ISBN 978-93-86878-07-6
16th IIE International Conference on Computer, Control and Communication Engineering (IC4E'2017)
Dubai (UAE) Dec. 21-22, 2017

Multi-bit Error Correction Technique for NTC based Network on Chip

*Naveed Khan Baloch[1], Waqar Ahmed[2], Ayaz Hussain[3], Waqar Ahmad[4], M. Iram Baig[5]

[1,2,3,4,5] University of Engineering and Technology, Taxila, Pakistan

***Abstract*:** *To fulfill the communication requirements of System on Chip(SoC) the latest promising solution is Network on Chip(NoC). It is packet switching based technology used in multicore SoC. Unreliability is one of the significant challenge faced by NoC architectures due to technology limitations. Near-Threshold Computing can also be introduced in NoC, but it comes with transient faults. Through error control techniques NoC can achieve reliability as well as energy efficiency. In this paper duplicate consecutive sextuple error correction technique is proposed to handle multiple errors, and its results are assessed in NoC. It is proved that the proposed error correction scheme is better in error correction capability than previous error correction schemes to make NoC architectures reliable and energy efficient, with lower latency.*

Keywords: *Logic Voltage Induced (LVI), Near-Threshold Computing (NTC), Network on Chip (NoC), Single Event Upset (SEU), Timing Voltage Induced (TVI), Ultra-Deep Submicron (UDSM).*

1. Introduction

On chip communication energy consumption and reliability recently became a foremost apprehension due to the shrinking sizes, the upsurge in frequencies, and reduction in chips' operational voltages.

NoC is evolved as the latest technology to accomplish this efficient and reliable communication on-chip [1], [2].

Modern systems on the chip have energy constraints and limiting power budgets. Near-threshold computing (NTC) has the proficient energy advantage as it enables the system to operate on a relatively lower supply voltage which is close to the transistor threshold voltage.

However, due to process variation, this power efficiency costs performance loss as well as performance inconsistency [3].

Furthermore, NTC substantially increases the power efficiency of traditional architectures which were designed for standard voltage operation [6].

As the sizable portion of current literature aims to enhance the performance of the computing cores, therefore NTC impacts on NoC have taken a back seat. In this background, our proposed work is intended to preserve signal integrity which may be affected due to NTC implementation in NoC.

It is a well-known fact that with an increased number of transistors on a chip as per Moore's Law, with this scalability NoC architectures endeavor to achieve energy efficiency, better performance, and reliability. As supply voltage has a positive impact on both static and dynamic energy deception, therefore by reducing the supply voltage, we can achieve power efficiency. Near Threshold Voltage (NTV) involves scaling down the supply voltage very close to the threshold voltage of the transistor. Visibly it has the potential to deliver almost 10% improvement in energy efficiency [4]. Noise sources, which include alpha particles, power supply noise, electromagnetic interference, and others can be a source of transient faults in NoC [5]. Transient errors in the

form of logical SEU and crosstalk are the primary reason for system failures in NTC based architectures, and they constitute a significant percentage of system failures [6]. This is because the supply voltage is near to transistors threshold voltage and operating frequency is lower. Therefore process voltage temperature (PVT) variation critically affects NTC based architectures. Therefore, SEU and crosstalk constitute a significant cause of failure in NTC based NoC, and it needs to be addressed to achieve energy efficiency, performance, and reliability.

Proposed work combine error control including crosstalk prevention to address both Logic Voltage Induced (LVI) and Timing Voltage Induced (TVI) based SEU faults. Earliest schemes, Hamming code [7] duplicate add parity (DAP) [8], dual rail (DR) [8], boundary shift code (BSC) [8], and modified dual rail code (MDR) [9], accomplished single error correction (SEC) [8]. These coding schemes achieve an increase in speed of signal arrival, resilience against SEU and reduction in energy consumption.

Recently, more useful error correction techniques to detect and correct multi-bit errors have been proposed for addressing noise in NoC. Hamming product code using skewed changeovers [10] achieved multi-bit error detection, but it was restricted to two errors. Another crosstalk avoidance and dual error correction technique were proposed in [11] named as CADEC. It encodes the data through Hamming SEC and passing the resultant code word into the DAP encoder. This technique was later upgraded to the joint crosstalk avoidance and triple error correction (JTEC) scheme and the JTEC with Simultaneous quadruple error detection (JTEC-SQED) but same three-bit correction scheme [12]. This three-bit error detection/correction allowed a lesser voltage swing, which resulted in efficiently reducing power consumption and increasing area overhead while maintaining the target reliability.

Another problem regarding encoding in NOCs is emphasized in [11]. According to this work, forward error correction was found less energy efficient than if the data is retransmitted after error recognition. In UDSM the problems are much more severe that arise due to transient noise. In the concluding remarks of [8], it is mentioned that UDSM domain communication on-chip energy will overcome computation energy. Retransmission will increase the number of communications on the same link and therefore eventually it is not going to be very energy efficient.

Due to the supply voltage and temperature variations, actual noise may vary while all these error correction techniques are aimed to accomplish target reliability in worst case situation [1], [13]. Therefore, there architecture is complex, energy consuming and has area overhead.

In this brief, network on chip (NoC) is considered as an evolving archetype for on-chip communication. We propose a multi-bit error control code provides the necessary strength to counter SEU up to three consecutive bits furthermore it also reduces power consumption and has less area overhead. Section II refers to the proposed multi-bit error correction scheme. Section III refers to the implementation of Duplicate Consecutive Sextuple Error Correction D_CSEC in NTC based NoC. Section IV describes the schemes from voltage swing and word error probability perspectives. Section V describes average latency results.

2. Proposed Multi-Bit Error Correction Technique

The suggested technique used the inherent features of the error control technique to reduce hardware complexity and achieve reliability in providing multi-bit error correction/detection. This method is mainly based on a technique proposed in [12] but additionally, explores address crosstalk by combining original technique with DAP.

Suppose a code with c check bits and k data bits. 2k binary values are possible with this code. The code word has k+c bit places that may have a single error, k+c-1-bit places for double contiguous bit errors, k+c-2-bit positions for triple adjacent bit errors and k+c-2 bit positions for double almost contiguous bit errors respectively. Together with all possibilities of correct values, there are (k+c) +(k+c-1)+(k+c-2)+(k+c-2)+1 possibilities of 2k binary values that should be denoted in the code word. This infers

$$2^k((k+c)+(k+c-1)+(k+c-2)+(k+c-2)+1) \leq 2^{k+c} \quad (1)$$

Excluding 2^k term which is typical from both sides of (1), the result is following equation

$$k+c-1 \leq 2^{c-2} \quad (2)$$

where c, which is some check bits should be curtailed to decrease the length of the code.

2.1. Multi-bit Error Correction Encoder

$$\begin{aligned}
&d0 \oplus d3 \oplus d4 \oplus d5 \oplus d8 \oplus d12 \oplus d13 = c0 \\
&d1 \oplus d4 \oplus d7 \oplus d8 \oplus d11 \oplus d13 \oplus d14 = c1 \\
&d2 \oplus d5 \oplus d6 \oplus d9 \oplus d10 \oplus d11 \oplus d14 = c3 \\
&d0 \oplus d4 \oplus d9 \oplus d12 \oplus d15 = c3 \\
&d1 \oplus d5 \oplus d8 \oplus d10 \oplus d11 \oplus d12 \oplus d14 = c4 \\
&d2 \oplus d7 \oplus d7 \oplus d10 \oplus d11 \oplus d12 \oplus d15 = c5 \\
&d3 \oplus d6 \oplus d9 \oplus d11 \oplus d12 \oplus d13 \oplus d15 = c6
\end{aligned} \quad (3)$$

Where $\oplus$ shows XOR, due to c0–c6, r0–r6 becomes zero in case of fault-free situations. Eq. (3) can be converted into a matrix form as under.

$$\begin{bmatrix}
1001110010001100100000 \\
0100100110010110010000 \\
0010011001110010001000 \\
1000100001001001000100 \\
0100010010111010000010 \\
0010000101111001000001 \\
0001001001011101000000 1
\end{bmatrix}
\begin{bmatrix}
d0 \\ d1 \\ d2 \\ d3 \\ d4 \\ d5 \\ d6 \\ d7 \\ d8 \\ d9 \\ d10 \\ d11 \\ d12 \\ d13 \\ d14 \\ d15 \\ c0 \\ c1 \\ c2 \\ c3 \\ c4 \\ c5 \\ c6
\end{bmatrix}
= [r0r1r2r3r4r5r6r7] \quad (4)$$

Suppose we have 16 data bits (d0–d15), there should be minimum 7 number of check bits according to (2). Eq. (3) Can be used to calculate c0-c6 check bits. Supposing that check 23-bit code word is d0–d15-c0–c6 (c0-c6 are check bits), the subsequent equations are used to correct a single bit fault, a double contiguous fault, a triple contiguous fault or a double almost contiguous fault.

2.2. Multi-bit Error Correction Decoder

$$
\begin{aligned}
&d0 \oplus d3 \oplus d4 \oplus d5 \oplus d8 \oplus d12 \oplus d13 \oplus c0 = r0 \\
&d1 \oplus d4 \oplus d7 \oplus d8 \oplus d11 \oplus d13 \oplus d14 \oplus c1 = r1 \\
&d2 \oplus d5 \oplus d6 \oplus d9 \oplus d10 \oplus d11 \oplus d14 \oplus c2 = r2 \\
&d0 \oplus d4 \oplus d9 \oplus d12 \oplus d15 \oplus c3 = r3 \\
&d1 \oplus d5 \oplus d8 \oplus d10 \oplus d11 \oplus d12 \oplus d14 \oplus c4 = r4 \\
&d2 \oplus d7 \oplus d7 \oplus d10 \oplus d11 \oplus d12 \oplus d15 \oplus c5 = r5 \\
&d3 \oplus d6 \oplus d9 \oplus d11 \oplus d12 \oplus d13 \oplus d15 \oplus c6 = r6
\end{aligned} \tag{5}
$$

The leftmost matrix which controls which check bits and data bits are XORed is called check matrix, while the rightmost matrix is called syndrome matrix which has all zero values in case of error-free situations while it is non- zero in case of error.

Syndrome matrix corresponds to a single error, consecutive double errors, almost successive errors or three consecutive errors which is described in [8] by detail table. Different combinations of syndrome bits precisely determine the location of an error bit in the code word.

Now, these inherited properties of the scheme are when combined with DAP it can be used to handle transient errors including crosstalk. This scheme called as Duplicate Consecutive Sextuple Error Correction (D_CSEC) Scheme.

2.3. D_CSEC Encoder

It is a well-known concept that minimum Hamming distance between any two codewords is three in case of SEC scheme and duplicating the code word avoids crosstalk between adjacent wires, D_CSEC uses the same concept. First, the information k-bits are encoded with the MEC code described above. Then these encoded bits are duplicated. Finally, from one of the copies, an overall parity bit is calculated and then this parity is added to the encoded bits. Thus, if the actual MEC code was (nk) code, the concluding number of bits is 2n+1. 16 data bits are encoded to 23 bits, with duplicating and adding parity final bits are 47. Therefore for 16 un-coded bits D_CSEC, the encoded bits are (47, 16).

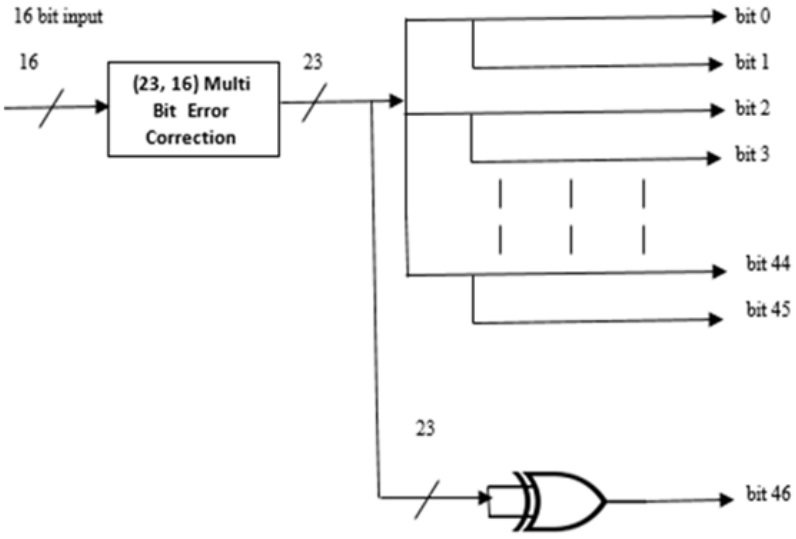

Fig. 1 Duplicate Consecutive Sextuple Error Correction (D_CSEC) encoder

This enables up to sextuple consecutive faults correction. The encoding technique for the D_CSEC code is shown in Fig 1.

2.4. D_CSEC Decoder

The decoder needs syndrome calculation of both the copies and locally calculated parities of each copy also need to be compared with original parity. D_CSEC decoder is shown in Fig2 and describe here.

1. First (A) and Second (B) copies of the original code and original overall parity bit P0 are separated, PA and PB the parties of two separated copies A and B are calculated respectively.

2. SA is syndrome of first (A) copy and SB is syndrome of second (B) copy. If SA is not zero and SB is not zero then the copy A is selected.

3. If both the syndromes are non-zero, then P0 is XORed with PA and PB. If the result is 1, it means either there would be single or triple consecutive error in the respective copy else there would be double consecutive or almost consecutive error. Therefore, in either case, each syndrome is compared with the first column of Table I for single consecutive error, second and third for double consecutive error or double almost consecutive error and the fourth column for triple consecutive, to get the original bits.

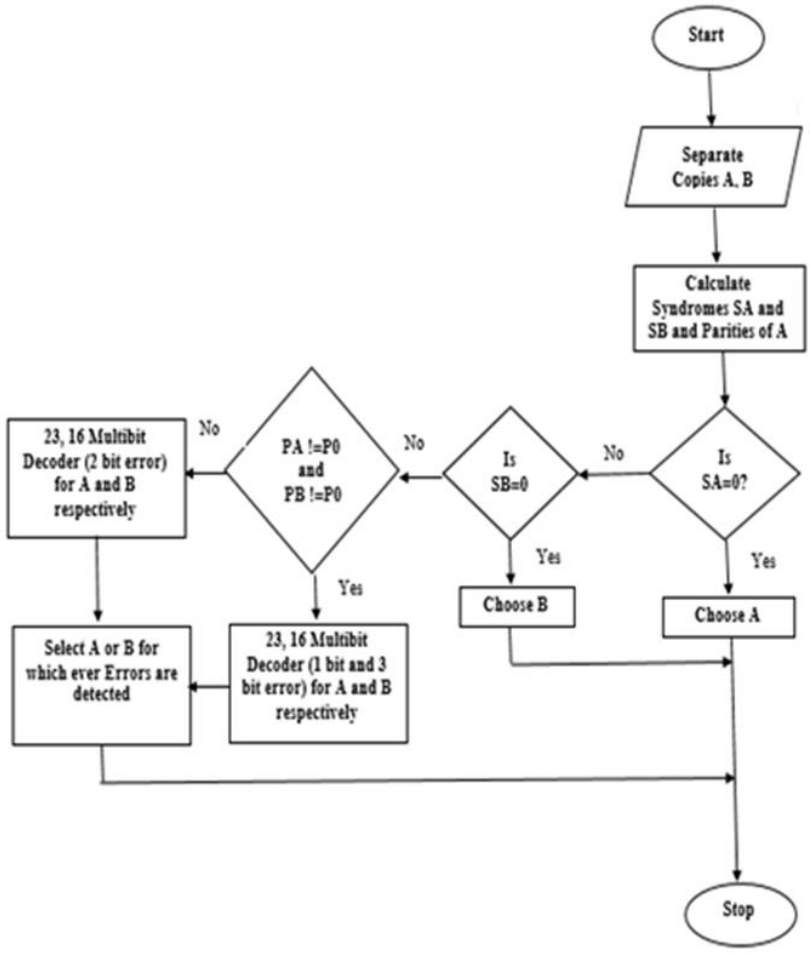

Fig. 2 Duplicate Consecutive Sextuple Error Correction (D_CSEC) decoder flowchart

Table I shows that D_CSEC can correct/detect up to six errors among which three should be consecutive and handle SEU as well as crosstalk due to duplication. Therefore, it gives much-improved reliability.

TABLE I: Comparison of Error Correction and Detection Capability of Different Techniques with D_ CSEC

Technique	Payload	Number of Errors Corrected and Detected	Can Correct (SEU, Crosstalk)
Hamming	5 bits	1-bit error correction and detection	SEU
DAP	Duplication of bits	2-bit error correction and detection	SEU, Crosstalk
JTEC	5 bits, Duplication of bits plus 1 parity bit	4-bit error correction and detection	SEU, Crosstalk
JTEC-SQED	5 bits, Duplication of bits plus 1 parity bits	4-bit error correction and 5-bit error detection	SEU, Crosstalk
S_CSEC	7 bits, Duplication of bits plus 1 parity bit	6-bit error correction plus detection	SEU, Crosstalk

3. Implementation of D_CSEC In NTC based NoC

There are many ways to embed encoder/decoder in the network. These are end-to-end (E2E), hop-to-hop (H2H) or selective hop-to-hop. In case of E2E packets are encoded and decoded at Network Interfaces which gives the minimum latency to packets traversing the network and hence the minimum Average Network Latency. H2H gives the maximum latency because packets are encoded and decoded in each router. Area of routers is also increased with the inclusion of these extra components. In the selective H2H area is reduced compared to H2H because additional elements are included only in the selected routers. There are many techniques how these routers are selected for the inclusion of error correction circuitry. We had a focus on SEU and crosstalk type of errors and proposed ECC technique can correct up to six errors, so we are using E2E as shown in figure 3. Packets are only encoded at source router and decoded at destination routers to correct the errors.

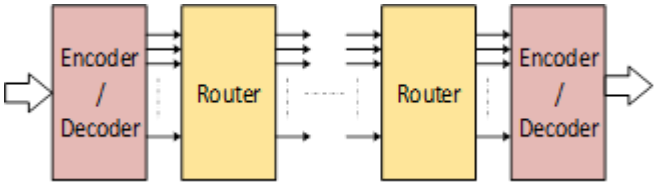

Fig. 3 End-to-end error correction

Figure 4 shows the 4x4 network on routers connected in the mesh topology. Source and Destination routers are shown with brown and blue colors respectively. Intermediate routers are also shown here in yellow in which the packets are moving without error correction. Figure 7 explains the concept of faults accumulation. The proposed method can quickly correct the one error in the path. There is a possibility that another transient error affects the same packet on its way to destination router as shown in figure 7 (b, c, and d). The proposed Sextuple technique can correct up to six errors so if more errors affect the packet bits, can be corrected at the destination node.

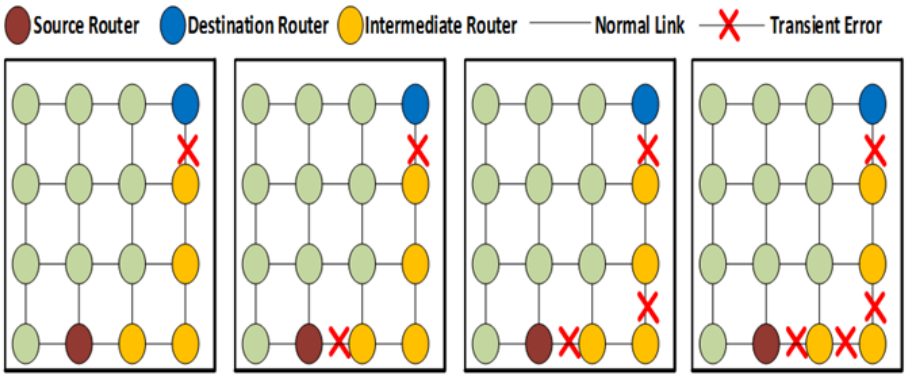

Fig. 4 Transient Faults Injected At Different Location Affecting the Same Packet

4. Source Voltage Lowering with Probability of Word Error

Reliability of communication channel is increased by introducing error control scheme as communication channel becomes robust against transient errors. In the UDSM technology, energy dissipation and nodes reliability are two attached issues. The voltage swings on interconnect wires are reduced with an increase in reliability. Author in [13] describes how to model collective transient UDSM noise using an overall bit error rate (BER) ε, is described in Eq (6)

$$\varepsilon = Q\left(\frac{V_{dd}}{2\sigma}\right) \tag{6}$$

Q-function is described below as

$$Q(x) = \frac{1}{\sqrt{2\pi}}\int_{x}^{\infty} e^{\frac{y^2}{2}}\,dy \tag{7}$$

The Probability of word error is a related to the Bit Error Rate BER ε. Using (6), we can decrease the source voltage by error encoding to V_{dd}, given by [13].

$$\hat{V}_{dd} = V_{dd}\frac{Q^{-1}(\hat{\varepsilon})}{Q^{-1}(\varepsilon)} \tag{8}$$

In (8), V_{dd} is the supply voltage without error encoding, V_{dd} (cap) is the abridged voltage with error encoding scheme and $\hat{\varepsilon}$ is the BER given as

$$\hat{P}_{ECC}(\hat{\varepsilon}) = P_{UNC}(\varepsilon) \tag{9}$$

By lowering the voltage increase the multi-bit faults probability, therefore multi-bit fault correction technique becomes the necessity. We calculate the word error probability for the JTEC–SQED and our proposed D_CSEC schemes.

4.1. Word Error Probability

The residual probability of word errors of JTEC-SQED and D_CSEC are computed to calculate the voltage swing drop due to the presence of these techniques. The word error probability in terms of correct probability is calculated using the following

$$P_{ECC} = 1 - P_{correct} \tag{10}$$

where P_{ECC} is the probability of word error in the mentioned error correction scheme, and $P_{correct}$ correct decoding probability.

4.1.1. Probability of word error in JTEC

The JTEC-SQED encoding technique can correct three errors per data unit. The calculation is for data k information bits. In SEC–DED n data bits are encoded to 2n+1 bits.

JTEC-SQED decodes correctly if error bits are less than three. Thus, correct decoding probability $P_{correct}$ is given by

$$P_{correct} \geq P(2n+1,0) + ... + P(2n+1,4) \tag{11}$$

If n-bits have m number of errors, then BER of ε is given by

$$P(n,m) = \binom{n}{m} \varepsilon^{m} (1-\varepsilon)^{n-m} \tag{12}$$

Thus, the word error probability is calculated using (10), in conjunction with $P_{correct}$ for the JTEC-SQED technique from (6). For lesser values of ε, the probability can be calculated as

$$P_{JTEC\ -SQED} = \binom{2n+2}{5} \varepsilon^{5} \tag{13}$$

4.1.2. Word Error Probability in D_CSEC

Let us suppose that a total bits in one flit is (2n+1) and duplicate copies of original code plus parity bit to calculate the word error probability. Note, encoding code is 1 or 2 bits greater than Hsiao, but that is considered negligible here. D_CSEC can detect and correct up to six errors. Lower limit of probability of error correction is

$$P_{correct} \geq P(2n+1,0) + ... + P(2n+1,6) \tag{14}$$

Using (10) and (14) the probability of word error of this scheme can be approximated as

$$P_{D\ \_\ CSEC} = \binom{2n+1}{6} \varepsilon^{6} \tag{15}$$

Errors that are not consecutive are not corrected by this scheme which can be subtracted from above equation but still its residual word error probability is better than JTEC-SQED. As the capaility of error correction of technique increases its word error probability decreases and voltage swing also reduces. Hence the voltage reduction is achieved by D_CSEC scheme better than JTEC-SQED.

5. Average Latency Results

Gem5 [18] and Garnet [19] is used for the simulation of the NoC. Transient faults due to NTC are injected at various locations of the network during the simulation. The router used for the simulation of the proposed work is five-input, five-output ports and each port have four Virtual Channels (VC's). Packets are divided into five flits. 4X4 network of routers connected in a mesh topology is used for the simulation purpose for synthetic traffic patterns. Traffic patterns selected for simulation purpose are Uniform Random and Tornado with injection rates of 0.1, 0.2, and 0.3 to 1.0. The router operating temperature is selected to be 75˚C. Simulation results show that for synthetic traffic patterns of Uniform Random and Tornado the Average Latency measured in cycles/flit is only increased by 0.5% compared to the fault-free scenario.

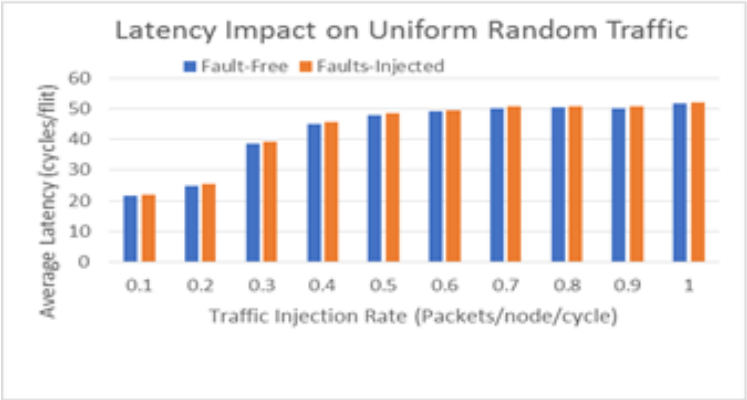

Fig. 5 Latency Impact on Uniform Random Traffic by D_CSEC Scheme

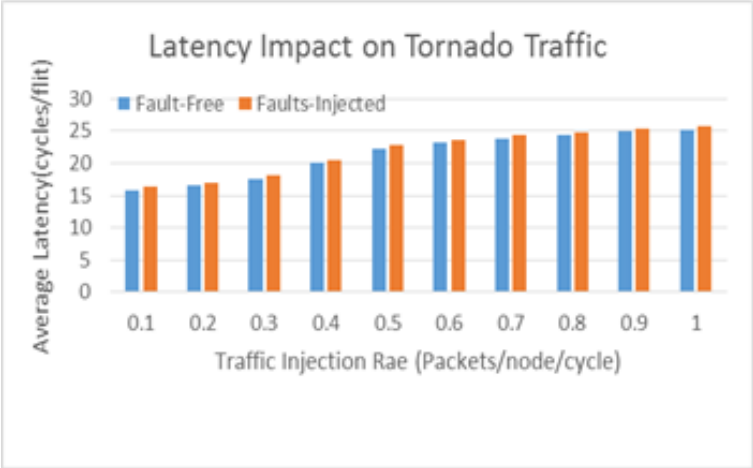

Fig. 6 Latency Impact on Tornado Traffic by D_CSEC Scheme

6. Conclusion

NoC is becoming the auspicious solution for on-chip communication. With increasing number of transistors per square area on the chip, NoC designs are being unprotected to the miscellaneous type of temporary errors. Lowering the energy deception is possible through implementation of NTC concept in NoC, which has drawbacks of transient faults. Therefore, in this work, we propose a multi-bit error correction technique for SEUs as well as crosstalk. Our technique can solve up to six errors per flit. D_CSEC is much more energy efficient in different traffic patterns as compared to the current error rectification practices, though it can correct difficult number of temporary errors.

7. Acknowledgment

We would like to show our gratitude to the Digital Architecture research group in Computer Department, University of Engineering and Technology, Taxila, Pakistan.

8. References

[1] L. Benini and G. De Micheli, “Networks on chips: A new SOC paradigm,” IEEE Comput., vol. 35, no. 1, pp. 70–78, Jan. 2002.

[2] S. Vangal, “An 80-tile 1.28TFLOPS network-on-chip in 65 nm CMOS,” in Proc. IEEE Int. Solid-State Circuits Conf., pp. 98–589, Feb. 2007.

[3] Dresli NS KI, R. G., and others Near-Threshold Computing: Reclaiming Moore’s Law through Energy Efficient Integrated Circuits. Proceedings of the IEEE 98, 253–266, Feb 2010

[4] Sparsh Mithal, “A Survey of Architectural Techniques for Near-Threshold Computing,” ACM Journal on Emerging Technologies in Computing Systems.vol.12, no.4, Article 46, December 2015.

[5] E. Dupont, M. Nicolaidis, and P. Rohr, “Embedded robustness IPs for transient-error-free ICs,” IEEE Design Test Comput., vol. 19, no. 3, pp. 54–68, May-Jun. 2002.

[6] Chidhambaranathan Rajamanikkam, Rajesh JS Koushik Chakraborty, and Sanghamitra Roy, “BoostNoC: Power Efficient Network-on-Chip Architecture for Near-Threshold Computing,” IEEE Computer-Aided Design (ICCAD), 2016 IEEE/ACM International Conference, vol. 13, no. 6, pp. 655–667, Jan 2017.

[7] D. Rossi, V. E. S van Dijk, R. P Kleihorst, A. H. Nieuwland, and C. Metra, “Coding scheme for the low energy consumption fault-tolerant bus,” in Proc. IEEE 8th Int. On-Line Test. The workshop, Bendor, France, 2002, pp. 8–12.

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

[9] Ahmed Garamoun, “Error Correction Techniques on NoC Protocol Layers,” University of Stuttgart, Institute of Computer Architecture and Computer Engineering, June 2009.

[10] B. Fu, “Crosstalk-Aware Multiple Error Control for Reliable On-Chip Interconnects,” Ph.D. thesis, Dept. Electr. Comput. Eng., Univ. Rochester, Rochester, NY, USA, 2010.

[11] D. Bertozzi, L. Benini, and G. De Micheli, “Error control schemes for on-chip communication links: The energy-reliability tradeoff,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 24, no. 6, pp. 818–831, Jun. 2005.

[12] Xiaoxuan She, N. Li, and D. Waileen Jensen, “SEU Tolerant Memory Using Error Correction Code,” IEEE Transactions on Nuclear Ccience,vol.59, no.1, Feb 2012

[13] S. R. Sridhara and N. R. Shanbhag, “Coding for system-on-chip net-works: A unified framework,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 13, no. 6, pp. 655–667, Jun. 2005.

[14] L. Lin, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, “Adaptive error protection for energy efficiency,” in Proc. IEEE Int. Conf. Comput. Aided Des, 2003, pp. 2–7.

[15] D. Bertozzi, L. Benini, and G. D. Micheli, “Low power error resilient encoding for on-chip data buses,” in Proc. Conf. Des., Autom. Test Eur., 2002, pp. 102–109.

[16] F. Z. Rokhani and G. E. Sobelman, “Bus energy consumption for multi-level signals,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 1, 64–71, Jan. 2010.

[17] “Predictive Technology Model (PTM),” Nanoscale Integr. Model. (NIMO) Group, Tempe, AZ, USA, 2012. [Online]. Available: http://ptm. asu.edu/

[18] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood, “The gem5 simulator,” in ACM SIGARCH Comput. Archit. News, vol. 39, no. 2, pp. 1–7, 2011

[19] N. Agarwal, T. Krishna, L. S. Peh, and N. K. Jha, “Garnet: A detailed on-chip network model inside a full system simulator,” in Proc. IEEE Int. Symp. Perform. Anal. Syst. Softw., 2009, pp. 33–42