Enhancing Delay Fault Testability for FIR Filters Based on Realistic Sequential Cell Fault Model

Shyue-Kung Lu and Mau-Jung Lu
Department of Electronic Engineering
Fu-Jen Catholic University, Taipei, Taiwan, R.O.C.

Abstract

In this paper, *C*-testability conditions based on *Realistic Sequential Cell Fault Model* (RS-CFM) [3] are proposed and applied to the block FIR filters. Based on RS-CFM, an ILA is said to be *C-testable* for cell delay faults if it is possible to apply all *SIC (single input change)* pairs [2] of a cell to each cell of the array in such a way that the number of test pairs for the array is a constant. A novel design-for-testability technique based on the functional bijectivity property [1] is used to make the FIR array C-testable for delay faults. C-testability conditions guarantee 100% single-cell-fault and cell-delay-fault testability with a constant number of test patterns. The hardware overhead is about 5.66% to make it C-testable for cell delay faults. The number of test patterns is 80 regardless of the word length, filter order, and the block size. It is only 31% of that for pseudoexhaustive testing of the FIR filters. The reduction of test generation complexity is significant. Since the derived 80 two-pattern tests include the exhaustive test patterns for functional testing of the ILA, combinational faults can also be detected.

I Introduction

Finite Impulse Filters (FIRs) are widely used in various applications, such as linear filtering, correlation analysis, and spectrum analysis. FIR filters are generally characterized by their linear phase, low output noise, and more stable due to the lack of any feedback. The basic FIR filter is characterized by the following equation:

$$y(n) = \sum_{k=0}^{N-1} h(k)x(n-k),$$

where $h(k)$, $k = 0, 1, 2, \ldots, N-1$ are the impulse response coefficients of the filter, and N is the filter length which represents the number of filter coefficients. This equation is a time domain equation and describes FIR filter in a non-recursive form: the current output sample, $y(n)$, is a function of only the past and present values of the inputs, $x(n)$.

A practical FIR chip can contain millions of gates. Owing to the low controllability and observability of the manufactured chips, efficient design-for-testability (DFT) and BIST techniques are required in order to guarantee high-quality products. However, new defect mechanisms exist in the fabricated chips by using today's advanced VLSI technology. Therefore, the traditional single cell fault models are not sufficient. In this paper, *C*-testability conditions based on *Realistic Sequential Cell Fault Model* (RS-CFM) are proposed and applied to the block FIR filters. Based on RS-CFM, an ILA is said to be *C-testable* for cell delay faults if it is possible to apply all *SIC (single input change)* pairs of a cell to each cell of the array in such a way that the number of test pairs for the array is a constant. It has been proven that *SIC* pairs are more effective than classical *multiple input change (MIC)* test sequences when high robust delay fault coverage is targeted. A novel design-for-testability technique based on the functional bijectivity property is used to make the FIR array C-testable for delay faults. C-testability conditions guarantee 100% single-cell-fault and cell-delay-fault testability with a constant number of test patterns. The hardware overhead is about 5.66% to make it C-testable for cell delay faults. The number of test patterns is 80 regardless of the word length, filter order, and the block size. It is only 31% of that for pseudoexhaustive testing of the FIR filters. The reduction of test generation complexity is significant. Since the derived 80 two-pattern tests include the exhaustive test patterns for functional testing of the ILA, combinational faults can also be detected. According to the testability conditions, the BIST structure for the FIR filter is also easy to implement. The test pattern generator (TPG) is simply a binary counter and a shift register. Moreover, the output response analyzer can be implemented with comparators.

II Definitions and Fault Model

We follow the definitions used in [1]. We assume a *cell* in an ILA with function f is a combinational machine (Σ, Δ, f), where $f: \Sigma \rightarrow \Delta$ is the cell function, and $\Sigma = \Delta = \{0, 1\}^w$, w denotes the word length of a cell. An ILA is an array of cells. Let $\Sigma = \Sigma^x \times \Sigma^y$, where Σ^x is the horizontal input set and Σ^y is the vertical input set of a cell. Similarly, $\Delta = \Delta^x \times \Delta^y$, where Δ^x is the horizontal output set and Δ^y is the vertical output set of a cell

Definition 1: The *x-projection* of f, $f^x: \Sigma^x \times \Sigma^y \rightarrow \Delta^x$, is defined to be $f^x(i, j) = k$, where $f(i, j) = (k, l)$. The *y-projection* is defined analogously. We say f is *x-injective* when $\forall j, \forall i_1 \neq i_2, f(i_1, j) \neq f(i_2, j)$; we say f^x: is *x-bijective* if it is *x-injective* and $\Sigma^x = \Delta^x$

Definition 2: An *x-fault* exists when $\exists (i, j) \in \Sigma$ such that

Proceedings of the Second IEEE International Workshop on Electronic Design, Test and Applications (DELTA'04)
0-7695-2081-2/04 $ 20.00 © 2004 IEEE

IEEE COMPUTER SOCIETY

$f^x(i, j) = k/l$ where k is the intended horizontal output but $l \neq k$ is the actual horizontal output. A *y-fault* is defined analogously.

Definition 3: The *Single Input Change (SIC)* pair is a pair of vectors $<v_1, v_2>$, where the Hamming distance between v_1 and v_2 is 1, $v_1, v_2 \in \{t_1, t_2, t_3, ... t_n\}$, $n = 2^w$.

Definition 4: We say that the function f of a cell is *bijective* when $\forall\theta_1 \neq \theta_2, f(\theta_1) \neq f(\theta_2), \theta 1, \theta_2 \in \Sigma$, and the length of the minimal complete input sequence and that of the minimal complete output sequence are identical, i.e., $|\Sigma| = |\Delta|$.

Definition 5: A *complete SIC sequence* (SIC_{com}) is a two-pattern sequence that contains exhaustive SIC pairs of a cell. A *minimal complete SIC sequence* (SIC_{min}) is a two-pattern sequence that contains exhaustive SIC pairs of a cell and its length is minimal, i.e., $| SIC_{min}| = w2^w$.

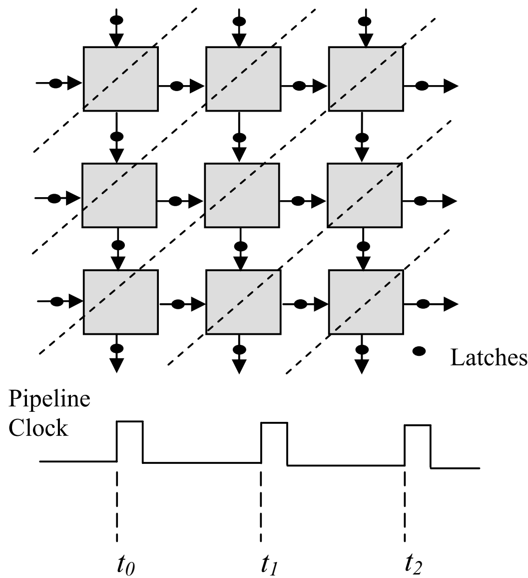

Fig. 1: Hardware model of ILAs.

A *cell delay fault* occurs if and only if an input transition event of an arbitrary cell in the array through a path in the cell to its output cannot be propagated in desired clock period. A delay test is a vector pair $<v_1, v_2>$. In Fig. 1, the black dots represent the pipeline latches. The vector pair $<v_1, v_2>$ is applied to the cell under test of the mesh-connected ILA at time steps t_0 and t_1, and the output is sampled at time step t_2. If the signals cannot propagate through the combinational cell in specified clock period, then the sampled output values of the cell may be incorrect, and a cell delay fault is said to occur. Note that after v_1 is applied, it is necessary to allow the circuit to stabilize before applying v_2. The number of exhaustive two-pattern tests for a basic cell is 2^{2w}.

It is shown that SIC pairs are sufficient to detect all robustly detectable path delay faults [2]. According to the adopted of RS-CFM , we examine the application of SIC_{com} to every cell of the ILA. According to the proposed conditions [4] , SIC_{com} of a cell can be applied to each cell in the array. In other words, we show that a constant (independent of the array size) number of two-pattern tests is sufficient to test cell delay faults in the ILA if the cell function meets the testability conditions. By using SIC pairs under RS-CFM, the complexity of test generation is reduced significantly.

The cell behavior of a pipelined ILA can be modeled as a *finite state machine (FSM)*. The transition graph $G_t(V, E)$ of the cell function f is a directed graph, where $V = \Sigma$ and $E = \{(v, f(v))| \ v \in V\}$. Since each delay fault requires a two-pattern test, therefore, we should define the transition graph of f with respect to a 2-pattern sequence. This transition graph is denoted as

$$G_t^2 = (V_t^2, E_t^2), \tag{1}$$

where

$$\begin{aligned} V_t^2 &= \{< v_1, v_2 >| \ v_1, v_2 \in \Sigma\}, \\ E_t^2 &= \{(v, f(v)) \mid v \in V_t^2\}. \end{aligned} \tag{2}$$

It is clear that there are 2^{2w} possible two-pattern tests for a single cell, i.e., the complete two-pattern test sequence is $<t_1, t_2, ...t_n, n = 2^{2w}>$. In other words, G_t^2 contains 2^{2w} vertices. If $w = 8$, then 64K tests are required. If $w = 16$, the number of tests required is 4G. This is impossible for practical applications. Therefore, extracting some *efficient* tests from the 2^{2w} tests is an important problem. Fortunately, it is proven that exhaustive SIC pairs of a cell are sufficient to detect all robustly detectable path delay faults of a cell [4]. This will reduce the number of tests significantly. For example, the number of exhaustive SIC pairs is only about 3% of that of the exhaustive two-pattern tests when $w = 8$. However, even the number of exhaustive SIC pairs is greatly less than the number of exhaustive two-pattern tests. How can we send them to each cell in the array and propagate faulty effects are still problems. Therefore, we try to find sufficient conditions which can send all SIC pairs to each cell in the FIR array and propagate faulty effects to the primary outputs.

Theorem: A mesh-connected ILA as shown in Fig. 1 is C-testable under RS-CFM if each of the 2^{2w} vertices in G_t^2 contained in components [4].

III Testable Design of FIR Filters for Delay Faults

A word-level ILAs for a 4-tap FIR filter (i.e., $y_n = a_0x_n + a_1x_{n-1} + a_2x_{n-2} + a_3x_{n-3}$) with a block size of 4 is shown in Fig. 2, where each cell represents an inner-product processor. The bit-level architecture of an inner-product processor for FIR filters are essentially a 2-D mesh-connected array of multiplier cells with only unilateral connections. The tap values for the FIR filter are often fixed. Therefore, it is not necessary to change the values of the registers frequently. The structure of a 4

Proceedings of the Second IEEE International Workshop on Electronic Design, Test and Applications (DELTA'04)
0-7695-2081-2/04 $ 20.00 © 2004 IEEE
IEEE COMPUTER SOCIETY

× 3 array multiplier and its cell structure are shown in Fig. 3 and Fig. 4 respectively.

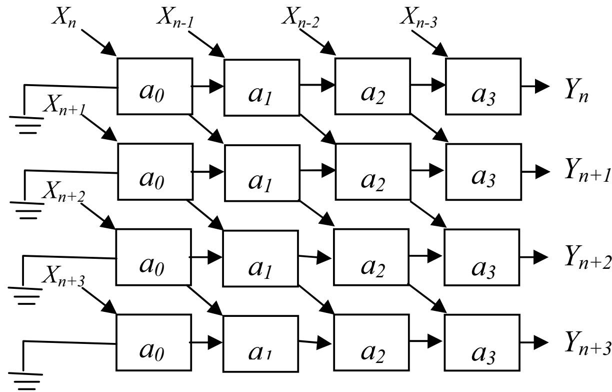

Fig. 2: A 4-tap FIR filter with a block size of 4.

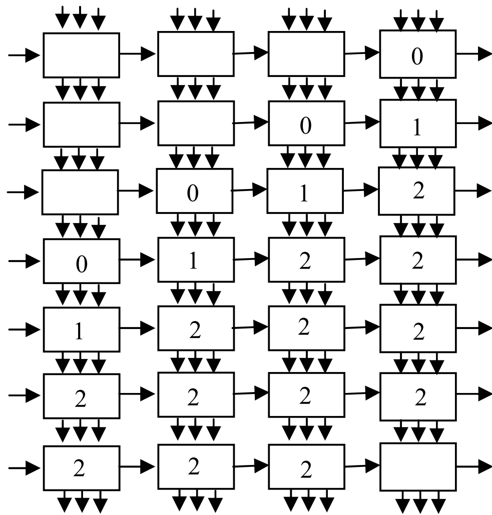

Fig. 3: A 4 × 3 array multiplier.

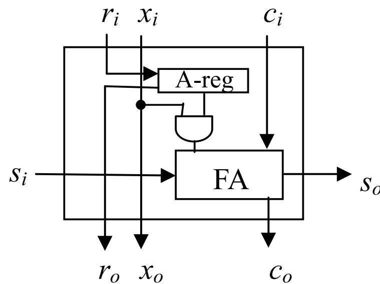

Fig. 4: The multiplier cell.

We modify the array multiplier and make its cell function bijective. According to the DFT techniques in [1], the modified cell funcions for A = 0 and A = 1 are shown in Table 2(a) and Table 2(b), respectively. The structure of the modified is shown in Fig. 5 where a multiplexder (*M*) is added. When the *test* signal = 0, it performs the normal function. When test = 1, the cell function is made bijective and Thm. 1 can be applied.

IV Conclusion

In this paper, a design-for-testability approach is used to make the FIR filter arrays *C-testable* for cell delay faults

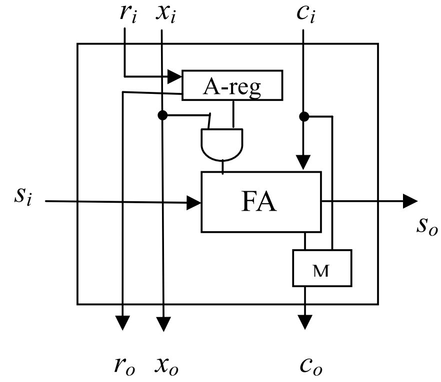

Fig. 5: The modified multiplier cell.

Table 1: Truth tables of the modified multiplier cell (a) A=0 (b) A=1.

x_i	s_i	c_i	x_i	s_i	c_i
0	0	**0**	0	0	**0**
0	0	**1**	0	1	**1**
0	1	**0**	0	1	**0**
0	1	**1**	0	0	**1**
1	0	**0**	1	0	**0**
1	0	**1**	1	1	**1**
1	1	**0**	1	1	**0**
1	1	**1**	1	0	**1**

(a)

x_i	s_i	c_i	x_i	s_i	c_i
0	0	**0**	0	0	**0**
0	0	**1**	0	1	**1**
0	1	**0**	0	1	**0**
0	1	**1**	0	0	**1**
1	0	**0**	1	1	**0**
1	0	**1**	1	0	**1**
1	1	**0**	1	0	**0**
1	1	**1**	1	1	**1**

(b)

based on the proposed testability conditions. The hardware overhead is about 5.66% to make it C-testable for cell delay faults. The number of test patterns is 80 regardless of the word length, filter order, and the block size. It is only 31% of that for pseudoexhaustive testing of the FIR filters. The reduction of test generation complexity is significant. Since the derived 80 two-pattern tests include the exhaustive test patterns for functional testing of the ILA, combinational faults can also be detected.

Reference

[1] C. W. Wu and P. R. Cappello, "Easily Testable Iterative Logic Arrays," *IEEE Trans. Comput.,* vol. 39*,* pp. 64-652, May 1990.

[2] R. David, P. Girard, C. Landrault, S. Pravossoudovitch and A. Virazel, "On Hardware Generation of Random Single Input Change Test Sequences," *Proceedings 6th IEEE European Test Workshop,* pp. 117-123, 2001.

[3] M. Psarakis, D. Gizopoulos, A. Paschalis and Y. Zorian, "Sequential fault modeling and test pattern generation for CMOS iterative logic arrays," *IEEE Trans. Computers*, vol. 49, no. 10, pp. 1083-1099, Oct. 2000.

[4] S. K. Lu, "Delay Fault Testing for CMOS Iterative Logic Arrays with a Constant Number of Patterns," *IEICE Transactions on Information and Systems,* vol. E86-D, no. 12, pp. 1-7, Dec. 2003.

Proceedings of the Second IEEE International Workshop on Electronic Design, Test and Applications (DELTA'04)
0-7695-2081-2/04 $ 20.00 © 2004 IEEE

IEEE COMPUTER SOCIETY