e-ISSN : 2349-9745
p-ISSN : 2393-8161
Scientific Journal Impact Factor (SJIF): 1.711
International Journal of Modern Trends in Engineering and Research
www.ijmter.com

A NEW DATA ENCODER AND DECODER SCHEME FOR NETWORK ON CHIP

DURAIMURUGAN.M[1], Ms.BHARATHI.V,M.E[2]
[1]*ME COMMUNICATION SYSTEM, Dept of ECE, Dhanalakshmi Srinivasan Engineering college, Perambalur, Tamilnadu, India,*
[2]*Assistant Professor -ECE, Department of ECE, Dhanalakshmi Srinivasan Engineering college, Perambalur, Tamilnadu, India,*

Abstract—System-on-chip (soc) based system has so many disadvantages in power-dissipation as well as clock rate while the data transfer from one system to another system in on-chip. At the same time, a higher operated system does not support the lower operated bus network for data transfer. However an alternative scheme is proposed for high speed data transfer. But this scheme is limited to SOCs. Unlike soc, network-on-chip (NOC) has so many advantages for data transfer. It has a special feature to transfer the data in on-chip named as transitional encoder. Its operation is based on input transitions. At the same time it supports systems which are higher operated frequencies. In this project, a low-power encoding scheme is proposed. The proposed system yields lower dynamic power dissipation due to the reduction of switching activity and coupling switching activity when compared to existing system. Even-though many factors which is based on power dissipation, the dynamic power dissipation is only considerable for reasonable advantage. The proposed system is synthesized using quartus II 9.1 software. Besides, the proposed system will be extended up to interlink PE communication with help of routers and PE's which are performed by various operations. To implement this system in real NOC's contains the proposed encoders and decoders for data transfer with regular traffic scenarios should be considered.

Keywords— Coupling switching activity, data encoding, interconnection on chip, low power, network-on-chip (NOC), power analysis**.**

I.INTRODUCTION

In general digital logic design has two main criteria. One is SOC (system on-chip) and another one is NOC (network on-chip).earlier versions of processing systems such micro-processors, micro-controllers and DSP processors belongs to SOC has performance degradation in all kinds. However these designs are designed using VLSI CAD tools.

Shifting from a silicon technology node to the next one result 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. 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. The network on-chip (NoC) design paradigm is recognized as the most viable way to tackle with scalability and variability issues that characterize the ultra deep submicron meter era[1]. As the design complexity increases, the total length of the interconnection wires increases, resulting in long transmission delay and higher power consumption. In addition, the distance between wires shrinks with technology, increasing coupling capacitance, and the height of the wire material increases resulting in greater fringe capacitance[2].

Nowadays, the on-chip communication issues are as relevant as, and in some cases more relevant than, the computation related issues[8]. 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[9].

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[3]. 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 packet[7]s. We focus on data encoding schemes as a viable way to reduce power dissipated by the network links. The basic idea is to opportunely encode the data before their injection in the network in such a way as to reduce the switching activity of the links.

The proposed data encoding schemes are assessed on a set of traffic scenarios both synthetic and extracted from real applications. The analysis takes into consideration not only the power and energy saving due to the reduction of the switching activity in network links, but also the overhead (both in terms of power dissipation and silicon area) due to the encoding and decoding logic integrated into the NI. We show that up to 37% of power dissipation and up to 18% of energy consumption can be saved adopting the proposed encoding schemes without impacting the overall performance of the network[4].

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[10]. 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 others. Finally, this paper is concluded in Section VI.

II.OVERVIEW OF THE PROPOSAL

The fundamental idea of the projected approach is encoding the flits by they are injected into the network with the aim of minimizing the coupling switching activity and the self-switching activity in the links crossed by the flits. Moreover, coupling switching activity and self-switching activity are reliable for link power dissipation. In this paper, we suggest to the end-to-end scheme. This end-to-end encoding method takes benefit of the pipeline nature of the wormhole switching method. Note that since the same series of flits passes throughout all the links of the routing pathway, the encoding result taken at the NI may offer the same power reduction for all the links. For the proposed system, an encoder and a decoder block are additional to the NI. Apart from the header flit, the encoder encodes the leaving flits of the packet such that the power dissipated by the inter-router point-to-point link is reduced.

III. ENCODING SCHEMES

The proposed encoding scheme is designed to reduce the power dissipation by minimizing the self-switching and coupling switching activities on the links of interconnected network. The dynamic power scattered by the drives and interconnects are

$$P = [\, T_{0-->1} (C_s + C_l) + T_c C_c \,] V^2_{dd} F_{ck} \quad (1)$$

where $T_{0-->1}$ is the number of 0 → 1 transitions in the bus in two consecutive transmissions, *Tc* is the number of correlated switching between physically adjacent lines, *Cs* is the line to substrate capacitance, *Cl* is the load capacitance, *Cc* is the coupling capacitance, *V*dd is the supply voltage, and *F*ck is the clock frequency.

Four types of coupling transitions as described.

Type I transition → occurs when one of the lines switches when the other remains unchanged.

Type II transition → occurs to one line switches from low to high while the other makes transition from high to low.

Type III transition → corresponds to the case where both lines switch simultaneously.

Type IV transition → both lines do not change.

A) Power Model

The effective switched capacitance varies from type to type,

$T_Y = T_2 + T_1 - T_1^{***}$ (2)

but $T_1 = T_1^{*} + T_1^{**} + T_1$ *** (3)

substitute (3) in (2) we get

$T_Y = T_2 + T_1^{*} + T_1^{**} + T_1$ *** $- T_1^{***}$ (4)

$T_Y = T_2 + T_1^{*} + T_1^{**}$ (5)

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.

Table 1 Effect of odd inversion on change of transition types

Time	Normal			Odd Inverted		
	Type – I			Type – I		
t-1 t	00 11 01 10 T_1^{*}	00 11 00 10 01 10 00 11 T_1^{**}	01 10 11 00 T_1^{***}	00 11 11 00	00 11 01 10 00 11 01 10	01 10 10 01
	Type – II			Type – I		
t-1 t	01 10 10 01			01 10 11 00		
	Type – III			Type – I		
t-1 t	00 11 11 00			00 11 10 01		
	Type – IV			Type – I		
t-1 t	00 11 01 10 00 11 01 10			00 11 01 10 01 10 00 11		

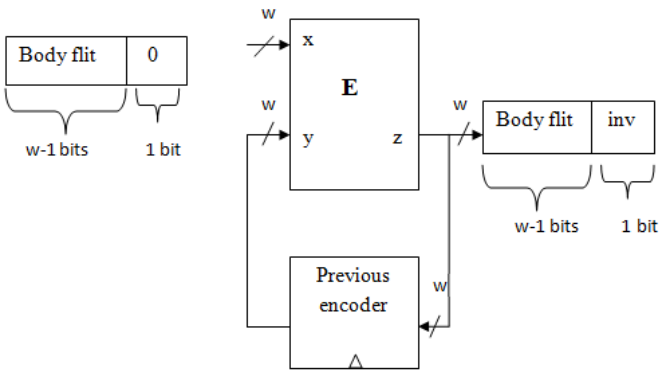

Figure 1 Encoder circuit diagram

The encoding 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. The encoding logic, each *Ty* block takes the two adjacent bits of the input flits (e.g., *X*1*X*2*Y*1*Y*2, *X*2*X*3*Y*2*Y*3, *X*3*X*4*Y*3*Y*4, 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.

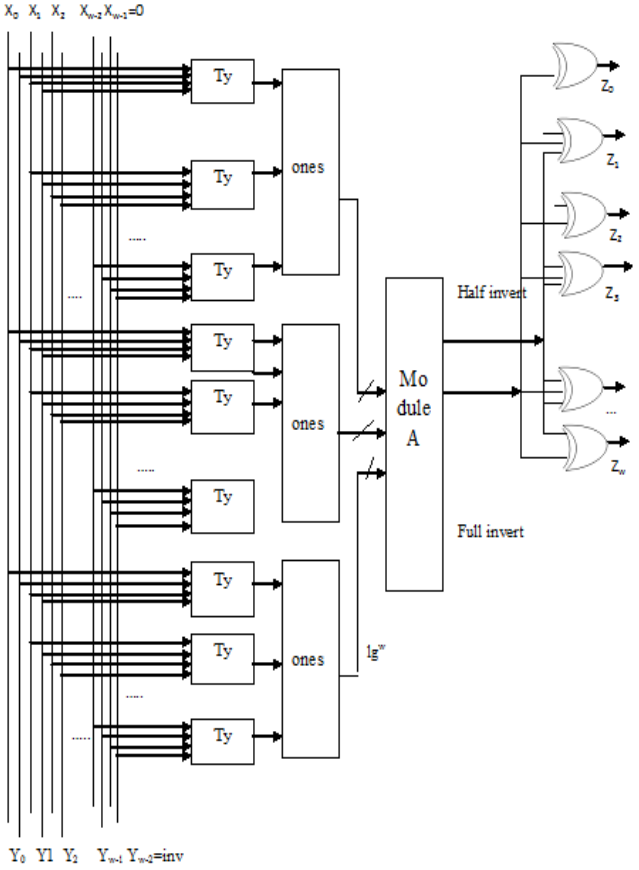

Figure 2 Internal view of encoder block

Each and every transition types (totally four types of transitions are employed but specifically the above mentioned types are used in scheme-I architecture)defines the specific transition. For example t2 defines the transitions 0110_2 (6_{10}), 1001_2 (10_{10}). In the same way $t1^*$ defines the transitions 0010_2 (2_{10}), 1101_2 (13_{10}) and t_1^{**} defines the transitions 0001_2 (1_{10}), 1110_2 (14_{10}), 0100_2 (4_{10}), 1011_2 (11_{10}).

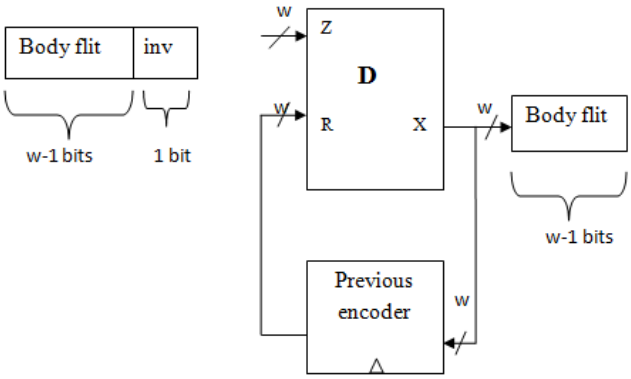

Figure 3 Decoder circuit diagram

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.

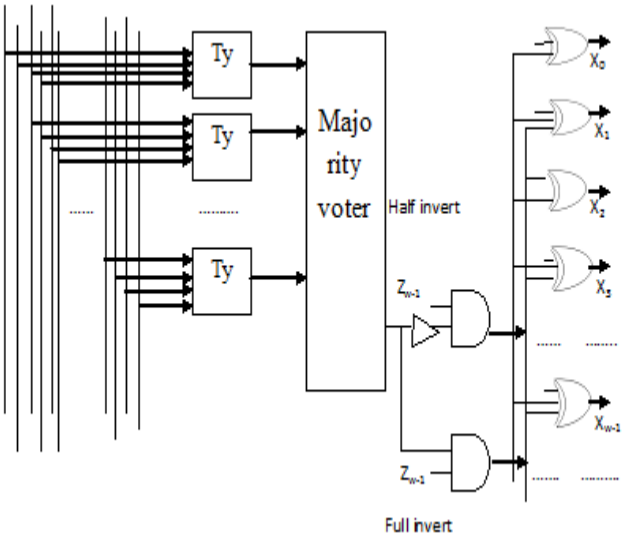

Figure 4 Internal view of decoder block

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.

Table 2 Effect of even inversion on change of transition types

Time	Normal			Even Inverted		
	Type – I			Type – II, III and IV		
t-1 t	01 10 00 11 T_1^*	00 11 01 10 10 01 11 11 T_1^{**}	00 11 01 10 T_1^{***}	00 11 11 00 Type – II	00 11 01 10 00 11 01 10 Type - IV	01 10 10 01 Type - III
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		

Defining

$$Te = T2 + T1 - T1^* \quad (6)$$

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

$$Te + Tr = w - 1. \quad (7)$$

The operating principles of this encoder are similar to those of the encoders. The proposed encoding architecture is based on the even invert condition, the full invert condition and the odd invert condition. 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 $T2$, $T1^{**}$, and $T1^{***}$ is detected for each pair bits of their inputs. This module determines if odd, even, full, or no invert action corresponding to the outputs "10,""01," "11," or "00," respectively, should be performed

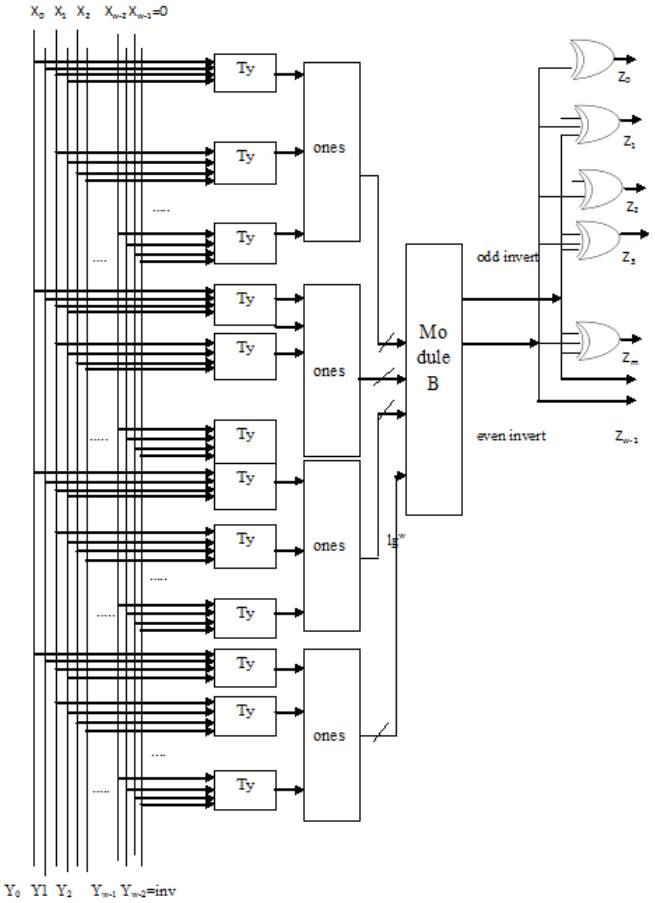

Figure 5 Modified encoder architecture

IV. RESULT

The proposed data encoding scheme have been simulated by quartus software. The power estimation models such as NI's, routers and links were considered. The link power dissipation was computed using $T_{0\rightarrow1,}$ T_1 and $T_{2.}$ The encoder and the decoder were designed in Verilog HDL described at the RTL level. The area and power of the proposed encoding schemes are compared with bus invert coding, couple driven bus invert coding and the forbidden pattern condition codes. The dynamic power dissipation will be observed rather than other power dissipations such Static, I/O, thermal power dissipations. The theoretical model says that the dynamic power dissipation is reduced due to switching activity and coupling activity. Practically, the dynamic power dissipation decrease will be observed.

Table 3 Comparison of existing and proposed encoder

ENCODER (TRADITIONAL)	ENCODER (PROPOSED-SCHEME)
19.45mw	7.87mw

Table 4 comparison of existing and proposed decoder

DECODER (TRADITIONAL)	DECODER (PROPOSED-SCHEME)
19.35mw	7.66mw

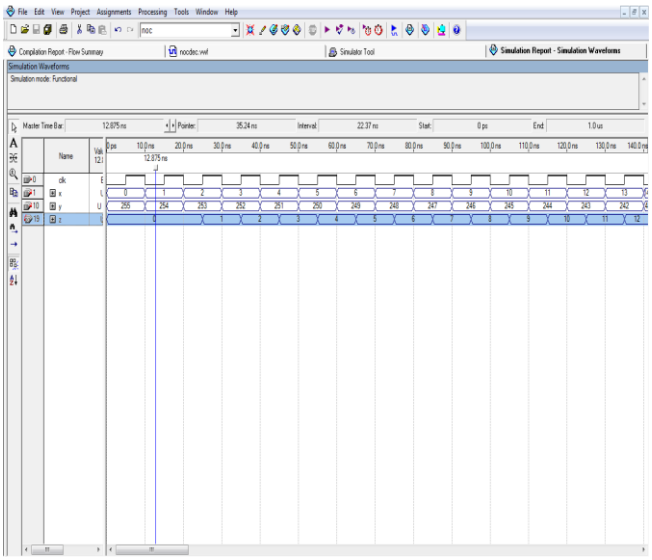

Figure 5 simulated waveform for encoder

INFERENCE:

This simulated result shows that the comparison of speed, power utilization and area consumption. By this result performance of each gate will be determined and the timing analysis of each gate will be viewed by the synthesis report.

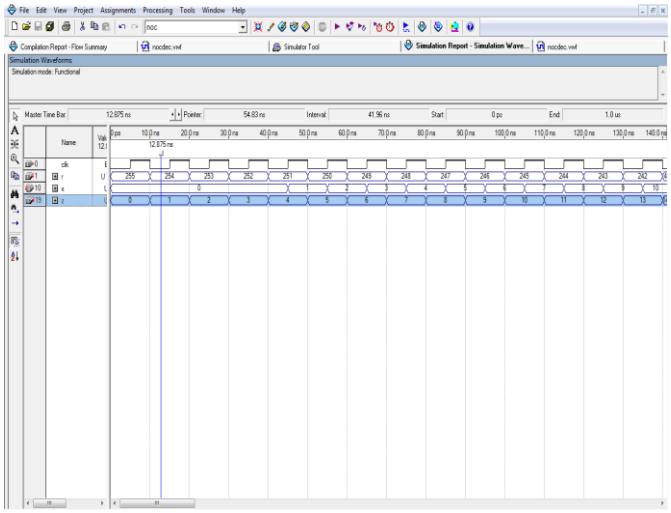

Figure 6 simulated waveform for decoder

INFERENCE:

The decoder architecture is designed by quartus simulator. The output of encoder changes with accordance of transitional activities. The dynamic dissipation will be observed.

V.CONCLUSION

The proposed new data encoding scheme has lower dynamic power dissipation. It's operation based on total number of transitions occurred on input side. Even-though area and speed wise, the proposed system is not considerable, in power dissipation wise it has no performance degradation.
When compared to classical system, the proposed encoding scheme produce desired output to users. But classical system utilizes only one scheme. The proposed system is designed using verilog HDL and synthesized using quartus II design software tool. The power dissipation is calculated using power play power analyzer tool.

This work has been extended to multi-processor and multi-computer architecture, various PE involved in data manipulation with common memory and private memory respectively. In future work a regular NOC based MAC unit will be designed. In this technique three PEs (performs addition, accumulation and multiplication respectively) will be used. This also enables the inter-link PE communication (communication between PE to PE without using routers). Thus the overall system speed is increased and power dissipation is decreased. As per the results, the proposed approach will be implemented in any one network on-chip architecture which operates under low-power.

REFERENCES

[1] M. R. Stan and W. P. Burleson, "Bus-invert coding for low-power I/O," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 3, no. 1,
pp. 49–58, Mar. 1995.
[2] 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.
[3] 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.
[4] S. Youngsoo, C. Soo-Ik, and C. Kiyoung, "Partial bus-invert coding for power optimization of application-specific systems," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 9, no. 2, pp. 377–383, Apr. 2001.
[5] Z. Yan, J. Lach, K. Skadron, and M. R. Stan, "Odd/even bus invert with two-phase transfer for buses with coupling," in *Proc. Int. Symp. Low
Power Electron. Design*, 2002, pp. 80–83.
[6] C. G. Lyuh and T. Kim, "Low-power bus encoding with crosstalk delay elimination," *IEE Proc. Comput. Digit. Tech.*, vol. 153, no. 2,
pp. 93–100, Mar. 2006.
[7] 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.
[8] 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.

