

Low Latency and Area Optimization for Serial Links Using Embedded Transition Inversion Coding

V.Seetha Raman
Department of Electronics and
Communication Engineering,
Sasi Institute of Technology
and Engineering,
Tadepalligudem.
v.s.raman233@gmail.com

V. Krishnan
Assistant Professor
Department of Electronics and
Communication Engineering,
Sasi Institute of Technology
and Engineering,
Tadepalligudem.
krishnan@sasi.ac.in

P.Sivadurgarao
Assistant Professor
Department of Electronics and
Communication Engineering,
Sasi Institute of Technology
and Engineering,
Tadepalligudem.
sivadurgarao@sasi.ac.in

ABSTRACT-*This paper proposes embedded transition inversion (ETI) coding to reduce bit transitions in serializing parallel buses, implies reduction in area, delay and power. This paper proposes an embedded transition inversion (ETI) coding scheme that uses the phase difference between the clock and data in the transmitted serial data to tackle the problem of an extra indication bit. This technique is implemented in an optimized fashion using pipelining so that it can be used in practical systems with only a slight compromise in performance. B2I block is replaced with logic expansion of XOR gate operation. The complete proposed Embedded transition inversion architecture is coded in VHDL and the design is simulated using ModelSim and Xilinx ISE 12.1i, Spartan 3E family and device xc3s250e. In this proposed system area is reduced by 22% and delay is reduced by 34%.*

Keywords-Embedded transition inversion coding (ETI), (Transition Inversion Coding)TIC, Phase encoding, B2I(Bit2inversion), Serialized buffer, optimization.

I. INTRODUCTION

Low power design, in a system perspective, happens at all levels of the digital electronic system stack. It is being done from the lowermost device level design to the topmost software design. And there are the intermediate levels where a lot of effort is being expended to make systems run at low power, keeping the compromise in performance to be minimum. The increasing density of the integrated circuits as postulated by Moore's law makes it even more important to have low power systems since the power supply for such a dense integrated circuit may not keep track in size with the miniaturization of the electronic components. Serial link on-chip bus architecture is proposed to lower interconnect power [1].

Serialization reduces the number of wires and leads to a larger interconnect width and spacing. A large interconnect spacing reduces the coupling capacitance; while the wider interconnects reduce the resistivity[5]. A significant improvement in the interconnect energy dissipation is achieved by applying different coding schemes and their pro-posed multiplexing techniques. However, the power reduction decreases when the degree of multiplexing increases [1][2]. The transportation of data packets among various cores in a network on chip can be performed by the use of either a serial or a parallel link. Parallel links make use of a buffer-based architecture and can be operated at a relatively lower clock rate in order to reduce power dissipation[8][9]. However, these parallel links will incur high silicon cost due to inter-wire spacing, shielding and repeaters. This can be minimized up to a certain limit by employing multiple metal layers. On the other hand, serial links allow savings in wire area, reduction in signal interference and noise, and further eliminate the need for having buffers[10].

A. Bus Schemes

There are four types of bus schemes, including parallel (P), serial (S), encoding followed by serial (ES), and serial followed by encoding (SE) as shown in Fig1. Their average activity factor are AFP(activity factor parallel), AFS (activity factor serial), AFES(activity factor encoding followed by serial), and AFSE(activity factor serial encoder), respectively. The bit stream in this paper refers to the transmitted data in each wire of the input parallel bus[3]. The AF (activity factor) analysis results of serializing two bit streams into a single output[1]. The AF of each bit stream $AF_i (i = 1,$

2) can be viewed as the probability of bit switching within a cycle time. The bit switching in one bit stream within a cycle time could be 00,01,10 and 11.

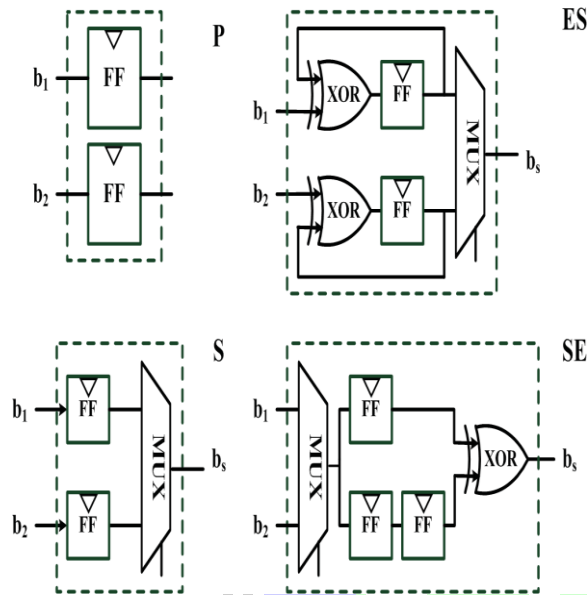


Fig.1. Four types of bus schemes: P, S, ES and SE.

II. EXSITING TECHNIQUE

Transition inversion coding (TIC) technique to reduce switching activity for random data and to detect errors. Their technique counts the transitions in the data word, and inverts the transition states if the number of transitions in a data word is more than half of the word length[4]. The scheme sets the current bit in the serial stream to be the same as the previous encoded bit when there is a transition. Otherwise, it is set to the inversion of the previous encoded bit.

A transition indication bit is added in every data word. This extra bit not only increases the number of transmitted bits, but also increases the transitions and latency. Serial link on-chip bus architecture is proposed to lower interconnect power. A significant improvement in the interconnect energy dissipation is achieved by applying different coding schemes and their proposed multiplexing techniques. However, the power reduction decreases when the degree of multiplexing increases. The embedded transition inversion (ETI) coding scheme to solve the issue of the extra indication bit [1][7].

This scheme eliminates the need of sending an extra bit by embedding the inversion information in the phase difference between the clock and the encoded data. When there is an inversion in the data word, a phase difference is generated between the clock and

data[6]. Otherwise, the data word remains unchanged and there is no phase difference between the clock and the data. This ETI coding scheme reduces transition by 31% compared with the SE scheme. The improvement of transition reduction is 19% compared with that of the transition inversion coding.

III. PROPOSED TECHNIQUE

A. ETI Encoder

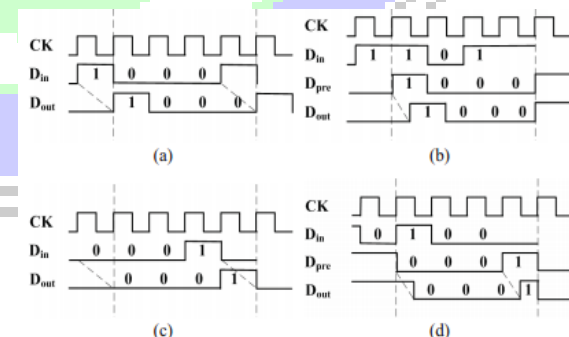
When word exceeds the threshold N_{th} , the bits in the data word should be encoded. Otherwise, the data word remains the same. When an encoding is needed in a data word, it checks every two bit in the data word. Every two bit in the serial stream is combined as a base to be encoded. In this case, the $b_{11}b_{21}$ is a base and the $b_{31}b_{41}$ is another base. The 2-bit in a base is denoted as b_1b_2 and the encoded output is denoted as $b_{e1}b_{e2}$. When the N_t in a data word is less than N_{th} , b_1b_2 remains unchanged. Otherwise, the inversion coding and the phase coding is performed. For the inversion coding, the bit streams "01" and "10" are mapped to "00" and "11," respectively. The bit streams "00" and "11" are mapped to "01" and "10," respectively. For the phase coding, we embed the inversion information in the phase difference between the clock and the encoded data.

The inversion encoding operation can be expressed as

$$b_{e1} = b_1$$

$$b_{e2} = \begin{cases} b_2, & \text{with } N_t < N_{th} \\ !b_2, & \text{with } N_t \geq N_{th}. \end{cases}$$

Since this operation is on a two-bit basis and only the second bit is inverted, it is called bit-two inversion (B2INV).



Graph.1. Dout "1000" obtained from (a) Din "1000" without encoding, (b) from Din "1101" with encoding. Dout "0001" obtained from (c) Din "0001" without encoding, and (d) Din "0100" with encoding.

B. Phase Coding

The Embedded transition inversion (ETI) coding uses the phase difference between the data and the clock to encode the indication information. The ETI_{pre} (Embedded transition inversion preceded data) has the same data word as the TIC, except that it removes the extra bit b_{ex} (bit extra). Removing the b_{ex} leaves eight sets of data words that are exactly the same. For example, there are two “1000” data words after the ETI_{pre} coding. Within every data word duration, the phase difference between the data and the clock distinguishes these two data words, as Graph.1 (a) illustrates Dout “1000” in If Din “1000” and “1101” without and with inversion. A half clock cycle difference between Dout and CK is shown in Graph.1 (b), indicating that Din has been encoded. The Dout and CK are aligned in Graph.1 (a), indicating that Din has not changed. Dout “0001” is the same in Graph.1(c) and (d) from Din “0001” and “0100” without and with inversion. This approach is able to identify whether Dout has been encoded or not as long as there is a half cycle delay between the Dout and CK. The internal architecture of XOR is shown in Fig.2.

C. B2I Using XOR

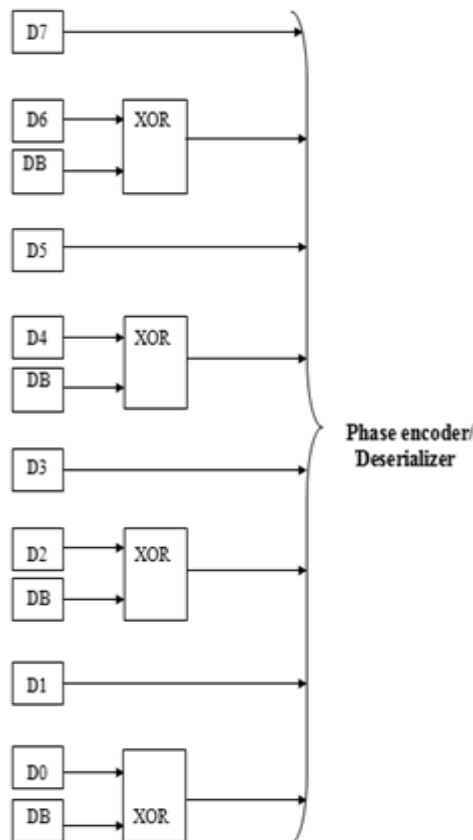
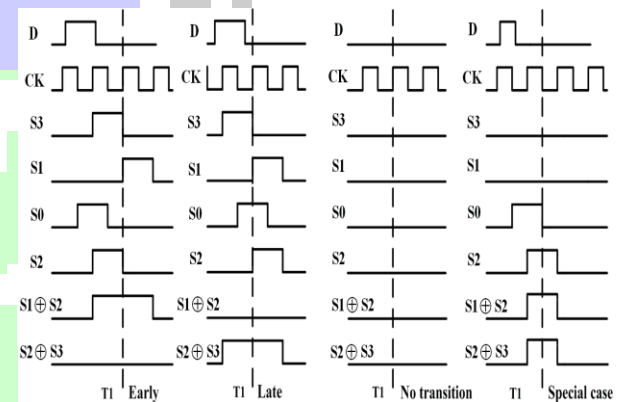


Fig.2. B2I using XOR

D. ETI Decoder

The Embedded transition inversion (ETI) encoder generates the phase difference between the clock and the data word. Normally, a Phase detector (PD) identifies an early or delayed phase. A variety of PDs detect the phase difference. This paper adopts the commonly used Alexander PD, which uses three consecutive clock edges to generate four sampling signals (S0, S1, S2, and S3). The PD is controlled by the clock CK and input data Din. When the clock CK and input data Din are valid, the PD is activated to identify.



Graph.2. clock and phase difference for determining the coding state.

The PD can determine whether a data transition exists from the condition that the clock leads or lags the data. The basic waveform is as shown in Graph.2. to judge the un-inverted, inverted, no transition, or the special data word. According to table.1 If the clock leads the data (early conditions), the signal $S1 \oplus S2$ is high and the $S2 \oplus S3$ is low. Conversely, if the clock lags the data (late conditions), the signal $S1 \oplus S2$ is low and $S2 \oplus S3$ is high. Thus, $S1 \oplus S2$ and $S2 \oplus S3$ provides the clock and data relations.

Table.1. Operation of XOR gate using Clock

$S1 \oplus S2$	$S2 \oplus S3$	Clock	Coding state
High	Low	Early	Has not been encoded
Low	Low	No transition	Has not been encoded
Low	High	Late	Has been encoded
High	High	Special case	Has been encoded

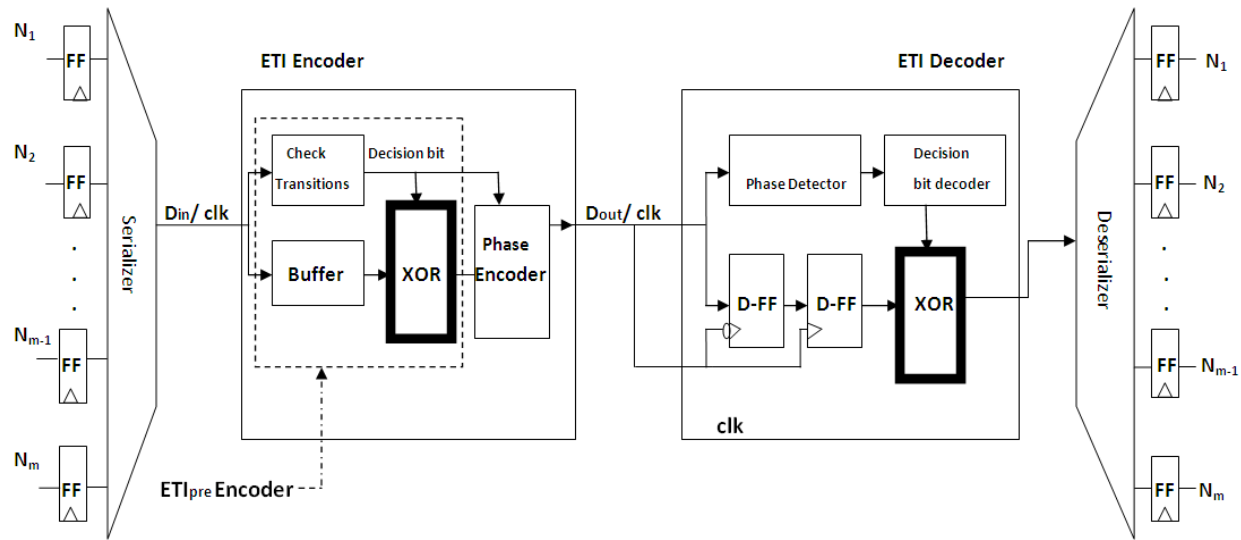


Fig. 3. Proposed Architecture of the ETI scheme.

IV. EXPERIMENTAL RESULTS

The complete proposed embedded transition inversion architecture is coded in VHDL and the design is simulated using ModelSim and Xilinx ISE 12.1i, Spartan 3E. The reduction of area, latency and power are also compared between existing and proposed systems.

second bit gets inverted. The given input is "01010101" and now every second bit gets inverted i.e "00000000". The inversion is done because of the divider which divides clock by two, and provides an indication signal for the first or the second bit in a pair of bitstream. The bit gets inverted when decision bit is high. If the decision bit is low input directly goes to the output. Here B2i output is obtained as "00000000" this is a special case, if all "0" or all "1" are obtained then the first bit gets inverted i.e shown as transmitter output. Final transmitter output is "10000000". Because this is a special case two bitstreams become "1". Receiver side, output is decoded and yielded as "11010101".

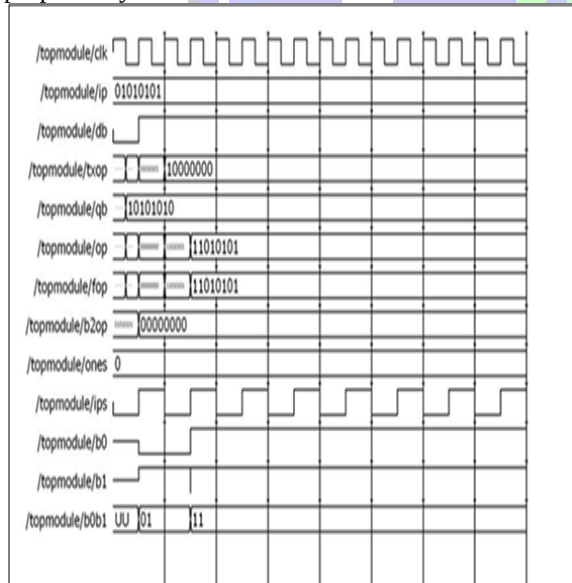


Fig.4. Existing[1] Results with B2I block

In Fig.4 The clock is set high and the input is given as "01010101". The number of transitions are seven. Since, the number of transitions are high data need to be processed through B2I block. In B2I the

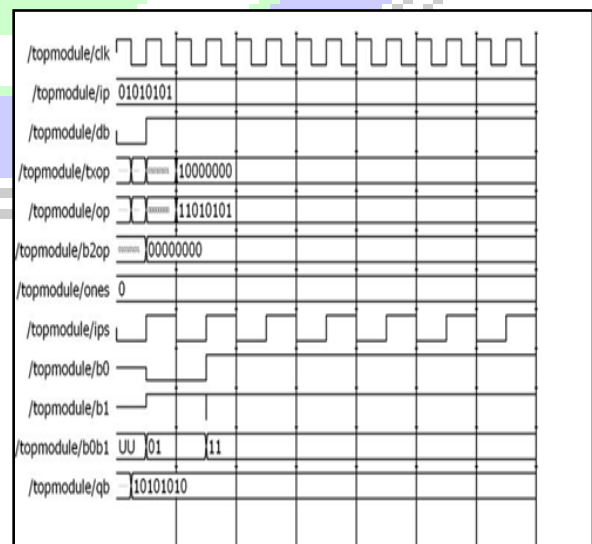


Fig.5. Proposed B2I replacing with XOR

In Fig.5 The clock is set high and the input is given as "01010101". The number of transitions are seven. Since, number of transitions are high data need to be processed through B2I block. Here B2I is replaced with XOR. In this XOR the 2nd bit is directly given to the XOR. In XOR the second bit gets inverted. The given input is "01010101" and now every second bit gets inverted i.e "00000000". The bit gets inverted when decision bit is high. If the decision bit is low input directly goes to the output. Here XOR output is obtained as "00000000" this is a special case, if all "0" or all "1" are obtained then the first bit gets inverted i.e shown as transmitter output. Final transmitter output is "10000000". Because this is a special case two bitstreams become "1". Receiver side, output is decoded and yielded as "11010101". The same results are obtained in both existing and proposed but there is reduction in area, delay and power. Above simulations are done without any B2I block. Finally, output is obtained with less power consumption and low area occupancy. Both area and delay comparison tables are shown below.

Table.2. Area comparison between existing [1] and proposed architectures.

S.NO	NAME	EXISTING	PROPOSED	% AREA OPTIMIZATION
1	Number of slices	66	58	12%
2	Number of flipflop's	6	5	16%
3	Number of Input LUT's	130	105	19%
4	Number of IO's	43	27	37%
5	Number of Bonded IOB's	43	27	37%
6	Number of IOB flip-flop	1	1	0%
7	Number of Gclk's	1	1	0%
	Total	290	224	22%

Table.3. Delay comparison between existing[1] and proposed architectures.

S.NO	NAME	EXISTING	PROPOSED	% DELAY REDUCTION
1	Minimum period	9.385ns	3.174ns	66%
2	Minimum input arrival time before clock	10.017ns	7.485ns	25%
3	Maximum output required time after clock	15.601ns	11.256ns	27%
4	Maximum combinational path delay	15.983ns	11.639ns	27%
	Total	50.986ns	33.554ns	34%

B2I block is used for encoding the data to reduce number of transitions for less power consumption. Here, initially zero is stored in D-flip flop. AND gate is connected with D-flip flop output and DB. Multiplexer first input is data, second input is NOT gate data, and multiplexer selection line is output of AND gate. AND gate output is determined by DB. If DB is '0' output of AND is '0', implies mux first input (original data) is passed to output. If DB is '1' output of AND is changes from 1 to 0 and vice versa, implies data and not (data) alternately passed to mux output. In existing technique[1] D-Flip flop, XOR gate, Multiplexer and NOT gate are needed. But, by using only XOR gates, we can design B2I block. Alternates bits of input data is fed to one of the input of XOR gate. XOR another input is DB. If DB is '0' directly input data is passed to output through XOR gate. If DB is '1' directly not of input data is passed to output through XOR gate.

V. CONCLUSION

The Embedded transition inversion (ETI) scheme reduces the extra bit used in the TIC scheme and reduces the crosstalk, energy dissipation. ETI scheme uses the phase difference between the data and clock to indicate the bit inversion. In this paper, the proposed ETI scheme reduces the area compare with the existing ETI scheme. In this proposed system area is reduced by 22% and delay is reduced by 34%. This enhanced technique is more useful in multipurpose applications due to the modification of XOR in place of B2I it reduces area, delay, and power consumption factors.

REFERENCES

- [1] Ching-te chiu, Wen-chih huang, chih-hsing lin, Wei-chih lai and Ying-fang tsao "Embedded transition inversion coding with low switching activity for serial links" IEEE transactions on very large scale integration (vlsi) systems, vol. 21, no. 10, october 2013.
- [2] M. Chennakesavulu1, A. Raghavi2 "Design and Analysis of Effective Coding Technique for Serial Links". International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Volume 3 Issue 4, April 2014.
- [3] D. N. Sarma and G. Lakshminarayanan, "Encoding technique for reducing power dissipation in network on chip serial links," in Proc. Int. Conf. Comput. Intell. Commun. Syst. Conf., 2011, pp. 323–327.
- [4] Abineth R., Bhargava R., Suresh Purini, Govindarajulu Regeti, "Transition Inversion

- based Low Power data coding scheme for Buffered Data Transfer”, 23rd International Conference on VLSI Design, January 2010.
- [5] R. Narasimha and N. Shanbhag, “Design of energy-efficient high-speed links via forward error correction,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 57, no. 5, pp. 359–364, May 2010.
- [6] E. Azadi, H. Ghasemizadeh, A. Khoei, and K. Hadidi, “5 Gb/s phase tracking clock recovery with new line coding in 0.35 μ CMOS process,” in Proc. Int. Symp. Telecommun. 2010, pp. 458–463.
- [7] M. Ghoneima, Y. Ismail, M. Khellah, J. Tschanz, and V. De, “Serial-link bus: A low-power on-chip bus architecture,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 56, no. 9, pp. 2020–2032, Sep. 2009.
- [8] S. H. Lin and S. I. Liu, “Full-rate bang-bang phase/frequency detectors for unilateral continuous-rate CDRs,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 55, no. 12, pp. 1214–1218, Dec. 2008.
- [9] Srikanth Gondi and Behzad Razavi, Fellow, IEEE “Equalization and Clock and Data Recovery Techniques for 10-Gb/s CMOS Serial-Link Receivers” IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 42, NO. 9, SEPTEMBER 2007.
- [10] M. R. Stan, W. P. Burleson. “Bus-Invert Coding for Low Power I/O”, IEEE Transactions on Very Large Integration Systems, Vol. 3, No. 1, pp. 49-58, March 1995.