Design And Implementation Of DMC Encoder And Decoder For Secured Applications

V. Naga Jyothirmai

M. Tech Student, ECE (DECS) Gudlavalleru Engineering College Gudlavalleru, Krishna Dist, AP, India-521356
e-mail: vanjarapujyothirmai43@gmail. com

A. V. N. Tilak

Professor of ECE, Dean-P. G. Studies and R&D Gudlavalleru Engineering College Gudlavalleru, Krishna Dist, AP, India-521356
e-mail: avntilak@yahoo. com
Telephone No.: 08674-273737, Fax No: 08674-273957

Abstract

Recently memory systems have experienced significant technology developments. Error tolerance for emerging memory systems is of increasing importance. Error Correcting Codes (ECC) are frequently used for improving the reliability of a memory system. Errors are controlled in a memory system by using encoders and decoders for ECCs. ECCs do not protect from errors in the encoders and decoders. The main objective of this work is to design encoder and concurrent decoders to detect and correct errors in given pattern of input. Concurrent Error Detection (CED) employing Decimal Matrix Codes (DMC) for the syndrome generator reduces area overhead. These codes have the capability of correcting information even the transmission medium is very much susceptible for errors. Decimal Matrix codes can detect and correct errors in parallel manner. This design is implemented using Xilinx 14. 7 ISE Simulator in VHDL By using DMC codes area reduced by 13%.

Keywords: Memory Systems, Encoders and Decoders, ECC, Decimal Matrix Codes, *VHDL*

I. Introduction

Error rate is more in logic circuits. Most of the logic circuits consist of memory. From past ten years several researchers concentrated on memory cells. They designed circuits to reduce faults in memory. Even though the memory cells are protected, the error rate is not reduced to an appreciable level. Then the concentration is moved on to the surrounding circuitries of the memory to reduce error rate [1]. In most of the logic circuits at the transmitter, receiver ends encoder and decoder circuits respectively used[1]. To reduce error rate in logic circuits a fault secure encoder and decoder are needed. Single bit upset is a major concern about memory reliability, multiple cell upsets (MCUs) have become a serious reliability concern in some memory applications. In this work, novel decimal matrix code (DMC) based on divide-symbol is proposed to provide enhanced memory reliability. The proposed DMC utilizes decimal algorithm (decimal integer addition and decimal integer subtraction) to detect errors. The advantage of using decimal algorithm is that the error detection capability is maximized so that its reliability is enhanced. Besides, the encoder-reuse technique (ERT) is proposed to minimize the area overhead of extra circuits (encoder and decoder) without disturbing the whole encoding and decoding processes, because ERT uses DMC encoder itself to be part of the decoder.

II. DMC Encoder and Decoder

DMC is used to get reliability in the presence of MCUs with reduced performance overheads. Block diagram of the proposed system used in encoder and decoder circuits is shown in Figure. 1.

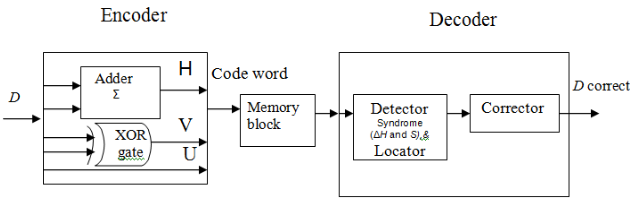
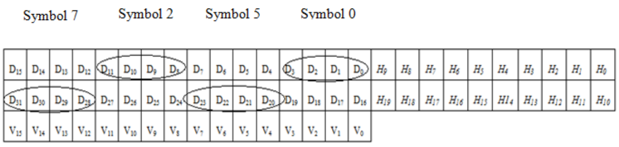


Figure 1:Block diagram of proposed DMC codes.

A. DMC Encoder

In the DMC, first, the input is divided to symbols and arrange in matrix form are performed, i. e. , the *N*-bit word is divided into *k* symbols of *m* bits ($N = k \times m$), and these symbols are arranged in a $k1 \times k2$ 2-D matrix ($k = k1 \times k2$, where the values of *k*1 and *k*2 represent the numbers of rows and columns in the logical matrix respectively). Second, the horizontal redundant bits *H* are produced by performing decimal integer addition of selected symbols per row. Here, each symbol is regarded as a decimal integer. Third, the vertical redundant bits *V* are obtained by binary operation among the bits per column. It should be noted that both divide-symbol and arrange-matrix are implemented in logical instead of physical. For a 32-bit word shown in Figure. 2. The cells from *D*0 to *D*31 are information bits. This 32-bit word has been divided into eight symbols of 4-bits each. $k1 = 2$ and $k2 = 4$ have been chosen simultaneously (k1=rows, k2=columns respectively). H_0-H_{19} are horizontal check bits. V_{15}-V_0 are vertical redundant bits, and the remaining bits U_{31}-

U_0 are the information bits which are directly copied from D_{31}-D_0. The *En* is enable signal. When $k = 2 \times 4$ and $m = 4$, the maximum correction capability is up to 5 bits and the number of redundant bits are 36. In order to enhance the reliability of memory, the error correction capability is considered, so $k = 2 \times 4$ and $m = 4$ are utilized to construct DMC.


Symbol 7
Symbol 2
Symbol 5
Svmbol 0

D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	H_9	H_8	H_7	H_6	H_5	H_4	H_3	H_2	H_1	H_0
D_{31}	D_{30}	D_{29}	D_{28}	D_{27}	D_{26}	D_{25}	D_{24}	D_{23}	D_{22}	D_{21}	D_{20}	D_{19}	D_{18}	D_{17}	D_{16}	H_{19}	H_{18}	H_{17}	H_{16}	H_{15}	HI_4	H_{13}	H_{12}	H_{11}	H_{10}
V_{15}	V_{14}	V_{13}	V_{12}	V_{11}	V_{10}	V_9	V_8	V_7	V_6	V_5	V_4	V_3	V_2	V_1	V_0										

Figure 2: 32-bit DMC logical organization

The encoder that computes the redundant bits using multibit adders and XOR gates is shown in figure.

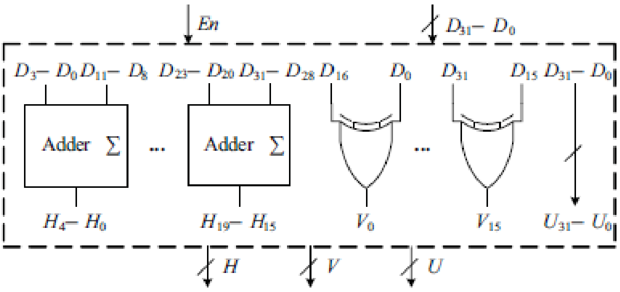


Figure 3: Encoder Structure

The horizontal redundant bits *H* can be obtained by decimal integer addition as follows:

$$H_4H_3H_2H_1H_0 = D_3D_2D_1D_0 + D_{11}D_{10}D_9D_8 \tag{1}$$

$$H_9H_8H_7H_6H_5 = D_7D_6D_5D_4 + D_{15}D_{14}D_{13}D_{12} \tag{2}$$

$$H_{14}H_{13}H_{12}H_{11}H_{10} = D_{16}D_{17}D_{18}D_{19} + D_{24}D_{25}D_{26}D_{27} \tag{3}$$

$$H_{19}H_{18}H_{17}H_{16}H_{15} = D_{20}D_{21}D_{22}D_{23} + D_{28}D_{29}D_{30}D_{31} \tag{4}$$

where "+"represents decimal integer addition. For the vertical redundant bits *V*, we have

$$V0 = D_0 \oplus D_{16} \tag{5}$$

$$V1 = D_0 \oplus D_{17} \tag{6}$$

and similarly for the remaining vertical redundant bits.

B. Fault-Tolerant Memory

During the encoding (write) process, information bits are given to the encoder, and then the horizontal redundant bits H and vertical redundant bits V are obtained from the DMC encoder. When the encoding process is completed, the obtained DMC code word is stored in the memory. If errors occur in the memory, errors can be corrected in the decoding (read) process[7]. Because of the advantage of decimal algorithm, the proposed system has higher fault-tolerant capability with lower performance overheads. In the fault-tolerant memory, the ERT technique is proposed to reduce the area overhead of extra circuits.

C. DMC Decoder

To obtain the corrected word decoding process is needed. The horizontal syndrome bits
$\Delta\, H_4H_3H_2H_1H_0$ and the vertical syndrome bits $S3 - S0$ can be calculated as

$$\Delta\, H_4H_3H_2H_1H_0 = H_4H_3H_2H_1H_0' - H_4H_3H_2H_1H_0 \tag{7}$$

$$S_0 = V_0' \oplus V_0 \tag{8}$$

and similarly for the other vertical syndrome bits, where "–" represents decimal integer subtraction. When $\Delta\, H_4H_3H_2H_1H_0$ and $S3 - S0$ are equal to zero the stored code word has original information bits in symbol '0' where no errors occur. When $\Delta H_4H_3H_2H_1H_0$ and $S3 - S0$ are nonzero, the induced errors are detected and located in symbol 0, and then these errors can be corrected by

$$D_{o\ correct} = D_0 \oplus S_0. \tag{9}$$

The proposed DMC decoder is divided to sub modules, for executing a specific task in the decoding process. Those sub modules are syndrome calculator, error locator, and error corrector. It can be observed from the Figure. 3 that the redundant bits must be recomputed from the received information bits *D* and compared to the original set of redundant bits in order to obtain the syndrome bits Δ*H* and *S*. Then error locator uses Δ*H* and *S* to detect and locate errors that occur in the input bits. Finally in the error corrector, these errors can be corrected by inverting the values of error bits.

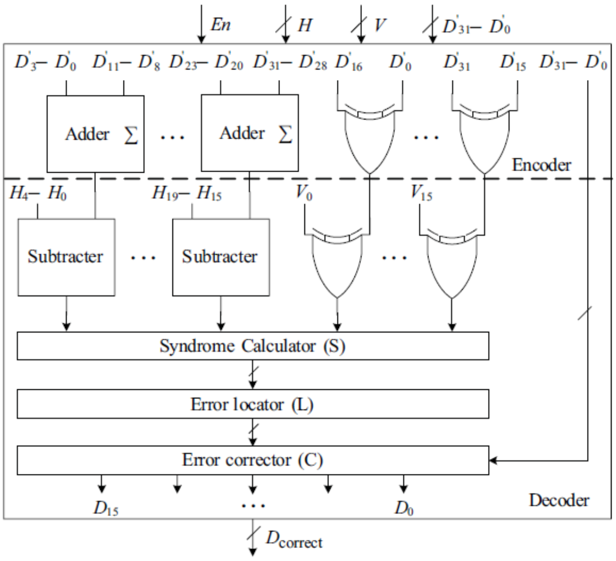
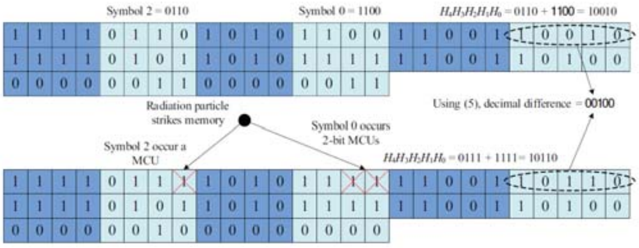


Figure 4: Decoder structure using ERT

In the proposed system area of DMC is minimized by reusing its encoder. This is called the Encoder Reusing Technique (ERT).

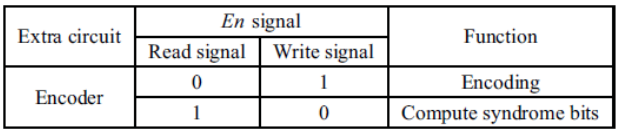

Symbol 2 = 0110
Symbol 0 = 1100
$H_4H_3H_2H_1H_0$ = 0110 + 1100 = 10010
Using (5), decimal difference = 00100
Radiation particle strikes memory
Symbol 0 occurs 2-bit MCUs
Symbol 2 occur a MCU
$H_4H_3H_2H_1H_0$ = 0111 + 1111= 10110

1	1	1	1	0	1	1	0	1	0	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0
1	1	1	1	0	1	0	1	1	0	1	0	1	1	1	1	1	1	0	0	1	1	0	1	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1										

1	1	1	1	0	1	1	1	1	0	1	0	1	1	1	1	1	1	0	0	1	1	0	1	1	0
1	1	1	1	0	1	0	1	1	0	1	0	1	1	1	1	1	1	0	0	1	1	0	1	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0										

Figure 5: The process of detecting and correcting error

The ERT can reduce the area overhead of DMC without disturbing the whole encoding and decoding processes shown in figure. 4. . It can be observed that the DMC encoder is also reused for obtaining the syndrome bits in DMC decoder. Therefore, the whole circuit area of DMC can be minimized as a result of using the existing circuits of encoder. The proposed decoder uses an enable signal *En* for deciding whether the encoder needs to be a part of the decoder or encoder. In other words, the *En* signal is used for distinguishing the encoder from the decoder, and it is under the control of the write and read signals in memory.



Extra circuit	*En* signal		Function
	Read signal	Write signal	
Encoder	0	1	Encoding
	1	0	Compute syndrome bits

Figure 6: Functions of enable signal

Therefore, in the encoding (write) process, the DMC encoder is only an encoder to execute the encoding operations shown in figure. 6. However, in the decoding (read) process, this encoder is employed for computing the syndrome bits in the decoder. These clearly show how the area overhead of extra circuits can be reduced.

When the decimal algorithm is used to detect errors, these errors can be detected so that the decoding error can be avoided as shown in figure. 5. . The detection procedure of decimal error detection using the proposed structure is the horizontal redundant bits $H_4H_3H_2H_1H_0$ are obtained from the original information bits in symbols 0 and 2

- $H_4H_3H_2H_1H_0 = D_3D_2D_1D_0 + D_{11}D_{10}D_9D_8$

= 1100 + 0110

= 10010.

When MCUs occur in symbol 0 and symbol 2 the bits in symbol 0 are upset to “1111” from “1100” ($D_3D_2D_1D_0$= 1111) and the bits in symbol 2 are upset to “0111” from “0110” ($D_{11}D_{10}D_9D_8$= 0111). During the decoding process, the received horizontal redundant bits $H_4H_3H_2H_1H_0$ are computed as follows:

$H_4H_3H_2H_1H_0 = D_{11}D_{10}D_9D_8 + D_3D_2D_1D_0$

= 0111 + 1111

= 10110.

Then, the horizontal syndrome bits Δ$H4H3H2H1H0$ can be obtained using decimal integer subtraction

$\Delta H_4H_3H_2H_1H_0 = H_4H_3H_2H_1H_0' - H_4H_3H_2H_1H_0$

= 10110 − 10010

= 00100.

The decimal value of Δ $H_4H_3H_2H_1H_0$ is not “0, ” which represents that errors are detected and located in symbol 0 or symbol 2 .

III. Reliability And Overheads Analysis

The proposed DMC has been implemented in HDL. It has superior protection level compared with other codes. This technique provides single-and double-error correction, but can also provide effective tolerance capabilities against large MCUs that exceed the performance of other codes.

A. Reliability Estimation

The reliability of system can be analyzed in terms of the mean time to failure (MTTF). It is assumed that MCUs arrive at memories following a Poisson distribution. For one word, the correctable probability *R (S)* after *S* radiation events can be given by [13].

$$R(S) = \sum_{i+j+\cdots+z \le T} P_i^1 P_j^2 \cdots P_z^S \tag{10}$$

where

T is the maximum number of errors and *P* is the correctable probability upon the reception of radiation event
S which causes *z* errors.
For a memory with *M* words, the correctable probability
J (S) after *S* radiation events can be given by[13].

$$J(S) = \sum_{a+b+\cdots+e=S} \frac{C_M^x}{M^x} R_a^1 R_b^2 \cdots R_e^x \tag{11}$$

where *x* (*x ≤ S)* is the number of words affected by radiation events, *C* is the selection of *x* from *M* words in memory, and *R* represents the correctable probability when *e* radiation events affect *x* words.
Then the MTTF can be given by[13]

$$\text{MTTF} = \int_0^{\infty} J(t)dt. \tag{12}$$

B. Overheads Analysis

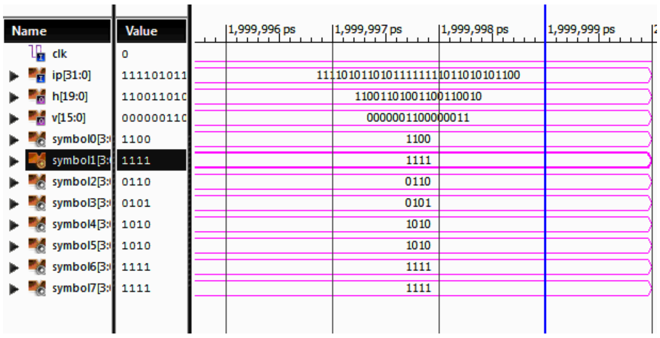
The area overheads of OLS and DMC are 80% and 63% respectively. This indicates that the memory with the proposed scheme performs faster than OLS codes. However the proposed DMC decoding algorithm is quite simple so that the overheads are minimal.

IV. Simulation results

The simulation results are obtained by executing the VHDL files of encoder, detector, memory, corrector, and decoder on the simulator tool.

Fig 7, 8 and 9shows simulation results of Encoder and Decoder. In decoder simulation results for error detector and error corrector are shown. The information vector is given to the encoder block. The information is encoded and horizontal and vertical parity check bits are generated by using decimal algorithm. The output of encoder is stored in memory. The output of memory is input to the decoder. In the decoder encoded vector is detected and corrected.

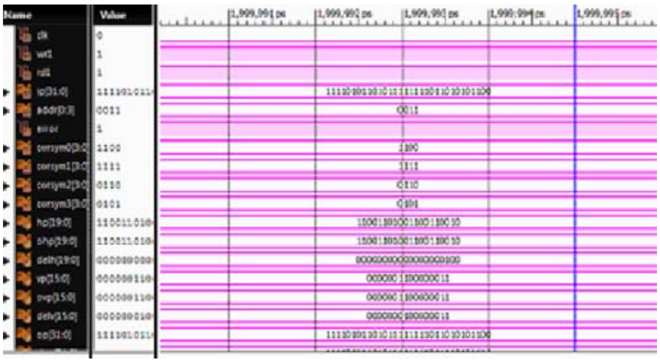
Simulation output of Encoder:



```
Name           Value       |1,999,996 ps   |1,999,997 ps   |1,999,998 ps   |1,999,999 ps
clk            0
ip[31:0]       111101011    11110101101011111111011010101100
h[19:0]        110011010    11001101001100110010
v[15:0]        000000110    0000001100000011
symbol0[3:    1100         1100
symbol1[3:    1111         1111
symbol2[3:    0110         0110
symbol3[3:    0101         0101
symbol4[3:    1010         1010
symbol5[3:    1010         1010
symbol6[3:    1111         1111
symbol7[3:    1111         1111
```

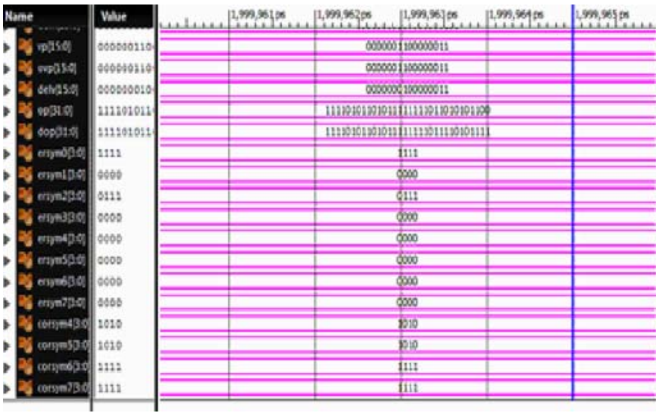
Figure 7: Simulation results of encoder

Simulation output of Decoder:



```
Name           Value       |1,999,991 ps   |1,999,992 ps   |1,999,993 ps   |1,999,994 ps   |1,999,995 ps
clk            0
wr1            1
rd1            1
ip[31:0]       111101011    11110101101011111111011010101100
addr[0:3]      0011         0011
error          1
corsym0[3:0]   1100         1100
corsym1[3:0]   1111         1111
corsym2[3:0]   0110         0110
corsym3[3:0]   0101         0101
hp[19:0]       110011010    11001101001100110010
shp[19:0]      110011010    11001101001100110010
delh[19:0]     000000000    00000000000000000000
vp[15:0]       000000110    0000001100000011
svp[15:0]      000000110    0000001100000011
delv[15:0]     000000010    0000000100000011
op[31:0]       111101011    11110101101011111111011010101100
```

Figure 8: Simulation results of error detector



```
Name           Value       |1,999,961 ps   |1,999,962 ps   |1,999,963 ps   |1,999,964 ps   |1,999,965 ps
vp[15:0]       000000110    0000001100000011
svp[15:0]      000000110    0000001100000011
delv[15:0]     000000010    0000000100000011
op[31:0]       111101011    11110101101011111111011010101100
dop[31:0]      111101011    11110101101011111111011110101111
ersym0[3:0]    1111         1111
ersym1[3:0]    0000         0000
ersym2[3:0]    0111         0111
ersym3[3:0]    0000         0000
ersym4[3:0]    0000         0000
ersym5[3:0]    0000         0000
ersym6[3:0]    0000         0000
ersym7[3:0]    0000         0000
corsym4[3:0]   1010         1010
corsym5[3:0]   1010         1010
corsym6[3:0]   1111         1111
corsym7[3:0]   1111         1111
```

Figure 9: Simulation results of error corrector

V. Conclusion

In this work, DMC was proposed for the reliability of memory. The DMC code utilizes decimal algorithm to detect errors, so that more errors are detected and corrected. The obtained results shows that proposed scheme reduces number of errors that occur in memory. The area reduction of DMC is 13% respectively.

VI. References

[1] Kazuteru Namba and Fabrizio Lombardi "Concurrent error detection of binary and nonbinary OLS Parallel decoders", IEEE transactions on device and materials reliability vol. 14, no. 1, march. 2014.

[2] P. Reviriego, S. Pontarelli, and J. A. Maestro, "Concurrent error detection for orthogonal Latin squares encoders and syndrome computation, "IEEE Trans. Very Large Scale Integr. (VLSI) Syst. , vol. 21, no. 12, pp. 2334-2338, Dec. 2013.

[3] S. Sabeen1, N. Bharathi Raja2 , "Enhanced memory reliability using parity matrix code" International Journal of Scientific & Engineering Research, Volume 5, Issue 4, April-2014.

[4] K. Namba and F. Lombardi, "Non-binary orthogonal latin square codes for a multilevel phase charge memory (PCM), " Dept. ECE, Northeastern Univ. , Internal Rep. , Jul. 2013.

[5] Arijit Raychowdhury, Georgia Institute of technology, "Spin torque devices in embedded memory, model studies and design space exploration", IEEE/ACM International conference, pp. 572-575, 2013.

[6] N. Savage, "Z-RAM takes on DRAM, " IEEE Spectrum, vol. 47, no. 7, p. 18, Jul. 2010.

[7] H. Naeimi and A. DeHon, "Fault secure encoder and decoder for nanomemory applications, " IEEE Trans. Very Large Scale Integr. (VLSI) Syst. , vol. 17, no. 4, pp. 473-486, Apr. 2009.

[8] R. Naseer and J. Draper, "Parallel ouble error correcting code design to mitigate multi-bit upsets in SRAMs, " in Proc. 34th Eur. Solid-State Circuits, pp. 222-225, Sep. 2008.

[9] C. Wilkerson, H. Gao, A. R. Alameldeen, Z. Chishti, M. Khellah, and S.-L. Lu, "Trading off cache capacity for reliability to enable low voltage operation, " in Proc. Annu. Int. Symp. Comput. Archit, pp. 203-214, 2008.

[10] D. Ielmini, A. L. Lacaita, and D. Mantegazza, "Recovery and drift dynamics of resistance and threshold voltages in phase-change memories, " IEEE Trans. Electron Device, vol. 54, no. 2, pp. 308-315, Feb. 2007.

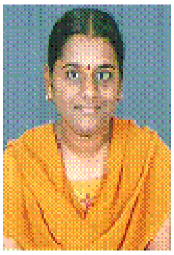
[11] G. C. Cardarilli, S. Pontarelli, M. Re, and A. Salsano, "Concurrent error detection in Reed-Solomon encoders and decoders, " IEEE Trans. Very Large Scale Integr. (VLSI) Syst. , vol. 15, no. 7, pp. 842-846, Jul. 2007.

[12] M. Kund, G. Beitel, C.-U. Pinnow, T. Röhr, J. Schumann, R. Symanczyk, K.-D. Ufert, and G.

Müller, "Conductive bridging RAM (CBRAM): An emerging non-volatile memory technology scalable to sub 20 nm, " in IEDM *Tech. Dig.* , pp. 754-757, 2005.

[13] F. Alzahrani, and T. Chen, "On-chip TEC-QED ECC for ultra-large, single-chip memory systems, " in Proc. IEEE Int. Conf. Comput Design Design , Very-Large-Scale Integr. (VLSI) Syst. Comput. Process. , pp. 132-137, Oct. 1994.

BIOGRAPHY



V. NagaJyothirmai received B. Tech degree in Electronics and Communication Engineering from DMS SVH College of Engineering, Machilipatnam, AP, India. She is pursuing M. Tech (Digital Electronics and Communication Systems) in Gudlavalleru Engineering College, Gudlavalleru.

A. V. N. Tilak has obtained his B. E. , M. Tech. and Ph. D. from MIT Manipal, IIT Kanpur, and IIT Madras respectively. His areas of interest are Microelectronics, Digital Design, and Low Power VLSI Design. Dr. Tilak is a member of IEEE, Fellow IETE, Fellow IE (I), and Life member of ISTE.