International Journal of Recent Engineering Science (IJRES),
ISSN: 2349-7157, *Volume 2 Issue 5 September to October 2015*

Comparision on Codec Design for Reducing Energy Consumption in NOC

[1]P.Mahesh , [2]DR.CH.Ravi Kumar,
[1]PG student , prakasam engineering college, kandukur, prakasam(DT), India
[2]Prakasam engineering college, kandukur, prakasam(DT), India.

Abstract

The aim of this thesis is a set of data encoding schemes are aimed at reducing power dissipated by the links of NOC. In this project three schemes are going to design by using VHDL code and simulate and compare in terms of a activity factor, power, area, delay and delay product. Coupling activity (Tc) plays the main role in the power model analysis which causes reduces power consumption in parallel NOC. An investigational result has shown the effectiveness of the proposed schemes, with respect of power dissipation and area overhead in the Network Interface (NI) as compared with data encoding which allows to save up to 54% of power dissipation and 11% of energy consumtion with out any significant performance degradation and with less than18% area overhead in the NI

Key words: *Coupling Switching Activity, Data Encoding, Inter Connect On Chip, Low Power, NOC, Power Analysis.*

I. INTRODUCTION

In accordance with Moore‘s law density of transistors doubles every 18 months and currently we all know that there are millions of FETs on a single chip is known as VLSI. Integrating these FETs combine together to perform set of operations and applications such as DSP, Communications, Robotics and medical filed. Network on chip is a communication subsystem an on integrated circuit typical between IP cores in a system on a chip (SOC). NOC Technology applied methods to on chip communication and brings notable improvement over conventional bus and crossbar interconnections. NOC improves the scalability of SOC‘s and the power efficiency of complex SOC‘s compared to other designs. A network on chip uses packets to transfer data between IP core interfaces within a chip. The NOC based system on chips imposes various design issues on the fabrication of such integrated chips. Firstly, the suitable topology for the target NOCs such that the presentation supplies and design constraints are satisfied Secondly, the design of network interfaces to access the on chip network and routers provide the physical interconnection mechanisms to transport data between processing cores. Finally, as technology scales and switching speed increases, future network on chips will become more responsive and prone to errors and faults.

On-chip communication problems are more relevant to compare to the computational relevant problems. The computational subsystem has major objectives like including cost, performance, power dissipation, energy consumption; reliability thus, the total power of a system on chip depends on the communication subsystem. In this work, we are going to reducing the power dissipation in the network links. The power dissipation in the network on chip is relevant to the power dissipation in the routers and Network Interfaces (NIs). For highly integrated electronic systems, the reduction of on-chip power dissipation is a essential one. The amount of power consumption in a NOC grows linearly by increasing the amount of bit transitions in consequent data packets sent through the interconnect architecture. By using the coding schemes we are reducing the switching activity on both wires and logic in this way we are reducing the power consumption in the NOC. The power due to self-switching activity of individual bus lines while ignoring the power dissipation owing to their coupling switching activity. Data encoding is mainly used for reducing the number of bit transition over interconnects. Bus invert (BI), Adaptive coding, Gray coding and Transition method these are the various encoding techniques used in the NOC. We are using the data encoding with gray input is mainly reducing the power dissipation on the NOC.

II. RELATED WORK AND MOTIVATION

A. Regular Network-On-Chip

The topology of a NoC specifies the physical organization of the interconnection network. It defines how nodes, switches and links are connected to each other. Topology for NoCs can be classified into two broad categories: 1) direct network topologies, in which each node (switch) is connected to at least one core (IP/PE), and 2) indirect network topologies, in which we have a subset of switches (nodes) not connected to any core (IP/PE) and performing only network operation. Both direct and indirect topology can be regular like meshes, tori, k-ary n-cubes and fat trees or

irregular topology. Some examples of regular topologies are shown in Figure 1. Most NoCs implement regular forms of network topology that can be laid out on a chip surface (a 2-dimensional plane) for example, k-ary 2-cube (where k is the degree of each dimension and 2 is the number of dimensions) commonly known as grid-based topologies. Besides the form, the nature of links adds an additional aspect to the topology. In k-ary 2-cube networks, popular NoC topologies based on the nature of link are the mesh which uses bidirectional links and torus which uses unidirectional links. For a torus, a folding can be employed to reduce long wires. In the NOSTRUM NoC presented by Millberg et al. [4], a folded torus is discarded in favor of a mesh with the argument that it has longer delays between routing nodes. Generally, mesh topology makesbetter use of links (utilization), while tree-based topologies are useful for exploiting locality of traffic. The standard regular topologies such as meshes, tori, k-ary n-cubes or fat trees as shown in Figure 1 are popularly used as the wires can be well structured in such topologies. NOSTRUM [5], SOCBUS [6] are regular 2D-Mesh architectures. In [7] torus architecture (NTNU) is described. An alternate FAT tree based structure is used in the SPIN [8] and PROPHID [9] approaches. These approaches are adequate for general purpose systems where the traffic characteristics of the system cannot be predicted statically, as in homogeneous chip-multiprocessors[10].

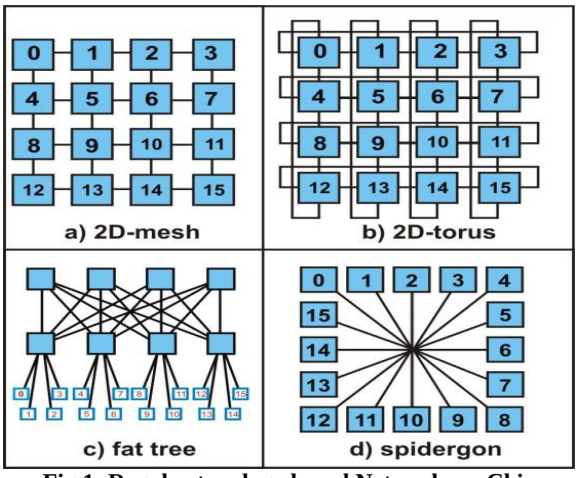


Fig 1: Regular topology based Network-on-Chip

The *k-ary tree* and the *k-ary n-dimensional* fat tree are two popular regular topologies for regular NoC. Where *k* is the degree of each dimension and *n* is the number of dimensions. The network area and power consumption scales predictably for increasing size of regular NoCs. Most NoCs prefer regular topologies that can be implemented on a 2-dimensional plane of a chip. Such topologies are generally referred as grid-based topologies. Another popular regular NoC is Octagon NoC [11]. Its basic configuration is a ring of 8 nodes connected by 12 bidirectional links which provides two-hop communication between any pair of nodes in the ring and a simple shortest-path routing algorithm can be used for packet routing. Such rings can then be connected edge-to-edge to form a larger, scalable network.

The XY routing [12] and odd-even routing are the most used deadlock free routing algorithms for the popular 2D-mesh based NoCs. They are both theoretically guaranteed to be free of deadlock and livelock. The XY routing strategy can be applied to regular two-dimensional mesh topologies without obstacles. The position of the mesh nodes and their nested network components is described by coordinates, the x-coordinate for the horizontal and the y-coordinate for the vertical position. A packet is routed to the correct horizontal position first and then in vertical direction. XY routing produces minimal paths without redundancy, assuming that the network description of a mesh node does not define redundancy. The odd-even turn model is a shortest path routing algorithm that restricts the locations where some types of turns can take place such that the algorithm remains deadlock-free. More precisely, the odd-even routing prohibits the east to north and east to south turns at any tiles located in an even column. It also prohibits the north to west and south to west turns at any tiles located in an odd column.

The accessibility of chips are growing every years. In the next several years, the availability of cores with1000 cores is foreseen[3]. Since the focus of this paper is on reducing the power dissipated by the links, here we briefly review some of the works in the area and link power reduction. Also these include some technique. There are, use of shielding [4], [5], increasing line-to-line spacing [6], [7], and repeater insertion [8]. Thus the above all the techniques
having large area overhead. Another one method is the data encoding technique it mainly focus on reducing the link power reduction. The data encoding technique is classified into two categories. In the first category is mainly concentrate on minimizing the power due to self-switching activity of each bus lines and avoid the power dissipation due to coupling switching activity. In this category, bus invert [BI] [9] and INC-XOR [10] have been proposed. When the random patterns are transmitted via these lines. On the other hand, gray code [11], T0 [12], working-zone encoding, and T0-XOR have been proposed for the case of correlated data patterns.

In this first category of encoding is not suitable for applied in deep sub-micron meter technology nodes

where the coupling capacitance is a main part of the total interconnects capacitance. This causes the power due to the coupling switching activity to become a large portion of the link power reduction. In the second category concentrate on reducing power dissipated through the reduction of the coupling switching [7]. The technique proposed a method on power effective Bus Invert. they presented a method based on Odd/Even Bus-Invert techniques. If the number of switching transitions is half of the line width means the odd inversion is performed. In [9],the number of transitions from 0 to 1 for two data packets is counted. The number of 1‘s in the data packet is larger than the half of the links means the inversion will be performed and the number of 1‘s is reduced to 0 transitions when the packets are transfer through the links. This technique is used to reducing the coupling switching. From this method, the encoder counts the Type I transitions with the weighting coefficient of one and the Type II transitions with the weighting coefficient of two. If the number of 1‘s is larger than half of the links means the inversion will be performed and it reducing the power consumption on the links. The technique proposed in [1] using the data encoding Technique. This technique illustrate if the bits are encoded before they are injected into the network with the goal of minimizing the self-switching and the coupling switching in the links. These two are the main reason for the link power dissipation. Here they are classified the encoding technique into three scheme based on the four Types. In scheme 1Using the odd inversion and scheme 2 using the both odd inversion and full inversion and scheme 3 using the odd, full and even inversion. Based on the odd, full and even inversion the power dissipation is reduced on the Network on chip (NOC) links. In this paper we present gray encoding technique, which focused on reducing the errors during the transition from transmitter to receiver and reducing the power dissipation in the links.

III. OVERVIEW OF THE PROPOSAL

The basic idea of the proposed technique is the packets are transferred through the network after that the bits are encoded. This technique is more helping to reduce the switching activity and coupling switching activity in the links traversed by the packets. This self-switching activity and coupling switching activity are responsible for the link power dissipation. Here we refer to end-to-end scheme. Based on the end to end scheme we are having a better advantage. The advantage is a pipeline nature of the wormhole switching technique. Since the same sequence of all the packets passes through all the links of the routing path. The NI may provide the same power saving for all the links. The advanced scheme, an encoder and decoder block are added to the NI. The gray input is applied for all the three scheme encoders. The gray coding technique is used for the error correction application. The encoder encodes all the leaving bits of the packets other than header bit such that the power dissipated by the inter router and point-to—point link is minimized.

A. DFS

A communication link in NoC is capable of scaling power consumption gracefully commensurate with traffic workload. This scalability allows for the efficient execution of energy-agile algorithms. Suppose that a link can be clocked at any nominal rate up to certain maximum value. This implies that different levels of power will be consumed for different clock frequencies. One option would be to clock all the links at the same rate to meet the through put requirements. However, if there was only one link in the design that required to be clocked at a high rate, the other links could be clocked at a lower rate, consuming less power. The total power consumption in an SoC is the combination of dynamic and static sources. In this paper, our focus is on the dynamic power consumption which arises from circuit switching activity, due to charging and discharging of the switched capacitance. The dynamic power consumption depends on four parameters: a switching activity factor (a), physical capacitance
(C), supply voltage (V), and the clock frequency(f)

$$P_D = \frac{1}{2}\alpha C V^2 f \quad \text{.....(1)}$$

$$f_{max} = \eta \frac{(V - V_{th})^{\beta}}{V} \quad \text{.....(2)}$$

Eq. (2) establishes the relationship between the supply voltage V and the maximum operating frequency f max, where Vth is the threshold voltage, and Z and b are experimentally derived constants.

Dynamic power consumption can be reduced by lowering the supply voltage. This requires reducing the clock frequency accordingly to compensate for the additional gate delay due to the lower voltage. The use of this approach in run-time, which is called dynamic voltages caling (DVS), addresses the problem of how to adjust the supply voltage and clock frequency of the link according to the traffic level. The basic idea is that because of high variance in network traffic, when a link is under-utilized, the linkcan be slowed down without affecting performance. However, DVS requires thousands of clock cycles during transition between voltage levels and additional hardware overhead for each link. The other way to manage power consumption is DFS. DFS only adapts the system clock frequency by setting all links in the network to the same voltage, but it does not always reduce the total energy consumption. For instance, the power consumed by a network can be

reduced by halving the operating clock frequency, but if it takes as long to forward the same amount of data, the total energy consumed will be similar.DFS is valid when the target system does not support DVS or the goal is to reduce peak or average power dissipation, indirectly reducing the chip's temperature [4]. An alternative to save link power is to add hardware such that a link can be powered down when it is not used heavily.

B. Related works

System level power management has been applied to some interconnection networks. Wei and Kim proposed chip-to-chip parallel [5] and serial [6] link design techniques where links can operate at different voltage and frequency levels. When link frequency is adjusted, supply voltage can track to the lower suitable value. Although this link was not designed for both dynamic voltage and frequency settings, previously the link architecture was used for DVS link model. There are three kinds of approaches for DVS. One is an on-line scheme which adjusts the link speed dynamically, based on a hardware prediction mechanism by observing past link traffic activities. Shang et al. [7] developed a history-based DVS policy which adjusts operating voltage and clock frequency of a link according to the utilization of link and input buffer. Worm et al.[8] proposed an adaptive low-power transmission scheme for on chip networks. They minimized the energy required for reliable communication, while satisfying QoS constraints. One of the potential problems with hardware prediction scheme is that a misprediction of traffic can be costly from performance and power perspectives.

Li et al. [9] proposed a compiler-driven approach where a compiler analyzes application code and extracts communication patterns among parallel processors. These patterns and the inherent data dependency information of the underlying code help the compiler decide the optimal voltage/frequency to be used for communication links at a given time frame. Shin and Kim[10]proposed an off-line link speed assignment algorithm for energy efficient NoC. Given the task graph of a periodic real-time application, the algorithm assigns an appropriate communications peed to each link, while guaranteeing the timing constraints of real time applications.

Soteriou et al. [11] proposed a software-directed methodology that extends parallel compiler flow in order to construct a power aware interconnection network, by combining both on-line and off-line approaches. However, current DVS techniques require not only thousands of clock cycles to shift between voltage levels, limiting their ability to respond to high frequency changes in network band width demands [12],but also additional hardware over head such as transmitter, receiver, PLL, and adaptive power supply regulator for each link. Kim et al. [12] proposed dynamic link shutdown(DLS),which powers down links intelligently when their utilizations are below a certain threshold level and a subset of highly used links can provide connectivity in the network. An adaptive routing strategy that intelligently uses a subset of links for communication was proposed, thereby facilitating DLS for minimizing energy consumption. Soteriou and Peh [2] explored the design space for communication channel turn-on/off based on a dynamic power management technique depending on hardware counter measurement obtained from the network during run-time. Chen et al. introduced a compiler-directed approach, which increases the idle periods of communication channels by reusing the same set of channels for as many communication messages as possible. Li et al. proposed a compiler-directed technique in order to turn off the communication channels to reduce NoC energy consumption. Even though it saves power significantly during idle period, it has reactivation penalty including delay and additional power consumption during a transition. Hsu saved 30% of power consumption in the MPEG core by applying DFS power management mechanism using only three frequency levels (25, 50, and 100 MHz). However, DFS was applied to a core, not to inter connection network, in a tile-based NoC architecture. To the best of our knowledge, this paper is the first proposal which addresses DFS for interconnection network. The novel contributions of our work are:

(1)a DFS link proposal for on-chip interconnection network which offers not only fast response time reducing the frequency transition penalty, but also reduces hardware cost, as compared to DVSlink, suitable for system integration;

(2)use of narrow control period, as compared to conventional DVS control, reducing misprediction penalty that occurs in a hardware prediction scheme by adjusting the frequency more often;

(3) implementation of a variable frequency link that judiciously adjusts link frequency based on the link utilization estimation, reducing power consumption. 3.Variable frequency link The clock boosting router was proposed to increase through put and reduce latency of an adaptive worm hole router[3].The key idea of clock boosting mechanism is the use of different clocks in a head flit and body flits because body flits can continue advancing along the reserved path that is already established by the head flit, while the head flit requires the support of complex logic, increasing critical path. Thus, it reduces latency and increases throughput to far outer by applying faster clock frequency to a boosting clock in order to forward body

flits. DFS only adapts the system clock frequency by setting all links in the network to the same voltage. The clock boosting router can be modified to support a variable frequency link that is applicable for DFS with negligible hardware cost and fast response time to frequency changes. In addition, the operating frequency of a system is not limited by the critical path of the route decision logic because it only changes clock frequency for the body flit transmission. Thus, the proposed method not only provides variable frequency link but also increases interconnection network performance. Also, fast response time of the clock domain variations makes it possible to use narrow control period for DFS, where clock frequency is adjusted more frequently. An example of the proposed variable frequency link using clock boosting mechanism. The system has multiple clock frequencies represented by Fi. Link controller selects boosting clock frequency for the clock boosting router among supported clock frequencies by using link utilization level. Fig. 2 shows the time-space diagram for variable frequency links. In this example, the link supports three different frequencies (F1, F2, and F3). A conventional variable frequency link changes clock domain for the entire control period, while the proposed variable frequency link applies different clock frequencies only to the body flit transmission. The original clock frequency (F1 in this example) is still used for the head flit transmission as well as for idle cycles. In this paper, we use multiple clock frequencies (1, 2 and 4) as the boosting clock frequency for the body flit transmission, in order to reduce implementation complexity.

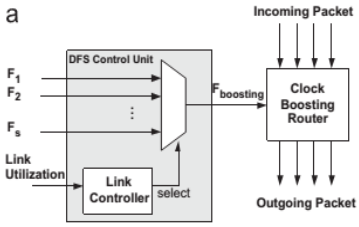


Fig2. Architecture of a DFS link

IV. PROPOSED ENCODING SCHEMES

In this section, we tend to gift the proposed encoding scheme whose goal is to scale back power dissipation by minimizing the coupling transition activities on the links of the interconnection network. Allow us to initial describe the ability model that contains completely different elements of power dissipation of a link. The dynamic power dissipated by the interconnects and drivers is

$$P = [T0\rightarrow1\ (Cs + Cl\ ) + TcCc]\ v_{dd}^2 Fck \qquad (1)$$

where $T0\rightarrow1$ is the number of $0 \rightarrow 1$ 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, Vdd is the supply voltage, and Fck is the clock frequency.

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
$t-1$	Type II			Type I		
t	01, 10			01, 10		
	10, 01			11, 00		
$t-1$	Type III			Type I		
t	00, 11			00, 11		
	11, 00			10, 01		
$t-1$	Type IV			Type I		
t	00, 11, 01, 10			00, 11, 01, 10		
	00, 11, 01, 10			01, 10, 00, 11		

One will classify four forms of coupling transitions as represented in. a Type I transition happens once one among the lines switches once the opposite remains unchanged. in a very Type II

transition, one line switches from low to high whereas the opposite makes transition from high to low. a Type III transition corresponds to the case wherever each lines switch at the same time. Finally, in a very Type IV transition each lines do not changed. These Types shown in Table-I.

The effective switched capacitance varies from Type to Type, and hence, the coupling transition activity, Tc, may be a corresponding weighted sum of various forms of coupling transition contributions. Therefore

$$T_c = K_1 T_1 + K_2 T_1 + K_3 T_3 + K_4 T_4 \quad (2)$$

where *Ti* is the average number of Type *I* transition and *Ki* is its corresponding weight. According to [26], we use *K*1 = 1, *K*2 = 2, and *K*3 = *K*4 = 0. The relevance chance of Types I and II for a random set of knowledge is 1/2 and 1/8, respectively. Using (2), one could categorical (1) as

$$P = [T0\rightarrow1\ (Cs + Cl\ ) + (T1 + 2T2)\ Cc]\ v^2_{dd}\ Fck. \quad (3)$$

According to [3], *Cl* are often neglected

$$P \propto T_{0\rightarrow1}Cs + (T1 + 2T2)Cc. \quad (4)$$

Here, we have a tendency to calculate the prevalence chance for various Type of transitions. Regarding contemplate take into account that flit (t−1)and flit (t) seek advice from the previous flit that was transferred via the link and therefore the flit that is about to meet up with the link, respectively. We think about solely 2 adjacent bits of the physical channel. Sixteen completely different mixtures of those four bits may occur. Note that the primary bit is the worth of the generic ith line of the link, wherever because the second bit represents the worth of its (i+1)th line. The range of transitions for varieties I, II, III, and IV square measure 8, 2, 2, and 4, respectively. For a random set of information, every of those sixteen transitions has an equivalent chance. Then the occurrence probability for Types I, II, III, and IV are 1/2, 1/8, 1/8, and 1/4, respectively. within the remainder of this section, we have a tendency to design 3 data encoding schemes for reducing the dynamic power dissipation of the network links beside a attainable hardware implementation of the decoder.

A. Scheme I

In scheme I, we specialize in reducing the numbers of Type I transition(by changing them to Type III and Type IV transitions) and Type II transitions (by changing them to Type I transition). The scheme compares the current incoming flit data with the previous flit data will cause the link power reduction.

1) ***Power Model:*** If the flit is odd inverted before being transmitted, the dynamic power on the link is

$$P' \alpha\ T'_{0\rightarrow1} + (K_1 T'_1 + K_2 T'_2 + K_3 T'_3 + K_4 T'_4)\ C_c \quad (5)$$

where $T'_{0\rightarrow1}, T'_1, T'_2, T'_3, T'_4$ are the self–transition activity, and the coupling transition activity of Types I, II, III, and IV, respectively. Table I reports, for every transition, the relationship between the coupling transition activities of the flit once transmitted as is and once its bits are odd inverted. Data are organized as ,the primary bit is the worth of the generic ith line of the link, whereas the second bit represents the worth of its (i+1)th line. for every partition, the primary (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. within the case of Type I transitions, the inversion ends up in one in every of Types II, III, or Type IV transitions. above all, the transitions indicated as T1* , T1**, T1*** within the table convert to Types II, III, and IV Transitions , respectively. Also, we have a tendency to have $T'_{0\rightarrow1} = T_{0\rightarrow1(odd)} + T_{0\rightarrow1(even)}$ where odd/even refers to odd/even lines. Therefore, (5) can be expressed as

$$P\ \alpha\ (T_{0\rightarrow0(odd)} + T_{0\rightarrow1(even)})\ C_s$$

$$+[K_1(T_2 + T_3 + T_4) + K_2 T_1^{***} + K_3 T_1^{*} + K_4 T_1^{**}]C_c. \quad (6)$$

Thus, if $P > P'$, it is convenient to odd invert the flit before transmission to cut back the link power dissipation. Using (4) and (6) and noting that Cc/Cs = 4 [26], we have a tendency to acquire the following odd invert condition

$$\frac{1}{4}T_{0\rightarrow1} + T1 + 2T_2 > \frac{1}{4}(T_{0\rightarrow0(odd)} + T_{0\rightarrow1(even)}) + T_2 + T_3 + T_4 + 2T_1^{***}$$

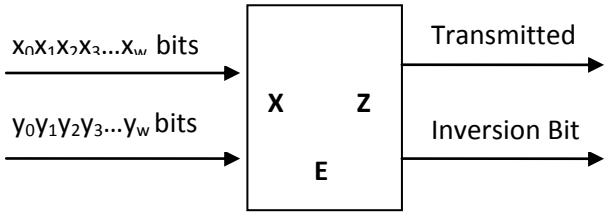


Fig3. Encoder architecture scheme I.

Also, since $T_{0\to1} = T_{0\to0(odd)} + T_{0\to1(even)}$, one might write

$$\frac{1}{4}T_{0\to1(odd)}+T1+2T_2 > \frac{1}{4}T_{0\to0(odd)}+T_2+ T_3+T_4+2T_1^{***}. \quad (7)$$

which is that the actual condition to be wont to decide whether or not the odd invert needs to be performed. Since the terms T0→1(odd) and T0→0(odd) weighted with an element of 1/4, for link widths bigger than sixteen bits, the misprediction of the invert condition will not exceed 1.2% on the average [23]. Thus, we are able to approximate the precise condition as

$$T1 + 2T_2 > T_2 + T_3 + T_4 + 2T_1^{***}. \quad (8)$$

Of course, the utilization of the approximated odd invert condition reduces the effectiveness of the encoding scheme because of the error induced by the approximation however it simplifies the hardware implementation of encoder. Now, defining

$$T_x = T_3 + T_4 + T_1^{***}$$

and

$$T_y = T_1 + T_2 - T_1^{***} \quad (9)$$

one will rewrite (8) as

$$T_y > T_x \quad (10)$$

Assuming the link dimension of w bits, the overall transition between adjacent lines is w −1, and hence

$$T_y + T_x = w - 1 . \quad (11)$$

Thus, we are able to write (10) as

$$T_y > \frac{(w-1)}{2} \quad (12)$$

This presents the condition wont to confirm whether or not the odd inversion should be performed or not.

2) ***Proposed Encoding Architecture:*** The proposed encoding architecture, that relies on the odd invert condition outlined by (12), is shown in Fig. 1. we tend to take into account a link width o w bits. If no encryption is employed, the body flits classified in w bits by the NI and transmitted via the link. In our approach, one little bit of the link is employed for the inversion bit, that indicates if the flit traversing the link has been inverted or not. additional specifically, the NI packs the body flits in w−1 bits [Fig. 1]. The encryption logic E, that is integrated into the NI, is answerable for deciding if the inversion ought to occur and activity the inversion if required. The generic diagram shown in Fig. 1 is that the same for all 3 encoding schemes proposed during this paper and solely the block E is totally different for three schemes. The encoder block E consider the first two flits from incoming data.Aftre that finds the total no.of. transitions occurred in the incoming flits, by using these values measure the thresold condition value. If thresold value satisfies the odd inverstion condition(18) then second flit i.e. $Y(y_0y_1y_2y_3\ldots y_w)$ is odd inverted. Otherwise transmitted same data. For the encrypition perpous sends the invertion bit inv as inv=1 if inversion performed or inv=0 no inversion perforemed. The decoder cicuit simply decodes the received flit by using invertion bit which gives the original transmitted data.

B. Scheme II

Within the proposed encoding scheme II, we tend to create use of each odd (as mentioned previously) and full inversion. the complete or full inversion operation converts Type II transitions to Type IV transitions. The scheme compares this information with the previous one to determine whether or not the odd, full, or no inversion of this information will make to the link power reduction

1) ***Power Model****:* allow us to indicate with *P*, *P*', and *P*" the power dissipated by the link once the flit is transmitted with no inversion, odd inversion, and full inversion, respectively. The odd inversion makes to power reduction when *P*'< *P*" and *P*' < *P*. The power *P*" is given by [23]

$$P'' \ \alpha \ T_1+2T_4^{**} \quad (13)$$

Neglecting the self-switching activity, we tend to get the condition *P*' < *P*" as [see (7) and (13)]

$$T_2 + T_3 + T_4 + 2T_1^{***} < T_1+2T_4^{**} \quad (14)$$

Therefore, mistreatment (9) and (11), we are able to write

$$2(T_2 - T_4^{**}) < 2T_y - w + 1 . \quad (15)$$

Based on (12) and (15), the odd inversion condition is obtained as

$$2(T_2 - T_4^{**}) < 2T_y - w + 1 , \ T_y > \frac{(w-1)}{2}. \quad (16)$$

Similarly, the condition for the full inversion is obtained from *P*" < *P* and *P*" < *P*' . The difference *P*" < *P* is

$$T_2 > T_4^{**} \quad (17)$$

Therefore, mistreatment (15) and (17), the complete inversion condition is obtained as

$$2(T_2 - T_4^{**}) > 2T_y - w + 1, \quad T_2 > T_4^{**}. \qquad (18)$$

When none of (16) or (18) is glad, no inversion are performed.

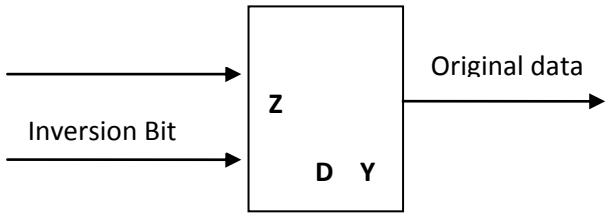


Fig. 4. Decoder architecture scheme II.

2) ***Proposed Encoding Architecture:*** The operation principles of this encoder is same just like those of the encoder implementing scheme I. The proposed encryption design, that relies on the odd invert condition of (16) and Therefore the full invert condition of (18). Here again, The encoder block consider the first two flits from incoming data.Aftre that finds the total no.of. transitions occurred in the incoming flits, by using these values measure the thresold condition value. If thresold value satisfies the odd inverstion condition(16) then second flit i.e. $Y(y_0y_1y_2y_3\ldots y_w)$ is odd inverted. The extra threshold value for full invertion condition(18) is added to the scheme-I to make the full invertion operation which makes the better power reduction. Otherwise transmitted same data. For the encrypition perpous sends the invertion bit inv as inv = "01" if odd inversion performed or inv = "11" for full inversion or inv = "00" no inversion perforemed.The decoder circuit diagram is shown in Fig. 2. The decoder takes inputs as encoder transmitted data and the invertion bit. Decoder simplily decodes the received flit by using inv bit if Inv = "01" odd inversion or inv = "11" full inversion or Inv = "00" No inversion perfomed to the data.

C. Scheme III

within the proposed encoding scheme III, we tend to add even inversion to scheme II. The explanation is that odd inversion converts a number of Type I (T1***) transitions to Type II transitions. As may be determined from Table II , if the flit is even inverted, the transitions indicated as T1**/ T1*** within the table are regenerate to Type IV/ Type III transitions. Therefore, the even inversion might scale back the link power dissipation still. The scheme compares the present information with the previous one to make your mind up whether or not odd, even, full, or no inversion of the present information will make to the link power reduction.

1) ***Power Model:*** allow us to indicate with *P*", *P*', *P*"' and the and the facility dissipated by the link once the flit is transmitted with no inversion, odd inversion, full inversion, and even inversion, severally. just like the analysis given for scheme I, we are able to approximate the condition *P*"'<*P* as

Table II: 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, 01, 10	00, 11	00, 11	00, 11, 01, 10	01, 10
t	00, 11	01, 10, 00, 11	01, 10	11, 00	00, 11, 01, 10	10, 01
	T1*	T1**	T1***	Type III	Type IV	Type II
$t-1$	Type II			Type I		
t	01, 10			01, 10		
	10, 01			00, 11		
$t-1$	Type III			Type I		
t	00, 11			00, 11		
	11, 00			01, 10		
$t-1$	Type IV			Type I		
t	00, 11, 01, 10			00, 11, 01, 10		
	00, 11, 01, 10			10, 01, 11, 00		

$$T1 + 2T_2 > T_2 + T_3 + T_4 + 2T_1^* \qquad (19)$$

Defining

$$T_e = T_1 + T_2 - T_1^* \quad (20)$$

we acquire the condition $P''' < P$ as

$$T_e > \frac{(w-1)}{2} \quad (21)$$

Similar to the analysis given for theme II, we are able to approximate the condition

$$T_2 + T_3 + T_4 + 2T_1^{***} > T_2 + T_3 + T_4 + 2T_1^* \quad (22)$$

Using (9) and (20), we are able to rewrite (22)

$$T_e > T_y. \quad (23)$$

Also, we tend to acquire the condition $P''' < P''$ as [see (13) and (19)]

$$T_2 + T_3 + T_4 + 2T_1^* > T_1 + 2T_4^{**}. \quad (24)$$

Now, define

$$T_r = T_3 + T_4 + T_1^* \text{ and } T_e = T_2 + T_1 - T_1^*. \quad (25)$$

Assuming the link dimension of w bits, the overall transition between adjacent lines is w − one, and hence

$$T_e + T_y = w - 1. \quad (26)$$

Using (26), we are able to rewrite (24) as

$$2(T_2 - T_4^{**}) < 2T_e - w + 1. \quad (27)$$

The even inversion results in power reduction once $P'''< P$,

$P''< P$, and $P'''< P''$. Based on (21), (23), and (27), we obtain

$$T_e > \frac{(w-1)}{2},\ T_e > T_y,\ 2(T_2 - T_4^{**}) < 2T_e - w + 1 \quad (28)$$

The full inversion results in power reduction once when $P''<P$, $P'' < P'$, and $P'' < P'''$. Therefore, mistreatment (18) and (27)

$$2(T_2 - T_4^{**}) > 2T_e - w + 1,\ (T_2 > T_4^{**}),$$

$$2(T_2 - T_4^{**}) > 2T_y - w + 1 \quad (29)$$

Similarly, the condition for the odd inversion is obtained from $P' < P$, $P' < P''$ and $P' < P'''$. supported (16) and (23), the odd inversion condition is glad once

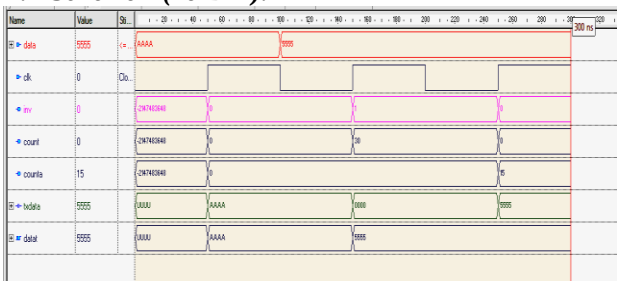
$$2(T_2 - T_4^{**}) < 2T_y - w + 1,\ (T_e < T_y),\ T_y > \frac{(w-1)}{2}. \quad (30)$$

When none of (28), (29), or (30) is glad, no inversion can be performed.

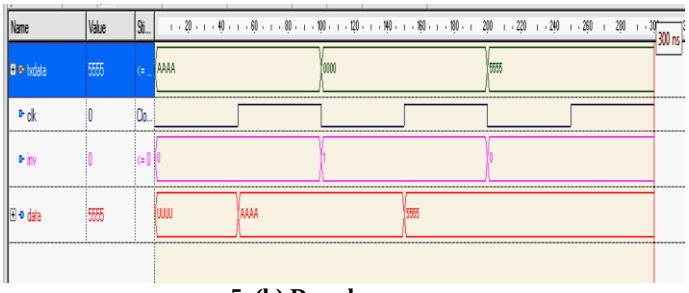
2) ***Proposed Encoding Architecture*:** The operative principles of this encoder are just like those of the encoders implementing Schemes I and II. The proposed encoding architecture, that is predicated on the even invert condition of (28), the complete invert condition of (29), and also the odd invert condition of (30). The wth little bit of the antecedently encoded body flit is indicated by inv that shows if it had been even, odd, or full inverted (inv = 1) or left because it was (inv = 0). the primary stage of the encoder determines the transition Types whereas the second stage is calculate threshold condition. If thresold value satisfies the coresponding inverstion can be done to the second flit i.e. $Y(y_0y_1y_2y_3\ldots y_w)$. The extra threshold value for Even invertion condition(28) is added to the scheme-II to make the Even invertion operation which makes the better power reduction. Otherwise transmitted same data. For the encrypition perpous sends the invertion bit inv as inv = "01" if odd inversion performed or inv = "11" for full inversion or inv = "10" for Even inversion or inv = "00" no inversion perforemed.

V. RESULTS AND DISCUSSION

A. Scheme-I (16-BIT):

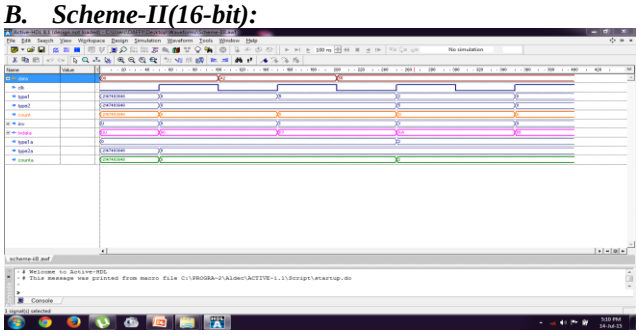


5 (a) Encoder



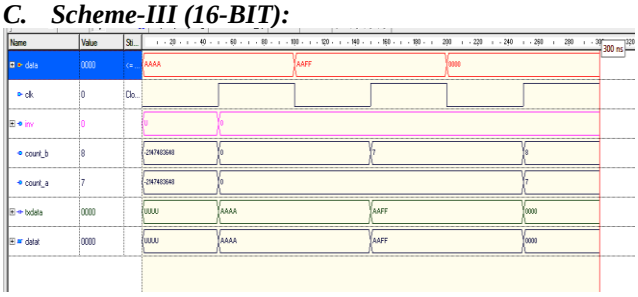
5 (b) Decoder

B. Scheme-II(16-bit):

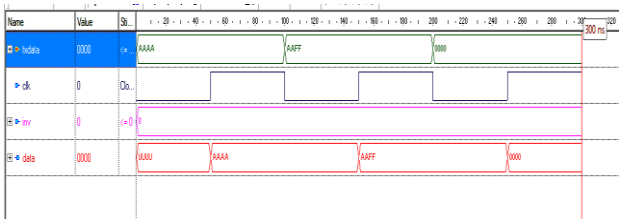


6 (a) Encoder

C. Scheme-III (16-BIT):



7 (a) Encoder



7 (b) Decoder

VI. CONCLUSION

In this work, the three encoding techniques are implemented for reducing the transition activity in the NOC. This encoding schemes are 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. The proposed encoding schemes are agnostic with respect to the underlying NOC architecture in the sense that our application does not require any modification neither in the links nor in the links. The proposed architecture is coded using VHDL language and is simulated and synthesized using cadence software. Overall, the application scheme allows savings up to 42% of power dissipation and with less than 5% area overhead in the NI compared to the data encoding scheme**.** In the future, the Network on Chip (NOC) implementation using different types of router technique will be analyzed. Comparison on many encoding techniques such as gray encoding techniques and Data Shuffling will be analyzed in which the area, delay, power and the performance of the NOC will be investigated and use for high speed applications. By using Data Shuffling Method one can achieve better results from Scheme-I rather than Scheme-II and Scheme-III.

REFERENCES

[1] Nima Jafarzadeh, Maurizio Palesi, Member, IEEE, Ahmad Khademzadeh, and Ali Afzali-Kusha, Senior Member, IEEE, "Data Encoding Techniques for Reducing Energy Consumption in Network-on-Chip", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol. 22, No. 3, March 2014.

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

[3] 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.

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

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

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

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

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.