

Prüfgerechter Entwurf und Test hochintegrierter Schaltungen

H.-J. Wunderlich und M.H. Schulz

Universität-GH Siegen und Siemens Nixdorf, München

Zusammenfassung. Der Beitrag gibt einen Überblick über die wichtigsten praxisrelevanten Teststrategien, wobei unter einer Teststrategie nicht nur die Verfahren zur Testsatzerzeugung und zur eigentlichen Testdurchführung, sondern auch das zugrunde liegende Fehlermodell und die erforderlichen testfreundlichen Entwurfsmaßnahmen, die die Voraussetzung für die Anwendung dieser Verfahren darstellen, zu verstehen sind. Es werden die gängigsten Methoden zum konventionellen externen Test vorgestellt und bewertet sowie das Prinzip der immer breitere Anwendung findenden Selbsttestmethoden und ihre Vorteile erläutert. Nach einem kurzen Ausblick auf die Fortschritte, die Verfahren zur automatischen Synthese testbarer Schaltungen erhoffen lassen, werden schließlich Aspekte des Systemtests und insbesondere das Boundary-Scan-Prinzip und die damit verbundenen Vorteile diskutiert.

Schlüsselwörter: Prüfgerechter Entwurf, Test, Selbsttest, hochintegrierte Schaltungen

Summary. This paper's aim is to present a survey of testing strategies which are most important and relevant from the practical point of view. Thereby, a test strategy is considered to comprise not only the methods for test generation and actual test execution, but also the appropriate fault model and the required design for testability techniques, which enable and are the prerequisite for the successful application of these methods. In particular, the most wide-spread methods for the conventional external test will be described and rated. Furthermore, the basic principle of the increasingly popular built-in self-test techniques and their advantages will be elucidated. Finally, after a brief view of the future progress, which automatic synthesis tools that guarantee the synthesized circuits to be easily testable raise hopes for, board- and system-level testing strategies and, in particular, the boundary scan approach and the advantages emerging from it will be discussed.

Key words: Design for testability, Test, Built-in self-test, VLSI

Computing Reviews Classification: B.7.3, B.1.3, B.2.3, B.5.3, B.6.2–3

1. Das Testproblem für hochintegrierte Schaltungen

Hochintegrierte Schaltungen durchlaufen in ihrem Lebenszyklus viele aufwendige, verschiedenartige Tests. Vor Aufnahme der Serienfertigung müssen zunächst Prototypen getestet werden. Während der Serienfertigung selbst treten mit statistischer Gesetzmäßigkeit Defekte auf, die die Produktionstests erkennen sollen. Schließlich müssen die Schaltungen während der Reparatur und der Wartung des Gesamtsystems geprüft werden. Auch sorgfältigere Herstellungsverfahren machen den Test und die Aussonderung defekter Schaltungen nicht überflüssig. Die steigende Integrationsdichte mit Leitungsbreiten in der Größenordnung von $1\ \mu\text{m}$ führt dazu, daß immer kleinere Defekte den Ausfall einer Schaltung verursachen können.

Ausfallursachen, Defektmechanismen, zu erwartende Fehlfunktionen und damit auch die Testerzeugung sind in hohem Maße technologieabhängig. Mit Hilfe empirisch gewonnener Daten und analytischer Verfahren wird aus dem Layout einer Schaltung hergeleitet, mit welchen Defekten in der Schaltungsstruktur zu rechnen ist, wie häufig sie auftreten und zu welchen Fehlfunktionen im Schaltungsverhalten sie führen können [8, 17]. Um die rechnergestützte Behandlung der unterschiedlichen Defektarten zu ermöglichen, werden *Fehlermodelle* eingeführt, die die in der Praxis relevanten Fehlermechanismen entsprechend ihren Auswirkungen mehr oder weniger genau beschreiben. Für die modellierten Fehler ist dann ein Testsatz zu erzeugen, der sie möglichst vollständig erfaßt.

In der Vergangenheit wurden zahlreiche Fehlermodelle jeweils für bestimmte Entwurfsstile und Herstellungstechniken entwickelt. Neben dem klassischen, für bipolare Schaltungen adäquaten Haftfehlermodell, dem die Annahme zugrunde liegt, daß fehlerbehaftete Signale

ständig den logischen Wert 0 oder 1 haben, finden in jüngerer Zeit verstärkt dynamische und komplexe, insbesondere für MOS-Schaltungen typische Fehler wie stuck-open- und stuck-on-Fehler Berücksichtigung. Da ein sehr komplexes Fehlverhalten die Testerzeugung erschwert, werden Entwurfsregeln untersucht, die dies ausschließen oder zumindest unwahrscheinlich machen. Nicht die Verifikation der korrekten Schaltungsfunktion ist also das Ziel des Tests, sondern lediglich der Nachweis, 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 Umfang und den für das geforderte Qualitätsniveau notwendigen *Fehlererfassungsgrad*, der als das Verhältnis der Zahl der durch den Test erkannten Fehler zur Gesamtzahl aller durch das Fehlermodell modellierten Fehler definiert ist. Das Ziel des Tests ist es somit, mit hoher Wahrscheinlichkeit zu verhindern, daß fehlerhafte Schaltungen ausgeliefert und verwendet werden.

Der Test von hochintegrierten Schaltungen unterscheidet sich grundsätzlich vom Softwaretest, der bekanntlich ein Programm vor seiner Auslieferung und Installation beim Kunden validieren soll. Der Softwaretest beschränkt sich dabei auf *eine* Implementierung des Programms, von der beliebig viele Kopien weitergegeben werden. Man geht davon aus, daß das Kopierprogramm getestet und korrekt ist, und verzichtet auf eine Überprüfung der Programmkopien. Dagegen müssen integrierte Schaltungen nicht nur als Prototyp geprüft werden, sondern es ist jeder auszulieferende Chip zu testen. Daraus ergeben sich deutlich andere Anforderungen an die Wirtschaftlichkeit und an die Effizienz einer Teststrategie.

2. Testkosten

Der Aufwand zur Testerzeugung und Testdurchführung wächst überproportional mit der Zahl der in einem Chip integrierten Funktionen. Bereits heute können die Testkosten für anwendungsspezifische Schaltungen (ASICs) 60 bis 70 % der Gesamtkosten ausmachen.

Der unverhältnismäßig große Kostenanstieg für den Test hat mehrere Ursachen. Die *Zugänglichkeit* der in einer Schaltung realisierten Moduln verschlechtert sich mit steigender Integrationsdichte. Zur Erzeugung eines qualitativ hochwertigen *Testdatensatzes* müssen Probleme von hoher Komplexität gelöst werden, der Aufwand dafür wächst im schlimmsten Fall exponentiell und in der Praxis quadratisch bis kubisch mit der Schaltungsgröße. Zwar steigt die für einen ausreichenden Fehlererfassungsgrad erforderliche *Zahl der Testmuster* in der Regel linear an, die *Zugänglichkeitsprobleme* können jedoch zu der Notwendigkeit führen, die Testmuster seriell in die Schaltung einzugeben. Da auch die Zahl der zu dieser seriellen Eingabe notwendigen Taktzyklen typischerweise linear mit der Schaltungsgröße wächst, kann die *Testdurchführungszeit* quadratisch zunehmen.

Die umfangreichen, aus den eigentlichen Testmustern und den Sollantworten bestehenden Prüfdaten müssen

der Schaltung in großer Geschwindigkeit zugeführt werden, da zahlreiche Defekte nicht die logische Funktion, jedoch die Schaltzeit beeinträchtigen (dynamische Fehler). Insgesamt müssen die *Prüfgerate* mindestens so leistungsfähig wie die zu testenden Schaltungen sein, d. h. hierfür muß die neueste Technologie zu hohen Kosten verwendet werden.

Die enormen Fortschritte beim automatischen Entwurf und bei der Herstellung sowie die damit zu erzielende Komplexität integrierter Schaltungen sind schließlich ein weiterer wesentlicher Faktor für den überproportionalen Anstieg der Testkosten. Zum einen werden immer neue Anwendungsbereiche erschlossen, die in vielen Fällen sicherheitsempfindlich sind und demzufolge besonders hohe Qualitätsanforderungen an hochintegrierte Schaltungen stellen, die nur durch aufwendige Verfahren für Tests und Fehlertoleranz erfüllt werden können. Als Beispiele seien die Automobiltechnik, die Luft- und Raumfahrt und die Medizintechnik erwähnt. Zum anderen konnte die zur Schaltungsentwicklung benötigte Zeit mit den zahlreichen neuen Verfahren des rechnergestützten Entwurfs um mehrere Größenordnungen gesenkt werden. Beispielsweise stellen Standardzellensysteme dem Entwerfer eine Bibliothek von Bauelementen zur Verfügung, mit Hilfe derer er eine die Schaltung beschreibende Netzliste erstellen kann. Anschließend kann die Schaltung automatisch plaziert und verdrahtet werden (physikalischer Entwurf). Die zunehmende Verfügbarkeit von sogenannten Silicon Compilers ermöglicht eine automatische Generierung relativ umfangreicher Standardzellenblöcke wie beispielsweise RAMs. Solche Werkzeuge können 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 der Test bereits beim automatischen Entwurf berücksichtigt wird.

3. Teststrategien

Eine Schaltung muß um so umfassender, d. h. auf der Basis eines genaueren Fehlermodells und mit höherem Fehlererfassungsgrad geprüft werden, je unzuverlässiger der Produktionsprozeß und je höher die Qualitätsanforderungen sind [26]. Zur Anpassung an die konkreten Bedürfnisse kann der Entwerfer unter zahlreichen Teststrategien wählen. Unter einer Teststrategie versteht man sowohl das Fehlermodell als auch ein Verfahren zur Testmustererzeugung und zur Testdurchführung zusammen mit den Entwurfsmaßnahmen, die bei der Anwendung dieser Verfahren vorausgesetzt werden müssen. Eine erste Einteilung unterscheidet Strategien zur Unterstützung des externen Tests und Methoden des Selbsttests.

Beim externen Test werden 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 Test-

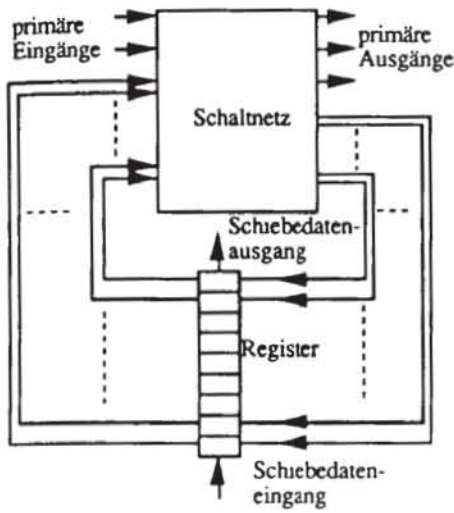


Abb. 1. Prinzip des Scan Design

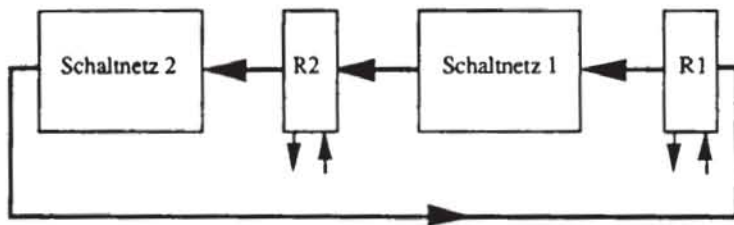


Abb. 2. Selbsttest mit multifunktionalen Schieberegistern

muster und das hierfür geeignete Verfahren davon ab, inwieweit bereits beim Entwurf entsprechende Vorbereitungen getroffen wurden. Eine solche vorbereitende Maßnahme ist beispielsweise die Integration eines Prüfpfades. Bei diesem sog. "Scan Design" werden alle Speicherelemente der Schaltung zu einem oder mehreren getrennten Schieberegistern zusammengefaßt und dadurch während des Tests über Schiebeeingänge und -ausgänge direkt ladbar und auslesbar gemacht. Das Scan Design ist hauptsächlich zur Unterstützung des externen Tests gedacht. Die Muster werden mit Hilfe des Prüfautomaten in den Prüfpfad bzw. die Prüfpfade geschoben und die Testantworten des Schaltnetzes in das oder die Register übernommen und schließlich aus der Schaltung ausgeschoben (Abb. 1). Bei Anwendung des Scan Design genügt es, 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 bestimmte Zeit selbst prüft und anschließend mit einer Statusmeldung anzeigt, ob der Test fehlerfrei abgeschlossen wurde. Bei den meisten Selbsttestverfahren werden in der Schaltung vorhandene Register 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 (Abb. 2).

In der Schaltung nach Abb. 2 erzeugt zunächst das Register R1 die Testmuster für das Schaltnetz 1. Register R2 wertet die Antworten des Schaltnetzes aus und komprimiert sie. Nach Ende des vollständigen Tests gibt der In-

halt von Register R2 an, ob der Test erfolgreich war, und wird ausgelesen. Dann beginnt R2 mit der Mustererzeugung für Schaltnetz 2, und R1 wertet aus.

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

Zahlreiche, im wesentlichen wirtschaftliche Kriterien bestimmen die Auswahl einer 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 das Erkennen aller angenommenen Fehler nur durch Schaltungsmodifikationen erreicht werden kann.

Einfluß auf die Wahl der Teststrategien hat auch die geplante Stückzahl, 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 weniger ins Gewicht fallen, lohnt sich ein einmaliger ho-

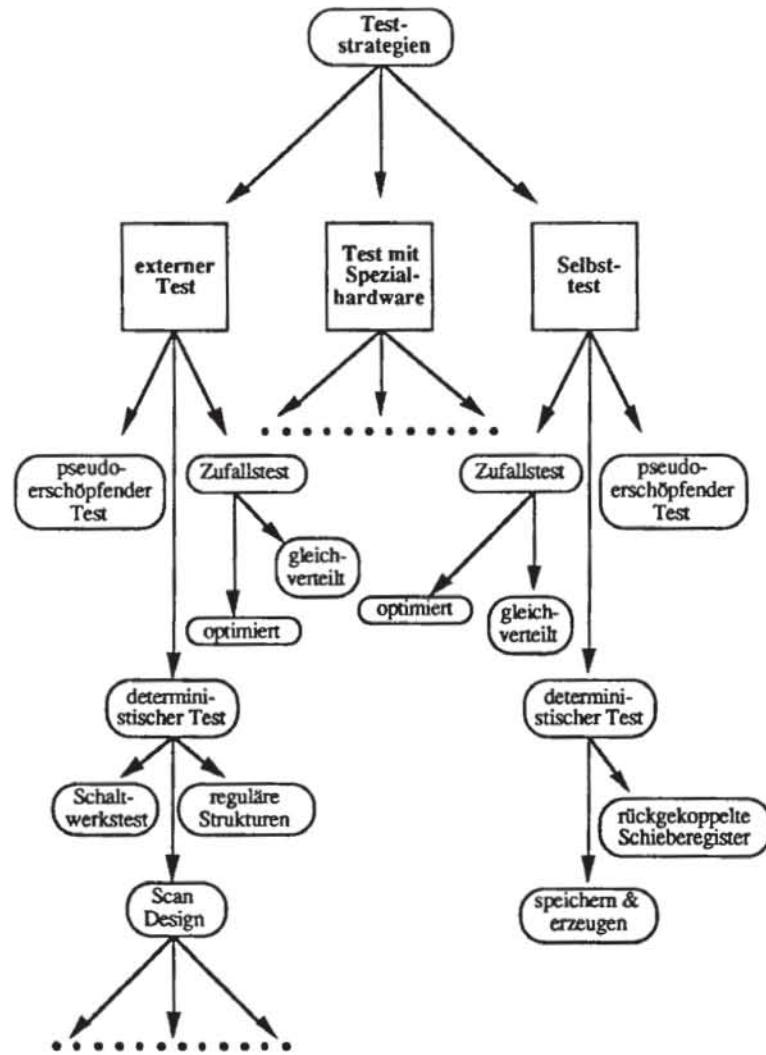


Abb. 3. Teststrategien

her Aufwand gerade für anwendungsspezifische Schaltungen in kleinen und mittleren Auflagen nicht, so daß dann günstiger Selbstteststrategien implementiert und keine teuren Testautomaten eingesetzt werden. In jedem Fall muß die Teststrategie sicherstellen, daß die aus wirtschaftlicher Sicht äußerst wichtigen Vorteile einer schnellen Markteinführung ("Time-to-Market") der jeweiligen Produkte gewahrt werden, was wiederum nur durch die Entwicklung und den durchgängigen Einsatz von leistungsfähigen rechnergestützten Verfahren erreichbar ist. Darüber hinaus sollte sie flexibel und erweiterbar sein, um auch für zukünftige Produktgenerationen effizient eingesetzt werden zu können.

4. Der externe Test

Die wichtigsten Schritte beim externen Schaltungstest sind die Erzeugung eines Testsatzes, die Fehlersimulation und die Testdurchführung.

4.1. Deterministische Testsatzerzeugung

Die deterministische Testsatzerzeugung gehört wie viele andere bei der Prüfvorbereitung auftretende Aufgaben zu einer Problemklasse, für deren Lösung nur Algorithmen mit exponentiellem Zeitaufwand bekannt sind. Basierend auf dem von Roth erstmals 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 [9] und das automatische Testmustererzeugungssystem SOCRATES [22] erwähnt. Diese Verfahren organisieren die Testmustererzeugung als Suche, bei der die in einem Suchbaum enthaltene Menge aller Eingangsbelegungen implizit nach einem Testmuster für einen vorgegebenen Zielfehler durchsucht wird. Wenn bei dieser Suche an einem Knoten des Suchbaums festgestellt wird, daß keine der diesem Knoten zugeordneten Eingangsbelegungen ein Testmuster für den Zielfehler darstellt, und folglich im Suchbaum zurückgesprungen werden muß, tritt ein sog. "Backtrack" auf. Da die Zahl der anfallenden Backtracks den Rechenaufwand zur deterministischen Testmustererzeugung wesentlich beeinflusst, zielen alle Methoden zur Verbesserung deterministischer Testsatzerzeugungsalgorithmen primär darauf ab, 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 logischer Implikationen und zwingend notwendiger Maßnahmen zur Sensibilisierung einzelner Pfade im Schaltnetz sowie intelligente Heuristiken zur Steuerung des bei der Suche zu vollziehenden Entscheidungsprozesses. Die Aufgabe der Implikationsprozedur besteht darin, alle logischen Werte, die

aufgrund bereits vorgenommener Wertzuweisungen eindeutig bestimmt sind, zu erkennen und den entsprechenden Signalen zuzuweisen. Unter zwingend notwendigen Sensibilisierungsmaßnahmen hingegen sind Wertzuweisungen zu verstehen, die unbedingt getroffen werden müssen, um die Effekte des gerade betrachteten Zielfehlers zu einem der Schaltungsausgänge fortpflanzen zu können [20].

Im Hinblick auf Schaltwerke gibt es jedoch für die Testsatzerzeugung theoretische Grenzen, da rückgekoppelte Schaltungen eine exponentiell wachsende Testlänge benötigen können und daher mit keinem Verfahren die Erstellung eines Testsatzes in akzeptabler Rechenzeit garantiert werden kann.

4.2. Pseudo-erschöpfende Testverfahren

Insbesondere bei MOS-Technologien können zahlreiche Fehlfunktionen auftreten, die mit klassischen Fehlermodellen auf Gatterebene nicht beschrieben und demzufolge von den darauf zugeschnittenen Testerzeugungsverfahren nicht behandelt werden können. Für Schaltungen, bei denen sehr hohe Anforderungen an die Produktqualität gestellt werden, sind daher Testverfahren zweckmäßig, die von einer Schaltungsbeschreibung auf Gatterebene ausgehen und dennoch eine möglichst vollständige Erfassung der tatsächlich zu erwartenden Fehler garantieren.

Eine hohe, von der Fertigungstechnologie weitgehend unabhängige Fehlererfassung garantiert bei Schaltnetzen der pseudo-erschöpfende Test. Dabei wird für jeden Ausgang o eines Schaltnetzes das minimale Teilschaltnetz $K(o)$ aller Bausteine bestimmt, von denen o abhängt, und durch Aufzählen aller möglichen Eingangsbelegungen erschöpfend getestet. Die Teilschaltnetze $K(o)$ heißen Abhängigkeitskegel der Ausgänge o (Abb. 4).

Der pseudo-erschöpfende Test gestattet im Gegensatz zum Test mit Mustern, die von üblicherweise auf der Einzelfehlerannahme beruhenden deterministischen Testsatzerzeugungsalgorithmen erstellt werden, die Erfassung aller Mehrfachfehler innerhalb der einzelnen Kegel sowie eines Großteils der Mehrfachfehler in der Gesamtschaltung. Zeitabhängige Fehler können durch geeignete Zusatzmaßnahmen und die Durchführung des Tests mit den entsprechenden Geschwindigkeiten ebenfalls erkannt werden [31]. Ein weiterer, nicht unbedeutender

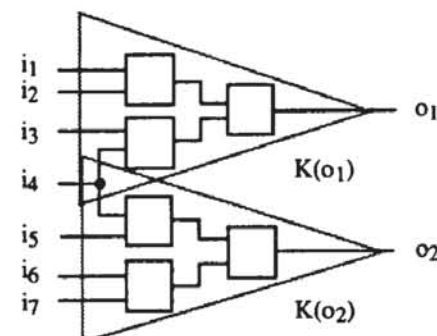


Abb. 4. Beispielschaltnetz mit zwei Abhängigkeitskegeln

Vorteil des pseudo-erschöpfenden Tests ist schließlich, daß die in der Regel aufwendige Fehlersimulation nicht erforderlich ist. Darüber hinaus kann der Aufwand zur Testmuster-generierung eingespart werden, wenn die erschöpfenden Tests der einzelnen Kegel nacheinander durchgeführt werden.

Die Zahl t der für einen pseudo-erschöpfenden Test notwendigen Testmuster läßt sich zu $2^n \leq t \leq |O| \cdot 2^n$ abschätzen, wobei O die Menge der primären Ausgänge und n die maximale Anzahl der Eingänge in die einzelnen Teilschaltnetze bezeichnet. Um eine Testdurchführung im Sekundenbereich zu gewährleisten, darf daher n eine Schranke, die von der vorgegebenen Testdauer abhängt, nicht überschreiten.

Wenn 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 [18]. 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 [11]. Erwähnt sei an dieser Stelle, daß der pseudo-erschöpfende Test derzeit weitgehend Gegenstand der Forschung ist und in der industriellen Praxis erst vereinzelt verwendet wird.

4.3. 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 die die Schaltung korrekt antworten muß, damit Fehlerfreiheit mit ausreichend hoher 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 sehr komplexes Problem ist, muß auf Approximationsverfahren zurückgegriffen werden.

Darüber hinaus haben sich zahlreiche Schaltungen als resistent gegenüber gleichverteilten Zufallsmustern erwiesen, benötigen also eine unwirtschaftlich lange Folge von Zufallsmustern. Die Testmenge für derartige Schaltungen kann entscheidend reduziert werden, wenn ungleichverteilte Zufallsmuster erzeugt werden, die jeden Eingang der Schaltung mit einer für ihn spezifischen, optimalen Wahrscheinlichkeit auf 1 setzen. Entsprechende Optimierungsverfahren wie PROTEST finden sich in der

Literatur [27]. Da in einer Schaltung unterschiedliche Fehler widersprüchliche Anforderungen an die Verteilung der Zufallsmuster stellen können, wird die Menge der Fehler so partitioniert, daß jede Teilmenge mit einer relativ kleinen Zahl von Mustern gemäß einer gesonderten Verteilung getestet werden kann [29, 25].

Pseudozufallsmuster lassen sich technisch so günstig erzeugen, daß sie in größerer Zahl und zugleich in kürzerer Zeit als abgespeicherte Mustermengen an die Schaltung gelegt werden können. Dies führt zu höheren Fehlererfassungsgraden auch bei komplexen Fehlfunktionen, z. B. Verzögerungs- und Übergangsfehlern.

4.4. Fehlersimulation

Ein entwickelter Testsatz muß validiert werden, indem mit Hilfe einer Fehlersimulation bestimmt wird, welche Fehler des gewählten Modells erkannt werden. Da die Zahl der Fehler linear mit der Schaltungsgröße, d. h. der Zahl der Gatter, wächst und die Auswirkung eines jeden Fehlers durch die gesamte Schaltung propagiert und beobachtet werden muß, nimmt die Rechenzeit pro simuliertem Muster im schlimmsten Fall quadratisch mit der Schaltungsgröße zu. Das ursprüngliche Verfahren der parallelen Fehlersimulation wurde mittlerweile weitgehend von der deduktiven und von der "concurrent" Fehlersimulation abgelöst. 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. Die Vorteile der parallelen Fehlersimulation bezüglich Rechenzeitbedarf gegenüber der klassischen seriellen Fehlersimulation verlieren an Bedeutung, wenn im Verlauf der Simulation nur wenige Fehler zu abweichendem Verhalten führen und dadurch große Teile des Rechnerworts keine nützliche Information tragen.

Die deduktive und die "concurrent" Fehlersimulation bestimmen hingegen für jeden Schaltungsknoten stets die Liste aller Fehler, die dort zu einem abweichenden Verhalten führen. Somit läßt sich Rechenzeit einsparen, wenn nur wenige Fehler einen Knoten beeinflussen, allerdings kann der Speicherbedarf sehr groß werden, falls pro Knoten umfangreiche Fehlerlisten zu berücksichtigen sind.

Um Größenordnungen effizientere Algorithmen sind möglich, wenn nur Schaltnetze zu simulieren sind und deren Zeitverhalten im Rahmen der Fehlersimulation nicht berücksichtigt werden muß. Dies ist beispielsweise beim Einbau eines Prüfpfades oder von Selbsttestregistern der Fall. Das PPSFP-Verfahren (Parallel Pattern Single Fault Propagation, [24]) nutzt die Möglichkeiten der Parallelverarbeitung boolescher Operationen auf einem Universalrechner besser als die konventionelle parallele Fehlersimulation. Dabei werden nicht unterschiedliche Fehler, sondern w unterschiedliche Muster in einem Maschinenwort der Breite w gleichzeitig simuliert und die von ihnen erkannten Fehler ermittelt. Hierfür wird jedem Signal der Schaltung ein Wort WI zugeordnet, dessen i -tes Bit dem logischen Wert dieses Signals beim Anlegen des i -ten Musters entspricht. Während der eigentlichen Fehlersimula-

tion werden die Fehler nacheinander jeweils einzeln injiziert und in Maschinenwörtern W_2 die zugehörigen fehlerhaften Signalbelegungen 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 mit dem Wort W_2 weitersimuliert, bis ein Primärausgang erreicht und der Fehler erkannt ist. Nacheinander werden auf diese Weise sämtliche Fehler untersucht, und anschließend 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 zu reduzieren, die explizit simuliert werden müssen. 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 [3].

Die Fehlersimulation kann durch den Einsatz kommerzieller Spezialrechner oder durch die parallele Bearbeitung von Schaltungsteilen auf verteilten Rechen-systemen 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. Ungeklärt ist bislang das Problem, auch für sequentielle Schaltungen derart effiziente Methoden zu finden.

4.5. Testdurchführung

Während beim Prototypentest die aufwendigere Fehlerdiagnose im Mittelpunkt steht, um Entwürfe zu validieren und gegebenenfalls zu korrigieren, beschränkt man sich beim Produktionstest auf die Fehlererkennung. Dennoch ist der Produktionstest in der Serienfertigung kostenintensiver, da er für jeden einzelnen Chip durchgeführt werden muß. Zunächst werden in einem Vor-Test die Wafer- und Chipplätze ausgewählt, die keine offensichtlichen Fehler aufweisen. Danach muß jeder einzelne Chip mehrere Testschritte durchlaufen. Es wird geprüft, ob zwischen den primären Anschlüssen der Schaltung Kurzschlüsse bestehen, um danach anhand eines kurzen funktionalen Tests zu entscheiden, ob die Fortsetzung der Tests überhaupt sinnvoll ist. Nach der Messung ausgewählter statischer und dynamischer Parameter wird die logische Funktion der Schaltung getestet. Sie wird mit einer Menge von Bitmustern stimuliert, die Antworten werden aufgenommen und mit den Sollwerten verglichen. Dazu sind Testgeräte erforderlich, die die Prüfdaten mit hinreichender Geschwindigkeit erzeugen und mit hinreichender Auflösung aufnehmen können. Die Anforderungen an diese Testautomaten sind daher zwangsläufig höher als die an die zu testenden Schaltungen.

4.5.1. Testautomaten

Wesentliche Parameter eines Testautomaten ("Automatic Test Equipment", ATE), die auch seine Kosten bestimmen, sind Testgeschwindigkeit, Zahl der testbaren Anschlüsse, Durchsatz, Genauigkeit und Auflösung. Heutige Testgeräte können bis zu 512 Anschlüsse bedienen, Testmuster mit einer Genauigkeit im Pikosekunden-Bereich anlegen und Testantworten mit einer ähnlichen Auflösung aufnehmen. Sie erreichen eine Taktrate von über 100 MHz. Hochleistungstestgeräte müssen durch schnelle automatische Handtierung einen hohen Durchsatz gewährleisten und zugleich den Test ausreichend genau durchführen können. Beide Anforderungen verteuern die Geräte sehr; die Preise liegen derzeit (Stand 1990) zwischen einer halben Million und mehreren Millionen DM.

Eines der Hauptprobleme beim Einsatz von Testautomaten ist das stetige Anwachsen der Datenmenge. Wenn sie die Kapazität des schnellen Speichers des Testautomaten übersteigt, muß vom Host-Rechner nachgeladen werden. Ein Nachladevorgang kostet die sechs- bis zehnfache Zeit, die für den eigentlichen Test mit dem Speicherinhalt erforderlich ist. Dies reduziert den Durchsatz, was möglicherweise durch den Einsatz zusätzlicher Tester ausgeglichen werden muß, und verteuert die Schaltungen dadurch wesentlich. In [4] wird prognostiziert, daß mit herkömmlichen Methoden der Testdurchführung eine für die 90er Jahre typische CMOS-Produktgeneration bis zu 13 Nachladevorgänge erfordern wird. Dies ist nicht mehr wirtschaftlich, so daß man derzeit als Alternativen den eingebauten Selbsttest sowie den Test mit Spezialschaltungen (s. 4.5.2) untersucht und in die industrielle Praxis überführt [25, 4].

4.5.2. Test mit Spezialschaltungen

Grundlage für den externen Test mit Zufallsmustern sind Spezialschaltungen, die mittels linear rückgekoppelter Schieberegister Pseudozufallsfolgen erzeugen, sie miteinander gemäß programmierbarer boolescher Funktionen verknüpfen und auf diese Weise gewichtete, ungleichverteilte Zufallsmuster ausgeben. Diese Muster werden in den Prüfpfad und an die Primäreingänge des Prüflings gegeben (Abb. 5). Es sind daher keine umfangreichen Testmengen, sondern lediglich einige Wahrscheinlichkeitswerte abzuspeichern. Ein Nachladen entfällt, und die Testmuster können algorithmisch von der Spezialschaltung mit der Betriebsfrequenz der zu testenden Schaltung erzeugt werden [29, 25, 4, 30]. Ähnliche Spezialschaltungen wurden auch für pseudo-erschöpfende Testmengen vorgeschlagen [12].

5. Selbsttestbare Schaltungen

Selbsttestbare Schaltungen können intern Prüfmuster erzeugen und auswerten. Dies bietet eine Reihe von Vorteilen. Da sich im günstigsten Fall die Anforderungen an externe Testgeräte darauf reduzieren, den Chip zu

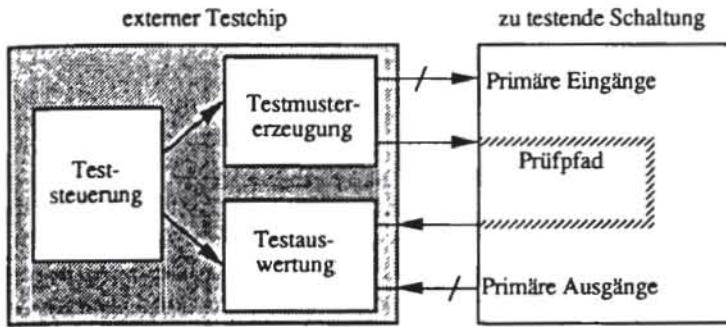


Abb. 5. Externer Test mit Zufallsmustern

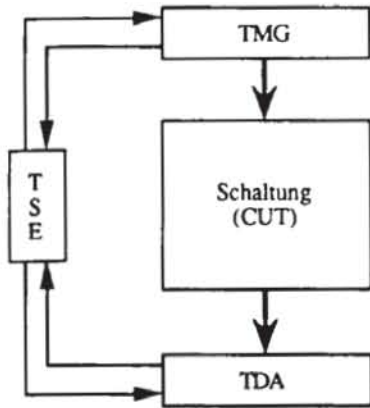


Abb. 6. Selbsttestbare Schaltungen

initialisieren, ihn für den autonomen Betrieb mit den notwendigen Taktsignalen zu versorgen und schließlich vom Chip eine Signatur oder ein Statussignal aufzunehmen, sind wesentlich billigere Testgeräte als beim konventionellen externen Test möglich. Die Schaltung muß für die Erfassung mancher technologieabhängiger Fehler mit sehr hoher Geschwindigkeit betrieben werden, was mit externen Testautomaten nur zu sehr hohen Kosten oder mitunter gar nicht realisiert werden kann. Ein Selbsttest kann dagegen meist mit der üblichen Betriebsgeschwindigkeit der Schaltung durchgeführt werden. Darüber hinaus verkürzen sich beim Selbsttest die für den externen Test wegen der notwendigen seriellen Eingabe der Testmuster in den Prüfpfad typischerweise langen Testdurchführungszeiten drastisch, was sich wiederum positiv auf die Gesamtkosten der Chips auswirkt. Ein weiterer wichtiger Vorteil des Selbsttests ist schließlich, daß die Selbsttesteinrichtungen nicht nur zum Chiptest, sondern auch zum Systemtest und zur Diagnose im Rahmen der Systemwartung herangezogen werden können.

Für den Selbsttest einer Schaltung benötigt man im allgemeinen *Testmustergeneratoren* (TMG) zur Erzeugung der Testmuster, *Testdatenauswerter* (TDA), um die Testantworten zu komprimieren, und eine *Teststeuereinheit* (TSE), die den Testablauf überwacht (Abb. 6).

Die Testantworten werden heute meist durch Signaturanalyse mit linear rückgekoppelten Schieberegistern ausgewertet [10, 30]. Dieses Verfahren hat sich anderen Ansätzen hinsichtlich der Fehlermaskierung und des Flächenbedarfs überlegen gezeigt. Wichtige Unterschiede gibt es jedoch bei der Mustererzeugung. Die Testmuster können entweder vorher bestimmt und in einem Speicherfeld in der Schaltung abgelegt sein oder von der

Schaltung selbst mit Hilfe multifunktionaler Testregister erzeugt werden.

Ein elementarer Ansatz des Selbsttests besteht darin, die Testmuster in einem auf dem IC integrierten ROM zu speichern. Mikroprozessoren und Mikrocomputer, die ohnehin Speicherstrukturen enthalten, sind hierfür besonders geeignet. Für anwendungsspezifische VLSI-Schaltungen (ASICs) ist dieses Vorgehen wegen des beträchtlichen Mehraufwandes an Hardware in der Regel weniger günstig, und man verwendet eher multifunktionale Testregister nach Abb. 2. Mit derartigen Registern werden gleichverteilte und ungleichverteilte Zufallsmuster sowie deterministische und pseudo-erschöpfende Testmengen erzeugt.

Für den Entwerfer einer Schaltung steht die Systemfunktion im Mittelpunkt, und es ist sinnvoll, ihn von der Implementierung des Selbsttests zu entlasten. Dafür sind neue Werkzeuge erforderlich, die den Einbau der Selbsttestkomponenten automatisieren und die Vorteile des Selbsttests damit effizienter und leichter nutzbar machen. In der Regel ist es nicht nötig, daß alle Register Muster erzeugen und Testdaten auswerten können. Natürlich sollten die Testregister automatisch ausgewählt und dabei besonderes Gewicht auf die Minimierung des Mehraufwandes für die Selbsttesteinrichtungen gelegt werden. Dies kann dadurch erreicht werden, daß die einzelnen Testmustergeneratoren und Testdatenauswerter für möglichst viele Moduln der Schaltung genutzt werden. Da der Selbsttest einer modular aufgebauten Schaltung typischerweise aus mehreren Testsitzungen besteht und daher im Hinblick auf die Testdurchführungszeit der Selbsttest dieser Moduln weitestgehend parallel durchgeführt werden sollte, sind geeignete Testplanungsverfahren notwendig. Um alle Vorteile des Selbsttests ausschöpfen zu können, muß die zugehörige Teststeuerung als Steuerwerk implementiert und auf dem Chip integriert werden. Obwohl einige der hier erwähnten Aspekte und Möglichkeiten des Selbsttests noch Gegenstand der Forschung sind, finden Selbsttestverfahren wegen ihres großen Potentials für die Zukunft schon heute ständig wachsende Akzeptanz und immer breiteren Einsatz in der Industrie [14].

6. Synthese testbarer Schaltungen

Um den Entwurfsaufwand zu reduzieren und die Entwicklungszeiten abzukürzen, werden derzeit Verfahren untersucht, die Schaltungsstrukturen aus Verhaltensbeschreibungen automatisch synthetisieren [19]. Der klassische Syntheseansatz zerfällt in zwei Phasen. In der ersten Phase wird eine Schaltung mit der gewünschten Funktionalität synthetisiert, in der zweiten Phase wird diese Struktur um spezielle Schaltungsteile, beispielsweise einen Prüfpfad, erweitert, um die Testbarkeit der Schaltung sicherzustellen. Durch die unzureichende Kopplung der beiden Schritte gehen Optimierungsmöglichkeiten verloren, da einerseits bei Entwurfsentscheidungen die Testbarkeitsaspekte kaum berücksichtigt werden und andererseits die Zusatzausstattung zur Gewährleistung der Testbarkeit im Normalbetrieb ungenutzt bleibt.

In jüngster Zeit wurden innovative Syntheseverfahren und Schaltungsstrukturen erarbeitet, die die Funktionen der beiden oben genannten Phasen eng miteinander koppeln und damit zu effizienteren Realisierungen führen. Dabei wird die funktionale Spezifikation der Schaltung zunächst um eine die zu implementierende Teststrategie beschreibende Testspezifikation ergänzt. Davon ausgehend wird eine Schaltungsstruktur synthetisiert, die von Anfang an sowohl die funktionale Spezifikation als auch die Testspezifikation erfüllt und mit kleinstmöglichem Aufwand realisiert ("Synthesis for Testability", [2, 7]).

Üblicherweise partitioniert man die zu synthetisierende Struktur in Datenpfad und Steuerwerke. Für beide Teilstrukturen ist die Synthese zwei- oder mehrstufiger Schaltnetze aus partiell definierten booleschen Funktionen ein wichtiger Arbeitsschritt. Die dabei verwendeten Programme zur Logikminimierung basieren auf Heuristiken, die nicht notwendigerweise zu einer minimalen Lösung führen. Redundanzen erhöhen nicht nur den Flächenbedarf, sondern erschweren auch die fundierte Beurteilung der Qualität eines Testsatzes, da Fehler in redundanten Schaltungsteilen prinzipiell nicht erkennbar sind. Daher sucht man Syntheseverfahren, die automatisch redundanzfreie Schaltnetze generieren, oder verwendet die beschriebenen Algorithmen zur deterministischen Testsatzerzeugung, um redundante Schaltungsteile zu identifizieren und zu eliminieren. Dies ist auch wichtig für den Fall, daß der eigentliche Test nicht mit deterministisch erzeugten Mustern, sondern beispielsweise im Selbsttest durchgeführt wird.

Wegen der unregelmäßigen und stark vermaschten inneren Struktur von Steuerwerken und der daraus resultierenden großen sequentiellen Tiefe sind hier die üblichen für Datenpfade entwickelten Testverfahren nur sehr eingeschränkt anwendbar. Konventionelle Ansätze führen regelmäßig zu erheblichem Flächenmehraufwand bei reduzierter Betriebsgeschwindigkeit. Da es bereits ausgereifte Verfahren zur Synthese von Steuerwerken gibt (z. B. [19, 16]), die aber auf die Randbedingungen des Tests keine Rücksicht nehmen, ist die Synthese effizient realisierbarer und testbarer Steuerwerke ein besonderer Schwerpunkt aktueller Arbeiten auf diesem Gebiet.

Auch wenn der kombinatorische Teil einer Schaltung für sich betrachtet keine Redundanzen enthält, können sequentielle Schaltungen weitere prinzipiell nicht testbare Fehler enthalten. In [6] wird eine abschließende Darstellung von Arbeiten gegeben, in denen Verfahren für die Synthese irredundanter sequentieller Schaltungen erarbeitet wurden. In [2] wird darüber hinaus eine Möglichkeit vorgestellt, auch Prüfpfade in die funktionale Spezifikation der Steuerwerke einzubeziehen.

Besondere Probleme bereitet die flächeneffiziente Realisierung selbsttestbarer Steuerwerke. Allerdings kann die Funktionalität der Testregister dazu genutzt werden, den Aufwand zur Realisierung der Systemfunktion zu verringern. Derzeit werden Syntheseverfahren entwickelt, die beliebige Selbsttestregister wie linear oder nichtlinear rückgekoppelte Schieberegister oder zellulare Automaten in die Implementierung der Systemfunktion einbeziehen [7].

7. Systemtest

In der Regel besteht ein digitales System aus vielen hochintegrierten Schaltungen, die zu Baugruppen zusammengefaßt werden. Für die Baugruppen ist ebenfalls ein Test erforderlich, der jedoch wesentlich komplizierter, zeitaufwendiger und kostenintensiver als der Chiptest ist, wenn die verwendeten Schaltungen keine Testhilfen enthalten oder diese nicht aufeinander abgestimmt sind. Da meist Chips von verschiedenen Herstellern verwendet werden, ist eine Standardisierung der Testhilfen notwendig. Aus diesem Grund haben sich im November 1985 auf Initiative der Firma Philips zahlreiche große Halbleiter-Hersteller und -Anwender aus Europa und Nordamerika zur Joint Test Action Group (JTAG) zusammengeschlossen, um eine verbindliche Testschnittstelle der Chips festzulegen. Eine entsprechende Norm wurde vom US-amerikanischen Verband der Elektroingenieure IEEE inzwischen als "IEEE 1149.1 Standard Test Access Port and Boundary-Scan Architecture" verabschiedet [13].

Eine Baugruppe kann aus mehreren hundert hochintegrierten Schaltungen bestehen, die auf eine Leiterplatte montiert sind. Der Test der Baugruppe erfordert üblicherweise die Lösung der drei folgenden Teilaufgaben:

- a) Überprüfung der Funktion jedes Moduls;
- b) Überprüfung der Verbindungen zwischen den Modulen;
- c) Überprüfung der Gesamtfunktion der Schaltung.

Die Teilaufgabe c) kann in der Regel nicht vollständig gelöst werden, so daß man sich auf einen kurzen Funktionstest beschränkt und annimmt, daß dies zusammen mit der erfolgreichen Durchführung von a) und b) zu einem Test ausreichender Qualität führt.

Die Teilaufgaben a) und b) können mit einem Funktionstest ausschließlich von der Board-Peripherie nur unzureichend und mit sehr hohem Rechenaufwand behandelt werden. Daher verwendet man hierfür meist einen „In-Circuit“-Test, bei dem mittels Testnadeln Signale an die Chipanschlüsse und Leiterbahnen gebracht und von diesen abgegriffen werden können. Bei gedruckten Leiterplatten werden die Chips nur auf einer Seite montiert, die Chipanschlüsse durchkontaktiert und die Leiterbahnen auf der Rückseite weitergeführt. Legt man daher eine derartige Baugruppe mit der Rückseite auf ein sogenanntes Nadelbett, ist jeder Anschluß unmittelbar zugänglich. Die Beobachtung eines Signals auf einer Leitung mit einer Testnadel bereitet kaum Schwierigkeiten. Will man jedoch ein neues Signal auf eine Leitung setzen, muß das vorhandene Signal überschrieben werden (*Backdriving*). Hierbei kann aber der treibende Baustein auf der Leiterplatte im Test zerstört werden. Man ist daher bestrebt, diese Technik nur selten anzuwenden.

Derzeit werden Baugruppen in zunehmendem Maße in SMT (Surface Mount Technology) montiert, bei der die Leiterbahnen innerhalb der Platine verlaufen, die z. B. aus Keramik besteht. Die Leiterbahnen verlaufen auf mehreren Ebenen, damit auch bei starker Miniaturisierung die gesamte Baugruppe verdrahtet werden kann. Auch wegen der steigenden Zahl von Anschlüssen pro Chip, die nicht mehr nur die Seiten des Gehäuses, sondern

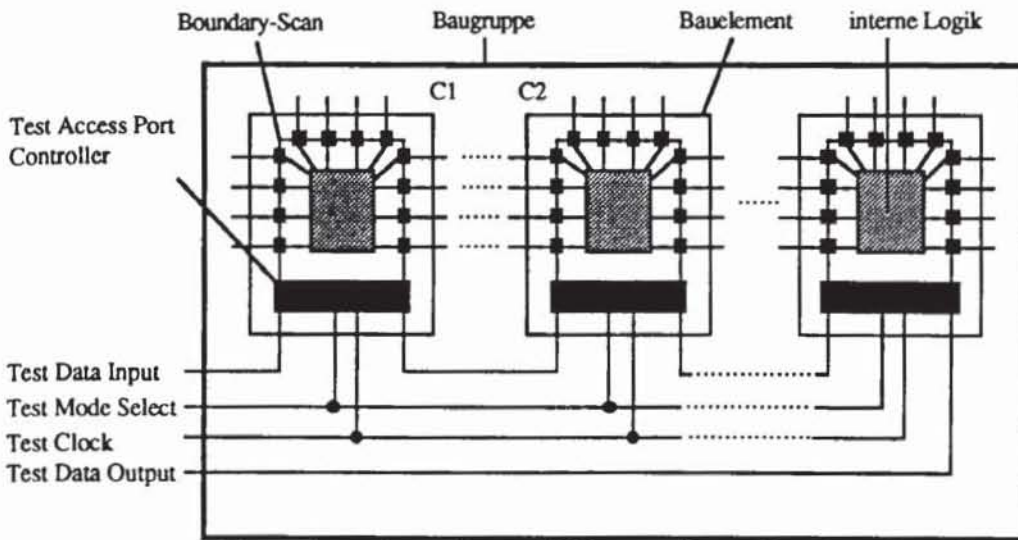


Abb. 7. Prinzip des "Boundary-Scan"

die gesamte Unterseite in Anspruch nehmen können, sind Mehrlagenverdrahtungen unerlässlich. Dadurch sind zahlreiche Chipanschlüsse für die Testnadeln nicht mehr zugänglich. Da auf SMT-Baugruppen die Chips sehr dicht mit einem Abstand bis zu 0,1 mm gepackt sind, läßt sich eine Meßspitze auch auf die offenliegenden Pins nur schwer positionieren. Weitere in der Entwicklung befindliche Techniken wie die Montage der Chips auf Silicium schränken die Zugänglichkeit interner Signale noch weiter ein.

Die Zugänglichkeit eines Signals läßt sich erhalten, wenn es zusätzlich an ein Flipflop angeschlossen wird und alle Flipflops in einen globalen Prüfpfad (Boundary-Scan) durch die gesamte Baugruppe integriert werden (Abb. 7).

Mit diesem Prüfpfad lassen sich die externen Anschlüsse eines in der Baugruppe montierten Chips belegen und die Antworten des Chips nach außen schieben. Zugleich kann der Prüfpfad auch für den Test der Verbindungen zwischen den Chips genutzt werden.

Damit ein einheitlicher Prüfpfad durch die Schaltungen unterschiedlicher Hersteller gelegt werden kann, definiert IEEE 1149.1 eine genormte Schnittstelle als Test Access Port (TAP) zu den Testhilfen auf den einzelnen Chips. Sie besteht aus einem Steuerwerk (TAP Controller) und vier externen Anschlüssen TDI, TMS, TCK und TDO. An TCK (Test Clock) wird der Takt angeschlossen, der den Testbetrieb steuert, mit TDI (Test Data Input) werden die Daten aufgenommen, die in eines der Register geschoben werden sollen, und TDO (Test Data Output) gibt die Registerinhalte wieder aus. TMS (Test Mode Select) schließlich steuert den TAP Controller in einem seiner sechzehn Zustände. Einer der Zustände steht zur Verfügung, um über TDI Befehle zu laden und dadurch die Betriebsweise der Testeinrichtung festzulegen. Es müssen mindestens die folgenden drei Betriebsarten möglich sein:

1) *Externer Test*: Dies ist die wichtigste Betriebsweise beim Baugruppentest. Der Prüfpfad wird verwendet, um die externen Verbindungen zwischen den Chips zu prüfen. Dazu wird beispielsweise das Boundary-Scan-Regi-

ster des Chips C1 (Abb. 7) seriell geladen, so daß an den Ausgängen von C1 das entsprechende Muster anliegt und in das Boundary-Scan-Register von C2 übernommen werden kann, falls die Verbindungsleitungen korrekt sind. Der Inhalt des Boundary-Scan-Registers von C2 wird schließlich wieder herausgeschoben.

2) *Interner Test*: Diese Betriebsweise unterstützt hauptsächlich den Prototypentest und die Diagnose. Das Boundary-Scan-Register wird mit Testmustern für das Chip-Interne geladen, und die entsprechenden Testantworten können ausgelesen werden.

3) *Sample-Test*: Häufig will man während der Systementwicklung wissen, welche Daten im Normalbetrieb an einem Bauelement anliegen. Diese Daten können in dieser Betriebsweise, ohne den Normalbetrieb zu stören, in den Prüfpfad geladen und nach außen gebracht werden.

Optional können noch weitere Betriebsarten unterstützt werden, wie

4) *Run-BIST*: Wenn im Chip ein Selbsttest implementiert ist, kann er in diesem Betriebsmodus ablaufen.

Der Boundary-Scan ist eine genormte On-Chip-Testhilfe für den Baugruppentest, die sowohl den In-Circuit-Test als auch den Funktionstest unterstützt und drastisch erleichtert. Da er den Baugruppentest ohne Testnadeln und mechanische Adaptierung ermöglicht, wird er vielfach als elektronischer Nadeladapter ("Electronic Bed of Nails") bezeichnet. Mit ihm können die hohen Kosten für die mechanisch aufwendigen und deshalb teuren Nadelbetten umgangen und die erwähnten Justierungsprobleme der Leiterplatten auf dem Nadeladapter vermieden werden. Weitere durch den Boundary-Scan erzielbare Vorteile sind u.a. in der Verbesserung der Produktqualität, der wirtschaftlich wichtigen Verkürzung der Zeit bis zur Markteinführung der Baugruppen, der Möglichkeit zum Einsatz automatischer Verfahren zur Prüfvorbereitung und den deutlich besseren Diagnosemöglichkeiten zu sehen. Abschließend sei noch erwähnt, daß die Ausnutzung der Boundary-Scan-Technik bereits effektiv und hilfreich ist, wenn nicht alle Chips der Baugruppe mit Boundary-

Scan ausgestattet sind, was besonders im derzeitigen Einführungsstadium von großer Bedeutung ist.

Literatur

Als einführende Werke zur Übersicht sind insbesondere [1], [10] und [30] geeignet.

1. Abramovici, M., Breuer, M. A., Friedman, A. D.: *Digital Systems Testing and Testable Design*. New York: Computer Science Press 1990
2. Agrawal, V. D., Cheng, K.-T.: An Architecture for Synthesis of Finite State Machines. Proc. European Design Automation Conf., Glasgow 1990, S. 612–615. IEEE Computer Society Press
3. Antreich, K. J., Schulz, M. H.: Accelerated Fault Simulation and Fault Grading in Combinational Circuits. IEEE Trans. Computer-Aided Design CAD-6, 704–712 (1987)
4. Bassett, R. W., Butkus, B. J., Dingle, S. L., Gillis, P. S., Panner, J. H., Petrovick, J. G., Wheeler, D. L.: Low-Cost Testing of High-Density Logic Components. IEEE Design & Test of Computers, April 1990, S. 15–29
5. Bardell, P. H., McAnney, W. H., Savir, J.: *Built-In Test for VLSI Pseudorandom Techniques*. New York: Wiley 1987
6. Devadas, S., Ma, H.-K. T., Newton, A. R., Sangiovanni-Vincentelli, A.: Irredundant Sequential Machines Via Optimal Logic Synthesis. IEEE Trans. Computer-Aided Design CAD-9, 8–18 (1990)
7. Eschermann, B., Wunderlich, H.-J.: Optimized Synthesis of Self-Testable Finite State Machines. Proc. 20th Int. Symp. Fault-Tolerant Computing, New Castle, IEEE Computer Society Press 1990
8. Ferguson, F. J., Shen, J. P.: Extraction and Simulation of Realistic CMOS Faults Using Inductive Fault Analysis. Proc. IEEE Int. Test Conf., Washington 1988, S. 475–484
9. Fujiwara, H., Shiono, T.: On the Acceleration of Test Generation. IEEE Trans. Comput. C-32, 1137–1144 (1983)
10. Gerner, M., Müller, B., Sandweg, G.: *Selbsttest digitaler Schaltungen*. München: Oldenbourg 1990
11. Hellebrand, S., Wunderlich, H.-J.: Automatisierung des Entwurfs vollständig testbarer Schaltungen. Proc. GI – 18. Jahrestagung II, 1988, S. 145–159. Informatik-Fachberichte 188 Berlin – Heidelberg – New York: Springer
12. Hellebrand, S., Wunderlich, H.-J., Haberl, O.: Generating Pseudo-Exhaustive Vectors for External Testing. Proc. IEEE Int. Test Conf., Washington 1990, S. 670–679
13. IEEE Standard Test Access Port and Boundary-Scan Architecture, IEEE Std 1149.1-1990, May 21, 1990
14. Illmann, R.: Built-In Self-Test for the MACROLAN Chip. IEEE Int. Test Conf., Washington 1989, S. 735–744
15. Krishnamurthy, B.: Hierarchical Test Generation: Can AI Help? IEEE Int. Test Conf., Washington 1987, S. 694–700
16. Lin, B., Newton, R.: Synthesis of Multiple Level Logic from Symbolic High-Level Description Language. Proc. VLSI'89, S. 187–196 (G. Musgrave, U. Lauther, eds.). Amsterdam: Elsevier, North-Holland 1990
17. Maly, W.: Realistic Fault Modeling for VLSI Testing. Proc. 24th Design Automation Conf., Miami Beach 1987, S. 173–180
18. McCluskey, E. J., Bozorgi-Nesbat, S.: Design for Autonomous Test. IEEE Trans. Comput. C-30, No. 11 (1981); IEEE Trans. Circ. Syst. CAS-28, No. 11 (1981)
19. Marwedel, P., Rosenstel, W.: Synthese von Register-Transfer Strukturen aus Verhaltensbeschreibungen. Inf.-Spektrum 15, 5–22 (1992) – in diesem Heft
20. Schulz, M. H.: Testmuster-generierung und Fehlersimulation in digitalen Schaltungen mit hoher Komplexität. Informatik-Fachberichte 173, Berlin – Heidelberg – New York: Springer 1988
21. Schulz, M. H., Auth, E.: Advanced Automatic Test Pattern Generation and Redundancy Identification Techniques. Proc. FTCS-18, Tokyo 1988, S. 30–35
22. Schulz, M., et al.: SOCRATES: A Highly Efficient Automatic Test Pattern Generation System. IEEE Trans. Computer-Aided Design CAD-7, 126–137 (1988)
23. Villa, T., Sangiovanni-Vincentelli, A.: NOVA: State Assignment of Finite State Machines for Optimal Two-Level Logic Implementation. IEEE Trans. Computer-Aided Design CAD-9, 905–924 (1990)
24. Waicukauski, J. A., et al.: Fault Simulation for Structured VLSI VLSI Systems Design, Dec 1985, S. 20–32
25. Waicukauski, J. A., Lindbloom, E., et al.: WRP: A Method for Generating Weighted Random Patterns. IBM J. Res. Dev. 33, 149–161 (1989)
26. Williams, T. W., Brown, N. C.: Detect Level as a Function of Fault Coverage. IEEE Trans. Computer-Aided Design C-30, No. 12 (1981)
27. Wunderlich, H.-J.: PROTEST: A Tool for Probabilistic Testability Analysis. Proc. IEEE/ACM 22nd Design Automation Conf., Las Vegas 1985, S. 204–211
28. Wunderlich, H.-J.: Probabilistische Verfahren für den Test hochintegrierter Schaltungen. Informatik-Fachberichte 140 Berlin – Heidelberg – New York: Springer 1987
29. Wunderlich, H.-J.: Multiple Distributions for Biased Random Test Patterns. IEEE Int. Test Conf., Washington 1988, S. 236–244
30. Wunderlich, H.-J.: *Hochintegrierte Schaltungen: Prüfgerechter Entwurf und Test*. Berlin – Heidelberg – New York: Springer 1991
31. Wunderlich, H.-J., Hellebrand, S.: Generating Pattern Sequences for the Pseudo-Exhaustive Test of MOS-Circuits. Proc. IEEE 18th Int. Symp. on Fault-Tolerant Computing, FTCS-18, Tokyo 1988, S. 36–45

Eingegangen 6.8.1990, in überarbeiteter Form 3.5.1991

Prof. Dr. Hans-Joachim Wunderlich
 Universität-GH Siegen
 FB12, Fachgruppe Rechnerstrukturen
 Holderlinstr. 3
 W-5900 Siegen

Michael H. Schulz
 Siemens Nixdorf Informationssysteme AG
 Abt. AP 274
 Postfach 83 09 51
 W-8000 München 83