International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

A New Methodology for Reducing Switching Activity in Serial Links of SoC

Binny P. Yohannan[1], Ajeesh A.V.[2]

[1]P G Scholar, VLSI and Embedded Systems , Department of ECE, T K M Institute of Technology, Kollam, India

[2]Assistant Professor, Department of ECE, T K M Institute of Technology, Kollam, India

Abstract: **As technology advances, the number of on-chip module will increase. Long and multi-bit parallel buses between these modules may cause several problems such as skew, cross-talk, wiring difficulty, and large area. Serial communication has many advantages over multi-bit parallel communication. Multiplexing parallel buses into a serial link enables an improvement in terms of reducing interconnect area and cross-talk, but it may increase the overall switching activity factor. Therefore, an efficient coding method that reduces the switching activity is an important issue in serial interconnects design. Most existing coding techniques in serial links like Bus-Invert coding and Transition Inversion Coding (TIC) need an additional bit for indicating the encoded data. This extra bit not only increases the number of transmitted bits, but also increases the transitions and switching activity. The new coding technique called Embedded Transition Inversion (ETI) coding is used for low switching activity in serial links. It uses the phase difference between the clock and the encoded data to avoid the problem of an extra indication bit.**

Keywords: System-on-chip(SOC), Coding Techniques, Serial Links, Switching activity, Phase detector.

1. Introduction

System-On-Chip (SoC or SOC) is an integrated circuit (IC) that integrates all the components of an electronic system into a single chip. Silicon technology integrates millions of transistors to a single chip, which makes system-on-chip (SOC) design possible. As technology advances, system-on-chip (SOC) devices are targeting complex applications. To do complex applications in SOC, the number of on-chip module will increase and also increases the on-chip buses for connecting these modules. Energy efficient communication between various SOC components is vital. Power consumption and delay in interconnects are expected to increase in future technologies due to increasing interconnects. Parallel communication between SOC modules may increase the efficiency but it may cause increase in area and cross-talk.

Serializing parallel buses reduces these disadvantages. Serial communication occupies less area due to less communication lines. Cross-talk problem can be avoided by wide spacing of serial lines. But it may cause increase in switching activity. Therefore, an efficient coding method that reduces the switching activity becomes an important requirement in serial interconnects design. Different coding techniques exist that reduces the switching activity in the serial links. They have both merits and demerits. Existing coding techniques in serial links like Bus-Invert coding and Transition Inversion Coding (TIC) need an additional bit for indicating the encoded data. This extra bit not only increases the number of transmitted bits, but also increases the transitions. The new coding technique called Embedded Transition Inversion (ETI) coding is used for reducing switching activity in serial links of system-on-chip (SOC).

Serial communication has many advantages over multi-bit parallel communication in the aspects of area, cross-talk, wiring difficulty etc. Serial communication occupies less area due to less communication lines. Cross-talk problem can be avoided by wide spacing of serial lines. However, the serial wire tends to dissipate more energy than parallel bus due to bit multiplexing. In Serialized Low Energy Transmission (SILENT) [7] coding technique, modulo-2 addition of present and previous bit of data bits gives the encoded data. The original data word from the sender unit can be recovered by XORing the encoded word and previously decoded word. The main disadvantage of this coding technique is the high power dissipation for large data bits.

Bus-invert coding [5] can be applied to inter-chip data buses for reducing switching activity. The idea is to send either the original or the inverted form of the next data word, making that decision by calculating the Hamming distance between two consecutive words on the bus. One additional signal line is needed for the receiver to be able to correctly interpret the bus values.

Today's system-on-chip (SoC) devices are targeting complex applications where there is a need for significant amount of computing power and data transfer. This implies that the number of on-chip modules will increase, and so will the number of on-chip buses connecting these modules. A serial link bus [4] means that multiplexing the data of m bus lines onto one line. This will reduces the number of physical bus lines. The four bus schemes are parallel (P),serial (S), encoding followed by serial (ES) and serial followed by encoding (SE). Transition inversion coding (TIC) [1] technique reduces the switching activity of data bits in the serial links. A word length is defined as the number of bits in the data word and a transition is defined as the bit changing from 0 to 1 or 1 to 0. Then TIC 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. 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 to indicate the receiver that the received data

is encoded or not. At the receiver, if the received indication bit is 1 then it has to decode the original data else no decoding is needed. This extra bit not only increases the number of transmitted bits, but also increases the transitions and latency.

2. Methodology

Embedded Transition Inversion (ETI) coding technique is used to solve the problem of an extra bit, which is used to indicate the receiver about the incoming data. The removal of extra indication bit reduces the switching activity of serial links in system-on-chip (SOC). The Embedded Transition Inversion (ETI) coding technique eliminates the need of an extra bit by introducing phase difference between encoded data and clock. If the data word is encoded in the transmitter, a phase difference is generated between data and clock to indicate the receiver that the incoming data is encoded. There is no phase difference between data and the clock if the data word is not encoded in the transmitter. Thus the removal of extra bit reduces number of transmitted bits and thus reduces the unnecessary transitions.

A. ETI coding scheme

Many coding algorithms are used in serial links for reducing switching activities. Most of the existing coding techniques use an additional bit to indicate the receiver that the incoming data is encoded. Embedded Transition Inversion (ETI) coding technique removes all the extra indication bit. A n/m ETI serial link with 'n' input bit streams under the degree of multiplexing 'm' is shown in the fig:1.

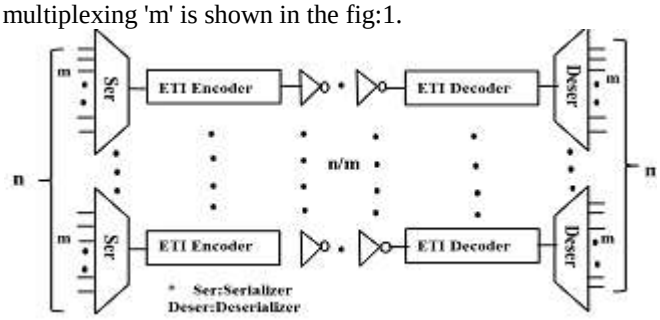

Figure 1: n/m ETI serial link with 'n' input bit streams under the degree of multiplexing 'm'

Each serial links has 'm' input bitstreams which are multiplexed by the serializer. After ETI encoding, the encoded data is transmitted through the serial links. This encoded data are received by ETI decoder. After ETI decoding, the decoded data are demultiplexed by the deserializer.

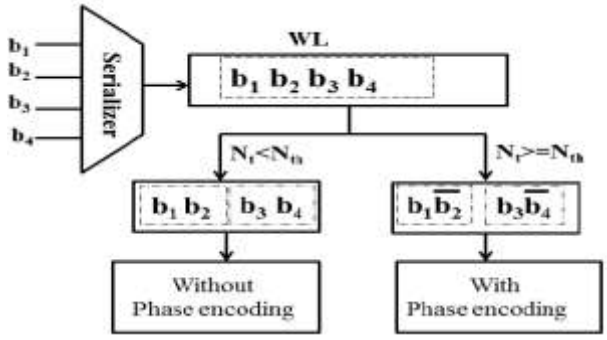

Figure 2: ETI coding scheme for one serial link

The ETI coding scheme includes both inversion coding and phase encoding. ETI encoding scheme for one serial link is shown in the fig:2. Inversion coding includes the encoding of data bits. If the data word is encoded then the phase coding phase generate a phase difference between data and clock.

1) Inversion Coding

The word length is defined as the total number of bits in the data word and it is denoted as WL. Transition is defined as the bit changing from 0 to 1 or 1 to 0. Threshold value is defined as the half of the word length and is denoted as N_{th}. For example, the word length, threshold value and number of transitions of the data word "0101" is four, two and three respectively. In Embedded Transition Inversion (ETI) coding scheme shown in fig2: , first the 'm' input data are multiplexed by a serializer. When the number of transitions in the data word exceeds the threshold value, the data word is encoded, otherwise the data word remains the same. The encoding scheme includes both inversion coding and phase coding. For example, consider the data word '0101'. Here the number of transitions is three and the threshold value is two,ie, the number of transitions is greater than the threshold value. Hence the data word must be encoded. A B2INV block is used for encoding the data word. The encoding operation is on a two-bit basis, the first bit is kept same and the second bit is inverted. Hence it is called as bit-two inversion (B2INV). The 2-bit base is denoted as 'b_1b_2' and the corresponding encoded bits are denoted as '$b_{e1}b_{e2}$'. For inversion coding, the bit-streams '10' and '01' are encoded to '11' and '00' respectively. If the data word is encoded, phase coding is needed, so for the phase encoding, a phase difference is created between the encoded data and the clock instead of an extra bit.

2) Phase Coding

Phase coding uses the phase difference between the data and the clock to indicate the inversion information. If the data word is encoded, then the phase encoder in the transmitter creates a phase difference between the data and the clock to indicate the receiver that the incoming data is encoded.

B. Block Diagram

The overall architecture of the Embedded Transition Inversion (ETI) scheme is shown in the fig.3. This architecture consists of the serializer, ETI encoder, ETI decoder and the deserializer.

Serializer is used for multiplexing parallel data bits into serial bits. In the ETI encoder part, the input data bits are stored in the buffer until the check transition operation is completed. The check transition block compares the number of transition and threshold value. If the number of transitions is greater than or equal to the threshold value, then the check transition block outs the transition bit as '1'. This decision bit is given to both the B2INV block and the phase encoder block. If the decision bit is zero, no encoding is needed and B2INV passes the original bit stream, otherwise the encoded bitstream is transmitted. The decision bit is also given to the phase encoder block. If the decision bit is '1', phase encoder creates a phase difference between the encoded data and the clock, else the un-encoded data word is transmitted. In the ETI

decoder part, the phase decoder checks the phase difference between the incoming data and the clock. The decision bit is generated based on the phase information obtained from the phase encoder. This decision bit is used for decoding the original data from the B2INV block. A deserializer is used here to convert serial bits to parallel bits.

The check transition block of the ETI encoder is shown in fig:4. A DFF stores the previous bit. The output of the FF is XORed with the current bit for checking the transition. The XOR gate output becomes '1' when the input bits are different, which means that there is a transition. The number of transitions are counted by a counter and then compared with the threshold value. If the number of transitions is greater than the threshold value, then decision bit becomes one. The architecture of the B2INV block of ETI encoder is shown in fig:5

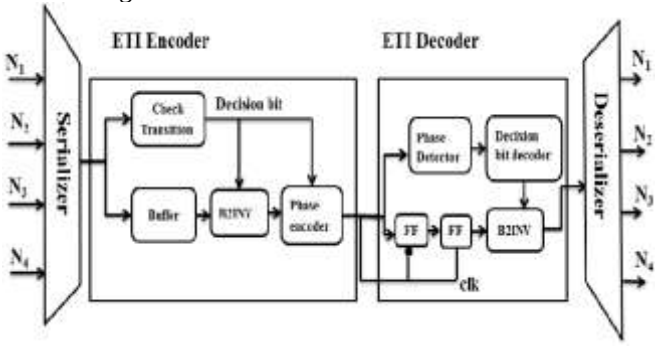

Figure 3: Overall architecture of the ETI scheme

B2INV block is used for encoding the data bits. It encodes the data bits only when the decision bit is high. At first the output of the DFF is set as '0'. Irrespective of the decision bit, the Din input will be selected by the MUX. In the next clock cycle, the DFF output will become '1', then the inverted input will be selected by the MUX if the decision bit is high.

Phase encoder shown in fig.6 is used for creating phase difference between encoded data and the clock signal. There are three types of phase encoding depending on the encoded data. They are one cycle delay, half cycle delay, and

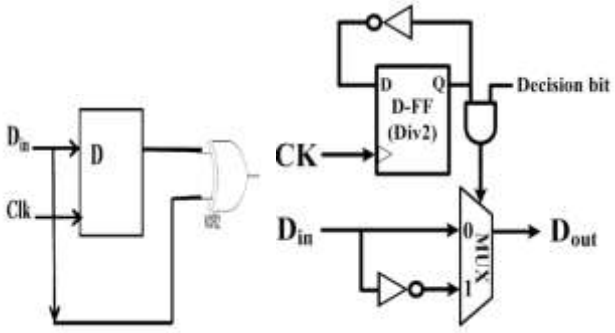

Figure 4: Check transition block Figure:5: B2INV block

encoding for the special data word. One cycle delay is shown in fig:7(a). Half cycle delay is shown in fig:7 (b) and special data word is shown in fig: 8

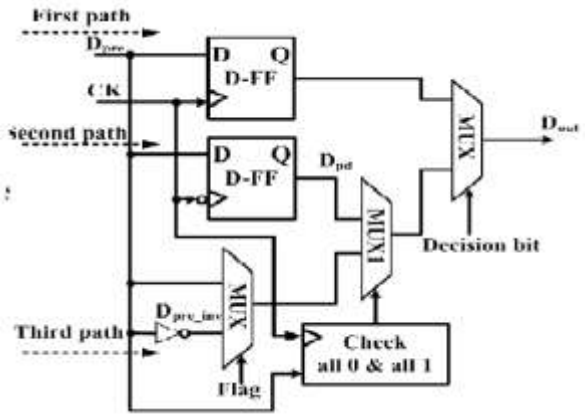

Figure 6: Phase encoder block

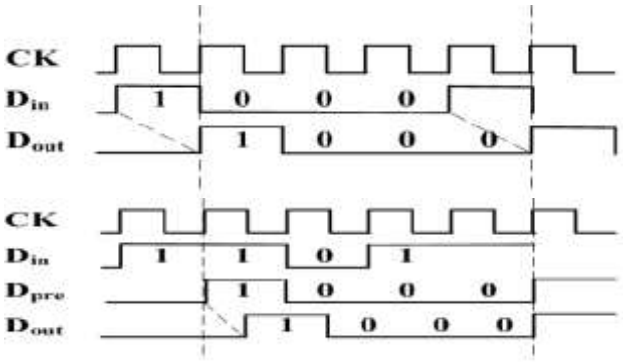

Figure 7: Dout ”1000” obtained from a)Din ”1000” without encoding, b) from Din ”1101” with encoding

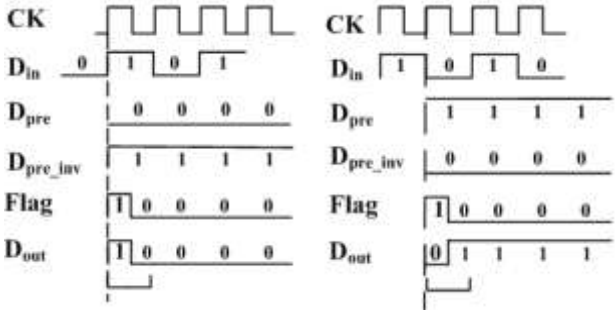

Figure 8: Waveform for ”0000” and ”1111

First path shows the phase encoding of the one cycle delay. The second and the third path show the half cycle delay and the special data word respectively. A predefined flag signal is used for special data word. The "check all 0 and all 1" block used for special data word when the encoded bits are "0000" or "1111". If the encoded data are not the special data word, the MUX1 selects the second path. Otherwise, MUX1 selects the third path. Depending on the decision bit, the data from the first path or output of the MUX1 is selected for the output.

ETI decoder consists of a phase detector (PD) to identify the phase relation between the received data and the clock. The Alexander phase detector is used here for identifying the phase difference. The architecture of Alexander PD is shown in fig:9

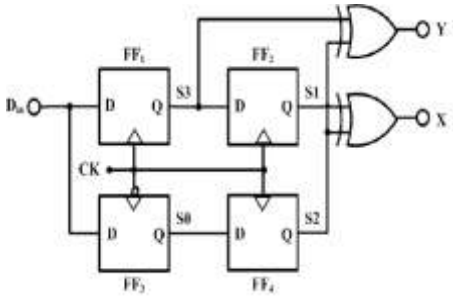

Figure 9: Alexander Phase Detector

Alexander PD uses three consecutive clock edges to generate four sampling signals such as S0, S1, S2 and S3. Based on phase information, the decision bit is decoded. Decision bit used for decoding the original data from the B2INV block, is same as that in the encoder part. For buffering, two DFF are used in front of B2INV block. When the decision bit is one, B2INV block extract the original data from the received data.

3. Results and Discussion

The ETI scheme consists of Serializer, ETI Encoder, ETI Decoder, Deserializer. Initially the coding for the individual block was completed and then all the modules were combined by using the structural modeling. The modules are modelled using VHDL in Xilinx ISE Design Suite 13.2 and the simulation of the design is performed using Xilinx ISim to verify the functionality of the design.. The RTL schematic view of the ETI scheme is shown in figure 10.

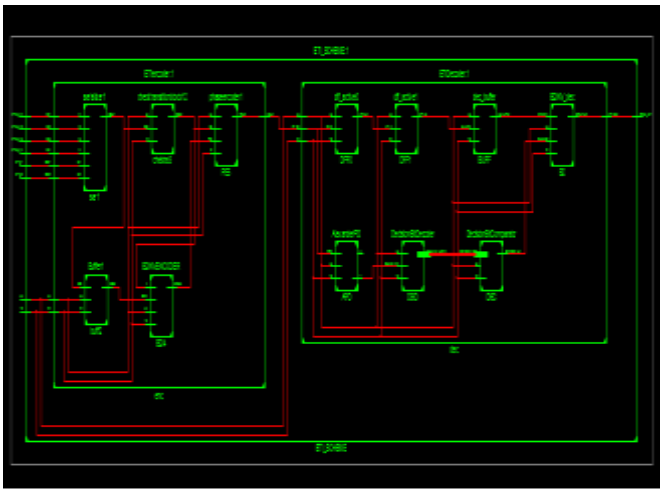

Figure 10: RTL-Schematic of ETI scheme

The simulation result of the ETI scheme is shown in figure:11. ETI Encoder encodes the data if the number of transitions in the data word is greater than half of the word length. If the dataword is encoded in the transmitter, a phase difference is created between the data and the clock. ETI decoder decodes the original data from the received data.

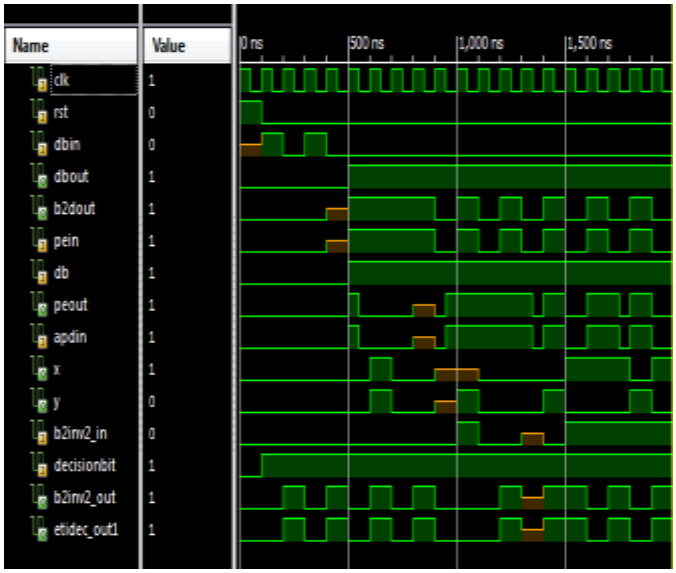

Figure 11: Simulation Result of ETI scheme.

4. Conclusion

As technology scales, serial communication between subsystems of SOC is one of the good way to reduce the area complexity. But switching activity is one of the important bottlenecks in the case of serial communication. Therefore, an efficient coding method that reduces the switching activity is an important issue in serial interconnects design. Different coding techniques are exists to reduce the switching activities in the serial links. Most existing coding techniques in serial links are Bus-Invert coding Transition Inversion Coding (TIC). This coding technique reduces the switching activities. Also these have some disadvantages. Bus-Invert Coding uses an additional channel to send an extra indication bit to indicate the receiver that the incoming data is encoded or not. Transition Inversion Coding (TIC) needs an additional bit for indicating the encoded data. This extra bit not only increases the number of transmitted bits, but also increases the transitions and latency. The new coding technique called Embedded Transition Inversion (ETI) coding is used for low switching activity in serial links. It uses the phase difference between the clock and the data in the transmitted serial data to avoid the problem of an extra indication bit.

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 Systems, VOL. 21, NO. 10, OCTOBER 2013.
[2] Rajan Narasimha and Naresh Shanbhag , ” Design of Energy-Efficient High-Speed Links via Forward Error Correction ”, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, VOL. 57, NO. 5, MAY 2010
[3] Sotirios Zogopoulos and Won Namgoong, ” High-Speed Single-Ended Parallel Link Based on Three-Level Differential Encoding ”, IEEE JOURNAL OF SOLID STATE CIRCUITS ,VOL. 44, NO. 2, FEBRUARY 2009.
[4] Maged Ghoneima, Yehea Ismail, Muhammad M. Khellah, James Tschanz and Vivek De, ” Serial-Link Bus: A Low-Power On-Chip Bus Architecture ”, IEEE

TRANSACTIONS ON CIRCUITS AND SYSTEMS ,VOL. 56, NO. 9, SEPTEMBER 2009.

[5] Fakhrul Z. Rokhani,Wen-Chih Kan, John Kieffer and Gerald E. Sobelman , " Optimality of Bus-Invert Coding ", IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMSII: EXPRESS BRIEFS,VOL. 55, NO. 11, NOVEMBER 2008.

[6] Srinivasa R. Sridhara and Naresh R. Shanbhag , " Coding for Reliable On-Chip Buses: A Class of Fundamental Bounds and Practical Codes ", IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 26, NO. 5, MAY 2007.

[7] Kangmin Lee, Se-Joong Lee, and Hoi-Jun Yoo, "SILENT: Serialized Low Energy Transmission Coding for On-Chip Interconnection Networks", in Proc. IEEE Int. Conf. Comput.- Aided Design Conf.,Nov. 2004.

[8] Behzad Razavi ," Challenges in the Design of High-Speed Clock and Data Recovery Circuits ", IEEE Communications Magazine,August 2002.