Neapolis University

HEPHAESTUS Repository **http://hephaestus.nup.ac.cy**

School of Information Sciences Articles

1996

Linear-testable and C-testable Nx~x Ny~ modified Booth multipliers

Gizopoulos, Dimitris

Institution of Engineering and Technology (IET)

http://hdl.handle.net/11728/6487

Downloaded from HEPHAESTUS Repository, Neapolis University institutional repository

Linear-testable and C-testable $N_x \times N_y$ modified Booth multipliers

D. Gizopoulos
A. Paschalis
D. Nikolos
C. Halatsis

Indexing terms: VLSI testing, Design for testability, Modified Booth multipliers

Abstract: The testability of modified Booth multipliers is examined with respect to the cell fault model, an implementation–independent fault model. This is especially useful in design environments where the cell realisations are unknown. First, a linear-testable multiplier is proposed which can be tested with $2N_x + \lceil Ny/2 \rceil + 40$ test vectors and requires only one extra primary input. Zero delay overhead and negligible hardware overhead is imposed. Then, a *C*-testable multiplier is proposed which can be tested with 70 test vectors and requires only two extra primary inputs. Both the hardware and delay overheads are very small and decrease rapidly with increasing N_x or N_y. The linear-testable design is superior to the *C*-testable one, for small multipliers since it requires less test vectors in addition to the much smaller hardware overhead and zero delay overhead it imposes. For larger multipliers there is a trade-off between the number of test vectors, which is smaller for the *C*-testable design, and the overheads, which are smaller for the linear-testable design.

1 Introduction

The increasing complexity of VLSI circuits makes their testing a difficult problem. To derive small and efficient test sets we have to exploit the special features of the specific VLSI circuit under test. Much work has been done or the derivation of a test set of array structures (iterative logic arrays, ILAs) implemented on a VLSI chip [1–4]. The fault model which has been extensively used is termed the cell fault model (CFM) [1] and assumes that only one cell can be faulty at a time and that any fault inside the faulty cell can occur as long as the faulty cell remains combinational. Such a fault model is independent of specific implementation. An ILA is *C*-testable if it can be tested with a constant number of test vectors irrespectively of its size [2].

Parallel multipliers with array structure belong to the class of ILAs and have the potential of being easily testable. The design of easily testable multipliers has been discussed in [5].

A multiplier based on the modified Booth algorithm [6], decreases the number of rows that are added together by one half, resulting in significantly faster and much more efficient implementations than standard array multipliers [6]. Hence, this kind of multiplier is widely used. The testability of multipliers based on the modified Booth algorithm, when the structure of most or all of its cells is known (and thus the solution is implementation dependent), was examined in [7–9]. In [7] the testability of a multiplier-accumulator based on the modified Booth algorithm and designed using the Piramid silicon compiler [10] was examined. The fault model used in [7] is an extension of the stuck-at fault model towards switch level faults in pass transistor logic for all the cells of the multiplier, except the full adder where, the CFM was used. A test set for the multiplier consisting of $2\lceil N_y/2 \rceil + 36$ test vectors was given, with the addition of two extra primary inputs and the modification of the half adders of the multiplier array into full adders causing extra hardware and delay overheads. The multiplier was also modified to be *C*-testable with 18 test vectors. Extra hardware is required for the *C*-testable design and three extra primary inputs are required. Hardware and delay overheads were not explicitly calculated. In [8] the specific cell implementations of another silicon compiler, namely Cathedral [11], were used. The fault model used in [8] includes node stuck-at, transistor stuck-open and transistor stuck-close faults for the specific cell implementations, which include symmetric and non-symmetric CMOS gates. Three extra primary inputs are needed to make the multiplier *C*-testable with a test set of 50 test patterns and with hardware and delay overheads, not explicitly calculated. In [9] a *C*-testable modified Booth multiplier design, implemented using DCVS logic, was given. For the specific DCVS implementations of the multiplier cells, the multiplier is *C*-testable with 31 test patterns, detecting all detectable single stuck-at, stuck-on and stuck-open faults. The addition of six primary inputs, two primary outputs and some logic is required for the design to be *C*-testable. In this work also the hardware and delay overheads were not explicitly calculated.

© IEE, 1996

IEE Proceedings online no. 19960008

Paper first received 15th March 1995 and in revised form 28th September 1995

D. Gizopoulos and A. Paschalis are with the Institute of Informatics & Telecom., NCSR "Demokritos", 153 10 Ag. Paraskevi, Athens, Greece

D. Nikolos is with the Department of Computer Engineering and Informatics, University of Patras, 265 00, Patra, Greece

C. Halatsis is with the Department of Informatics, University of Athens, TYPA Buildings, 157 71, Athens, Greece