

# Crosstalk-Aware Channel Coding Schemes for Energy Efficient and Reliable NOC Interconnects

Amlan Ganguly, *Student Member, IEEE*, Partha Pratim Pande, *Member, IEEE*, and Benjamin Belzer, *Member, IEEE*

**Abstract**—Network-on-chip (NOC) is emerging as a revolutionary methodology to integrate numerous intellectual property blocks in a single die. It is the packet switching-based communications backbone that interconnects the components on multicore system-on-chip (SoC). A major challenge that NOC design is expected to face is related to the intrinsic unreliability of the interconnect infrastructure under technology limitations. By incorporating error control coding schemes along the interconnects, NOC architectures are able to provide correct functionality in the presence of different sources of transient noise and yet have lower overall energy dissipation. In this paper, designs of novel joint crosstalk avoidance and triple-error-correction/quadruple-error-detection codes are proposed, and their performance is evaluated in different NOC fabrics. It is demonstrated that the proposed codes outperform other existing coding schemes in making NOC fabrics reliable and energy efficient, with lower latency.

**Index Terms**—Crosstalk avoidance, error correction coding (ECC), multiple error correction, network on chip (NOC), reliability.

## I. INTRODUCTION

CURRENT commercial system-on-chip (SOC) designs integrate a number of embedded functional and storage blocks typically in the range of 10–100 or more [1], [2]. This number is predicted to increase significantly in the near future. Specifically molecular-scale computing will allow single or even multiple order-of-magnitude improvements in device densities. Network-on-chip (NOC) has emerged as an enabling methodology to achieve this high degree of integration [1], [3]. It is well known that with shrinking geometry, NOC architectures will be increasingly exposed to different sources of transient noise, affecting signal integrity and system reliability. Data-dependent crosstalk between adjacent wires is a major source of such transient noise. Worst case crosstalk happens when the two neighbors transition in opposite directions with respect to the victim wire. With shrinking geometry, the inter-wire spacing decreases rapidly [4], while the height and width of the wires do not scale at the same rate. This in turn tends to increase the cross-sectional aspect ratio, increasing the effective coupling capacitance between intralayer adjacent wires with negative effects not only on signal integrity but also on delay and energy dissipation. The fact that the dielectric constant does

not scale down at the same rate also contributes to the increase in coupling capacitance between adjacent wires in the same metal level. Besides crosstalk, there are several other important sources of transient errors like ground bounce, supply voltage scaling, electromagnetic radiation, and alpha particle hits etc. [5], which can cause random data upset. As noted in [6] due to shrinking feature size in future technologies the soft error rate (SER) due to high-energy particles is predicted to increase by several orders of magnitude. As these soft errors are not necessarily correlated, a higher SER can cause uncorrelated multiple bit errors in data blocks. By incorporating crosstalk avoidance coding (CAC) in NOC data streams, the effective coupling capacitance of the wire segments and hence the communication energy can be reduced, as they are linearly related [7]. But CACs are not sufficient to protect the NOC from other transient errors. In the current generation of NOCs, simple single-error correction (SEC) codes are applied to achieve both reliability and low power [8], [9]. But these SECs are not capable of reducing the effective coupling capacitance of the wires of the communication channel. Moreover, with the reduction of feature sizes and power-supply voltages and the increase in operating frequencies, circuits are much more susceptible to transient noise. This results in much higher error rates that ultimately overwhelm SECs, rendering them insufficient for future NOCs. In this paper, we propose design of joint crosstalk avoidance and multiple-error-correction codes (CAC/MEC) and quantify their performance in making NOC fabrics reliable and energy efficient.

## II. RELATED WORK

Applicability of error control coding in designing robust SOCs has been explored previously. In [10], the authors have presented a unified framework for applying coding for SOCs. But this was principally targeted to traditional bus-based systems. The worst case switching capacitance of a wire is  $(1 + 4\lambda)C_L$ , [11], where  $\lambda$  is the ratio of the coupling capacitance to the bulk capacitance and  $C_L$  is the load capacitance, including the wire's self-capacitance. A few joint crosstalk avoidance and single-error-correction codes (CAC/SEC) have been proposed by different research groups. Among these joint codes, dual rail (DR) code [12], duplicate add parity (DAP) [10], boundary shift code (BSC) [13], and modified dual rail code (MDR) [14] reduce the switching capacitance associated with crosstalk to  $(1 + 2\lambda)C_L$ . In [9], the authors have addressed error resilience in NOC fabrics and the tradeoffs involved in various error recovery schemes. In this paper, the authors investigated simple error detection codes like parity or cyclic redundancy check (CRC) codes

Manuscript received October 18, 2007; revised March 25, 2008. First published March 16, 2009; current version published October 21, 2009. This work was supported in part by the National Science Foundation under Grant CCF-0635390.

The authors are with the School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99163 USA (e-mail: ganguly@eecs.wsu.edu; pande@eecs.wsu.edu; belzer@eecs.wsu.edu).

Digital Object Identifier 10.1109/TVLSI.2008.2005722

and single-error-correcting, double-error-detecting Hamming codes. The performance of SEC Hamming codes, single-error correction and double-error-detection (SEC/DED) Hsiao codes, and symbol-error-correcting codes in NOC fabrics was evaluated in [15]. Most of the above works depended on SECs. But with technology scaling, SECs are not sufficient to protect NOCs from varied sources of transient noise. This was acknowledged for the first time in [10] in the context of traditional bus-based systems. It was pointed that with aggressive supply scaling and increase in deep submicron (DSM) noise, more powerful error-correction schemes than the simple CAC/SEC will be needed to satisfy reliability requirements. One specific problem pertaining to coding in NOCs is highlighted in [8]. In this work, it was concluded that error detection followed by retransmission is more energy efficient than forward error correction. But this work was done in a much older technology generation (0.25  $\mu\text{m}$  technology) than the ultradeep submicron (UDSM) regime, where the problems arising out of transient noise will be most severe. As mentioned in the concluding remarks of [8], in the UDSM domain communication energy is going to overcome computation energy. Retransmission will give rise to multiple communications over the same link and hence ultimately it will not be very energy efficient. In systems dominated by retransmission additional error-correction mechanisms for the control signals also need to be incorporated. To resolve the issues regarding the effectiveness of coding for energy-efficient protection of signal integrity in NOCs, we propose a series of studies on the design of novel joint CAC/MEC codes and their application in NOCs.

### III. JOINT CROSSTALK AVOIDANCE AND TRIPLE-ERROR-CORRECTION CODE

Aggressive scaling of device dimensions and the consequent increase in vulnerability to transient errors makes exploration of multiple-error-correcting codes imperative. However, higher order error-correcting codes alone are not enough to ensure the reliable performance of NOCs in the current and future technology nodes. Crosstalk avoidance must be made an integral part of any multiple-error-correction schemes. An important point to note here is that the proposed joint CAC/MEC scheme is not just the design of another multiple-error-correcting code, but one that reduces worst case crosstalk as well with little computational complexity. It has been shown in [10] that only a linear CAC can be implemented after any error-control-coding scheme to enable error correction and crosstalk avoidance simultaneously. Furthermore, it has been proven that to achieve maximum possible reduction in crosstalk there is no linear coding scheme with fewer wires than duplication [10]. Below, we propose a simple combined crosstalk avoiding triple-error-correction scheme called the Joint Crosstalk Avoidance and Triple Error Correction (JTEC) code.

#### A. JTEC Encoder

The encoder for the JTEC scheme utilizes the facts that the minimum Hamming distance between any two codewords of an SEC Hamming code is three and also that duplication avoids worst case crosstalk between adjacent wires. First the information bits, say  $k$  in number, are encoded with the SEC Hamming

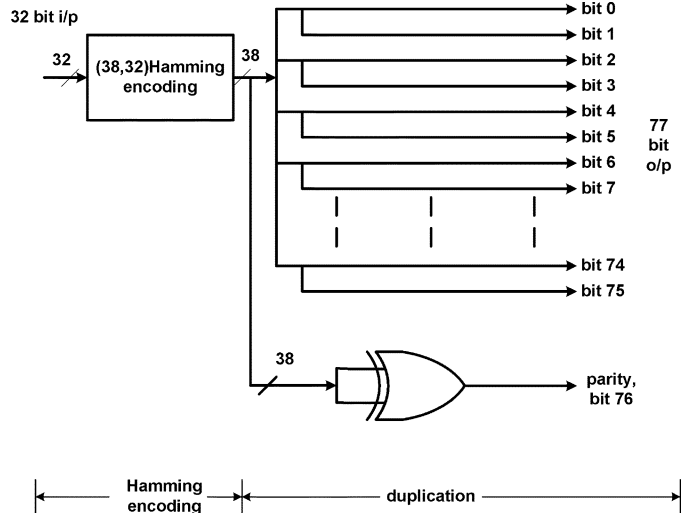


Fig. 1. JTEC encoder schematic.

code. Then each of these Hamming encoded bits is duplicated. Finally, an overall parity bit, calculated from either one of the Hamming copies, is appended to the encoded bits. Thus, if the initial SEC Hamming code was an  $(n, k)$  code, the final number of bits in the encoded bit is  $2n + 1$ . For example, if the original information word consisted of 32 bits then after encoding with an SEC  $(38, 32)$  shortened Hamming code it becomes 38, and after the duplication and addition of the overall parity bit it becomes 77. Thus for an uncoded 32 bit wide flit, JTEC is a  $(77, 32)$  coding scheme. The Hamming distance of the  $(38, 32)$  SEC Hamming codes is 3. The duplication process increases this to 6, and addition of an overall parity bit makes the final minimum Hamming distance between the codewords to be 7. Thus this enables triple-error correction. The duplication simultaneously serves to avoid opposite bit transitions in adjacent wires so that the worst case transition of a bit pattern from 101 to 010 and vice versa can be avoided. Consequently, the worst case effective crosstalk capacitance of a wire segment of the communication channel can be reduced from  $(1 + 4\lambda)C_L$  to  $(1 + 2\lambda)C_L$ . The encoding mechanism for the JTEC code is shown in Fig. 1 through a schematic diagram.

#### B. JTEC Decoder

The decoder for this scheme requires syndrome computation on the two copies and comparisons of the transmitted overall parity bit with the locally generated parities recomputed at the decoder from each individual copy. The algorithm for the JTEC decoder is shown through a flowchart in Fig. 2(a) and is outlined as follows.

- 1) The two Hamming copies A and B and the transmitted overall parity bit  $P_0$  are isolated. Also, two parity bits are calculated separately from A and B, say  $P_A$  and  $P_B$ .
- 2) If the syndrome of copy A ( $S_A$ ) is nonzero then it implies that it can have one or two errors.
  - o Now, if  $P_0$  is equal to  $P_A$  then it means A has two errors, and B can have at the most a single error. So, copy B is chosen for the final SEC Hamming decoding stage which will correct this single error.

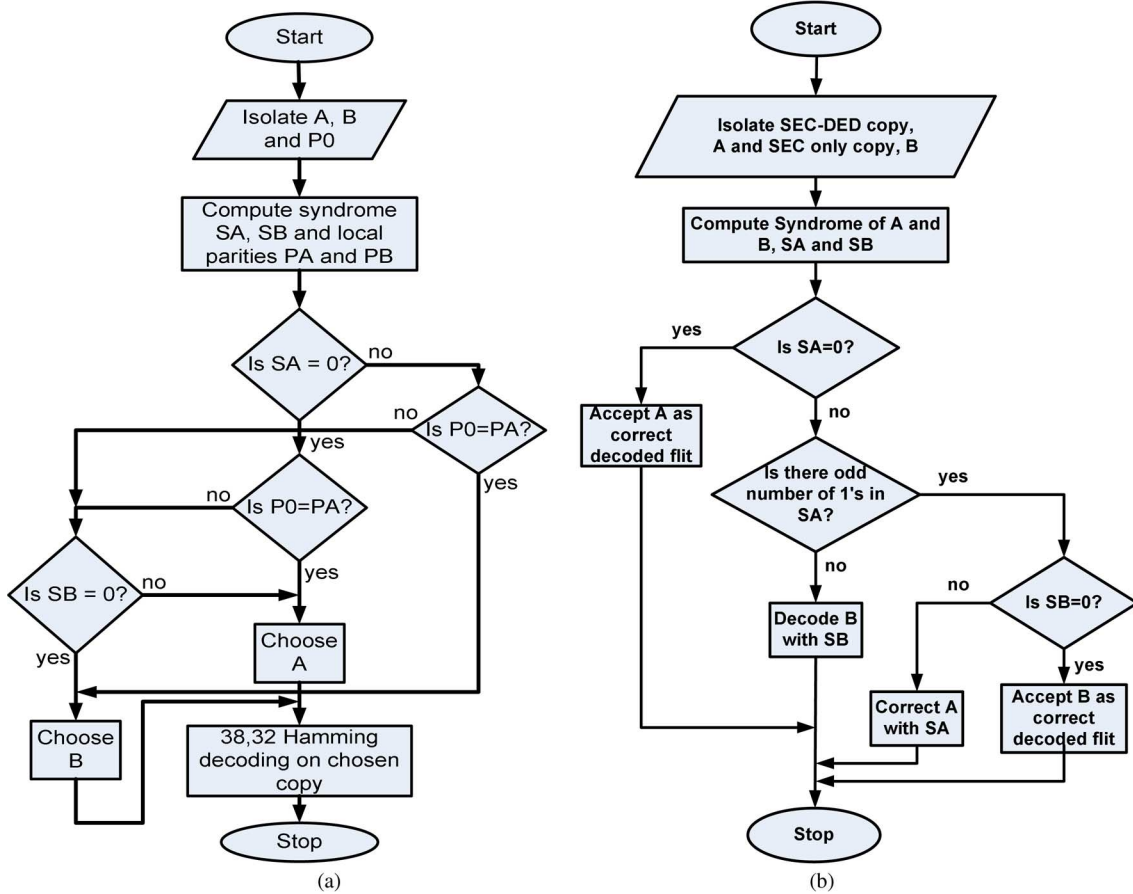


Fig. 2. Flowcharts for the decoding schemes for (a) JTEC and (b) optimized JTEC.

o However, if  $P_0$  is not equal to  $P_A$  then the syndrome of copy B ( $S_B$ ) is computed and copy B is chosen if  $S_B$  is zero or copy A is chosen if  $S_B$  is nonzero as A has a single error then.

3) If the syndrome of copy A was zero then A can have none or three errors.

o In this case if  $P_0$  is the same as  $P_A$  then copy A is chosen.

But if the two parity bits do not match then the syndrome of copy B is computed, and if it is nonzero then copy A is chosen. Copy B is chosen if the syndrome is zero.

The final chosen copy is sent for SEC Hamming decoding to produce the triple error corrected output.

Both the encoding and the decoding processes discussed above essentially necessitate the use of long chains of XOR gates to compute the overall parity bits. This happens because the overall parity bits are modulo-2 summation of all the Hamming encoded bits. Thus, for large flit widths, this may imply prohibitively complex hardware with negative effect on energy dissipation and timing. The hardware complexity and critical path delay of the codec block can be reduced by adopting an optimization method as outlined in the next subsection.

C. Optimization of the Code

Both the encoder and the decoder for the JTEC scheme use long chains of XOR gates. The complexity of both the circuits

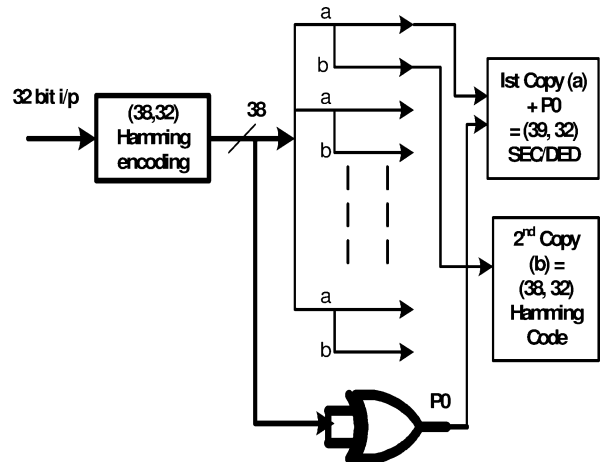


Fig. 3. Encoder modification for optimization.

can be optimized by using a two-fold approach. First, the overall parity bit in conjunction with one of the  $(n, k)$  Hamming coded copies is used as an  $(n + 1, k)$  -SEC-DED codes. For the specific example of 32 original information bits, the  $(38, 32)$  Hamming coded bits become  $(39, 32)$  SEC-DED codes after appending the overall parity. This modification is shown in Fig. 3. A syndrome computation on these SEC-DED codes can be used to indicate a single or a double error in those 39 bits. If there is a single error then it can be corrected using the syndrome. If

$$\begin{aligned}
 H_{\text{HammingSEC-DED}} &= \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & \dots & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & \dots & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & \dots & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & \dots & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \dots & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \\
 & \quad \text{(a)} \\
 H_{\text{HsiaoSEC-DED}} &= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \dots & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & \dots & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & \dots & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \dots & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & \dots & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & \dots & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \dots & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\
 & \quad \text{(b)}
 \end{aligned}$$

Fig. 4.  $H$ -matrix for (a) (39, 32) Hamming SEC-DED code and (b) (39, 32) Hsiao SEC-DED code.

there are two errors in these 39 bits then the other copy cannot have more than a single error for a triple-error-correction code to be able to correct the error pattern. This can then be corrected by the syndrome computation on that copy. If the first 39 SEC-DED bits have all the three errors then this triple error cannot be corrected by the SEC-DED codes, but then the other copy will be error free and can be accepted. This algorithm is explained through a flowchart in Fig. 2(b). This modified decoding approach reduces hardware complexity considerably, as the step of locally recomputing the overall parity bits  $P_A$  and  $P_B$  is avoided. Also, the last step of a Hamming SEC decoding becomes redundant in the optimized scheme. Thus the decoding circuit can be simplified by this step.

The second level of optimization consists of replacing the (39, 32) Hamming SEC-DED with the (39, 32) Hsiao SEC-DED code [16]. The last parity bit of the Hamming SEC-DED scheme is basically an overall parity bit computed as the XOR sum of all the 38 bits of the Hamming encoded flit. This is indicated by the last row of the  $H$ -matrix for the Hamming SEC-DED codes in Fig. 4(a) which has all "1" entries. However, if the Hamming SEC-DED are replaced by the Hsiao SEC-DED codes then the number of XOR gates required to compute any of the parity bits can be restricted to the average number of XOR gates for all the seven parity bits [16]. For the (39, 32) Hsiao code, this average number of XOR gates turn out to be 14.7 and hence some of the seven parity bits need 14 and others 15 XOR gates as shown by the  $H$ -matrix for this scheme in Fig. 4(b). Consequently, the number of XOR gates can be drastically reduced by using Hsiao code instead of Hamming SEC-DED, and the delays along the critical paths of both the encoder and decoder are also reduced as they do not have long chains of 38 XOR gates any longer.

Another important point to be noted here is that the second copy which was originally a duplicated (38, 32) Hamming SEC code will now just be a duplication of the 38 bits from (39, 32) Hsiao code including the 32 original information bits and any six of the seven parity bits generated by the Hsiao coding. It is shown in Appendix I that these 38 bits will still have single-error

correction capability, which is vital for the overall triple-error correction as discussed earlier.

This twofold approach reduces the delay and hardware requirements for not only the decoder but also for the encoder. The encoder now will have to encode using the generator matrix of the Hsiao code which has either 14 or 15 XOR gates for each parity bit, unlike the Hamming SEC-DED codes which used an overall parity bit using 38 such gates for the seventh parity bit. Though the above optimization technique is explained with the specific example of the (39, 32) Hsiao SEC-DED code, the principle generally holds for flits of all lengths as in essence, this optimization methodology uses the fact that the Hsiao SEC-DED code is more optimized in terms of hardware complexity compared to the standard Hamming SEC-DED.

#### IV. SIMULTANEOUS TRIPLE-ERROR CORRECTION AND QUADRUPLE-ERROR DETECTION

The JTEC scheme explained above can be modified to achieve simultaneous triple-error correction and quadruple-error detection to detect all uncorrectable error patterns in case there are any. Thus, the JTEC and Simultaneous quadruple-error-detection code (JTEC-SQED) can correct up to all three-error patterns on the fly as well as detect all four-error patterns that cannot be corrected by the JTEC scheme alone. The modification and associated overheads are discussed in the following subsection.

##### A. JTEC-SQED Encoder

The encoder uses the Hsiao SEC-DED code of an appropriate size to achieve simultaneous triple-error correction and quadruple-error detection. The original information bits are first encoded according to Hsiao SEC-DED where the minimum Hamming distance between codewords becomes 4. Then all the encoded bits are duplicated to increase the Hamming distance to 8 which will enable detection of quadruple-error patterns. This code will also have the same crosstalk avoidance capability as the JTEC. Hsiao SEC-DED is used because of the advantages in optimization mentioned in Section III. Essentially,

the encoded flit now contains two Hsiao SEC–DED copies. The JTEC–SQED scheme achieves simultaneous triple-error correction and quadruple-error detection, as it differs from the JTEC only in appending a second copy of the last parity bit of the Hsiao SEC–DED code to the JTEC bits, preserving all the bits necessary for the JTEC decoding scheme.

### B. JTEC–SQED Decoder

The decoder needs to set a flag whenever it encounters a four-error pattern that cannot be corrected by the triple-error-correcting algorithm. In the following, we discuss the several cases that may lead to this and how each of the cases can be detected.

- 1) When each of the two Hsiao SEC–DED encoded copies have double errors, then the syndromes of both copies will be able to detect the presence of such double error patterns.
- 2) When there is a single error in one copy and a triple error in the other, the triple-error pattern in the Hsiao SEC–DED code will always give an odd-weight syndrome; this fact is proved in Appendix II. The syndromes are used to decode each individual copy. If both decoded copies do not match then there must have been a triple error in one of the copies, indicating an overall quadruple error pattern.
- 3) The only other possibility is when there are four errors in one copy and none in the other. In that case, the syndrome of the erroneous copy can be either zero, if the errors make it another Hsiao codeword, or nonzero. If it is zero then the copies will be different indicating a quadruple error pattern. If the syndrome of the erroneous copy is nonzero then the JTEC decoding algorithm will be able to select the correct copy.

The JTEC–SQED scheme simultaneously corrects triple errors and detects quadruple error patterns with additional hardware as compared to the JTEC scheme alone. The result of the triple-error correction has to be discarded if a quadruple-error pattern is detected, because that result maybe inaccurate if there is a quadruple error pattern in the flit. A quantitative analysis of the overheads in terms of energy dissipation, timing, and area requirements of the proposed schemes is elaborated in the following sections.

## V. VOLTAGE SWING REDUCTION WITH RESIDUAL WORD ERROR PROBABILITY

Incorporation of error-control coding enhances the reliability of the communication channel as it becomes robust against transient malfunctions. In the UDSM technology, nodes reliability and energy dissipation are two inseparable issues. Increase in reliability by incorporating coding can be translated into a reduction in voltage swing on the interconnect wires, as they can tolerate lower noise margins. Hence, this results in savings in energy dissipation, as it depends quadratically on the voltage swing. In this section, we quantify these gains by modeling the voltage swing reduction as a function of increased error-correction capability.

The cumulative effect of all transient UDSM noise sources can be modeled as an additive Gaussian noise voltage  $V_N$  with variance  $\sigma_N^2$  [10]. Using this model, the bit error rate (BER),  $\varepsilon$ ,

depends on the voltage swing,  $V_{dd}$ , according to the following relation:

$$\varepsilon = Q\left(\frac{V_{dd}}{2\sigma}\right) \quad (1)$$

where the  $Q$ -function is given by

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-\frac{y^2}{2}} dy. \quad (2)$$

The word error probability is a function of the channel BER  $\varepsilon$ . If  $P_{UNC}(\varepsilon)$  is the residual probability of word error in the uncoded case and  $P_{ECC}(\varepsilon)$  is the residual probability of word error with error-control coding, then it is desirable that  $P_{ECC}(\varepsilon) \leq P_{UNC}(\varepsilon)$ . Using (1), we can reduce the supply voltage in presence of coding to  $\hat{V}_{dd}$ , given by [10]

$$\hat{V}_{dd} = V_{dd} \frac{Q^{-1}(\hat{\varepsilon})}{Q^{-1}(\varepsilon)}. \quad (3)$$

In (3),  $V_{dd}$  is the nominal supply voltage in the absence of any coding,  $\hat{V}_{dd}$  is the reduced voltage swing with coding, and  $\hat{\varepsilon}$  is the BER such that

$$P_{ECC}(\hat{\varepsilon}) = P_{UNC}(\varepsilon). \quad (4)$$

Use of lower voltage swing makes the probability of multibit error patterns higher, necessitating the use of multiple-error-correcting codes in order to maintain the same word error probability as the uncoded case. To compute  $\hat{V}_{dd}$  for various coding schemes with different error-correction capabilities the residual word error probability,  $P_{ECC}(\varepsilon)$ , for each of the schemes need to be computed. In the following subsections, we compute the residual word error probability for the JTEC and the JTEC–SQED schemes.

### A. Residual Probability of Word Error

To compute the possible voltage swing reduction in presence of JTEC and JTEC–SQED, we compute the residual probability of word errors for these schemes. The probability of word error for the JTEC and JTEC–SQED can be easily computed by first calculating the probability of correct decoding. The set of correctly decoded words is always complementary to the set of residual word errors. Hence the residual word error probability can be computed using the equation as follows:

$$P_{ECC} = 1 - P_{\text{correct}} \quad (5)$$

where  $P_{ECC}$  is the residual word error probability in the presence of coding, and  $P_{\text{correct}}$  is the probability of correct decoding.

1) *Residual word error probability for the JTEC:* the JTEC coding scheme is capable of correcting up to three errors in a single flit. Taking into consideration all the cases where correct decoding is possible the residual error probability of the coding scheme is computed. The formulations below hold for any flit of  $k$  information bits which are first coded by  $(n+1, k)$  Hsiao

SEC–DED into  $n + 1$  bits and then only  $n$  bits are duplicated to make the total encoded flit  $2n + 1$  bits wide.

Correct decoding in case of JTEC is possible when the count of errors in the entire flit is three or less. It might also be able to correct some higher number of errors. Thus, the lower bound on the probability of correct decoding,  $P_{\text{correct}}$  is given by

$$P_{\text{correct}} \geq P(2n + 1, 0) + \dots + P(2n + 1, 3) \quad (6)$$

where the probability of  $m$  errors in  $n$  bits with a BER of  $\varepsilon$  is given by

$$P(n, m) = \binom{n}{m} \varepsilon^m (1 - \varepsilon)^{n-m}. \quad (7)$$

Therefore, the probability of the residual word error is given in accordance with (5), using  $P_{\text{correct}}$  for the JTEC scheme from (6). For small values of  $\varepsilon$ , this probability can be approximated as

$$P_{\text{JTEC}} = \binom{2n + 1}{4} \varepsilon^4. \quad (8)$$

2) *Residual word error probability of JTEC–SQED*: to compute the residual word error probability for the JTEC–SQED scheme let us assume that the total number of bits in the flit is  $2n + 2$ , where there are two copies of  $(n + 1, k)$  SEC–DED code. Since JTEC–SQED can either correct or detect up to four errors, the lower bound on the probability of correct decoding can be obtained as

$$P_{\text{correct}} \geq P(2n + 2, 0) + \dots + P(2n + 2, 4). \quad (9)$$

Using (5) and (9), the residual word error probability of the JTEC–SQED scheme for small values of  $\varepsilon$  can be approximated as

$$P_{\text{SQEDR}} = \binom{2n + 2}{5} \varepsilon^5. \quad (10)$$

Using (3), (8), and (10) for the residual probability of word errors, the voltage swing reduction for the proposed schemes can be computed. Fig. 5 shows the reduction in voltage swing,  $\hat{V}_{\text{dd}}$  as a function of word error probability. For the sake of comparison, other coding schemes proposed earlier are also considered. Specifically, the sole error detecting scheme without any crosstalk avoidance, energy dissipation (ED) employing the Hamming code [8], the joint crosstalk avoiding single-error correcting code like DAP/DR [10], [12], and the joint crosstalk avoiding double error correction code, CADEC [17] are considered along with the newly proposed JTEC and JTEC–SQED schemes.

As the error correction capability of the coding scheme increases the residual word-error probability commensurately decreases. Hence, the voltage swing can also be reduced. Consequently, JTEC and JTEC–SQED can achieve more voltage reduction than the existing schemes. However, the voltage swing cannot be reduced to arbitrarily low values by increasing the error-correction capability of the code due to the saturating nature of the inverse- $Q$  function used in (3). Fig. 6 depicts the reduction in voltage swing against the error correction capability of the codes using the model described in (1) through (3). The value of the word-error rate chosen for this plot is  $10^{-20}$  [10].

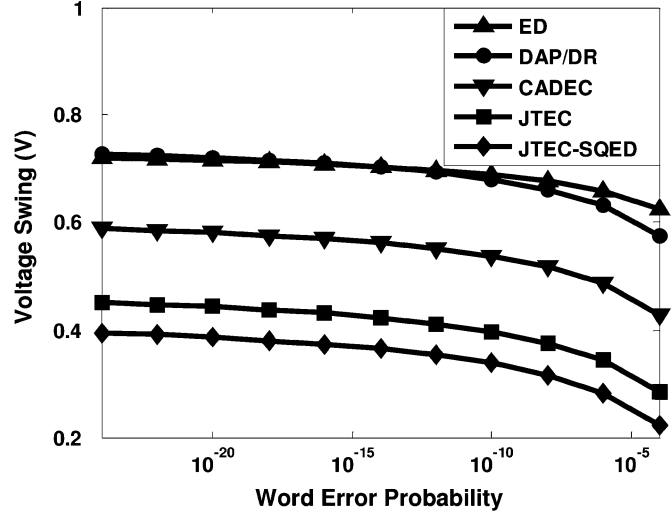


Fig. 5. Plot of voltage swing reduction as a function of word error rates.

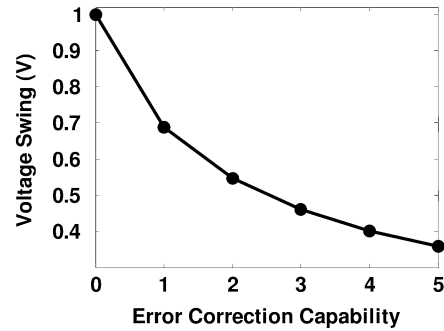


Fig. 6. Voltage Swing Reduction as a function of error correction capability.

The plot is made by considering the fact that the residual probability of the word error of any ECC is proportional to  $\varepsilon^{j+1}$ , where  $j$  is the error-correcting capability of the corresponding code.

According to Fig. 6, the achievable reduction in voltage swing shows an asymptotic trend as the correction capability of the code is increased. For example, the difference in voltage swing between triple- and quintuple-error correction is much less than that between single and triple. As the voltage swing reduction along the wire segments is the predominant source of energy savings in the NOC, beyond the quadruple-error-correction/detection code the energy dissipation in the codes may overshadow the savings in the interconnects. Hence, it may not be advantageous to use arbitrarily high-order error-correction codes.

It should be noted that well-known multiple-error-correcting codes (MEC) like BCH codes have no inherent crosstalk avoidance properties. Single-error correcting BCH codes are equivalent to the SEC Hamming codes used in JTEC. On the other hand, MEC BCH codes have substantially higher parity bit overhead requirements than the Hamming codes employed in JTEC. Hence, implementation of a linear CAC (e.g., duplication) on BCH codewords would require significantly more parity overhead than JTEC and JTEC–SQED, though it would provide more than triple-error correction. Furthermore, MEC BCH codes have substantially higher decoding complexity than the SEC Hamming codes [18]. But in Fig. 6, it is shown that

there is a diminishing return on the amount of voltage swing reduction achievable for a given error-correction capability  $t$ , and that very small reductions occur for values of  $t > 4$ . Since voltage swing reduction is the main cause of energy savings in CAC/MEC schemes, a linear BCH-based CAC/MEC scheme could actually increase the energy dissipation, due to the increased parity and computational requirements of BCH codes. Consequently, linear BCH-based CAC/MEC schemes will be unsuitable for implementation in NOC interconnects.

## VI. ENERGY DISSIPATION IN NOC INTERCONNECTS

In NOC architectures, the functional cores communicate with each other through switches. We assume wormhole routing [27] as the data transport mechanism where the packet is divided into fixed length flow control units or flits. When flits travel between the switches on the interconnection network, both the interswitch wires and the logic gates in the switches toggle, resulting in energy dissipation. To quantify the energy dissipation characteristics of the proposed schemes, we need to determine the energy dissipated per cycle by the entire NOC fabric. In the uncoded case, the energy dissipated per cycle is given by

$$E_{\text{NOC/cycle}}^{\text{uncoded}} = E_{\text{link}}^{\text{uncoded}} \eta_{\text{interswitch}} + \frac{E_{\text{switch}}}{N_{\text{switch}}} \eta_{\text{intraswitch}} \quad (11)$$

where  $E_{\text{link}}^{\text{uncoded}}$  and  $E_{\text{switch}}$  are the energy dissipation of the interswitch link and the NOC switches, respectively. The numbers of flits traversing the interswitch and the intraswitch stages in a single cycle are given by  $\eta_{\text{interswitch}}$  and  $\eta_{\text{intraswitch}}$ , respectively. The NOC switch architecture adopted for this paper has multiple pipelined stages as discussed later in Section VII. Since a single flit cannot occupy more than one stage in one cycle, the energy dissipation of the switch per flit per cycle is obtained by dividing  $E_{\text{switch}}$  by the number of stages that it is pipelined into  $N_{\text{switch}}$ . After incorporating the coding schemes the energy dissipation per cycle can be obtained as follows:

$$E_{\text{NOC/cycle}}^{\text{coded}} = E_{\text{link}}^{\text{coded}} \eta_{\text{interswitch}} + \left( \frac{E_{\text{codec}} + E_{\text{interface}}}{N_{\text{codec}}} + \frac{E_{\text{switch}}}{N_{\text{switch}}} \right) \eta_{\text{intraswitch}} \quad (12)$$

where  $E_{\text{codec}}$  and  $E_{\text{interface}}$  are the energy dissipations of the codecs and the interface circuitry used to obtain low voltage on the interconnects. Similar to the switch, the energy dissipation of the codecs per cycle need to be considered and are hence divided by the number of stages,  $N_{\text{codec}}$ . The pipelined architecture in the presence of coding is described under timing analysis in Section VII.

The main reason for incorporating coding in NOCs is to achieve the dual purpose of enhancing reliability and lowering energy dissipation. The principal source of lowered energy dissipation is the reduced voltage swing on the interconnects enabled by increased reliability through coding. Additionally, lowering the effective crosstalk capacitance of interswitch wires augments the gains in energy savings. However, while computing the energy dissipation profiles, the overheads caused by the coding schemes must also be taken into account. The coding schemes introduce redundant bits in the flits and hence

increase the number of wires. The extra wires also dissipate energy and hence are considered as a part of  $E_{\text{link}}^{\text{coded}}$  in (12). The encoders and decoders including the interface circuitry used to achieve a lower voltage swing on the wires also dissipate energy and are included in the computation in (12). Following this, the savings in energy compared to the uncoded case in each cycle,  $E_{\text{savings/cycle}}$  is given as

$$E_{\text{savings/cycle}} = E_{\text{NOC/cycle}}^{\text{uncoded}} - E_{\text{NOC/cycle}}^{\text{coded}}. \quad (13)$$

$E_{\text{NOC/cycle}}^{\text{uncoded}}$  can be calculated using (11) considering the fact that there is no codec and interface overhead, while  $E_{\text{NOC/cycle}}^{\text{coded}}$  can be calculated from (12) considering all the overheads. Therefore, it can be seen from (11), (12), and (13) that the savings in energy dissipation compared to the uncoded case does not depend on the energy dissipation of the NOC switches.

The energy dissipated in each switch,  $E_{\text{switch}}$ , and each codec,  $E_{\text{codec}}$  is determined using Synopsys<sup>TM</sup> Prime Power as discussed in Section VIII. The interconnect energy,  $E_{\text{link}}$ , depends on the length of each interswitch wire segment which varies depending on the NOC topology [19], [27]. For Mesh architecture the interswitch wire length is given by

$$l = \frac{\sqrt{\text{Area}}}{\sqrt{M} - 1} \quad (14)$$

where ‘‘Area’’ is the area of the silicon die used, and  $M$  is the number of intellectual property (IP) blocks in the SOC. The interswitch wire length for folded-torus architecture is twice that of the Mesh [27]. The interswitch wire length for the BFT architecture between levels  $a + 1$  and  $a$  is given by (15), where  $levels$  is the total number of levels in the BFT architecture given by  $\text{Log}_4 M$ :

$$l_{a+1,a} = \frac{\sqrt{\text{Area}}}{2^{\text{levels}-a}}. \quad (15)$$

The capacitances of each interconnect stage and subsequently  $E_{\text{link}}$  was obtained through HSPICE simulations taking into account the specific layout for each topology [27]. The energy dissipated by the low-swing interface circuitry was also obtained through HSPICE simulations.

To obtain the number of flits traversing each stage per cycle  $\eta_{\text{interswitch}}$  and  $\eta_{\text{intraswitch}}$ , a cycle-accurate network simulator is employed. It is flit-driven and uses wormhole routing. The simulator is capable of handling different types of traffic injection process. Messages can be injected by each IP into the network following different stochastic distributions. In our experiments the traffic injected by the functional IP blocks followed self-similar distributions [20]. This type of traffic has been observed in the bursty traffic typical of on-chip modules in MPEG-2 video applications [21], as well as various other networking applications [22]. It has been shown to closely model real traffic.

## VII. TIMING CHARACTERISTICS OF NOC COMMUNICATION INFRASTRUCTURES

The exchange of data among the constituent blocks in a SOC is becoming an increasingly difficult task because of growing system size and non-scalable global wire delay. To cope with

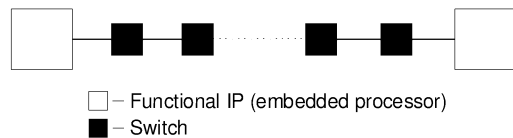


Fig. 7. Data transfer in NOC fabric.

these issues, designers must divide the communication medium into multiple pipelined stages, with the delay in each stage comparable to the clock-cycle budget. In a NOC, the inter-switch wire segments, along with the switch blocks, constitute a pipelined communication medium as shown in Fig. 7.

In any NOC between a source and destination pair there is a path consisting of multiple switch blocks involving several interswitch and intra-switch stages. The number of intra-switch stages can vary with the design style and the features incorporated within the switch blocks. It may consist of a single stage for a low-latency switch design or may be deeply pipelined [23], [24]. In the best case we need at least one intra and one interswitch stage [23]. The codec blocks might be considered as additional pipelined stages within a switch. If the delay of the codec blocks can be constrained within the one clock cycle limit then the pipelined nature of the communication will be maintained, though it will increase the overall message latency. However, there is an increasing drive in the NOC research community for design of low-latency NOCs adopting numerous techniques both at the routing as well as network interface (NI) level [25], [26]. Due to the crosstalk avoidance characteristic of the joint codes introduced in this work the crosstalk induced bus delay (CIBD) [12] of the interswitch wire segments will decrease. On the other hand the codecs will introduce additional delay requiring an elaborate analysis of the total timing overhead.

### VIII. EXPERIMENTAL RESULTS

In order to characterize the performance of the proposed coding schemes in NOC communication infrastructures, we considered a system consisting of 64 IP blocks and mapped them onto mesh, folded torus, and butterfly-fat-tree (BFT) based NOC architectures as shown in Fig. 8. We assumed the NOC to be spread over a die size of 20 mm  $\times$  20 mm. We compared the performance of the JTEC and the JTEC-SQED schemes with the already proposed schemes like ED, DR, DAP, BSC, and MDR. Since DR, DAP, MDR, and BSC are all joint crosstalk avoidance and single-error correction codes their performance is very similar and hence we have shown only one representative scheme namely DAP/DR for the sake of comparison. We also considered the performance of the joint crosstalk avoidance and double error correction code (CADEC) in this analysis. The routing mechanism used in the simulations depends on the particular network architecture adopted. For the Mesh and Folded Torus architectures *e-cube* (dimension order) routing was used whereas, for the BFT architecture, least common ancestor (LCA) routing methodology was adopted [27]. The particular switch architecture adopted [27] had three functional stages, namely, input arbitration, routing/switch traversal and output arbitration. The input and output ports

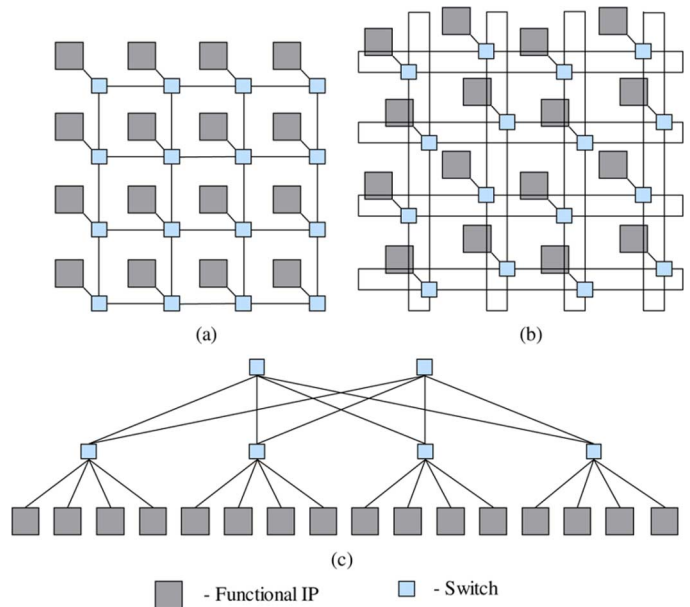


Fig. 8. NOC architectures: (a) mesh; (b) folded-torus; and (c) butterfly-fat-tree.

have four virtual channels, each having buffer depth of 2 flits. The pipelined data path of a flit through this switch architecture along with the encoder and decoder blocks is shown in Fig. 9. The energy dissipations as functions of injection load are plotted for each of the three NOC architectures mentioned above. The injection load is measured as the number of flits injected by each IP core into the network in each cycle. The energy dissipation profiles give the energy dissipated by all messages in the NOC per simulation cycle.

Simulations were performed using 90 nm standard cell libraries from CMP [28]. The clock cycle was assumed to be 600 ps, which is typical for this process [29]. The energy dissipation of each interswitch wire segment is a function of  $\lambda$ , the ratio of the coupling capacitance to the bulk capacitance. For a given interconnect geometry, the values of  $\lambda$  depend on the metal coverage in upper and lower metal layers. At the 90-nm technology node, the two extreme values of  $\lambda$  are 1 and 6 respectively [30]. A large set of data patterns were fed into the gate-level netlists of the switch blocks and codecs and by running Synopsys Prime Power their energy dissipation was obtained.

All the schemes have different number of bits in the encoded flit. A fair comparison in terms of energy savings demands that the redundant wires be also taken into account while comparing the energy dissipation profiles. The metric used for comparison thus takes into the account the savings in energy due to the reduced crosstalk and reduced voltage level on the wires, the additional energy dissipated by the codecs, the extra redundant wires and the interface circuitry used to achieve reduced voltage swing on the interconnect. Energy dissipated by the retransmission buffers and control signals requesting retransmissions for the ED and JTEC-SQED schemes are also considered. An uncoded 32-bit wide flit is considered as the standard for comparison. Table I gives a split report on the energy dissipation of each component for the Mesh based NOC at network saturation. The switch energy reported in Table I consists of the contributions from all the stages. The switch blocks and the codecs are driven

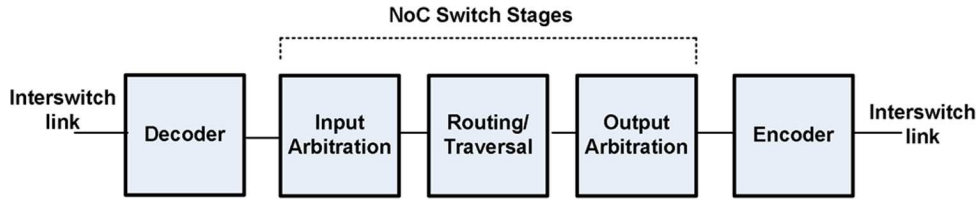
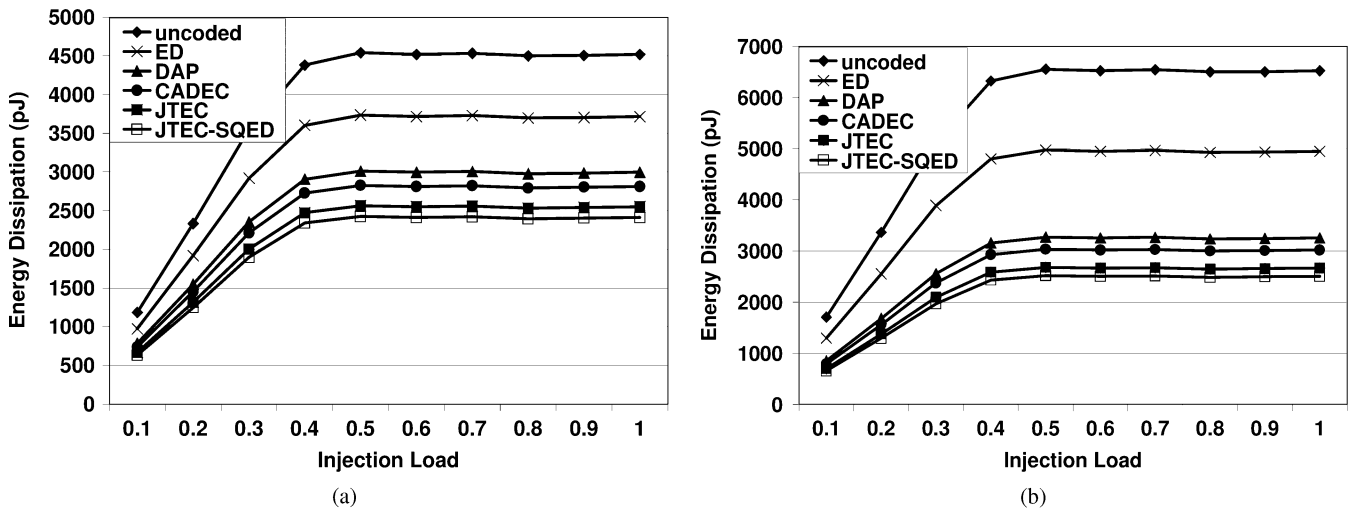


Fig. 9. Pipelined data path through an NOC switch including codecs.

TABLE I  
ENERGY DISSIPATION OF EACH COMPONENT IN A SINGLE SIMULATION CYCLE FOR A MESH-BASED NOC

Coding Scheme	$E_{Link}$ (pJ)	$E_{codec}$ (pJ)	$E_{switch}$ (pJ)	$E_{interface}$ (pJ)	$E_{NoC/cycle}$ (nJ)	$E_{savings/cycle}$ (nJ)	Percentage Savings (%)
Uncoded	17.75	0	40	0	4.52	0	0
Spacing	12.78	0	40	0	3.20	1.32	29.2
ED	10.93	2.38	40	0.11	3.71	0.80	17.8
DAP	6.73	2.61	40	0.19	2.99	1.52	33.7
BSC	6.73	2.65	40	0.19	3.00	1.52	33.6
MDR	6.83	2.63	40	0.20	3.02	1.50	33.2
CADEC	5.35	2.96	40	0.23	2.81	1.71	37.8
JTEC	2.98	3.92	40	0.23	2.55	1.97	43.6
JTEC-SQED	2.37	3.79	40	0.23	2.41	2.11	46.6

Fig. 10. Energy Dissipation Profile for the Mesh based NOC at (a)  $\lambda = 1$  and (b)  $\lambda = 6$ .

with the nominal  $V_{dd}$  of 1 V, whereas the interswitch wires are driven by the lowered voltage swing as explained in Section V. To achieve the lower voltage swing on the interconnects the level converting register (LCR) [31] interface was incorporated in the switch blocks. This particular interface circuitry enables a quadratic reduction in the energy dissipation on the interswitch wires due to the use of NMOS only push-pull drivers driven by a lower voltage signal [31]. The energy dissipation overheads due to the interface circuitry for each scheme are also shown in Table I. As the coding schemes under consideration have different number of encoded bits in a flit their interface energy values also vary. The total NOC energy dissipation in a single clock cycle can be obtained using (11) and (12). Table I also includes the energy dissipation when the interswitch wires are spaced by twice the distance compared to the uncoded case. Due to reduction in crosstalk capacitance by the same amount as the joint codes and no codec overhead it dissipates less energy than the ED scheme, which is a sole error detection code without

any crosstalk avoidance. However, as the joint codes can also reduce voltage swing on the wires they consume less energy compared to the spacing approach. Spacing reduces the interswitch wire delay by the same amount as the joint codes due to similar crosstalk avoidance properties. But as a result of higher energy dissipation and absence of any error correction capabilities it is not considered in the following analysis.

It may be noted that as shown in (13) the absolute value of the savings in energy dissipation remains unchanged irrespective of the particular switch implementation, however the percentage savings over the uncoded baseline case depends on the energy dissipation by the switch and hence may vary with the particular implementation style. Energy dissipation of NOC switches are shown to vary widely [9], [26], [32]. However irrespective of the particular switch design the overall savings in energy remains unchanged due to coding.

Fig. 10(a) and (b) show the energy dissipation profile per cycle for all the coding schemes (ED, DAP, CADEC, JTEC,

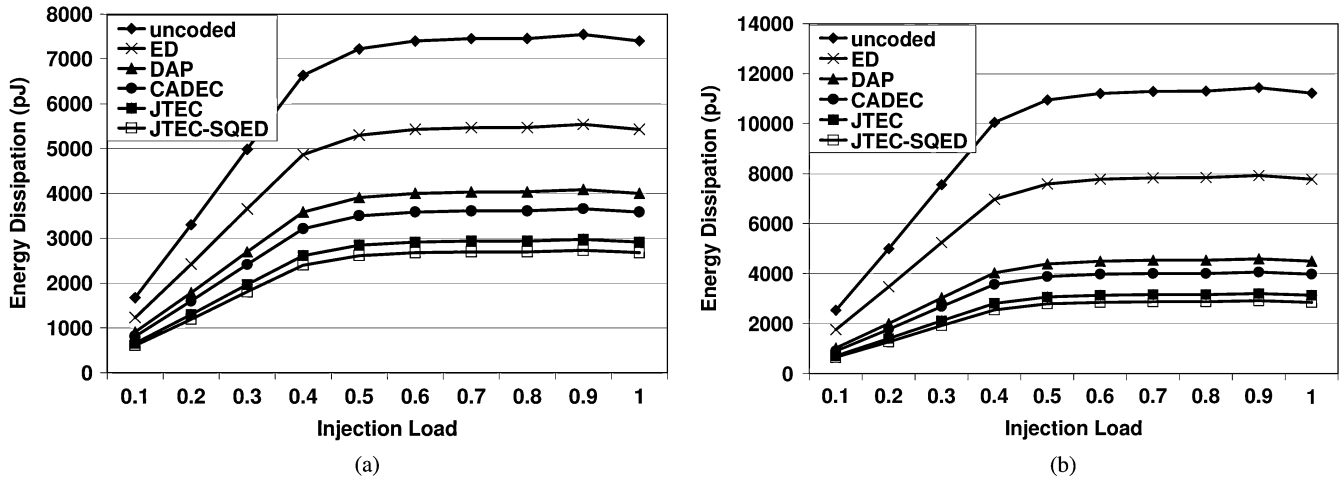


Fig. 11. Energy dissipation profile of a folded-torus-based NOC at (a)  $\lambda = 1$  and (b)  $\lambda = 6$ .

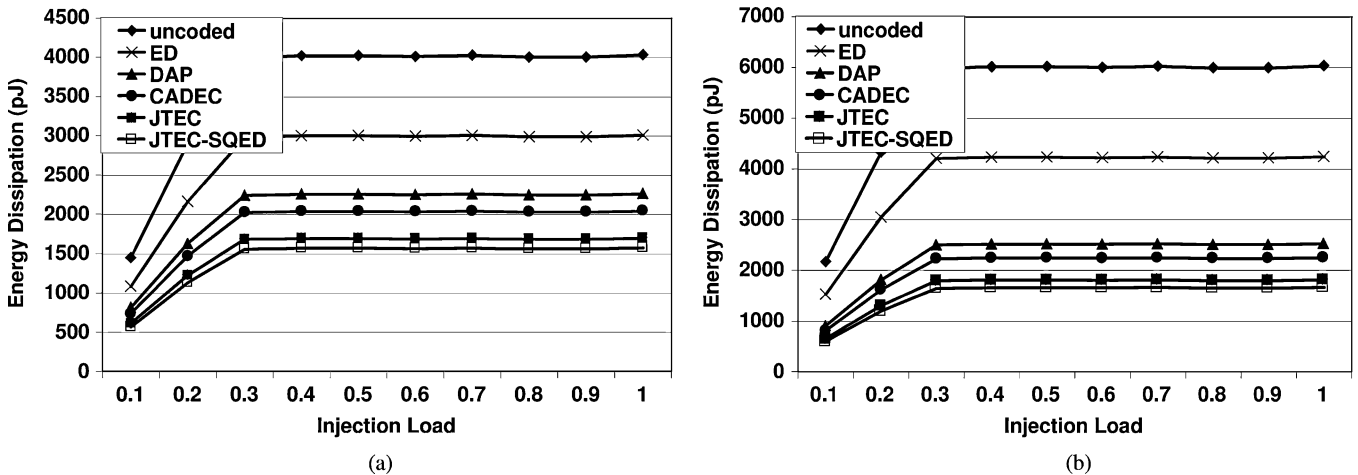


Fig. 12. Energy dissipation profile of a butterfly-fat-tree-based NOC at (a)  $\lambda = 1$  and (b)  $\lambda = 6$ .

and JTEC-SQED) for  $\lambda = 1$  and  $\lambda = 6$  respectively, in a Mesh-based NOC architecture. The channel BER is assumed to be  $10^{-20}$  [10] in these simulations.

Figs. 11(a) and (b) show the energy dissipation profile with  $\lambda = 1$  and  $\lambda = 6$  respectively for a folded-torus based NOC fabric. Fig. 12(a) and (b) show the energy dissipation profile for a butterfly-fat-tree architecture for the same two extreme values of  $\lambda$ . The energy expenditure per cycle is least in the case of JTEC-SQED, followed by JTEC, as those can reduce the voltage swing more than any of the other schemes due to their quadruple-error-detection and triple-error-correction capability, as discussed in Section V. In addition to this, the joint codes (DAP, CADEC, JTEC and JTEC-SQED) also reduce the effective mutual switching capacitances on the interswitch wire segments, which is another contributing factor in lowering the energy dissipation. The reduction in effective switching capacitance happens only when crosstalk is avoided but not in the ED scheme, which uses Hamming code and hence does not address crosstalk. Thus among all the coding schemes the maximum energy dissipation corresponds to ED.

The energy savings depend on the length of the interswitch wire segments as the savings is only along the wires. Consequently, the longer the interswitch wires, the higher the savings

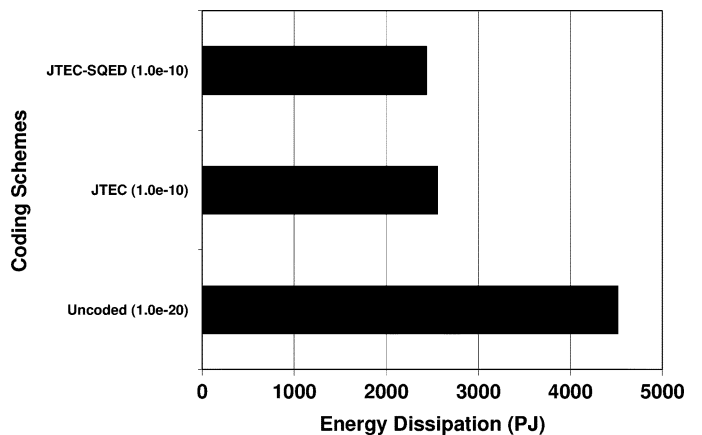


Fig. 13. Energy dissipation at different word error rates.

in energy due to the implementation of coding. Hence in architectures with longer interconnects like folded-torus and BFT the savings is more than that in Mesh.

The energy dissipation characteristics for JTEC and JTEC-SQED are studied over a wide range of possible word error rates. Fig. 13 shows that the energy dissipation of

the Mesh-based NOC by incorporating JTEC and JTEC–SQED for a higher word error rate of  $10^{-10}$  is still less than the energy dissipation of an uncoded system at a much lower word error rate of  $10^{-20}$ . Though for an increased word error rate the reduction in voltage swing is less, it is still enough to give substantial savings in energy dissipation. The energy dissipation numbers quoted in Fig. 13 are at network saturation for  $\lambda = 6$ .

This shows that even with higher error rates, implementation of channel coding scheme on NOC interconnects reduce energy dissipation compared to an uncoded case with a lower error rate.

## IX. TIMING CHARACTERISTICS

As discussed in Section VII, introduction of the joint codes affects the timing characteristics of the NOC. In the following subsections we present an elaborate analysis of the interswitch wire and codec delays influencing the performance of the NOC communication fabric.

### A. Inter-Switch Wire Delay

Due to crosstalk among adjacent wires the delay of data propagation through an interconnect increases. This Crosstalk Induced Bus Delay (CIBD) [12] is a function of the worst case crosstalk capacitance between the adjacent wires and it depends on the correlation between transmitted signals. More correlated signals incur less propagation delay compared to completely uncorrelated signals. For an uncoded interconnect the data patterns can be generally considered uncorrelated and consequently it is possible to have the worst case switching scenario, where a data pattern can have a 101 to 010 transition or vice versa. Due to opposite transitions in neighbors on both sides of the victim wire the coupling capacitance of the victim increases by twice for each neighbor and hence it becomes  $(1+4\lambda)C_L$  [12] where  $C_L$  is the load capacitance of the wire including self-capacitance and  $\lambda$  is the ratio of the coupling capacitance to the bulk capacitance as mentioned earlier. The CIBD for such a situation becomes  $(1+4\lambda)\tau_0$ , where,  $\tau_0$  is the delay of a single individual wire without any coupling.

When coding is employed, the correlation between the transmitted data depends on the particular error control code used. For the ED scheme, which is implemented using a Hamming code there are no inherent crosstalk avoidance characteristics and hence in general the coded data is uncorrelated. Consequently the worst case transition of two neighbors transitioning in opposite directions cannot be avoided and hence the CIBD of the interswitch wires for the ED scheme is  $(1+4\lambda)\tau_0$ .

For the DAP, CADEC, JTEC and JTEC–SQED schemes all the individual bits are duplicated and hence a 101 or 010 pattern can never occur at all in any code word. This enhances the correlation between transmitted signals. As a result the worst case coupling in the case of such coding schemes reduces to  $(1+2\lambda)C_L$ . The worst case CIBD thus becomes  $(1+2\lambda)\tau_0$ . Table II shows the delays incurred by the flits, while traversing the interswitch wire segments for different coding schemes for  $\lambda = 6$ . These delay figures include the propagation and setup times of the sending and receiving modules and are obtained using HSPICE. It should be noted that for Mesh and Folded Torus architectures all the interswitch wire lengths are the same and hence their delays are equal and less than the clock cycle budget.

TABLE II  
INTER-SWITCH WIRE DELAY

Coding Scheme	Architecture	Length (mm)	Delay (ps)
Uncoded/ED	Mesh	2.86	243
	Folded Torus	5.72	612
	BFT_3_2*	10	1620
	BFT_2_1**	5	495
Spacing/DAP/CADEC/JTEC/JTEC-SQED	Mesh	2.86	184
	Folded Torus	5.72	375
	BFT_3_2*	10	900
	BFT_2_1**	5	315

\* Inter-switch wire spanning levels 2 & 3

\*\* Inter-switch wire spanning levels 1 & 2

By contrast, in the BFT architecture the wire lengths vary with the level of the hierarchy in the tree. As a result the wire delays also vary with the level. For a 64-IP system the BFT-based NOC will have  $\log_4 64 = 3$  levels of switches. Specifically the delay of the top level interswitch wire is high, necessitating use of multiple stages. As shown in Table II, as the transmitted signals for DAP, CADEC, JTEC and JTEC–SQED schemes are more correlated than those for Uncoded and ED schemes, they incur less delay in interswitch wire traversal. Another point worth noting is that DAP, CADEC, JTEC and JTEC–SQED reduce the wire capacitance by the same amount and hence they incur identical interswitch delays. As the delays along all the interswitch links after coding are less than the clock period of 600 ps, buffer insertion is not necessary except in the BFT top level link where two stages in link traversal is assumed.

### B. Codec Delay

Through RTL design followed by post synthesis place and route using 90-nm technology standard cell libraries from CMP [28] we obtain the delays along the critical paths of each encoder and decoder for all the coding schemes. The delay values corresponding to all the coding schemes are shown in Table III. It also includes the delay added by the low swing interface circuitry. It is evident that all the coding schemes achieve the target delay values within the limit of one clock cycle. Consequently, the pipelined nature of communication is maintained, however for all the coding schemes the combined delay of the codec blocks and interswitch wires is more than the uncoded interswitch wire delay with the exception of the top most stage of the BFT architecture in presence of JTEC and JTEC–SQED. Hence, there will be a corresponding latency penalty compared to the uncoded case. However, use of a tree-based implementation of XOR gates rather than a linear cascade in the codecs of the JTEC and JTEC–SQED schemes in the post synthesis place and routed design along with the optimization techniques discussed in Section III-C the delays of their encoder and decoder are significantly lower. Figs. 14(a) and (b) show the penalties in the average message latency for the different coding schemes in comparison with the baseline uncoded case for the Mesh and BFT architectures. As JTEC and JTEC–SQED have very similar delay overheads only one is shown in Fig. 14, for clarity. In the BFT architecture the top most interswitch wire is so long that it incurs a significantly higher delay in the uncoded situation. This delay is so high, that in presence of coding the latency penalty arising out of this stage is small, whereas for JTEC and

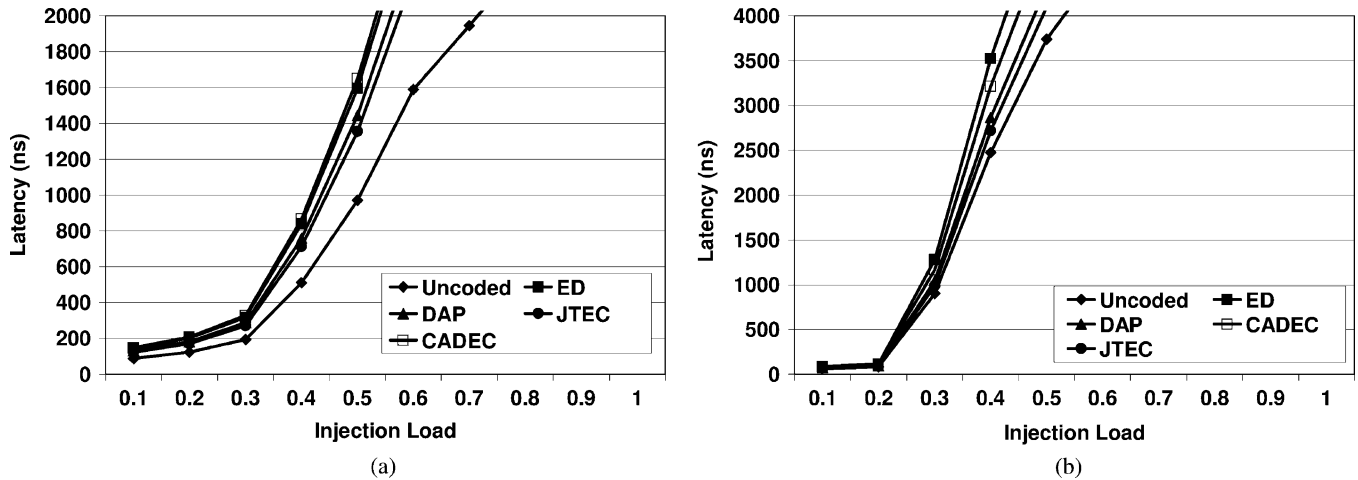


Fig. 14. Variation of average message latency with injection load for (a) mesh and (b) BFT.

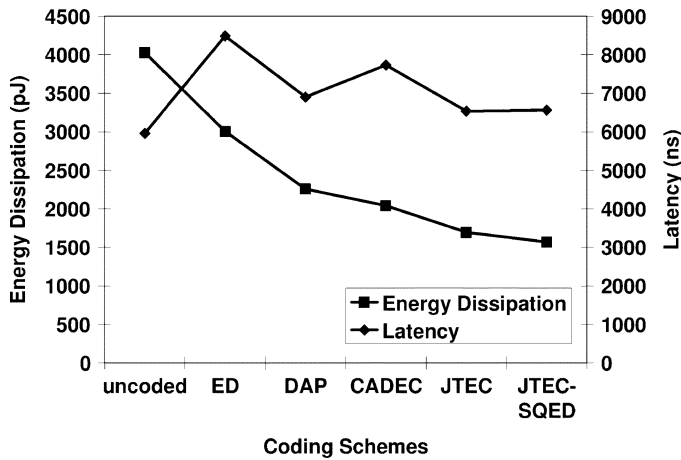


Fig. 15. Gains in energy dissipation against increase in average message latency for BFT architecture.

TABLE III  
CRITICAL PATH DELAY FOR EACH CODING SCHEME

Coding Scheme	Codec Delay (ps)	
	Encoder	Decoder
ED	410	520
DAP	290	475
CADEC	525	545
JTEC-unoptimized	525	675
JTEC	190	440
JTEC-SQED	190	450

JTEC-SQED there are gains. This reduces the overall latency penalty in BFT architecture compared to a Mesh, which has much smaller interswitch wires. From Fig. 14, it is evident that the JTEC and JTEC-SQED schemes incur less overhead in latency compared to other existing coding schemes. Fig. 15 shows the tradeoffs between gains in energy dissipation and the associated penalty in average message latency for a BFT-based NOC. It can be inferred from Fig. 15 that JTEC and JTEC-SQED are able to reduce both latency and energy dissipation compared to the other existing joint codes.

It can be noted from Tables II and III that the delay of each encoder and decoder as well as the interswitch links is less than

TABLE IV  
AREA OVERHEAD OF THE CODING SCHEMES

Coding Scheme	Area (2-input NAND gate)
ED	447
DAP	396
CADEC	1145
JTEC	1459
JTEC-SQED	1675

the clock cycle budget of 600 ps. The only exception to this is the longest link in the BFT architecture where extra pipelined stages is assumed as mentioned earlier. However, with coding, the delay on this segment is only reduced and hence the same pipelining technique will alleviate the issue of the delay on this link. Thus implementation of the coding schemes still enable an operating frequency of 1.67 GHz (time period of 600 ps) but incur penalties in latency as shown in Fig. 14.

## X. AREA OVERHEAD

For the sake of complete comparison, we also report the silicon area required by the codec blocks for each of the coding schemes. The silicon area consumed by each codec per NOC switch port is shown in Table IV. The area figures are expressed in units of a minimum sized 2-input NAND gate with a fan-out of 4 (FO4) loading.

In our implementation the switches along with the network interface (NI) consist of approximately 30 K NAND gates. Consequently, considering contribution from all the switch ports the area overhead due to the proposed codes may be upto 22% of the overall switch area.

## XI. CONCLUSION

Network-on-chip (NOC) has emerged as a revolutionary methodology for integrating a very high number of intellectual property (IP) cores in a single chip. With technology scaling NOC architectures are increasingly exposed to multiple sources of transient errors. By incorporating error-control coding, it is possible to protect the NOC fabrics from different transient malfunctions and at the same time lower the energy dissipation in communication. In this paper we have proposed design of novel joint crosstalk avoidance and simultaneous

triple-error-correction and quadruple-error-detection codes, namely JTEC and JTEC–SQED respectively. Performances of these codes in different common NOC architectures are evaluated. JTEC and JTEC–SQED are much more energy efficient in all the architectures investigated here with lower latency compared to the existing coding schemes, though they can tolerate higher transient error rates.

#### APPENDIX I

*Theorem 1:* The shortened Hsiao SEC–DED code, formed by dropping a single parity bit from the standard Hsiao SEC–DED code has single-error correction capability.

*Proof:* Shortening the Hsiao code implies removal of a single column and a single row from the  $H$ -matrix. The characteristic of the Hsiao  $H$ -matrix is that all columns have odd weight and no 3 columns add to zero. Removing a column from the  $H$ -matrix does not alter this property of the  $H$ -matrix in any way. Now, if after removal of the row, no 2 columns add up to zero then the minimum Hamming distance of the code will be 3 enabling single-error correction. Let us consider two arbitrary columns  $h_i$  and  $h_j$  and show that even after removal of a row they can never add to zero. Two situations are possible. In the first case both the columns had either “0” or “1” entries on the row that was removed. In this case the columns will not add to zero after the row is removed as this would mean they were identical even before the removal. The second case is when exactly 1 of the columns had a “1” on the removed row. In that case the column which lost a “1” will now have even weight whereas the other column will have odd weight. Hence, they can never add to “0”. Thus no 2 columns of the  $H$ -matrix of the shortened Hsiao code can add to zero making the minimum Hamming distance equal 3 and hence enabling single-error correction.

#### APPENDIX II

*Theorem 2:* A triple error pattern will always manifest itself as an odd-weight syndrome of the Hsiao SEC–DED code.

*Proof:* A single error is identified by an odd weight syndrome and a double error by an even weight syndrome in a SEC–DED code. The syndrome is formed essentially by adding the columns of the  $H$ -matrix corresponding to the bits in error due to relation shown below

$$S = eH^T \quad (16)$$

where  $S$  is syndrome and  $e$  is the error pattern as row vectors.

A triple error pattern would result in the syndrome equaling the sum of three distinct odd weight columns. The syndrome must then have odd weight since the modulo-2 sum of any three odd weight binary  $n$ -tuples also has odd weight.

#### 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.*, Feb. 2007, pp. 98–589.
- [3] P. Magarshack and P. G. Paulin, “System-on-chip beyond the nanometer wall,” in *Proc. 40th Design Autom. Conf. (DAC)*, 2003, pp. 419–424.
- [4] R. Ho, K. W. Mai, and M. A. Horowitz, “The future of wires,” *Proc. IEEE*, vol. 89, no. 4, pp. 490–504, Apr. 2001.
- [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] N. R. Shanbhag and M. Zhang, “Soft-error-rate-analysis (SERA) methodology,” *IEEE Trans. Comput.-Aided Design Circuits Syst.*, vol. 25, no. 10, pp. 2140–2155, Oct. 2006.
- [7] P. P. Pande, H. Zhu, A. Ganguly, and C. Grecu, “Crosstalk-aware energy reduction in NOC communication fabrics,” in *Proc. IEEE Int. SOC Conf. (SOCC)*, 2006, pp. 225–228.
- [8] 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.
- [9] S. Murali, G. De Micheli, L. Benini, T. Theodorides, N. Vijaykrishnan, and M. Irwin, “Analysis of error recovery schemes for networks on chips,” *IEEE Design Test Comput.*, vol. 22, no. 5, pp. 434–442, Sep. 2005.
- [10] S. R. Sridhara and N. R. Shanbhag, “Coding for system-on-chip networks: A unified framework,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 13, no. 6, pp. 655–667, Jun. 2005.
- [11] P. P. Sotiriadis and A. P. Chandrakasan, “A bus energy model for deep submicron technology,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 10, no. 3, pp. 341–350, Jun. 2002.
- [12] D. Rossi, C. Metra, A. K. Nieuwland, and A. Katoch, “Exploiting ECC redundancy to minimize crosstalk impact,” *IEEE Design Test Comput.*, vol. 22, no. 1, pp. 59–70, Jan. 2005.
- [13] K. N. Patel and I. L. Markov, “Error-correction and crosstalk avoidance in DSM busses,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 10, pp. 1076–1080, Oct. 2004.
- [14] D. Rossi, C. Metra, A. K. Nieuwland, and A. Katoch, “New ECC for crosstalk effect minimization,” *IEEE Design Test Comput.*, vol. 22, no. 4, pp. 340–348, Jul.–Aug. 2005.
- [15] D. Rossi, P. Angelini, and C. Metra, “Configurable error control scheme for NOC signal integrity,” in *Proc. IEEE Int. On-Line Test Symp.*, 2007, pp. 43–48.
- [16] M. Y. Hsiao, “A class of optimal minimum odd-weight-column SEC–DED codes,” *IBM J. Res. Dev.*, vol. 14, no. 4, pp. 395–401, 1970.
- [17] A. Ganguly, P. P. Pande, B. Belzer, and C. Grecu, “Addressing signal integrity in networks on chip interconnects through crosstalk-aware double error correction coding,” in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI (ISVLSI)*, 2007, pp. 317–324.
- [18] S. Lin and D. J. Costello, *Error Control Coding: Fundamentals and Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1983.
- [19] C. Grecu, P. P. Pande, A. Ivanov, and R. Saleh, “Timing analysis of network on chip architectures for MP-SOC platforms,” *Microelectron. J.*, vol. 36, no. 9, pp. 833–845, 2005.
- [20] K. Park and W. Willinger, *Self-Similar Network Traffic and Performance Evaluation*. New York: Wiley, 2000.
- [21] D. R. Avresky, V. Shubranov, R. Horst, and P. Mehra, “Performance evaluation of the servernet<sup>R</sup> SAN under self-similar traffic,” in *Proc. 13th Int. 10th Symp. Parallel Distrib. Process.*, 1999, pp. 143–147.
- [22] G. V. Varatkar and R. Marculescu, “On-chip traffic modeling and synthesis for MPEG-2 video applications,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 8, no. 3, pp. 335–339, Jun. 2000.
- [23] R. Mullins, A. West, and S. Moore, “Low-latency virtual-channel routers for on-chip networks,” in *Proc. 31st Annu. Int. Symp. Comput. Archit. (ISCA)*, 2004, pp. 188–197.
- [24] L. Benini and D. Bertozzi, “Xpipes: A network-on-chip architecture for gigascale systems-on-chip,” *IEEE Circuits Syst. Mag.*, vol. 4, no. 2, pp. 18–31, Apr.–Jun. 2004.
- [25] D. Park, “A distributed multi-point network interface for low-latency, deadlock-free on-chip interconnects,” in *Proc. 1st Int. Conf. Nano-Networks and Workshops*, 2006, pp. 1–6.
- [26] A. Kumar, “A 4.6 Tbits/s 3.6 GHz single-cycle NOC router with a novel switch allocator in 65 nm CMOS,” in *Proc. IEEE Int. Conf. Comput. Design (ICCD)*, 2007, pp. 63–70.
- [27] P. P. Pande, C. Grecu, M. Jones, A. Ivanov, and R. Saleh, “Performance evaluation and design tradeoffs for network on chip interconnect architectures,” *IEEE Trans. Comput.*, vol. 54, no. 8, pp. 1025–1040, Aug. 2005.
- [28] Circuits Multi-Projects, Grenoble, France, “CMP 90 nm Technology Library,” 2008. [Online]. Available: <http://cmp.imag.fr/products/ic/?p=STCMOS090>

- [29] International Technology Roadmap for Semiconductors, 2007. [Online]. Available: [http://www.itrs.net/Links/2007ITRS/2007\\_Chapters/2007\\_SystemDrivers.pdf](http://www.itrs.net/Links/2007ITRS/2007_Chapters/2007_SystemDrivers.pdf)
- [30] D. Sylvester and C. Hu, "Analytical modeling and characterization of deep-submicrometer interconnect," *Proc. IEEE*, vol. 89, no. 5, pp. 634–664, May 2001.
- [31] H. Zhang, V. George, and J. Rabaey, "Low-swing on-chip signaling techniques: Effectiveness and robustness," *IEEE Trans Very Large Scale Integr. (VLSI) Syst.*, vol. 8, no. 3, pp. 264–272, Jun. 2000.
- [32] D. Milojevic, I. Montperrus, and D. Verkest, "Power dissipation of the network-on-chip in a system-on-chip for MPEG-4 video encoding," in *Proc. IEEE Asian Solid-State Circuits Conf.*, 2007, pp. 392–395.



**Amlan Ganguly** (S'07) received the B.Tech (Hons.) degree in electronics and electrical communication engineering from the Indian Institute of Technology, Kharagpur, India. He is currently working toward the Doctoral degree at the School of Electrical Engineering and Computer Science, Washington State University (WSU), Pullman.

After a brief stint of five months with Intel India Development Centre, Bangalore, India, he joined the graduate program at WSU. His research interests include design of fault tolerant interconnection infrastructures for multi-processor SoC platforms and novel architectures for on-chip networks



**Partha Pratim Pande** (M'05) received the M.S. degree in computer science from the National University of Singapore, Singapore, in 2002, and the Ph.D. degree in electrical and computer engineering from the University of British Columbia, Vancouver, BC, Canada, in 2005.

He is currently an Assistant Professor with the School of Electrical Engineering and Computer Science, Washington State University, Pullman. His current research interests include the areas of design and test of networks on chip, fault tolerance, and reliability of multiprocessor SOC (MP-SOC) platforms, 3-D integration, and on-chip wireless communication network.

Dr. Pande is a member of the Program committees of different International Conferences, like IOLTS, ATS, MWSCAS, and DELTA.



**Benjamin Belzer** (S'93–M'96) received the B.A. degree in physics from the University of California at San Diego, San Diego, in 1982, and the Ph.D. degree in electrical engineering from the University of California at Los Angeles, Los Angeles, in 1996.

From 1981 to 1991, he was a Software Engineer for Beckman Instruments, Hughes Aircraft, Northrop Corporation, and Source Scientific in Southern California, and Develco, Inc. in Northern California. Since 1996, he has been on the faculty of the School of Electrical Engineering and Computer Science, WSU, Pullman, where he is currently an Associate Professor. His current research interests include coding for networks on chip, iterative detection and equalization, coded modulation for wireless communications, and combined source.