International Journal of Scientific Engineering and Research (IJSER)
www.ijser.in
ISSN (Online): 2347-3878
Volume 1 Issue 2, October 2013

Crosstalk Avoidance using Fibonacci Code Word for Sub Micrometer on Soc Application

Srinivas Rao. G[1], P. M. Francis[2], B. Prasad Kumar[3]

[1]M. Tech, GITAS College, Bibbili, A.P, India

[2]Department of ECE, HOD, Assistant Professor, GITAS, College, Bobbili, India

[3]Department of ECE, Assistant Professor, GITAS College, Bobbile, India

Abstract: ***In this paper we are proposing the long bus wire connection and cross correlation principals that provide the Propagation delay across long on-chip buses is significant manner when adjacent wires are transitioning in opposite direction (i.e., crosstalk transitions) as compared to transitioning in the same direction. By exploiting Fibonacci number system, we propose a family of Fibonacci coding techniques for crosstalk avoidance, relate them to some of the existing crosstalk avoidance techniques, and show how the encoding logic of one technique can be modified to generate code words of the other technique.***

Keywords: On-chip bus, crosstalk, Fibonacci coding.

1. Introduction

In the deep submicrometer CMOS process technology, the interconnect resistance, length, and inter-wire capacitance are increasing significantly, which contribute to large on-chip interconnect propagation delay [1], [2]. Data transmitted over interconnect determine the propagation delay and the delay is very significant when adjacent wires are transitioning in opposite directions (i.e., crosstalk transitions) as compared to transitioning in the same direction. Several techniques have been proposed in literature to eliminate crosstalk transitions. A simple technique to eliminate crosstalk transitions is to insert a shield wire between every pair of adjacent wires [3]. As there is no activity on shield wires, the shielding (SHD) technique completely eliminates crosstalk transitions. Abstracted from the concept of shielding, forbidden transition coding (FTC) technique with/without memory is proposed in [4]. For 32-bit data, both memory-based and memory-less FTC techniques require 40 and 46 wires, respectively, as compared to 63 wires required by the SHD technique. Note that the memory-based FTC technique is very complex as compared to the memory-less FTC technique. Forbidden pattern coding (FPC) technique [5] prohibits 010 and 101 patterns from codewords, which in turn eliminates crosstalk transitions. It requires 52 wires for a 32-bit bus. No adjacent transition (NAT) coding is proposed in [6]. (n,b,t)-NAT codes, where **'b'** is the dataword width, **'n'** is the codeword width, and **t**'is the maximum number of 1s allowed in codewords, are designed in such a way that no two adjacent 1s are present in codewords. NAT codes are transmitted using the transition signaling technique [7]. For **'n'** bit codewords, the maximum number of (n,b,t) – NAT code is $\sum_{i=0}^{t} C(n+1,-i,i),0 \leq t \leq \left[\frac{n}{2}\right]$ (6)
where t = n/2 the cardinality of the(n,b,t) – NAT code word set is $\boldsymbol{fn}$ where fn is the n Fibonacci series . By relating Fibonacci number system to crosstalk-free codes, we proposed a crosstalk-free bus encoding technique [8] and provided a recursive procedure to generate such codes. Crosstalk-free codes generated in [8] are same as that of the memory-less FTC technique [4]. By combining the ideas of [4], [5], [8], efficient codec designs for crosstalk avoidance are proposed in [9], [10]. In forbidden transition free crosstalk avoidance coding (FTF-CAC) [9], data are encoded using Fibonacci number system in such a way that 01 or 10 on two adjacent bits are prohibited. In forbidden pattern free crosstalk avoidance coding [10], data are encoded using Fibonacci number system in such a way that 010 and 101 patterns are prohibited. An iterative implementation strategy for generating crosstalk-free codes is proposed in [11], wherein a set of n bit crosstalk-free codes can be used to derive (n+1) -bit crosstalk-free codes. As a case study, the authors have implemented (n,b,[n/2]) NAT coding technique [6] using Fibonacci number system (n+1, b ,[(n+1)/2]) –NAT codewords are generated using the subgroups of (n-1,b [(n-1)/2]) NAT codewords and (n,b,[n/2]) NAT codewords and (n,b,[n/2]) _-NAT codewords is related to a Fibonacci number.

One common thing among the techniques proposed in [8]–[11] is that for a given dataword, an equivalent codeword is generated in Fibonacci number system, i.e., for every dataword $d = dn\ldots\ldots d0$, a $code$ $word$ $c = cn\ldots\ldots c0$ is generated such that $\sum_{i=0}^{n} di\ 2i = \sum_{i=0}^{m} cifi$ where fi is the I the Fibonacci number. By exploiting Fibonacci number system, we propose a family of Fibonacci coding techniques for crosstalk avoidance, give a generalized framework to generate crosstalk avoidance codes, and establish relationship between different crosstalk avoidance coding techniques.

2. Fibonacci Number System

A number system S= (U,C) is defined by a strictly increasing sequence of positive integers U =(un) n .>0 and a finite subset C of positive integers. Elements of sets U and C

International Journal of Scientific Engineering and Research (IJSER)
www.ijser.in
ISSN (Online): 2347-3878
Volume 1 Issue 2, October 2013

are called the *basis* elements and *digits* of the number system, respectively. A positive integer N is the number system S = (U,C) is represented by a finite sequence of elements $d = dn\ldots\ldots d0$ *of* C *such that* $N = \sum_{i=0}^{n} diUi$ The *binary number system* is defined as S = ((2n)n>0{0,1}) *Fibonacci number system* [12] of order s,s>2 is defined as S= (Fs,{0,1}) where Fs = (*fn*)n>0 such that

fi= 2i for 0< i< s-1
fi = fi-1 +…+fi-s for i> s

It has been shown that Fibonacci number system of order s,s>2 is complete [13], i.e., every integer has a representation in S = (Fs,,{0,1}). Note that an integer may have multiple representations in Fibonacci number system of order s, s>2. To overcome the ambiguity in representing integers in Fibonacci number system of order s,s>2 a *normal-form* [13] is defined, wherein each integer has a unique representation which does not contain s consecutive bits equal to 1.

3. Exploiting Fibonacci Representations For Crosstalk Avoidance

Throughout the paper, we use notation *dataword* and *codeword* for data to be encoded and encoded data, respectively. We assume that datawords are represented in the binary number system. For every dataword, we give a codeword using Fibonacci number system of order 2 such that the decimal equivalent of the dataword is same as that of the codeword.

A. Normal-Form Fibonacci (NFF) Coding Technique

We describe NFF technique in two parts, namely, *data encoding* and *data transmission*. For data encoding, we use normal-form Fibonacci number system of order 2. For a □-bit dataword, D = dn-1dn-2….d0 using the NFF technique, the unique m bit codeword, nc= cm-1cm-2….c0 can be generated using NFF encoding algorithm as shown in Table I. Let *NFFn be the set*

Input: d;
$r_m = d$;
for $k = m - 1$ to 1 **do**
if $r_{k+1} < f_k$ **then**
$c_k = 0$;
else
$c_k = 1$;
end if
$r_k = r_{k+1} - f_k \cdot c_k$;
end for
$c_0 = r_1$;
Output: $c_{m-1}c_{m-2}\cdots c_0$;

of n bit NFF code words. Table II gives 4-bit code words for 3-bit data words.

Input: d;
$r_m = d$;
for $k = m - 1$ to 1 **do**
if $r_{k+1} < f_{2\lceil \frac{k-1}{2} \rceil}$ **then**
$c_k = 0$;
else
$c_k = 1$;
end if
$r_k = r_{k+1} - f_{k-1} \cdot c_k$;
end for
$c_0 = r_1$;
Output: $c_{m-1}c_{m-2}\cdots c_0$;

Table 1: Conversation for 3 bit data code to 4 bit data code

data-word			Fibonacci codeword											
			$\mathcal{NFF}_4$				$\mathcal{RF}_4$				$\mathcal{CRF}_4$			
4	2	1	5	3	2	1	3	2	1	1	3	2	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	1	0	0	1	0
0	1	0	0	0	1	0	0	1	0	0	0	0	1	1
0	1	1	0	1	0	0	0	1	0	1	1	0	0	0
1	0	0	0	1	0	1	0	1	1	1	1	0	1	0
1	0	1	1	0	0	0	1	1	0	0	1	0	1	1
1	1	0	1	0	0	1	1	1	0	1	1	1	1	0
1	1	1	1	0	1	0	1	1	1	1	1	1	1	1

Second column in the table refers to the NFF code words. can be generated recursively as follows:

NFF0 = Ø
NFF1 = {0,1}
NFFm+2 = {0x,10y| x €}

Table IV shows 1-bit to 4-bit NFF codewords. Note that NFFm is same as the set of (n,b,[n/2])NAT code words [6].

For implementing -NAT coding technique, though it is not explicitly mentioned in the paper [11], the authors indeed considered normal-form Fibonacci number system of order 2. Hence, NFF technique can implement (n,b,[n/2])-NAT coding technique, and vice versa. Uniqueness property of the normal-form Fibonacci number system of order 2 prohibits two consecutive 1s to present in code words.

Table 2: 1 bit to 4 bit data code word

$\mathcal{NFF}_1$	$\mathcal{NFF}_2$		$\mathcal{NFF}_3$			$\mathcal{NFF}_4$			
1	2	1	3	2	1	5	3	2	1
0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	1	0	0	0	1
	1	0	0	1	0	0	0	1	0
			1	0	0	0	1	0	0
			1	0	1	0	1	0	1
						1	0	0	0
						1	0	0	1
						1	0	1	0

From Table IV, we can see that NFF codewords do not contain adjacent 1s. We now formally prove this fact.

ISSN (Online): 2347-3878
Volume 1 Issue 2, October 2013

Table 3: Illustration of different coding techniques

data-word			Fibonacci codeword															
			$\mathcal{NFF}_4$				$\mathcal{NFF}_4^{TS}$				$\mathcal{RF}_4$				$\mathcal{CRF}_4$			
0	**1**	**0**	0	0	1	0	0	0	1	0	0	1	0	0	0	0	1	1
1	**0**	**1**	**1**	**0**	**0**	0	1	0	1	0	1	1	0	0	1	0	1	1
0	**1**	1	**0**	**1**	**0**	0	1	1	1	0	0	1	0	1	1	0	0	0
1	1	0	**1**	**0**	**0**	**1**	0	1	1	1	1	1	0	1	1	1	1	0
1	1	1	1	**0**	**1**	**0**	1	1	0	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0
0	**0**	**1**	0	**0**	**0**	**1**	1	1	0	0	0	0	0	1	0	0	1	0
0	**1**	**0**	0	**0**	**1**	**0**	1	1	1	0	0	1	0	0	0	0	1	1

4. Dependency Among Different Crosstalk Avoidance Codes

In this section we show how one set of Fibonacci codes is related to the other set of Fibonacci codes. We also relate our Fibonacci techniques with other crosstalk avoidance coding techniques. Lemma 4: Rfm ={x ØAm | €NFFm } where Øid the bit wise

4.1 Redundant Fibonacci (RF) Coding Technique

We now present a coding technique which does not require the TS technique to eliminate crosstalk transitions. In the case of NFF technique, Fibonacci numbers are considered as the basis elements to generate bit codewords. Similar to the NFF technique, in *redundant Fibonacci* (RF) coding technique, we consider Fibonacci numbers as the basis elements with the exception that is used twice. That is, in order to generate bit RF codewords, we consider as the basis elements. As is considered twice in the RF technique, we get two sets of RF codewords, each is a complement of the other. We consider these two sets as *redundant Fibonacci* (RF) and *complement redundant Fibonacci* (CRF) codeword sets. sHence, the encoding logic given in [9] can be used for implementing the RF technique. CRF codewords are generated by taking bit-wise complement of each codeword from the set of RF codewords. Let be the set of bit CRF codewords. Shows 1-bit to 4-bit CRF codewords. Third and fourth columns of Table III give 4-bit RF and CRF codewords, respectively, for given 3-bit datawords. CRF encoding algorithm as shown in Table II is similar to the encoding algorithm given in [9] for implementing FTF-CAC technique. The only difference is the comparison operation. Instead from the implementation point of view, the CRF algorithm has the same complexity as that of the FTF-CAC algorithm [9].

5. Experimental Result

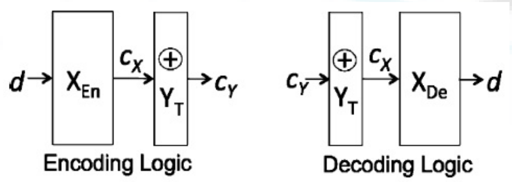

Figure 1: Describe the Code word how we are uploading

$Y_T \rightarrow$	**(0101)**	**(1010)**
NFF	RF	CRF
0000	0101	1010
0001	0100	1011
0010	0111	1000
0100	0001	1110
0101	0000	1111
1000	1101	0010
1001	1100	0011
1010	1111	0000

$Y_T \rightarrow$	**(0101)**	**(1111)**
RF	NFF	CRF
0000	0101	1111
0001	0100	1110
0100	0001	1011
0101	0000	1010
0111	0010	1000
1100	1001	0011
1101	1000	0010
1111	1010	0000

$Y_T \rightarrow$	**(1010)**	**(1111)**
CRF	NFF	RF
0000	1010	1111
0010	1000	1101
0011	1001	1100
1000	0010	0111
1010	0000	0101
1011	0001	0100
1110	0100	0001
1111	0101	0000

Figure 2: Explains the FTF Method with the code word adjustment

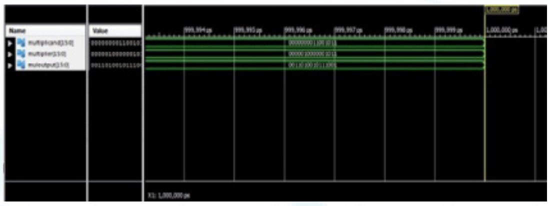

Figure 3: Compression data of Fibonacci series

6. Conclusion

By exploiting Fibonacci number system, we proposed a family of Fibonacci coding techniques for crosstalk avoidance. We showed the inter-dependency among the proposed techniques describe and provided a formal procedure to convert a codeword set into another codeword set. We also related our proposed techniques with some of the existing crosstalk avoidance coding techniques. The proposed techniques eliminate crosstalk completely, but not inductance. The worst-case inductance occurs when adjacent lines transition in the same direction. We plan to come up with a suitable mechanism to minimize the inductance effects using Fibonacci codes in future.

References

[1] F. Caignet, S. Delmas-Bendhia, and E. Sicard, "The challenge of signal integrity in deep-submicrometer CMOS technology," Proc. IEEE, vol. 89, no. 4, pp. 556–573, Apr. 2001.

[2] D. Pamunuwa, L.-R. Zheng, and H. Tenhunen, "Maximizing throughput over parallel wire structures in the deep submicrometer regime," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 11, no. 2, pp. 224–243, Apr. 2003.

[3] R. Arunachalam, E. Acar, and S. Nassif, "Optimal shielding/spacing metrics for low power design," in Proc. IEEE Comput. Soc. Annu.Symp. VLSI, 2003, pp. 167–172.

[4] B. Victor and K. Keutzer, "Bus encoding to prevent crosstalk delay," in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, 2001, pp. 57–63.

[5] C. Duan, A. Tirumala, and S. Khatri, "Analysis and avoidance of crosstalk in on-chip buses," in Proc. Hot Interconnects, 2001, pp. 133–138.

[6] P. Subramanya, R. Manimeghalai, V. Kamakoti, and M. Mutyam, "A bus encoding technique for power and cross-talk minimization," in Proc. IEEE Int. Conf. VLSI Design, 2004, pp. 443–448.

[7] M. Stan and W. Burleson, “Limited-weight codes for low power I/O,” in Proc. IEEE/ACM Int. Workshop Low Power Design, 1994, pp. 209–214.
[8] M. Mutyam, “Preventing crosstalk delay using Fibonacci representation,” in Proc. IEEE Int. Conf. VLSI Design, 2004, pp. 685–688.
[9] C. Duan, C. Zhu, and S. Khatri, “Forbidden transitions free crosstalk avoidance codec design,” in Proc. Design Autom. Conf., 2008, pp. 986–991.
[10] C. Duan, V. C. Calle, and S. Khatri, “Efficient on-chip crosstalk avoidance codec design,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., Vol. 17, no. 4, pp. 551–560, Apr. 2009.
[11] K. Karmarkar and S. Tragoudas, “Scalable codeword generation for coupled buses,” in Proc. Design, Autom., Test Eur., 2010, pp. 729–734.
[12] A. Fraenkel, “Systems of numeration,” Amer. Math. Monthly, no. 92, pp. 105–114, 1985.
[13] W. Kautz, “Fibonacci codes for synchronization control,” IEEE Trans. Inf. Theory, vol. 11, no. 2, pp. 284–292, Apr. 1965.
[14] P. Sotiriadis and A. Chandrakasan, “Reducing bus delay in sub-micron technology using coding,” in Proc. IEEE/ACM Asia South Pacific Design Autom. Conf., 2001, pp. 109–114.