International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

Efficient Data Encoding and Decoding for Network-On-Chip Application

Palagani Yellappa†, S K. Mahaboob Basha† and K. Neelima†

†Department of Electronics and Communication Engineering, Sree Vidyanikethan Engineering College (Autonomous), Tirupathi, India

Accepted 10 Aug 2015, Available online 18 Aug 2015, **Vol.5, No.4 (Aug 2015)**

Abstract

Network-on-Chip (NoC) is composed of three main building blocks links, routers and network interfaces (NIs). In NoC links are the major power dissipation sources and self-switching, coupling-switching activities are responsible for link power dissipation. This paper introduces set of efficient data encoding and decoding schemes for reduction of link power dissipation, delay. First self switching is reduced by checking the switching transition and then the coupling between the links is checked and ensured that the power consumption is reduced. In this paper, we refer to the end-to-end encoding technique, this technique takes advantage of the pipeline nature of the wormhole switching technique. Especially the decoding schemes are focused on reducing hardware and delay. This paper proposing the concept of on-chip networks, sketches a data encoding & decoding internal views and analyzes the power and delay reduction of both encoding & decoding schemes in Xilinx Spartan3E family (XC3S500E-5FG320).

Keywords: *Coupling switching activity, Data encoding, Data decoding, NoC, Power reduction and verilog HDL.*

1. Introduction

Every technology has facing some problems like power dissipation, energy consumption problems etc. In VLSI technology in order to estimate and optimize the power consumption of a digital circuit, it is necessary to know how energy is dissipated. Generally the inter-connection architecture is based on dedicated wires or shared busses. If system has a limited number of cores then dedicated wire architecture is effective, otherwise increases the system complexity. A shared bus is a set of wires which is common to multiple cores, so this approach is more flexible and is totally reusable. But it allows only one communication transaction at a time, all cores share the same communication bandwidth in the system and its scalability is limited to few dozen IP cores [**ITRS online**].

Plenty of traditional inter-connect schemes like point-to-point, crossbars and buses are available to interconnect small number of cores. While achieving fast and efficient communication with point-to-point communication schemes, wire density is a barrier for adapting them to many core architectures. Moreover, buses are simpler in design, they suffer from the scalability and arbitration issues along with bandwidth bottleneck as the number of cores increases [**D. Yeh and L. S. Peh *et al.* 2008**]. Similarly the area and power requirements of a crossbarlimit its applicability. Hence, in many core architectures like Chip Multi-processors (CMP) and Multi processor System-on-Chip (MPSoC), emerge the need of an efficient communi-cation infrastructure as traditional solutions fails to handle the communication challenges.

Network-on-Chip (NoC) is a scalable and modular design approach, it is a attractive alternative for the traditional shared buses or dedicated wires due to many reasons. First, NoCs represent a scalable solution to on-chip communication paradigm, because they provide scalable bandwidth at low power and area overheads. Second, NoCs are very efficient in terms of use of wiring and multiplexing many traffic flows on the same channels providing quality of service (QoS) and higher bandwidth. Finally, on-chip networks with regular topologies have short interconnects that can be optimized and reused using regular iterative blocks, thus making the verification process easy.

In NoC architecture network interface (NI), switches and links are the main components. The network interface or network adapter makes the logical connection between the IP core and the network, since each IP may have a distinct interface protocol with respect to the network. Switches carry out the task of dispatching packets inside the network, depending on the particular routing scheme choosen. The number of ports depends on the topology of the network. Links are used to transmit packet between routers.

*Corresponding author **Palagani Yellappa** is a M.Tech Student, VLSI; **S K. Mahaboob Basha and K. Neelima** are working as Assistant Professors

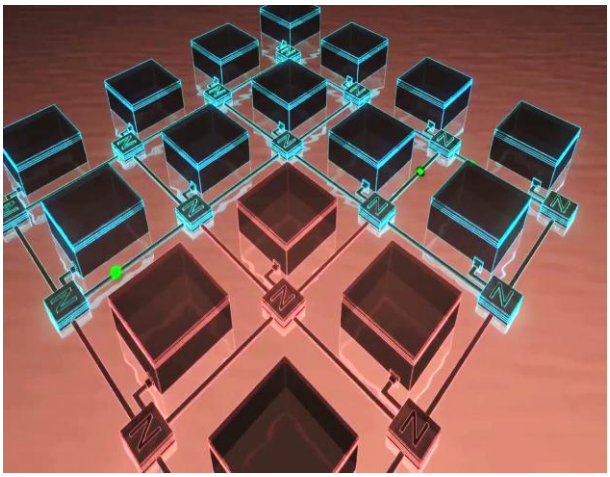


Fig.1 Network-on-Chip power dissipation sources (links)

The rest of this paper is organized as follows. We briefly discuss existing methods (Bus-Invert and Gray-code) in Section II, while Section III presents an overview of the proposed data encoding schemes. The proposed data encoding and decoding 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.

2. Existing Methods

The data encoding techniques may be classified into two categories. In the first category, encoding techniques are concentrates on lowering the power due to self-switching activity of each bus lines. In this category Bus-Invert and INC-XOR were proposed for the case of random data patterns are transmitted via these lines. Among these methods, Bus-Invert encoding is a more popular technique for reduce dynamic switching power and the number of transitions on the bus. In the second category Gray-code encoding, working-zone encoding, T0 and T0-XOR were suggested for the case of correlated data patterns. Among these methods, Gray-code encoding is widely used technique for reducing the errors and power dissipation by the links of an NoC. Application specific approaches (ASP) have also been proposed, but this category is not suitable to be applied in the deep sub-micron meter technology nodes where the coupling capacitance constitutes a major part of the total inter-connect capacitance.

2.1 Bus-Invert (BI) encoding method

The Major idea behind Bus-Invert encoding originated by noting that more power is wasted during data transmission in off-chip bus lines. This is due to the the high capacitance lines and high switching activities therefore power could be saved by minimizing the number of transitions occurring on these bus lines. The major types of switching transitions are self-switching and coupling-switching [**A. Sathish *et al*. 2011**] . Self-switching transition is defined as transition on the capacitance between a data bus line and substrate (ground). Coupling-switching transition on data bus is defined as transition on capacitance between adjacent bus lines. Let's consider data value (piece of information) and bus value (actual value on the bus). The Bus-Invert method uses one extra control bit called 'invert-bit'. In this method the peak power dissipation can be decreased by half by coding as follows [**Mircea R *et al*. 1995**].

1) Compute the Hamming distance between the present bus value and the next data value.
2) If the Hamming distance is larger than n/2 ('n' is number of bits), set invert =1 and make the next bus value equal to the inverted next data value.
3) Otherwise let invert = 0 and let the next bus value equal to the next data value.
4) At the receiver side the contents of the bus must be conditionally inverted according to the invert-line, unless the data is not stored encoded as it is (e.x., in a RAM). In any case the value of invert must be transmitted over the bus (the method increases the number of bus lines from 'n' to 'n+1').

The Bus-Invert method generates a code that has the property that the maximum number of transitions per time-slot is reduced from 'n' to 'n/2' and thus the peak power dissipation is reduced by half. Using the Bus-Invert coding in order to decrease the number of transitions.

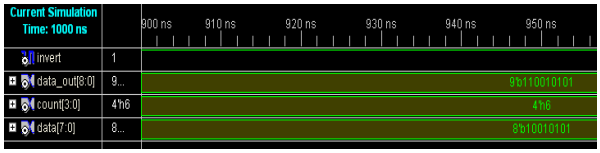


Fig.2 Bus-Invert encoding

2.2 Gray-Code encoding method

The power dissipated by the links of a NoC due to noise increases the risk of errors in the communication subsystem. Using the data encoding with Gray input is mainly reducing the errors and power dissipation by the links of an NoC. This method consists of a Binary to Gray converter, Encoder, Decoder and Gray to Binary converter and uses the Binary to Gray conversion at the transmitter section and Gray to Binary conversion at the receiver section [**S. Kavitha *et al*. 2014**].

The Binary to Gray converter is used to convert the binary data to gray data. The reflected binary code also known as Gray code. Gray codes are widely used to facilitate error correction in digital communications such as digital terrestrial television (DTT) and some cable TV systems.

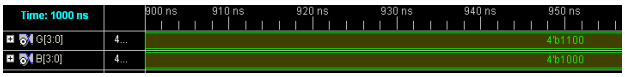


Fig.3 Binary to Gray code conversion

Encoder and Decoder are the basic elements of any digital communication system. An Encoder is a device that converts information from one format to another format, for the purposes of standardization, speed and compressions. A simple Encoder circuit can receive a single active input out of '2n' input lines generate a binary code on 'n' parallel output lines. A Decoder is a device which does the reverse operation of an Encoder, so that the original information (input data) can be retrieved. A decoder converts binary information from 'n' input lines to a maximum of '2n' unique output lines. The Encoder and Decoder are responsible for implementing the tolerance against transient faults. Finally Gray to Binary converter is used to convert the gray data to binary data.

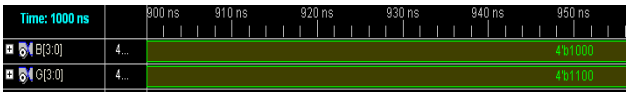


Fig.4 Gray to Binary code conversion

3. Overview of the Proposal

Energy consumption and power dissipation are today recognized as the most important design optimization objectives. 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 and coupling-switching activities. In fact, self-switching and coupling-switching activities are responsible for link power dissipation. In this paper we prefer End-to-End technique, this technique takes advantage of the pipeline nature of the wormhole switching technique [**Ioannis Nousias and Tughrul Arslan, 2006**]

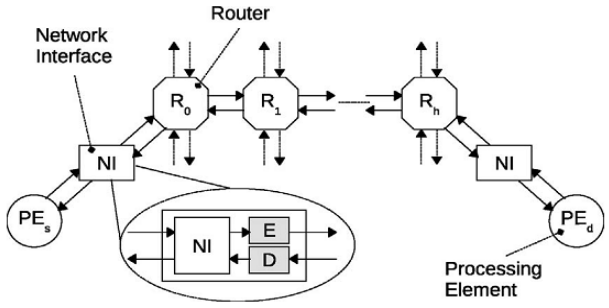


Fig.5 General Scheme of Proposed Approach

This paper introduces three encoding and decoding schemes. In Scheme I, we focus on reducing Type I transitions and in Scheme II, both Types I and Type II transitions are taken into account for deciding between halfinvert and fullinvert, which is depending up on the amount of switching reduction. Finally, in Scheme III, we consider the fact that Type I transitions show different behaviors in the case of odd and even inverts and make the inversion which leads to the higher power saving. Our aim is to covert Type I and Type II to Type III and Type IV bit combinations as far as possible. This is because, Type III and Type IV combinations result in less coupling-switching and normal-switching activities.

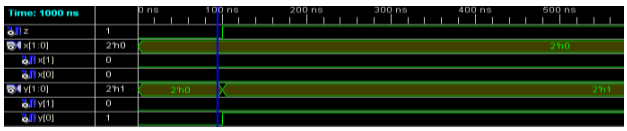
4. Proposed Encoding and Decoding schemes

The main goal of the proposed encoding scheme is to reduce the power dissipation by minimizing the coupling transition activities on the links of the inter-connection network. Coupling transitions are classified into four types Type I, Type II, Type III and Type IV are shown in the below Table 1.

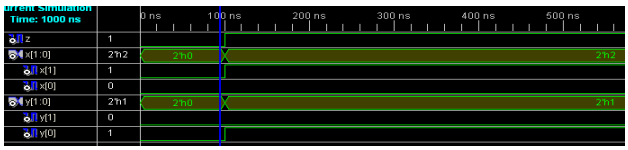
Table 1 Effect of odd inversion on change of transition types

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

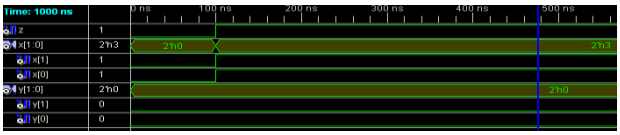
A Type I transition occurs when one of the lines switches when the other remains unchanged (ex., 00,01 or 00,10 etc.).



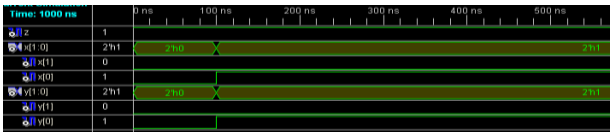
In a Type II transition, one line switches from low to high while the other makes transition from high to low (ex., 01,10 or 10,01 etc.).



A Type III transition corresponds to the case where both lines switch simultaneously (ex., 11,00 or 00,11 etc.).



Finally, in a Type IV transition both lines do not change (ex., 00,00 or 10,10 etc.).



The effective switched capacitance varies from type to type (I to IV) and hence, the coupling transition activity (Tc) is a weighted sum of different types of coupling transition contributions. Therefore [**Nima Jafarzadeh *et al*. 2014**]

$$T_C=K_1T_1+K_2T_2+K_3T_3+K_4T_4 \quad (1)$$

Where Ti is the average number of Type i transition and Ki is the corresponding weight. 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 [**K. W. Ki and B. Kwang Hyun *et al*. 2000**].

4.1 Scheme I

In scheme 1, our main goal is to reducing the number of Type I and Type II transitions. Type I transitions is converted into Type III and Type IV. Type II transitions is converted into Type I transitions. This scheme compares the two data's based on to reducing the link power reduction by doing odd inversion or no inversion operation.

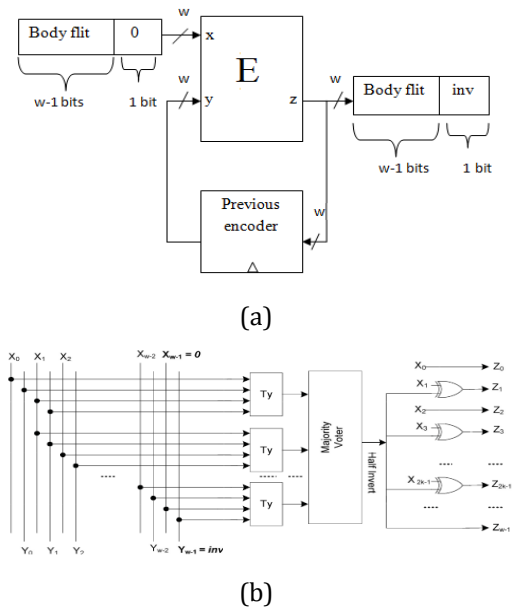


Fig.6 Encoder architecture scheme I(a) Circuit diagram (b) Internal view of the encoder block (E)

Now, defining [**Nima Jafarzadeh *et al*. 2014**]

$$Ty = T_2 + T_1 - T_1*** \quad (2)$$

The majority voter block, determines if the below condition (3) is satisfied

$$Ty > 0.5\ (w-1) \quad (3)$$

The general block diagram in Fig 6(a) is same for scheme I, scheme II and scheme III. The 'w-1' ('w' is total number of bits) bits are given to the encoder block (E) and another input of the encoder block is the previously encoded output. Block E compares these two inputs and performing the any one of the inversion based on the transition types. Comparing the current data and previous encoded data to decide which inversion is performed for link power reduction. Here the 'Ty' block takes two adjacent bits (eqn. 2) from the given inputs checks what type of transitions occurs, whether more number of type I and type II transitions is occurring means it set the output state to '1', otherwise it set the output to '0'. The odd inversion is performed for these types of transitions. Then the next block is the Majority voter (eqn. 3) it checks the state, if the number of 1's is greater than 0's or not. The last stage using the XOR circuits, these circuits is used to perform the inversion on odd bits. The decoding is performed by simply inverts the encoder circuit when the invert bit is high.

4.2 Scheme II

In scheme II, our main goal is to reducing the number of Type II transitions, so Type II transitions are converted into Type IV transitions. This scheme compares the two data's based on to reducing the link power reduction by doing full inversion or odd inversion or no inversion operation. Full and odd inversion based this advanced encoding architecture consist of w-1 link width and one bit for inversion bit which indicate if the bit travel through the link is inverted or not. W bits link width is considered when there is no encoding is applied for the input bits. Here the Ty block (eqn. 2) from scheme I is added in scheme II.

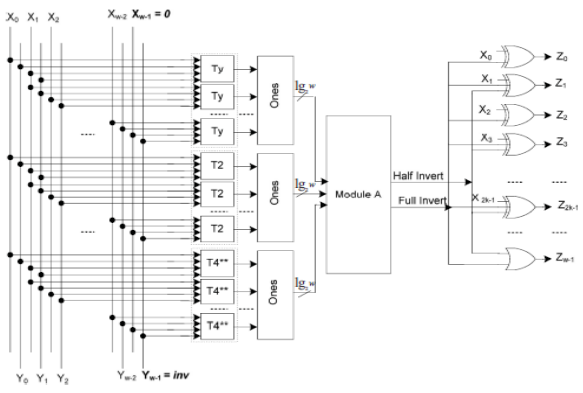


Fig.7 Encoder architecture Scheme II

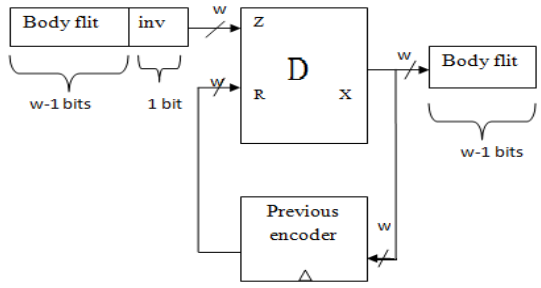
The odd inversion condition is obtained as

$$2(T_2 - T_4^{**}) < 2Ty - w + 1 \qquad Ty > (w-1)/2 \qquad (4)$$

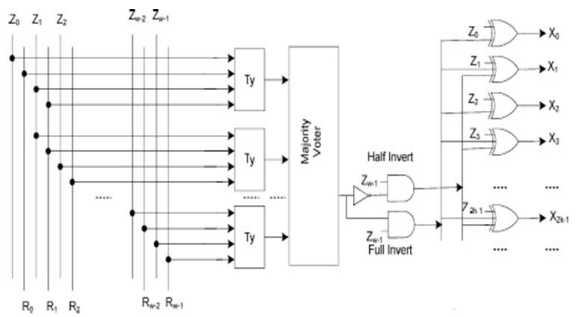
The full inversion condition is obtained as

$$2(T2 - T4^{**}) > 2Ty - w + 1 \qquad T2 > T4^{**} \qquad (5)$$

The operating principles of scheme II encoder is similar to Scheme I encoder. The proposed encoding architecture, which is based on the odd invert condition of (4) and the full invert condition of (5), is shown in Fig. 7. Here again, the wth 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 T2 and T4** blocks which determine if the inversion based on the transition types T2 and T4** should be taken place for the link power reduction. The second stage is formed by a set of 1's blocks which count the number of 1's in their inputs. The output of the top 1's block determines the number of transitions that odd inverting of pair bits leads to the link power reduction. The middle 1's 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 1's for each transition type, Module A decides if an odd or full invert action should be performed for the power reduction.



(a)



(b)

Fig.8 Decoder architecture Scheme II (a) Circuit diagram (b) Internal view of the decoder block (D)

The circuit diagram of the decoder is shown in Fig. 8. The w-1 bits are given to decoder circuit and another input of the decoder is previous decoded output. The decoder block compares the two input data's and inversion operation is performed and w-1 bits output is produced. The remaining one bit is used to indicate the inversion is performed or not. In decoder circuit diagram (Fig. 8) consist of Ty block and Majority vector and XOR circuits. Based on the encoder action the Ty block is determined the transitions. Based on the transitions types the majority blocks checks the validity of the inequality given by (eqn. 3). The output of the majority voter is given to the XOR circuit. Half inversion, full inversion and no inversion is performed based on the logic gates.

4.3 Scheme III

Table 2 Effect of even inversion on change of transition types

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

In scheme III, we are add the even inversion into scheme II. Because the odd inversion converts Type I (T_1^{***}) transitions into Type II transitions. From table II, T1**/T1*** are converted into Type IV/Type III transitions by the flits is even inverted. The link power reduction in even inversion is larger than the Odd inversion [**Z. Yan and J. Lach *et al*. 2002**].

Defining

$$Te = T_2 + T_1 - T_1^* \qquad (6)$$

The even inversion leads to power reduction [**Nima Jafarzadeh *et al*. 2014**]

$$Te>(w-1)/2,\ Te>Ty,\ 2(T_2-T_4^{**})< 2Te - w+1 \qquad (7)$$

The full inversion leads to power reduction

$$2(T_2-T_4^{**}) > 2Ty-w+1, \qquad (T_2 > T_4^{**})$$
$$2(T_2-T_4^{**}) > 2Te-w+1 \qquad (8)$$

$2(T_2-T_4^{**}) < 2Ty-w+1, \quad Ty > (w-1)/2$
$Te < Ty$ (9)

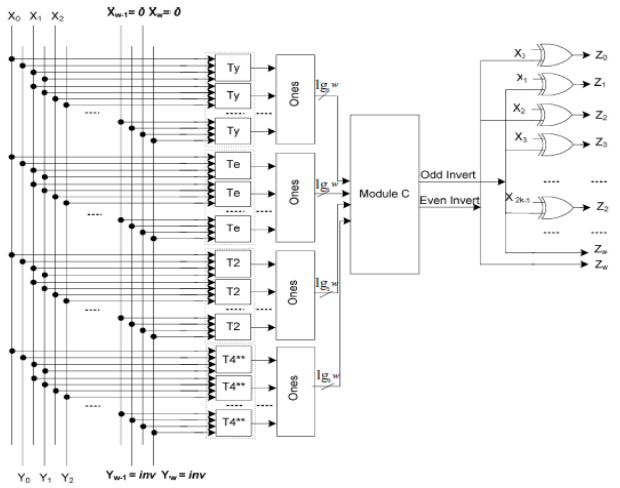


Fig.9 Encoder architecture Scheme III

The encoding architecture (Fig. 9) in scheme III is same of encoder architecture in scheme I and II. Here we are adding Te block to the scheme II. This is based on even invert, full invert and odd invert conditions. It consists of w-1 link width input and the w bit is used for the inversion bit. The full, half and even inversion is performed means the inversion bit is set '1', otherwise it set as '0'. The Te block is determined if any of the detected transition of types T2, T1** and T1***. For these transition types, the even invert action yields link power reduction. The one's block determines the number of ones in the corresponding transmissions of Ty, T2, Te and T4**. This number of one's is given to the Module C block. This block check if odd, even, full or no invert action corresponding to the outputs '10', '01', '11' or '00' respectively, should be performed. The decoder architecture of scheme II and scheme III are same is shown in the belo Fig.10.

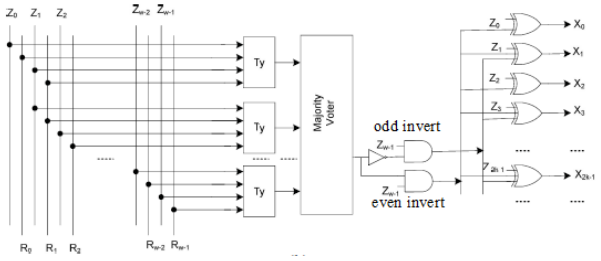


Fig.10 Decoder internal view of scheme III

5. Results and Discussion

The proposed data encoding and decoding schemes are simulated and verified using Verilog HDL in Xlinix ISE 10.1i for the target device xc3s500e-5fg320.

5.1 Power comparison Existing Methods

Table 3 Power dissipation and delay in Bus-Invert Encoding method

Name	Power (W)	Used	Total Available	Utilization (%)
Logic	0.000	37	9312	0.4
Signals	0.000	45	---	---
IOs	0.017	22	232	9.5
Total Quiescent Power	0.081			
Total Dynamic Power	0.017			
Total Power	0.098			

Total 30.776ns (18.351ns logic, 12.425ns route)
(59.6% logic, 40.4% route)

Table 4 Power dissipation in Gray-code Encoding method

Name	Power (W)	Used	Total Available	Utilization (%)
Logic	0.000	3	9312	0.0
Signals	0.000	7	---	---
IOs	0.005	8	232	3.4
Total Quiescent Power	0.081			
Total Dynamic Power	0.005			
Total Power	0.086			

5.2 Proposed schemes discussion and comparison

A. Scheme I

The scheme I main goal is reducing the number of Type I transitions and Type II transitions. In the encoding logic, each Ty block takes the two adjacent bits of the input flits ($X_1X_2Y_1Y_2$, $X_2X_3Y_2Y_3$, etc.).

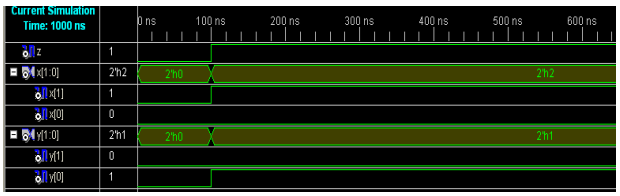


Fig.11 Simulate output of Ty block

The Majority voter checks the state, if the number of one's is greater than zeros or not.

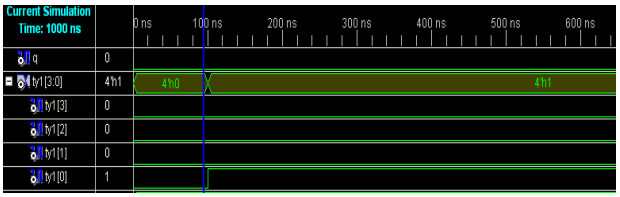


Fig. 12 Simulate output of Majority voter

The output of the scheme I reducing the number of Type I and Type II transitions by using the odd invert condition. This means that the odd inverting for this pair of bits leads to the reduction of the link power dissipation.

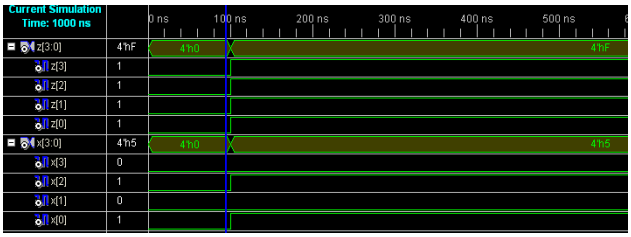


Fig.13 Simulate output of Scheme I Encoding (4-bit)

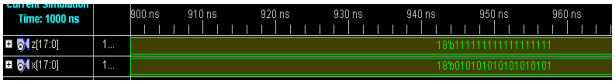


Fig.14 Simulate output of Scheme I Encoding (18-bit)

B. Scheme II

In scheme II the type II transition is converted into Type IV transitions by using the odd and full inversion condition. Based on the one's block the Module A takes the decision of which inversion (odd or full) should be performed for the link power reduction. For this module (eqn. 4 & 5) is satisfied, the corresponding output is set to '1' otherwise output is set to '0' (no inversion is takes place).

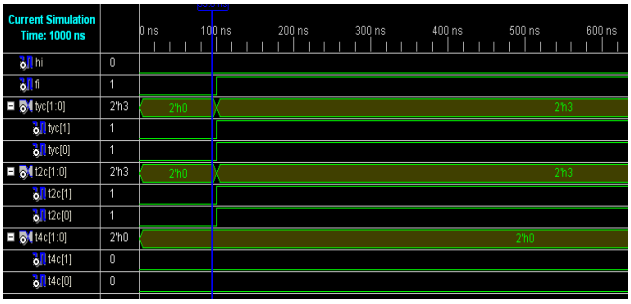


Fig.15 Simulate output of Module A

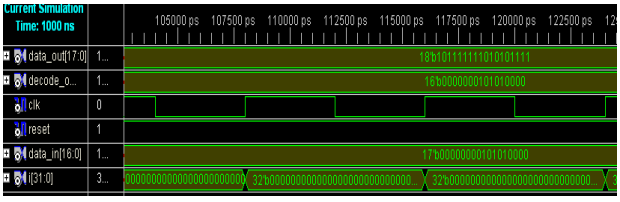


Fig.16 Simulate output of Scheme II Encoding and Decoding (18-bit)

C. Scheme III

Adding the 'Te' block to the scheme II, this is based on even invert and Odd invert conditions.

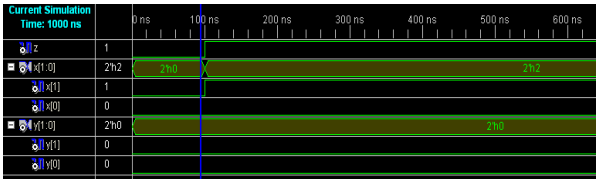


Fig.17 Simulate output of Te block

Module C check if odd, even, full or no invert action corresponding to the outputs '10' '01', '11' or '00' respectively, should be performed.

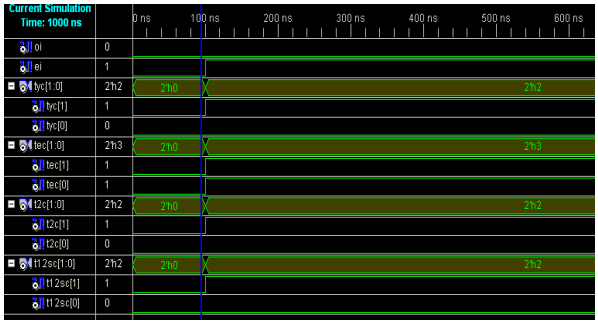


Fig.18 Simulate output of Module C

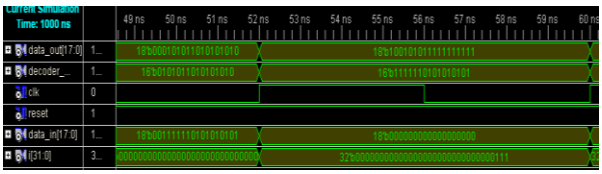


Fig.19 Simulate output of Scheme III Encoding and Decoding (18-bit)

Power comparison Proposed Methods:

Table 5 Power dissipation in scheme I

Name	Power (W)	Used	Total Available	Utilization (%)
Logic	0.000	17	9312	0.2
Signals	0.000	35	---	---
IOs	0.022	36	232	15.5
Total Quiescent Power	0.081			
Total Dynamic Power	0.022			
Total Power	0.103			

Table 6 Power dissipation and delay in scheme II

Name	Power (W)	Used	Total Available	Utilization (%)
Clocks	0.000	1	---	---
Logic	0.000	438	9312	4.7
Signals	0.002	449	---	---
IOs	0.001	38	232	16.4
Total Quiescent Power	0.081			
Total Dynamic Power	0.003			
Total Power	0.084			

```
Total                     20.406ns (12.409ns logic, 7.997ns route)
                                   (60.8% logic, 39.2% route)
```

Table 7 Power dissipation and delay in scheme III

Name	Power (W)	Used	Total Available	Utilization (%)
Clocks	0.000	1	---	---
Logic	0.000	320	9312	3.4
Signals	0.001	336	---	---
IOs	0.001	37	232	15.9
Total Quiescent Power	0.081			
Total Dynamic Power	0.002			
Total Power	0.083			

```
Total                     17.678ns (10.593ns logic, 7.085ns route)
                                   (59.9% logic, 40.1% route)
```

Future Enhacement

3-bit Coupling Transitions

Fig.20 RTL schematic of 3-bit Coupling transition

A Type I transition occurs when one of the lines switches when the other remains unchanged (ex., 000,001 or 000,010 etc.).

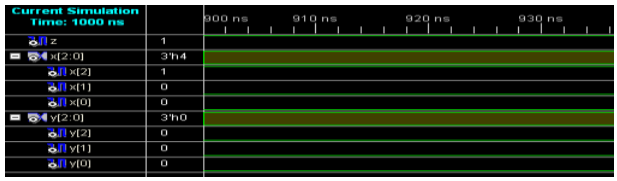


Fig.21 Simulate output of Type I Coupling transition

In a Type II transition, one line switches from low to high while the other makes transition from high to low (ex., 001,110 or 010,10 etc.).

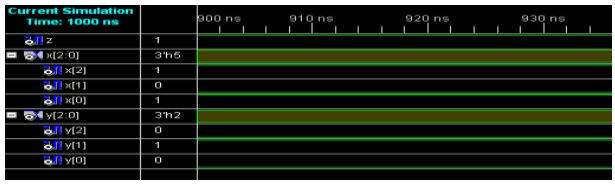


Fig.22 Simulate output of Type II Coupling transition

A Type III transition corresponds to the case where three lines switch simultaneously (ex., 000,111 or 111,000 etc.).

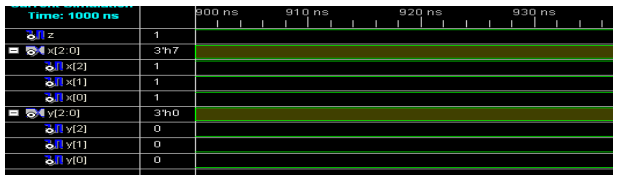


Fig.23 Simulate output of Type III Coupling transition

Finally, in a Type IV transition three lines do not change (ex., 000,000 or 110,110 etc.).

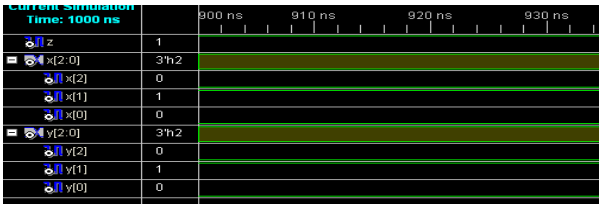


Fig.24 Simulate output of Type IV Coupling transition

Conclusion

In this paper three encoding and decoding schemes are proposed (namely scheme I, scheme II and scheme III) for reducing coupling and normal switching activities in links of NoC. As the consecutive bits are taken care, not to have opposite values so that coupling-switching activity is reduced. Similarly, the bits passed through particular links are encoded in such a way that toggling (opposite previous and present values) of the bit values in that particular links are prevented, to reduce normal switching activity. The main aim of this paper is to covert Type I and Type II to Type III and Type IV bit combinations in the best possible way, as Type III and Type IV combinations result in less coupling switching and normal switching activities. From results it is clear that scheme III has even lesser coupling switching and normal switching activity when compared to scheme II which is achieved by the inclusion of 'even inversion' module 'te module' in the Scheme-III. So that Type I and Type II bit combinations are converted into Type III and Type IV bit combinations. Also Scheme II converts some Type I bit combinations to Type II bit combinations. Hence, Scheme III proves to be more optimized than Scheme II.

References

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

D. Yeh, L. S. Peh, S. Borkar, J. Darringer, A. Agarwal, andW. M. Hwu, (2008), Thousand-core chips roundtable, *IEEE,* vol. 25, 272–278.

A. Sathish, M.Madhavi Latha and K. Lalkishor, (2011), An Efficient Switching Activity Reduction Technique for On-Chip Data Bus, *IJCSI*, Vol. 8, 407-413.

Mircea R. Stan and Wayne P. Burleson, (1995), Bus-Invert Coding for Low-Power I/O, *IEEE*, Vol. 3, 49-58.

S. Kavitha (2014), Data Encoding Technique Using Gray Code in Network-on-Chip, *IJST*, Vol. 2, 186-193.

S. Anusuyahdevi and Dr.S.Jayashri, (2014), Performance Analysis of an Efficient Low Power NOC Router System Using Gray Encoding Techniques, *IJIRCCE*, Vol. 2, 7463-7469.

Ioannis Nousias, Tughrul Arslan, (2006), Wormhole Routing with Virtual Channels using Adaptive Rate Control for Network-on-Chip (NoC), *NASA/ESA Conference on Adaptive Hardware and Systems.*

Nima Jafarzadeh, Maurizio Palesi, Ahmad Khademzadeh and Ali Afzali-Kusha, (2014), Data Encoding Techniques for Reducing Energy Consumption in Network-on-Chip, *IEEE*, Vol. 22, 675-685.

K. W. Ki, B. Kwang Hyun, N. Shanbhag, C. L. Liu, and K. M. Sung, (2000), Coupling-driven signal encoding scheme for low-power interface design, *IEEE*, Vol. 2, 318-321.

Maurizio Palesi, Giuseppe Ascia, Fabrizio Fazzino and Vincenzo Catania, (2011), Data Encoding Schemes in Networks on Chip, *IEEE*, Vol. 30, 774-786.

Z. Yan, J. Lach, K. Skadron, and M. R. Stan, (2002), Odd/even bus invert with two-phase transfer for buses with coupling, 80–83.

Palagani Yellappa, is currently pursuing his M.Tech in the specialization of VLSI in ECE Department of Sree Vidyanikethan Engineering college (Autonomous), Tirupathi. He has completed B.Tech from Sree Rama Engineering College, Tirupathi. His current research interests include network-on-chips, lowpower inter-connection and VLSI Signal Processing.

Sk. Basha, is currently working as an Assistant Professor in the ECE Department of Sree VidyanikethanEngineering College (Autonomous), Tirupathi. He has completed his M.Tech in VLSI Design, in Satyabhama University . His research area of interest includes Digital System Design and VLSI Signal Processing.

.

K. Neelima, is currently working as an Assistant Professor in the ECE Department of Sree VidyanikethanEngineering College (Autonomous), Tirupathi. She has completed her M.Tech in VLSI Design, in Satyabhama University . Her research area of interest includes Digital System Design and VLSI Signal Processing.