Volume 2, Issue 12, December-2015, pp. 842-848 ISSN (O): 2349-7084

International Journal of Computer Engineering In Research Trends

Available online at: www.ijcert.org

A Novel Design for Encoding and Decoding to Minimize Crosstalk in VLSI Circuits

[1]J.Prema,[2] Akash Venkatachalam

[1]M.Tech,[2]B.Tech
SRM UNIVERSITY

Abstract: - In the developing world much know-how are growing faster and faster as they are becoming reduced, one such is the very large scale integrated design-VLSI. Many trials are faced in VLSI, among them is the crosstalk frequency. Global buses in deep-submicron (DSM) system-on-chip designs Ingest noteworthy extents of power, have large transmission interruptions and are prone to errors due to DSM noise-crosstalk. Owing to this, crosstalk manifestation on long on-chip buses is progressively becoming a limiting factor in high-speed designs. Crosstalk among nearby wires on the bus may create a remarkable portion of the transmission delay. Placing a guard wire between each signal wire assuages the crosstalk problem but doubles the area used by the bus, which is an objectionable concern. Instead, it is proposed to employ data encoding and decoding for special codes called boundary shift codes to abate crosstalk inside a bus.

Keywords– VLSI, Deep-Submicron (DSM) system, Crosstalk, Hamming code, Encoder and Decoder

1. INTRODUCTION:

VLSI stands for "Very Large Scale Integration". The incredible growth in VLSI has come from steady miniaturization of transistors and improvements in manufacturing processes. Though transistors became smaller, they are faster, dissipate less power and are cheaper to manufacture. The VLSI technology provides the user with a new and more complex range of 'off-the-shelf' circuits. Moreover, the design processes are such that system designers can readily design their own special circuits of considerable complexity. This provides a new degree of freedom for designers and it is probable that some very significant advances will result.

It contains a lot of adeptness on many fronts within the same field. Also, by conforming to Moore's law, i.e. the number of transistors per square inch on integrated circuits had doubled every year since the integrated circuit was invented. Examples are embedded systems, where smart and self-sufficient devices are put inside everyday objects and commonplace computing where small computing devices are made to expand and act in such a manner that even the footwear worn by a person may actually do something worthwhile like monitoring the functioning of the person's heartbeat. These two fields are related and getting into their description can easily lead to the world of VLSI [2].

2. CROSSTALK IN VLSI:

The exponential increase in signal changeover speed and compactness density of integrated circuits leads to rousing design and test hitches. The circuit linking wires that were once considered to be electrically secluded earlier can now impede with each other and have an effect on system output and efficiency. One such interaction caused by linked wires is known as crosstalk.

Crosstalk can be defined in many ways [2, 5]. Crosstalk can create logic errors in the circuit. The designs of encoder and decoder for boundary shift codes to reduce crosstalk and also to rectify the error have been discussed. This can be beneficial in a range of applications comprising nanotechnology, low-swing signaling and radiation-hardened circuits.The exhaustive description of crosstalk has been discussed [3].

2.1 Boundary Shift Codes: Boundary Shift Codes are used for error rectification and for minimizing crosstalk in VLSI. Data words are mapped to code words which are self-shielding and hence have no invalid change, which allows cross talk to be curtailed. The suggested design of encoder and decoder for this code to minimize crosstalk is also discussed.

2.2 VLSI Features: VLSI technology initiates and creates the innovative designs and systems which can change the way we live. The integration enhances the design [3]. It comprises features like lower parasitic which equals higher speed, lower power, physically smaller, more trustworthy than discrete system, can design more intricate system, high speed of circuits on chip due to lesser size etc. For example, the microprocessor is a VLSI device. The term has been unique as chips have improved in complexity into the hundreds of millions of transistors.

2.3 Challenges Faced In VLSI: The increase in package density as well as the clock frequency of the VLSI circuits has led to capacitive coupling noise. This leads to crosstalk which is one of the most challenging problems in the design and verification of modern VLSI circuits.

In addition, the interconnected lines get denser and finer (and lengthier in case of global interconnects), which results in the escalation of crosstalk noise amplitude time and the circuit faults triggered by such noise sources. Therefore as the VLSI technology scales down the role of interconnect parasitic effects in the signal, integrity becomes even more evident.

2.4 Occurrence of Crosstalk in VLSI:

In integrated circuit design, crosstalk usually refers to a signal disturbing another nearby signal. The circuit components have contracted in size while the overall number of components inside a fixed chip area has increased multifold.

The two terms to be familiar with which leads to crosstalk are the wire to substrate capacitance and the cross coupling capacitance. The vast number of components alongwith small device characteristics increases the juxtaposition between different components of the design. Each specific component gives rise to a capacitance relative to the substrate.

To know about the crosstalk coupling consider the following figures 1.1 and 1.2.

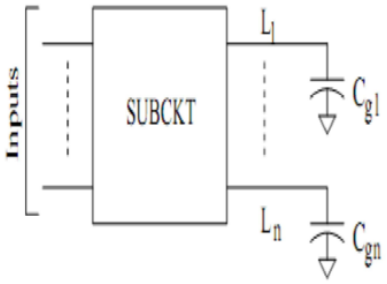

Figure 1.1:Sub-circuit with substrate capacitance

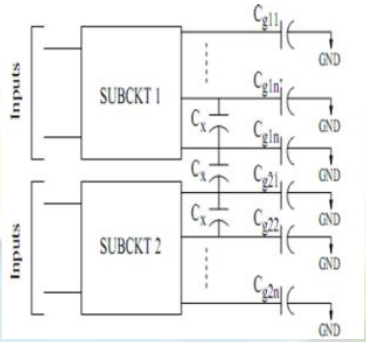

Figure 1.2: Two circuits in close proximity

For example, in Figure 1.1, 1.2 each output line of the CMOS sub circuit has a capacitance Cg, relative to the ground.

Thus, n-output lines have wire-to-substrate and capacitances Cg1through Cgn and Cg11 through Cg2n. Alternatively, a capacitance Cx may exist between closely-located wires of a common sub circuit, closely-located wires of different sub-circuits which are in the proximity of each other [4].

Cross-coupling capacitance is the name of such a capacitance. Both types of cross-coupling capacitances are shown in Figure 1.1, 1.2. If the wire-to-substrate capacitance Cg is much higher than the cross-coupling capacitance Cx, then each line has high driving power and diverse lines do not affect each other in any way.

The cross coupling capacitance to the wire-to-substrate capacitance could be of the ratio 3:1, which is extensively high. Hence the actions which were not related earlier, such as transitions on two closely-located lines begin to impact one another. This occurrence is known as crosstalk and is highly unwelcome in digital circuits, since they have negative effects on both the delay and the power usage in the circuit, leading to signal integrity and reliability botches. This will cause the circuit to fail completely or worse still, function erratically.

For example, a memory bank involving memory cells may be forbidden because of coupling faults between neighbouring memories cells, even though this may be in conformance to the design specification.

3. SOURCES OF CROSSTALK:

The interconnection adopts growing prominence in the design with change in technology. The interconnect not only dominates the delay in the circuit but also takes up about 30% of the total active power in the design [5,7]. The common interconnects are buses as bus-based interconnects diminishes the number of links associated to other interconnect styles. ➢

Buses comprise long lengths of wires running in parallel and in close proximity. With decrease in the space crammed between the lines in comparison to the cross section area of the wires, the cross-coupling capacitance between one bus wire and the others become similar and frequently surpass their wire-to-substrate capacitance.

A detailed portrayal of the various capacitances in relation to the interconnects which run adjacent to one another is provided in Figure 2.

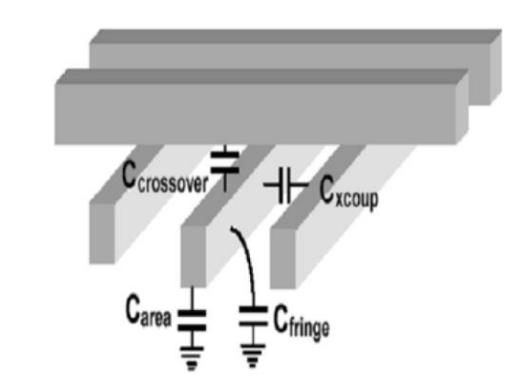

Figure 2: Various capacitance in a design

The area capacitance C area and capacitance due to fringe fields Cfringe both constitute the wire to substrate capacitance. The coupling capacitance Cxcoup subsists amongst neighboring wires running in planes which are alike. The coupling capacitance Crossover subsists amid close wires in dissimilar planes.

When Cxcoup+Ccrossover turn out to be analogous to Carea+Cfringe, the crosstalk occurrence arises and is put to effect. Owing to crosstalk, the circuit gives out incorrect erroneous outcomes due to undesirable delays on some of the lines [6]. Instead, the circuit could error in functioning due to prompted delays on some of the lines which could infringe their required timing. The lines which cause delays on other lines are called aggressors; the lines which are affected by these aggressors are known as victims.

The crosstalk effects amongst aggressors and victims are double and are given as follows:

If the victim wire is at a stable condition value (either logical '0' or logical '1') while the aggressor wire is switching (either from '0' to '1' or from '1' to '0'), it could prompt an undesirable positive or negative spike on the target wire.

If the target wire is making a transition from '0' to '1' and the aggressor wire is also making a transition from '0' to '1', the transition of the target wire will be accelerated since the transition is being assisted by the aggressor.

Alternatively, if the target wire is transitioning from '0' to '1' and the aggressor is transitioning from '1' to '0', the changeover of the victim wire will be hindered since the aggressor is now conflicting the victim's change. The changeover of the victim wire from '1' to '0', with regard to the aggressor's changeover, can be investigated in a similar manner.

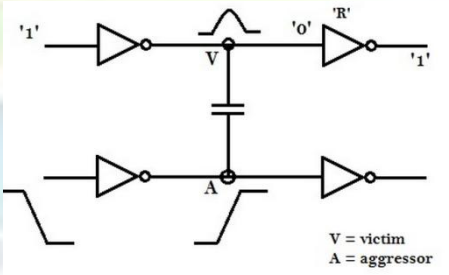

Figure 3 : Crosstalk noise due to coupling capacitance

3.1 Effects of Crosstalk:

- Crosstalk is a cause for worry since it can be a foremost provider to the expanse of jitter present in a device. Jitter is the deviance of average from its usual point. A huge quantity of jitter in a serial communication link may lead to bit errors in the received serial bit stream.
- Crosstalk noise may cause detrimental effects as well as too much overshoot, undershoot, extra signal delay and even a decrease in signal delay.
- To be specific, logic speeds and clock speeds have increased considerably, thus leading to quicker transition times. Also, at such great speeds the inductive properties of the wires, especially mutual inductance, come into play.
- Crosstalk may be the reason for a signal to assume an incorrect value. This is particularly critical when the signal is about to be latched, for a wrong value can be loaded into a storage element.
- Crosstalk may delay the settling of the signal to the exact value. This is called noise-on-delay.

Eg: A large quantity of jitter can cause a timing budget failure in a parallel system, or it can cause a clock and data recovery PLL to incorrectly recapture the data in a serial system.

These effects have increased the interactions between signals and decreased the noise immunity of digital CMOS circuits. This has led to crosstalk noise being a significant problem for digital ICs that must be considered to tape out.

4. DESIGN OF ENCODER AND DECODER FOR BOUNDARY SHIFT CODES:

Switching is one of the most vital contributors to the crosstalk in VLSI circuits. It is vital to encode the bits in such a way that the switching action on the buses is condensed. The design of encoder and decoder for boundary shift codes for reducing crosstalk is discussed here [7]. The bus is modeled as an n-bit communication channel as shown in example figure 3. The data words to be encoded have been denoted by symbols; the symbols and actual data words are mapped to one another.

The values placed on the channel by the encoder are called codeword's; the mapping between symbols and code words is called codebook. If there is a change in codebook over time, then the encoding is said to have memory. During each signaling interval, the encoding scheme is used to select and transmit an n-bit word, called a codeword, from a possibly dynamic set called a codebook.

The codebook can be a function of previously transmitted code words and hence dynamic. The overall encoding scheme is called a code. A code is memory less if it uses a fixed codebook. The rate of code is defined as log base 2mod(Cmin), where mod(Cmin) is the number of code words in the smallest codebook. This is the smallest number of bits that can be encoded during every signaling interval.

Table 1 : Valid and invalid transitions during the encoding process

Code word at time 1:	0010	0000	0100	0100
	↓	↓	↓	↓
Code word at time 2:	0011	1111	0001	0010
	Valid	valid	valid	invalid

In Table 1, the valid and invalid transitions during the encoding process are shown. The four bits are used for transition from one codeword to other, the first, second, third are valid transitions since there is no adjacent bits to switch in opposite directions. But in the fourth codeword, the transition from the code word 1 to code word 2 causes the adjacent bits (bit 2 and 2) to switch in opposite directions. Hence it is invalid.

5. SELF-SHIELDING CODES:

A pair of code words contains an invalid transition if transitioning from one code word to the other causes adjacent bits to switch in opposite directions. For example, the code words in Table 2 contain invalid transition by bits 5 and 6. The transitions shown in Table 2 are undesirable because they increase crosstalk noise.

A code is self-shielding if it does not allow invalid transitions. It is assumed that neighboring wires are routed in parallel. Therefore, the encoded bus is effectively self-shielding. In addition to avoiding invalid transitions, the code words are to be differentiated reliably even in the presence of errors (bit flips).

Table 2 : Invalid transition by bits 5 and 6

C1	→	0	1	1	0	0	0	1	1
C2	→	1	1	0	1	1	0	1	0

This is possible if the code words in the codebook have a large enough Hamming distance (The number of bits which differ between two binary strings, i.e. the distance between two strings A and B is $\sum$ | Ai - Bi |) between them. For example, if the code words have a minimum Hamming distance of three between them, any single error can be corrected since the "noisy" codeword must be closer to the original codeword than to any other. In general, if the minimum Hamming distance between any two code words is 'd', then it can be either corrected up to ([d-1]/2) errors, or detected up to (d- 1) errors.

A binary code is said to be linear if the bitwise sum (mod 2) of any two codeword is a code word. A linear code can be represented by the independent set of basis code words. All other code words can then be formed by a linear combination of these.

A standard representation of a linear code is the generator matrix, a matrix whose rows are an independent set of basis code words. In coding theory, a generator matrix is the basis for a linear code, generating all its possible code words. If the matrix is 'G' and the linear code is C, w= cG, where 'w' is a unique codeword of the linear code C, 'c' is an unique row vector and a bijection exists between 'w' and 'c'. A generator matrix for a (n, M = qk, d), q-code is of dimension k×n. Here 'n' is the length of a codeword, 'k' is the number of information bits, 'd' is the minimum distance of the code and 'q' is the number of symbols in the alphabet (thus, q = 2 indicates a binary code, etc.).

The number of redundant (Redundancy is the number of bits used to transmit a message minus the number of bits of actual information in the message. i.e. , it is the amount of wasted "space" used to transmit certain data) bits is denoted by r = n - k.

The standard form for a generator matrix is G=[Ik P], where 'Ik' is a k×k identity matrix and 'P' is of dimension k×r.

For example, a generator matrix for the code {0000;0011; 0101; 0110; 1001; 1010; 1100; 1111} is

$$G = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

The generator matrix provides a simple way to map information bits to code words by multiplying the generator matrix by a column vector of information bits. A length 'n' binary linear code encoding 'k' bits with minimum hamming distance 'd' is called an [n, k, d] code. Hence these codes are computationally infeasible for moderate to large bus sizes.

5.1. Boundary Shift Codes: A general construction method for practical codes is discussed here. A dependent boundary in a codeword as a position where two adjacent bits differ has been defined. The location is denoted by the position of the leftmost bit of the boundary. If two code words do not share any dependent boundaries, they cannot form an invalid transition. For example, consider the code words shown in Table 3, where c1 and c2 have dependent boundaries {1; 3; 5} and {2; 4; 7}, respectively. Since there is no overlap, the transition must be valid.

Table 3 : Valid Transition

C1	→	0	1	1	0	0	1	1	1
C2	→	1	1	0	0	1	1	1	0

Using this property, it is noted that if a codebook has code words with only even dependent boundaries, then performing a 1-bit circular right shift yields a new codebook with no even dependent boundaries. Since the two codebooks do not have overlapping dependent boundaries, they can be alternated to obtain a self-shielding code. This is commonly called a boundary shift code.

For this construction, an error correction code is needed with no odd dependent boundaries. Let C be an [n; k; d] code and let C^-1 be formed by duplicating each bit position in C. Then C^-1 is a [2n; k; 2d] code with no odd dependent boundaries, since every bit in an odd bit position is followed by a copy. By alternating between C^-1 and a shifted version of it, a [2n; k; 2d] self-shielding code is obtained.

In addition, puncturing C^-1 in the last bit position, i.e., eliminating the final bit in every codeword, yields a [2n- 1; k; 2d-1] code. Using a single parity check code in the above construction gives an infinite class of single error-correcting codes. In a [k+ 1; k; 2] single even parity check code the 'k' data bits are appended with a final bit chosen to make the parity of the codeword even. Applying this construction, a [2k+ 1; k; 3] single error-correcting self-shielding code is obtained.

6. DESIGN OF ENCODER:

In the transmission side, the input data has to be encoded before being transmitted; here the method of encoding the input data by using boundary shift codes is described.

As an example, the [4,3] boundary shift code with generator matrices is considered.

Here 9 is the length (total no of encoded bits to be transmitted), 4 is the dimension (actual no of input bits), 3 is the minimum hamming distance.

2 matrices

$$G0 = \left[\begin{array}{cc|cc|cc|cc|c} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{array}\right]$$

$$G1 = \left[\begin{array}{c|cc|cc|cc|cc} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{array}\right]$$

Generator matrices G0 and G1 are are used for encoding during even and odd clock cycles respectively. Go may be used for all cycles in an equivalent manner with the output then right shifted for odd cycles. It can be inferred that x0, x1, x2, x3 are the four data inputs, y0 to y8 are the encoded data's out of given four input data's. Here the encoded bits y1, y3, y5, y7 are directly got from the input data x0, x1, x2, x3. Here a five 2:1 Mux are used to produce the rest of the encoded bits y0, y2, y4, y6, y8, i.e., five parity bits. The following example illustrates how this code would be used to encode a 4-bit bus. The intermediate pre-shifted output is shown in Table 4 for clarity.

Time	input	preshifted output	output
x3	x2	x1	x0
0	1 0 1 0	110011000	110011000
1	0 1 1 1	001111111	100111111
2	1 0 0 0	110000001	110000001
3	0 1 0 0	001100001	100110000

Y8 y7 y6 y5 y4 y3 y2 y1y0

Table 4 : Preshifted outputs

Consider first input 1010 at time 0, the pre-shifted output is 1 1 0 0 1 1 0 0 0.

Here the hyperlinked bits (i.e. y7, y5, y3. y1) are the original bits of the input (x3, x2, x1, x0).

Then the encoded output, i.e. y8, y6, y4, y2 also represent the input bits 1010. The y0 is based on the given data input. These actions are performed by replicating each input bit, this yields the pre-shifted output. Then append the parity check at the end i.e. y0 yielding the pre-shifted output. Here the time cycle is considered as even so there is no shift, therefore the encoded output is 110011000.

Consider the 1st time cycle which is odd, here the input is 0111, the pre-shifted output is 00111111 1, the hyperlinked bits are the original bits of the input, then on-hyperlinked bits also represent the input bits, y0 depends on the input.

But the change comes because of the clock cycle. Since it is odd, a 1-bit circular right shift is performed before transmitting. Then the encoded bits are transmitted.

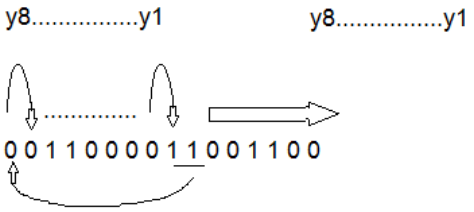

Figure 4 :Preshifted and encoded output

7. DESIGN OF DECODER:

The input data has been encoded and transmitted. At the receiver side, the right shift is first undone, if the clock cycle is odd. Then decoding can be done by popular vote, where the two "copies" of the desired bit are augmented by a third, generated by taking the sum(mod 2) of one copy of each of the other information bits and the parity check.

Figure 4 gives the full description of decoding. The operation of which is illustrated below.

noisy output		majority vote	data
100011000	→	(101)(000)(111)(000)	1010
001111110	→	(001)(110)(110)(110)	0111
010000001	→	(011)(001)(001)(001)	1000
011100000	→	(011)(110)(001)(001)	1100

Table 5 : Output at the receiver

For example, in an even cycle three independent copies of the first information bit are given by y0, y1 and (y2+y4+y6+y8) mod2. A single error will affect at most one of the three copies and is therefore correctable. The following example shows how noisy versions of the code words in the previous example would be decoded after unshifting. The highlighted bits relate to errors. [6, 2] The three code words which are first decoded correctly and correct output are produced at the receiver, because the hamming distance is less than three.

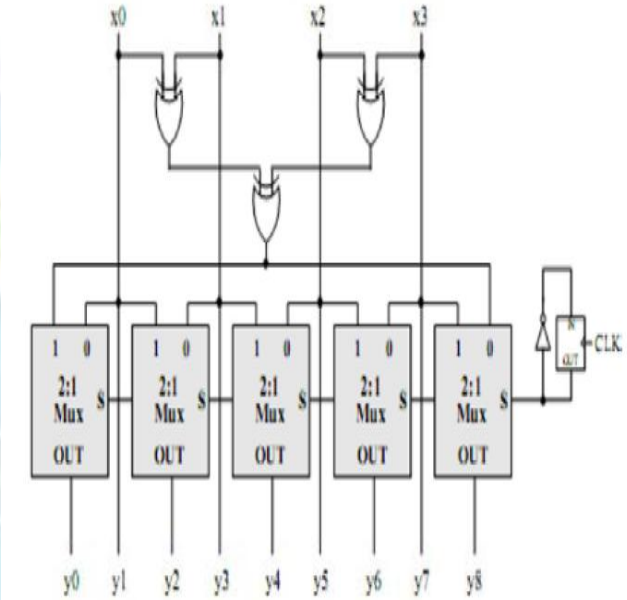

Fig 5.1: Encoding circuit for [9,4,3] code.

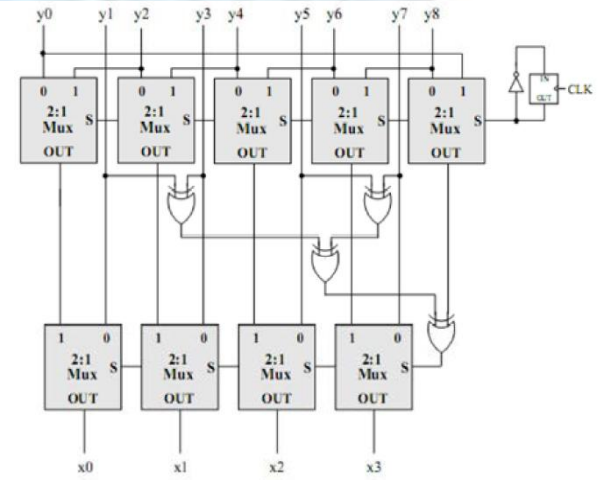

Fig 5.2: Decoding circuit for [9,4,3] code

The last codeword is decoded erroneously as it contains two errors and hence beyond the code's error rectifying capability. This decreases the switching of buses, thus reducing crosstalk.

8. ILLUSTRATION:

For large bus sizes the increased circuit depth may lead to significant delay. This can be reduced by breaking the bus into smaller sub-buses with shielding wires inserted between them. These results in a slight increase in the number of wires and gates needed, but limits the circuit depth.
In addition, it also increases the error correction capability, since single errors in each sub-bus can then be corrected independently. Since the codes constructed in the previous section are based on very simple error-correcting codes, they can be encoded and decoded in an effective manner. Figures 5.1 and 5.2 show an encoder and decoder respectively for the [4 ,3] code.
These circuits can be general in a direct way for larger single error-correcting codes. Gate counts and maximum circuit depths are given in Table 6, for a range of bus sizes and closed form, expressions are given for the general case.

Table 6 : Encoder/decoder gate counts and delay for single error correcting boundary shift codes.

Bus	Wire	Encoder		Decoder	
size	s	Gates	Delay	Gates	Delay
4	9	10	3 gates	15	4 gates
8	17	18	4 gates	27	5 gates
16	33	34	5 gates	51	6 gates
32	65	66	6 gates	99	7 gates
64	129	130	7 gates	195	8 gates
n	2n+1	2n+2	[log2 n] +1	3n+3	[log2 (n+1)] +1

9. ADVANTAGES:

A potentially useful feature of the proposed codes is that they are systematic, that is, the information bits are embedded in the encoded codeword and can therefore be obtained without any decoding logic. For example, the information bits of the [4, 3] code are given by bits y1, y3, y5 and y7.

Simply using these bits rather than decoding the full codeword sacrifices the error correction capabilities of the code. However one can imagine a scenario where error correction may only be necessary for certain destinations on the bus that are relatively far from the source, while other destinations may opt for error detection or simply picking the information bits off of the encoded codeword.

A clear advantage of boundary shift codes is that they do not suffer from error propagation since the codebook does not depend on the choice of previous code words transferred; but it is only a function of the time index. As long as the source and destination are synchronized, no error propagation will occur.

Scalable construction-simple encoder/decoder- integer rates; systematic.

10. CONCLUSION:

The designing bus encoding schemes that provide both crosstalks minimization and active error correction have been considered here. This is an important improvement and is particularly applicable to scenarios, such as nanotechnology and radiation hardened circuits, where random errors are a concern in addition to crosstalk interference.
One of the most significant contributions of this paper is a practical class of error-correcting self-shielding codes called boundary shift codes. For a particular case of single error-correcting boundary shift codes, given gate level encoding and decoding circuits, writing a verilog code for the above proposal, simulate it and execute it have been considered

REFERENCES:

[1] Victor, B. and K. Keutzer, "Bus encoding to prevent crosstalk delay". Proc. IEEE/ACM Intl. Conf. on Computer Aided Design, pp: 57-69. 2001.
[2] Ketann, Patel AndIgorl and Marko, "Error-correction and crosstalk avoidance in DSM busses".
[3] Sainarayanan, K.S., C. Raghunandan, Ravindra and J.V.R. Srinivas, "Bus coding to minimize redundant bit transitions" M.B. TENCON. 2007.
[4] Saravanan, T. and R. Udayakumar, "Comparison of Different Digital Image watemarking techniques", Middle-East Journal of Scientific Research, ISSN:1990-9233, 15(12): 1684-1690. 2013.
[5] Saravanan, T. and R. Udayakumar, "Optimization of Machining Hybrid Metal matrix Composites using desirability analysis", Middle-East Journal of Scientific Research, ISSN: 1990-9233,15(12): 1691-1697. 2013.
[6] Saravanan, T. and R. Udayakumar, "Simulation Based line balancing of a single piece flow line", Middle-East Journal of Scientific Research, ISSN:1990-9233, 15(12): 1698-1701. 2013
[7] Saravanan, T. and R. Udayakumar, "Comparison of Different Digital Image watemarking techniques", Middle-East Journal of Scientific Research, ISSN:1990-9233, 15(12): 1684-1690. 2013.

