ISSN: 2302-9285, DOI: 10.11591/eei.v9i5.1876

# Lightweight hamming product code based multiple bit error correction coding scheme using shared resources for on chip interconnects

Asaad Kadhum Chlaab<sup>1</sup>, Wameedh Nazar Flayyih<sup>2</sup>, Fakhrul Zaman Rokhani<sup>3</sup>

<sup>1,2</sup>Department of Computer Engineering, University of Baghdad, Iraq <sup>3</sup>Department of Computer and Communication Systems Engineering, Universiti Putra Malaysia, Malaysia

#### **Article Info**

# Article history:

Received Oct 27, 2019 Revised Feb 3, 2020 Accepted Mar 15, 2020

# Keywords:

Extended hamming product code
Multi bit error
On chip interconnect
Residual flit error rate

#### **ABSTRACT**

In this paper, we present multiple bit error correction coding scheme based on extended Hamming product code combined with type II HARQ using shared resources for on chip interconnect. The shared resources reduce the hardware complexity of the encoder and decoder compared to the existing three stages iterative decoding method for on chip interconnects. The proposed method of decoding achieves 20% and 28% reduction in area and power consumption respectively, with only small increase in decoder delay compared to the existing three stage iterative decoding scheme for multiple bit error correction. The proposed code also achieves excellent improvement in residual flit error rate and up to 58% of total power consumption compared to the other error control schemes. The low complexity and excellent residual flit error rate make the proposed code suitable for on chip interconnection links.

This is an open access article under the CC BY-SA license.



1979

# Corresponding Author:

Wameedh Nazar Flayyih, Department of Computer Engineering, University of Baghdad, P.O. Box 10071, Jadiriya, Baghdad, Iraq. Email: wam.nazar@coeng.uobaghdad.edu.iq

# 1. INTRODUCTION

Interconnection of all processing elements (PEs) or intellectual property (IP) cores on a single chip by employing traditional on-chip communication infrastructure, like shared buses or multi-layer buses, results in issues with scalability and IP reusability. This motivates system on chip architects to shift to network on chip (NoC). NoC provides scalable and high bandwidth communication infrastructure to various multi-core and many-core architectures [1-3]. In very deep submicron technology (VDSM), on chip interconnect errors are caused by different effects like supply voltage fluctuation, electromagnetic interference (EMI), process variation and crosstalk [4-9]. Reliability can be improved by applying error control techniques, such as automatic repeat request (ARQ), forward error correction (FEC), and hybrid ARQ (HARQ) to on-chip interconnects [10-14]. These three error correction techniques have different error correction capability and different hardware complexity. In [12,13], the techniques were able to correct single bit or two-bit random errors only. But, the probabilities of occurrence of multiple random and burst errors are getting higher [14], which urged the need for more powerful coding techniques. The combintion of crosstalk avoidance code with error control code can improve the error correction capability [15]. In [16], the use of simple parity calculation along with message triplication can achieve two random error correction capability was

increased by using Hamming code with message duplication to correct three errors. Further optimization was applied for this scheme for triple error correction and quadruple error detection (JTEC-SQED) [17]. In joint crosstalk aware multiple error correction (JMEC) the use of changed interleaving distance between adjacent bits made the correction of nine adjacent errors possible [18]. Duplication with two dimensional parities was also proposed to provide up to seven errors detection [19] or six errors detection and single error correction [20]. In multi bit random and burst error correction (MBRBEC) five errors correction was possible by using extended Hamming code with message triplication [21]. Quintuplicated manchester error correction (QMEC) achieved nonuple errors correction [22]. All the duplication, triplication and quintuplication based coding schemes allowed high error correction due to the high redundancy which is translated into high link size.

Hamming product codes [23] are used to correct both multiple random and burst errors without high link size overhead. In [24-27] the authors use three-stage decoding circuit for Hamming product codes with type II HARQ to achieve a good correction capability (up to five random errors and burst errors). The only drawback of this approach is the use of complex design of three stage circuit. In [28], the authors designed two stage row-column decoding and keyboard scan-based error flipping instead of the three-stage decoding design proposed in [24-26]. However, the reduction in decoder area was at the cost of correction capability because it can correct less random bits and four burst error only. In [29], the authors used the same principle of hamming product code but with different arrangement. They used extended hamming on rows and simple parities on columns so that they can reduce the circuit complexity and parity bit size. The smaller message size allowed them to send the message at once, without the use of ARQ technique, but at the same time there is a huge drop in correction capability compared with work in [24-26] as will be reanalyzed later in this paper. The proposed design in this paper follows the same target authors followed in [28, 29] towards the reduction of circuit compexity in [24-26]. The proposed design differs by not sacrificing the correction capability to gain the (area/power) savings. The proposed coding makes use of the concept of resource sharing and used it on the traditional three stage hamming product code decoding. The proposed circuit has the same correction capability of three stage hamming method with less area and power consumption.

## 2. EXTENDED HAMMING PRODUCT CODE WITH TYPE II HARO

The input flit (k) is arranged into a matrix  $(k_1 \times k_2)$ , as shown in Figure 1. Row parity check bits are obtained by encoding the  $(k_1)$  bits in each row using  $(n_1, k_1)$  extended Hamming row encoder, where  $n_1$  is the row encoded word. Column parity check bits are obtained by encoding the  $(k_2)$  column bits using  $(n_2, k_2)$  extended Hamming column encoder, where  $n_2$  is the column encoded word. Checks-on-checks can be generated by encoding the column parity check bits using row encoder. In [24], the authors used extended Hamming Product with type II HARQ Code to reduce the number of interconnection links.

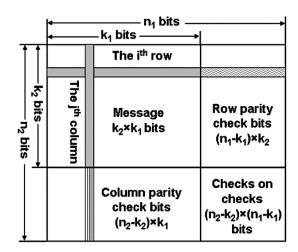


Figure 1. 2-D product codes [24]

#### 2.1. The design of encoder

The encoding process of Hamming product codes with type-II HARQ in [24-26] is shown in Figure 2(a). K-bit input message is encoded using row and column encoders. Extended Hamming codes EH  $(n_1, k_1)$  are used for row encoding and EH  $(n_2, k_2)$  are used for column encoding. All the output of row

encoders will pass through the row-column interleaver. Then the interleaved output will be sent through the link to decoders. The column encoders outputs are saved in a buffer, if there is a NACK received the saved parities will pass through second row encoders to generate checks-on-checks. The checks-on-checks and column parity check bits will be fed into row-column interleaver and sent to the decoder.

ISSN: 2302-9285

The proposed design shown in Figure 2(b). exploits the similarities between the first circuit (row encoders) and the second circuit (column encoders) and combines them into one circuit (general encoder). In addition, the new proposed general encoder is responsible for the checks on check calculation, eliminating the need for the third circuit (second row encoders). The proposed design makes use of study done in [24-26] work that shows the best arrangement for hamming product code to get minimum link size is to have always four row encoders whatever is the message size. So, the column encoders are always in the size of four bits extended hamming circuit. If we have  $(k_1)$  column encoders of size four input, we can combine (M) of them to form one row encoder. The number of column encoders M to form one row encoder can be written as:

$$M = (K/K_2)/4 \tag{1}$$

Figure 3 shows the internal design of the general encoder where each (M) column encoders output will be input to one Column-to-Row circuit. This circuit takes  $(n_2-k_2)$  parity bits from (M) column encoder and Xor them to form one set of  $(n_1-k_1)$  parity bits. The proposed encoder first encodes k input data using row encoder and saves a copy of the original data in a buffer. When the NACK is received the saved data will be encoded using column encoder and sent back to calculate Check on Checks using row encoders. As mentioned earlier, Check on Checks calculation circuit works same as row encoders in [24-26] so the third circuit was replaced using only the general encoder to work as row encoder. The buffer size in the proposed encoder is the same as in the previous work of [24-26]. As was mentioned earlier, the authors used  $k_2$ =4 and the resultant check bits of extended Hamming code for four bits data EH (8, 4) is also four bits. This makes  $((n_2 - k_2) \times k_1)$  bits saved in the buffer in their work is the same size to k bit saved in our encoder. It should be noticed that in the proposed design two clock cycles are consumed in the encoder to calculate the column checks and checks on checks after NACK. This is one clock cycle more than that in [24-26]; a small penalty to pay for the gained power savings since the proposed encoder performs column encoding only after NACK is received.

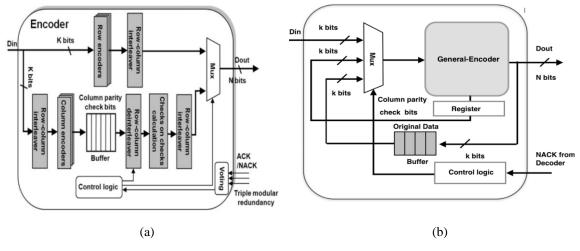


Figure 2. (a) (fu2009) encoder [24], (b) Proposed encoder

# 2.2. The design of decoder

The decoding process of Hamming product codes with type-II HARQ in [24-26] is shown in Figure 4(a). The encoded data is applied to the extended Hamming row decoder for decoding. The row decoder corrects any single error that occurs in each row. If the errors are detectable but not correctable NACK signal is sent back to the encoder after storing the row decoded data and row parity bits in a buffer. At the same time, the row condition vector (RCV) is formed, which contains information about each row. When there is no error or one error in row the RCV will indicate zero, otherwise if there are double errors the RCV will indicate one in the corresponding RCV bit. When the column parity check bits and checks on checks are received, they are first passed through the row decoder to correct any error. The previous row decoded data stored in the buffer is row-column interleaved. The column parity check bits and row column interleaved data

are passed to the second stage column decoding. The column decoder corrects any single error that occurs in each column. The column condition vector (CCV) is formed. Then the output is sent to the third stage row decoders to form a new RCV after the column decoder correction is done in the second stage. The third stage decoders contain a simple flipping circuit used in [24-26] to correct rectangular errors using CCV and newly formed RCV. In the proposed decoder shown in Figure 4(b), the third stage row decoders are removed and the row-decoder in first stage is shared to perform the row decoding in the first and third stages. The flipping circuit is separated to be used only in the third stage of decoding. The proposed work works same as the previous circuit in [24-26] but instead of using the third stage we used the feedback to do third row decoding and flipping circuit to correct rectangular errors.

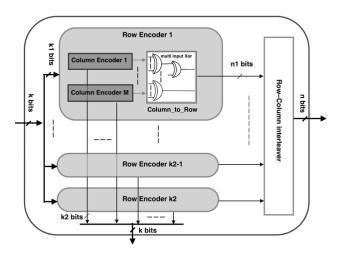


Figure 3. General encoder internal design

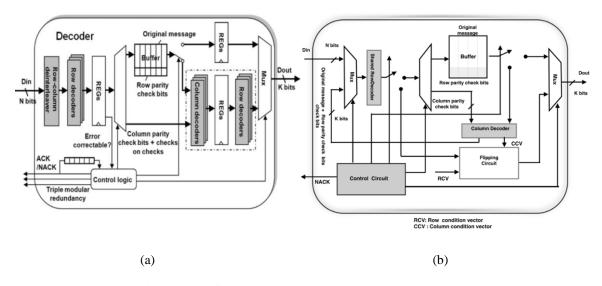


Figure 4. (a) (fu2009) decoder [24], (b) Proposed decoder

## 3. RESULTS AND ANALYSIS

In this section, reliability analysis for Hamming product code with Type II HARQ and MECCRLB code is present, then a comparison between them is carried out in terms of error correction capability, link swing voltage, link power consumption, codec area, codec power and codec delay.

# 3.1. Reliability analysis

The reliability is measured by the residual error probability  $P_{residual}$  which represents the probability of decoder error or failure [30-32]. So,  $P_{residual}$  can be express as:

$$P_{residual} = 1 - P_{pd} \tag{2}$$

where P<sub>pd</sub> is the probability of proper decoding which is the sum of probabilities of correcting random errors and burst errors. Random errors will be considered only since they are the most frequent and for the purpose of simplicity.

## 3.1.1. Extended hamming product code with type II HARQ

 $P_{residual}$  depends on both the error detection capability in the first transmission and error correction capability after the retransmission.  $P_{residual}$  is estimated as given in [24] as:

$$P_{residual} = P_{ud} + P(e_{decoding}, e_{detect})$$
(3)

where  $P_{ud}$  is the undetectable error probability in the first transmission and  $P(e_{decoding}, e_{detect})$  is the probability of error after retransmission and three stage decoding is over.  $P_{ud}$  can be expressed as:

$$P_{ud} = 1 - (P_{ne} + P_c + P_d) (4)$$

where  $P_{ne}$  is the probability of no error and  $P_c$ ,  $P_d$  are the probability of correctable error patterns, and the probability of detectable but uncorrectable error patterns in the first retransmission, respectively.  $P(e_{decoding}, e_{detect})$  can be expressed as:

$$P(e_{decoding}, e_{detect}) = P_d \cdot (1 - P_{ne} - P_c)$$
(5)

By inserting (4) and (5) in (3) we get:

$$P_{residual} = 1 - (1 + P_d).(P_{ne} + P_c) \tag{6}$$

Since any error pattern with single error in one row and the others in different row can be corrected in the first transmission,  $P_C$  for random errors can be written as:

$$P_c = \sum_{t=1}^{n_1} \binom{k_2}{t} n_1^t \varepsilon^t \tag{7}$$

where  $k_2$  is the number of rows,  $n_1$  is the row bit size, and  $\varepsilon$  is the bit error rate. After retransmission, the proposed work can correct five random errors so  $P_d$  for random errors is defined in (8) as given in [24]. The first term is the error detection probability when two or three random errors occur in the first transmission. The second and third terms in (8) are the error detection probability of four and five random errors.

$$P_{d} = \sum_{t=2}^{3} \binom{k_{2}}{1} \binom{n_{1}}{2} \binom{(k_{2}-1)n_{1}}{t-2} e^{t} + \binom{k_{2}}{1} \binom{n_{1}}{2} \binom{(k_{2}-1)n_{1}}{2} e^{4} + \binom{k_{2}}{1} \binom{n_{1}}{2} \binom{(k_{2}-1)n_{1}}{2} e^{5}$$

$$(8)$$

The probability of no error in first transmission can be expressed as in [24]

$$P_{ne} = (1 - \mathcal{E})^{k2n1} \tag{9}$$

If we substitute (7-9) in (6) we can easily find  $P_{residual}$ .

#### 3.1.2. MECCRLB

The work in [29], is an FEC-based coding scheme, where there is no retransmission available. As a result, P<sub>residual</sub> depends on the coding correction capability.

$$P_{residual} = 1 - P_c \tag{10}$$

The authors in [26] indicate that MECCRLB can correct up to 11 random errors in 32-bit message so P<sub>c</sub> for random errors was expressed in [26] as:

$$P_c = \binom{k_2 \times n_1 + 10}{12} \varepsilon^{12} \tag{11}$$

However, that equation is not accurate as it is not possible to correct all 11 random errors applying the MECCRLB decoding [8]. Figure 5 shows some cases where MECCRLB fails to correct two, three, and four errors. Instead, (12) can express the correction capability for up to four random errors, where the first term represents single error correction in different rows while the second term expresses double errors correction in message except if one of errors happens in 10-bits parity checks. Term three expresses the correction of three errors in message except if one or two of errors happen in 10-bits parity checks. Term four expresses the correction of four errors in three cases; first case when all four errors are in one row, the second case when three errors are in one row and the other in any other message bits, and the third case when two errors are in one row and the other two errors elsewhere in the message. If we substitute (12) in (10) we can find  $P_{residual}$  for MECCRLB for random errors.

$$P_{c} = \sum_{t=1}^{k_{2}} \binom{k_{2}}{t} n_{1}^{t} \varepsilon^{t} + \binom{k_{2} \times n_{1} + 10}{2} - \binom{k_{2}}{1} \binom{n_{1}}{1} 10$$

$$+ \binom{k_{2} \times n_{1} + 10}{3} - \binom{k_{2}}{1} \binom{n_{1}}{2} 10 - \binom{k_{2}}{1} \binom{n_{1}}{1} 10) \varepsilon^{3} + \binom{k_{2}}{1} \binom{n_{1}}{4} + \binom{k_{2}}{1} \binom{n_{1}}{3} \binom{k_{2} \times n_{1}}{3} + \binom{k_{2}}{1} \binom{n_{1}}{2} \binom{(k_{2} - 1)}{2}$$

$$(12)$$

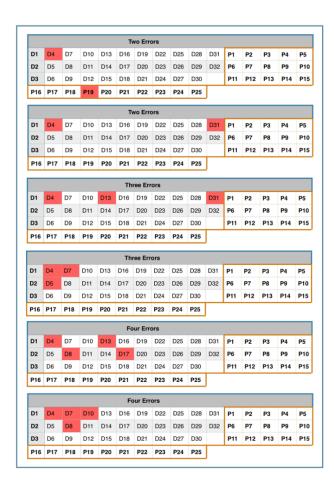


Figure 5. Examples of failure cases for work in [29] in case of two, three and four errors

ISSN: 2302-9285

Now, after getting the equation for  $P_{residual}$  for both Extended Hamming Product Code with type II HARQ and MECCRLB and for 32-bits message size, the equations become a function of  $\varepsilon$ . Figure 6 shows  $P_{residual}$  in estimation and simulation for different  $\varepsilon$  values. A C++ program was developed to simulate the two techniques and random errors are generated at different error rates as shown in Figure 6. The results show that the estimated residual flit error rate is close to the simulated results. It can also be noticed by looking to the Y-axis that for the same values of  $P_{residual}$  HPC can sustain a higher bit error rate. However, in [33] more detailed comparsion was made between the two schemes which shows the superiority of HPC in terms of reliability.

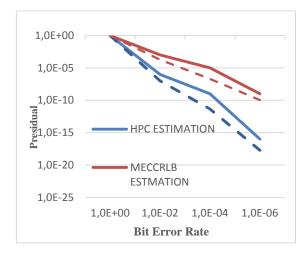


Figure 6.  $P_{residual}$  for different bit error rates

## 3.2. Link swing voltage

On-chip communication errors can be attributed to voltage perturbations induced by noise from many sources. The error probability of a single wire can be modeled by a Gaussian pulse function [34].

$$\varepsilon = Q\left(\frac{V_{swing}}{2\sigma_N}\right) = \int_{\frac{V_{swing}}{2\sigma_N}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy \tag{13}$$

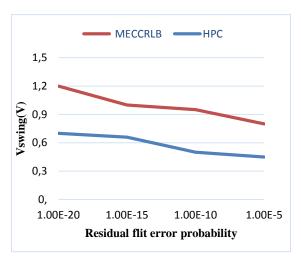
where  $V_{swing}$  is the link swing voltage and  $\sigma$  is the standard deviation of the noise voltage, which is assumed to be a normal distribution. Therefore, adoption of highly reliable error correcting coding technique in NoC [11], results in reduction of link swing voltage from (7):

$$V_{swing} = 2\sigma_N Q^{-1}(\varepsilon') \tag{14}$$

where  $Q^{-1(\epsilon')}$  is the inverse Gaussian function and  $\dot{\epsilon}$  is the value at which  $P_{residual}(\epsilon')$  is equal to the probability of maximum permissible residual error. Figure 7 compares the link swing voltage for different error control schemes where  $\sigma_N$  is assumed as 0.1V. The Hamming product code achieves lower swing voltage compared to MECCRLB. The link power consumption  $Pw_L$  is related to the interconnect capacitance  $C_L$ , the wire switching factor  $\alpha$ , the link width  $W_L$ , the link swing voltage  $V_{swing}$  and clock frequency  $f_{clk}$ . The link power  $P_{WL}$  can be expressed as:

$$Pw_L = C_L \alpha W_L V_{swing}^2 f_{clk} \tag{15}$$

where  $\alpha$  is assumed as 0.1 and  $W_L$  depends on the error control schemes. The link swing voltage  $V_{swing}$  depends on the reliability requirement of different error control schemes according to (14). For the given reliability requirement, the error control codes with low error correction capability need a higher link swing voltage than the error control scheme with high error correction capability. Figure 8 shows the link power consumption for different error control schemes for two given reliability requirements, namely  $P_{residual}$  of  $10^{-20}$  and  $10^{-5}$ . The power consumption is estimated for 45nm technology. The wire capacitance  $C_L$  is assumed as 208 fF/mm and the clock frequency is 500 MHz. Because of the higher detection capability of the Hamming product coding scheme, it uses low swing voltage that results in low link power consumption as compared to MECCRLB code.



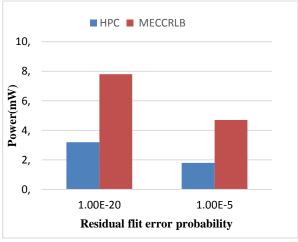


Figure 7. Link swing voltage

Figure 8. Link power consumption of different error control schemes

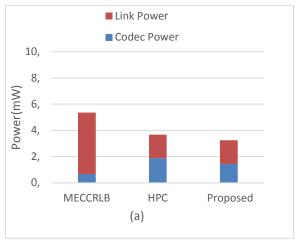
## 3.3. Area, power and delay

The area and delay of the proposed error control scheme and other error control schemes are shown in Table 1. All three were developed in Verilog HDL and functionally verified in ModelSim. Then they were synthesized in SDC using Nandgate 45nm library. Table 1 shows that the proposed implementation of Hamming product code (52, 32) consumes lower area by 20% compared to the HPC in [24] since the proposed scheme combines three circuits (row, column, row) encoders in one general encoder at the encoder side and also uses one shared row decoder instead of two row decoders at the decoder side. The proposed work has a higher area than MECCRLB [29], which represents the cost for its more powerful code in terms of error correction. The table also shows the encoder and decoder critical path delays for the three schemes, and as expected, MECCRLB achieves the lowest delay due to its lower complexity circuits (encoder, decoder). The proposed coding scheme introduces a slight increase in the decoder delay as compared to HPC in [24] due to its additional multiplexing circuit with the shared row decoder. In pipelined operation, where the encoder and decoder are separate pipeline stages, the maximum frequency is limited by the slowest stage which is the decoder for all the schemes.

Accordingly, MECCRLB, Hamming product code, and the proposed coding achieve a maximum frequency of 1.1, 1, and 0.9 GHz respectively. Figure 9(a) and (b) shows the link and codec power at 500 MHz frequency, with two values for  $P_{residual}$  10<sup>-5</sup> and 10<sup>-20</sup>. From Figure 9(a) it can be noticed that the MECCRLB consumes less codec power, but higher link power compared to other two works. That is because the authors used simple coding circuits that consume less codec power at the cost of lower correction capability that makes the voltage swing higher. The higher voltage swing is translated to higher link power consumption as the power consumption of the link is directly proportional to the square of the link swing voltage. Both, the proposed coding scheme and that in [24], have the same link power since they use the same correction technique. But the proposed work reduces the codec power by 28% by its optimized encoding and decoding circuits. Figure 9(b) shows (link and codec power) with  $P_{residual}$ =10<sup>-20</sup>. The codec power is not affected and it is the same as in (a), but there is an increase in link power for the three schemes since the increase in the target reliability (lower  $P_{residual}$ ) will increase the voltage swing accordingly, which in turn increases the link power. It should be noted that even in lower  $P_{residual}$  the hamming product code used in [24] and in our work results in lower link power consumption as compared to MECCRLB due to higher correction capability.

Table 1. Implementation results

Error correction code	Area	Delay(ns)	
	$(\mu m^2)$	Encoder	Decoder
MECCRLB	744	0.5	0.9
Hamming product code with Type II HARQ	3574	0.8	1.0
Proposed	2850	0.8	1.1



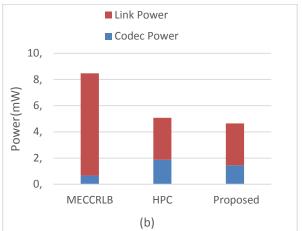


Figure 9. (a) Power at  $P_{res}=10^{-5}$ , (b) Power at  $P_{res}=10^{-20}$ 

#### 4. CONCLUSION

This paper presented a lightweight realization of the Hamming product code with type II HARQ which is capable of correcting 100% of error patterns that have five errors (in full message transmission). The proposed code can also correct burst errors of up to 16 bits or a combination of random and burst errors. The resource sharing technique used reduced the area by 20% and the power by 28% with only a slight increase in the decoder delay. Because of the high error correction capability of the proposed error control code, it achieved low swing voltage, which resulted in low link power consumption. The low swing voltage resulted in the reduction of the total power consumption by up to 58% compared to other error control codes.

## CONFLICT OF INTEREST

The authors declare that there is no conflict of interest.

# ACKNOWLEDGMENTS

The authors would like to acknowledge the partial funding and support provided by University Putra Malaysia grant, Crest, and MIMOS.

#### REFERENCES

- [1] N. Jafarzadeh, M. Palesi, S. Eskandari, S. Hessabi and A. Afzali-Kusha, "Low energy yet reliable data communication scheme for network-on-chip," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 12, pp. 1892-1904, Dec. 2015.
- [2] F. Caignet, S. Delmas-Bendhia and E. Sicard, "The challenge of signal integrity in deep-submicrometer CMOS technology," in *Proceedings of the IEEE*, vol. 89, no. 4, pp. 556-573, April 2001.
- [3] C. Duan, V. H. Cordero Calle and S. P. Khatri, "Efficient on-chip crosstalk avoidance CODEC design," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 17, no. 4, pp. 551-560, April 2009.
- [4] C. Constantinescu, "Trends and challenges in VLSI circuit reliability," in *IEEE Micro*, vol. 23, no. 4, pp. 14-19, July-Aug. 2003.
- [5] M. Lajolo, M. S. Reorda and M. Violante, "Early evaluation of bus interconnects dependability for system-on-chip designs," VLSI Design 2001. Fourteenth International Conference on VLSI Design, Bangalore, India, pp. 371-376, 2001.
- [6] D. Sylvester and Chenming Wu, "Analytical modeling and characterization of deep-submicrometer interconnect," in *Proceedings of the IEEE*, vol. 89, no. 5, pp. 634-664, May 2001.
- [7] A. V. Mezhiba and E. G. Friedman, "Scaling trends of on-chip power distribution noise," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 4, pp. 386-394, April 2004.
- [8] Y. S. Jeong, S. M. Lee, S. E. Lee, "A Survey of fault-injection methodologies for soft error rate modeling in systems-on-chips," *Bulletin of Electrical Engineering and Informatics (BEEI)*, vol. 5, no. 2, pp. 169-17, 2016.
- [9] G. P. Acharya, M. A. Rani, "Berger code based concurrent online self-testing of embedded processors," *Journal of Semiconductors*, vol. 39, no. 11, 2018.
- [10] L. Benini and G. De Micheli, "Networks on Chips: Technology and Tools," San Francisco, CA, USA: Morgan Kaufmann, 2006.

[11] S. R. Sridhara and N. R. Shanbhag, "Coding for system-on-chip networks: a unified framework," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 6, pp. 655-667, June 2005.

- [12] S. Murali, T. Theocharides, N. Vijaykrishnan, M. J. Irwin, L. Benini and G. De Micheli, "Analysis of error recovery schemes for networks on chips," in *IEEE Design & Test of Computers*, vol. 22, no. 5, pp. 434-442, 2005.
- [13] D. Bertozzi, L. Benini and G. De Micheli, "Error control schemes for on-chip communication links: the energy-reliability tradeoff," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 6, pp. 818-831, June 2005.
- [14] A. Ejlali, B. M. Al-Hashimi, P. Rosinger and S. G. Miremadi, "Joint Consideration of Fault-Tolerance, Energy-Efficiency and Performance in On-Chip Networks," 2007 Design, Automation & Test in Europe Conference & Exhibition, Nice, pp. 1-6, 2007.
- [15] L.Zhou, N.Wu and F.Ge, "A joint-coding scheme with crosstalk avoidance in network on chip," TELKOMNIKA Telecommunication, Computing, Electronics and Control, vol. 11, no. 1, pp 1-8, Jan. 2013.
- [16] A. K. Kummary, P. Dananjayan, K. Viswanath and V. Reddy "Combined crosstalk avoidance code with error control code for detection and correction of random and burst errors," Coding Theory, Sudhakar Radhakrishnan and Muhammad Sarfraz, IntechOpen, Jun. 2019.
- [17] A. Ganguly, P. P. Pande and B. Belzer, "Crosstalk-aware channel coding schemes for energy efficient and reliable NOC interconnects," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 17, no. 11, pp. 1626-1639, Nov. 2009.
- [18] M. Gul, M. Chouikha and M. Wade, "Joint crosstalk aware burst error fault tolerance mechanism for reliable onchip communication," in *IEEE Transactions on Emerging Topics in Computing* 2017.
- [19] W. N. Flayyih, K. Samsudin, S. J. Hashim, F. Z. Rokhani and Y. I. Ismail, "Crosstalk-aware multiple error detection scheme based on two-dimensional parities for energy efficient network on chip," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 7, pp. 2034-2047, July 2014.
- [20] W. N. Flayyih, K. Samsudin, S. J. Hashim, Y. Ismail and F. Z. Rokhani, "Multi-bit error control coding with limited correction for high-performance and energy-efficient network on chip," *IET Circuits, Devices & Systems*, vol. 14, no. 1, pp. 7-16, 1 2020.
- [21] M. Maheswari and G. Seetharaman, "Multi bit random and burst error correction code with crosstalk avoidance for reliable on chip interconnection links," *Microprocessors and Microsystems*, vol. 37, no. 4-5, pp. 420-429, 2013.
  [22] P. Narayanasmy and S. Muthurathinam, "Design of Crosstalk Prevention Coding scheme based on Quintuplicated
- [22] P. Narayanasmy and S. Muthurathinam, "Design of Crosstalk Prevention Coding scheme based on Quintuplicated Manchester error correction method for Reliable on chip Interconnects," *Advances in Electrical and Computer Engineering*, vol. 18, no. 4, pp 113-130, 2018.
- [23] Bo Fu and P. Ampadu, "A multi-wire error correction scheme for reliable and energy efficient SOC links using hamming product codes," 2008 IEEE International SOC Conference, Newport Beach, CA, pp. 59-62, 2008.
- [24] B. Fu and P. Ampadu, "On hamming product codes with type-II hybrid ARQ for on-chip interconnects," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 56, no. 9, pp. 2042-2054, Sept. 2009.
- [25] B. Fu and P. Ampadu, "Error control for network on chip links," Springer, New York, 2012.
- [26] B. Fu and P. Ampadu, "An energy-efficient multi wire error control scheme for reliable on chip interconnects using hamming product codes," *VLSI Design*, pp.1-14, 2008.
- [27] B. Fu and P. Ampadu, "Burst Error Detection Hybrid ARQ with Crosstalk-Delay Reduction for Reliable On-chip Interconnects," 2009 24th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, Chicago, IL, pp. 440-448, 2009.
- [28] M. Maheswari, G. Seetharaman, "Hamming product code based multiple bit error correction coding scheme using keyboard scan based decoding for on chip interconnects links," *Applied Mechanics and Materials*, vol. 241-244, pp. 2457-2461, 2013.
- [29] M. Vinodhini, N. S. Murty, "Reliable low power NoC interconnect," *Microprocessors and Microsystems*, vol 57, pp.15-22, March 2018.
- [30] Q. Yu and P. Ampadu, "Dual-Layer Adaptive Error Control for Network-on-Chip Links," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, no. 7, pp. 1304-1317, July 2012.
- [31] M. Maheswari, G. Seetharaman, "Design of a novel error correction coding with crosstalk avoidance for reliable on-chip interconnection link," *International Journal of Computer Applications in Technology*, vol. 49, no. 1, pp. 80-88, 2014.
- [32] M. Maheswari, G. Seetharaman, "Enhanced low complex double error correction coding with crosstalk avoidance for reliable on-chip interconnection link," *Journal of Electronic Testing*, vol. 30, no. 4, pp. 387–400, 2014.
- [33] A. C. Kadum, W. N. Flayyih, F. Z. Rokhani, "Reliability analysis of multibit error correcting coding and comparison to hamming product code for on-chip interconnect," *Journal of Engineering*, University of Baghdad, vol, 26, no. 6, pp. 94-106, 2020.
- [34] R. Hegde and N. R. Shanbhag, "Toward achieving energy efficiency in presence of deep submicron noise," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 8, no. 4, pp. 379-391, Aug. 2000.

#### **BIOGRAPHIES OF AUTHORS**



**Asaad Kadhum Chlaab** recieved his Bachelor of computer of engineering from Universisity of Baghdad in 2013. He has spent four years working in various computer engineering fields, Currently, he is working towards his master degree in computer engineering. His research interests include Fault Tolerant in NoC.

ISSN: 2302-9285



Wameedh Nazar Flayyih received the B.Sc. and M.Sc. degrees in Computer Engineering from University of Baghdad, Iraq, in 2001 and 2004, respectively. He received the Ph.D. in Computer Systems Engineering from University Putra Malaysia, Serdang, Malaysia in 2014. He is currently a lecturer at the Department of Computer Engineering, University of Baghdad. His research interests include computer architecture, VLSI design, Network on Chip, and fault tolerance.



**Fakhrul Zaman B. Rokhani** received the B.S. degree in electrical-mechatronics engineering from the University Technology Malaysia, Johor Bahru, Malaysia, in 2002, and the M.S. and Ph.D. degrees in electrical engineering from the University of Minnesota, Minneapolis, MN, USA, in 2004 and 2009, respectively. He is currently a Lecturer at the Department of Computer and Communication Systems Engineering, University Putra Malaysia, Serdang, Malaysia. His current research interests include intelligent computer and embedded systems design, nanoelectronics VLSI design, and fault-tolerant system/network-on-chip design.