International Journal of Scientific Research in Engineering and Management (IJSREM)

Volume: 05 Issue: 06 | June - 2021 ISSN: 2582-3930

ADAPTIVE ENCODING FRAMEWORK

Kunda Praveen, Mohd.Azhar pasha, Meghanshi reddy, N.Naga sai anish varma

Assit. Prof(ECE)

Student(ECE)

Student(ECE)

Student(ECE)

Abstract-*This paper presents the encoding scheme that reduces the switching activities and works without any prior knowledge about the signal switching statistics. Power consumption is a serious issue especially at high- speed communication. In this technique the data is observed and then formed into a cluster with the most correlated bit line. This method considers spatiotemporal redundancy to prevent information loss. This method will have extensive logic computations that results in little overhead.*

Keywords- *switching transitions, redundancy, encoder, bit-wise computing*

1. INTRODUCTION

The rapid progress in chip scaling technologies does not just decrease the chip size and so transistor area but also motivates for high-speed communication. In high-speed communication the leakage power will increase and the efficiency tends to decrease. Power optimization will also result in high reliability and low packaging cost.

Most of the power is wasted because of the switching at the external pins, which is mainly due to large off-chip capacitance compared to that of internal nodes. This can be minimized by reducing bus capacitance, total number of transitions in bit lines or by reducing supply power. Through bus encoding schemes we can enhance power savings and also can reduce transitional count. Usually some encoding schemes use spatial redundancy as an extra bit line or temporal redundancy in the form of extra bit transmission for transition reduction encoding scheme.

In general, for bus encoding we need data characteristics in advance, so that it becomes easy to encode but, in this frame work we don't need prior information of data statistics hence the name adaptive. As data bits are random compared to address bits so this technique is highly advised for data buses. The reduction in transition count will get degraded if the switching characteristics vary a lot temporally and spatially. When we don't know data statistics before and transitional probabilities of each bit line are changing over time with probabilities among the bit lines varying from low to high, then the consideration of the fixed subgroup or cluster of bit lines reduces the savings margin, since the transition correlation changes with time. The best way to enrich the transition reduction is to extract the signal statistics before application of encoding by observing the data over time. This ensures the establishment of the transitional correlation among bit line adaptively and dynamic formation of cluster with high correlated bit lines within a fixed observation window. This gives it an advantage over existing encoding schemes which cannot efficiently handle the situations when the transmitted data characteristics change abruptly.

In this paper, we will focus on reducing the transitional activity on data bits whose characteristics are not known prior and switching characteristics change spatially and temporally. In this technique we extract switching statistics and from that we form subgroups that has high correlation in that window.

In this paper, we focus on reducing transitions of data stream without knowing about the data characteristics. We propose adaptive technique that does subgrouping of highly correlated data patterns. We divided the paper into five sections, where section I is the basic introduction to the method. In section II, we discuss about previous contribution in this field of work and the adaption to that method. In section III, we discuss about the proposed method, its algorithm. In section IV, we discuss about the design of encoder and its switching savings. In section V, we discuss about the comparison between the various bus encodings and adaptive bus encoding. Based on the results we conclude it in section VI. And finally, in section VII, we listed the references from which we collected the information.

2. PREVIOUS WORK AND CONTRIBUTION

In this section we discuss about previous bus encoding schemes for power optimization and switch savings. We can classify bus encoding schemes as data and address bus encoding. Mostly nonadaptive encoding techniques are used for address buses because address bits are more sequential than data bits. The sequential property of the address bits is utilized by gray and T0 code to reduce the transitions. In T0 code, an extra bit line is used to indicate whether the bus is sequential or not. In case of consecutive accesses, the same data are transferred with no switching activity, otherwise actual address is sent. The T0 coding is further modified as Dual T0, T0B1, Dual T0-B1 which further reduces the transitions. However, as the sequential characteristics of data stream reduces, the usage of this technique started decreasing. Later, Ramprasad et al. [1] proposed INC-XOR encoding technique for instruction address stream, which outperformed existing techniques at that time. The proposed technique reduces transition activity on buses

where data sources are characterized in a probabilistic manner with the use of decorrelation and entropy coding function. He also introduced lower bound on minimum achievable average transition activity using information theory.

Benini et al. [2] proposed another address bus encoding, the beach solution, which is best suited for special purpose systems where it exploit temporal correlations between patterns.

Musoll et al. [3] proposed working zone encoding technique. In this register banks are used to store information on working zones. This technique reduces the transition by sending only the highly sequential offset with respect to the reference addresses of each working zone. Whenever data switches to a new working zone then the index selects the current working zone's value from one of the register banks. It is limited to specific applications because of its high hardware and delay overheads.

Luca et al. [4] proposed Asymptotic zero-transition activity encoding for address buses, this technique relies on observation. In this condition the devices located at the receiving end of the bus will automatically calculate the address to be received at next clock cycle, consequently the new pattern transmission can be avoided, resulting in switching activity to decrease. This method works because of the consecutive nature of address bits.

Mircea et al. [5] proposed bus-invert method of coding I/O which lowers the bus activity and decreases the I/O peak power dissipation by 50% and the I/O average power dissipation by 25%. It is best suited for uncorrelated data patterns. It sends the data either in the original form or in inverted form depending on the Hamming distance between two consecutive patterns. For N-bit non-multiplexed bus, if hamming distance is greater than N/2 then the second pattern is inverted, otherwise data are sent in its original form. One redundant bit is used to notify destination whether original data or its complement is sent. The efficiency of BIC increases when applied separately into a subgroup of bit lines. However, due to the requirement of high control bit count, the technique becomes expensive.

Alamgir et.al [6] proposed Bus switch encoding scheme. In this, to minimize toggling in physical bus wires they dynamically reordered bus line positions. The data is shifted circularly for reducing the transitions and increasing power savings than BIC. Later shin et al. [7] proposed Partial BIC(PBIC), which is improved version of BIC. In this method the effectiveness of BIC is improved but, it requires earlier knowledge of signal statistics to select subgroups of the bus based on transition correlation and transition probability. Later, shin and choi et al. [8] proposed combination of BIC and transitional signaling to achieve high switching reduction. An improved version of PBIC is proposed by siegmund et al. [9] as adaptive partial BIC(APBIC), here the cluster of lines that are to be encoded can be chosen adaptively based on local switching probabilities of each observation window. But, it has the disadvantages of high hardware complexity and degradation of performance with increase in temporal variation of signal statistics.

Mamidipaka et al. [10] proposed an adaptive encoding technique for data and address bus, which selects the minimum hamming distance code to minimize overall transitions. This encoding mechanism uses self-organizing lists that reorganizes themself with the change of data statistics. However, the complex encoder and decoder hardware with long lists requirement make this technique less impressive.

Benini et al. [11] proposed another adaptive encoding method which operates bitwise and does encoding by observing the bit streams over a period of time. But, the hardware overhead will also increase with window size Lv et al. [12] proposed dictionary based adaptive encoding scheme which target high correlation among adjacent bit lines. It transmits the index part of recurrent pattern of the data segment along with uncompressed part. In case of dictionary miss entire data word is sent.

J.Yang et al. [13] proposed Frequent value encoding which do not need prior information of data statistics. It uses CAM with least recently used replacement policy to encode the data. Before transmitting the data, encoder will compare the current data with stored data. In case of full match or partial match with varying widths, encoder sends one hot code to minimize overall transition. However, memory as well as address and data buses for retrieving information from memory necessitates high power consumption and hardware requirement.

Pappalardo et al. [14] proposed a novel bus-switch mechanism, which tentatively encode, cluster, and reorder wide data buses, according to a fixed coding function and a reordering pattern. The high hardware complexity limits its application to high capacity off-chip buses.

3. PROPOSED ENCODING SCHEME

The bus encoding schemes works mainly as two types. Under type1, various known statistical properties of input data stream are exploited. These type of encoding schemes are mainly used in address buses. Whereas under type2, the data characteristics are not known before, so it is mostly used in data buses. Addresses are generally issued in a sequential order from memory, so the Hamming distance between consecutive content of an address bus is usually small but, in data buses the content will be random. Since in majority of application it is difficult to know the data characteristics in advance so adaptive encoding technique is highly advisable for data buses.

In this technique, the highly correlated bit lines form a subgroup, which changes across different observation windows as local switching probability changes. In each observation window, one bit line is set as a basis line, which has maximum correlated switching transitions with the other lines. The lines that have maximum correlation with the basis are clustered together. The procedure for every observation window can be written as follows.

1) Find total number of switching transitions(α) in all the bit lines
2) For i = 0 to (N − 1) choose ith line as basis
3) For j = 0 to (N − 1), j ≠ i then put jth line in cluster if jth line has more switching transitions than jth line XOR-ed with basis line
4) XOR all the clustered bit lines with the basis line
5) Find number of switching transitions (β)in this modified set
6) Savings (μ) = α-β.
7) If b_z = argmax(μ), 0 ≤ i ≤ (N − 1), then the zth bit-line is the basis line for that observation window.

Basis and cluster information is sent to receiver by an extra bit line and using extra time slot. Extra bit line does not increase any switching transitions, and insertion of control information between two observation windows has minimal effect on the number of switching transitions.

4. PROPOSED ENCODER DESIGN

In this section, we will discuss the design of encoder Fig1. shows basic block diagram of proposed encoding methodology for bus of width N. It consists of decision blocks, delay elements, multiplexer and set of XOR gates. Decision block consists of eliminator, basis selection unit and bitwise savings computation block. It generates the control information, corresponding to subgroup for each observation window and the multiplexer will insert the temporal redundancy bits. Each element of the row will evaluate the savings contribution by each line and decides the presence of bit line in cluster for each observation window.

Savings computation unit at the end of each row computes overall savings for each bit line if it was chosen as basis. Savings computation unit is implemented using balanced carry save adder tree. Basis selection is nothing but an index selection unit, which compares the overall savings contribution due to selection of each bit line as basis and finally determines the potential basis among them.

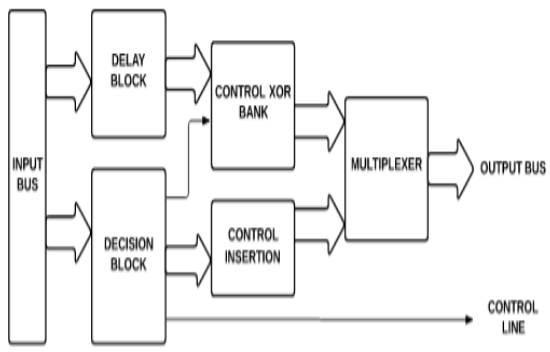

Fig.1. Block diagram of encoder

The presence of eliminator block reduces the internal node switching of encoder. It eliminates bit lines from subgroup at early stage without computing $\alpha_{i,j}$. It also groups potential bit lines as basis among all, which further decreases internal switching count. Fig. 3 shows the basic eliminator block when window size for transition observation is of 16 clock cycles.

The hardware implementation is optimized by removing the bit lines with switching probability less than 0.25 from basis consideration. This simplification is justified since these lines have a very low probability of being chosen as the basis; and eliminating them at this stage leads to power savings in the encoder. Reg i or Reg j stores the self-switching count. The number of these registers is equal to bus width. Hardware components of this section are shared among all other elements of the matrix. Positive value of $\alpha_{i,j}$ can be ensured if switching count of b j is greater than half of the switching count of basis. Eliminator block takes this scenario into account using the comparator to eliminate unwanted computations. The enable line of comparator output enables the computation of $\alpha_{i,j}$. Fig.2.diagram demonstrates one possible way to implement computation of bitwise switching savings for a particular observation window. One of the inputs of the final subtracter is the switching count of the basis for that window, 4-bit register stores the joint transitional count of basis bi and line bj . Adder and stored value in the register are shared with mirror element of (bi, b j). This reduces encoder dynamic power by minimizing internal node switching.

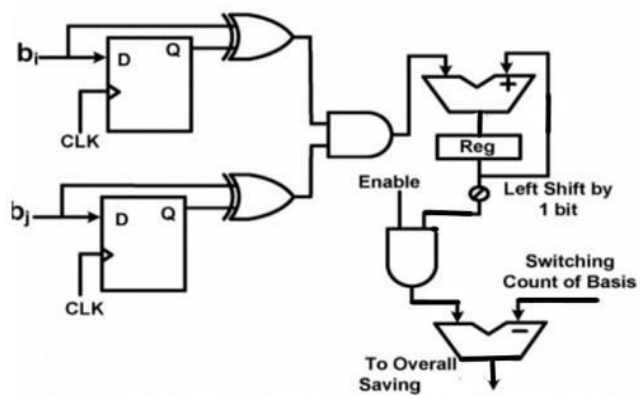

Fig.2. Savings computation block

The overall savings computation unit is an adder stage which takes input from the bitwise savings computation blocks and computes the total switching savings when a particular line is chosen as the basis. The basis selection unit takes the output of all the overall savings computation units and finally selects one of the bit lines as basis

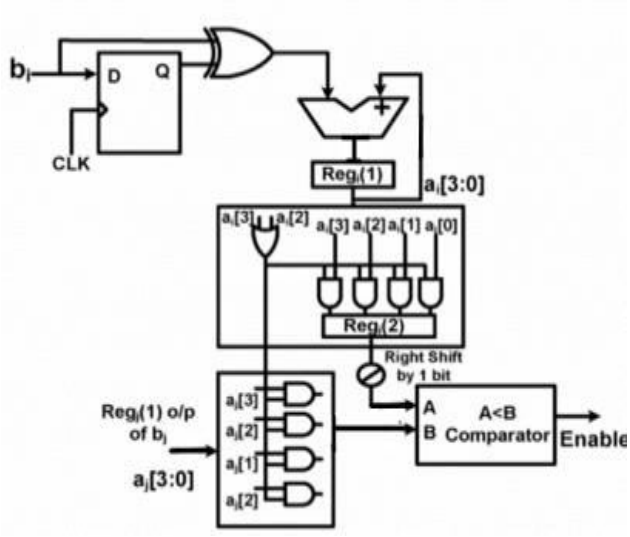

Fig.3. Eliminator block

5. RESULTS AND DISCUSSION

In this section, we present the various results of encoding. The comparison is performed with respect to different previous encoding techniques. Encoder power consumption is assessed by using XILINX ISE simulator. There is 3%–6% reduction in overall savings in simulation due to temporal reduction in between observation windows. The critical path delay is found to be 2.1ns.

Area of the entire architecture can be further optimized by increasing utilization target limit and by manually optimizing the RTL description. These results are for a 16-bit data bus, but the concept can be extended to larger bus widths. The ABE scheme can be applied to the entire bus width. With increase in the number of possible candidates for the basis, the area overhead and complexity of the encoder and decoder do increase. However, the cost of spatial redundancy reduces with larger bus widths.

In lower technology node, overall power consumption of encoder and decoder circuits is expected to decrease due to reduction of internal node capacitance value, thereby improving the value proposition of the proposed power optimization scheme that trades off internal node switching for reduction in off-chip transitions.

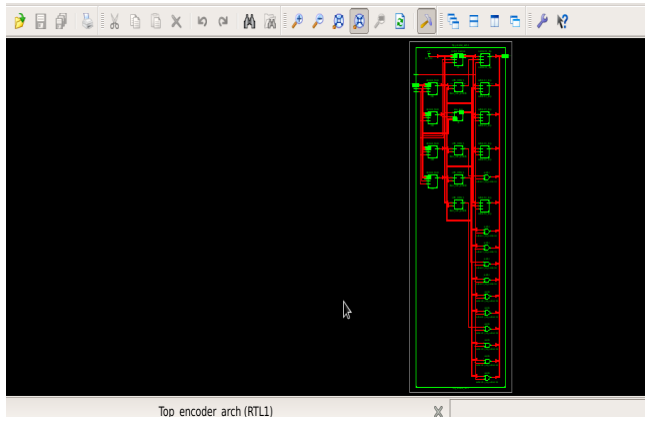

Fig.4. Encoder RTL schematic

Due to the presence of temporal redundancy, an extra time instant at the beginning of each window to specify the cluster information, the throughput increases by one clock period for every observation window. Hence, if the timing constraints are not hard, our approach can be used in any existing system where increased number of wait states are tolerated.

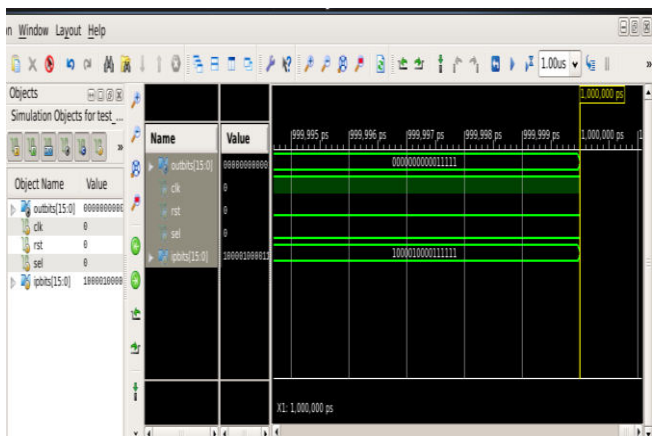

Fig.5.Simulation waveform of Encoder

The proposed algorithm shows the best adaptive encoding, giving much higher savings than BIC or BSC. In particular, if switching probability of the bit lines is approximately 0.5 and follows a uniform distribution then, BIC or APBIC may ultimately perform better than our algorithm, as the savings from the method may decrease by 2%–6% when temporal redundancy is considered. The voltage level and frequency of operation in the bus are considered to be the same as that used for synthesizing the encoder and decoder circuits

TABLE-1:Comparison with different methods:

Specification	Bus invert	APBI	Bus switch	ABE
Leakage	190	238	890	230

power(mW)				
Switching savings(%)	3.18	40.09	6.41	51.45
Delay(nS)	9.3	7.9	233.8	2.1
Area	7593	47544	208423	163900

6. CONCLUSION

In conclusion, a new bus encoding technique is proposed, it reduces the switching activities and works without any prior knowledge about the signal switching statistics. When we don't know data statistics before and transitional probabilities of each bit line are changing over time with probabilities among the bit lines varying from low to high, then the consideration of the fixed subgroup or cluster of bit lines reduces the savings margin, since the transition correlation changes with time. The best way to enrich the transition reduction is to extract the signal statistics before application of encoding by observing the data over time. This ensures the establishment of the transitional correlation among bit line adaptively and dynamic formation of cluster with high correlated bit lines within a fixed observation window. This gives it an advantage over existing encoding schemes which cannot efficiently handle the situations when the transmitted data characteristics change abruptly. The power and switching savings of the proposed algorithm and from existing bus encoding techniques are calculated in XILINX and compared. It is found that there is a slight reduction in the savings due to temporal redundancy, but overall, our algorithm consistently gives high savings. The advantage of this architecture is that parallel computation of all the bits can be done. Thus, this implementation of this encoder is found to be more efficient than the conventional encoder in terms of power consumption, delay and savings.

7. REFERENCES

[1] 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.
[2] 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. IEEE Great Lakes Symp. (GLS-VLSI), Urbana, IL, USA, Mar. 1997, pp. 77–82.
[3] E. Musoll, T. Lang, and J. Cortadella, "Working-zone encoding for reducing the energy in microprocessor address busses," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 6, no. 4, pp. 568–572, Dec. 1998.
[4] 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. IEEE Great Lakes Symp. (GLS-VLSI), Urbana, IL, USA, Mar. 1997, pp. 77–82.
[5] 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.
[6] M. Alamgir, I. I. Basith, T. Supon, and R. Rashidzadeh, "Improved bus-shift coding for low-power I/O," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), Lisbon, Portugal, May 2015, pp. 2940–2943.
[7] Y. Shin, S.-I. Chae, and K. Choi, "Partial bus-invert coding for power optimization of system level bus," in Proc. Int. Symp. Low Power Electron. Design, Aug. 1998, pp. 127–129.
[8] Y. Shin and K. Choi, "Narrow bus encoding for low power systems," in Proc. Conf. Asia South Pacific Design Autom. (ASP-DAC), 2000, pp. 217–220.
[9] R. Siegmund, C. Kretzschmar, and D. Müller, "Adaptive bus encoding technique for switching activity reduced data transfer over wide system busses," in Proc. Int. Workshop Power Timing Modeling Optim. Simulation, 2000, pp. 66–75.
[10] M. Mamidipaka, D. Hirschberg, and N. Dutt, "Low power address encoding using self-organizing lists," in Proc. Int. Symp. Low Power Design, Aug. 2001, pp. 188–193.
[11] L. Benini, A. Macii, M. Poncino, and R. Scarsi, "Architectures and synthesis algorithms for power-efficient bus interfaces," IEEE Trans. Comput.-Aided Design Integr., vol. 19, no. 9, pp. 969–980, Sep. 2000.
[12] T. Lv, J. Henkel, H. Lekatsas, and W. Wolf, "An adaptive dictionary encoding scheme for SOC data busses," in Proc. Design Autom. Test Europe (DATE) Conf., 2002, pp. 1059–1064.
[13] J. Yang and R. Gupta, "FV-encoding for low power data I/O," in Proc. ACM/IEEE Int. Symp. Low Power Electron. Design (ISLPED), Aug. 2001, pp. 84–87.
[14] M. Olivieri, F. Pappalardo, and G. Visalli, "Design issues for bus switch systems in deep sub-micro metric CMOS technologies," in Proc. IASTED Conf. Circuits, Signals, Syst. (CSS), Nov. 2005, pp. 112–117