SIMULATION AND SYNTHESIS OF INTERPOLATION BASED CHASE BCH DECODER

J. Chinna Babu,B. VeerendraVarma,
1Assistant Professor, Dept. of ECE, 2Dept. of VLSI System design,
Annamacharya Institute of Technology & Sciences, Rajampet, Andhra Pradesh, India

ABSTRACT

BCH codes are used in many systemssuch asflash memory,digital video broadcasting,and optical communications.By trying 2ηtest vectors, the softdecision Chase decodingalgorithm of BCH codes can achieve significant coding gain than harddecision decoding. Traditional one-pass Chase schemes findthe error locators based on the Berlekam's algorithm and needhardwaredemanding selection methods to decide which locatorcorresponds to the correct code word. In this brief, a new interpolationbased onepass Chase decoder is proposed for BCHcodes. By using the binary property of BCH codes, aninnovative lowcomplexity method is developed to select theinterpolation output which leads to successful decoding without bringingany performance loss. The code word recovery step is simplified through important mathematical derivations.From architectural analysis, the proposed decoder with η= 4fora (4200, 4096) BCH code has 2.3 times higher efficiency interms ofthroughput-over-area ratio than the prior one-pass Chase decoderbased on the Berlekamp's algorithm, while achieving the sameerror-correcting performance..

Keywords: BCH Codes, Syndrome Block, Chien search block, Error detection

1. INTRODUCTION

The information theory and coding theory are used intelecommunication applications computer communication. Error detection and correction are the techniques used in the above mentioned applications that enable reliable delivery of digital data over unreliable communication channels. Many communication channels are subjected to channel noise which introduces the errors during transmission of messages from the source to receiver. The channel coding theory states that the reliable transmission is achievable by performing proper coding.

Channel Coding is the technique used to sustain the originality of the information bits, to avoid the retransmission of information bits as well as to detect and correct any error which has been occurred during transmission. Error detection is the detection of errors caused by noise or other impairments during transmission from the transmitter to the receiver. It uses the concept of redundancy, which means adding of extra bits for detecting errors at the destination. In error correction the receiver can use any of the error-correcting code, which can automatically corrects certain errors and enables reconstruction of the original data.

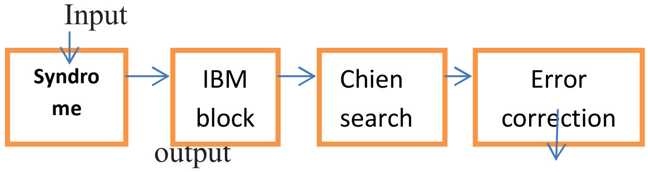

Figure 1: Block Diagram of BCH Decoder

2. INTERPOLATION BASED CHASE BCH DECODER

An (n,k) *t*-bit error-correcting BCH code over GF(2p) is a subfield subcode of an (n,k) t-symbol error-correcting RS code over GF(2p). Inother words, all the (n,k) BCH code words form a subset of the (n , k) RS code words. n k= 2t, and n kis equal to or slightly less than *pt*. The interpolation based decoding is developed based on the interpretation that the code word symbols are evaluation values of the message polynomial. BCH codes cannot be encoded this way since the evaluation values of a binary message polynomial over finite field elements are usually not binary. Hence, BCH code words are considered as RS code words in order to apply the interpolation based decoding. Applying RS systematic re-encoding to the last kcode positions, n–kpoints remain to be interpolated for each test vector. The same backward–forward interpolation scheme can be adopted to derive the interpolation results of all vectors in one run. Nevertheless, by making use of the property that *r* is binary in BCH decoding, substantial simplifications can be made on the polynomial selection and code word recovery steps. There are 2pkcode words for the (n,k) RS code over GF(2p). However, only a tiny proportion, 2kof them, are also code words of the (n,k) binary BCH code. Hence, the chance of returning a binary BCH code word is extremely small if the test vector is undecodable, particularly for long codes. Inspired by this, the interpolation output polynomials can be selected based on whether they will lead to binary code words. Nevertheless, testing if each symbol in the decoded word is binary requires the code word recovery step to be completed first. Instead, we propose to check only a few symbols that are easy to compute from the interpolation output. Which symbols to test depends on the specifics of the computations used in the decoding. Adopting the re-encoding technique, the decoding is actually carried out on ¯*r* = *r* + *φ*. Then, the returned code word ¯*c* is added up with *φ*to compute the code word *c*. φi = riis binary for i∈S. In addition, using the code word recovery¯*ci* for *i* ∈*S*_ is zero unless the corresponding code position is a root of *q*1(*x*). *q*1(*x*) has at most *t* roots and *t* _ *k*_ for high-rate codes. Therefore, *ci* for *i* ∈*S*_ is mostly binary and cannot be used to tell whether the entire *c* is binary. On the other hand, ¯*ci* and *φi*for *i* ∈¯*S*_ are mostly non-binary for undecodable cases, and the chance that ¯*ci* + *φi*(*i* ∈¯*S*_) is binary is extremely small. Here, ¯*S*_ denotes the first *n k*_ code positions. Accordingly, we propose to select the interpolation output whose corresponding ¯*c* + *φ*is binary in the first two symbols. *φ*is available, and the two symbols of ¯*c* can be easily computed from the interpolation output as will be explained in the next paragraph. The BER of the chase BCH decoding adopting this polynomial selection the test vectors are ordered according to reducing reliability as much as possible, and the first interpolation output passing the test is selected. In Fig.2,

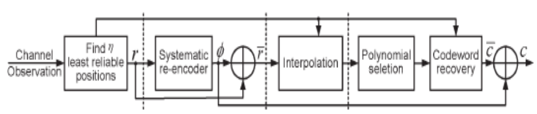

Fig. 2.Re-encoded interpolation-based Chase decoder.

The proposed scheme almost overlaps with that for the Chien search based polynomial selection. Testing any two symbols in ¯*S*_ would yield similar results. However, testing only one symbol leads to very small performance loss since some symbols can be accidentally binary in undecodable cases. When a test vector is decodable, the corresponding interpolation output has a factor *y* *f*(*x*)', where *f*(*x*)' is the message polynomial for the correct ¯*c* in evaluation map encoding. Reversing the coordinate transformation, it has been derived that

$$\bar{f}(x) = \frac{q_0(x)\ v(x)}{q_1(x)\ s_{f(x)}}$$

The bch decoder has four modules as mentioned below

- Syndrome calculator
- solving the key equation
- Error location
- Error correction

3. SYNDROME CALCULATOR

The syndrome calculator is the first module at the decoder also the design of this module is almost same for all the BCH code decoder architecture. The input to this module is corrupted code word. The equations for the code word, received bits and the error bits are given in equations Codeword equation

c(x) = c0 + c1x + c2x2 + ... + cn-1xn-1
Received bits equation
n r(x) = r0 + r1x + r2x2 + ... + rn-1xn-1
Error bits equation
e(x) = e0 + e1x + e2x2 + ... + en-1x¯

Thus, the final transmitted data polynomial equation is given as below: r(x) = c(x) + e(x).The 1st step at the decoding process is to store the transmitted data polynomial in the buffer register and then to calculate the syndromes sj. The important characteristic of the syndromes is that depends on only error location not on transmitted information. The equations of the syndromes are given as follows [4]: Define the syndromes *Sj*as

$$S_j = \sum_{i=0}^{n-1} r_i \alpha^{i \cdot j} \qquad \text{for } (1 \leq j \leq 2t).$$

Rewrite the syndrome equation as

$$S_j = \sum_{i=0}^{n-1} (c_i + e_i)\alpha^{i \cdot j} = \sum_{i=0}^{n-1} c_i \alpha^{i \cdot j} + \sum_{i=0}^{n-1} e_i \alpha^{i \cdot j}$$

The above equation is indicates the output of the syndrome calculator. From the equation it can be observed that the syndromes are depends on only error polynomial e(x), so if there is no error occurs during the transmission then all the generated syndromes will be zero.

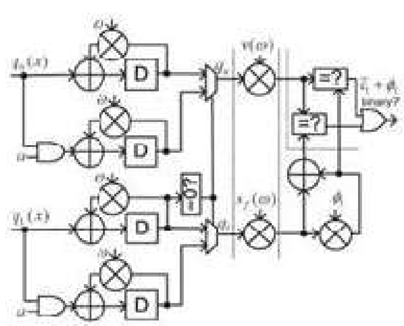

Figure 3: Architecture for Proposed Polynomial Selection

4. ERROR LOCATION – CHIEN'S SEARCH

To calculate the error location is the next step of decoding process, which can be done using chain search block.
The roots are calculated as follows
(1) For each power of α for ($j = 0$ *to* $n - 1$), α*j* is taken as the test root
(2) Calculate the polynomial coefficients, of the current root using, coefficients of the past iteration, using, **Λi(j) = Λi(j-1) αi**during the jthiteration
(3) Calculate the sum of the polynomial coefficients
(4) Continue to Step 1 till j = n-1

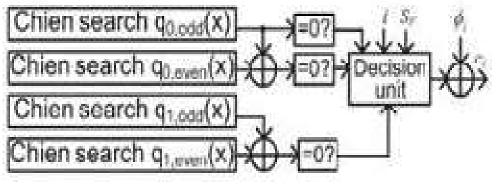

Figure 4: ChienSearch architecture – Error Location

5. ERROR CORRECTION

The output of the chain search block is called roots of equation. The reciprocal of the roots of equations are added with the corresponding location of the corrupted code word received by decoder. The result of this addition is the original code word that was encoded by the encoder before transmission.

6. SIMULATION RESULTS

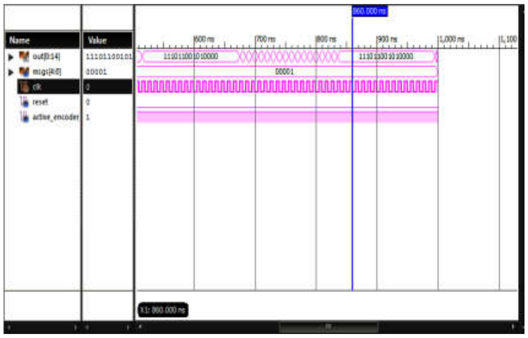

Figure 5: LFSR encoder

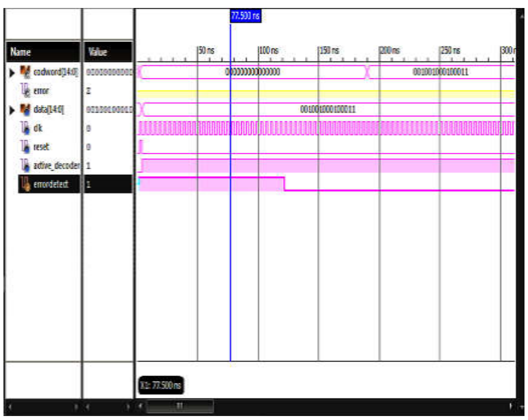

Figure 6: Error detected

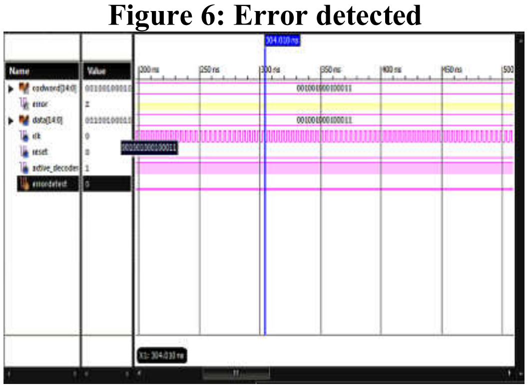

Figure 7: Corrected

Figure 8: Syndrome calculation

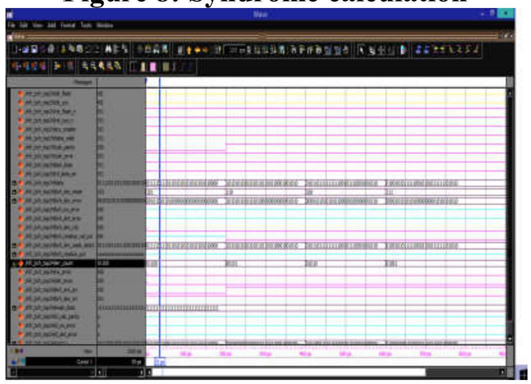

Figure 9: Interpolation based BCH decoder

7. CONCLUSION

This brief developed an efficient interpolation-based one pass Chase BCH decoder. By making use of the binary property of BCH codes, novel polynomial selection and code word recovery schemes were proposed. In particular, the proposed polynomial selection led to significant complexity reduction without sacrificing the error-correcting performance. Future work will be directed to further speeding up the interpolation based BCH decoder without incurring large area overhead.

References

[1] X. Zhang and J. Zhu, "Algebraic soft-decision decoder architectures for long Reed-Solomon codes".
[2] Y.-M. Lin, H.-C.Chang, and C.-Y. Lee, "An improved soft BCH decoder with one extra error compensation," in Proc. IEEE Int. Symp. Circuits Syst., May 2010, pp. 3941–3944.
[3] Y. Wu, "Fast Chase decoding algorithms and architectures for Reed–Solomon codes," IEEE Trans. Inf. Theory, vol. 58, no. 1, pp. 109– 129, Jan. 2012.
[4] IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 57, no. 10, pp. 787–792, Oct. 2010.
[5] hhhN. Kamiya, "On algebraic soft-decision decoding algorithms for BCH codes," IEEE Trans. Inf. Theory, vol. 47, no. 1, pp. 45–58, Jan. 2001.
[6] R. Kötter, "On algebraic decoding of algebraic-geometric and cyclic codes," Ph.D. dissertation, Dept. Elect.Eng., Linköping Univ., Linköping, Sweden, 1996.
[7] X. Zhang, J. Zhu, and Y. Wu, "Efficient one-pass Chase soft-decision BCH decoder for multi-level cell
[8] NAND flash memory," in Proc. IEEE Int. Midwest Symp. Circuits Syst., Aug. 2011, pp. 1–4.
[9] W. J. Gross, F. R. Kschischang, R. Koetter, and P.
[10] Gulak, "A VLSI architecture for interpolation in soft-decision decoding of Reed- Solomon codes," in Proc. IEEE Workshop Signal Process. Syst., Oct. 2002, pp. 39–44.