International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 22
November 2018

Design and Implementation of Encoding Schemes for Minimiging Switching and Coupling Transitions for Link Dissipation

N. Divya Jyothi [1] and T. Vidya[2]

[1]*M.Tech (VLSI), Department of ECE, BVC College of Engineering, Rajahmundry,A.P, India*

(E-mail: nambudivya789@gmail.com)

[2]*Associate Professor, Department of ECE, BVC College of Engineering, Rajahmundry, A.P, India*

(E-mail: vidyatall@gmail.com)

Abstract -- As the technologies shrink, the power dissipated by the communication subsystem, namely, the routers, NIs, and links increases. This paper discusses a set of data encoding and consumption in links of the NoC. Their application does not require any modification of the routers and link architectures. This proposed system is transparent and general with respect to the NoC fabric. Finally the area and power efficient encoder and decoder are designed.

Index Terms – NoC, Data encoding, Low power, on chip interconnection.

I. INTRODUCTION

Now a day the communication system plays a very important role in our life. Power consumption and delay in the interconnection of the elements plays a vital role in the communication system. Reducing the interconnection reduces the power levels of the system. The encoding scheme concentrates on reducing link power dissipation. For this purpose two categories of encoding techniques can be used. The first one reduces the power by self switching activity of individual bus lines. In this bus invert, according to M.R.Stan and W.P.Burleson and W.P. Burleson[1], gray code according to Sumant Ramprasad et al.[2] and T0-XOR according to E.Musoll et al.[3] have been proposed.

S.E Lee and N. Bagherzadeh[4] described, the encoded bits were isolated using shielding wires. This technique effectively decreases the coupling switching activity. Although the technique decreases the power consumption considerably, it increases the data transfer time.

N. Rajesh e al.[5] proposed the power reduction and impact on area is very less. The data encoding scheme describes the wormhole switching techniques and work on an end-to-end basis (i.e. flits are encoded by the (NI) before they are injected in the network). Also proposed data encoding scheme called self and coupling driven bus invert (SCDBI) reduces the switching power by 34.64% without any significant degradation in area and performance.

The related work for the paper is discussed in section II, In section III, the encoder schemes are discussed. Section IV presents the simulation result of the paper. Finally Section V presents the conclusion of the paper.

II. RELATED WORK

In existing system, the number of transitions from 0 to 1 for two consecutive flits (the flit that just traversed and the one which is about to traverse the link) is counted. If the number is larger than half of the link width, inversion will be performed to reduce the number of 0 to 1 transitions when the flit is transferred via the link. This technique is only concerned about the self switching without worrying about the coupling switching. Generally the coupling capacitance in the state-of-the-art silicon technology is considerably larger (e.g., four times) compared with the self- capacitance, and hence, should be considered in any scheme proposed for the link power reduction.

This scheme deals with reducing the coupling switching. In this method, a complex encoder counts the number of Type I (Table I) transitions with a weighting coefficient of one and the number of Type II transitions with the weighting coefficient of two. If the number is larger than half of the link width, the inversion will be performed. In addition to the complex encoder, the technique only works on the patterns whose full inversion leads to the link power reduction while not considering the patterns whose full inversions may lead to higher link power consumption. Therefore, the link power reduction achieved through this technique is not as large as it could be. This scheme was also based on the hop-by-hop technique. The coding technique reduces the coupling switching activity by taking the advantage of end-to-end encoding. It is based on lowering the coupling switching activity by eliminating only Type II transitions.

II. PROPOSED SYSTEM

In the proposed system an additional decoder is designed for scheme 3 which reduces the switching activity of the decoder.

A. Encoder

The general block diagram of the encoder is shown in Figure. 1. In this, input flit is given in W-1 bit with one bit used for indicating the presence of encoding or not. If the bit is 1 then data is encoded else no encoding takes place for the data.

e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 22
November 2018

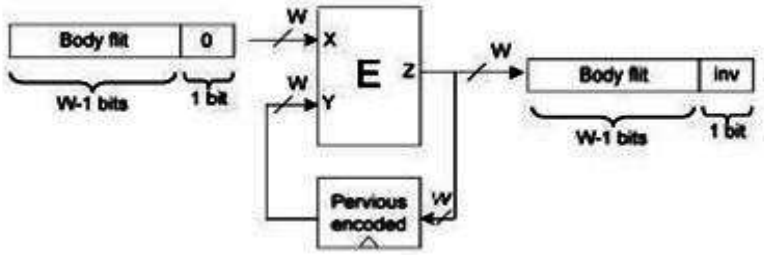

Figure 1: General Circuit Diagram of Encoder

The circuit diagram of an encoder for all the schemes remains the same. The internal block diagram of block " E" in encoder scheme changes according to each scheme. There is a need for different block for different type of encodings of the data.

B. Decoder:

The general circuit diagram of decoder is shown in fig.2 below

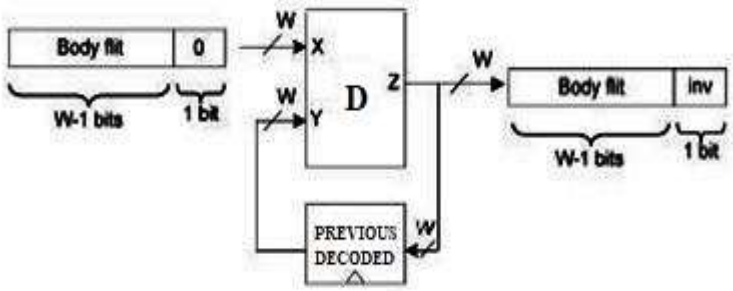

Figure 2 : General circuit diagram of Decoder

The circuit diagram of decoder for all the schemes remains same. The internal block diagram of block D of decoder circuit changes according to each scheme. In decoders the inverse operation of encoder takes place. There is a need of only one block Ty to determine which action has to be taken. The tabulation for the type1, type2, type3, and type4 in odd inversion and even inversion is shown in Tables 1 and 2.

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	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		

Table 2 : EFFECT OF EVEN INVERSION ON CHANGE OF TRANSITIONS TYPES

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

C SCHEME I:

In the proposed encoding scheme I, we make use of only odd inversion, the odd inversion converts type I and type II transitions to type III and type IV transitions. The scheme compares the current data with the previous one to decide whether the odd or no inversion of the current data can give rise to link power reduction.

1)Scheme 1 Encoding Architecture :

The operating principal of the encoder based on the odd invert condition, the w_{th} bit of the previously encoded body flit is indicated by inv which defines if (inv=1) it was odd inverted or (inv=0) then no inversion takes place. In the encoding logic the T_y structure takes the two adjacent bits of the input flits and sets its output to "1" if any of the transition types is detected. The subsequent stage of the encoder is a majority vector which determines total number of transitions and the inversion is performed on odd bits. The decoder circuit simply inverts the received flit when the inversion bit is high.

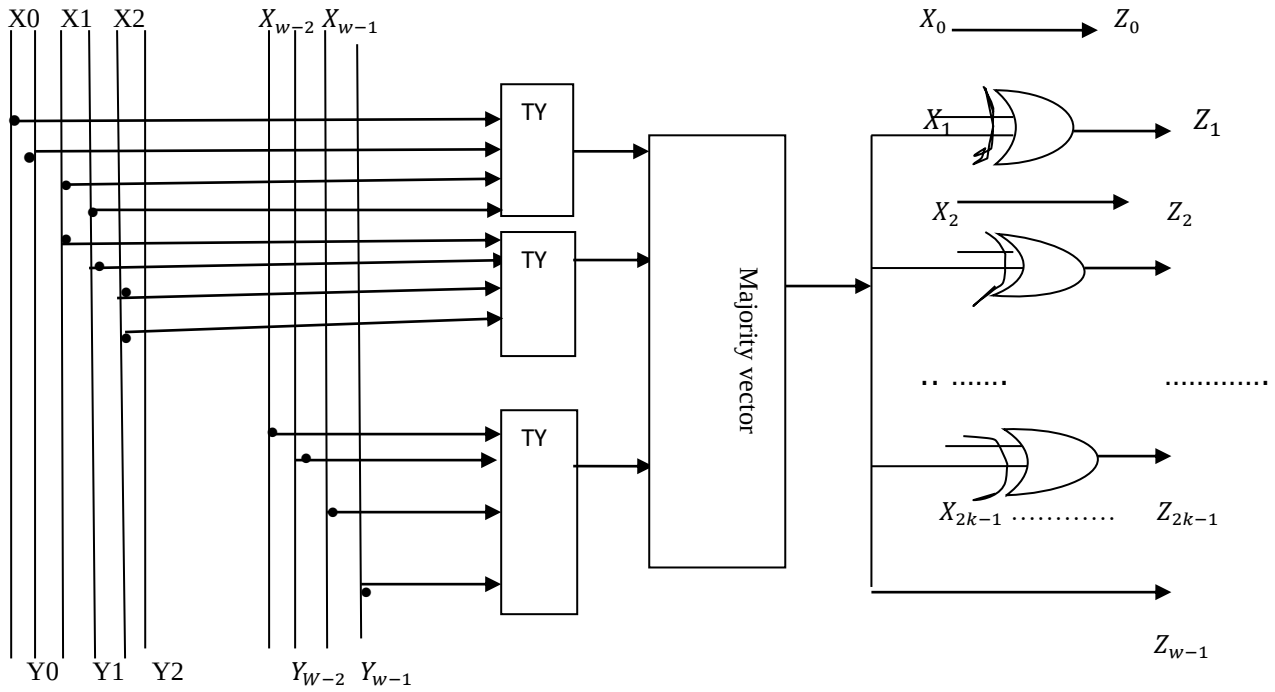

Figure 3 Encoder architecture Scheme I

D SCHEME II:

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

1) Scheme 2 Encoding Architecture:

The operating principles of this encoder are similar to those of the encoder implementing scheme. I. The scheme II encoding architecture, which is based on the odd invert condition and the full invert condition, is shown in Fig.4. The wth bit of the previously encoded body flit is indicated with in which defines if it was odd or full inverted (inv-1) or left s it was (inv-0). In this encoder, in addition to the Ty block in the Scheme I encoder, we have the T2 and blocks which determine if the inversion based on the

transition type T2 and should take place for the link power reduction. The second stage is formed by a set of 1s blocks which count the number of 1s in their inputs.

The output of these blocks has the width of log2w. The output of the 1s block determines the number of transitions that odd inverting of pair bits leads to the link power reduction.

The middle 1s block identifies the number of transitions whose full inverting of pair bits leads to the link power reduction. Finally, the bottom 1s block specifies the number of transitions whose full inverting of pair bits leads to the increased link power. Based on the number of 1s for each transition type, Module A decides if an odd invert or full invert action should be performed for the power reduction. In case no invert action needs to take place, none of the output is set to "1". Module A can be implemented using full-adder and comparator blocks.

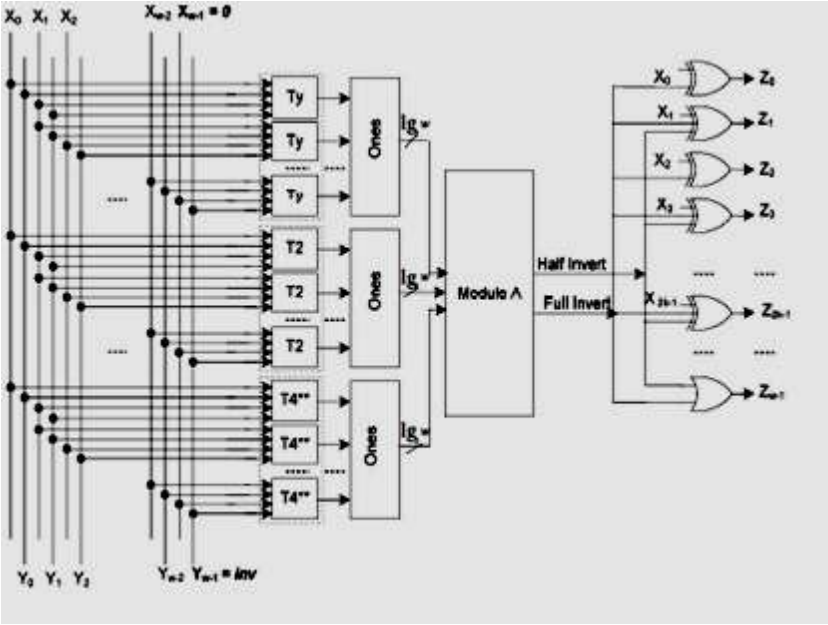

Figure 4: Encoder architecture Scheme II

2) Scheme 2 decoding Architecture:

The internal block diagram for block D of scheme 2 decoder is showing fig 5 The w bits of the incoming (previous) body flit are indicated by Zi (Ri), i =0,1,...,w−1. The wth bit of the body flit is indicated by inv which shows if it was inverted (inv =1) or left as it was (inv =0). For the decoder, we only need to have the Ty block to determine which action has been taken place in the encoder

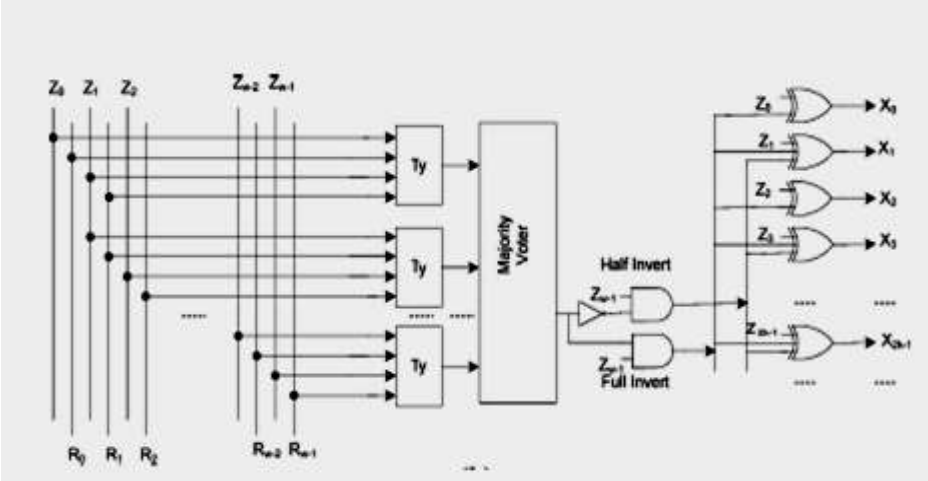

Figure 5 : Decoder architecture of Scheme II

Based on the outputs of these blocks, the majority voter block checks the validity of the inequality given by (12). If the output is "0" ("1") and the inv=1, it means that half (full) inversion of the bits has been performed. Using this output and the logical gates, the inversion action is determined. If two inversion bits were used, the overhead of the decoder hardware could be substantially reduced.

E SCHEME 3:

In the encoding Scheme III, even inversion to Scheme II is added. The reason is that odd inversion converts some of type I transitions to type II transitions. As can be observed from table II, if the flit is even inverted, the transitions indicated as in the table is converted to type IV/ type III transitions. Therefore, the even inversion may reduce the link power dissipation as well. The scheme compares the current data with the previous one to decide whether odd, even, full, or no inversion of the current data can give rise to link power reduction

I) Scheme 3 Encoding Architecture:

The operating principles of this encoder are similar to those of the encoders implementing Schemes I and II. The proposed encoding architecture, which is based on the even invert condition, and the odd invert condition, is shown in fig 6.

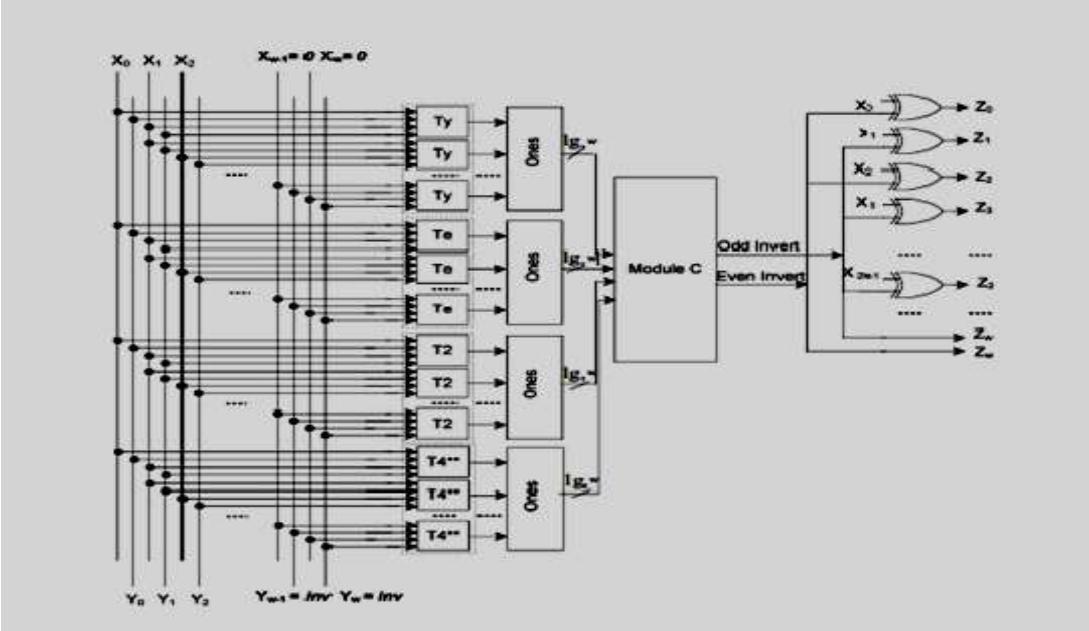

Figure 6: Encoder architecture Scheme III

The wth bit of the previously encoded body flit is indicated by inv which shows if it was even, odd, or full inverted (inv=1) or left as it was (inv=0). The first stage of the encoder determines the transition types while the second stage is formed by a set of 1s blocks which count the number of ones in their inputs. In the first stage, the Te blocks is added if any of the transition types of T2 or T4∗∗, is detected for each pair bits of their inputs. For these transition types, the even invert action yields link power reduction. Again, we have four ones blocks to determine the number of detected transitions for each Ty, Te, T2, T4∗∗, blocks. The output of the Ones blocks are inputs for Module C. This module determines if odd, even, full, or no invert action corresponding to the outputs “10,” “01,” “11,” or “00,” respectively, should be performed. Similar to the procedure used to design the decoder for scheme II, the decoder for scheme III can be designed.

1) Scheme 3 decoding Architecture :

The internal block diagram for block D in the generic circuit diagram for scheme 3 is as shown fig 6

International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 22
November 2018

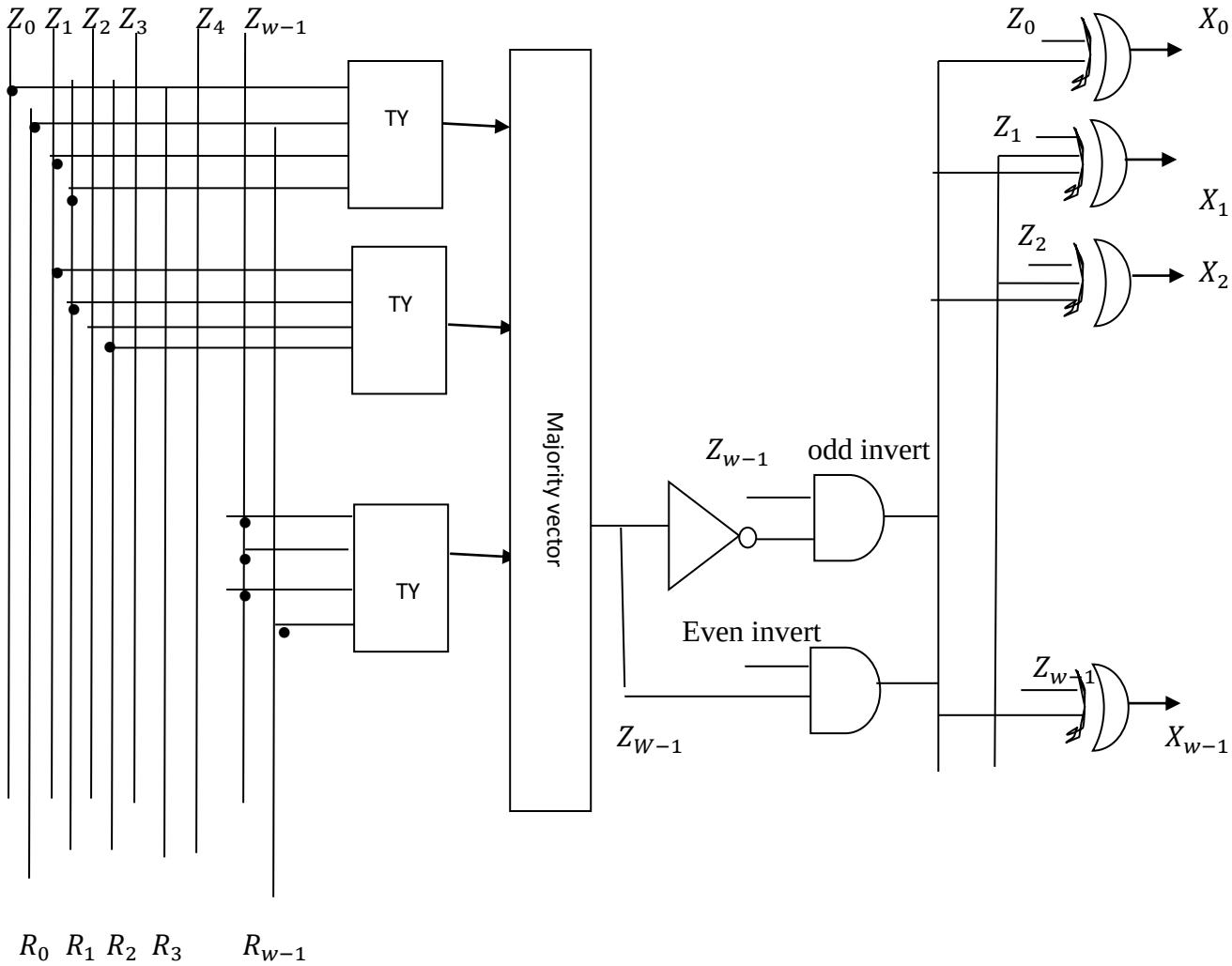

Figure 7 : Decoding architecture of scheme III

The w_{th} bit of the body flit is indicated by inv which shows if it was inverted (inv =1) or left as it was (inv = 0) i.e. no inversion has taken place. 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 vector block checks the validity of the inequality. If the output is " 1" ("0") and the inv =1, it means that even (odd) inversion of the bits has been performed

IV SIMULATION RESULTS

The below figure shows the RTL schematic of scheme I

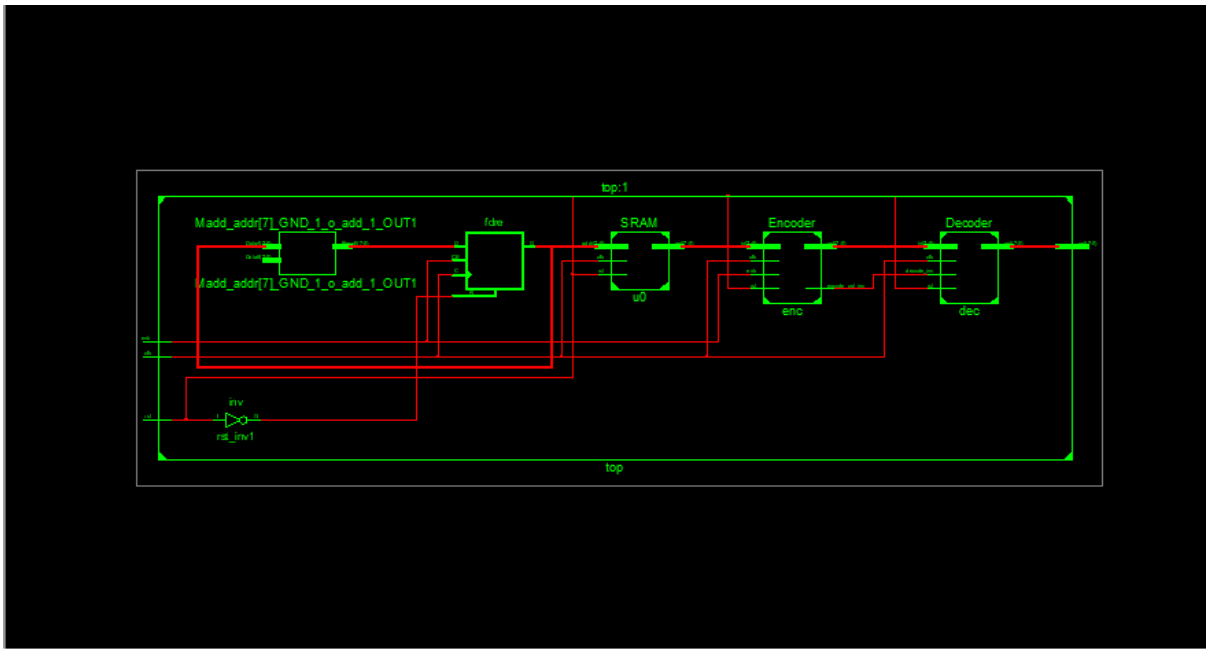

Figure 7: RTL schematic of scheme I

Technology schematic of scheme I

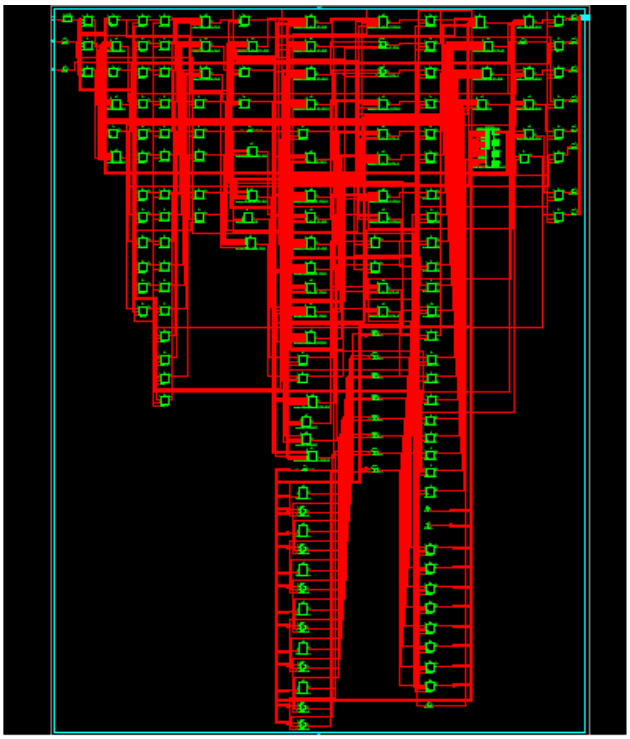

Figure 8 Technology schematic of scheme 1

The below figure shows simulation output of scheme I

International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 22
November 2018

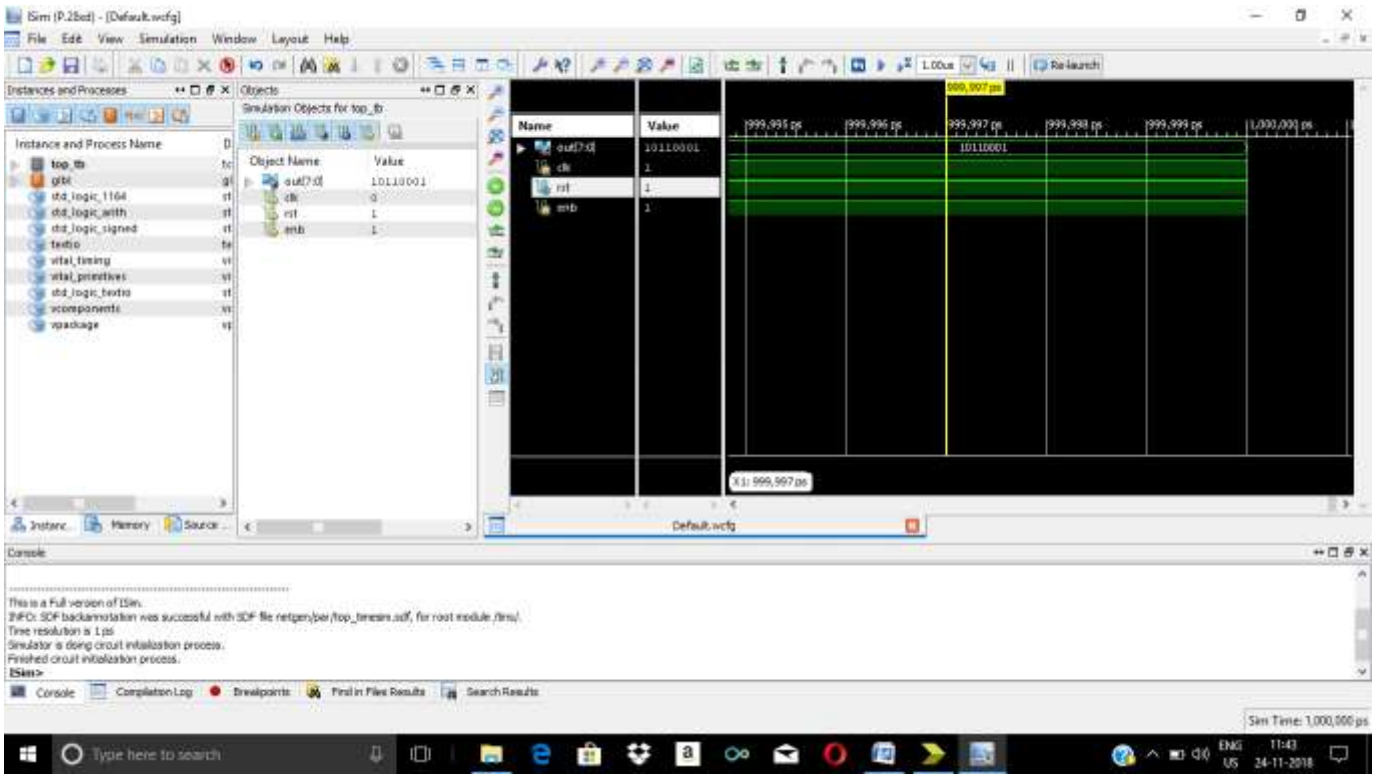

Figure 9 Simulation output of scheme I

The below figure shows the power analyzer output of scheme I

The below figure shows the Technology schematic of scheme II

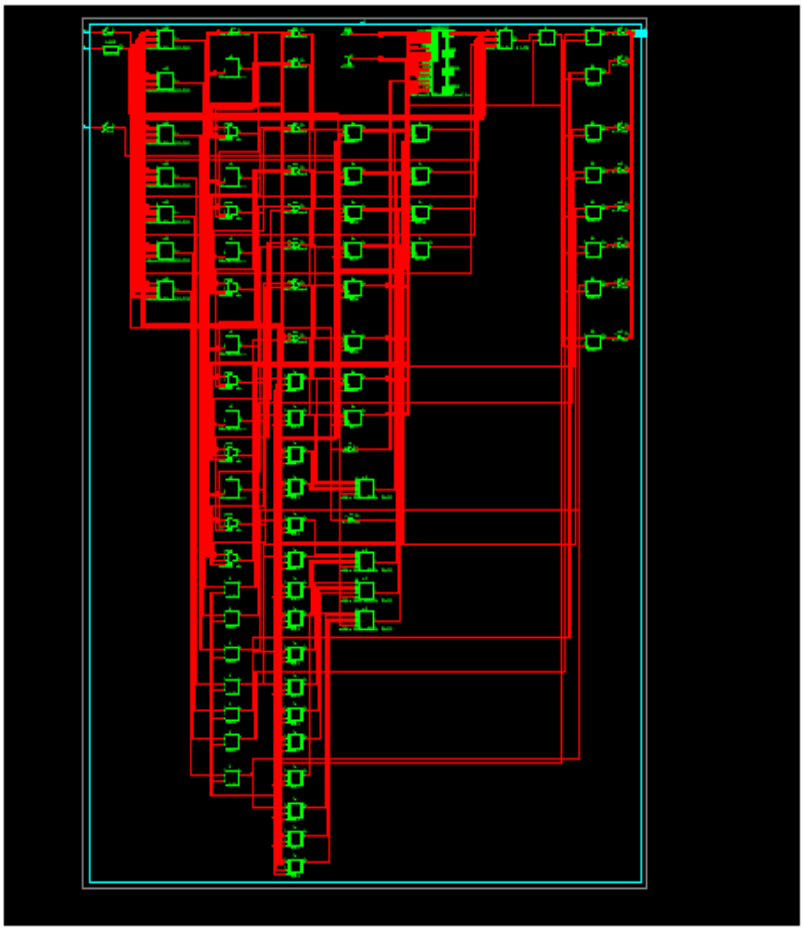

Figure 12 Technology schematic of scheme II

International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 22
November 2018

The below figure shows the simulation result of scheme II

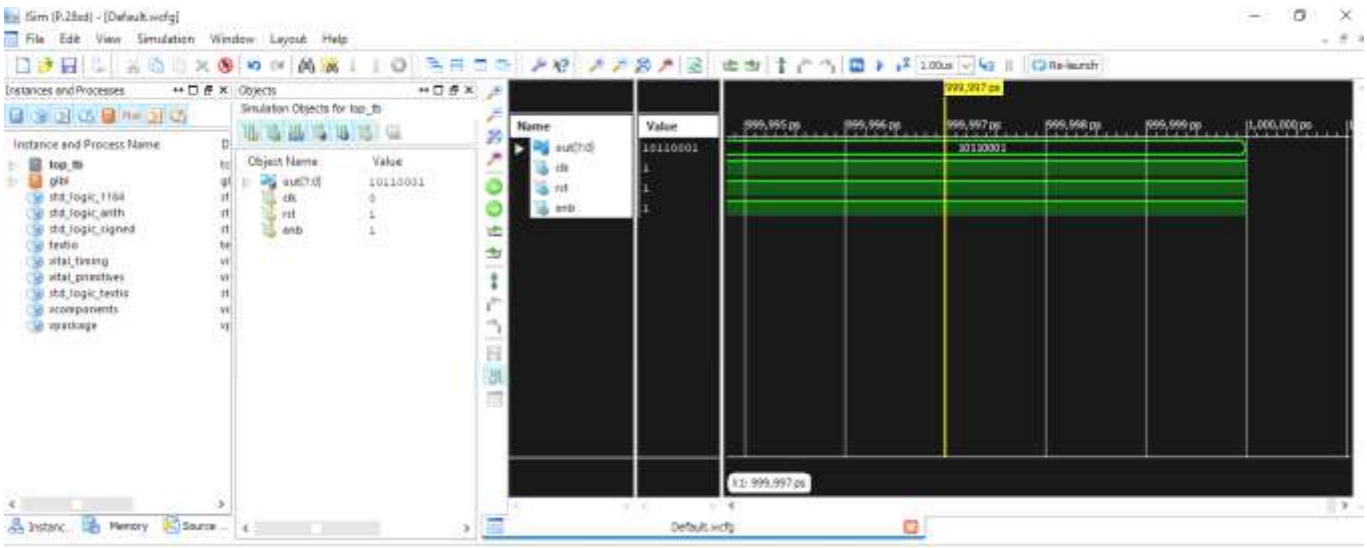

Figure 13 : simulation output

The below figure shows the Power analyzer output

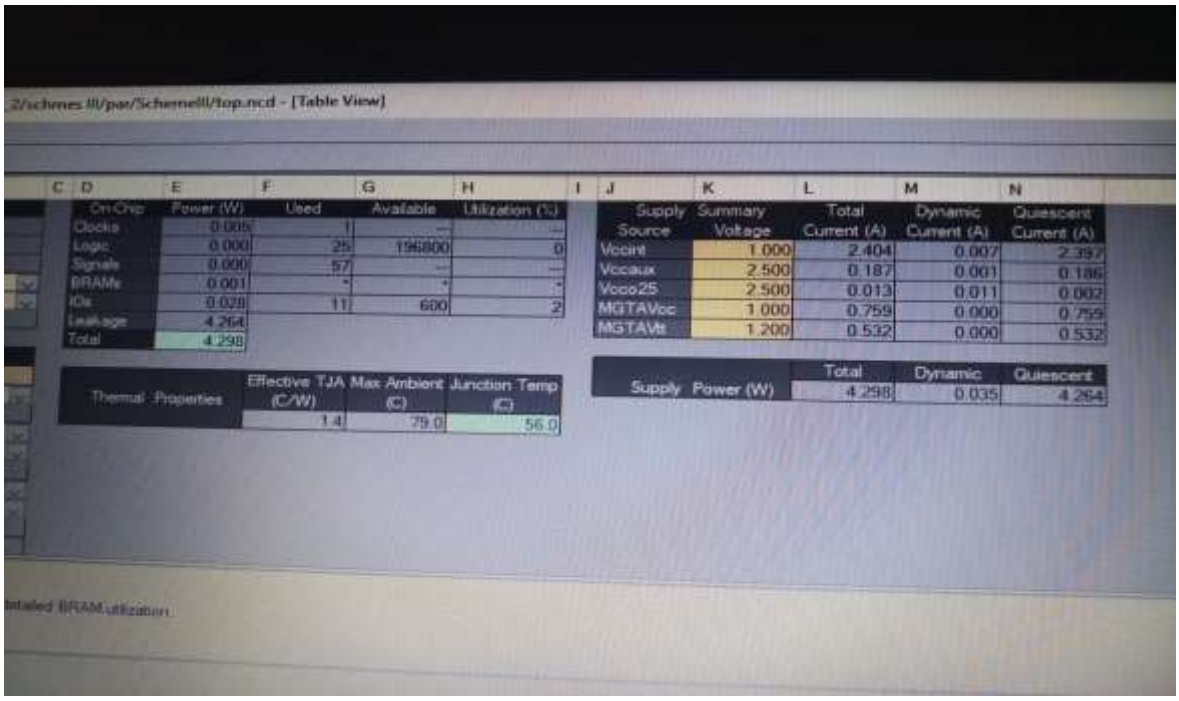

Figure 14: Power analyzer output for scheme II

The below figure shows the RTL schematic of scheme III

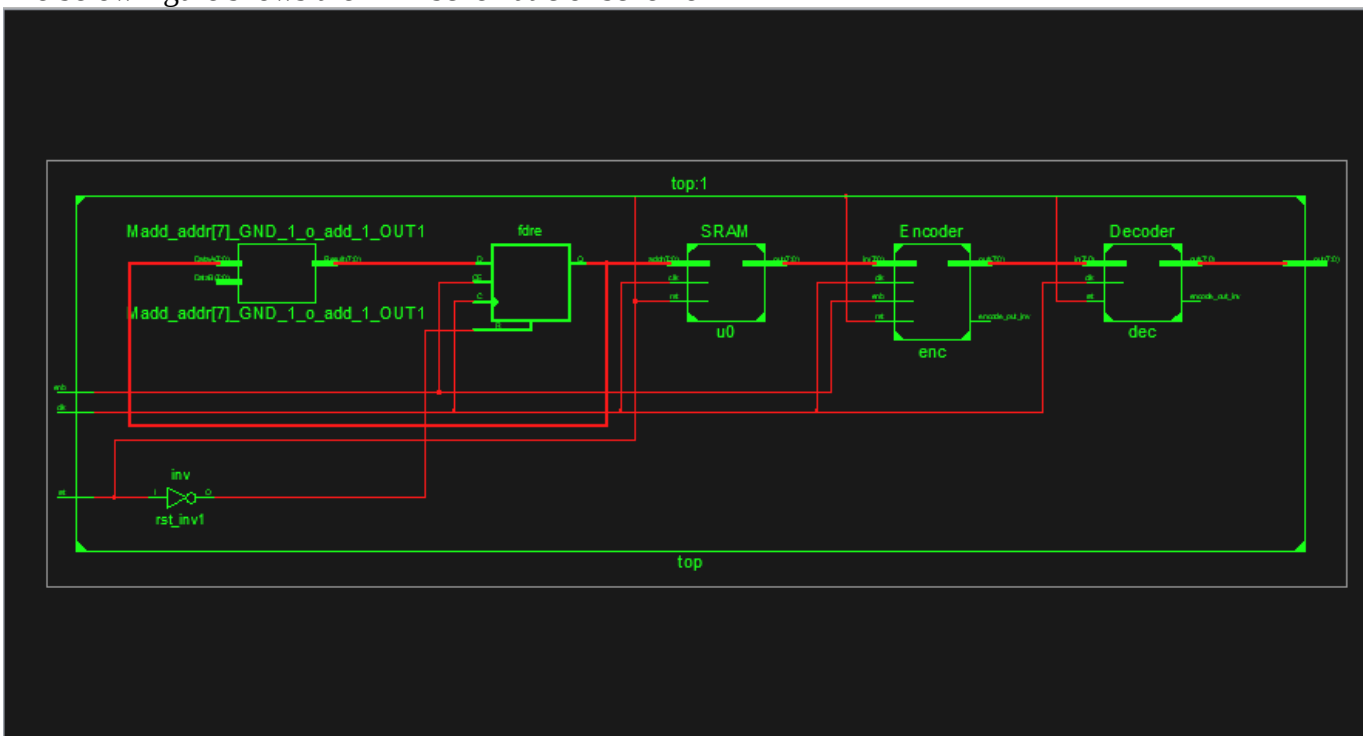

Figure 15 : RTL schematic of scheme III

The below figure shows theTechnology schematic of scheme III

International Journal of Research
Available at https://pen2print.org/index.php/ijr/
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 05 Issue 22
November 2018

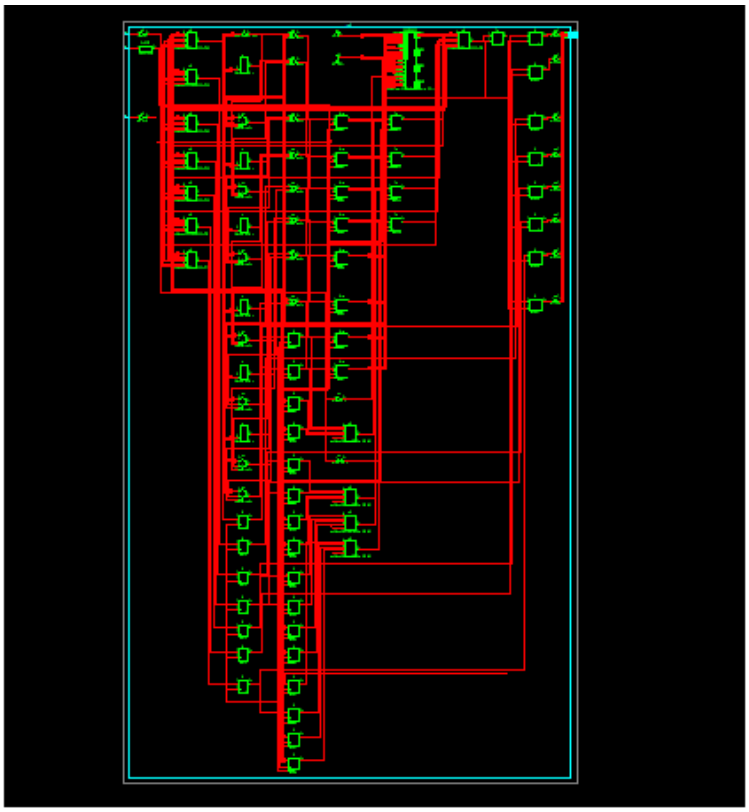

Figure 16: Technology schematic of scheme III

The below figure shows the simulation output of scheme III

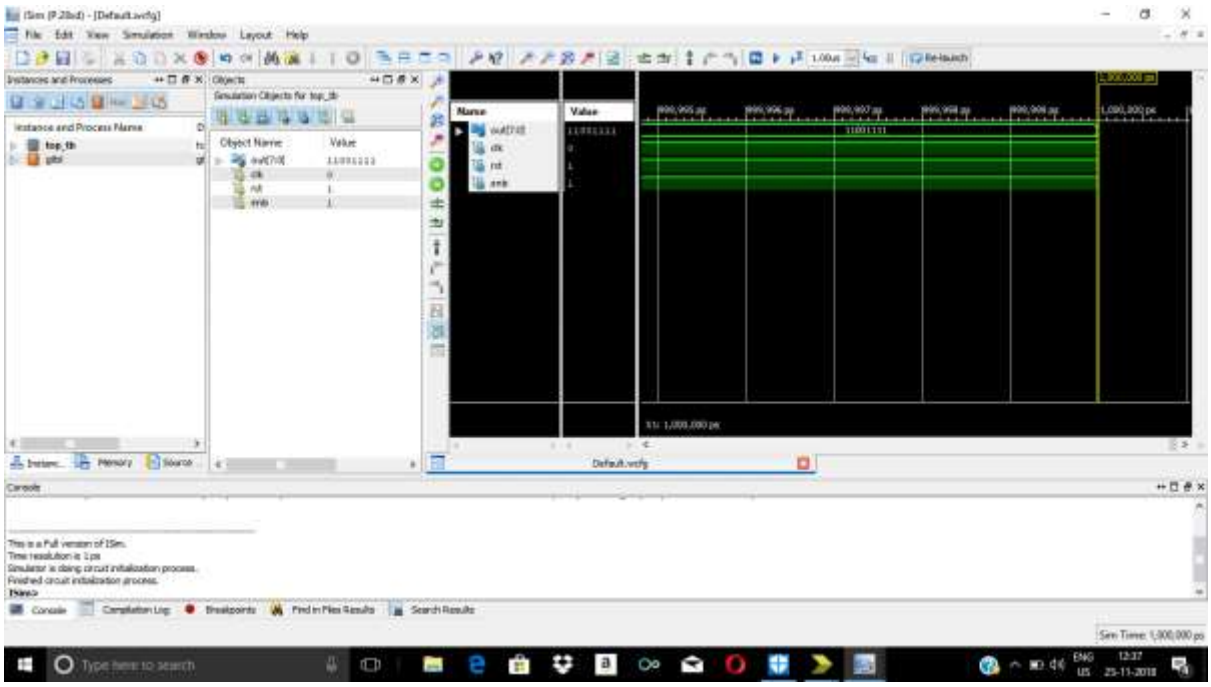

Figure 17 : SIMULATION OUTPUT FOR SCHEME III

The below figure shows the power analyzer output for scheme III

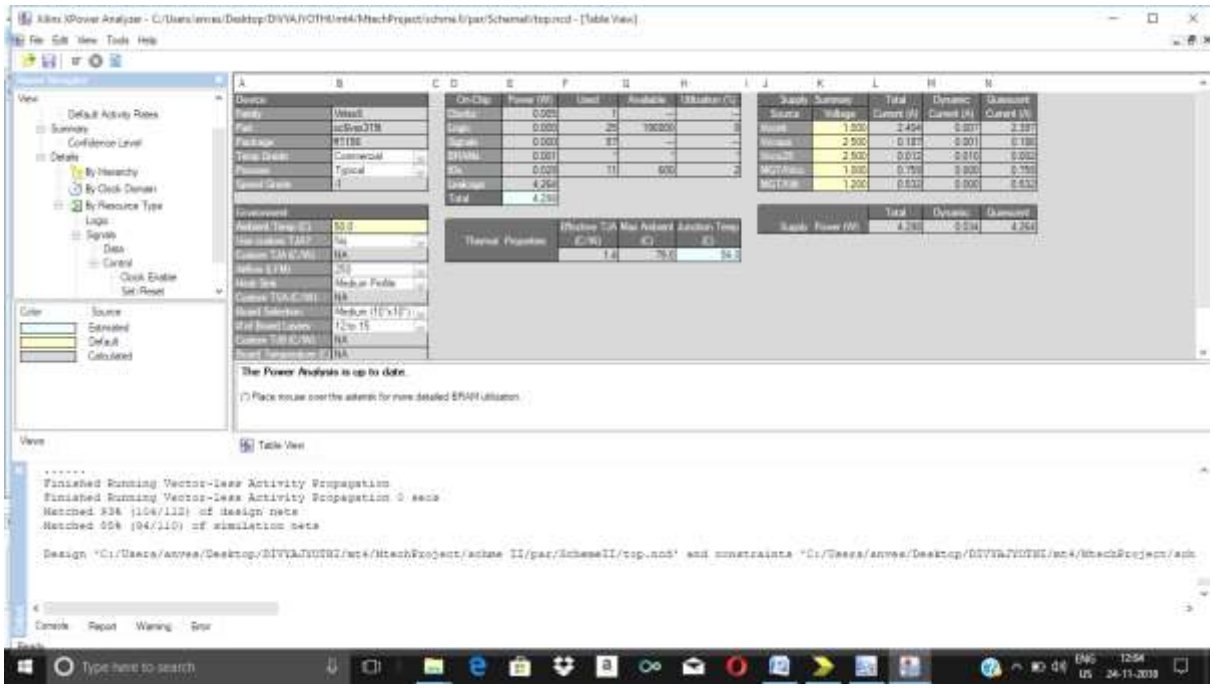

Figure 18 : Power analyzer output for scheme III

V CONCLUSION

A set of data encoding and decoding schemes have been discussed whose goal is to reduce the power consumption in links of the NoC. This proposed system is transparent and general with respect to the NoC fabric. The function of encoder and decoder is verified in simulation. Hence, we conclude that the encoder and decoder design is power efficient.

REFERENCES

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

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

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

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

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

[6] Vittal and M. Marek-Sadowska, "Crosstalk reduction for VLSI," IEEE Trans. Computer.-Aided Design Integrated. Circuits Syst., vol. 16, no. 3, 290–298, Mar. 1997

[7] 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. Computer.-Aided Design Integrated. Circuits System., vol. 25, no. 5, pp. 821–836, May 2006.

[8] 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.

[9] 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 Auto. Conf. Asia South Pacific, vol. 2. Jan. 2005, pp. 729–734

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

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

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

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

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

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

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

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

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

[19] .G. Ascia, V. Catania, M. Palesi, and A. Parlato, "Switching activity reduction in embedded systems: A genetic bus encoding approach," IEE Proc. Computer. Digit. Tech., vol. 152, no. 6, pp. 756–764, Nov. 2005.

[20] R. Siegmund, C. Kretzschmar, and D. Muller, "Adaptive Partial Bus-invert encoding for power efficient data transfer over wide system buses," in Proc. 13th Symp. Integrated. Circuits System. Design, Sep. 2000