Power Dissipation Reduction in NOC links by Enhanced Data Encoding Schemes

Ashok kumar Illa[1], Mr. M.Damodhar rao [2]

Department Of DECS, Gudlavalleru Engineering College

Assistant professor Department Of ECE, Gudlavalleru Engineering College

Abstract- **As technology shrinks, the power dissipated by the links of a network-on-chip (NoC) starts to compete with the power dissipated by the other elements of the communication subsystem, namely, the routers and the network interfaces (NIs). In this paper, we present a set of data encoding schemes aimed at reducing the power dissipated by the links of an NoC. The proposed schemes are general and transparent with respect to the underlying NoC fabric (i.e., their application does not require any modification of the routers and link architecture). Experiments carried out on both synthetic and real traffic scenarios show the effectiveness of the proposed schemes, which allow to save power dissipation and energy consumption without any significant performance degradation and with less than area overhead in the NI**.

Index Terms- **NoC, NI, Data Encoding automation**

I.INTRODUCTION

Shifting from a silicon technology node to the next one results in faster and more power efficient gates but slower and more power hungry wires [1]. In fact, more than 50% of the total dynamic power is dissipated in interconnects in current processors, and this is expected to rise to 65%–80% over the next several years [2]. Global interconnect length does not scale with smaller transistors and local wires. Chip size remains relatively constant because the chip function continues to increase and RC delay increases exponentially. At 32/28 nm, for instance, the RC delay in a 1-mm global wire at the minimum pitch is 25× higher than the intrinsic delay of a two-input NAND fanout of 5 [1]. If the raw computation horsepower seems to be unlimited, thanks to the ability of instancing more and more cores in a single silicon die, scalability issues, due to the need of making efficient and reliable communication between the increasing. number of cores, become the real problem [3]. The network on- chip (NoC) design paradigm [4] is recognized as the most viable way to tackle with scalability and variability issues that characterize the ultra-deep submicron meter era. Nowadays, the on-chip communication issues are as relevant as, and in some cases more relevant than, the computation related issues [4]. In fact, the communication subsystem increasingly impacts the traditional design objectives, including cost (i.e., silicon area), performance, power dissipation, energy consumption, reliability, etc. As technology shrinks, an ever more significant fraction of the total power budget of a complex many-core system-on-chip (SoC) is due to the communication subsystem. In this paper, we focus on techniques aimed at reducing the power dissipated by the network links. In fact, the power dissipated by the network links is as relevant as that dissipated by routers and network interfaces (NIs) and their contribution is expected to increase as technology scales [5]. In particular, we present a set of data encoding schemes operating at flit level and on an end-to-end basis, which allows us to minimize both the switching activity and the coupling switching activity on links of the routing paths traversed by the packets. The proposed encoding schemes, which are transparent with respect to the router implementation, are presented and discussed at both the algorithmic level and the architectural level, and assessed by means of simulation on synthetic and real traffic scenarios. The analysis takes into account several aspects and metrics of the design, including silicon area, power dissipation, and energy consumption. The results show that by using the proposed encoding schemes up to 51% of power and up to 14% of energy can be saved without any significant degradation in performance and with 15%

area overhead in the NI. The rest of this paper is organized as follows. We briefly discuss related works in Section II, while Section III presents an overview of the proposed data encoding schemes. The proposed data encoding schemes along with possible hardware implementations and their analysis are described in Section IV. In Section V, the results for the hardware overhead, power and energy savings, and performance reduction of the proposed data encoding schemes are compared with those of other approaches. Finally, this paper is concluded in Section VI.

II.OVERVIEW OF THE PROPOSED

The basic idea of the proposed approach is encoding the flits before they are injected into the network with the goal of minimizing the self-switching activity and the coupling switching activity in the links traversed by the flits. In fact, self-switching activity and coupling switching activity are responsible for link power dissipation. In this paper, we refer to the end-to-end scheme. This end-to-end encoding technique takes advantage Of the pipeline nature of the wormhole switching technique [4]. Note that since the same sequence of flits passes through all the links of the routing path, the encoding decision taken at the NI may provide the same power saving for all the links. For the proposed scheme, an encoder and a decoder block are added to the NI. Except for the header flit, the encoder encodes the outgoing flits of the packet such that the power dissipated by the inter-router point-to-point link is minimized [23].

III.PROPOSED DATAENCODING SCHEMES

In this section, we present the proposed encoding scheme whose goal is to reduce power dissipation by minimizing the coupling transition activities on the links of the interconnection network. Let us first describe the power model that contains different components of power dissipation of a link. The dynamic power dissipated by the interconnects and drivers is

$$P = [T_{0\to1}(C_s + C_l) + T_cC_c]V^2_{dd}F_{ck} \qquad (1)$$

Where $T_{0\to1}$ is the number of 0 → 1 transitions in the bus in two consecutive transmissions, T_c is the number of correlated switching between physically adjacent lines, C_s is the line to substrate capacitance, C_l is the load capacitance, C_c is the coupling capacitance, V_{dd} is the supply voltage, and F_{ck} is the clock frequency. One can classify four types of coupling transitions as described in [26]. A Type I transition occurs when one of the lines switches when the other remains unchanged. In a Type II transition, one line switches from low to high while the other makes transition from high to low. A Type III transition corresponds to the case where both lines switch simultaneously. Finally, in a Type IV transition both lines do not change. The effective switched capacitance varies from type to type, and hence, the coupling transition activity, T_c, is a weighted sum of different types of coupling transition contributions [26]. Therefore

$$T_c = K_1T_1 + K_2T_2 + K_3T_3 + K_4T_4 \qquad (2)$$

Where T_i is the average number of Type i transition and K_i is its corresponding weight. According to [26], we use $K_1 = 1$, $K_2 = 2$, and $K_3 = K_4 = 0$. The occurrence probability of Types I and II for a random set of data is 1/2 and 1/8, respectively. This leads to a higher value for K_1T_1 compared with K_2T_2 suggesting that minimizing the number of Type I transition may lead to a considerable power reduction. Using (2), one may express (1) as

$$P = [T_{0\to1}(C_s + C_l) + (T_1 + 2T_2)C_c]V^2_{dd}F_{ck}. \qquad (3)$$

According to [3], C_l can be neglected

$$P \propto T_{0\to1}C_s + (T_1 + 2T_2)C_c. \qquad (4)$$

Here, we calculate the occurrence probability for different types of transitions. Consider that flit $(t-1)$ and flit (t) refer to the previous flit which was transferred via the link and the flit which is about to pass through the link, respectively. We consider only two adjacent bits of the physical channel. Sixteen different combinations of these four bits could occur (Table I). Note that the first bit is the value of the generic ith line of the link, whereas the second bit represents the value of its $(i+1)$th line. The number of transitions for Types I, II, III, and IV are 8, 2, 2, and 4, respectively. For a random set of data, each of these sixteen transitions has the same probability. Therefore, the occurrence probability for Types I, II, III, and IV are 1/2, 1/8, 1/8, and 1/4, respectively. In the rest of this section, we present three data encoding schemes designed for reducing the dynamic power dissipation of the network links along with a possible hardware implementation of the decoder.

A. *SHEME I*

In scheme I, we focus on reducing the numbers of Type I transitions (by converting them to Types III and IV transitions) and Type II transitions (by converting them to Type I transition). The scheme compares the current data with the previous one to decide whether odd inversion or no inversion of the current data can lead to the link power reduction.

1)Power Model: If the flit is odd inverted before being transmitted, the dynamic power on the link is where *T*_0→1, *T* _1, *T* _ 2, *T* _ 3, and *T* _4, are the self-transition activity, and the coupling transition activity of Types I, II, III, and IV, respectively. Table I reports, for each transition, the relationship between the coupling transition activities of the flit when transmitted as is and when its bits are odd inverted. Data are organized as follows. The first bit is the value of the generic *i*th line of the link, whereas the second bit represents the value of its (*i* + 1)th line. For each partition, the first (second) line represents the values at time *t* − 1 (t). As Table I shows, if the flit is odd inverted, Types II, III, and IV transitions convert to Type I transitions. In the case of Type I transitions, the inversion leads to one of Types II, III, or Type IV transitions. In particular, the transitions indicated as 1 in the table convert to Types II, III, and IV transitions, respectively. Also, we have *T* _0→1= *T*0→0*(*odd*)* + *T*0→1*(*even*)* where odd/even refers to odd/even lines. Therefore, (5) can be expressed as

$$P \propto \left(T_{0\to 0(\text{odd})} + T_{0\to 1(\text{even})}\right) C_s$$
$$+[K_1\,(T_2+T_3+T_4)+K_2T_1^{***}+K_3T_1^{*}+K_4T_1^{**}]C_c. \quad (6)$$

Thus, if *P* > *P'*, it is convenient to odd invert the flit before transmission to reduce the link power dissipation. Using (4) and (6) and noting that Cc/Cs = 4 [26], we obtain the following odd invert condition. Also, since *T*0→1 = *T*0→1*(*odd*)* + *T*0→1*(*even*)*, one may write

$$\frac{1}{4}T_{0\to 1(\text{odd})} + T_1 + 2T_2 > \frac{1}{4}T_{0\to 0(\text{odd})} + T_2 + T_3$$
$$+T_4 + 2T_1^{***} \quad (7)$$

which is the exact condition to be used to decide whether the odd invert has to be performed. Since the terms *T*0→1*(*odd*)* and *T*0→0*(*odd*)* are weighted with a factor of 1/4, for link widths greater than 16 bits, the misprediction of the invert condition will not exceed 1.2% on average [23]. Thus, we can approximate the exact condition as

$$T1 + 2T2 > T2 + T3 + T4 + 2T\,^{***} \quad (8)$$

Of course, the use of the approximated odd invert condition reduces the effectiveness of the encoding scheme due to the error induced by the approximation but it simplifies the hardware implementation of encoder. Now, defining

$$Tx = T3 + T4 + T\,1^{***}$$

$$\&\ \ Ty = T2 + T1 - T\,1^{***} \quad (9)$$

one can rewrite (8) as

$$Ty > Tx\,. \quad (10)$$

Assuming the link width of *w* bits, the total transition between adjacent lines is *w* − 1, and hence

$$Ty + Tx = w - 1. \quad (11)$$

Thus, we can write (10) as

$$Ty > (w - 1)/2 \quad (12)$$

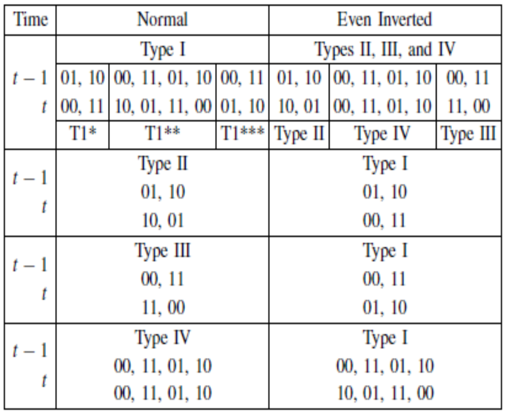
This presents the condition used to determine whether the odd inversion has to be performed or not.

TABLE I

EFFECT OF ODD INVERSION ON CHANGE OF TRANSITION TYPES

Time	Normal			Odd Inverted		
	Type I			Types II, III, and IV		
t − 1	00, 11	00, 11, 01, 10	01, 10	00, 11	00, 11, 01, 10	01, 10
t	10, 01	01, 10, 00, 11	11, 00	11, 00	00, 11, 01, 10	10, 01
	T1*	T1**	T1***	Type III	Type IV	Type II
t − 1 *t*	Type II 01, 10 10, 01			Type I 01, 10 11, 00		
t − 1 *t*	Type III 00, 11 11, 00			Type I 00, 11 10, 01		
t − 1 *t*	Type IV 00, 11, 01, 10 00, 11, 01, 10			Type I 00, 11, 01, 10 01, 10, 00, 11		

TABLE II
EFFECT OF EVEN INVERSION ON CHANGE OF TRANSITION TYPES



Time	Normal			Even Inverted		
	Type I			Types II, III, and IV		
$t-1$ t	01, 10 00, 11	00, 11, 01, 10 10, 01, 11, 00	00, 11 01, 10	01, 10 10, 01	00, 11, 01, 10 00, 11, 01, 10	00, 11 11, 00
	T1*	T1**	T1***	Type II	Type IV	Type III
$t-1$ t	Type II 01, 10 10, 01			Type I 01, 10 00, 11		
$t-1$ t	Type III 00, 11 11, 00			Type I 00, 11 01, 10		
$t-1$ t	Type IV 00, 11, 01, 10 00, 11, 01, 10			Type I 00, 11, 01, 10 10, 01, 11, 00		

B. *Scheme II*

In the proposed encoding scheme II, we make use of both odd (as discussed previously) and full inversion. The full inversion operation converts Type II transitions to Type IV transitions. The scheme compares the current data with the previous one to decide whether the odd, full, or no inversion of the current data can give rise to the link power reduction.

1) Power Model: Let us indicate with P, P’, and P’’ the power dissipated by the link when the flit is transmitted with no inversion, odd inversion, and full inversion, respectively. The odd inversion leads to power reduction when P’ < P”and P’< P. The power P’’ is given by [23]

$$P'' \propto T1 + 2T4^{**} \qquad (13)$$

Neglecting the self-switching activity, we obtain the condition

$P' < P'$ as [see (7) and (13)]

$$T2+T3+T4+2T\ \ 1^{***} < T1+2T4^{**} \qquad (14)$$

Therefore, using (9) and (11), we can write

$$2(T2-T4^{**})<2Ty-w+1. \qquad (15)$$

$$2(T2-T4^{**})<2Ty-w+1 Ty>(w-1) \qquad (16)$$

Similarly, the condition for the full inversion is obtained from $P'' < P$ and $P'' < P'$. The inequality $P'' < P$ is satisfied when [23]

$$T2 > T4^{**} \qquad (17)$$

Therefore, using (15) and (17), the full inversion condition is obtained as

$$2\,(T2 - T4^{**}) > 2Ty - w + 1\ \ T2 > T4^{**} \qquad (18)$$

When none of (16) or (18) is satisfied, no inversion will be performed.

C. *Scheme III*

In the proposed encoding Scheme III, we add even inversion to Scheme II. The reason is that odd inversion converts some of Type I ($T\ ^{**}{}_{*}\ 1$) transitions to Type II transitions. As can be observed from Table II, if the flit is even inverted, the transitions indicated as $T^{**}\,1\,/T^{**}\_1$ in the table are converted to Type IV/Type III transitions. Therefore, the even inversion may reduce the link power dissipation as well. The scheme compares the current data with the previous one to decide whether odd, even, full, or no inversion of the current data can give rise to the link power reduction.

1) Power Model: Let us indicate with P', P'', and P''' the power dissipated by the link when the flit is transmitted with no inversion, odd inversion, full inversion, and even inversion, respectively. Similar to the analysis given for Scheme I, we can approximate the condition $P''< P$ as

$$T1+2T2>T2+T3+T4+\ \ 2T\ 1^{*} \qquad (19)$$

Defining

$$Te = T2 + T1 - T\ 1^{*} \qquad (20)$$

We obtain the condition *P*“‘< *P* as

$$Te > (w-1)/2 \qquad (21)$$

Similar to the analysis given for scheme II, we can approximate the condition *P*“‘< *P’* as

$$T2+T3+T4+2T1^{*}<T2+T3+T4+2T1^{***} \qquad (22)$$

Using (9) and (20), we can rewrite (22) as

$$Te>Ty \qquad (23)$$

Also, we obtain the condition *P*“‘< *P’’* as [see (13) and (19)]

$$T2 + T3 + T4 + 2T\ 1^{*}< T1 + 2T\ 4^{**} \qquad (24)$$

Now, define

$Tr=T3+T4+T1^*$ & $Te=T2 + T1 – T1^*$ (25)

Assuming the link width of w bits, the total transition between adjacent lines is $w - 1$, and hence

$Te + Tr = w - 1.$ (26)

Using (26), we can rewrite (24) as

$2(T2–T4^{**}) <2Te-w$ (27)

The even inversion leads to power reduction when P"'$< P$, P"'$< P'$, and P"'$< P''$. Based on (21), (23), and (27), we obtain

$Te>(w - 1)/2Te>Ty$, $2(T2 – T4^{**}) < 2Te - w+1$(28)

The full inversion leads to power reduction when P'' < P, P'' < P', and P'' < P"'Therefore, using (18) and (27), the full inversion condition is obtained as 2

$(T2 - T\ 4^{**}) > 2Ty - w + 1$, $(T2 > T\ 4^{**})2\ (T2 - T\ 4^{**}) > 2Te - w + 1$ (29)

Similarly, the condition for the odd inversion is obtained from P' < P, P' < P'', and P' < P''' Based on (16) and (23), the odd inversion condition is satisfied when

$2\ (T2 - T\ 4^{**}) < 2Ty - w + 1$, $Ty >(w - 1)2Te < Ty$

II. IMPLEMENTATION

A. *Scheme I*

1) *Implementation:*

The proposed encoding architecture, which is based on the odd invert condition defined by (12), is shown in Fig. 1. We consider a link width of w bits. If no encoding is used, the body flits are grouped in w bits by the NI and are transmitted via the link. In our approach, one bit of the link is used for the inversion bit, which indicates if the flit traversing the link has been inverted or not. More specifically, the NI packs the body flits in $w - 1$ bits [Fig. 1(a)]. The encoding logic E, which is integrated into the NI, is responsible for deciding if the inversion should take place and performing the inversion if needed. The generic block diagram shown in Fig. 1(a) is the same for all three encoding schemes proposed in this paper and only the block E is different for the schemes. To make the decision, the previously encoded flit is compared with the current flit being transmitted. This latter, whose w bits are the concatenation of $w - 1$ payload bits and a "0" bit, represents the first input of the encoder, while the previous encoded flit represents the second input of the encoder [Fig. 1(b)]. The $w - 1$ bits of the incoming (previous encoded) body flit are indicated by Xi (Yi), $i = 0, 1, . . . ,w - 2$. The wth bit of the previously encoded body flit is indicated by inv which shows if it was inverted (inv = 1) or left as it was (inv = 0). In the encoding logic, each Ty block takes the two adjacent bits of the input flits (e.g., $X1X2Y1Y2$, $X2X3Y2Y3$, $X3X4Y3Y4$, etc.) and sets its output to "1" if any of the transition types of Ty is detected. This means that the odd inverting for this pair of bits leads to the reduction of the link power dissipation (Table I). The Ty block may be implemented using a simple circuit. The second stage of the encoder, which is a majority voter block, determines if the condition given in (12) is satisfied (a higher number of 1s in the input of the block compared to 0s). If this condition is satisfied, in the last stage, the inversion is performed on odd bits. The decoder circuit simply inverts the received flit when the inversion bit is high.

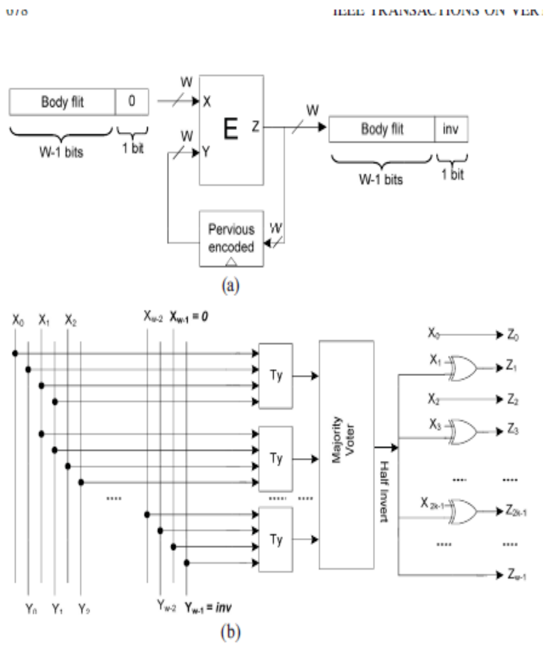


Fig. 1. *Encoder architecture scheme I. (a) Circuit diagram [27]. (b) Internalview of the encoder block (E).*

B. *Scheme II*

1) *Implementation*

The operating principles of this encoder are similar to those of the encoder implementing Scheme I. The proposed encoding architecture, which is based on the odd invert condition of (16) and the full invert condition of (18), is shown in Fig. 2. Here again, the *w*th bit of the previously and the full invert condition of (18) is shown in Fig. 2. Here again, the *w*th bit of the previously encoded body flit is indicated with inv which defines if it was odd or full inverted (inv = 1) or left as it was (inv = 0). In this encoder, in addition to the *Ty* block in the Scheme I encoder, we have the *T*2 and *T* ** 4 blocks which determine if the inversion based on the transition types *T*2 and *T* ** 4 should be taken place for the link power reduction. The second stage is formed by a set of 1s blocks which count the number of 1s in their inputs. The output of these blocks has the width of log2 *w*. The output of the top 1s block determines the number of transitions that odd inverting of pair bits leads to the link power reduction. The middle 1s block identifies the number of transitions whose full inverting of pair bits leads to the link power reduction. Finally, the bottom 1s block specifies the number of transitions whose full inverting of pair bits leads to the increased link power. Based on the number of 1s for each transition type, Module A decides if an odd invert or full invert action should be performed for the power reduction For this module, if (16) or (18) is satisfied, the corresponding output signal will become “1.” In case no invert action should be taken place, none of the output is set to “1.” Module A can be implemented using full-adder and comparator blocks. The circuit diagram of the decoder is shown in Fig. 3. The *w* bits of the incoming (previous) body flit are indicated by *Zi* (*Ri*), *i* = 0, 1, . . . ,*w* − 1. The *w*th bit of the body flit is indicated by inv which shows if it was inverted (inv = 1) or left as it was (inv = 0). For the decoder, we only need to have the *Ty* block to determine which action has been taken place in the encoder. Based on the outputs of these blocks, the majority voter block checks the validity of the inequality given by (12). If the output is “0” (“1”) and the inv = 1, it means that half (full) inversion of the bits has been performed. Using this output and the logical gates, the inversion action is determined. If two inversion bits were used, the overhead of the decoder hardware could be substantially reduced.

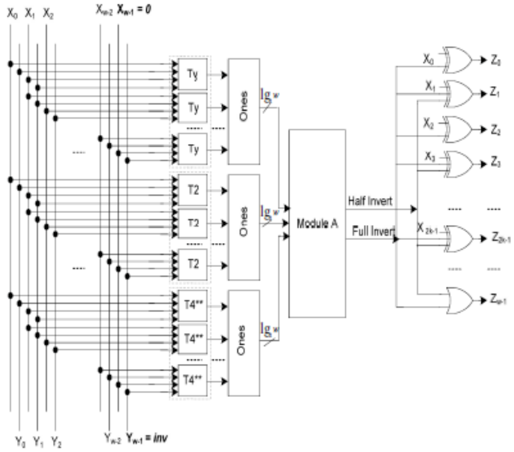


Fig. 2. *Encoder Achitecture for Sheme II*

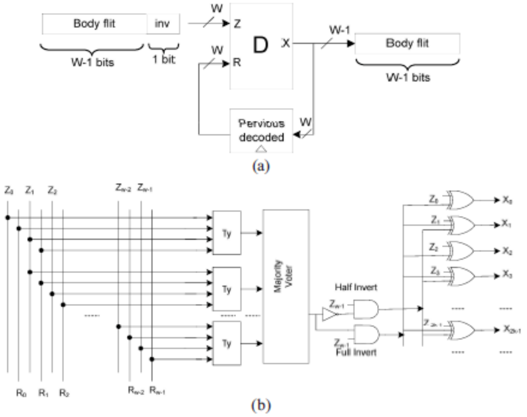


Fig. 3. *Decoder Architecture for Scheme II*

C. *Scheme III*

1) *Implementation*

The operating principles of this encoder are similar to those of the encoders implementing Schemes I and II. The proposed encoding architecture, which is based on the even invert condition of (28), the full invert condition of (29), and the odd invert condition of (30), is shown in Fig. 4. The *w*th bit of the previously encoded body flit is indicated by inv which shows if it was even, odd, or full inverted (inv = 1) or left as it was (inv = 0). The first stage of the encoder determines the transition types while the second stage is formed by a set of 1s blocks which count the number of ones in their inputs. In the first stage, we have added the *Te* blocks which determine if any of the transition types of *T*2, *T* **1 , and *T* *** 1 is detected for each pair bits of their inputs. For these transition types, the even invert action yields link power reduction. Again, we have four Ones blocks to determine the number of detected transitions for each *Ty*, *Te*, *T*2, *T* **4 , blocks. The output of the Ones blocks areinputs for Module C. This module determines if odd, even, full, or no invert action corresponding to the outputs "10," "01," "11," or "00," respectively, should be performed. The outputs "01," "11," and "10" show that whether (28), (29), and (30), respectively, are satisfied. In this paper, Module C was designed based on the conditions given in (28), (29), and (30). Similar to the procedure used to design the decoder for scheme II, the decoder for scheme III may be designed.

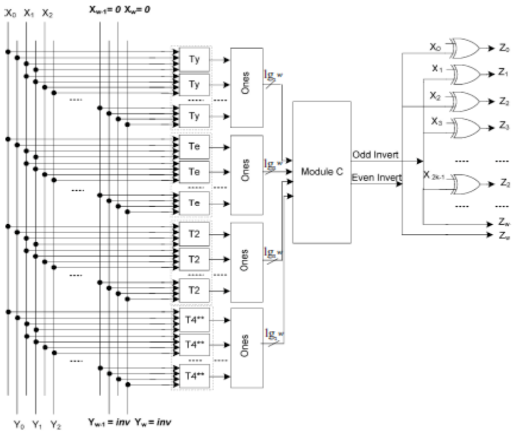


Fig. 4. *Proposed architecture for Scheme III*

III. SIMULATION AND SYNTHESIS RESULT

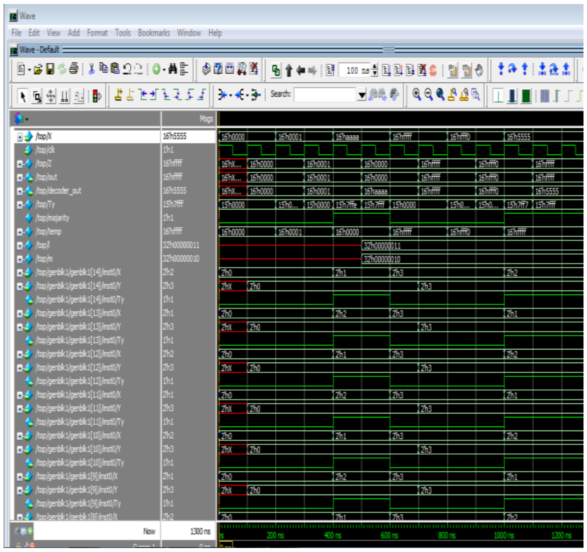


Fig. 5. *Simulation Result For Encoder and Decoder for Scheme I*

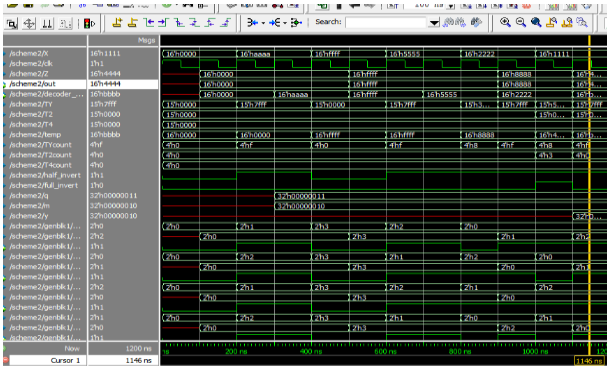


Fig. 6. *Simulation Result For Encoder and Decoder Scheme II*

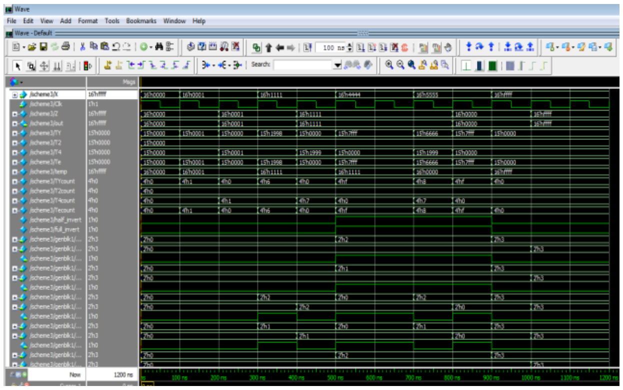


Fig. 7. *Simulation Result for Encoder and Decoder for Scheme III*

TABLE III SYNTHESIS RESULT OF SCHEME I

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	31	17,344	1%	
Number of 4 input LUTs	62	17,344	1%	
Number of occupied Slices	38	8,672	1%	
Number of Slices containing only related logic	38	38	100%	
Number of Slices containing unrelated logic	0	38	0%	
Total Number of 4 input LUTs	62	17,344	1%	
Number of bonded IOBs	49	250	19%	
IOB Flip Flops	9			
Number of BUFGMUXs	1	24	4%	
Average Fan out of Non-Clock Nets	3.09			

POWER SUPPLY SUMMARY:

```
-------------------------------------------------------------
----   Power Supply Summary                |
-------------------------------------------------------------
---- | Total    | Dynamic | Quiescent |
-------------------------------------------------------------
----   Supply Power (mW)  | 167.35  | 8.72 | 158.63
--------------------------------------------------------
```

-----Total Delay 8.473ns (5.158ns logic, 3.315ns route) (60.9% logic, 39.1% route)

TABLE IV SYNTHESIS RESULT OF SCHEME II

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	32	17,344	1%	
Number of 4 input LUTs	192	17,344	1%	
Number of occupied Slices	99	8,672	1%	
Number of Slices containing only related logic	99	99	100%	
Number of Slices containing unrelated logic	0	99	0%	
Total Number of 4 input LUTs	192	17,344	1%	
Number of bonded IOBs	49	250	19%	
Number of BUFGMUXs	1	24	4%	
Average Fanout of Non-Clock Nets	3.96			

POWER SUPPLY SUMMARY

```
----------------------------------------------------------------
| Power Supply Summary                                  |
----------------------------------------------------------------
|  Total | Dynamic  | Quiescent |
----------------------------------------------------------------
| Supply Power (mW) |165.32|  6.63  | 158.69     |
----------------------------------------------------------------
```

Total Delay: 12.862ns (7.270ns logic, 5.592ns route) (56.5% logic, 43.5% route)

TABLE V SYNTHESIS RESULT OF SCHEME III

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	16	17,344	1%	s
Number of 4 input LUTs	213	17,344	1%	
Number of occupied Slices	111	8,672	1%	
Number of Slices containing only related logic	111	111	100%	
Number of Slices containing unrelated logic	0	111	0%	
Total Number of 4 input LUTs	213	17,344	1%	
Number of bonded IOBs	33	250	13%	
Number of BUFGMUXs	1	24	4%	
Average Fan out of Non-Clock Nets	3.76			

POWER SUPPLY SUMMARY

```
-------------------------------------------------------------
| Power Supply Summary                   |
-------------------------------------------------------------
| | Total    | Dynamic     | Quiescent |
-------------------------------------------------------------
| Supply Power (mW) | 162.02  | 3.51 | 158.51     |
-------------------------------------------------------------
```

Total Dealy: 13.443ns (7.939ns logic, 5.504ns route) (59.1% logic, 40.9% route)

IV. CONCLUSION

we have presented a set of new data encoding schemes aimed at reducing the power dissipated by the links of an NoC. In fact, links are responsible for a significant fraction of the overall power dissipated by the communication system. In addition, their contribution is expected to increase in future technology nodes. As compared to the previous encoding schemes proposed in the literature, the rationale behind the proposed schemes is to minimize not only the switching activity, but also (and in particular) the coupling switching activity which is mainly responsible for link power dissipation in the deep submicron meter technology regime. The proposed encoding schemes are agnostic with respect to the underlying NoC architecture in the sense that their application does not require any modification neither in the routers nor in the links. An extensive evaluation has been carried out to assess the impact of the encoder and decoder logic in the NI. The encoders implementing the proposed schemes have been assessed in terms of power dissipation and silicon area. The impacts on the performance, power, and energy metrics have been studied using a cycle- and bit accurate NoC simulator under both synthetic and real traffic scenarios. Overall, the application of the proposed encoding.

REFERENCE

[1] International Technology Roadmap for emiconductors. (2011) [Online]. Available: http://www.itrs.net

[2] M. S. Rahaman and M. H. Chowdhury, "Crosstalk avoidance and error correction coding for coupled RLC interconnects," in Proc. IEEE Int.Symp. Circuits Syst., May 2009, pp. 141–144.

[3] W. Wolf, A. A. Jerraya, and G. Martin, "Multiprocessor system-on-chip MPSoC technology," IEEE Trans. Comput.-Aided Design Integr. CircuitsSyst., vol. 27, no. 10, pp. 1701–1713, Oct. 2008.

[4] L. Benini and G. De Micheli, "Networks on chips: A new SoC paradigm," Computer, vol. 35, no. 1, pp. 70–78, Jan. 2002.

[5] S. E. Lee and N. Bagherzadeh, "A variable frequency link for a poweraware network-on-chip (NoC)," Integr. VLSI J., vol. 42, no. 4, pp. 479–485, Sep. 2009.

[6] D. Yeh, L. S. Peh, S. Borkar, J. Darringer, A. Agarwal, andW. M. Hwu,"Thousand-core chips roundtable," IEEE Design Test Comput., vol. 25,no. 3, pp. 272–278, May–Jun. 2008.

[7] A.Vittal and M. Marek-Sadowska, "Crosstalk reduction for VLSI,"IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 16, no. 3,pp. 290–298, Mar. 1997.

[8] M. Ghoneima, Y. I. Ismail, M. M. Khellah, J. W. Tschanz, and V. De,"Formal derivation of optimal active shielding for low-power on-chip buses," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 25,no. 5, pp. 821–836, May 2006.

[9] L. Macchiarulo, E. Macii, and M. Poncino, "Wire placement for crosstalk energy minimization in address buses," in Proc. Design Autom. Test Eur. Conf. Exhibit., Mar. 2002, pp. 158–162.

[10] R. Ayoub and A. Orailoglu, "A unified transformational approach for reductions in fault vulnerability, power, and crosstalk noise and delay on processor buses," in Proc. Design Autom. Conf. Asia South Pacific, vol. 2. Jan. 2005, pp. 729–734.

[11] K. Banerjee and A. Mehrotra, "A power-optimal repeater insertion methodology for global interconnects in nanometer designs," IEEE Trans. Electron Devices, vol. 49, no. 11, pp. 2001–2007, Nov. 2002.

[12] M. R. Stan and W. P. Burleson, "Bus-invert coding for low-powerI/O," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 3, no. 1,pp. 49–58, Mar. 1995.

[13] S. Ramprasad, N. R. Shanbhag, and I. N. Hajj, "A coding framework for low-power address and data busses," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 7, no. 2, pp. 212–221, Jun. 1999.

[14] C. L. Su, C. Y. Tsui, and A. M. Despain, "Saving power in the control path of embedded processors," IEEE Design Test Comput., vol. 11, no. 4,pp. 24–31, Oct.–Dec. 1994.

[15] L. Benini, G. De Micheli, E. Macii, D. Sciuto, and C. Silvano, "Asymptotic zero-transition activity encoding for address busses in low-power microprocessor-based systems," in Proc. 7th Great Lakes Symp. VLSI,Mar. 1997, pp. 77–82.

[16] E. Musoll, T. Lang, and J. Cortadella, "Working-zone encoding for reducing the energy in microprocessor address buses," IEEE Trans. VeryLarge Scale Integr. (VLSI) Syst., vol. 6, no. 4, pp. 568–572, Dec. 1998.

[17] W. Fornaciari, M. Polentarutti, D. Sciuto, and C. Silvano, "Power optimization of system-level address buses based on software profiling,"in Proc. 8th Int. Workshop Hardw. Softw. Codesign, May 2000, pp. 29–33.

[18] L. Benini, G. De Micheli, E. Macii, M. Poncino, and S. Quer, "Power optimization of core-based systems by address bus encoding," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 6, no. 4, pp. 554–562, Dec. 1998.

[19] L. Benini, A. Macii, M. Poncino, and R. Scarsi, "Architectures and synthesis algorithms for power-efficient bus interfaces," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 19, no. 9, pp. 969–980,Sep. 2000.

[20] G. Ascia, V. Catania, M. Palesi, and A. Parlato, "Switching activity reduction in embedded systems: A genetic bus encoding approach," IEE

ProcComput. Digit. Tech., vol. 152, no. 6, pp. 756–764, Nov. 2005.

[21] R. Siegmund, C. Kretzschmar, and D. Muller, “Adaptive Partial Businvert encoding for power efficient data transfer over wide system buses,” in Proc. 13th Symp. Integr. Circuits Syst. Design, Sep. 2000, pp. 371–376.

[22] S. Youngsoo, C. Soo-Ik, and C. Kiyoung, “Partial bus-invert coding for power optimization of application-specific systems,” IEEE Trans. VeryLarge Scale Integr. (VLSI) Syst., vol. 9, no. 2, pp. 377–383, Apr. 2001.

[23] M. Palesi, G. Ascia, F. Fazzino, and V. Catania, “Data encoding schemes in networks on chip,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 30, no. 5, pp. 774–786, May 2011.

[24] C. G. Lyuh and T. Kim, “Low-power bus encoding with crosstalkdelay elimination,” IEE Proc. Comput. Digit. Tech., vol. 153, no. 2, pp. 93–100, Mar. 2006.

[25] P. P. Pande, H. Zhu, A. Ganguly, and C. Grecu, “Energy reduction through crosstalk avoidance coding in NoC paradigm,” in Proc. 9th EUROMICRO Conf. Digit. Syst. Design Archit. Methods Tools, Sep. 2006, pp. 689–695.

[26] K. W. Ki, B. Kwang Hyun, N. Shanbhag, C. L. Liu, and K. M. Sung, “Coupling-driven signal encoding scheme for low-power interface design,” in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, Nov. 2000, pp. 318–321.

[27] L. Rung-Bin, “Inter-wire coupling reduction analysis of bus-invert coding,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 7, pp. 1911 1920, Aug. 2008.

[28] Z. Khan, T. Arslan, and A. T. Erdogan, “Low power system on chip bus encoding scheme with crosstalk noise reduction capability,” IEE Proc. Comput. Digit. Tech., vol. 153, no. 2, pp. 101–108, Mar. 2006.

[29] Z. Yan, J. Lach, K. Skadron, and M. R. Stan, “Odd/even bus invert withtwo-phase transfer for buses with coupling,” in Proc. Int. Symp. Low Power Electron. Design, 2002, pp. 80–83.

[30] C. P. Fan and C. H. Fang, “Efficient RC low-power bus encoding methods for crosstalk reduction,” Integr. VLSI J., vol. 44, no. 1, pp. 75–86, Jan. 2011.

[31] S. R. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, W. James, D. Finan, A. P. Singh, T. Jacob, S. Jain, V. Erraguntla, C. Roberts, Y. V. Hoskote, N. Y. Borkar, and S. Y. Borkar, “An 80-tile Sub-100-W TeraFLOPS processor in 65-nm CMOS,” IEEE J. Solid-State Circuits, vol. 43, no. 1, pp. 29–41, Jan. 2008.

[32] S. Murali, C. Seiculescu, L. Benini, and G. De Micheli, “Synthesis of networks on chips for 3D systems on chips,” in Proc. Asia South Pacific Design Autom. Conf., Jan. 2009, pp. 242–247.

[33] C. Seiculescu, S. Murali, L. Benini, and G. De Micheli, “SunFloor 3D: A tool for networks on chip topology synthesis for 3-D systems on chips,” in Proc. IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 29, no. 12, pp. 1987–2000, Dec. 2010.

[34] S. Murali and G. De Micheli, “Bandwidth-constrained mapping of cores onto NoC architectures,” in Proc. Design, Autom. Test Eur. Conf.Exhibit., vol. 2. Feb. 2004, pp. 896–901.

[35] M. Palesi, R. Tornero, J. M. Orduñna, V. Catania, and D. Panno, “Designing robust routing algorithms and mapping cores in networks-onchip: A multi-objective evolutionary-based approach,” J. Univ. Comput. Sci., vol. 18, no. 7, pp. 937–969, 2012