INTERNATIONAL JOURNAL OF RECENT TRENDS IN ENGINEERING & RESEARCH
Impact Factor 4.101 ISSN -2455-1457
RECENT ADVANCES IN ELECTRONICS & COMMUNICATION ENGINEERING (RAECE-2K17)

High speed Single Symbol Error Correction Codes Based on Reed Solomon Codes using verilog

Kurumaiah. N
Assist. Prof. Dept. of ECE
Nalla Narasimha Reddy Education Society's group of institutions

Sneha Talari
Assist. Prof. Dept. of ECE
Nalla Narasimha Reddy Education Society's group of institutions

Abstract
To keep away from information debasement, error correction codes (ECCs) are broadly used to secure recollections. ECCs present a postpone punishment in getting to the information as encoding or translating must be performed. This restrains the utilization of ECCs in rapid recollections. This has prompted the utilization of basic codes, for example, single error correction double error detection (SEC-DED) codes. In any case, as innovation scales multiple cell upsets (MCUs) turn out to be more typical and cutoff the utilization of SEC-DED codes unless they are joined with interleaving. A comparative issue happens in a few sorts of recollections like DRAM that are ordinarily gathered in modules made out of a few gadgets. In those modules, the insurance against a gadget disappointment as opposed to detached piece blunders is additionally alluring. In those cases, one alternative is to utilize further developed ECCs that can remedy numerous piece blunders. The primary challenge is that those codes ought to limit the deferral and region punishment. Among the codes that have been considered for memory assurance are Reed-Solomon (RS) codes. These codes depend on non-paired images and in this way can redress different piece blunders. In this paper, single image blunder adjustment codes based on Reed-Solomon codes that can be executed with low deferral are proposed also, assessed. The outcomes demonstrate that they can be executed with a generously bring down postponement than customary single blunder adjustment RS codes.

Keywords
Error correction codes, single error correction double error detection (SEC-DED) codes .

Contents

1. INTRODUCTION

Information defilement caused by mistakes is a noteworthy issue in recollections. Mistakes can be caused for instance by radiation prompted delicate blunders that influence at least one memory cells and change their qualities. Different sorts of disappointments cause perpetual harm with the end goal that the gadget never again gives identified and revised. These extra bits diminish the powerful limit of the memory. Different overheads presented by the ECC are the encoding and interpreting hardware. This hardware has an effect additionally on the deferral as the information must be encoded when composing into the memory and decoded when perusing from it. By and large, the deciphering is more intricate than the encoding and restricts the speed of the ECC. Generally single error correction double error detection (SEC-DED) codes are utilized to ensure recollections. These codes have a base Hamming separation of four with the end goal that solitary piece mistakes can be amended while two fold blunders are identified and not miscorrected. SEC-DED codes can be executed with a generally low range and postpone overhead and a few enhancements have been proposed as of late. Numerous piece blunders are an issue when SEC-DED codes are utilized. At the point when numerous mistakes influence cells that are physically close, similar to the instance of radiation instigated multiple cell

remedy information. To guarantee that information isn't tainted when disappointments happen, error correction codes (ECCs) are broadly utilized as a part of recollections. ECCs include some extra equality check bits to every memory word with the end goal that mistakes can be upsets (MCUs), SEC-DED codes can be consolidated with interleaving to guarantee that the mistakes influence just 1 bit for each consistent word. That is likewise the situation when a blunder causes the breakdown of a gadget

in a memory module. All things considered, the word is separated in sub-squares and an ECC is utilized for each of them. At that point the sub-pieces are interleaved in the gadgets with the end goal that in a gadget there is just 1 bit of a given sub-piece. In any case, the utilization of interleaving affects the memory plan and can increment region and power. For a memory module, the utilization of interleaving expands the quantity of equality check bits required, as extra bits are required per each of the sub-pieces. At long last, when various mistakes are caused by autonomous blunder occasions, all the more intense ECCs are expected to guarantee the amendment of mistakes. A wide number of various piece ECCs have been proposed to secure recollections. These incorporate Bose-Chaudhuri-Hocquenghem (BCH), Euclidean Geometry, Difference Set, Orthogonal Latin Squares and Reed-Solomon codes. Reed-Solomon (RS) codes have a particular component when contrasted and alternate codes: they are not paired. They utilize images from a Galois Field with the end goal that every image is spoken to by different bits. In this manner a SEC RS code can adjust different piece blunders as long as they influence a solitary image. This is extremely appealing for memory modules as when the quantity of bits in the gadgets coordinates those of the images in the RS code, disappointments in a single gadget can be amended. Actually, hence RS codes are ordinarily used to ensure primary recollections in PC frameworks for space applications, for example, those portrayed. As a rule, the information that structures a RS codeword are considered as polynomial coefficients with values having a place with the Galois Field. The polynomial relating to a codeword is a different of a particular polynomial, called generator polynomial g(x). Intrigued per users can allude to a traditional reading material on mistake control codes (e.g., [19]) for additionally points of interest. On account of SEC RS codes, the codeword is formed by affixing two check images to a dataword of k images. To characterize the SEC RS codes, normally the grid portrayal is favored, in this way the images forming the codeword are vectors of Galois Field components. With respect to double codes, a code is characterized utilizing a grid H, called equality check network. A codeword is a vector v, to such an extent that Hv ¼ 0. The encoding procedure of an information vector d is performed beginning from a generator grid G, by registering Gd ¼ v, which guarantees that Hv ¼ 0. Since the SEC RS code is normally divisible, the G network expect the frame G ¼ ½Ik P , where Ik is a k k personality lattice and P is of measurement k 2. Similarly as with other propelled ECCs one issue for the utilization of RS codes in recollections is the postponement presented by the unraveling. On account of RS codes, a few number juggling operations in the Galois Field are expected to encode or disentangle a square. This outcomes in a considerably bigger delay than that of customary SEC-DED codes. To relieve the affect on postpone when utilizing propelled

ECCs, one choice is to perform blunder discovery first and final when mistakes are identified continue to the remedy stage. As blunders are uncommon, the normal deferral will be near that of the blunder free case, which is given by the time required to perform mistake identification as it were. This is much lower than the time required for rectification. Be that as it may, even with this alteration postponement can be expansive as the blunder identification time for RS codes can be huge. In this paper, changes to customary RS codes to decrease the blunder location delay are introduced. This decrease would successfully decrease the postponement to get to the information when the specified plan of performing mistake location first and continuing to the next periods of unraveling just if there are mistakes is utilized. The proposed plans are likewise executed and contrasted and customary RS codes. The outcomes demonstrate that huge decreases in postponement can be accomplished with the proposed adjustments. Whatever is left of the paper is sorted out as takes after. Segment 2 portrays SEC RS codes breaking down in detail the decoder. In Section 3, the proposed SEC codes in light of RS codes are displayed. At that point in Section 4, the new codes are evaluated in terms of area and delay and compared with existing SEC RS codes. Finally, the conclusions of this work are summarized in Section 5.

2. SINGLE ERROR CORRECTION (SEC) REED SOLOMON (RS) CODES

Reed Solomon codes are a subclass of non paired BCH codes developed with images from a Galois Field GFðqÞ where q is regularly an energy of two. For q ¼ 2m;m piece images are utilized to build the code. A RS code has the accompanying parameters: most extreme piece length n ¼ q 1, number of equality check images n k ¼ 2t and least separation dmin ¼ 2t þ 1. Every one of those parameters are communicated as far as q-ary images. As an image has m bits, the most extreme piece length in bits is mðq 1Þ what's more, the quantity of equality check bits 2mt. At the point when t ¼ 1, the base separate is three and hence the code can revise single image mistakes. These blunders can influence numerous bits as long as every one of them have a place with a similar image. RS codes are usually communicated as RS(n, k, m). The equality check network for a RS code is built as appeared in condition (1) where a will be a primitive component in GF(q):

$$H = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\ 1 & \alpha^2 & (\alpha^2)^2 & \cdots & (\alpha^2)^{n-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & \alpha^{2t} & (\alpha^{2t})^2 & \cdots & (\alpha^{2t})^{n-1} \end{bmatrix}. \quad (1) \tag{1}$$

For a single error correction code only two parity check symbols are needed and therefore the matrix is simply:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \alpha & \alpha^2 & \alpha^3 & \cdots & \alpha^{n-1} \end{bmatrix}. \quad (2) \tag{2}$$

The decoding of a SEC RS code starts with the computation of the syndrome vector, which for a block r is simply:

$$s = H \cdot r. \tag{3}$$

(3)

The disorder vector can be utilized to identify blunders as takes after. At the point when every one of the bits in the vector are zero, no mistake is recognized. At the point when no less than 1 bit is one, a blunder is identified and along these lines mistake revision must be performed. The postponement and many-sided quality of disorder calculation depends on the estimations of the equality check lattice. For instance, for the first column of (1) the figuring is straightforward as all esteems are ones while for the second a few augmentations are required. Accepting a solitary image mistake e in position I in the square, the disorder would be:

$$s = \begin{bmatrix} e \\ e \cdot \alpha^i \end{bmatrix}, \tag{4}$$

(4)

Along these lines for a solitary image mistake if the second image of the disorder is partitioned by the main, the es-teem ai is gotten. The mistake is then situated by finding the example of the remainder and can be at last redressed by subtracting e from the ith image. Rectification in this way requires one division and one logarithm operation. As the images are from GFðqÞ, all operations are done over that field. This implies the interpreting turns out to be more mind boggling as q ¼ 2m develops. From condi-tions (2) and (4) it is straightforward which is the most extreme square length of a SEC RS code. Assume that we need to include a further section in the shape ½ 1 an to the network in condition (2). Since in the Galois Field GFðqÞ a ¼ aq1 ¼ 1, a blunder in this segment would create a similar disorder of a mistake in the first segment of the equality check framework. This associating implies that there are uncorrectable mistakes when the codeword length surpasses n. Nonetheless, there is a notable aug-mentation of the SEC RS code, to permit broadening the most extreme square length up to q þ 1 images. This should be possible including the segments ½0 1 and ½1 0 to the H lattice, getting the accompanying framework:

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 & 1 & 0 \\ 1 & \alpha & \alpha^2 & \alpha^3 & \cdots & \alpha^{n-1} & 0 & 1 \end{bmatrix}. \tag{5}$$

(5)

3. PROPOSED CODES

In this segment, two altered SEC RS codes are displayed to decrease the postponement for encoding and for blunder location. The primary alteration tries to enhance the equality check network to diminish the deferral for encoding and disorder calculation. The second joins the first

TABLE 1
Parameters of the Codes Evaluated

Code	Data bits	Parity Check Bits	GF(q)
SEC-RS(10,8,8)	64	16	2^8
SEC-RS$_{mod1}$(10,8,8)	64	16	2^8
2xSEC-RS(10,8,4)	64	16	2^4
2xSEC-RS$_{mod1}$(10,8,4)	64	16	2^4
SEC-RS(18,16,8)	128	16	2^8
SEC-RS$_{mod1}$(18,16,8)	128	16	2^8
2xSEC-RS $_{mod2}$ (19,16,4)	128	24	2^4

with an augmentation of the utilization of RS codes for a given an esteem of q to empower longer piece lengths. This decreases the encoding and deciphering postpone further as the increases and other number juggling oper-ations are done over a less difficult Galois Field.

Optimizing Parity Check Matrix

As clarified in the past segment, for a SEC-RS code, the calculation of the equality check grid is unequal. The duplication of the primary column of the lattice for a piece r requires no duplications on the Galois Field while the second requires n 1 duplications on the Galois Field. The principal proposed adjustment to RS codes tries to adjust the unpredictability of the two figurings. This is finished by utilizing the accompanying equality check grid:

$$H = \begin{bmatrix} 1 & 1 & \alpha^{-2} & 1 & \cdots & \alpha^{-(n-1)} & 1 & 0 \\ 1 & \alpha & 1 & \alpha^{3} & \cdots & 1 & 0 & 1 \end{bmatrix}, \qquad (6)$$

(6)

and the corresponding parity generator matrix:

$$G = \begin{bmatrix} & & & I_k & & \\ 1 & 1 & \alpha^{-2} & \cdots & \alpha^{-(n-2)} & \\ 1 & \alpha & 1 & \cdots & 1 & \end{bmatrix}. \qquad (7)$$

(7)

The network of condition (6) can be specifically acquired from condition (5) increasing the I-th sections, when I is even, for the esteem ai. The adjusted code is as yet a SEC code as the segments are straightly autonomous. With this changes the calculation of each of the check images amid the encoding procedure, and of the disorder images amid the interpreting procedure, requires the same num-ber of augmentations. This diminishes the length of the basic way and in this way brings down the deferral. The deciphering for the proposed codes is like that of SEC RS codes. Assume to have a blunder of greatness e in the ith image. In the event that the disorder vector ½S0;S1¼ ½S0;0, at that point I ¼ n 1 also, e ¼ S0. On the off chance that ½S0;S1¼ ½0; S1, at that point I ¼ n and e ¼ S1. Something else, the disorder vector will be ½S0;S1¼ ½e; eaiwhen I is even and ½S0;S1¼ ½eafig; ewhen I is odd. Along these lines, S1=S0 ¼ ai for all I < n 1 and we can register the mistake area. At that point we can recover e ¼ S0 for even estimations of I, and e ¼ S1 for odd estimations of I and right the blunder. This procedure requires one division also, one logarithm, the same as the disentangling of customary SEC RS codes.

Extension to Longer Block Lengths

Another factor that effects the deferral of RS encoders and

TABLE 2
Encoder and Syndrome Computation Delay

Code	Data bits	Encoder	Syndrome
SEC-RS(10,8,8)	64	0.32	0.32
SEC-RS$_{mod1}$(10,8,8)	64	0.30	0.28
2xSEC-RS(10,8,4)	64	0.31	0.31
2xSEC-RS$_{mod1}$(10,8,4)	64	0.26	0.28
SEC-RS(18,16,8)	128	0.47	0.48
SEC-RS$_{mod1}$(18,16,8)	128	0.41	0.42
2xSEC-RS $_{mod2}$ (19,16,4)	128	0.32	0.32

decoders is the extent of the Galois Field that is utilized to build the code. This is because of the expansion of the unpredictability of the math operations for bigger Galois Fields. As clarified in the past area, for a given GF(q) the greatest square size of a customary RS code is q 1 images. This confinement powers the utilization of bigger Galois Fields for extensive square sizes hence expanding the deciphering delay. This issue can be explained by utilizing an adjusted SEC RS code with the accompanying H grid:

$$H = \begin{bmatrix} \alpha & 1 & 1 & \alpha^2 & 1 & 1 & \cdots & 1 & 0 & 0 \\ 1 & \alpha & 1 & 1 & \alpha^2 & 1 & \cdots & 0 & 1 & 0 \\ 1 & 1 & \alpha & 1 & 1 & \alpha^2 & \cdots & 0 & 0 & 1 \end{bmatrix}. \quad (8)$$

(8)

With this alteration, the square length can be up to 3ðq 1þ images. The extra line permits to segregate between disorder vectors that generally could have a similar disorder esteem, as is appeared in the accompanying case:

Example: Assume we have m ¼ 3 bits, and along these lines q ¼ 8 and n ¼ 7, what's more, assume we have a mistake of greatness e ¼ a6 in the first image (the one comparing to the H segment ½ a1 1). The comparing disorder would be s1 ¼ ½1 a6 . Presently assume to have a blunder of greatness e ¼ 1 on the seventeenth image, that relates to the H section ½ 1 a6. For this situation the disorder vector would be s2 ¼ ½1 a6. We can see that the two disorders vary just on the third vector arrange. This extra data permits to separate between disorders that something else would be indistinguishable. The disentangling is like the past case. Assume we have a mistake of extent e in the ith image. On the off chance that the disorder vector ½S0;S1;S2 contains two zeros, the mistake is in the relating check image. Something else, two disorder images will have the same esteem, relating to the mistake greatness e. The third disorder image (the one with an esteem not the same as the other two) will have an estimation of e aceilðði=3þ and in this way the blunder area can be gotten. All the more decisively, on the off chance that we signify Sa (with a ¼ 0, 1 or 2) this disorder image, the area I will be I ¼ 3 ðlogðSa=eÞ 1þ þ a. The interpreting takes again a few correlations, a division and a logarithm. The proposed codes have been approved utilizing irregular mistakes to guarantee that all single image blunders are adjusted. The encoder and decoder relating to the above displayed codes has been composed in VHDL and combined, and the outcomes are exhibited in the following segment.

4. EVALUATION

In this segment, the proposed changed RS codes are assessed and contrasted and existing RS codes. The application considered, comparatively to the situation is the security of memory modules with information widths of 64 or 128 bits regularly utilized as a part of registering to accept that the modules are fabricated utilizing 8bit memory gadgets and the security necessities are to have the capacity to redress any blunder that influences a solitary gadget. For those designs, a direct arrangement is utilize a SEC RS code with 8 bit images (i.e., m ¼ 8). For 64 bits, the code would be a SEC RS (10, 8, 8) code and for 128 bits a SEC RS (18, 16, 8) code. In the in the first place case to diminish the translating delay, one choice is to utilize two interleaved SEC RS (10, 8, 4) codes to such an extent that every gadget stores one image from each of the codes. This guarantees blunders influencing a single gadget would cause at most a solitary image blunder in each code and accordingly the mistakes can be redressed. As talked about some time recently utilizing 4 bit images improves the number juggling operations and subsequently brings down the deferral. For the adjusted codes, the principal alteration depicted in Section 3.1 will be meant as SEC RSmod1 and the second change as SEC RSmod2. On account of the primary alteration, a similar code parameters with respect to the conventional SEC codes can be utilized. Those are, for the 64 information bit module, a SEC RSmod1(10, 8, 8) code or two interleaved SEC RSmod1(10, 8, 4) codes and for the 128 information bit module a SEC RSmod1(18, 16, 8) code. The second adjustment empowers the utilization of two interleaved 4 bit images SEC RSmod2(19, 16, 4) codes for the 128 bits module. The parameters for the situations considered are compressed in Table 1. It can be watched that for a given number of information bits the quantity of equality check bits is the same aside from for the 2 interleaved SEC-RSmod2(19, 16, 4) code. This implies just the encoder and decoder should be changed to utilize the proposed adjustments. To assess the deferral and region of the diverse codes, they were executed in HDL and combined for a 45 nm library. As talked about some time recently, blunder adjustment is more perplexing than mistake discovery and in this way deferral can be lessened by performing mistake identification first with the goal that revision is done just when mistakes are recognized. This was the plan utilized as a part of the execution. For this execution the deferral in the blunder free case is given by that of disorder calculation. The clock cycle span is changed in accordance with that speed. If there should arise an occurrence of mistake, the deferral will be bigger and is given in the quantity of clock cycles required to redress the blunder. In a few cases, the mistake rectification may likewise restrict the speed notwithstanding when executed in various clock cycles. The deferral and zone comes about for the encoder and for the disorder calculation piece are

TABLE 3
Encoder and Syndrome Computation Area

Code	Data bits	Encoder	syndrome
SEC-RS(10,8,8)	64	769	876
SEC-RS$_{mod1}$(10,8,8)	64	840	978
2xSEC-RS(10,8,4)	64	704	746
2xSEC-RS$_{mod1}$(10,8,4)	64	672	762
SEC-RS(18,16,8)	128	1902	2112
SEC-RS$_{mod1}$(18,16,8)	128	1210	1731
2xSEC-RS $_{mod2}$ (19,16,4)	128	1830	2062

TABLE 4
Error Correction Block Area

Code	Combinational	Flip-flops	Total
SEC-RS(10,8,8)	7379	8448	15827
SEC-RS$_{mod1}$(10,8,8)	7506	8472	15978
2xSEC-RS(10,8,4)	3762	1754	5516
2xSEC-RS$_{mod1}$(10,8,4)	3836	1786	5622
SEC-RS(18,16,8)	8645	12134	20779
SEC-RS$_{mod1}$(18,16,8)	8709	12908	21617
2xSEC-RS $_{mod2}$ (19,16,4)	6650	2904	9554

introduced in Tables 2 and 3. The deferral is given in nanoseconds and the zone in mm2. For the encoder it can be watched that for 64 information bits the proposed change gives reserve funds of 6.25 percent when 8 bit im-ages are utilized and of 16.13 percent for 4 bit images. For 128 bits, the diminishment is 12.77 percent when the prin-cipal alteration is utilized and 31.91 percent at the point when the second one is utilized. The range of the 64 bits encoder is marginally expanded if there should arise an occurrence of m ¼ 8 (þ9%) and somewhat diminished at the point when m ¼ 4, (– 4.5%). For the 128 bits encoder, the region sparing is 36 percent for 8 bits images, and 4 percent when the RS (19, 16, 4) code is utilized. For the 64 bits disorder calculation obstruct, the defer sparing is 12 percent when 8 bit images are utilized and of 10 percent for 4 bit images. For the 128 bits disorder figuring, the lessening is 12.5 percent when the altered RSmod1(18, 16, 8) is utilized and 3 percent whenever RSmod2(19, 16,

4) is utilized. The range of the 64 bits disorder calculation increments by 11 furthermore, 2 percent for m ¼ 8 and m ¼ 4 individually. At long last, the region of the 128 bits disorder calculation square demonstrates a 18 percent lessening when the altered RSmod1(18, 16, 8) is utilized and 2.5 percent diminishment when RSmod2(19, 16, 4) is utilized. In synopsis, the proposed codes reliably diminish the encoding what's more, disorder calculation delay. The bigger decreases (over 30 percent) are acquired when the second alteration is utilized as for 128 bits the number jug-gling operations are done over GFð24Þ rather than over GFð28Þ. For culmination, Tables 4 and 5 demonstrate the range and deferral parameters for the blunder revision square of the decoders. Since an forceful pipeline is uti-lized to keep up rapid, the range of both the combinatorial and consecutive parts of the circuit has been detailed. It can be seen that the utilization of 4 bits images, permits a major zone sparing, since less pipeline stages are required. With respect to the speed of the mistake redress hinder, the decoder with 4 bits images can be continuously timed at the greatest recurrence given by the disorder calcula-

TABLE 5
Error Correction Block Delay

Code	Cycle delay	Number of cycles
SEC-RS(10,8,8)	0.43 (*)	10
SEC-RS$_{mod1}$(10,8,8)	0.39 (*)	10
2xSEC-RS(10,8,4)	0.31	4
2xSEC-RS$_{mod1}$(10,8,4)	0.28	4
SEC-RS(18,16,8)	0.45	10
SEC-RS$_{mod1}$(18,16,8)	0.41	10
2xSEC-RS $_{mod2}$ (19,16,4)	0.32	4
(*) maximum operating frequency of the decoder depends on this delay		

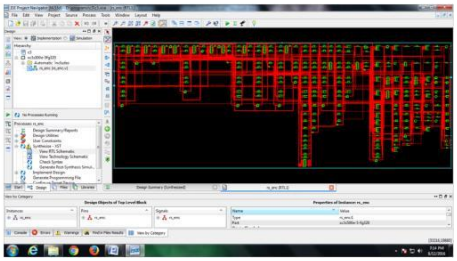

Figure 1. Synthesis

tion square, and can give the consequence of the mistake revision system after 4 clock cycles. Specifically, the proposed RSmod1(10, 8, 4) accomplishes the best planning outcomes for the 64 bits codeword, while the RSmod2(19, 16, 4) outflanks the 8 bits image based codes, both as far as clock cycle postponement and number of cycles. Indeed, all the 8 bits image codes require 10 clock cycles to accomplish high working frequencies, and on account of the SEC-RS (10, 8, 8) and SEC-RSmod1(10, 8, 8), they were not able come to the working recurrence of the disorder calculation square, along these lines restricting the most extreme working recurrence of the entire decoder. In any case, the SEC-RSmod1(10, 8, 8) code outflanks the standard RS (10, 8, 8) code with a deferral of 0.39 ns contrasted with 0.43 ns.

5. CONCLUSIONS

In this paper, new codes in view of changes of single error correction Reed Solomon (SEC RS) codes have been proposed with the target of diminishing deferral. The codes have been actualized and assessed. The cases utilized for assessment compare to genuine setups normally utilized as a part of memory modules. For those, the proposed codes empower critical defer diminishments in encoding what's more, deciphering delay. This makes the adjusted

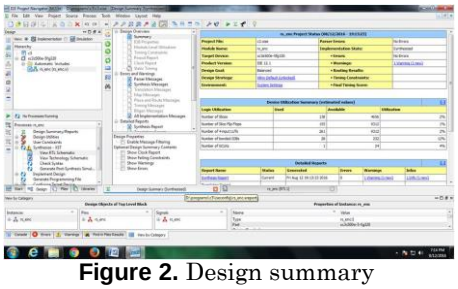

Figure 2. Design summary

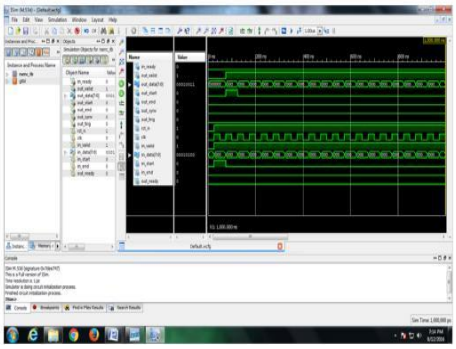

Figure 3. simulation

codes appealing for rapid recollections. Future work will consider the assessment of the proposed codes to secure different sorts of memory like for cases reserves.

References

[1] C. L. Chen and M. Y. Hsiao, "Error-correcting codes for semiconductor memory applications: A state-of-the-art review," IBM J. Res. Develop., vol. 28, no. 2, pp. 124–134, 1984.

[2] S. Lin and D. J. Costello, Error Control Coding, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 2004.

[3] M. Y. Hsiao, "A class of optimal minimum odd-weight column SEC-DED codes," IBM J. Res. Develop., vol. 14, pp. 395–301, Jul. 1970.

[4] M. Richter, K. Oberlaender, and M. Goessel, "New linear SEC-DED codes with reduced triple bit error miscorrection probability," in Proc. IEEE 14th Int. On-Line Testing Symp., 2008, pp. 37–42.

[5] V. Gherman, S. Evain, N. Seymour, and Y. Bonhomme, "Generalized parity-check matrices for SEC-DED codes with fixed parity," in Proc. IEEE 17th Int. On-Line Testing Symp., 2011, pp. 198–201.

