International Journal of Computer and Communication Technology

Volume 7 | Issue 2 Article 13

April 2016

Forbidden Free Pattern Crosstalk Avoidance

R. Srivasavi
Department Of Electronics & Comm. Engineering, Nimra College of Engineering & Technology, Ibrahimpatnam, Vijayawada, rsvasavi@gmail.com

P. Srinivas
Department Of Electronics & Comm. Engineering, Nimra College of Engineering & Technology, Ibrahimpatnam, Vijayawada, India, akurathi.srinivasarao@gmail.com

A.Srinivasa Rao
Department Of Electronics & Comm. Engineering, Nimra College of Engineering & Technology, Ibrahimpatnam, Vijayawada, asrinivasarao@gmail.com

Forbidden Free Pattern Crosstalk Avoidance

R.Srivasavi, A.Srinivasa Rao & P.Srinivas

Department Of Electronics & Comm. Engineering,
Nimra College of Engineering & Technology, Ibrahimpatnam, Vijayawada
E-mail : rsvasavi@gmail.com

Abstract **-** In this work, we present the crosstalk is a noise caused by inter-wire coupling capacitance between adjacent wires, and causes logic malfunctions and delay faults. Since the degree of crosstalk interference is determined by various factors, such as physical inter-wire coupling capacitance, length of wires running in parallel, switching speed of signals, and transition timing of adjacent signals, it is very difficult for VLSI designers to estimate the crosstalk influence in sufficient accuracy. In this paper, a technique for reduction of maximum bus delay caused by crosstalk is proposed. By approximated equation of bus delay, it becomes clear that our technique is effective for repeater-inserted bus. The result of simulation shows the total bus delay can be reduced.

Keywords **-** *Sub-Micron (DSM), crosstalk avoidance code, forbidden transition overlapping codes (FTOCs) and forbidden pattern overlapping codes (FPOCs), 16-level Encoder.*

I. INTRODUCTION

One of the greatest challenges in Deep Sub-Micron (DSM) design is inter-wire crosstalk, which becomes significant with shrinking feature sizes of VLSI fabrication processes and greatly limits the speed and increases the power consumption of an IC. This monograph presents approaches to avoid crosstalk in both on-chip as well as off-chip busses.

This work presented in the first part of this monograph focuses on crosstalk avoidance with bus encoding, one of the techniques that effectively mitigates the impact of on-chip capacitive crosstalk and improves the speed and power consumption of the bus interconnect. This technique encodes data before transmission over the bus to avoid certain undesirable crosstalk conditions and thereby improves the bus speed and/or energy consumption. We first derive the relationship between the inter-wire capacitive crosstalk and signal speed as well as power, and show the data pattern dependence of crosstalk. A system to classify busses based on data patterns is introduced and serves as the foundation for all the on-chip crosstalk avoidance encoding techniques. The first set of crosstalk avoidance codes (CACs) discussed are memory less codes. These codes are generated using a fixed code-book and solely dependent on the current input data, and therefore require less complicated CODECs. We study a suite of memory less CACs: from 3C-free to 1C-free codes, including code design details and performance analysis. We show that these codes are more efficient than conventional crosstalk avoidance techniques. We discuss several CODEC design techniques that enable the construction of modular, fast and low overhead CODECs. The second set of codes presented are memory based CACs. Compared to memory less codes, these codes are more area efficient. The corresponding CODEC designs are more complicated, however, since the encoding/decoding processes require the current input and the previous state. The memory-based codes discussed include a 4C-free code, which requires as little as 33% overhead with simple and fast CODEC designs. We also present two general memory-based codeword generation techniques, namely the "code-pruning"-based algorithm and the ROBDD-based algorithm. We then further extend the crosstalk avoidance to multi-valued bus interconnects. The crosstalk classification system is first generalized to multi-valued busses and two ternary crosstalk avoidance schemes are discussed. Details about the ternary driver and receiver circuit designs are also presented in the monograph.

II. BACKGROUND

It is seen for DSM processes, C_I (Inter connect Capacitance) is much greater than C_L (Load Capacitance). Based on the energy consumption and delay models given in [8], the bus energy consumption can be derived as a function of the total crosstalk over the entire bus. The worst case delay, which determines the maximum speed of the bus, is limited by the maximum crosstalk that any wire in the bus incurs. It

has been shown that reducing the crosstalk boosts the bus performance significantly [3]. Different approaches have been proposed for crosstalk reduction in the context of bus interconnects. Some schemes focus on reducing the energy consumption, some focus on minimizing the delay and other schemes address both. The simplest approach to address the inter-wire crosstalk problem is to shield each signal using grounded conductors.

Since the worst case crosstalk is a result of simultaneous transition of data bits in synchronous busses, skewing of transition times of adjacent bits can alleviate the crosstalk effect. Using the technique that intentionally introduces transition time skewing can prevent simultaneous opposite transitions between adjacent wires and consequently reduce the worst case bus delay.

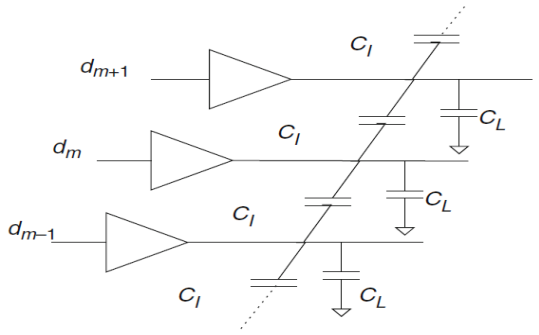


Fig 1. Bus Model Cross talk

This intentional skewing is achieved by adding different time shifts to different bits in the bus. The bus wires alternatively have normal timing and shifted timing, hence no adjacent wires switch at the same time. The shifting in time can be achieved by a delay line or by using a two-phased clock.

III. SELF SHIELDING CODES

Crosstalk avoidance bus encoding techniques manipulate the input data before transmitting them on the bus. Bus encoding can eliminate certain undesirable data patterns and thereby reduce or eliminate crosstalk, with much lower area overhead than the aforementioned straightforward shielding techniques [6]. These types of codes are referred to as crosstalk avoidance codes (CACs) or self shielding codes. Depends on their memory requirements, CACs can be further divided into two categories: memory less codes and memory-based codes. Memory based coding approaches generate a codeword based on the previously transmitted code and the current data word to be transmitted [4, 10]. On the receiver side, data is recovered based on the received codewords from the current and previous cycles. The memory less coding approaches use a fixed code book to generate codewords to transmit. The codeword is solely dependent on the input data. The decoder in the receiver uses the current received codeword as the only input in order to recover the data. Among all the memory less CACs proposed, two types of codes have been heavily studied.

The first is called forbidden pattern free (FPF) code and the second type of code has the property that between any two adjacent wires in the bus, there will be no transition in opposite directions in the same clock cycle. Different names have been used in the literatures for the second type of codes. In this paper, we refer these codes as forbidden transition free (FTF) CACs. Both FPF-CACs and FTF-CACs yield the same degree of delay reduction as passive shielding while requiring much less area overhead. Theoretically, the FPF-CAC has slightly better overhead performance that the FTF-CAC. In practice, for large size bus, this difference is negligible. In this monograph, we also discuss some memory-based encoding techniques. Two such coding approaches are offered in this monograph: the "code pruning" and the "ROBDD-based" method. Our theoretical analysis shows that memory-based codes offer improved overhead efficiency, compared to memory less codes, and the experimental results confirm this analysis.

Both bus partitioning based designs yield small and fast CODECs. Unfortunately, they are not efficient because of the additional wires required to handle crosstalk across adjacent groups. In this section, we propose two CODEC designs that allow us to encode data to the FPF-CAC without partitioning the bus. The mapping schemes allow us to systematically construct the FPF-CAC or FTF-CAC CODECs for busses of arbitrary size.

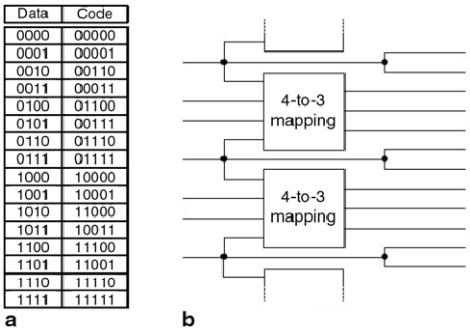


Fig. 2 FPF bit overlapping encoder

By "systematically", we mean that the CODEC for a larger bus is obtained by extending the CODEC of a smaller bus. The gate counts of the proposed CODEC implementations grow approximately quadratically with respect to the bus size, instead of exponentially for previous approaches.

The Fibonacci binary numeral system (FNS) was first mentioned in the context of CAC designs by Mutyam in [5]. The author proposed an inductive codeword generation algorithm for the forbidden transition free code. The algorithm is similar to those proposed in [3]. However, [5] failed to address the mapping scheme and CODEC design. We next describe our FNS-based mapping, and the resulting CODEC designs.

A numeral system is "a framework where numbers are represented by numerals in a consistent manner" [5]. The most commonly used numeral system in digital design is the binary numeral system, which uses powers of two as the basis. Its binary representation is defined below. The binary numeral system is complete and unambiguous, which means that each number has one and only one representation in the binary numeral system.

The Fibonacci-based numeral system N (Fm, {0, 1}) is the numeral system that uses Fibonacci sequence as the basis. The definition of the Fibonacci sequence [3] is given in below Eq 2 . A number is represented as the summation of some Fibonacci numbers, and no Fibonacci number is in the summation more than once, as indicated in Eq.1.

$$\left.\begin{aligned} v &= \sum_{k=1}^{n} b_k \cdot 2^{k-1} \qquad b_k \in \{0,1\} \\ &= \sum_{k=1}^{m} d_k \cdot f_k \qquad d_k \in \{0,1\} \end{aligned}\right\} \quad \text{Eq. (1)}$$

$$f_m = \begin{cases} 0 & \text{if } m = 0, \\ 1 & \text{if } m = 1, \\ f_{m-1} + f_{m-2} & \text{if } m \geq 2. \end{cases} \quad \text{Eq. (2)}$$

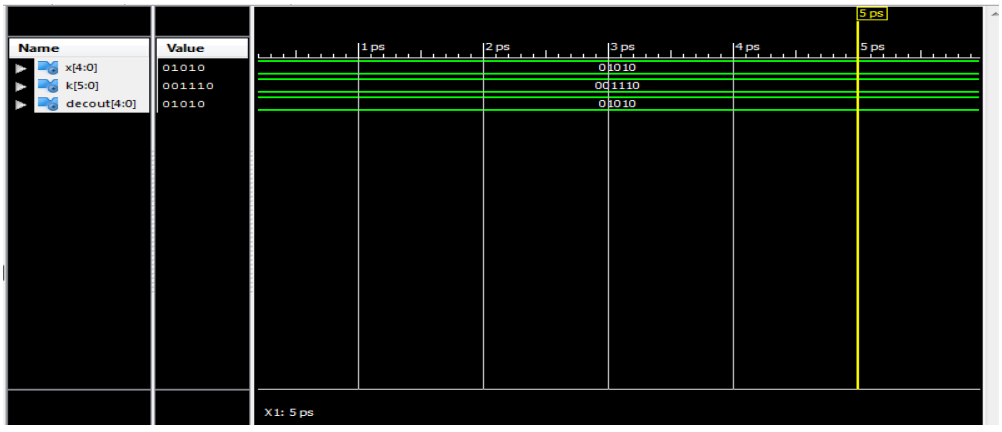
Similar to the binary numeral system, the Fibonacci-based numeral system is complete, and therefore any number v can be represented in this system. However, the Fibonacci-based numeral system is ambiguous. As an example, there are six 7-digit vectors in the Fibonacci numeral system for the decimal number 19: {0111101, 0111110, 1001101, 1001110, 1010001, 1010010}. For clarity, we refer to a vector in the binary numeral system as a binary vector or binary code; a vector in the Fibonacci numeral system is referred to as a Fibonacci vector or Fibonacci code.

All the Fibonacci vectors that represent the same value are defined as equivalent vectors. Another very important identity of the Fibonacci sequence that is used in the following discussions is

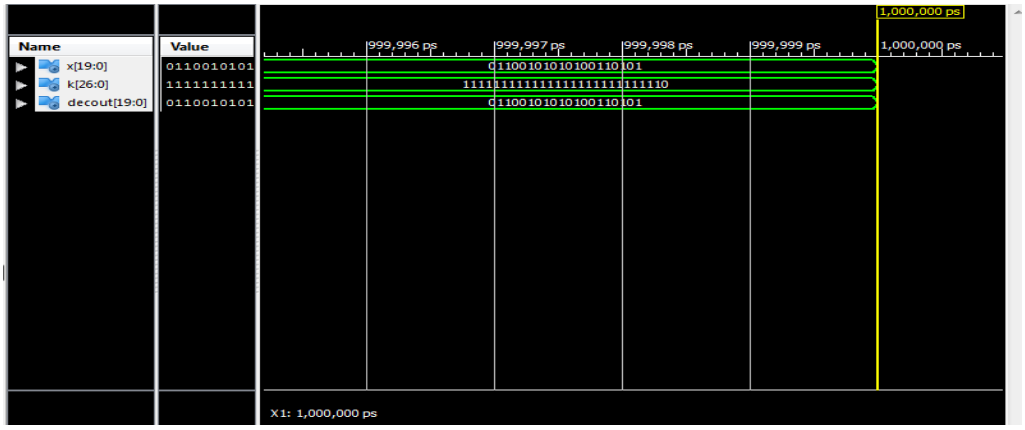
$$\left. f_m = \sum_{k=0}^{m-2} f_k + 1 \right\} \quad \text{Eq. (3)}$$

The n-bit binary vector can represent numbers in the range of [0, 2n−1], and therefore a total of 2^n values can be represented by n-bit binary vectors. From Eq. 3, we know that the range of an m-bit Fibonacci vector is [0, f_{m+2}−1], where the minimum value 0 corresponds to all the bits dk being 0, and the maximum value corresponds to all d_k being 1. Hence a total of f_{m+2} distinct values can be represented by m-bit Fibonacci vectors.

We first propose a coding scheme that converts the input data to a forbidden pattern free Fibonacci vector. The code is near-optimal since the required overhead is no more than 1 additional bit.. The coding algorithm is developed based on a result that states that any number v can be represented in FNS, in an FPF manner.



(a)



(b)

Fig 3. a & b : CODEC Simulation Result for 8 bit and 16 bit implementation

Algorithm Near-Optimal FPF-CAC encoding algorithm: FPF-CAC(v)

\\ MSB stage:
if $v \geq f_{m+1}$ **then**
  $d_m = 1$;
  $r_m = v - f_m$;
else
  $d_m = 0$;
  $r_m = v$;
end if
\\ other stages:
for k = m-1 to 2 **do**
  if $r_{k+1} \geq f_{k+1}$ **then**
    $d_k = 1$;
  else if $r_{k+1} < f_k$ **then**
    $d_k = 0$;
  else
    $d_k = d_{k+1}$;
  end if
  $r_k = r_{k+1} - f_k \cdot d_k$;
end for
LSB
$d_1 = r_2$;
return $(d_m d_{m-1} \cdots d_1)$;

2-bit	3-bits	4-bits	5 bit	
00	000	0000	00000	10100
01	001	0001	00001	10101
11	100	0100	00100	10111
	101	0101	00101	11100
	111	0111	00111	11101
		1100	10000	11111
		1101	10001	
		1111		

Table1: FPF-CAC codewords for 2,3,4 and 5 bit busses

IV. CONCLUSIONS

As part of a complete solution for memory less codes, we also discussed in detail some efficient CODEC design approaches for these codes. The bus-partition based solution is simple and allows us to control the CODEC size and speed. Our solutions based on the Fibonacci Numeral System are advantageous over conventional solutions because of their significantly reduced area, simple design, deterministic mapping, reusable modular structure and most importantly, the extreme low complexity and high speed. We also investigated memory-based codes and showed that they are more efficient in terms of area overhead. We believe that the issue of crosstalk avoidance in on-chip interconnect to reduce delay and power will attract more and more attention as the technology marches ahead. Beyond the research presented in this monograph, there is room for further improvement.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their comments which were very helpful in improving the quality and presentation of this paper.

REFERENCES:

[1] Berkeley predictive technology modeling homepage. http://www-device.eecs. berkeley.edu/bptm/.

[2] BSIM4 offcial release site. http://www-device.eecs.berkeley.edu/bsim3/bsim4.html.

[3] Fibonacci number (FromWikipedia). http://en.wikipedia.org/wiki/Fibonaccinumber.

[4] Moore's law. http://en.wikipedia.org/wiki/Moore´s_law.

[5] Numeral system. http://en.wikipedia.org/wiki/Numeral_system.

[6] Physical Design Modelling and Verification Project (SPACE Project). http://cas.et.tudelft.nl/research/space/html.

[7] K. Hirose and H. Yasuura, "A bus delay reduction technique considering crosstalk," in Proc. Des. Autom. Test Eur. Conf. Exhibition, 2000, pp. 441–445.

[8] B. Victor, "Bus encoding to prevent crosstalk delay," M.S. thesis, Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, 2001.

[9] S.R. Sridhara, A. Ahmed, and N. R. Shanbhag, "Area and Energy-Efficient Crosstalk Avoidance Codes for On-Chip busses", Proc. of ICCD, 2004, pp 12-17.

[10] C. Duan, A.Tirumala and S.P.Khatri, "Analysis and Avoidance of Cross-talk in On-chip Bus", HotInterconnects, 2001,pp 133-138.

[11] Bret Victor and K. Keutzer,"Bus Encoding to Prevent Crosstalk Delay", ICCAD, 2001, pp 57-63.

[12] C. Duan, K. Gulati and S. P. Khatri, "Memory-based Cross-talk Canceling CODECs for On-chip busses", ISCAS 2006, pp 4-9.

[13] Signal encoding scheme for low-power interface design,˘T Proc. of IEEE/ACM International Conference on Computer-Aided Design, Nov 2000.

Authors Profile:

R.Srivasavi

Ms. R.Srivasavi is pursuing masters degree in VLSI system design. She has keen interest in microcontrollers and Digital Electronics.

A.Srinivasa Rao
M.Tech

Mr. A .Srinivasa Rao is an Assistant professor in electronics and communication engineering in Nimra College of Engg & Tech. He got his M.tech from K.L university

P.Srinivas
M.Tech

Mr. P.Srinivas Is A young and dynamic Associate Professor of electronics and communication Engineering.. He Got his M.Tech From Acharya Nagarjuna University. He Worked in various reputed Engineering Colleges as an Assoc Professor and Head Of The Department.

◈◈◈