Review of FPGA Implementation of Reed-Solomon Encoder-Decoder

Vaibhav J. Babrekar
2nd Year M.E. (ENTC)
G.H.R.C.E.A.M. Amravati

Swati V. Sakhare
Assistant Professor (ENTC)
G.H.R.C.E.A.M. Amravati

ABSTRACT

An important function of any modern digital communication system is error control coding (ECC). Such coding is the field of communications that deals with techniques for detecting and correcting errors in a signal. ECC is especially useful in wireless communication systems. RS codes are the most powerful in the family of linear block codes and are arguably the most widely used type of error control codes. RS code is a type of Forward Error Correction (FEC) code and it is a non-binary, linear and cyclic block error correcting code. In this paper, the proposed work is to implement the encoder and decoder of Reed-Solomon (RS) coding scheme on the platform of VLSI using the Euclid's algorithm. Implementation will be done on VLSI Hardware Description Language (VHDL) and the operation and results can be seen on Field Programmable Gate Array (FPGA).

Keywords

Reed-Solomon (RS), Galois field (GF), Generator Polynomial g(x), Forward error Correction (FEC), Code Rate, Block Size

1. INTRODUCION

Digital communication is used to transport an information bearing signal from the source to a destination via a communication channel. A code is the set of all the encoded words, the code word that an encoder can produce. When the encoded data gets combined, it becomes a code. Reed-Solomon error detecting and correcting codes (RS codes) are widely used in communication systems and data storages to recover data from possible errors that occur during transmission and from disc error respectively. Reed Solomon (RS) encoders and decoders are extremely powerful error detecting and correcting tools that increase transmission quality to a great extent. RS codes operate on the information by dividing the original message stream into blocks of data and ads redundancy per block depending only on the current inputs. One typical application of the RS codes is the Forward Error Correction (FEC), the scheme is presented in Figure [a].

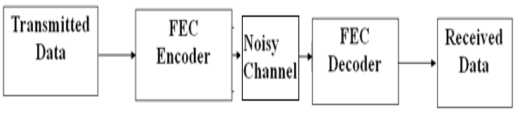

Fig [a] Forward Error Correction scheme

1.1 Reed Solomon Codes

The codes of Reed Solomon are non binary BCH codes belonging to the Galois fields GF ($q=2^4$). Each symbol q-areas of the body can be represented by **m** binary elements. The main Reed Solomon code parameters are (n, k, d) with n representing the length words of the codes, k representing the length of the information messages and d its Hamming distance. The (15, k, d) Reed Solomon code is wholly defined by the generator polynomial g (x). The primitive and irreducible polynomial is of the form $P(x) = x^4 + x + 1$. The Galois field contains 16 elements and α is a root of P(x). The generator polynomial g (x) characterizes the properties of the code. The size of the symbols is 4 bits.

A Reed-Solomon encoder or decoder needs to carry out arithmetic operations. These operations require special type of hardware or software functions to implement them. Reed-Solomon codes are QM-ary BCH codes (q=2). The parameters of these codes are

Code word length (n) = 2m - 1 (symbol)
Symbol length (width) = m (bit)
Error correcting capability = t (symbol)
Number of parity symbol (n – k) = 2t (symbol)

1.2 RS ENCODER

Encoding is achieved by affixing the remainder of a Galois Field (GF) generator polynomial division into message. The mathematics behind RS encoding is based on finite arithmetic. GF multipliers are used for the encoding the information. In this message of k*m bits is divided into k Symbols, each symbol being an m-bit element from GF (2m). A total 2t parity symbols are appended to the message. Mathematically the transmitted code-words are systematically encoded and defined in as a function of transmitted message m(x), generator polynomial g(x) and the number of parity symbols 2t as:

Multiply the message m(x) by x2t

From the parity b(x), by dividing the above result by generator polynomial,

From the codeword: $C(X) = m(x).x^{2t} + b(x)$

Where, g(x) is a generator polynomial of degree 2t and given by

$g(x) = (x + \alpha), (x + \alpha^2), (x + \alpha^3) \ldots\ldots.. (x + \alpha^{2t})$

1.3 Operation

Reed-Solomon codes are systematic codes, so for encoding the information symbols in the codeword are placed as the higher power coefficients. Therefore any Reed-Solomon encoder should perform the division operation and shifting operation properly. Both these operations can be performed using the Linear-feedback shift registers. The encoder architecture given below shows that one input to each multiplier is constant field element which is coefficient of generator polynomial g(x). For a specific block, the information polynomial m(x) is given to the encoder symbol by symbol basis. These symbols then appear at the output of the encoder after a desired latency is achieved, where control logic feeds it back through the adder to produce the related parity. This process continues until all of the k symbols of m(x) are input to the encoder. During this process, the control logic at the output enables only the input path, while keeping the path of parity disabled. The fig shows the block diagram of the RS encoder. The circuit performs polynomial division of the message polynomial m(x) by the field generator polynomial g(x). The remainder of the division, i.e. b(x) is stored in the 2t parity register.

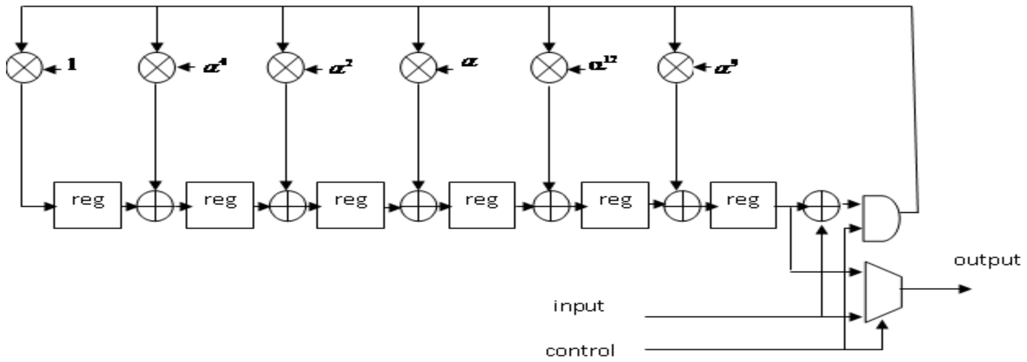

Figure 1: Encoder circuit for the RS (15, 9 and 7)

1.4 RS Decoding

The channel of transmission, especially in critical applications like space, submarine, nuclear introduces a big amount of noise into the information message. Thus the input codeword is received at the receiver end as codeword i.e. c(x), plus any channel induced errors, e(x), say r(x) = c(x) + e(x). The decoding procedure for Reed- Solomon coding scheme includes determining the locations and magnitudes of the errors in the received polynomial r(x). Locations are those powers of x in the received polynomials whose coefficients are in error. Magnitudes of the errors are the symbols that are added to the corrupted symbol to find the original encoded symbols. These locations and magnitudes constitute the error polynomial. Also, if the decoder is built to support erasure decoding, then the erasure polynomial has to be found. An erasure is an error with a known location. Thus, only the magnitudes of the erasures have to be found for erasure decoding. A Reed-Solomon (n, k) code can successfully correct as many as 2t = n-k erasures if no errors are present. Though the errors and erasures present, the decoder can then successfully decode if n-k ≥ 2t +e, where **t** is the number of errors, and **e** is the number of erasures present in the code.

A typical decoder usually follows the following stages in the decoding cycle,

1. Syndrome Calculation.

2. Determine error-location polynomial.

3. Solving the error locator polynomial.

4. Calculating the error Magnitude.
5. Error Correction.

1.4.1. Syndrome calculation

The first step in decoding the received symbol is to determine the data syndrome. Here the input symbols received are divided by the generator polynomial. The reminder should be zero. The parity is placed in the codeword to ensure that code is exactly divisible by the generator polynomial. If there is a remainder in the division, then there are errors. The remainder is called the syndrome.

1.4.2. Determination of error-locator polynomial

After computing the syndrome polynomial, next step is to calculate the error values and their respective locations in code. This stage involves the solving of the 2t syndrome polynomials from the previous stage. These polynomials have **v** unknown terms, where v is the number of unknown errors prior to decoding.

1.4.3. Solving the error locator polynomial:

Once the error locator and error evaluator polynomial has been obtained, the next step is to evaluate the error polynomial and to obtain its roots. The roots that obtained will now point to error location in the received message. RS decoding generally implements the Chien search algorithm to implement the same.

1.4.4. Calculate error value

Once the errors are located, the next step is to use the syndromes and the error polynomial roots to derive the error values. Forney's Algorithm is used for this purpose. It's an efficient way of performing a matrix inversion, and it involves two main stages.

1.4.5. Error correction

If the error symbol has any set bit, it means that the corresponding bit in the received symbol is having an error, and must be inverted. To automate this correction process each of the received symbol is read again, and at the each error location the received symbols are XORed with the error symbols. Thus the decoder corrects any errors as the received word is being read out from it.

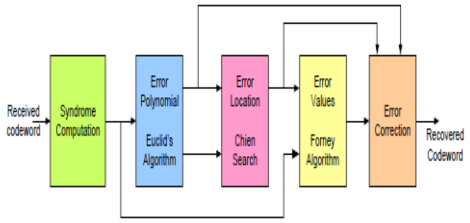

Fig [b] General Architecture of RS Decoder

2. APPLICATIONS OF RS CODES

The Digital Audio Disc

It can safely be claimed that Reed-Solomon codes are the most frequently used digital error control codes in the world. This claim the general condition in present that the digital audio disc or compact disc uses Reed-Solomon codes for error correction and error concealment. Special properties of Reed-Solomon (RS) codes make the sound quality of the compact disc as impressive as it is (the signal-to-noise ratio at the output exceeds 90 dB). The compact disc system uses pair of cross-interleaved Reed- Solomon codes. However, when Convolutional and Reed-Solomon codes are used in concatenated systems, acceptable coding gains are achievable. A Convolutional code is used as internal code, while a Reed-Solomon code is used to correct errors at the output of the Convolutional (Viterbi) decoder. The most famous application of the concatenated Reed-Solomon system was in the Voyager expeditions to Uranus and to Neptune.

Error Control coding for Systems with Feedback:

Wicker and Bartz examine various means for using Reed-Solomon codes in applications that allow the transmission of information from the receiver back to the transmitter. Such applications include mobile data transmission systems and high-reliability military communication systems.

Spread-Spectrum Systems:

Reed-Solomon codes can be used in the design of the hopping sequences in spectrum. If these sequences are selected carefully, the interference by unintended users in a multiple access environment can be reduced to great amount.

Satellite Broadcasting or Digital Video Broadcasting (DVB):

The demand for satellite transponder bandwidth continues to grow, especially in the area of television and IP traffic. Transponder availability and bandwidth constraints have put limits on this growth, because transponder capacity is determined by the selected modulation scheme and Forward Error Correction (FEC) rate. BPSK coupled with traditional Reed Solomon and Viterbi codes have been used for nearly 20 years for the delivery of digital satellite TV.

3. BLOCKSIZE PERFORMANCE OF RS CODES

As seen the family of curves in Figure 5 where the rate of the code is held at a constant *0.8*, while its block size increases from *n=7* symbols (*m=3* bits per symbol) to *n=127* symbols (with *m=7* bits / symbol). Thus the block size increases from *21, 60* bits to *889* bits. It is seen that the error-correcting codes become more efficient (error performance improves) as the block size increases because the effect of noise becomes less than that for small block size, the noise duration has to represent a relatively small percentage of the large code word, where the received noise should be averaged over a long period of time. This makes R-S code an attractive choice whenever long block code lengths are desired, hence optimum chosen is large code word as it is possible to approach better performance. But, very large code word will increase complexity in implementations as well**.**

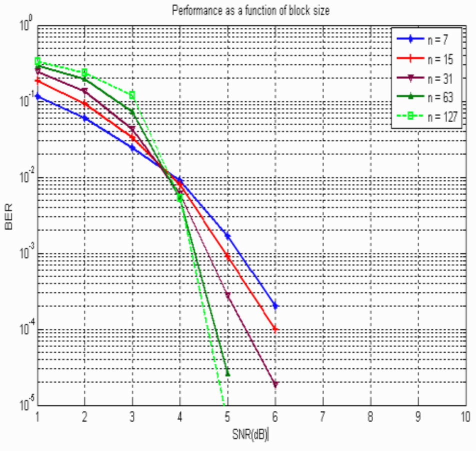

Fig [c] Reed-Solomon performance as a function of symbol

4. CONCLUSION

Reed–Solomon codes are one of the most powerful and efficient non-binary error – correcting codes for detecting and correcting burst errors. The analysis shows that code gain with high code rate is better than that of the lower one. The analysis showed that code gain with high code rate is better than that of low code rate. The work also showed that the error-correcting codes become more efficient as the block size increases because the effect of noise becomes less than that for small block size. An irreducible generator polynomial is used for generating the encoded data called codeword. All encoded data symbols are elements of the Galois field defined by the parameters of the application and the properties of the system. Usually the encoder is implemented using linear shift registers with the feedback. Then the decoder checks the received data for any errors by calculating the syndrome of the codeword by division. If an error is detected, the process of correction begins by locating the errors first. The approximate location of the errors is calculated by using “Chien-search algorithm”. Then the magnitude of the error is calculated using Forney’s algorithm. The magnitude of the error is added to the received codeword to obtain a correct codeword. Reed–Solomon codes are efficiently used for compact discs to correct the bursts which might occur due to scratches or fingerprints on the discs. Reed–Solomon is the single best solution for error control system that can match reliability performance in mobile environment. RS codes are finding increasing use in applications where highly efficient information transfers over bandwidth in the presence of data-corrupting noise is desired. The results showed that the area occupied is low and the latency is very high i.e. both the parameters are convincing. Hence, we have decreased the latency and the area occupied by adopting architecture in which each block is pipeline and/or parallelized.

5. ACKNOWLDEMENT

We would like to give special thanks to Prof. Swati V. Sakhare and Prof. Deepak Kushalani for their support and encouragement towards us.

6. REFERENCES:

[1] B.Tiwari and M.Rajesh, “FPGA Implementation of RS Codec for digital Video Broadcasting”, VSRD-IJEECE, Vol. 2, 2012, 86-77

[2] “Reed-Solomon error correction”, http://en.wikipedia.org/wiki/Reed-Solomon_error _correction, Jan. 11, 2011.

[3] “Reed-Solomon (RS) Coding Overview”, VOCAL Technologies, Ltd., Rev. 2.28n, 2010.

[4] Bernard Skalar and Pabritra Kumar Ray, “Digital Communications: Fundamentals & Applications”, Pearson, Edition 2, Jan. 2009.

[5] Kanny Chung Chun Wai and Dr Shanchieh jay Yang “Field Programmable Gate Array Implementation of Reed—Solomon code RS (255,239)”, http://www.ce.rit.edu/~sjyeec /paper/workshop-rs-codec.pdf, Sept. 10, 2010.

[6] A.Hikmat “Implementation of Reed Solomon Encoder/Decoder Using FPGA”, Journal of Engineering and Development, Vol. 10, No 3, September 2006.

[7] "Reed Solomon decoder", Lattice Semiconductor Corporation 2012. Http://www.latticesemi.com/products/intellectualproperty/ipcores/reedso lomondecoder.cfm

[8] H. Lee and A. Azam " pipelined recursive modified Euclidean algorithm block for low-complexity, high-speed Reed Solomon decoder", ELECTRONICS LETTERS 18th September 2003 Vol. 39 No. 19.

[9] Peter J. Ashenden, “The Designer‘s Guide to VHDL”, Morgan Kaufmann Publishers, Edition 2, 2004.

[10] J.Bhasker, “A VHDL PRIMER”, Prentice-Hall India, Edition 3, 1998.

[11] Peter J. Cameron, “Galois Fields,‖ The Encyclopedia of Design theory”, May 30, 2003.

[12] Mustafa ELHAROUSSI, Asmaa HAMYANI, Mostafa BELKASMI ENSIAS RABAT MAROC, “VHDL Design and FPGA Implementation of a Parellel Reed-Solomon (15, K, D) Encoder/Decoder” (IJACSA) 2013, Vol. 4

[13] Yung-Kuei Lu; Ming-Der Shieh, “Efficient Architecture for Reed-Solomon Decoder” VLSI Design, Automation and Test (VLSI-DAT) 2012 International Symposium on Digital Object Identifier.

[14] Aqib. Al Azad, Minhazul. Huq, Iqbalur. Rahman Rokon, “Efficient Hardware Implementation of Reed-Solomon Encoder and Decoder in FPGA using Verilog” (ICAEPE'2011) Bangkok Dec., 2011

[15] Barbosa, T.C.; Moreno, R.L.; Periera, T.C.; Ferriera, L.H.C. “FPGA Implementation of a Reed-Solomon CODEC for OTN G.709 Standard with Reduced Decoder Area” Wireless Communications Networking and Mobile Computing (wicom), 2010 6th International Conference on Digital Object Identifier.

[16] SUBHASHREE DAS “VHDL IMPLEMENTATION OF REED-SOLOMON CODING ” M.Tech. Thesis, Electronics and Communication Engineering, NIT Rourkela, 2011.

[17] V.K. Agrawal, Gaurav Mittal, Pankaj Goel, “REVIEW OF REED SOLOMON CODE FOR ERROR DETECTION AND CORRECTION” IJRIME JUNE-2012 Vol.2