Low Cost Concurrent Test Implementation for Linear Digital Systems [*]

Ismet Bayraktaroglu
Computer Science & Engineering Department
University of California, San Diego
La Jolla, CA 92093
ibayrakt@cs.ucsd.edu

Alex Orailoglu
Computer Science & Engineering Department
University of California, San Diego
La Jolla, CA 92093
alex@cs.ucsd.edu

Abstract

An implementation of a low-cost, time-extended invariant-based concurrent test scheme for linear digital systems is presented. Both feedback and non-feedback systems are analyzed to identify gate and RT level implementation requirements for high on-line fault coverage. Simulation results on implementations satisfying the outlined requirements indicate that low latency, 100% on-line fault coverage is attained within hardware costs comparable to those of scan insertion.

1 Introduction

VLSI technology has evolved to a level where large systems, once implemented as printed circuit boards with discrete components, can be integrated into a single IC. As industry continues to push the limits of VLSI technology, reliability of such systems becomes the primary concern. Manufacturing test provides a relatively easy way of eliminating defective ICs. However, it does not completely eliminate reliability concerns since the probability of failure during normal functionality increases with reduced sizes.

Linear digital systems have been utilized widely in signal processing during the last two decades. Signal processing is employed in many mission critical applications, such as military, satellite, and medical. Consumer applications, on the other hand, are not as critical. While a one second delay in error detection may even result in loss of life in a hospital environment, such a delay in a multimedia application may be tolerable, or even desirable, especially if introducing latency sharply reduces cost.

Various techniques have been developed for concurrent testing of digital systems. The embedded simple invariants of a DSP application, typically capable of exercising large segments of the implementation, make them ideal candidates for algorithm-based error detection approaches. In [1], Reddy and Banerjee propose an algorithm-based error detection scheme for FFT and QR factorization, in which an invariant property of such algorithms is employed. In [2], Huang and Abraham present an algorithm-based fault detection scheme for matrix operations. As linear systems can be represented as simple matrix operations, techniques developed for matrix operations can be easily applied to them. In [3], Chatterjee discusses an error detection scheme for linear analog systems that utilizes error detection of matrix operations. In [4], Chatterjee and Roy present a technique that extends algorithm-based approaches to non-linear systems; the technique suggested models the non-linear operators as linear time varying operators.

All of the previously proposed schemes utilize an invariant system property so as to detect faults on-line. While 100% fault detection for single *unit* faults may be attainable with such schemes, ensuring their practicality requires significant reduction in area overhead and increased attention to the effect of numerical inaccuracies on faults with low magnitude effects. We investigate in this paper concurrent error detection techniques that are low-cost and provide high fault coverage; significant reductions in area are attained by introducing a short error detection latency. Furthermore, as the proposed technique is based on observation of the average, rather than the instantaneous, behavior of circuits, it inherently tolerates numerical inaccuracies as long as the round-off scheme introduces no extraneous bias.

An invariant based-concurrent error detection scheme for acyclic linear digital systems is presented in [5]. We extend the analysis to general *cyclic* systems and outline associated implementation requirements in order to tolerate numerical inaccuracies and attain high fault coverage. Simulation results for the systems implemented by fulfilling the outlined requirements indicate that high concurrent fault coverage for both acyclic and cyclic systems is achievable within small area overhead.

Linear digital systems are briefly reviewed in section 2. While section 3 outlines the proposed concurrent error detection scheme, section 4 describes details of the implementation of the concurrent error detection hardware. Simulation results in section 5 are followed by conclusions in section 6.

[*] This work is being supported by Hughes Space and Communications and the UC Micro Program.

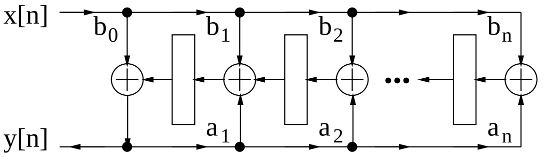


Figure 1: Transposed direct form implementation

2 Linear Digital Circuits

Linear digital systems can be represented as a difference equation of the following form.

$$y[n] - \sum_{k=1}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \qquad (1)$$

If a system has feedback (i.e., $a_k \neq 0$), its impulse response becomes infinite; such systems are called *infinite impulse response* (IIR). Systems with no feedback are denoted *finite impulse response* (FIR) systems. There are various implementation styles of linear digital systems. In this work, we utilize the transposed direct form [6] implementation as this implementation style exhibits high performance and low area; a canonical representation of the transposed direct form implementation is depicted in figure 1.

3 Error Detection Mechanism

We propose a concurrent error detection scheme that relies on observations of the long term average behavior of a system. Deviations from the average behavior can be detected through utilization of an invariant of the system. The invariant has to be violated by the average behavior of the faults in the system. Implementation considerations necessitate that evaluation of the invariant must be computationally simple in order to produce a low overhead detection scheme. An additional desirable consideration for latency-based schemes may be to ensure that the speed of detection is directly correlated with the magnitude of the fault; large scale system deviations in case of a fault manifestation can then be prevented in a timely manner.

The average behavior of a system can be estimated by observing inputs and outputs; as the number of observed input patterns increases, estimation becomes more accurate. Estimation accuracy directly corresponds to the magnitude of the errors that can be detected. Therefore, as the number of observed patterns increases, errors with smaller magnitude effects get detected as well.

A fault can be detected if the effect of the fault can be accumulated to violate the invariant. While complex accumulation techniques that do not rely on monotonicity of fault effects for noncancelling accumulation may be theoretically plausible, simplicity of hardware implementation suggests examination of techniques wherein the effect of the fault on invariant calculation is always in the same direction. A fault in a linear system can be modeled as an external input at the fault site. If the transfer function of the system from the fault site to the output of the invariant evaluator is monotonic, the effect of the fault on the invariant will always be in the same direction. A fault with such a behavior is defined to be a *monotonic* fault. If all the faults in a system behave monotonically with respect to an invariant, 100% concurrent fault detection can be achieved.

A fault behaves monotonically with respect to the invariant if all of the following conditions are satisfied.

- The invariant calculation is monotonic.
- The transfer function of the system is monotonic from the RT component embedding the fault to the output.
- The effect of the fault on the output of the RT component is monotonic.

Even satisfaction of all the conditions above does not guarantee full detection, if fault effects extend across clock cycles, as the summation of the fault effects in the temporal domain may cancel. In the case of an FIR system, faults affect the output for one clock cycle only unless they are located at the primary inputs. The above monotonicity conditions are consequently sufficient in order to achieve complete fault coverage for FIR systems but may need to be augmented in the case of IIRs.

We continue this paper in the next section by presenting an invariant that satisfies the first condition. The second condition is fulfilled inherently for linear systems as linearity is a subset of monotonicity. The third condition, on the other hand, requires special attention. The gate level implementation style dictates the effect of the faults on the RT level component. Therefore, we additionally provide a short overview of gate level implementation styles for components of linear digital systems in the next section.

4 Concurrent Test for Linear Systems

A well-known invariant property of linear systems is their DC behavior. Evaluation of the DC gain can be performed off-line from the coefficients and on-line from the inputs and outputs. The relation between the DC gain computed on-line and off-line for a set of P+1 patterns is given by

$$\left(1 - \sum_{k=1}^{N} a_k\right) \sum_{n=0}^{P} y[n] \quad = \quad \sum_{k=0}^{M} b_k \sum_{n=0}^{P} x[n] + \mathcal{T} \qquad (2)$$

where an upper bound on the tolerance, $\mathcal{T}$, is given by

$$\mathcal{T}_{max} = 2x_{max} \sum_{n=1}^{M} \left| \sum_{k=n}^{M} b_k \right| + 2y_{max} \sum_{n=1}^{N} \left| \sum_{k=n}^{N} a_k \right| \qquad (3)$$

Equation 2 provides a simple relation between the accumulated input and output of a linear digital system within a small tolerance range. Furthermore, the unidirectional accumulation property of the invariant calculation in equation 2 satisfies the first monotonicity condition outlined in the previous section. In equation 3, x_{max} and y_{max} denote the maximum input and output signal magnitude, respectively. While equation 3 provides an upper bound on $\mathcal{T}$ for both FIR and IIR systems, it is not tight for IIR systems since during determination of the upper bound, the output is assumed to vary

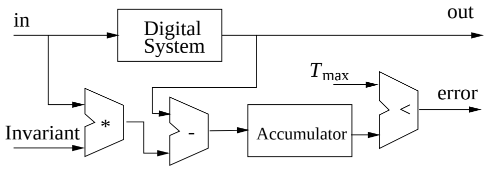
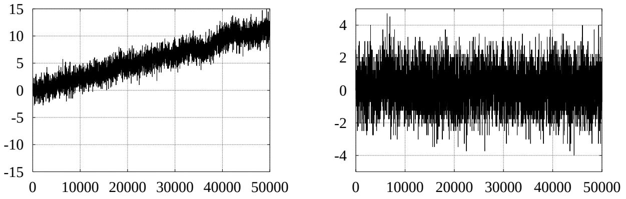


Figure 2: Concurrent test implementation for linear systems



Truncation Round-to-nearest-odd

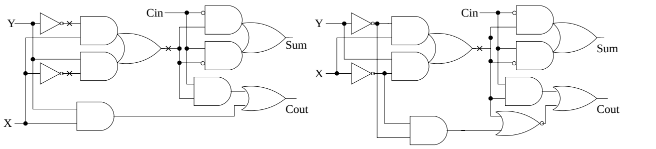
Figure 3: Round-off behavior

independently. However, the output depends on the inputs and cannot be arbitrarily chosen in order to maximize $\mathcal{T}$. A tight upper bound in the case of IIR systems, within a very small error, can be found by approximating IIR systems to FIR systems. Such an approximation is performed by truncating the impulse response of IIR systems at a point where the response falls below the smallest number representable by the fixed-point implementation. In the following equation, b'_k denotes the truncated response of an IIR system.

$$\mathcal{T}_{max} = 2x_{max} \sum_{n=1}^{M} \left| \sum_{k=n}^{M} b'_k \right| \tag{4}$$

Figure 2 depicts the concurrent error detection hardware utilized in this work. As the on-line detection capability of the invariant depends on input/output accumulation, bias introduced by numerical inaccuracies may produce false alarms. In our implementation, we utilize a round-to-nearest-odd scheme [7] to preclude occurrence of undesired bias. Simulation results, shown in figure 3 for an FIR implementation, indicate that the round-to-nearest-odd scheme introduces no bias; no false alarms are consequently encountered.

In the case of an IIR system, the monotonicity requirement has to be revisited further. Assume that a fault at time t always increases the output by x, $-2x$, and x at times t, $t+1$, and $t+2$, respectively. Even though the fault effects within each time step are monotonic, the temporal accumulation of such fault effects results in fault masking. An additional condition on the feedback coefficients, a_k, needs to be imposed to preclude such fault masking. The constraint can be mathematically reduced to $1/(1 - \sum a_k) \neq 0$, trivially satisfied as its violation would necessitate at least one of the a_k coefficients to be infinite. Nonetheless, further examination of the issue may be necessary as fault detection latency is inversely correlated to the distance of the fractional expression from zero, i.e., its magnitude. A linear system is stable though as long as the poles of the system are inside the unit circle [6], which guarantees that $|1 - \sum_{k=1}^{N} a_k| < 1$. Therefore, for practical, i.e. stable, linear system implementations, the monotonicity condition ceases to pose any additional restrictions on the implementation of IIR systems.



a) AND/OR/NOT implementation b) Improved implementation

Figure 4: Full adder implementations

Linear digital systems are implemented at the RT level as a set of multiplication, addition, and delay operations. Since the multipliers in such systems are constant multipliers, they are implemented using a series of shift and add operations. Delay operations are implemented as registers and monotonicity of the faults internal to registers can easily be shown. Therefore, in this work, we only investigate monotonicity of adder implementations.

In this paper, we only provide a summary of the analysis for ripple-carry adder implementations. Monotonic implementation of other adders can be found in [8]. Ripple-carry adders are composed of a chain of full adders. It can be seen by examining the functionality of a full adder that faults at the inputs (x, y, and c_i) and at the outputs (c_o and sum) behave monotonically. Faults internal to full adder cells, on the other hand, may behave non-monotonically depending on the implementation style.

A full adder implementation with AND/OR/NOT gates, shown in figure 4a, possesses 3 faults (indicated with an 'x' on the line in figure 4) with non-monotonic behavior. Modification of the logic, as shown in figure 4b, moves two of these faults to the monotonic fault list. Such modifications reduce the percentage of non-monotonic faults to less than 2.5%.

Searches for adders with stronger monotonicity characteristics and an existence proof of monotonic implementations showing that all 2-level implementations of monotonic functions have solely monotonic faults are shown in [8]. Since adders are linear, the results of [8] guarantee that cost-effective ripple-carry adder implementations with monotonic fault behavior indeed exist.

5 Results

Two 13-tap FIR filters and one 5-tap IIR filter, together with the associated on-line checking hardware, have been implemented using ripple carry adders, composed of 2-level full-adder cells. While filters are utilized in this work, the proposed scheme makes no assumptions about the nature of linear systems and therefore can be exploited in any linear system. A fractional number system of the form 1.9 (1 for the sign and 9 bits for the fraction) was selected for the input of the first FIR and the IIR filters. For the output of these circuits, a 1.13 form was utilized. In the case of the second FIR filter, the input and the output were selected to be of the form 1.7 and 1.11, respectively.

The hardware cost of on-line checking hardware is independent of the number of taps in the circuit; it depends strictly on the hardware requirements for the multiplier used to multiply

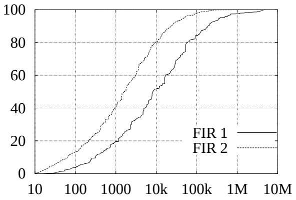


Figure 5: Fault simulation results for two FIR systems

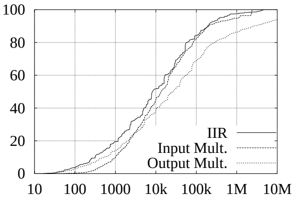


Figure 6: Fault simulation results for an IIR system

	Circuit		Test Overhead			
	Gates	FFs	Gates	FFs	%	$\mathcal{T}_{max}$
FIR 1	4394	169	1096	24	23.5	6.15
FIR 2	3014	139	411	19	13.6	0.56
IIR	4434	67	876	24	20.7	1.90

Table 1: Gate counts for system and error detection hardware

the input with the invariant. All circuits have been synthesized from a data flow graph description and optimized at the gate level by a set of tools developed in C during this work. Fault simulations have been performed by HOPE [9]. To calculate aggregate overhead percentiles, each flip-flop, denoted as FF in the table, has been assumed to be equivalent to 4 gates.

As can be seen in figure 5, 100% fault coverage is invariably achieved for FIR circuits, with an area cost that ranges for small circuits between an eighth to a quarter of the basic design. The results further indicate that the faults even inside the rounding logic are detected by this monotonic accumulation scheme. As the area cost of the proposed concurrent test is a constant function of design size, typical circuit sizes of 64 taps and 25,000 gate equivalents show approximate area costs of less than 5%.

Figure 6 indicates that near 100% coverage is attained for IIR circuits within 10 million patterns. Investigation of fault detection loss indicates that all undetected faults reside in the multipliers that multiply the output by the a_k coefficients. Loss of fault detection in this case is caused by the reduced activation of faults, as the output of the circuits is of reduced randomness compared to the inputs. As the number of activations is reduced, accumulation of fault effects so as to violate the invariant takes longer.

An analysis of the average power of the fault effects indicates that as the number of patterns increases, the signal to noise ratio (SNR) of the undetected faults increases, as shown in figure 7. It is apparent from the figure that faults with large magnitude and higher activation rates are detected earlier. An SNR comparable to the SNR of numerical inaccuracies is achieved at around 100,000 patterns for this 13-tap FIR filter.

6 Conclusion

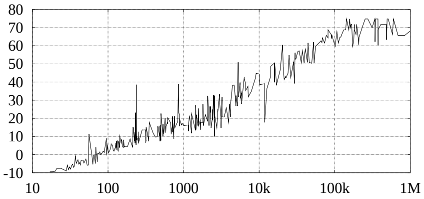


Figure 7: SNR caused by the detected faults in FIR 1

A low cost concurrent test scheme for a general class of linear digital systems has been developed. Low cost is achieved by observing the average behavior of the system, rather than the instantaneous behavior. Utilization of the average behavior of the system, though it introduces latency to fault detection, helps tolerate adverse effects of numerical inaccuracies.

In this work the conditions required for preventing fault masking, the only possible cause for fault coverage loss, are identified for both acyclic and cyclic systems. Furthermore, the effect of numerical inaccuracies on the proposed invariant scheme is investigated in order to prevent false alarms. Both acyclic and cyclic systems that satisfy the outlined design constraints are synthesized. Fault simulations performed verify that high fault coverage within a short latency is achievable for both cyclic and acyclic systems at low hardware overhead.

The proposed scheme, therefore, provides a concurrent test solution for applications that require low-cost reliability solutions. The twin advantages of low-cost and complete fault security of the proposed system are expected to enable its exploitation by a wide variety of consumer applications, thus expanding greatly the set of reliable electronic applications.

References

[1] A. L. Narasimha Reddy and P. Banerjee, "Algorithm-based Fault Detection for Signal Processing Applications", *IEEE Trans. on Computers*, vol. 39, n. 10, pp. 1304–1308, October 1990.

[2] K. H. Huang and J. A. Abraham, "Algorithm-based fault tolerance for matrix operations", *IEEE Trans. on Computers*, vol. 33, n. 6, pp. 518–522, June 1984.

[3] A. Chatterjee, "Concurrent Error Detection and Fault-Tolerance in Linear Analog Circuits Using Continuous Checksums", *IEEE Trans. on Very Large Scale Integration*, vol. 1, n. 2, pp. 138–150, June 1993.

[4] A. Chatterjee and R. K. Roy, "Concurrent Error Detection in Nonlinear Digital Circuits with Applications to Adaptive Filters", in *IEEE International Conference on Computer Design*, pp. 606–609, 1993.

[5] I. Bayraktaroglu and A. Orailoglu, "Low-cost on-line test for digital filters", in *IEEE VTS*, pp. 446–451, 1999.

[6] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing,* Prentice Hall, 1989.

[7] M. R. Santoro, G. Bewick and M. A. Horowitz, "Rounding algorithms for IEEE multipliers", in *IEEE Symposium on Computer Arithmetic*, pp. 176–183, 1989.

[8] I. Bayraktaroglu and A. Orailoglu, "Unifying Methodologies for High Fault Coverage Concurrent and Off-line Test", in *ISCAS'2000*, May 2000.

[9] H. K. Lee and D. S. Ha, "HOPE: an efficient parallel fault simulator", in *IEEE Design Automation Conference*, pp. 336–340, 1992.