ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 5, March 2016

An Efficient Coding Technique to Reduce Dynamic Power Consumption for Serial Links

T.Preethi[1], P.Meenakshi Vidya[2], S.Sudha[3]

PG Scholar, Dept. of ECE, Easwari Engineering College, Chennai, Tamilnadu, India[1]

Assistant Professor, Dept. of ECE, Easwari Engineering College, Chennai, Tamilnadu, India[2]

Professor, Dept. of ECE, Easwari Engineering College, Chennai, Tamilnadu, India[3]

ABSTRACT: An efficient coding technique is used to reduce the overall switching activity and number of transitions for serial links. Multiplexing parallel buses into serial links has its advantages such as reducing interconnect area, coupling capacitance and crosstalk. But serialization increases bit transition which increases the activity switching factor and power dissipation. Many coding schemes exist to optimize the activity switching factor but results in increasing number of bit transitions. Hence 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 the extra indication bit. The analysis and simulation results indicate that the ETI coding scheme produces a low bit transition for different kinds of data patterns. ETI encoder block is constructed by check transition block, B2INV, WL indicator block, and phase encoder blocks. The ETI decoder block is constructed by phase detector block. All the internal modules are designed and simulated using Verilog HDL language. The simulated modules are synthesized using Cadence RTL compiler. Area, Power and Timing constraints values are calculated for individual blocks. The individual blocks are synthesized by using CMOS SMIC 250nm technology.

KEYWORDS: Serial links, Coding techniques, Bit transitions.

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 bottom most device level design to the topmost software design. And there are many 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.

Therefore the process of multiplexing parallel buses into serial links deals with the replacement of parallel buses which occupy larger area by serial links. The serialization process reduces inter connect area, coupling capacitance and crosstalk which was a cause of concern in parallel buses. But still there are certain parameters associated with the serialization process like the activity switching factor and the power dissipation that has to be addressed while multiplexing the parallel buses into serial links before it can be implemented in the system on chip design. The activity switching factor and the power dissipation increases during the process of serialization. The activity switching factor tends to increase with the increase in bit transitions. 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. A significant improvement in the interconnect energy dissipation is achieved by applying different coding schemes and existing multiplexing techniques. However, the power reduction decreases when the degree of multiplexing increases.

The Embedded Transition Inversion (ETI) coding scheme is used to solve the issue of the extra indication bit. The ETI 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. Otherwise, the data word remains unchanged and there is no phase difference between the clock and the data.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 5, March 2016

1. TYPES OF BUS SCHEMES

Switching factor analysis for different kinds of coding schemes are of four types including parallel (P), serial (S), encoding followed by serial (ES) and serial followed by encoding (SE) [2]. The bit stream refers to the transmitted data in each wire of the input parallel bus.

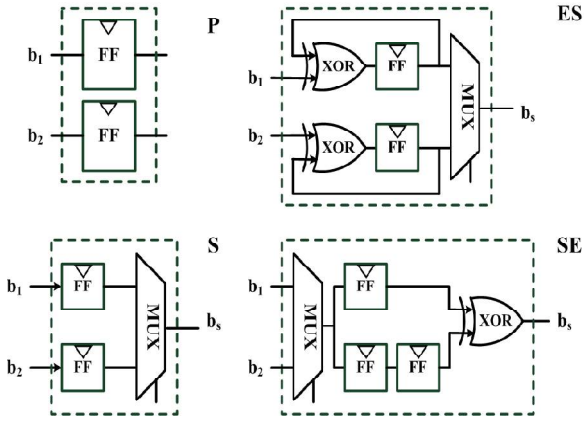

Fig.1: Four types of bus schemes: P,S,ES,SE

II. EMBEDDED TRANSITION INVERSION CODING (ETI)

Although many coding algorithms can reduce the switching AF, most of them are designed for specific applications, such as video streaming or strongly correlated data. The ETI coding scheme that operates on a two-bit basis and removes all the transition indication bits, (Fig.2) N/m ETI serial links with n input bit streams under degree of multiplexing m. (Fig.3) ETI coding scheme for one serial link, word length (WL), Nth is equal to WL/2, and number of transition is Nt. An n/m ETI serial links with n input bit streams under degree of multiplexing m is shown in Fig.2. Each serial link has m input bit streams that are multiplexed by a serializer, followed by the ETI encoding. The encoded stream is transmitted through the serial link and followed by the ETI decoding and a deserializer. The ETI coding scheme includes the inversion coding and phase coding as shown in Fig.2 and 3.

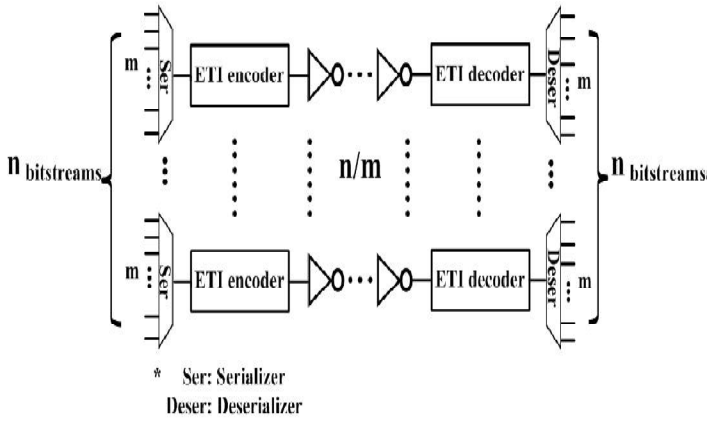

Fig.2: n/m ETI serial link with n input bit streams under degree of multiplexing m.

Inversion Coding: Define the word length (WL) as the number of bits in a data word and a threshold Nth as half of WL. A transition is defined as a bit changing from zero to one or from one to zero. For example, the bit stream “0100” has two transitions while “0101” has three transitions. When the number of transitions not in a data word exceeds the threshold Nth, 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, this method checks every two-bit in the data word. Every two bit in the serial stream

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 5, March 2016

is combined as a base to be encoded. In this case, the b11b21 is a base and the b31b41 is another base. The 2-bit in a base is denoted as b1b2 and the encoded output is denoted as be1be2. When the Not in a data word is less than Nth, b1b2 remains unchanged. Otherwise, we perform the inversion coding and the phase coding. 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 the inversion decoding operation for the decoded output bd1 bd2. Since this operation is on a two-bit basis and only the second bit is inverted, it is called bit-two inversion (B2INV).

Phase Coding: The ETI coding uses the phase difference between the data and the clock to encode the indication information. The ETI has the same data word as the TIC, except that it removes the extra bit bex. Removing the bex leaves eight sets of data words that are exactly the same. For example, there are two "1000" data words after the ETIpre coding. Within every data word duration, the phase difference between the data and the clock distinguishes these two data words.

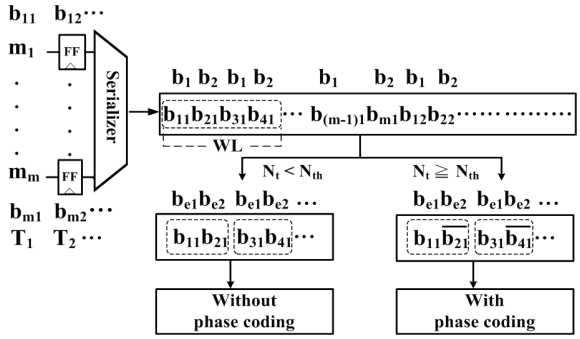

Fig.3: ETI coding scheme for one serial link

III. ETI ARCHITECTURE

A. ETI Encoder:

The overall architecture of the ETI scheme is shown in Fig. 3.3. It adds the ETIpre block and the TIC architecture for clarity. The ETIpre does not provide the decision bit information so it cannot be decoded in the receiver. The ETIpre encoder is shown by the dashed box in the ETI encoder in Fig.4. The TIC counts the transitions in the data word then uses this information to perform encoding. The transition indication bit is added to every data word to indicate whether there is an inversion or not. The decoder adopts the transition indication bit to perform the decoding, as shown in Fig.3.3. In the ETI encoder part, the input data Din are stored in the buffer to wait until the check transition operation is completed. The transition and threshold in a data word are used to set the decision bit. The decision bit is used to control the encoding process in the B2INV and the phase encoder block.

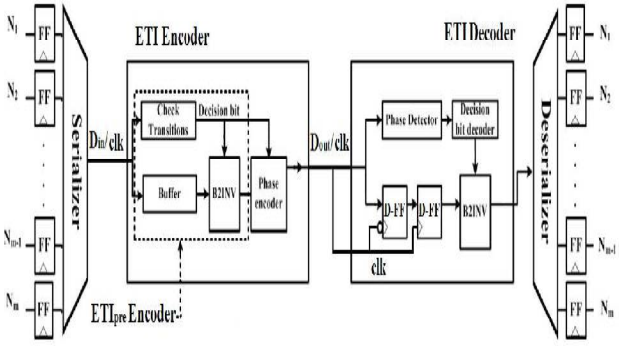

Fig.4: Overall architecture of ETI scheme

The ETI encoder includes the check transitions block, buffer, B2INV, and phase encoder. The check transition block is shown in Fig.5

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 5, March 2016

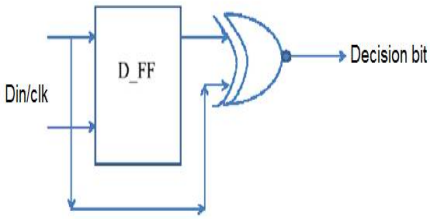

Fig.5: Check Transition Block

The check transition circuit is built by counting the transitions between consecutive bits in the bitstream. A transition between two bits is found in a simple manner by performing the equivalence operation of XOR (Exclusive OR) between them. The circuit using a simple XOR gate between consecutive incoming bits of the bit stream. The WL indicator block counts the length of the data word and generates a high signal at the first bit of the data word. This signal is used to reset the adder and the D-flip-flop (D-FF). The D-FF stores the previous bit that is used to XOR with the current bit for transition checking. Before transmission, the number of transitions on a line is counted. This is just counting the transitions of the bit stream in that line. This can be done by a simple XOR gate between consecutive bits and counting the number of 1's. The adder block calculates the number of transition in a data word and sets the decision bit to high when the $Nt \geq N$th. If the decision bit is set to 1 the input data becomes inverted. Word length (WL) defines the number of bits in a data word and a threshold Nth defines half of WL. A transition is defined as a bit changing from zero to one or from one to zero. For example, the bit stream "0100" has two transitions while "0101" has three transitions. When the number of transitions Ntin a data word exceeds the threshold Nth, 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, the method 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 $b11b21$ is a base and the $b31b41$ is another base. The 2-bit in a base is denoted as $b1b2$ and the encoded output is denoted as $be1be2$. When the Ntin a data word is less than Nth, $b1b2$ remains unchanged. Otherwise, we perform the inversion coding and the phase coding. 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, it embeds the inversion information in the phase difference between the clock and the encoded data. The bit stream is encoded if a transition inversion is needed. This is done as the data is being put on the bus. This can be done in an on-the-fly manner since the encoder need to only process the current and next bit. The decision bit is used to control the encoding process in the B2INV and the phase encoder block. When the decision bit is set to zero, the B2INV passes the non-inverted bit stream. Otherwise, the bit stream is encoded. This encoder needs to operate only for those cases where a transition inversion is needed. The D-FF on the incoming bit stream calculates the transition state just as the decision circuit did during the loading of the block. Once the transition state is known, it is inverted to generate an inverted state if the decision was to invert the transition. This inverted transition state is used to manipulate the next bit in such a way that the next bit will be in the inverted transition state in correspondence to the current bit. The inverter block is shown in the Fig.6.

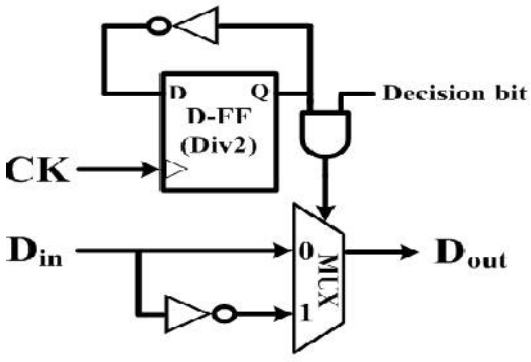

Fig.6: B2INV

The phase generator is used to generate phase difference between the encoded data (Dpre) and the clock (Clk) at each data word. Depending on the encoded data, there are three types of phase encoding: the one cycle delay, the half cycle

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 5, March 2016

delay and the special data word. The half cycle delay and the special data word are shown by the second and the third pathin fig.7

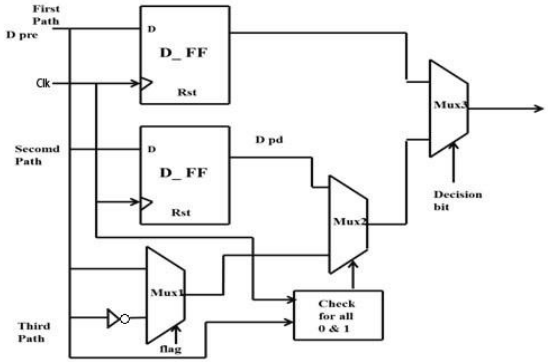

Fig.7: Phase Encoder Block

B. ETI Decoder

The ETI encoder generates the phase difference between the clock and the data word. Normally, a PD identifies an early or delayed phase. The decision bit is used in the B2INV for the decoding. The decoding operator in the B2INV is the same as that in the encoder. Two D-FFs are added in the front of the B2INV block for buffering and alignment. A larger bandwidth is needed in the ETI coding scheme due to phase shift. The last bit has half the pulse width of the other bits so that the interconnection has twice the bandwidth. It means that the serial link needs to run at a much frequency. The higher clock frequency leads to problems, such as buffering, clock synchronization, and design complexity. The other way is to wait an extra bit to check the transition information but that would lower the overall bit rate. Data and clock is sent from Transmitter to Receiver in this ETI scheme. In this simulation, it is assumed that the channel length of the clock and data links is equal and the phase skew between them is negligible.

IV. SIMULATION AND RESULTS

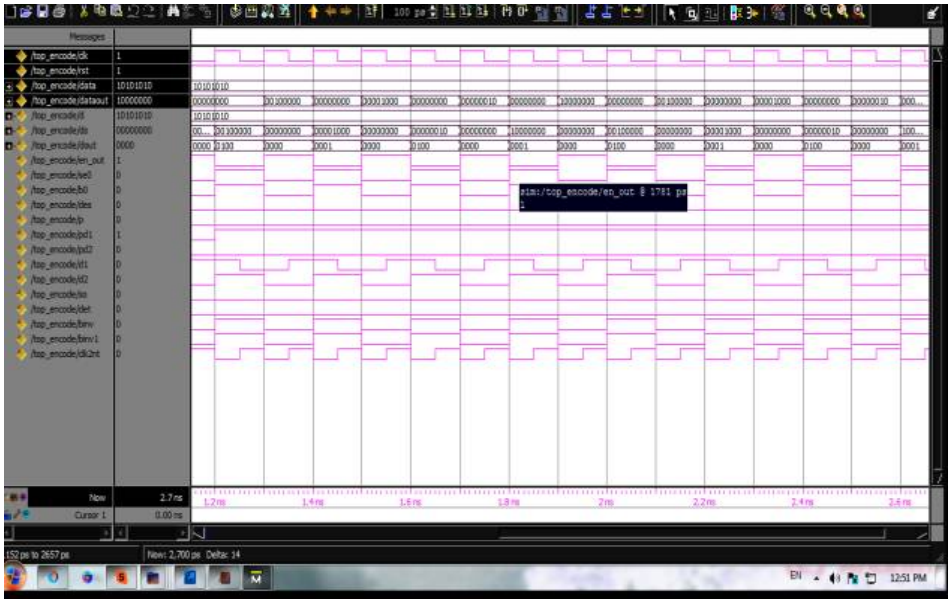

Fig. 8: Waveform of Overall ETI architecture

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Special Issue 5, March 2016

BLOCKS	POWER	AREA(m)
Check transition	**0.3nw**	**2685.00**
Phase Encoder	**0.23mw**	**3630.00**
ETI Encoder	**0.3899mW**	**6741.00**
B2INV	**33.82µW**	**584.00**

Table 1: Area and power Comparison

V. CONCLUSION

The ETI coding is a scheme that reduces the power dissipation of a serial link. The phase difference between the clock and the data to reduce the switching activity of the serial link uses ETI scheme. The analysis and simulation results indicated that the coding scheme achieves fewer transitions for most data patterns. Using the optimum degree of multiplexing, optimum width, and spacing, the ETI coding scheme achieves energy reduction compared with the parallel bus without overhead.

The ETI encoder architecture consists of check transition block, B2INV, phase encoder modules are designed by using Verilog HDL language. The internal modules and ETI encoder architecture are synthesized by using CMOS SMIC 250nm technology. The estimated power consumptions of check transition block is 0.3nW, B2INV power consumption is 33.82µW, phase encoder power consumption is 0.23mw and ETI encoder architecture total power consumption is 0.3899mW.

REFERENCES

[1].Abinesh.R, Bharghava.R, and Srinivas M.B, (2009), "Transition inversion based low power data coding scheme for synchronous serial communication", in Proc. IEEE Comput. Soc. Annu. Symp. VLSI Conf., pp. 103–108.

[2].Ghoneima.M, Ismail.Y, Khellah.M, Tschanz. J, and V. De, (2009), "Serial-link bus: A low-power on-chip bus architecture," IEEE Trans. Circuits Syst.I, Reg. Papers, vol. 56, no. 9, pp. 2020–2032.

[3]. Huang W.C, Lin C.H, and Chiu.C.T, (2011), "Embedded transition inversion coding for low power serial link", in Proc. IEEE Workshop Signal Process. Syst.Conf, pp. 102–105.

[4].Kuo C.H, Wu W.B, Wu Y.J, and Lin J.H,(2006), "Serial low power bus coding for VLSI", in Proc. IEEE Int. Conf. Commun., Circuits Syst.,pp. 2449–2453.

[5]Lee.K, Lee S.J, and Yoo H.J, (2004), "SILENT: Serialized low energy transmission coding for on-chip interconnection networks", in Proc. IEEE Int. Conf. Comput.-Aided Design Conf., pp.448–451.

[6] Lin. R.B and Tsai .C.M, (2002), "Weight-based bus-invert coding for low power applications", in Proc. Int. Conf. VLSI Design, pp. 121–125.

[7] .Sarma .D.N and Lakshminarayanan .G, (2011), "Encoding technique for reducing power dissipation in network on chip serial links", in Proc.Int. Conf. Comput.Intell.Commun. Syst. Conf., pp. 323–327.

[8]. Stan.M.Rand Burleson W.P.,(Mar.1995) "Bus-invert coding for low-power I/O", IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 3, no. 1, pp. 49–58.

[9]. Sotiriadis.P.P. andChandrakasan. A, (Nov.2000), "Bus energy minimization by transition pattern coding (TPC) in deep submicron technologies", in Proc. IEEE/ACM Int. Comput.-Aided Design Conf., pp. 320–327.

[10]. Shin. Y, Chae .S.I, and Choi.K, (Apr.2001), "Partial bus-invert coding for power optimization of application-specific systems", IEEE Trans.Very Large Scale Integr. (VLSI) Syst., vol. 9, no. 2, pp. 377–383.