HALBLEITER-TESTMETHODEN

Funktionale Sicherheit auf **Chip-Ebene**

Halbleiter-Schaltungen (ICs) müssen immer mehr die strengen Sicherheitsanforderungen der Norm ISO 26262 erfüllen. Die Fähigkeit, mögliche latente und transiente Fehler über die gesamte Lebensdauer abzudecken, bestimmt die Gesamtdiagnoseabdeckung der Schaltung, die das erreichbare Niveau der Sicherheitsintegrität (ASIL) bestimmt. Dies stellt Hersteller von ICs und Systemen für die Automobilindustrie zunehmend vor neue Herausforderungen.

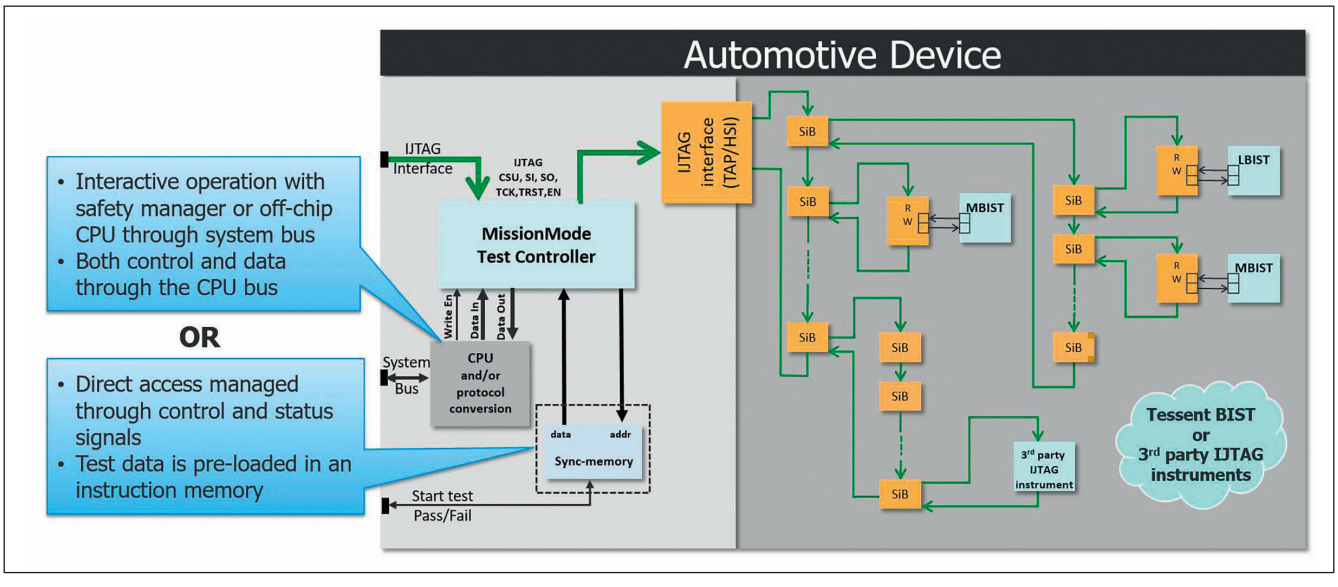

Bild 1: Testarchitektur auf Chipebene für systeminterne Tests. © Mentor, a Siemens Business

Integrierte Schaltungen (ICs) für Fahrzeuge werden zunehmend nach modernsten Verfahren entwickelt und hergestellt. Sie werden nicht mehr nur für einfache Funktionen wie die Steuerung von Fenstern oder der Lichtsignale eingesetzt, sondern jetzt auch für komplexere Funktionen im Zusammenhang mit Fahrerassistenzsystemen genutzt und zunehmend für autonome Fahranwendungen benötigt. Die für diese erweiterten Funktionen erforderliche Verarbeitungsleistung führt dazu, dass sehr große und komplexe ICs benötigt werden, bei denen eine optimale Energieeffizienz wichtig ist. Verbunden mit der Notwendigkeit, dass diese Schaltungen die strengen Sicherheitsanforderungen der Norm ISO 26262 erfüllen, stellt dies die Hersteller von Schaltungen und Systemen für die Automobilindustrie vor neue Herausforderungen. Es sind Lösungen erforderlich, die gewährleisten, dass neue komplexe elektronische Fahrzeugsysteme während der gesamten Lebensdauer des Fahrzeugs jederzeit sicher funktionieren. Dies wird als funktionale Sicherheit bezeichnet.

Die funktionale Sicherheit beruht auf Mechanismen innerhalb der Schaltung, die als Sicherheitsmechanismen bezeichnet werden und die den korrekten funktionalen Betrieb der Schaltung im Einsatz überwachen und überprüfen sollen. Äußerst beliebt ist hier der Ansatz, eine Reihe eingebetteter Überwachungsfunktionen zu verwenden, die über jede Halbleiterschaltung verteilt und über eine globale Kommunikationsinfrastruktur miteinander verbunden sind, was die schnelle Erkennung und Meldung zufälliger Fehler überall im System ermöglicht. Diese Überwachung muss stattfinden, ohne den normalen funktionalen Betrieb zu beeinträchtigen und die Flexibilität aufweisen, je nach Endanwendung des Halbleiters und der zugehörigen ASIL-Klassifizierung einen unterschiedlichen Grad an Fehlerabdeckung bereitzustellen. Ein Beispiel einer Testarchitektur auf Chipebene, die eine verteilte systemweite Überwachung unterstützt, ist in Bild 1 dargestellt.

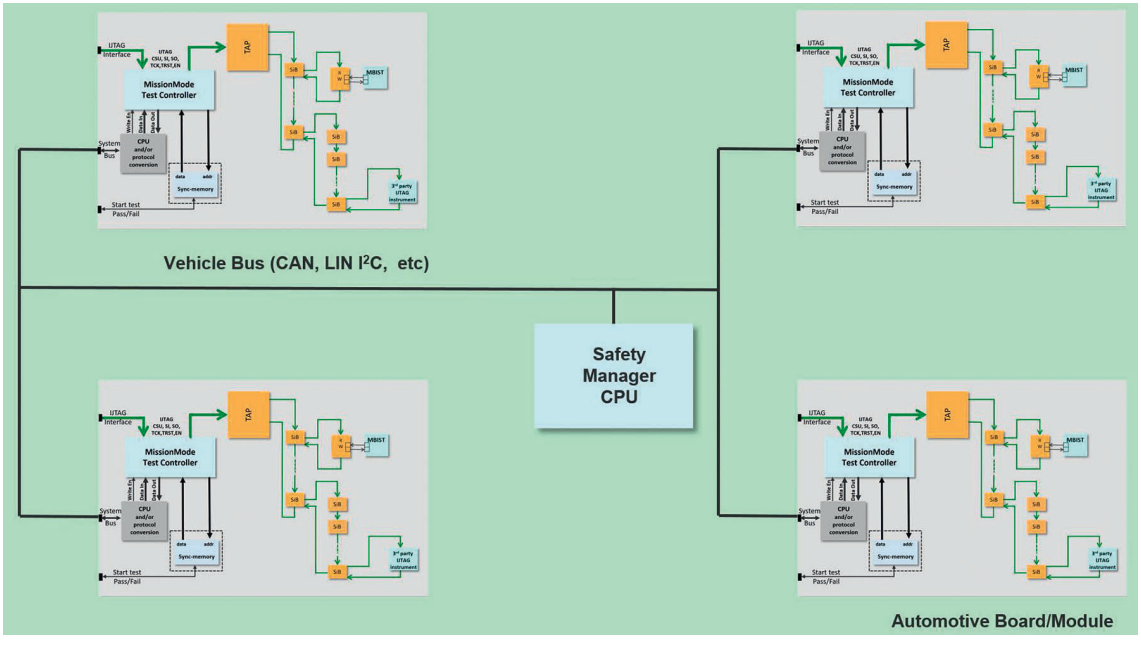

Bild 2: Testarchitektur auf Systemebene. © Mentor, a Siemens Business

Testzugriffsport

Ein standardmäßiger IEEE 1149.1-Testzugriffsport (TAP) bietet eine Schnittstelle zu allen On-Chip-Testressourcen während des Produktionstests. Der TAP stellt eine Verbindung zu einem rekonfigurierbaren seriellen Zugriffsnetzwerk her, das auf dem IEEE 1687-Standard basiert (häufig als IJTAG-Standard bezeichnet). Dieses IJTAG-Netzwerk besteht aus Konfigurations-Elementen, die als Segment Insertion Bits (SIBs) bezeichnet werden. Mit jedem SIB kann ein Teilnetzwerk eingeschaltet oder umgangen werden, wodurch ein optimierter Zugriff auf alle Testressourcen innerhalb des Netzwerks möglich wird. Auf das IJTAG-Netzwerk wird auch von einem IST-Controller (In-System Test) zugegriffen, in diesem Fall dem Tessent MissionMode-Controller. Der MissionMode-Controller kommuniziert über eine CPU/DMA-Schnittstelle entweder mit der Außenwelt oder einem internen Sicherheitsmanager und führt die für die Datenübermittlung zwischen dem CPU/DMA-Bus und dem internen IJTAG-Netzwerk erforderliche Parallel-Seriell- und Seriell-Parallel-Datenkonvertierung durch. Dieser IST-Controller ermöglicht eine Kommunikationsarchitektur auf Systemebene, wie in Bild 2 dargestellt. Ein Serviceprozessor kann über jeden eingesetzten Fahrzeugbus wie CAN oder I^2C auf den IST-Controller jedes Chips und damit auf jede On-Chip-Testressource zugreifen.

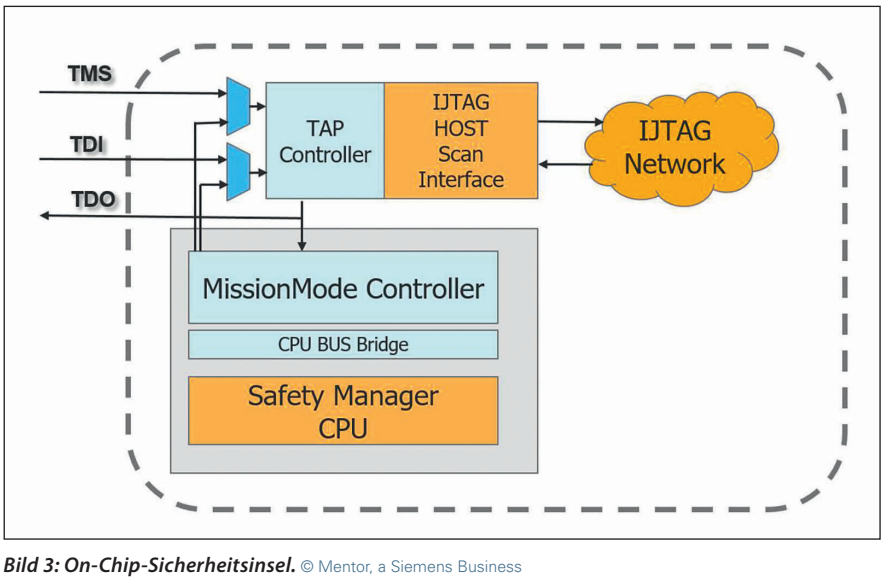

Bild 3: On-Chip-Sicherheitsinsel. © Mentor, a Siemens Business

Sicherheitsinsel

Bei erweiterten Ein-Chip-Systemen kann die Sicherheitsmanager-CPU als Teil der Schaltung eingebettet werden. Diese Architektur wird allgemein als „Sicherheitsinsel" bezeichnet. Wenn man den Sicherheitsmanager in die Sicherheitsinsel integriert, verringert sich die Wahrscheinlichkeit, dass er durch Defekte im funktionalen Teil der Schaltung beeinträchtigt wird. Dies wird als „Insel" bezeichnet, weil sie als separate physische und Leistungspartition auf dem Chip behandelt wird, häufig dedizierte Strom- und Steuersignale empfängt und so weit wie möglich physisch von der funktionalen Logik isoliert ist. Die einzige Datenverbindung zwischen der Sicherheitsinsel und der anderen funktionalen Logik sind die Verbindungen zum Testnetzwerk. Bild 3 zeigt die Hauptteile einer typischen Sicherheitsinsel.

Die Wirksamkeit dieses verteilten Systems auf einer einzelnen Schaltung oder mehreren Schaltungen hängt von den Testressourcen ab, die in den verschiedenen Schaltungen implementiert sind. Um die ISO 26262-Zertifizierung zu erhalten, müssen diese Ressourcen in der Regel eine Mischung aus funktionalen und strukturellen Sicherheitsmechanismen sein. Die wahrscheinlich häufigste Form der On-Chip-Strukturressource ist der Memory Built-In Self-Test (MBIST). Eine MBIST-Engine testet einen eingebetteten Speicher vollständig,

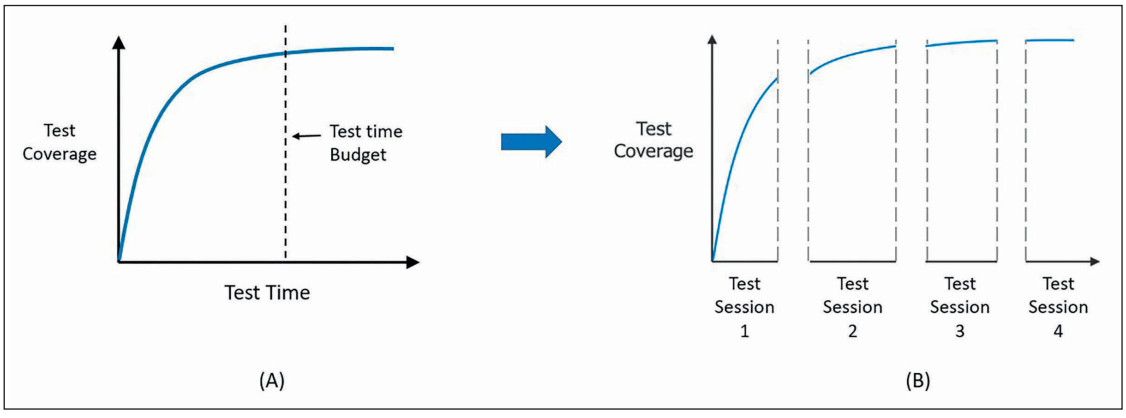

Bild 4: Management der Testzeit des Logic BIST. © Mentor, a Siemens Business

indem sie algorithmisch eine Folge von Lese- und Schreiboperationen erzeugt, die den gesamten Adressraum abdecken.

Memory Built-In Self-Test

Eine große Herausforderung bei der Durchführung eines solchen Speichertests während des Fahrzeugbetriebs besteht darin, dass der Speicher zuerst abgeschaltet werden muss, damit die MBIST-Engine die Kontrolle übernehmen kann. Es kann auch erforderlich sein, den Speicherinhalt vor dem Ausführen des Tests zu sichern und den Inhalt anschließend wiederherzustellen, da der Speichertest den Speicherinhalt vor dem Test löscht. Eine weitere Schwierigkeit besteht darin, dass das Abschalten des Speichers wahrscheinlich auch die Systemleistung beeinträchtigt, was bei einigen Anwendungen möglicherweise nicht akzeptabel ist. Eine nicht-destruktive MBIST-Technik wurde entwickelt, um all diese Probleme zu vermeiden. Bei diesem Ansatz testet die MBIST-Engine den Speicher mithilfe einer Reihe kurzer Transaktionssequenzen, die häufig als Bursts bezeichnet werden. Ein Burst dauert normalerweise nur eine kleine Anzahl von Taktzyklen (möglicherweise 20 bis 30) und richtet sich jedes Mal auf unterschiedliche Teile des Speichers. Der gesamte Speicher wird daher mittels einer großen Anzahl kurzer MBIST-Abläufe getestet. Dieser Ansatz ist nicht-destruktiv, da die durch einen Burst geänderten Speicherteile während jedes Bursts von der MBIST-Engine gespeichert und danach wiederhergestellt werden. Die funktionale Leistung wird hierdurch nicht wesentlich beeinträchtigt, da die Bursts nur ausgelöst werden, wenn die zwischen der MBIST-Engine und der Funktionslogik implementierte Arbitrierungslogik feststellt, dass der Speicher zugreifbar ist.

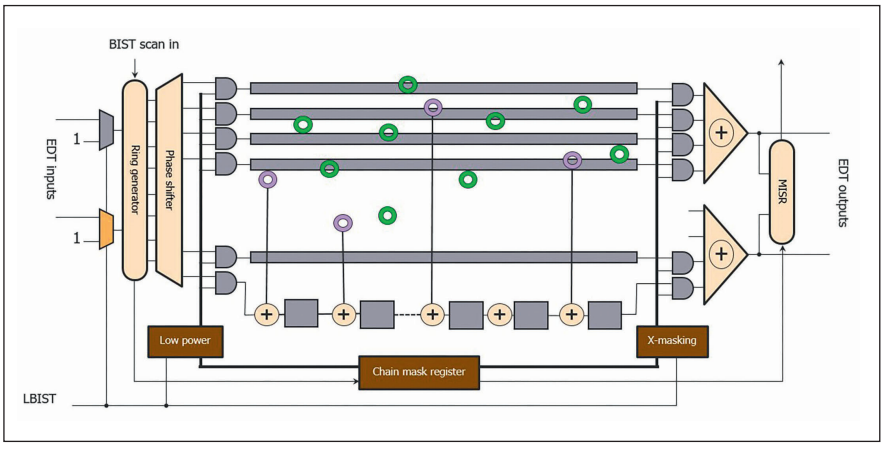

Bild 5: Logic BIST mit Observation Scan-Architektur. © Mentor, a Siemens Business

Logic BIST

Logic BIST ist eine weitere beliebte Form struktureller systeminterner Testressourcen, auf die über den IST-Controller zugegriffen werden kann. Diese Testlösung beruht auf der Erzeugung von Zufallstestmustern auf dem Chip, die auf Scanketten angewendet werden, um den Logikteil eines Chips zu testen. Die Antworten des Chips auf alle Zufallstestmuster werden zu einer Signatur zusammengefasst, die am Ende des Tests untersucht wird und ergibt, ob der Chips den Test bestanden oder nicht bestanden hat. Die Testabdeckung, die durch Anwenden einer zunehmenden Anzahl von Zufallstestmustern erreicht wird, wächst logarithmisch, wie in Bild 4 (A) gezeigt.

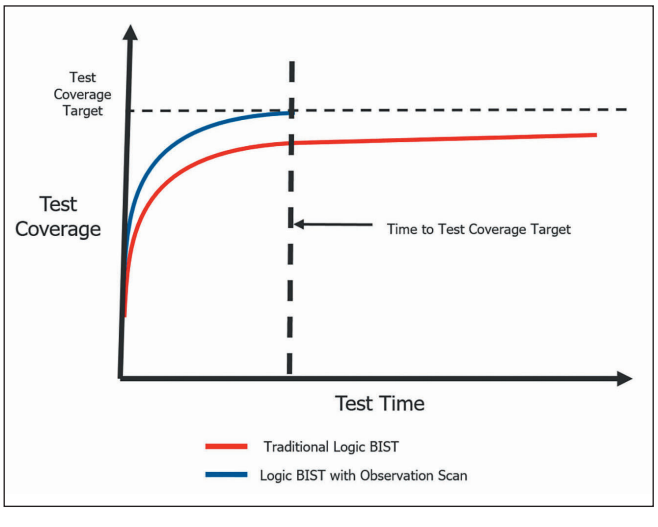

Bild 6: Verbesserung der Testzeit durch LBIST-OST. © Mentor, a Siemens Business

Bei diesem Ansatz stellt sich jedoch häufig das Problem, innerhalb eines bestimmten Zeitbudgets eine ausreichend hohe Testabdeckung zu erreichen. Eine Lösung für dieses Problem besteht darin, den Test in mehrere Abläufe zu unterteilen, wie in Bild 4 (B) dargestellt. Jeder aufeinanderfolgende Testablauf wird während einer verfügbaren Unterbrechung des funktionalen Betriebs durchgeführt. Beispielsweise könnte in einem Bildprozessor, der zum Verarbeiten visueller Daten verwendet wird, jeder Testablauf zwischen der Verarbeitung der einzelnen Bilder durchgeführt werden. Das Management dieser mehreren Teiltests erfordert eine sorgfältige Koordination zwischen dem IST-Controller und der Logic BIST-Engine. Der IST-Controller muss verfolgen, welcher Testabschnitt als Nächstes angewendet werden soll, die Logic BIST-Engine initialisieren, damit sie den richtigen Satz von Zufallstestmustern erzeugt, und dann die Zwischensignatur abrufen und vergleichen, um festzustellen, ob der Test bestanden wurde oder nicht.

Es gibt jedoch Fälle, in denen diese Art der Verteilung entweder nicht möglich ist oder immer noch nicht die erforderliche Abdeckung im FTTI (Fault Tolerant Time Interval) bietet. Hier ist eine neue Technologie nützlich, die die Logic BIST Testzeit erheblich verkürzt und so die Gesamtreaktionszeit erheblich verbessert. Dieser Logic BIST mit Observation Scan-Technologie (OST) verwendet spezielle Testpunkte, die in das Design eingefügt wurden, sowie eine kleine dedizierte Scan-Kette von Beobachtungs-Scan-Zellen, mit denen die Fehlerabdeckung der Funktionslogik bei jedem Zyklus effektiv erfasst werden kann, anstatt nur bei dem Erfassungszyklus jedes Musters. Dies wird in Bild 5 dargestellt. Das Ergebnis ist eine viel schnellere Testabdeckung der funktionalen Logik, so dass diese Sicherheitsmechanismen ihre erforderlichen Qualitätsziele schneller als bei einem herkömmlichen Logic BIST erreichen können. Bild 6 zeigt den Vergleich zwischen LBIST-OST und herkömmlichem Logic BIST.

Fazit

Alle hier beschriebenen Technologien und Methoden ermöglichen die Implementierung einer beliebigen Anzahl sicherheitsrelevanter Funktionen auf Systemebene. Alle Testfehler können an den Sicherheitsmanager zurückgemeldet werden, um Korrekturmaßnahmen durchzuführen. Auch bei IST-Steuerungen können Fehler vom Sicherheitsmanager überwacht und Maßnahmen ergriffen werden, die vom Deaktivieren bestimmter Funktionen bis zum Versetzen des Fahrzeugs in einen sicheren Betriebszustand reichen können. Da es von entscheidender Bedeutung ist, schnell Korrekturmaßnahmen zu ergreifen, ist die Reaktionszeit dieser Sicherheitsmechanismen so wichtig. ■ (oe)

www.mentor.com

Lee Harrison ist Automotive IC Test Solutions Manager in der Tessent-Gruppe bei Mentor, einem Siemens Unternehmen.