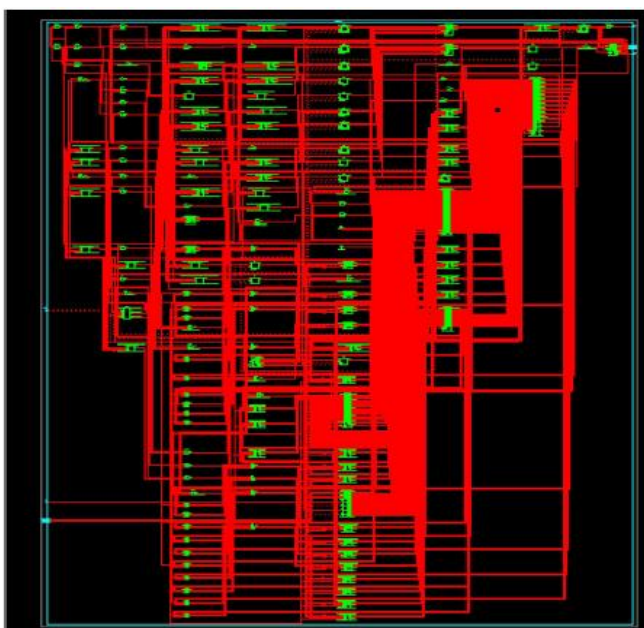


Fig. 7. The RTL view of RS decoder

TABLE I

Device Summary for RS (255,239) decoder

Logic Utilization	Used	Utilization
Number of Slice Registers	2786 out of 18824	15
Number of slice LUT	4018 out of 9112	44%
Number of fully used LUT-FF pairs	2131 out of 4440	47%
Number of bonded IOBs	21 out of 232	9%

## V. CONCLUSION

Reed Solomon encoder and decoder are successfully modelled in MATLAB and Verilog HDL respectively. Error detection and correction technique is used here for reliable communication over a noisy channel. The main element in the FPGA that is highest in the demand is the LUT's. In this paper less number of LUT's is used. This represents reduction in the cost and save a lot of area. This design will play a remarkable role with its significant speed and efficiency.

## REFERENCES

- [1] Sarwate, D.V., and Shanbhag, N.R.: 'High-speed architectures for Reed-Solomon decoders', IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 2001, 9, (5), pp. 641-655
- [2] H. Lee, "High-Speed VLSI Architecture for Parallel Reed-Solomon Decoder," IEEE Trans. on VLSI Systems, vol. II, no. 2, pp. 288-294, April. 2003.
- [3] Hsu, H.Y., Wu, A.Y., and Yeo, I.I.: 'Area-efficient VLSI design of Reed-Solomon decoder for IOGBase-LX4 optical communication systems' ,IEEE Trans. Circuits Syst. II, Express Briefs, 2006, 53, (11), pp. 1245-1249
- [4] Baek, J.H., and Sunwoo, M.H.: 'Enhanced degree computation less modified Euclid's algorithm for Reed-Solomon decoders', Electron.Lett., 2007, 43, (3)
- [5] Yuan, B., Wang, Z., Li, L., Gao, M., Sha, J., and Zhang, c.: 'Area efficient Reed-Solomon decoder design for optical communications' ,IEEE Trans. Circuits Syst. II, Express Briefs, 2009, 56, (6), pp. 469-473
- [6] <https://books.google.co.in/books?isbn=1402083912>
- [7] [www.indjst.org/index.php/indjst/article/view/29431](http://www.indjst.org/index.php/indjst/article/view/29431)