World Applied Sciences Journal 32 (12): 2362-2366, 2014
ISSN 1818-4952

DOI: 10.5829/idosi.wasj.2014.32.12.1380

Coding Techniques for Low Power Applications

[1]K. Senthamil Selvan, [2]B. Lakshmidhevi and [3]R. Varatharajan

[1]Department of ECE, Dhanalakshmi College of Engineering, Chennai, India
[2]Department of CSE, Anna University, Chennai, India
[3]Department of Electronics and Communication Engg, Sri Lakshmi Ammal Engineering College, India

Abstract: VLSI circuit design consuming low power become necessary for present technologies. one way of reducing dynamic power consumption in a CMOS circuit is to reduce the number of switching activity. In fact, self-switching and coupling switching activity is responsible for link power dissipation. The aim of low power data encoder is to transmit the data on the bus with reduced switching activity. In this paper the improved form of bus invert coding is implemented which is based on odd line and even line invert condition. Odd and even line conditions are used to minimize the unwanted transitions on the data path which provides better results in terms of power dissipation and switching activity when compared to traditional bus-invert method.

Key words: Bus-invert coding · Self-switching activity · Coupling switching activity · Low power · Data encoding · Link power

INTRODUCTION

As semiconductor technology scales to nanometer technology, A critical constraint of digital systems is power consumption due to widely used portable systems. Among the architectural components at the system level, buses that interconnect subsystems are important components, which consumes significant power. In VlSI on-chip bus lines loaded with large capacitance. The amount of power dissipated by bus is directly proportional to the product of average number of logic transitions and bus capacitance. The better way of reducing power dissipation on bus is to encode the data with reduced switching activity. The rest of the paper is organized as follows. Power classification is discussed in section II, Switching activity is briefly discussed in section III, Related work is discussed in section IV, Proposed work is discussed in detail in section V, section VI and VII gives results and conclusion respectively.

Power Classification: The total power consumption in the VLSI chip includes the following, dynamic power, short circuit power, static power and leakage power. The total power consumption can be evaluated from the following equation,

$$p = \frac{1}{2c} v_{dd}^2 fN + Q_{sc} V_{dd}^2 fN + I_{leak} v_{dc} \quad (1)$$

Switching Activity Power: Switching activity power is the power required to charge and discharge circuit nodes. Node capacitance is represented by C. The factor N is the switching activity, i.e. the number of gate output transitions per clock cycle. In addition to voltage and physical capacitance, switching activity is the third factor that determines the dynamic power consumption. Only the switching activity contributes to dynamic power even though a chip contains high amount of physical capacitance.

Short Circuit Power: Power dissipation occurs during on output transitions due to current flowing from the supply voltage to the ground, this is called as short-circuit current. The term Q➔SC represents the quantity of charge carried by the short-circuit current per transition.

Static Power: Static power is also called as leakage power, leakage current which is primarily determined by the technology used in its construction and consists of reverse bias current and sub-threshold current. In a MOS transistor bias current formed in the reverse direction in the parasitic nodes in between source and drain diffusions and the bulk region [1].

Corresponding Author: K. Senthamil Selvan, Department of ECE, Dhanalakshmi College of Engineering, Chennai, India.
E-mail: senthamilkselva@gmail.com.

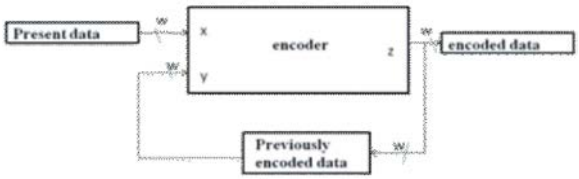

Fig. 1: Encoder architecture

Switching Activity: Switching activity can be classified into two types such as self-switching activity and coupling switching activity. Self-switching activity is defined as transitions on the capacitance between a bus line and the substrate (ground). The capacitance between the two adjacent links is defined as coupling transitions. Coupling transition in-between two adjacent links can be categorized as four types such as type-1, type-2, type-3 and type-4. Type-1 transition occurs if any one of the two lines changes state. Type-2 occurs if one wire switches from high to low and other switches from low to high.Type-3 occurs when both the line switches from low to high or high to low.Type-4 occurs when both line do not switch.

Related Work: Some of the existing solutions to reduce the coupling switching activity are as follows, increasing line to line spacing and non-uniform wire placement is a method to decrease the physical coupling capacitance between buslines, busordering, busswizzling, repeater staggering, skewing signal transition timing of adjacent lines and passive shielding is a well-known technique to reduce coupling effects is to insert static(ground or power) lines between bus data lines. This technique is known as passive shielding. Passive shielding eliminates the worst-case switching condition when two adjacent lines oppositely switch, resulting in a better worst-case delay [2]. A good alternative to passive shielding is active shielding. In this method a shield is allowed to switch depending upon switching pattern of its adjacent bus lines. Among all these method data encoding provides better result. This method have to be modified to target a power cost function, namely, weighted switching activity.

Proposed Work: The basic idea of the proposed approach is encoding the data before transmitting on the link with the goal of minimizing the self-switching activity and the coupling switching activity in the links traversed by the data. For the proposed scheme, an encoder and a decoder block are added to the each intellectual property on the chip. The decoder just does the opposite work of the encoder block. The general encoder architecture is shown in Fig. 1. The encoding logic changes according to the odd or even or full or no invert condition. Consider a link width of w bits. If no encoding is used, the incoming data is transmitted as it is via the link.

In this proposed work the incoming data is encoded according to encoding logic and transmitted via the link and the encoded data is decoded at the receiving end. In this approach one bit of the data is used for the inversion bit, which indicates if the data traversing the link has been inverted or not. w bits are the concatenation of w-1 payload bits and a "0"bit,represents the first input of the encoder, while the previous encoded data represents the second input of the encoder. The decoder circuit simply inverts the received data when the inversion bit is high. The power model is described that contains different components of power dissipation of a link. The dynamic power dissipated by the interconnects and drivers is

$$P = [T_{0\rightarrow1}(T_S+C_l) + T_C C_C]V_{dd}^2 F_{ck} \quad (2)$$

where $T_{0\text{-}1}$ is the number of 0 ➔ 1transitions in the bus in two consecutive transmissions, T_C is the number of correlated switching between physically adjacent lines, C_S is the line to substrate capacitance, C_l is the load capacitance, C_C is the coupling capacitance, V_{dd} is the supply voltage and F_{ck} is the clock frequency.

The effective switched capacitance varies from type to type and hence, the coupling transition activity, T_C is weighted sum of different types of coupling transition contribution. Therefore

$$T_C = K_1T_1 + K_2T_2 + K_3T_3 + K_4T_4 \quad (3)$$

where T_i is the average number of type I transition and K_i is its corresponding weight According to [1] $K_1 = 1$, $K_2 = 2$ and $K_3 = K_4$. The occurrence probability of types I and II for a random set of data is 1/2 and 1/8, respectively. This leads to a higher value for K_1T_1 compared with K_2T_2 suggesting that minimizing the number of type I transition may lead to a considerable power reduction. Using (3),(2) may be expressed as

$$P = [T_{0\rightarrow1}(C_S+C_l) + (T_S+2T_2)C_C]V_{dd}^2 F_{ck} \quad (4)$$

According to [2], C_l can be neglected

$$P \propto T_{0\text{-}1}C_S + (T_1 + 2T_2)C_c \quad (5)$$

Table 1: Effect of Odd Inversion

	Normal			Odd Inverted		
Time	Type I			Type 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
	Type II			Type I		
t-1	01,10			01,10		
t	10,01			11,00		
	Type III			Type I		
t-1	00,11			00,11		
t	11,00			10,01		
	Type IV			Type I		
t-1	00,11,01,10			00,11,01,10		
t	00,11,01,10			01,10,00,11		

Occurrence probability for different types of transitions is calculated as follows; consider the data at time (t-1) and data at time (t) refer to the previous data and present data respectively. Considering only two adjacent bits of the physical channel. Sixteen different combinations of these four bits could occur (Table I). Note that the first bit is the value of the generic ith line of the link, whereas the second bit represents the value of its (i+1)th line. The number of transitions for types I, II, III and IV are 8,2,2 and 4, respectively. For a random set of data, each of these sixteen transitions has the same probability for Type I,II,III and IV are 1/2, 1/8, 1/8 and 1/4, respectively.

Data's in table is organized as follows. The first bit is the value of generic ith line of the link, whereas the second bit represents the value of its (I+1) th line. For each partition, the first(second)line represents the value at time t-1(t)

ODD Line Encoder: Odd line encoder focus on reducing the number of Type I transitions(by converting them to types III and IV transitions)and type II transitions (by converting them to type I transitions).This method compares the current data with previous data to decide whether odd inversion or no inversion of the current data can lead to the link power reduction. Table I reports, for each transition, the relation-ship between the coupling transition activities of the data when transmitted as is and when its bits are odd inverted. As Table I shows, if the data is odd inverted, Types II, III and IV transitions convert to type I transitions. In case of type I transitions, the inversion leads to one of types II, III or IV transitions. In particular, the transition indicated as T_1^*, T_1^{**} and T_1^{***} respectively.

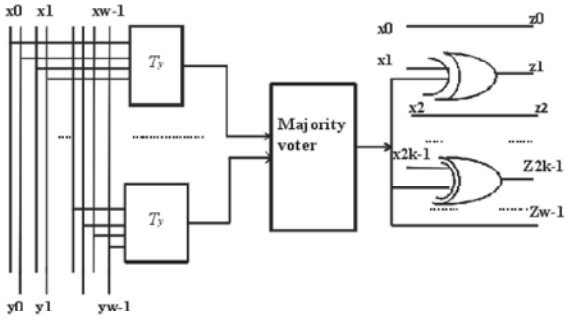

Fig. 2: Odd line encoder architecture

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

$$T_y = (w - 1)/2 \quad (6)$$

In the Fig. 2. the w-1 bits of the incoming(previous encoded) data are indicated by $X_i(Y_i)$, I=0,1,…w-2. The wth bit of the previously encoded data is indicated by inv which shows if it was inverted (inv=1)or left as it was(inv=0). In the encoding logic, each T_y block takes the two adjacent bits of the input data (e.g., $x_1\, x_2\, y_1\, y_2$, $x_2\, x_3\, y_2\, y_3$, $x_3\, x_4\, y_3\, y_4$) and sets its output to "1" if any of the transition types of T_y is detected. This means that the odd inverting for this pair of bits leads to the reduction of the link power dissipation (Table I). The T_y block may be implemented using a simple circuit. The second stage of the encoder, which is a majority voter block, determines if the condition (6) is satisfied (a higher number of 1s in the input of the block compared to 0s). If this condition is satisfied, in the last stage, the inversion is performed on odd bits. Thus the working principle of simple odd line encoder [3].

ODD/Full Line Encoder: This method makes use of both odd and full inversion. The full inversion operation converts Type II transitions to Type IV transitions. This encoding logic 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 link power reduction. The full inversion condition is obtained as,

$$T_2 > T_4^{**} \quad (7)$$

where T_4^{**} is the transition from type II to IV. The operating principle of odd/full line encoder are similar to those of odd line encoder. The proposed encoding architecture is based on odd invert condition (6) and the full invert condition (7) is shown in Fig. 3. Here again, the

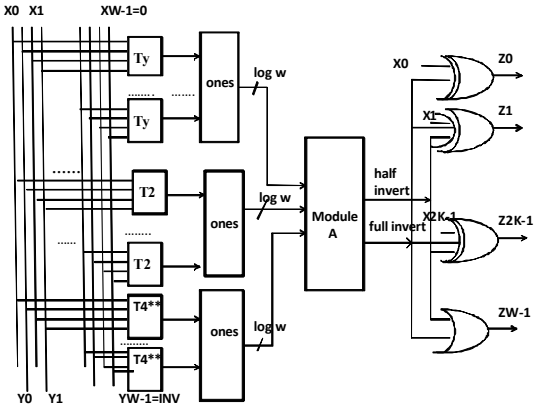

Fig. 3: Odd/full line encoder

Table 2: Effect of Even Inversion

	Normal			Even Inverted		
Time	Type I			Type 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
	Type II			Type I		
t-1	01,10			01,10		
t	10,01			00,11		
	Type III			Type I		
t-1	00,11			00,11		
t	11,00			01,10		
	Type IV			Type I		
t-1	00,11,01,10			00,11,01,10		
t	00,11,01,10			10,01,11,00		

wth bit of the previously and the full invert condition of(7) is show in Fig. 3. 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 T_y block in odd line encoder, the T_2 and T_4^{**} blocks which determine if the inversion based on the transition types T_2 and T_4^{**} should be taken place for the link power reduction. The second stage is formed by a set of 1s blocks which count the number of 1s in their inputs. The output of these blocks has the width of log_2w. The output of the top 1s block determines the number of transitions that odd inverting of pair bits leads to the link power reduction. The middle 1s block specifies the number of transitions that full inverting of pair bits leads to the link power reduction [4, 5]. Finally the bottom 1s block specifies the number of transitions whose full inverting of pair bits leads to the increased link power. Based on the number of 1s for each transition type, Module A decides if an odd invert or full invert action should be performed for the power reduction. For this module if (6) or (7) is satisfied, the corresponding output signal will become "1". In case no invert action should be taken place, none of the output is set to "1", Module A can be implemented using full-adder and comparator blocks.

ODD/Full/Even Line Encoder: In this method even inversion is added. The reason is that odd inversion converts some of type I (T_1^{***}) transitions to type II transitions. As can be observed from Table II, if the data even inverted, the transitions indicated as T_1^{**} / T_1^{***} in the table are converted to type IV/type III transitions. Therefore, the even inversion may reduce the link power dissipation as well. This method compares the current data with the previous data to decide whether odd, even, full, or no inversion of the current data can give rise to the link power reduction.

The even invert condition is given as

$$T_e > (w-1)/2,\ T_e > T_y,\ 2(T_2 - T_1^{**}) < 2T_e - w + 1 \quad (8)$$

The full inversion condition is obtained as

$$2(T_2 - T_1^{**}) > 2T_e - w + 1 \quad (9)$$

Then the condition for odd invert condition is given by

$$2(T_e - T_1^{**}) < 2T_y - w + 1,\ T_y > (w-1)/2,\ T_e < T_y \quad (10)$$

When none of (8),(9),(10) is satisfied, no inversion will be performed. The operating principle of this encoder is similar to those of the encoders implementing odd line and odd/full line encoder. The proposed encoding architecture, which is based on the even invert condition of (8), the full invert condition (9) and the odd invert condition (10) is show in Fig. 4. The wth bit of the previously encoded data 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 type 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 T_e block is added which determines if any of the transition types of T_2, T_1^{**}, T_1^{***} is detected for each pair bits of their inputs. For these transition types, the even invert action yields link power reduction. Again. four ones block is added to determine the number of detected transition for each T_y, T_e, T_2, T_4^{**},

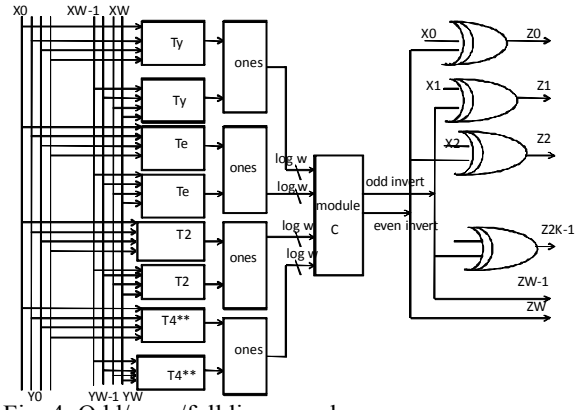

Fig. 4: Odd/even/full line encoder

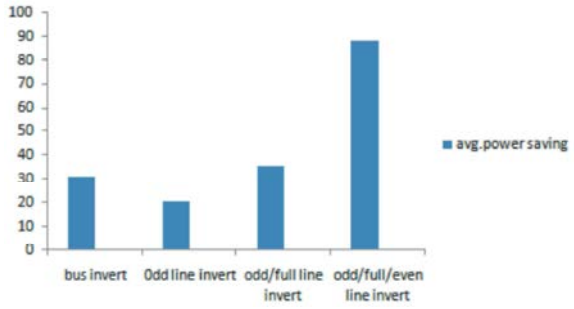

Fig. 5: Power saving chart

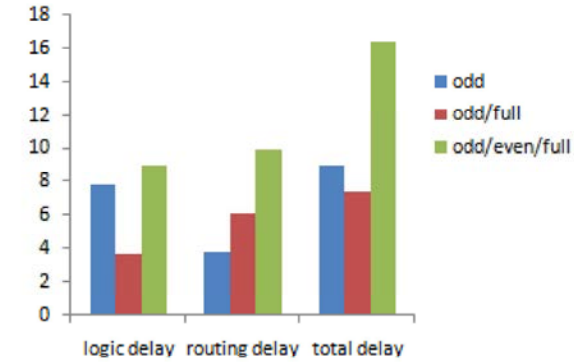

Fig. 6: Delay chart

blocks. The output of the ones block is input for Module C. This module determines if odd, even, full or no invert action corresponding to the outputs "10," "01,""11," or "00,"respectively,should be performed.

The outputs "01," "11,"and "10" show that whether (8), (9), (10), respectively, are satisfied. The Module C was designed based on the condition given in (8), (9), (10).

RESULTS AND DISCUSSION

From the performance chart in Fig. 5. it can be observed that the average power saving of odd line encoder is 20.6%,the average power saving of odd/full line encoder is 35.8% and the average power saving of odd/even/full line encoder is 88.12%.All these encoding approach add overhead in terms of power consuming by any integrated circuit incorporating the proposed encoder and also incurs area overhead on the chip shown in Fig.6.

CONCLUSION

Since the technology moving towards DSM to VDSM, technology the bus encoding has to overcome present scenario. The proposed data encoding technique called odd/even/full line encoder significantly reduces power dissipated by the links of a chip which accounts significant fraction of the total power budget. The performance of the encoder is evaluated in terms of switching activity and power saving.

REFERENCES

1. Ki, K.W., B. Kwang Hyun, N. Shanbhag, C.L. Liu and K.M. Sung, 2000. Coupling-driven signal encoding scheme for low-power interfacedesign, in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, pp: 318-321.
2. Wolf, W., A.A. Jerraya and Martin, 2008. Multiprocessor system-on-chip MPSOC technology, comput-Aided Design IntegrCcircuits Syst., 27(10): 1701-1713.
3. Yeh, D., L.S. Peh, S. Borkar, J. Darringer, A. Agarwal and W.M. Hwu, 2008. Thousand-core chips roundtable, IEEE Design Test comput., 25(3): 272-278.
4. Vittal, A. and M. Marek-Sadowska, 1997. Crosstalkreduction for VLSI, IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., 16(3): 290-298.
5. Ghoneima, M., Y.I. Ismail, M.M. Khellah, J.W. Tschanz and V. De, 2006. Formal derivation of optimal active shielding for low-power on-chip buses, IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., 25(5): 821-836.