

Methoden der Testvorbereitung

Hans-Joachim Wunderlich

Institut für
Rechnerentwurf und Fehlertoleranz
Universität Karlsruhe

Michael Schulz

Lehrstuhl für
Rechnergestütztes Entwerfen
Technische Universität München

Neben der eigentlichen Testdurchführung umfaßt eine Teststrategie die Auswahl eines geeigneten Fehlermodells, ein Verfahren für den prüfgerechten strukturierten Entwurf und die Testsatzerzeugung. Ziel dieser Prüfvorbereitung ist die Steigerung der Produktqualität und die Senkung der Kosten für die Testdurchführung.

1. Testkosten

Der Aufwand zur Testerzeugung und Testdurchführung wächst überproportional mit der Zahl der in einem Chip integrierten Funktionen, so daß die Testkosten für anwendungsspezifische Schaltungen (ASICs) bereits heute zwischen 60% und 70% der Gesamtkosten ausmachen können [2].

Der unverhältnismäßig große Kostenanstieg für den Test hat mehrere Ursachen. Die *Zugänglichkeit* der in einer Schaltung realisierten Moduln verschlechtert sich, da bei steigender Integrationsdichte die Zahl der außen verfügbaren Anschlüsse nicht in dem Maße vergrößert werden kann wie die Zahl der realisierten Transistoren. Zur *Erzeugung eines Testprogramms* müssen Probleme von hoher Komplexität gelöst werden, deren Aufwand im schlimmsten Fall exponentiell und in der Praxis quadratisch bis kubisch mit der Schaltungsgröße wächst. Die *Zahl der Testmuster* steigt in der Regel linear an, da die Muster jedoch wegen der erwähnten geringeren Zugänglichkeit zumeist seriell in die Schaltung eingegeben werden, kann die *Testdurchführungszeit* quadratisch zunehmen. Die umfangreichen Prüfdaten müssen in großer Geschwindigkeit der Schaltung zugeführt und ausgewertet werden, da es zahlreiche Defekte gibt, die nicht die logische Funktion, aber die Schaltzeit beeinträchtigen. Insgesamt müssen die *Prüfgeräte* mindestens so leistungsfähig wie die zu testenden Schaltungen sein, so daß hierfür die neueste Technologie zu hohen Kosten verwendet werden muß.

Schließlich ist der überproportionale Anteil der Testkosten auch durch die Fortschritte beim automatischen Entwurf und bei der Herstellung bedingt. Mit zahlreichen neuen Verfahren des rechnergestützten Entwurfs konnte die zur Schaltungsentwicklung benötigte Zeit um mehrere Größenordnungen gesenkt werden. Standardzellensysteme stellen dem Entwerfer eine Bibliothek von Bauelementen zur Verfügung, mit denen er eine Netzliste zusammenstellen kann, die automatisch plaziert und verdrahtet wird. Sogenannte Silicon Compilers generieren sich diese Elemente aufgrund parametrisierter Angaben des Benutzers selbst. Der Einsatz solcher Werkzeuge kann den benötigten Entwurfsaufwand für eine integrierte Schaltung von mehreren Mannjahren auf wenige Mannmonate reduzieren. Eine wichtige Maßnahme zur Reduktion der Herstellungskosten ist der Einsatz bereits teilweise vorgefertigter Schaltungen wie "Gate Arrays" oder "Sea-of-Gates". Die Kostenvorteile dieser neuen Techniken können nur dann voll zur Geltung gebracht werden, wenn beim automatisierten Entwurf bereits der Test berücksichtigt wird.

2. Fehlermodellierung und Fehleranalyse

Ausfallursachen, Defektmechanismen, die zu erwartenden Fehlfunktionen und damit auch die Testerzeugung sind in hohem Maße technologieabhängig. Mit empirisch gewonnenen Daten und mit analytischen Verfahren wird aus dem Layout einer Schaltung hergeleitet, mit welchen Defekten in der Schaltungsstruktur zu rechnen ist, wie häufig sie auftreten werden und zu welchen Fehlfunktionen im Schaltungsverhalten sie führen können [4], [8]. Diese werden in ein Fehlermodell aufgenommen, für das ein Testsatz zu erstellen ist.

In der Vergangenheit wurden zahlreiche realistische Fehlermodelle entwickelt, die jeweils für bestimmte Entwurststile und Herstellungstechniken von Bedeutung sind. Neben dem klassischen, für bipolare Schaltungen adäquaten Haftfehlermodell werden in jüngster Zeit auch komplexere, insbesondere für MOS-Schaltungen typische Fehler wie Verzögerungs- und Übergangsfehler berücksichtigt. Da ein sehr komplexes Fehlverhalten die Testerzeugung erschwert, werden Entwurfsregeln untersucht, die dieses ausschließen oder zumindest unwahrscheinlich machen.

Bereits hier wird deutlich, daß während des Tests nicht die korrekte Schaltungsfunktion verifiziert werden kann, sondern daß nur nachgewiesen wird, daß keine der angenommenen Fehlfunktionen vorliegt. Aus der Häufigkeit der Störungen und Defekte kann auf die Ausbeute an funktionierenden Schaltungen im Verhältnis zu den insgesamt produzierten geschlossen werden. Sie bestimmt wesentlich den notwendigen Umfang und die notwendige Fehlererfassung des Tests. Sein Ziel ist es somit, mit hoher Wahrscheinlichkeit zu verhindern, daß fehlerhafte Schaltungen ausgeliefert und weiterverwendet werden.

3. Teststrategien

Eine Schaltung muß umso genauer geprüft werden, je unzuverlässiger der Produktionsprozeß und je geringer die Ausbeute ist. Zu diesem Zweck kann der Entwerfer unter zahlreichen Teststrategien auswählen. Unter einer Teststrategie versteht man sowohl ein Verfahren zur Testmustererzeugung und zur Testdurchführung zusammen mit den Entwurfsmaßnahmen, die bei der Anwendung dieser Verfahren vorausgesetzt werden müssen, als auch das Fehlermodell. Eine erste Einteilung unterscheidet Strategien zur Unterstützung des externen Tests und Methoden des Selbsttests.

Beim externen Test werden zuvor bestimmte, auf ein Fehlermodell zugeschnittene Bitmuster durch Testautomaten an den Chip angelegt und die Antworten ausgewertet. In der Regel hängen der Aufwand zur Bestimmung der Testmuster und das hierfür geeignete Verfahren davon ab, inwieweit bereits beim Entwurf entsprechende Vorbereitungen getroffen wurden. Solche Vorbereitungen sind beispielsweise die Integration eines Prüfpfades. Bei diesem sogenannten Scan Design werden alle Speicherelemente der Schaltung zu einem Schieberegister zusammengefaßt und während des Tests direkt zugänglich gemacht. Das Scan Design ist zur Unterstützung des externen Tests gedacht, wobei im Testbetrieb der Prüfautomat die Muster in den Prüfpfad einschleibt und die Antworten des Schaltnetzes auf dieses Muster wiederum parallel in das Register geladen und herausgeschoben werden (vgl. Bild 1). Bei Anwendung des Scan Designs genügt es bekanntlich, Algorithmen zur Testsatzerzeugung und -bewertung für Schaltnetze zu entwickeln.

Die weitestgehenden Entwurfsmaßnahmen schließen einen Selbsttest der Schaltungen ein, so daß die Schaltung nur noch zu initialisieren ist, sich im autonomen Betrieb eine gewisse Zeit lang selbst prüft und anschließend mit einer Statusmeldung anzeigt, ob der Test fehlerfrei abgeschlossen wurde. Bei den meisten Selbsttestverfahren werden in der Schaltung vorhandene Re-

gister so ergänzt, daß sie als multifunktionale Schieberegister im Testbetrieb die Prüfmuster erzeugen und auswerten können, sich initialisieren lassen und natürlich auch die normale Systemfunktion erfüllen (Bild 2).

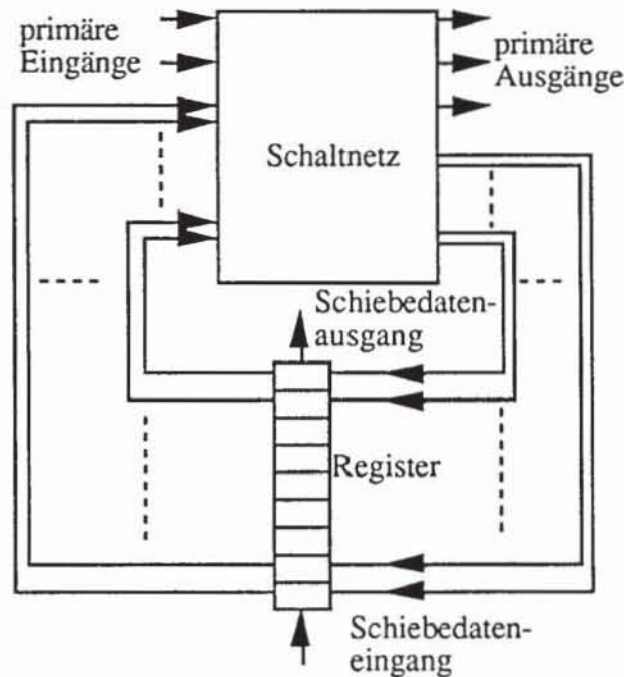


Bild 1: Prinzip des Scan Designs

In der Schaltung nach Bild 2 erzeugt zunächst das Register R1 die Testmuster für das Schaltnetz 1. Register R2 wertet die Antworten des Schaltnetzes aus und komprimiert sie. Der Inhalt von Register R2 gibt dabei an, ob der Test erfolgreich war. Er wird ausgelesen, und dann beginnt R2 mit der Mustererzeugung für Schaltnetz 2 und R1 wertet aus. Derartige Register werden zur Erzeugung gleichverteilter und ungleich verteilter Zufallsmuster sowie zur Erzeugung deterministischer und pseudoerschöpfender Testmengen eingesetzt.

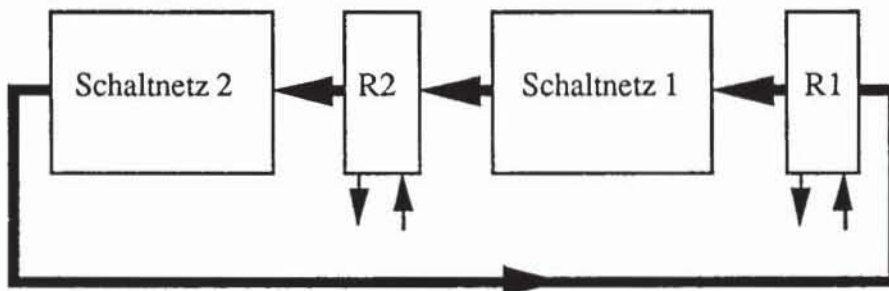


Bild 2: Selbsttest mit multifunktionalen Schieberegister

Zwischen dem externen Test und dem Selbsttest gibt es Abstufungen, bei denen auf teure Prüfautomaten verzichtet und zugleich die Zusatzkosten für den Selbsttest reduziert werden können, indem ein Teil der Selbsttestausstattung nach außen verlagert und günstig als Spezialschaltung realisiert wird. Bild 3 gibt einen Überblick über die gebräuchlichsten Teststrategien.

Zahlreiche, im wesentlichen wirtschaftliche Kriterien bestimmen die Auswahl einer geeigneten Teststrategie. Neben der erwarteten Ausbeute beeinflusst der Anwendungsbereich einer Schaltung die geforderte Güte und damit auch den Umfang des Tests. Dies betrifft nicht nur die Testerzeugung, sondern auch die Verfahren des prüfgerechten Entwurfs, da in vielen Fällen die

Erkennung aller angenommenen Fehler nur durch Schaltungsmodifikationen erreicht werden kann.

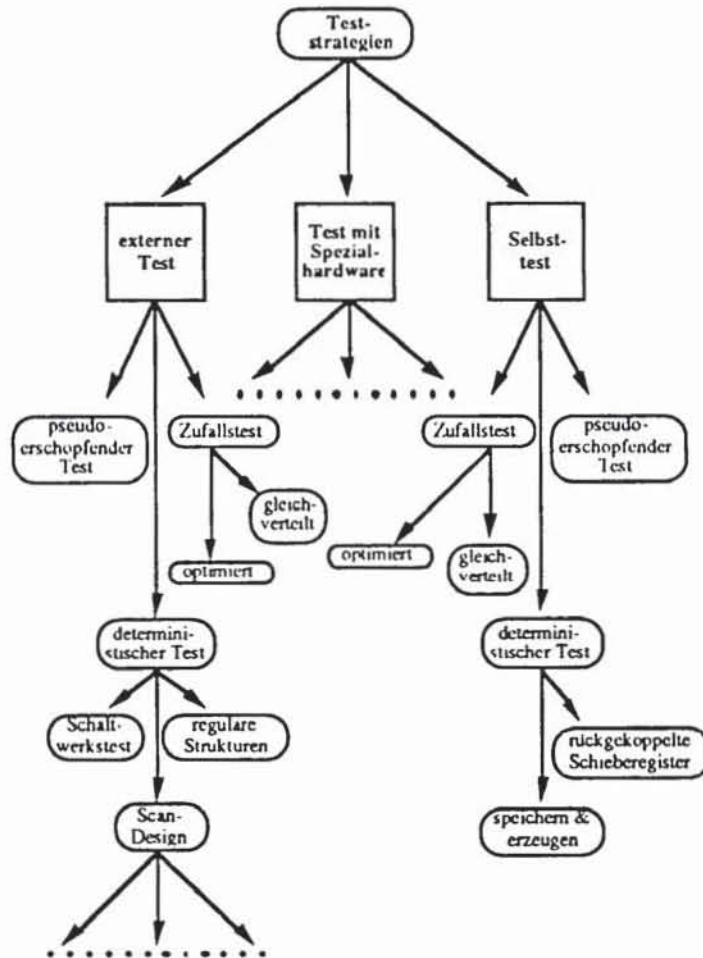


Bild 3: Teststrategien

Einfluß auf die Auswahl der Teststrategien hat auch die geplante Auflage, da Zusatzausstattungen bei jeder produzierten Schaltung Fläche einnehmen und damit Kosten verursachen. Während Entwurf und Testerzeugung bei einer großen Zahl produzierter Chips von den Kosten her geringer ins Gewicht fallen, lohnt sich ein einmaliger hoher Aufwand gerade für anwendungsspezifische Schaltungen in kleinen und mittleren Auflagen nicht, so daß dann günstiger Selbstteststrategien implementiert und keine teuren Testautomaten eingesetzt werden. Vielfach sind diese Zusatzausstattungen auch nützlich für den Systemtest und die Wartung des Gesamtsystems.

4. Der Test mit Zufallsmustern

Sowohl der Selbsttest mit multifunktionalen Schieberegistern als auch der externe Test mit Spezialhardware lassen sich am einfachsten mit Pseudozufallsmustern durchführen. Hierfür ist die Zahl der Muster zu bestimmen, auf welche die Schaltung korrekt antworten muß, damit Fehlerfreiheit mit einer ausreichend hohen Wahrscheinlichkeit angenommen werden kann. Diese notwendige Testlänge hängt von den Wahrscheinlichkeiten ab, mit denen Fehler des Fehlermodells von zufällig erzeugten Mustern erkannt werden. Da auch die Bestimmung von Fehlererkennungswahrscheinlichkeiten ein Problem mit sehr hoher Komplexität darstellt, muß auf Approximationsverfahren zurückgegriffen werden.

Darüber hinaus gibt es zahlreiche Schaltungen, die sich als resistent gegenüber gleichverteilten Zufallsmustern erwiesen haben und die aus diesem Grunde eine unwirtschaftlich lange Folge von Zufallsmustern benötigen. Die Testmenge für derartige Schaltungen kann entscheidend reduziert werden, wenn ungleich verteilte Zufallsmuster erzeugt werden und infolgedessen jeden Eingang der Schaltung mit einer für ihn spezifischen, optimalen Wahrscheinlichkeit auf 1 setzen. Entsprechende Optimierverfahren wie PROTEST finden sich in der Literatur [11]. Da in einer Schaltung unterschiedliche Fehler widersprüchliche Anforderungen an die Verteilung der Zufallsmuster haben können, wurden Verfahren entwickelt, die Menge der Fehler so zu partitionieren, daß jede Teilmenge mit einer relativ kleinen Zahl von Mustern gemäß einer gesonderten Verteilung getestet werden kann [12].

Pseudo-Zufallsmuster lassen sich technisch so günstig erzeugen, daß sie verglichen mit abgespeicherten Mustermengen in größerer Zahl und zugleich in kürzerer Zeit an die Schaltung gelegt werden können. Dies führt zu einer hohen Erfassung auch komplexer Fehlermodelle aus Verzögerungs- und Übergangsfehlern.

5. Testsatzerzeugung und Testsatzbewertung

5.1 Fehlersimulation

Ein entwickelter Testsatz und Zufallstests müssen validiert werden, indem mithilfe einer Fehlersimulation bestimmt wird, welche Fehler des gewählten Modells erkannt werden. Die Fehlersimulation ist ein sehr rechenzeitaufwendiger Schritt im Schaltungsentwurf. Da die Zahl der Fehler linear mit der Schaltungsgröße wächst und die Auswirkung eines jeden Fehlers durch die gesamte Schaltung fortgepflanzt und beobachtet werden muß, nimmt die Rechenzeit quadratisch mit der Schaltungsgröße zu. Das ursprüngliche Verfahren der parallelen Fehlersimulation ist mittlerweile weitgehend von der deduktiven und von der "concurrent" Fehlersimulation abgelöst worden. Bei der parallelen Simulation wird die Wortbreite w des Rechners genutzt, um gleichzeitig die fehlerfreie Schaltung und an den restlichen Bits das Verhalten von $w-1$ Fehlern zu simulieren. Das Verfahren verliert an Effizienz, wenn im Verlauf der Simulation nur wenige Fehler zu abweichendem Verhalten führen und dadurch große Teile des Rechnerworts keine Information tragen.

Die Methoden der deduktiven und "concurrent" Fehlersimulation bestimmen hingegen für jeden Schaltungsknoten stets die Liste aller Fehler, die dort zu einem abweichendem Verhalten führen.

Um Größenordnungen effizientere Algorithmen sind möglich, wenn das Zeitverhalten einer Schaltung nicht berücksichtigt werden muß und nur ein Schaltnetz zu simulieren ist. Dies ist beispielsweise beim Einbau eines Prüfpfades oder von Selbsttestregistern der Fall. Das PPSFP-Verfahren (Parallel Pattern Single Fault Propagation, [13]) nutzt die Möglichkeiten der Parallelverarbeitung auf einem Universalrechner günstiger als die konventionelle parallele Fehlersimulation. Hier werden nicht unterschiedliche Fehler, sondern unterschiedliche Muster in einem Maschinenwort W der Breite w gleichzeitig simuliert und die von ihnen erkannten Fehler ermittelt. Wie in der üblichen Logiksimulation wird jedem Signal der Schaltung ein Wort W_1 zugeordnet, dessen i -tes Bit dem logischen Wert dieses Signals beim Anlegen des i -ten Musters entspricht. Es wird ein Fehler des Modells injiziert und in Maschinenwörtern W_2 das zugehörige fehlerhafte Signal bestimmt. Unterscheiden sich an keinem Gatteranschluß die Wörter W_1 und W_2 , so ist der Fehler mit keinem der w Muster erkennbar und die Simulation wird abgebrochen. Ansonsten wird das Wort W_2 fortgeschaltet, bis ein Primärausgang erreicht und der Fehler erkannt ist. Nacheinander werden auf diese Weise sämtliche Fehler untersucht, und an-

schließlich startet der Prozeß für die nächsten w Muster und die bislang nicht erkannten Fehler von neuem.

Andere Verfahren analysieren die Schaltungsstruktur, um die Zahl der Fehler, die explizit simuliert werden müssen, zu reduzieren. Für Knoten des Schaltnetzes, an denen sich der Signalweg nicht verzweigt, kann mit geringem Aufwand bestimmt werden, ob sein logisches Signal bei einer gegebenen Eingabe an einem Schaltnetzausgang beobachtet werden kann, wenn sein unmittelbarer Nachfolger beobachtbar ist. Es sind daher nur Fehler an Verzweigungsknoten zu simulieren. Durch zahlreiche Heuristiken läßt sich auch diese Menge weiter reduzieren. Schließlich können diese Verfahren mit der PPSFP-Methode zu besonders effizienten Fehlersimulatoren kombiniert werden [1].

Die Fehlersimulation kann durch den Einsatz kommerzieller Spezialrechner oder durch die parallele Bearbeitung von Schaltungsteilen auf verteilten Rechensystemen beschleunigt werden. Gegenwärtige Forschungsarbeiten zielen auf eine Erweiterung dieser Verfahren, um auch das Zeitverhalten der einzelnen Gatter zu beobachten und so Verzögerungs- und Übergangsfehler in einem Schaltnetz berücksichtigen zu können. Ungelöst ist bislang das Problem, auch für sequentielle Schaltungen derart effiziente Methoden zu finden.

5.2 Deterministische Testsatzerzeugung

Die deterministische Testsatzerzeugung gehört wie viele andere im Rahmen der Prüfungsvorbereitung auftretende Aufgabenstellung zu einer Problemklasse, bei deren Lösung nur Algorithmen mit exponentiellem Zeitaufwand bekannt sind. Basierend auf dem von Roth erstmals im Jahre 1966 vorgestellten D-Algorithmus wurden im vergangenen Jahrzehnt jedoch zahlreiche Methoden und Heuristiken entwickelt, die zu einer wesentlichen Steigerung der Leistungsfähigkeit deterministischer Testsatzerzeugungsalgorithmen geführt haben. Stellvertretend für diese neue Generation von effizienten deterministischen Testsatzerzeugungsverfahren für Schaltnetze seien hier der FAN-Algorithmus [5] und das automatische Testmustererzeugungssystem SOCRATES [10] erwähnt. Diese Verfahren organisieren die Testermusterzeugung als Suchbaum, mit Hilfe dessen die Menge aller existierenden Eingangsbelegungen nach einem Testmuster für einen vorgegebenen Zielfehler durchsucht wird. Falls im Vorlauf dieser Suche an einem Knoten des Suchbaumes festgestellt wird, daß keine der diesem Knoten zugeordneten Eingangsbelegungen ein Testmuster für den Zielfehler dargestellt und folglich im Suchbaum zurückgesprungen werden muß, tritt ein sogenanntes "Back-Track" auf. Da die Zahl der anfallenden "Back-Tracks" den Rechenzeitaufwand zur deterministischen Testmustererzeugung wesentlich beeinflusst, zielen alle Methoden zur Verbesserung deterministischer Testsatzerzeugungsalgorithmen primär darauf, die Suche möglichst ohne Rücksprünge zum Erfolg, d. h. zum Auffinden eines Testmusters für den Zielfehler, zu führen. Von zentraler Bedeutung sind in diesem Zusammenhang leistungsfähige Prozeduren zur Durchführung von Implikationen und zwingend notwendige Sensibilisierungsmaßnahmen sowie intelligente Heuristiken zur Steuerung des bei der Suche zu vollziehenden Entscheidungsprozesses. Eine weitere Effizienzsteigerung ermöglicht den Einsatz von Techniken, die auch in anderen Bereichen wie z. B. der künstlichen Intelligenz (KI) Verwendung finden [7], [10].

Die Vorteile der deterministischen Testsatzerzeugung gegenüber dem Test mit Zufallsmustern sind vor allem in der Tatsache zu sehen, daß ein deterministischer Testmustererzeuger die Fähigkeit besitzt, Redundanzen in der Schaltung nachzuweisen, was weder mit gewichteten noch ungewichteten Zufallsmustern möglich ist. Der Nachweis redundanter Schaltungsteile ist von großer Bedeutung, da eine fundierte Beurteilung der Qualität eines Testsatzes nur anhand

nicht redundanter Schaltungsteile möglich ist. Darüber hinaus kann die Identifizierung redundanter Schaltungsteile im Entwurfsprozeß zur Schaltungsminimierung genutzt werden.

Andererseits sind deterministisch erzeugte Testmuster mit höheren Kosten bei der Testdurchführung sowie dem Entwurf und der Implementierung von Selbsttestlogik verbunden. Mit den genannten Verfahren sowohl zur Erzeugung gewichteter Zufallsmuster als auch zur deterministischen Testsatzerzeugung können die in der Praxis auftretenden Schaltnetze ausreichend effizient behandelt werden. Im Hinblick auf Schaltwerke existieren jedoch für beide Arten der Testsatzerzeugung theoretische Grenzen, da rückgekoppelte Schaltungen eine exponentiell wachsende Testlänge benötigen können und somit mit keinem Verfahren die Erstellung eines Testsatzes in praktikabler und akzeptabler Rechenzeit garantiert werden kann.

6. Pseudo-erschöpfende Testmengen

Eine höhere Fehlererfassung als der deterministische Test garantiert der pseudo-erschöpfende Test, bei dem die Funktion eines jeden Schaltungsausgangs vollständig geprüft wird. Während für die Gesamtschaltung ein erschöpfender Test aus Aufwandsgründen zumeist ausgeschlossen ist, hängt ein einzelner Ausgang zumeist nur von einer kleinen Menge der primären Eingänge ab und kann daher erschöpfend geprüft werden.

Falls ein Ausgang dennoch von zu vielen Primäreingängen abhängt und der pseudo-erschöpfende Test für diesen Ausgang somit nicht wirtschaftlich durchführbar ist, muß der Entwurf segmentiert werden. Hierfür stehen zwei grundlegende Ansätze zur Verfügung. Bei der Segmentierung durch Pfadsensibilisierung werden deterministisch Testmuster generiert, die Pfade zu einer hinreichend kleinen Teilstruktur aktivieren und den pseudo-erschöpfenden Test dadurch ermöglichen. Die Hardware-Segmentierung hingegen zerlegt die Schaltung logisch durch den Einbau von Zusatzzellen. Die klassische Methode hierfür sind Multiplexer [9]. Eine weitere Möglichkeit der Hardware-Segmentierung besteht in der Integration zusätzlicher Latches, die im Systembetrieb transparent geschaltet werden und während des Tests als Teil des Prüfpfades zusätzliche Eingänge und Ausgänge bilden [6].

Pseudo-erschöpfende Testmengen, die bei einem möglichst geringen Umfang alle Ausgangsfunktionen der Schaltung vollständig prüfen, können algorithmisch bestimmt und durch einen Testautomaten in einen Prüfpfad geschoben werden, es ist aber auch möglich, sie durch geeignete Schaltungen on-line zu generieren.

7. Aktuelle Entwicklungen

Gegenwärtige Forschungsarbeiten konzentrieren sich darauf, die Effizienz der Verfahren zur Testsatzerzeugung und -bewertung insbesondere für sequentielle Schaltungen zu verbessern, komplexere und realitätsnähere Fehlermodelle zu berücksichtigen und Verfahren zu entwickeln, die möglichst wenig Zusatzausstattung auf der Schaltung benötigen. Des Weiteren wird versucht, der wachsenden Schaltungs- und damit Problemkomplexität durch den Einsatz von Spezialrechnerarchitekturen (Vektorprozessoren, Parallelprozessoren, dedizierte Hardware-Beschleuniger) Herr zu werden und eine Verbesserung der Qualität der Problemlösungen zu erzielen.

Zur Verbesserung der Effizienz untersucht man u. a. den Nutzen der Modularisierung und der Hierarchiebildung. Zur Berücksichtigung realitätsnaher Fehler werden entsprechende Testmustergeneratoren und Fehlersimulatoren insbesondere für Verzögerungsfehler entwickelt. Eine modulare Testerzeugung ermöglicht es, die Tests auf Switch-Level oder gar auf elektrischer Ebene für einzelne Zellen zu generieren.

Zahlreiche Forschungsarbeiten beschäftigen sich mit den Problemen der Testerzeugung und der Fehlersimulation für synchrone sequentielle Schaltungen. Neben den Aktivitäten, die auf eine effiziente Behandlung sequentieller Schaltungen beliebiger Struktur und gänzlich ohne Prüfpfad zielen, ist vor allem eine Reihe von Arbeiten erwähnenswert, die sich mit der optimalen Konfiguration eines unvollständigen Prüfpfades befaßt. Bei entsprechend intelligenter Auswahl der in den unvollständigen Prüfpfad einzubindenden Speicherelemente lassen sich beträchtliche Hardware-Einsparungen im Vergleich zu einem vollständigen Prüfpfad erreichen und dabei dennoch eine ähnlich gute Testbarkeit der Schaltung gewährleisten. Für Schaltungen dieser Art werden derzeit nicht nur deterministische Testerzeugungs- und Fehlersimulationsverfahren entwickelt, sondern auch Methoden für den Zufallstest und den pseudo-erschöpfenden Test erforscht.

Die Punkte aktueller Arbeiten machen deutlich, daß der prüfgerechte Entwurf als wesentlicher Bestandteil der Testvorbereitung betrachtet werden muß. Er entscheidet, welches Fehlermodell angenommen werden muß und welche Verfahren der Testerzeugung angewendet werden können.

Literatur

- [1] Antreich, K.J.; Schulz, M.H.: Accelerated Fault Simulation and Fault Grading in Combinational Circuits; in: IEEE Trans. on CAD September 1987
- [2] Bennetts, R.G.: Design of Testable Logic Circuits; Addison-Wesley, 1984
- [3] Bardell, P. H.; McAnney, W. H.; Savir, J.: Built-In Test for VLSI: Pseudorandom Techniques, John Wiley & Sons, 1987
- [4] Ferguson, F. J.; Shen, J. P.: Extraction and Simulation of Realistic CMOS Faults Using Inductive Fault Analysis; in: Proc. IEEE 1988 Int. Test Conf.
- [5] Fujiwara, H.; Shimono, T.: On the Acceleration of Test Generation Algorithms, Proc. FTCS-13, 1983
- [6] Hellebrand, S.; Wunderlich, H.-J.: Automatisierung des Entwurfs vollständig testbarer Schaltungen; in: Proc. GI - 18. Jahrestagung II, 1988, Springer-Verlag
- [7] Krishnamurthy, B.: Hierarchical Test Generation: Can AI Help?; in: Proc. IEEE 1987 Int. Test Conf.
- [8] Maly, W.: Realistic Fault Modeling for VLSI Testing in: Proc. 24th Des. Aut. Conf., Miami Beach 1987
- [9] McCluskey, E.J.; Bozorgui-Nesbat, S.: Design for Autonomous Test; IEEE Trans. on Circuits and Systems, November 1981
- [10] Schulz, M.H.; Auth, E.: Advanced Automatic Test Pattern Generation and Redundancy Identification Techniques; in: Proc. FTCS-18, Tokyo, 1988
- [11] Wunderlich, H.-J.: PROTEST: A Tool for Probabilistic Testability Analysis; in: Proc. 22nd Des. Aut. Conf., 1985
- [12] Wunderlich, H.-J.: Multiple Distributions for Biased Random Test Patterns; in: Proc. 1988 IEEE Int. Test Conf., Washington
- [13] Waicukauski, J.A. et al.: Fault Simulation for Structured VLSI; in: VLSI Systems Design, December 1985