



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 602 08 442 T2** 2006.08.31

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 1 459 080 B1**

(51) Int Cl.⁸: **G01R 31/3185** (2006.01)

(21) Deutsches Aktenzeichen: **602 08 442.3**

(86) PCT-Aktenzeichen: **PCT/IB02/04661**

(96) Europäisches Aktenzeichen: **02 779 830.5**

(87) PCT-Veröffentlichungs-Nr.: **WO 2003/046724**

(86) PCT-Anmeldetag: **05.11.2002**

(87) Veröffentlichungstag

der PCT-Anmeldung: **05.06.2003**

(97) Erstveröffentlichung durch das EPA: **22.09.2004**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **28.12.2005**

(47) Veröffentlichungstag im Patentblatt: **31.08.2006**

(30) Unionspriorität:

997784 **30.11.2001** **US**

(84) Benannte Vertragsstaaten:

**AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB,
GR, IE, IT, LI, LU, MC, NL, PT, SE, SK, TR**

(73) Patentinhaber:

**Koninklijke Philips Electronics N.V., Eindhoven,
NL**

(72) Erfinder:

GOFF, C., Lonnie, NL-5656 AA Eindhoven, NL

(74) Vertreter:

Volmer, G., Dipl.-Ing., Pat.-Anw., 52066 Aachen

(54) Bezeichnung: **TOPOLOGIEREKONFIGURATION EINER PRÜFSCHALTUNG UND VERWENDUNGSVERFAHREN**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung

[0001] Die vorliegende Erfindung bezieht sich auf elektronische Anordnungen und bezieht sich besonders, aber nicht ausschließlich, auf die Verwendung von Testschaltungen zum Debuggen von Operationen und die Fähigkeit, Testschaltungstopologie zu rekonfigurieren.

[0002] Testen von elektronischen Leiterplatten und zugehörigen Komponentenverbindungen kann mit einem sogenannten „Nagelbrett“-Tester durchgeführt werden. Typischerweise enthalten Tester dieser Art Sonden, um ausgewählte elektrische Knoten zu kontaktieren, und erzeugen Testsignale, um richtigen elektrischen Durchgang und Isolation zu verifizieren. Mit dem Aufkommen von Leiterplatten mit mehrfachen leitfähigen Schichten und komplexeren integrierten Schaltungsanordnungen haben sich Nagelbretttester oft als unzureichend erwiesen.

[0003] Eine Bemühung, besseres Testen von Schaltungen zu schaffen, resultierte in der Schaffung des IEEE/ANSI-Standards 1149.1 der 'Joint Test Action Group' (JTAG). Wie hierin verwendet, verweist „JTAG-Standard“ auf die neueste Überarbeitung des am oder vor dem 1. November 2001 gültigen IEEE/ANSI-Standards 1149.1. Der JTAG-Standard beruht auf einem Testbus, der mindestens vier Signalleitungen enthält: eine Testdatenausgangs-(TDO)-Leitung, eine Testdateneingangs-(TDI)-Leitung, eine Testzustandsselektions-(TMS)-Leitung und eine Testtakt-(TCK)-Leitung. Der JTAG-Testbus kann optional eine fünfte Leitung für Testlogikreset (TRST) enthalten. Der JTAG-Testbus ist angeordnet, um mit in der zu testenden Schaltung eingebetteten JTAG-normkonformen Testzugriffspunkten (TAPs) zu kommunizieren. Mehrere TAPs können in einer vorgegebenen zu testenden Anordnung verwendet werden, wobei jeder ein zugehöriges TDO-Interface und ein TDI-Interface hat. Das TDO-Interface eines TAPs kann mit dem TDI-Interface eines anderen TAPs verbunden werden, um eine seriell verbundene Kette zu bilden. Testeinrichtungen können mit dem Testbus verbunden werden, um die zu testende Anordnung zu prüfen, wobei die TDO-Busleitung mit dem nicht angeschlossenen TDI-Interface des TAPs an einem Ende der Kette verbunden ist und die TDI-Busleitung mit dem TDO-Interface am anderen Ende der Kette verbunden ist, um einen seriell miteinander verbundenen Kommunikationsring zu bilden.

[0004] Im Gegensatz dazu sind die TMS- und TCK-Busleitungen zu jedem der TAPs parallel bereitgestellt. Der JTAG-Standard definiert eine Maschine endlicher Zustände, die mit dem Testtaktsignal TCK synchronisiert ist. Die verschiedenen Zustände dieser Maschine ermöglichen die Kommunikation von Daten oder Befehlen zu den TAPs durch die Kette, sodass eine Boundary-Scan-Operation durchgeführt werden kann. Die Sequenz von JTAG-Zuständen wird als Funktion des Zustands von TMS relativ zu der Anzahl der Zyklen, die TCK ausgeführt hat, bereitgestellt.

[0005] Mit dem fortgesetzten Anwachsen in der Schaltungskomplexität, inklusive, aber nicht darauf beschränkt, der Entwicklung der „System auf einem Chip“- (SoC)-Technologie ist es wünschenswert geworden, viele TAPs in einer Schaltungsanordnung und möglicherweise in derselben integrierte Schaltung einzubeziehen. Tatsächlich enthalten SoC-Anordnungen typischerweise Speicher, einen oder mehr Prozessoren und andere Logik, die wünschenswerterweise getestet werden sollten. Für diese komplexen Technologien kann die Fähigkeit, Schaltkreise auszusondern, die nur zu einem TAP gehören, oder weniger als alle verfügbaren TAPs seriell zu verknüpfen, nützlich in der Leistungsfähigkeit verschiedener Debug-Operationen sein. Um Chipplatz einzusparen und/oder die Notwendigkeit Debug-bestimmter Verbindungen auszuschließen, gibt es oft eine Vorliebe, Debug-Operationen mit existierender Testarchitektur wie den JTAG-Bus und TAPs durchzuführen.

[0006] Ein Versuch, existierende Testarchitektur für den JTAG-Standard zu verwenden, ist auf dem JTAG-Bypass-Befehl begründet. Während dieser Befehl verwendet werden kann, einen selektierten TAP zu isolieren, tut er das durch Behindern der Wiederherstellung von Befehlsregistern in anderen TAPs. In 'Proceedings of the ITC conference 1997', Veröffentlichung 3.3, Seite 69–78 mit dem Titel: 'An IEEE 1149.1 based test access architecture for ICs with embedded cores' von Lee Whetsel ist eine andere Lösung offenbart. Ein IC hat eine Anzahl von Kernen, wobei jeder seinen eigenen TAP-Controller hat. Die TAP-Controller sind ein TAP-Verbindungsmodul gekoppelt und sind konfiguriert, dedizierte Selektionssignale an das TAP-Verbindungsmodul zu kommunizieren, während das TAP-Verbindungsmodul angeordnet ist, die TAPs mit Freigabe- und Verbindungsbefehlen zu versorgen, alles unter Kontrolle des Testtaktes und des Testzustandsselektionssignals. Das TAP-Verbindungsmodul wird von den TAPs als ein geteiltes Register verwendet, durch welches TAPs durch andere TAPs freigegeben und/oder deaktiviert werden können.

[0007] In noch einem anderen Versuch wurde ein TAP-Verbindungsmodul vorgeschlagen, das selektiv untergeordnete TAPs adressiert. Leider erfordert dieses Modul-Verbindungsschema, dass von der Host-Einrichtung

Adressinformation an die Standard-JTAG-Kommunikationsdatenpakete angehängt wird, was Rückwärtskompatibilität verhindert. Als eine Konsequenz wurde ein zusätzlicher I/O-Pin zum Überschreiben des Verbindungsmoduls vorgeschlagen. Solche Modifikationen sind in vielen Fällen problematisch.

[0008] Also gibt es Bedarf für weitere Beiträge in diesem Technologiebereich. Die vorliegende Erfindung widmet sich diesem Bedarf mit einem System gemäß Anspruch 1 und einem Verfahren gemäß Anspruch 7.

[0009] Eine Ausführungsform der vorliegenden Erfindung ist eine eindeutige Technik zum Verwenden von Testschaltungen. Andere Ausführungsformen der vorliegenden Erfindung enthalten eindeutige Anordnungen, Verfahren, Systeme und Geräte, um Schaltungen zu testen und/oder Anordnungsfunktionieren zu debuggen.

[0010] Eine weitere Ausführungsform der vorliegenden Erfindung enthält Anhalten des Testtakts, um das Arbeiten mehrerer Testports auszusetzen; einen Schattencontroller zu betreiben während die Testports ausgesetzt sind; und das Arbeiten von einem oder mehreren der Testports nach dem genannten Betreiben wieder aufzunehmen.

[0011] In einer anderen Ausführungsform der vorliegenden Erfindung enthält ein Gerät einen Testbus, mehrere Testports, von denen jeder einen Controller enthält, und einen zwischen den Testbus und die Testports gekoppelten Schattencontroller. Die Testports sind alle während eines ersten Betriebszustands gemäß einem festgelegten Teststandard betriebsfähig. Der Schattencontroller enthält eine Speicheranordnung, um Information, die einer Selektion eines oder mehrerer der Testports während eines zweiten Betriebszustands entspricht, zu speichern. Diese Testports sind während des zweiten Betriebszustands inaktiv. Der eine oder mehrere der mit der Information selektierten Testports arbeiten bis zur Rückkehr zu dem ersten Betriebszustand gemäß dem festgelegten "Teststandard".

[0012] Noch eine andere Ausführungsform enthält eine prozessorlesbare Einrichtung, die mit Prozessorbefehlen codiert ist, die ausführbar sind, um einen Boundary-Scan-Test durch eine Scan-Kette mehrerer Testzugriffports gemäß dem festgelegten Testbusprotokoll durchzuführen. Die Befehle sind außerdem ausführbar, um ein Testtaktsignal auszusetzen, Information in einen Schattencontroller zu speichern, um eine unterschiedliche Konfiguration der Testzugriffports zu definieren und die unterschiedliche Konfiguration gemäß dem festgelegten Testbusprotokoll zu betreiben.

[0013] Noch eine weitere Ausführungsform der vorliegenden Erfindung enthält: Bereitstellen einer Anordnung, die auf ein Testtaktsignal reagiert und die eine Anzahl von Testports enthält, die in einer ersten Konfiguration angeordnet sind; Aussetzen des Arbeitens dieser Testports; Definieren einer zweiten Konfiguration der Testports während des Aussetzens; und Betreiben der Anordnung gemäß der zweiten Konfiguration.

[0014] Eine Aufgabe der vorliegenden Erfindung ist es, eine eindeutige Technik zu schaffen, um Testschaltungen zu verwenden.

[0015] Eine andere Aufgabe der vorliegenden Erfindung ist es, eine eindeutige Anordnung, Verfahren, System und Gerät zu schaffen, um Schaltungen zu testen und/oder Anordnungsfunktionieren zu debuggen.

[0016] Weitere Aufgaben, Ausführungsformen, Formen, Eigenschaften, Nutzen und Vorteile der vorliegenden Erfindung werden aus der Beschreibung und der hier enthaltenen Zeichnung deutlich.

[0017] Es zeigen:

[0018] [Fig. 1](#) eine schematische Darstellung eines Systems gemäß einer Ausführungsform der vorliegenden Erfindung;

[0019] [Fig. 2](#) eine schematische Darstellung eines SoCs des Systems von [Fig. 1](#), die weitere Details zeigt;

[0020] [Fig. 3](#) ein Flussdiagramm, das eine Routine einer Ausführungsform der vorliegenden Erfindung veranschaulicht, die mit dem System von [Fig. 1](#) ausgeführt werden kann; und

[0021] [Fig. 4–Fig. 6](#) Timing-Diagramme, die sich auf verschiedene Operationen der Routine von [Fig. 3](#) beziehen.

[0022] Während die vorliegende Erfindung auf viele verschiedenen Arten ausgeführt werden kann, wird zum

Zweck, das Verstehen der Prinzipien der Erfindung zu fördern, auf die Ausführungsformen, die in der Zeichnung gezeigt sind, Bezug genommen und eine spezifische Sprache benutzt, um dieselbe zu beschreiben.

[0023] [Fig. 1](#) stellt System **20** einer Ausführungsform der vorliegenden Erfindung dar. System **20** enthält Schaltungsanordnung **22**. Schaltungsanordnung **22** enthält Leiterplatte **24**, die eine Anzahl von integrierten Schaltungskomponenten **30** trägt. Komponenten **30** enthalten „System auf einem Chip“- (SoC)-Anordnung **32**, SoC-Anordnung **34** und Höchstintegrations-(VLSI)-Schaltungsanordnung **36**. Komponenten **30** können eine oder mehr andere Anordnungen (nicht gezeigt) enthalten oder können andere als die gezeigten Anordnungen enthalten, wie es Fachleuten einfallen würde.

[0024] Komponenten **30** sind operativ an den Bus **40** gekoppelt. Testbus **40** enthält parallele Testtaktbusleitung **42**, Testzustandsselektionsleitung **44** und Testlogikresetbusleitung **46**, entsprechend der TCK-, TMS- beziehungsweise TRST-Signale des JTAG-Standards. Testbus **40** enthält auch die serielle Scankette **48**. Kette **48** enthält die Testdatenausgangs-(TDO)-Busleitung **48a**, die am Testdateneingang (I/P) **32a** der SoC-Anordnung **32** bereitgestellt ist. Kette **48** enthält auch die serielle Testdatenleitung **48b**, die Testdatenausgang (O/P) **32b** von SoC-Anordnung **32** mit Testdateneingang **34a** der SoC-Anordnung **34** verbindet, und serielle Testdatenleitung **48c**, die Testdatenausgang **34b** der SoC-Anordnung **34** mit Testdateneingang **36a** der Schaltungsanordnung **36** verbindet. Kette **48** enthält auch Testdateneingang-(TDI)-Busleitung **48d**, die mit Testdatenausgang **36b** verbunden ist.

[0025] Testbusleitungen **42**, **44**, **46**, **48a** und **48d** sind mit Busmaster-Einrichtung **50** verbunden. Busmaster-Einrichtung **50** enthält eine oder mehr Bediener-Eingabe-(I/P)-Vorrichtung(en) **50a** und eine oder mehr Bediener-Ausgabe-(O/P)-Vorrichtung(en) **50b**. Eingabevorrichtung(en) **50a** kann/können eine konventionelle Maus, Tastatur, Trackball, Lichtgriffel, Spracherkennungssystem und/oder verschiedene Eingabevorrichtungstypen, wie es Fachleuten einfallen würde, enthalten. Ausgabevorrichtung(en) **50b** kann/können ein konventionelles Grafikdisplay wie ein Farb- oder Nicht-Farb-Plasma, Kathodenstrahlröhre (CRT) oder Flüssigkristallanzeige-(LCD)-Typ; Farb- oder Nicht-Farbdrucker; akustisches Ausgabesystem und/oder verschiedene Ausgabevorrichtungstypen, wie es Fachleuten einfallen würde, enthalten. Außerdem können in anderen Ausführungsformen mehr oder weniger Bedieneringabevorrichtung(en) **50a** oder Bedienerausgabevorrichtung(en) **50b** verwendet werden.

[0026] Einrichtung **50** enthält auch einen oder mehrere Prozessor(en) **52**, der/die operativ an Eingabevorrichtung(en) **50a**, Ausgabevorrichtung(en) **50b** und Speicher **54** gekoppelt ist/sind. Prozessor(en) **52** ist/sind von einer programmierbaren Art, die Softwareprogrammierungsbefehle ausführen kann, die in mindestens einem Teil von Speicher **54** gespeichert sind. Einrichtung **50** enthält ein oder mehr Programme, um den Testbus **40** als Busmaster unter JTAG-Standard zu steuern, und Signalemulation zum Testen gemäß dem JTAG-Standard zu liefern. Diese Programmierung sorgt außerdem für die Ausführung von Routine **120**, die ein verstecktes, Nicht-JTAG-Protokoll enthält, wie später in Verbindung mit [Fig. 3](#) ausführlicher beschrieben wird.

[0027] Prozessor(en) **52** kann/können eine oder mehrere integrierte Schaltungskomponenten, die eine oder mehrere zentrale Verarbeitungseinheiten (CPUs) definieren, umfassen. Prozessor(en) **52** kann/können zugehörige unterstützende Logik, analoge Anordnungen und Stromversorgungseinheiten enthalten. In einer Form beruht/beruhen Prozessor(en) **52** auf mindestens einem Mikroprozessor aus einer Standardvielfalt. Ausführungsformen, die mehrfache Prozessoren **52** enthalten, können in einer gemeinsamen Verarbeitungsanordnung bereitgestellt werden, die angeordnet ist, für eine verteilte Verarbeitung durch mehrfache Einheiten an verschiedenen Orten, durch ein Computernetzwerk und/oder auf eine andere Art verbunden, wie es Fachleuten einfallen würde, zu sorgen. In anderen Ausführungsformen ist ein Teil oder die Gesamtheit der durch Prozessor(en) **52** ausgeführten Logik durch Firmware; dedizierte digitale und/oder analoge Schaltungen; ausgetauscht; und/oder es werden solche verschiedenen Techniken verwendet, wie es Fachleuten einfallen würde.

[0028] Mindestens ein Teil von Speicher **54** umfasst eine auswechselbare Speicheranordnung (RMD; Removable Memory Device) **56**. RMD **56** ist vom nichtflüchtigen Typ, das kann eine Speicherplatte **58** einschließen, wie schematisch in [Fig. 1](#) dargestellt. Speicherplatte **58** kann magnetischer Art sein, so wie eine Floppydisk oder austauschbare Festplatte, oder ein optisch lesbarer Typ, so wie eine Compact Disk (CD) oder Digitale Video Disk (DVD), um nur ein paar Beispiele zu nennen. In anderen Ausführungsformen enthält RMD **56** alternativ oder zusätzlich andere bewegliche Speicherarten, wie z.B. ein Magnetband, Kassette und/oder nichtflüchtige Halbleitermedien. Es sollte verstanden werden, dass Speicherplatte **58** und/oder eine andere Form von RMD **56** als eine Alternative oder zusätzlich zu einem oder mehreren anderen Teilen von Speicher **54** verwendet werden kann, einige oder alle jeglichen Software-Programmierungsbefehle zu speichern, die mit Prozessor(en) **52** ausgeführt werden sollen. In noch anderen Ausführungsformen können RMD **56** und/oder Platte

58 nicht vorhanden sein. Neben RMD **56** kann Speicher **54** eine oder mehrere Komponenten eines elektronischen Festkörpertyps umfassen und zusätzlich oder alternativ andere Arten enthalten, wie z.B. die magnetische oder optische Vielfalt, um ein paar zu nennen. Beispielsweise könnte Speicher **54** elektronische Random-Access-Festkörperspeicher (RAM), sequenziell zugreifbare Speicher (SAM) (wie z.B. die First-In, First-Out (FIFO) Vielzahl oder Last-In First-Out (LIFO) Vielzahl), programmierbare Festwertspeicher (PROM), elektrisch programmierbare Festwertspeicher (EPROM) oder elektrisch löschbare und programmierbare Festwertspeicher (EEPROM); ein Massenspeicher in Form einer optisch lesbaren Platte (wie z.B. eine oder mehrere CDs oder DVDs) oder eine magnetisch codierte Platte (wie z.B. eine oder mehr Harddisks oder Floppydisks); oder eine Kombination von jedem dieser Typen. Auch Speicher **54** kann flüchtig, nichtflüchtig oder eine Kombination aus flüchtigen und nichtflüchtigen Vielfalten sein.

[0029] Einrichtung **50** ist als Host-Einrichtung getrennt von Anordnung **22** in [Fig. 1](#) dargestellt. In einer Form ist Einrichtung **50** ein Standardtyp einer programmierbaren Host-Einrichtung, das selektiv mit Testbus **40** von Anordnung **22** verbunden werden kann. In anderen Ausführungsformen kann Einrichtung **50** als Teil von Anordnung **22** bereitgestellt werden, als eine dedizierte Einrichtung mit Anordnung **22** in einer gemeinsamen Einheit enthalten, oder in unterschiedlichen Konfiguration wie es Fachleuten einfallen könnte. Wie hier verwendet, bezieht sich TDO auf den Testdatenausgang von Einrichtung **50** auf Testbus **40** und auf Testdaten, die an einem Testdateneingang von jeder der Komponenten **30** über Kette **48** empfangen werden; und TDI bezieht sich auf Testdateneingang von Einrichtung **50** von dem Testbus **40** und Testdaten, die von einem Testdatenausgang auf Kette **48** und Testdaten, die von einem Testdatenausgang auf Kette **48** von jeder der Komponenten **30** bereitgestellt werden. Entsprechend kann ein TDI-Signal von dem Testdatenausgang von einer der Komponenten **30** ein TDO-Signal mit Bezug zu dem Testdateneingang einer anderen Komponente **30** sein.

[0030] Jede der gezeigten Komponenten **30** enthält einen zugehörigen, zwischen den Testbus **40** und mindestens einem Testport (TP) **70** gekoppelten Schattencontroller (SC) **60**. Zusätzlich mit Bezug auf [Fig. 2](#) ist SoC-Anordnung detaillierter dargestellt. SC **60** von SoC-Anordnung **34** enthält Steuer-(CNTL)-Logik **62** und Datenspeicheranordnung **66** in Form eines Testportabbildungsregisters **64**. Logik **62** ist angeordnet, einen Testcontroller bereitzustellen, und andere Register, selektiv als ein Testzugangsport (TAP) unter dem JTAG-Standard zu arbeiten. Speicheranordnung **66** kann angeordnet werden, solche anderen Register und/oder Speicher zu enthalten, wie erforderlich, um SC **60** in der Form eines JTAG-TAP zu schaffen. Steuerlogik **62** ist an den Testbus **40** gekoppelt und ist konfiguriert, auf verschiedenen Signalprotokolle, die über Testbus **40** kommuniziert werden, zu reagieren. Testdateneingang (I/P) **34a** und Testdatenausgang (O/P) **43b**, den TDO- und TDI-Signalen des JTAG-Standards entsprechend, sind in [Fig. 2](#) für SC **60** dargestellt.

[0031] SoC-Anordnung **34** enthält drei Testports **70**, die jeder alternativ in [Fig. 2](#) als ein entsprechender Testzugangsport (TAP) **72a**, **72b** und **72c** bezeichnet sind. TAPs **72a**, **72b** und **72c** entsprechen dem JTAG-Standard und sind jeder einer unterschiedlichen Logik-Anordnung von SoC **34** dediziert. Speziell ist TAP **72a** bereitgestellt, um mit einer Computerschaltung mit reduziertem Befehlssatz (RISC) zu koppeln, TAP **72b** ist bereitgestellt, um mit einer Kommunikations-(COMM)-Schaltung zu koppeln und TAP **72c** ist bereitgestellt, um mit einer digitalen Signalprozessor-(DSP)-Schaltung zu koppeln. Jeder Testport **70** enthält einer zugehörigen Testcontroller **74**, der auf das JTAG-Standardsignalprotokoll reagiert, wenn es über SC **60** von Testbus **40** empfangen wird, und enthält verschiedene Register und Logik zum Arbeiten als TAP unter dem JTAG-Standard. TAPs **72a**, **72b** und **72c** sind jeder SC **60** von SiC-Anordnung **34** untergeordnet. SC **60** sind jeder zwischen Testbus **40** und seinem zugehörigen TPs **70** gekoppelt. Für SoC-Anordnung **32** gibt es zwei untergeordnete TPs **70**, für SoC **34** gibt es drei untergeordnete TPs **70**, und für Anordnung **36** gibt es einen untergeordneten TP **70**. SC **60** von anderen der Komponenten **30** sind genauso konfiguriert wie SC **60** der SoC-Anordnung **34**, und jeder TP **70** von anderen der Komponenten **30** sind konfiguriert wie ein JTAG-TAP.

[0032] Das Flussdiagramm von [Fig. 3](#) stellt Routine **120** einer Ausführungsform der vorliegenden Erfindung dar, die mit System **20** ausgeführt werden kann. In Stufe **122** von Routine **120** wird Anordnung **22** auf eine Standardweise angeschaltet oder neu gestartet. Die Neustartoperation kann mit Einrichtung **50** durchgeführt werden, indem der Status von Busleitung **46** entsprechend dem optionalen TRST-Signal des JTAG-Standards geändert wird. Stufe **122** initialisiert Anordnung **22** und Einrichtung **50** in einen TP aktiven Betriebszustand **130**. Im Zustand **130**, sind alle SCs **60** konfiguriert, TPs **70** effektiv durch zugehörige SCs **60** mit dem Testbus **40** entsprechend einer festverdrahteten, Standardtopologie zu verbinden. Typisch enthält diese Topologie die meisten, wenn nicht alle TPs **70** in einer seriellen Scankette. Während Zustand **130** Operation, sind SC **60s** transparent für die Operationen von TPs **70**, die mit Testbus **40** entsprechend der Standardtopologie verbunden sind, so dass Speicher und Register von SC **60** nicht von Testbus zugreifbar sind, sie sind in einer versteckten, Schattenbetriebsanordnung. In Stufe **132** von Zustand **130** wird ein Boundary-Scan-Test durchgeführt, wobei die TPS **70** mit dem Testbus **40** in der Standardtopologie verbunden sind.

[0033] Bedingung **134** testet, ob der Operationszustand geändert werden soll. Wenn der Test von Bedingung **134** negativ ist, werden TPs **70** in Stufe **136** unter der Kontrolle von Einrichtung **50** betrieben und verwenden die aktuelle Testzugriffstopologie. Stufe **136** sorgt für Standard-JTAG_Operationen der TP **70** um richtige Verarbeitung und Operation von Anordnung **22** zu verifizieren, während SCs **60** weiterhin transparent für Testbusoperationen sind. Die Ausführung von Stufe **136** kann Bedienereingabe über Eingabeeinrichtung(en) **50a** und/oder Ausgabe an einen Bediener über Ausgabeeinrichtungen **50b** enthalten, mit einer Anzahl von unterschiedlichen operationellen Parametern oder Optionen. Stufe **136** läuft nach der Ausführung zurück auf Bedingung **134**; aber es sollte verstanden werden, dass die Ausführungsschleife von Bedingung **134**/Schleife der Stufe **136** lediglich beschreibend für eine illustrative Form der vorliegenden Erfindung ist. In anderen Ausführungsformen könnte Stufe **136** auf Vollendung von Bedingung **134** ausgeführt als eine Programm- oder Bedienereingabe warten. Zum Beispiel könnte Bedingung **134** als eine vom Bediener ausgewählte Option, die durch Programmierung von Einrichtung **50** gestellt ist, ausgeführt sein. In einer Form könnte Bedienereingabe mit einem grafischen Benutzerinterface-(GUI)-Knopf, -Menu oder -Abfragebox bereitgestellt werden. Alternativ oder zusätzlich könnte Bedingung **134** von den Ergebnissen der Ausführung einer anderen Stufe abhängig sein und/oder in anderen Weisen, wie es Fachleuten einfallen könnte, ausgeführt sein.

[0034] Wenn der Test von Bedingung **134** bejahend ist, wird Zustand **130** dann beendet und SC Operationszustand **140** beginnt mit Stufe **142** von Routine **120**. In Stufe **142** wird Testtakt TCK an Busleitung **42** durch Einrichtung **50** auf einem niedrigen Logikpegel ausgesetzt. Unter dem JTAG-Standard wird, wenn Testtakt TCK an einen niedrigen Logikpegel geklemmt wird, Operation von JTAG-entsprechenden TAPs auch ausgesetzt. Entsprechend wird beim Anhalten des Testtakts von Testbus **40** Operation von jedem TP **70** angehalten.

[0035] Routine **120** geht von Stufe **142** zu Stufe **144** vor, um ein Signalprotokoll zu aktivieren, um auf SCs **60** zuzugreifen. Das Signalprotokoll ist entsprechend dem JTAG-Standard auf Grund der weitergehenden Aussetzung von Testtakt TCK auf einem niedrigen Logikpegel vor der Operation von Testbus **40** versteckt. Für dieses versteckte Signalprotokoll wird das Testzustandselektionssignal TMS hin und her geschaltet, um Steuersignale bereitzustellen und Information in die und aus den SCs **60** zu takten. Mit zusätzlichem Bezug auf das Timingdiagramm von [Fig. 4](#) wird eine Busalarmsequenz (BAS) dargestellt, in dem der Testtakt niedrig geklemmt wird und das TDO-Signal auf einem konstanten hohen oder niedrigen Logikpegel gehalten wird, während das Testzustandselektionssignal TMS zwischen einem hohen und niedrigen Logikpegel acht mal (Zyklen) hin und her geschaltet wird. Das BAS wird in Stufe **144** ausgeführt, um jeden SC **60** aufzufordern, Testbus **40** für weitere Information zu überwachen.

[0036] Routine **120** geht von Stufe **144** zu Stufe **146** vor. Stufe **146** entspricht der Selektion von einem der SCs **60**. Unter dem versteckten Signalprotokoll ist jedem SC **60** eine eindeutige Adresse zugeordnet. Nach dem BAS der Stufe **144** wird die SC-Adressierung geschaffen durch: (a) Halten des Testtakts TCK auf einem niedrigen Logikpegel, (b) Hin und Herschalten von TMS um als Takt relativ zu Logikstatusänderungen des TDO-Signals zu arbeiten, und (c) Ändern Status von TDO um die eindeutige Adresse mit dem Takten durch TMS zu kommunizieren. [Fig. 5](#) stellt ein Beispiel einer Adressierungssequenz für SC-Selektion, in der eine Adressierungspräambel von 011 binär und eine Adresse von 00101 binär (5 zur Basis zehn) in acht TMS-Zyklen gesendet wird. Dieses Beispiel wählt den SC mit einer Adresse von fünf stellt dabei Adressierbarkeit von bis zu 32 SCs mit dem Fünf Bit Adressteil der Sequenz bereit. Die TDO-Information wird seriell von Einrichtung **50** an jeden SC **60** in einer Kettenart in Stufe **146** kommuniziert.

[0037] Der adressierte SC **60** wird durch die Leistung von Stufe **146** in einen völlig aktiven Status versetzt, während die anderen SCs **60** in einem Busalarmzustand bleiben. Wenn Adressierung in Stufe **146** ausgeführt ist, wird die Operation von Testtakt TCK in Stufe **147** wieder aufgenommen. Der adressierte SC **60** nimmt Testtakt TCK wahr, wobei er als ein TAP unter dem JTAG-Standard in Stufe **148** arbeitet, während all die SCs **60** in einem Busalarmzustand weiter Testtakt TCK ignorieren und alle TPs **70** (inklusive der, die zu dem aktiven SC **60** gehören) ausgesetzt bleiben. Entsprechend ist für Stufe **148** der aktivierte SC **60** der einzige operationelle TAP-Controller und ermöglicht Debuggen und spezielles Testen von ScC **34**. Ein optionaler Befehl unter dem JTAG-Standard stellt auch ein Vehikel zum Ändern der Topologie der während Stufe **148** dem aktiven SC **60** untergeordneten TPs **70** bereit. In einem nicht einschränkenden Beispiel, wo SC **60** von ScC **34** aktiviert ist, kann Abbildregister **64** durch Verwenden dieses Befehls geladen werden. Für die drei TAPs **72a**, **72b**, **72c** von SoC **34**, kann Abbilden der verschiedenen TAP-Topologien mit Register **64** Tabelle I wie folgt entsprechen:

Wert des Mapping-Registers	Spezifizierter Testport oder serielle Testportkette
0x01	TAP 72a
0x02	TAP 72b
0x03	TAP 72a seriell mit TAP 72b verbunden
0x04	TAP 72c
0x05	TAP 72a seriell mit TAP 72c verbunden
0x06	TAP 72b seriell mit TAP 72c verbunden
0x07	TAPs 72a, 72b und 72c seriell in dieser Reihenfolge verbunden

Tabelle I

[0038] In Tabelle I wird der Wert in der höchstwertigen Spalte in Abbildregister **64** gespeichert, um die untergeordneten TPs **70** des aktivierten SC **60** von SoC **34** oder eine seriell verbundene Kette von zwei oder mehr dieser untergeordneten TPs **79** zu definieren. Entsprechend können in dieser Ausführungsform sieben verschiedene Topologien für TAP **72a**, TAP **72b** und TAP **72c** (die untergeordneten TPs **70** von SoC **34**) spezifiziert werden. Auf eine Einschalt- oder Resetbedingung hin enthält jeder SC **60** einen fest verdrahteten Wert, der in das Abbildregister **64** geladen wird, um die entsprechende Einschalt/Reset-Standardtopologie von TPs **70** bereitzustellen. Typischerweise würde diese Standardtopologie alle TPs verketteten, entsprechen zu 0x07 von Tabelle I.

[0039] Von Stufe **148** geht Routine **120** mit Bedingung **149** weiter. Bedingung **149** testet, ob einer neuer SC **60** zu adressieren und folglichweise zu aktivieren ist. Wenn ein unterschiedlicher SC **60** zu aktivieren ist, macht Routine **120** die Schleife zurück, um Stufen **142**, **144**, **146**, **147** und **148** zu wiederholen. Wenn der Test von Bedingung **149** negativ ist, macht die Routine mit Bedingung **150** weiter. Bedingung **150** testet, ob Zustand **140** aktiv bleiben muss. Wenn der Test von Bedingung **150** bejahend ist, macht die Routine die Schleife zurück zu Stufe **148**, um die isolierte Operation des aktivierten SC **60** unter Zustand **140** fortzusetzen. Wenn der Test der Bedingung **150** negativ ist, fährt Routine **120** fort, zu Zustand **130** zurückzukehren. Bedingungen **149** und **150** können auf jede der in Verbindung mit Bedingung **134** beschriebenen Arten implementiert werden.

[0040] Die Rückkehr zu Zustand **140** beginnt mit der Aussetzung von Testtakt TCK in Stufe **151** bei einem niedrigen Logikpegel. Während dieser Aussetzung wird in Stufe **151** eine Busnormalisierungssequenz (BNS) auf den Testbus **40** gesendet. Mit Bezug auf [Fig. 6](#) wird ein Beispiel einer BNS erläutert, die einmal Hin und Herschalten von TDO pro zwei Zyklen des TMS-Signals enthält, wobei Testtakt TCK auf einem niedrigen Logikpegel geklemmt bleibt. In [Fig. 6](#) wird der hohe Pegel von TDO durch eine binäre „1“ repräsentiert und der niedrige Pegel durch eine binäre „0“. TDO wird in serieller Kettenart durch die zugehörigen Eingänge und Ausgänge der Komponenten **30** an SCs **60** gegeben. Wenn der BNS empfangen ist, kehren alle SCs **60** zu dem Schattenzustand des Betriebs zurück. Als Antwort auf das BNS, wird jede neue, durch das Abbildregister **64** definierte Testzugriffsporttopologie wie in Tabelle I beschrieben implementiert, vorausgesetzt, es gibt kein Einschalten oder Reset, das Routine **120** zu Stufe **122** zurückbringt und dabei die Grundtopologie wiederherstellt.

[0041] Routine **120** geht weiter von Stufe **152** zu Stufe **154**, um Betrieb des Testtakts TCK auf Testtaktleitung **42** wieder aufzunehmen. Mit laufendem Testtakt TCK wird Zustand **130** mit Stufe **136** wieder aufgenommen ([Fig. 3](#)). Einmal wieder aufgenommen, wird Zustand **130** mit jeder während Zustand **140** geschaffenen Testzugriffstopologieänderung ausgeführt. Stufe **136** geht dann zu Bedingung **134** vor, um wieder zu testen, ob eine Zustandsänderung stattfinden soll. Also gibt es die Option, denselben oder einen anderen SC **60** nach dem Wiedereingang in Zustand **140** zu selektieren. Auch ein einzelner TP **70** oder ein seriell verketteter Teilsatz von TPs **70**, die dem aktivierten SC **60** untergeordnet sind, kann mit Zustand **140** spezifiziert werden. Die Fähigkeit, SC **60** als ein JTAG-TAP für eine gegebene Anordnung während Zustand **140** zu aktivieren und/oder die Fähigkeit, unterschiedliche Topologien während Zustand **140** für Implementierung auf die Rückkehr zu Zustand **130** zu definieren, kann wünschenswert sein, um spezielle Tests auszuführen, wie z.B. Debuggen von Operationen, die zu einen Prozessor oder logischen Maschine von einer gegebenen der Komponenten **30** von An-

Fig. 4–6:

Bus alert sequence	Busalarmsequenz
Bus normalize sequence	Busnormalisierungssequenz
held at logic level high or low	auf Logikpegel 'High' oder 'Low' gehalten
held at logic level low	auf Logikpegel 'Low' gehalten
Preamble	Präambel
SP Address	SP-Adresse

Patentansprüche

1. System zum Testen von Schaltungen, das Folgendes umfasst:

- einen Testbus (40);
- mehrere Testports (70), jeder mit einem Controller (60), wobei die Testports (70) während eines ersten Betriebszustands (130) entsprechend einem festgelegten Standard betriebsfähig sind; und
- einen ersten, zwischen den Testbus (40) und die Testports (70) gekoppelten Schattencontroller (60), wobei der erste Schattencontroller (60) eine Speicheranordnung (66) enthält, um Information zu speichern, die einer Selektion eines oder mehr der Testports (70) während eines zweiten Betriebszustands (140) entspricht, wobei die Testports während des zweiten Betriebszustands (140) inaktiv sind, der eine oder mehrere der mit der Information selektierten Testports (70) bei Rückkehr zu dem ersten Betriebszustand (130) entsprechend dem festgelegten Teststandard betriebsfähig sind, **dadurch gekennzeichnet**, dass der erste Schattencontroller (60) während des ersten Betriebszustands (130) inaktiv ist, der Testbus (40) betriebsfähig ist, ein Testtaktsignal bereitzustellen, und der erste Schattencontroller (60) auf einen ausgesetzten Zustand des Testtaktsignals damit reagiert, zwischen dem ersten Betriebszustand (130) und dem zweiten Betriebszustand (140) zu wechseln.

2. System nach Anspruch 1, das außerdem einen zweiten, zwischen den Testbus (40) und zwei oder mehr andere Testports (70) gekoppelten Schattencontroller (60) umfasst, wobei der erste Schattencontroller (60) und der zweite Schattencontroller (60) während des zweiten Betriebszustands (140) beide eindeutig adressierbar sind.

3. System nach Anspruch 1 oder 2, in dem das System eine Schaltungsanordnung (22) umfasst, wobei die Schaltungsanordnung (22) den Testbus (40), die Testports (70) und den ersten Schattencontroller (60) umfasst und außerdem Busmaster-Mittel (50) zum Testen und Debuggen der Anordnung (22) umfasst, wobei die Busmaster-Mittel (50) an den Testbus (40) gekoppelt sind.

4. System nach einem der Ansprüche 1–3, in dem die Speicheranordnung (66) die Form eines Registers (64) hat.

5. System nach einem der Ansprüche 1–4, in dem:

- der Testbus (40) einen seriellen Datenausgangspfad, einen seriellen Dateneingangspfad, einen Testtaktpfad und einen Testzustandsselektionspfad enthält; und
- der festgelegte Teststandard der JTAG-Standard ist, der serielle Datenausgangspfad einer TDO-Busleitung (48a) des JTAG-Standards entspricht, der serielle Dateneingangspfad einer TDI-Busleitung (48d) des JTAG-Standards entspricht, der Testtaktpfad der TCK-Busleitung (42) des JTAG-Standards entspricht und der Testzustandsselektionspfad der TMS-Busleitung (44) des JTAG-Standards entspricht.

6. System nach einem der Ansprüche 1–5, das außerdem Folgendes umfasst: eine prozessorlesbare Anordnung (54, 56, 58), die mit einer Vielzahl von Prozessorbefehlen codiert ist, die ausführbar sind, um: einen Boundary-Scan-Test durch eine seriell verknüpfte Kette der Testports (70) entsprechend einem festgelegten Testbus-Protokoll durchzuführen, ein Testtaktsignal zu halten, um den Betrieb der Testports (70) auszusetzen, und den ersten Schattencontroller (60) während des Aussetzens der Testports (70) zu betreiben.

7. Verfahren zum Testen von Schaltungen, das Folgendes umfasst:

- Bereitstellen einer Schaltungsanordnung (22) mit mehreren Testports (70), die in einer ersten Topologie gemäß einem festgelegten Teststandard arbeiten;
- Aussetzen des Betriebs der Testports (70), um eine zweite Topologie eines oder mehrerer der Testports (70)

mit einem an einen Testbus (40) gekoppelten Schattencontroller (60) einzurichten;

– Betreiben des einen oder mehrerer der Testports (70) der zweiten Topologie gemäß dem festgelegten Teststandard nach dem genannten Aussetzen, wobei der Schattencontroller (60) während des genannten Betriebs inaktiv ist, gekennzeichnet dadurch, dass:

das genannte Aussetzen das Festlegen des Aussetzens eines Testtaktsignals auf dem Testbus (40) enthält und außerdem Folgendes umfasst:

- Aktivieren des Schattencontrollers (60) während des genannten Aussetzens, gemäß dem festgelegten Teststandard zu arbeiten; und
- Deaktivieren des Schattencontrollers (60) während des genannten Aussetzens.

8. Verfahren nach Anspruch 7, in dem die erste Topologie einer seriell verbundenen Kette der Testports (70) entspricht und weniger Testports (70) an der zweiten Topologie teilnehmen als an der ersten Topologie.

9. Verfahren nach Anspruch 7 oder 8, in dem die Schaltungsanordnung (22) den Testbus (40) und den Schattencontroller (60) enthält, und das außerdem Selektieren der zweiten Topologie enthält, um die Schaltungsanordnung (22) zu debuggen.

10. Verfahren nach Anspruch 7, das Folgendes umfasst:

- Halten eines Testtaktes, um den Betrieb mehrerer Testports (70) auszusetzen;
- Betreiben eines Schattencontrollers (60) während die Testports (70) ausgesetzt sind; und Wiederaufnehmen des Betriebs eines oder mehrerer der Testports (70) nach dem genannten Betreiben.

11. Verfahren nach Anspruch 10, das außerdem das Laufenlassen des Testtakts während des genannten Betriebs, während die Testports (70) ausgesetzt bleiben, umfasst.

12. Verfahren nach Anspruch 10 oder 11, in dem der Schattencontroller (60) an einen den Testtakt bereitstellenden Testbus (40) gekoppelt ist und mindestens ein Teil der Testports (70) an den Schattencontroller (60) gekoppelt sind.

13. Verfahren nach einem der Ansprüche 7–12, in dem der Schattencontroller (60) und die Testports (70) gemäß dem JTAG-Standard betriebsfähig sind.

Es folgen 5 Blatt Zeichnungen

Anhängende Zeichnungen

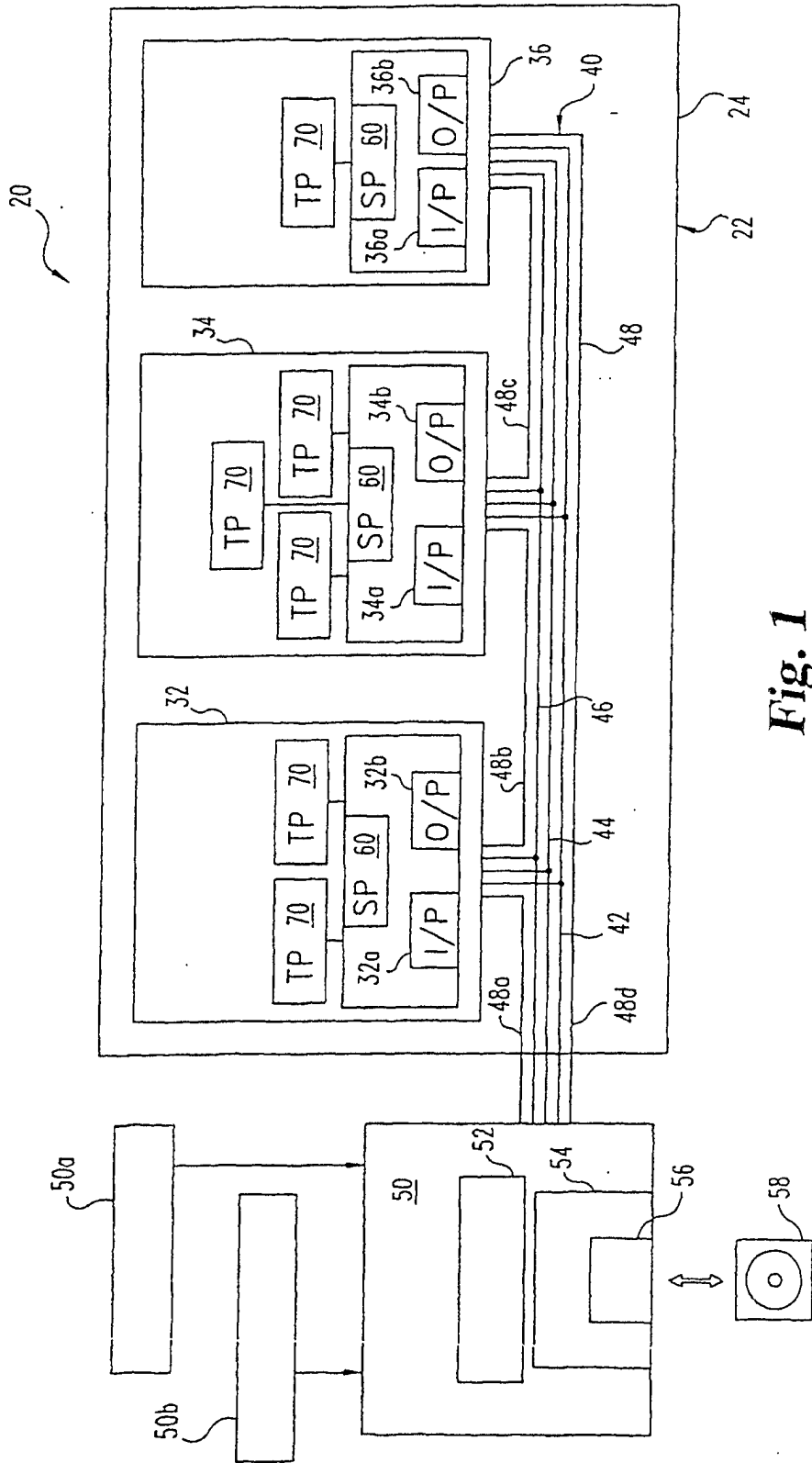


Fig. 1

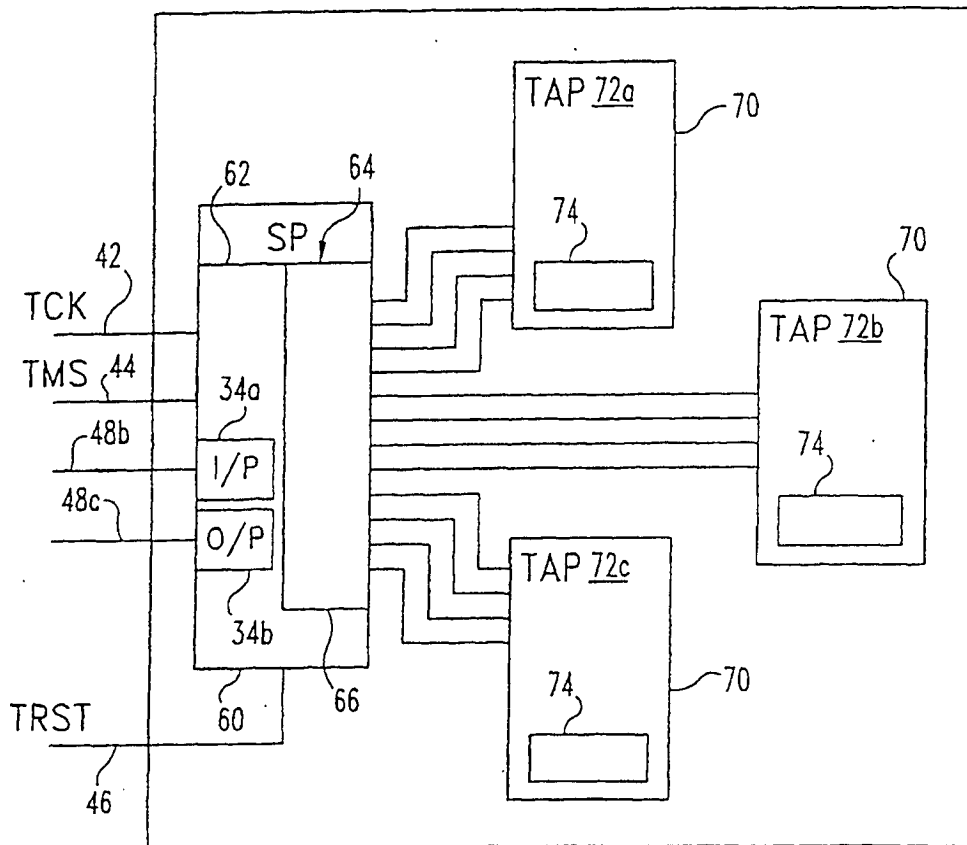


Fig. 2

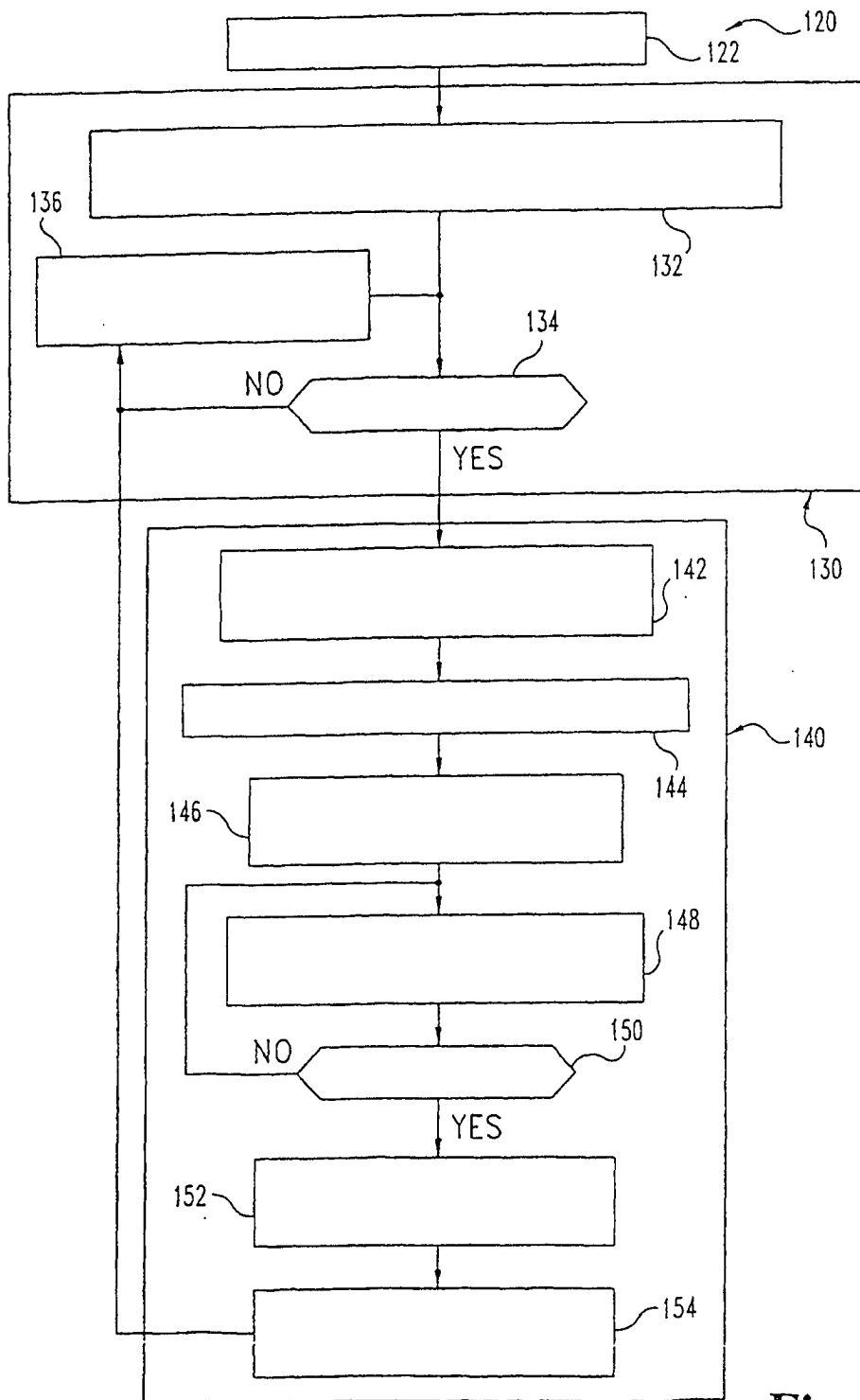


Fig. 3

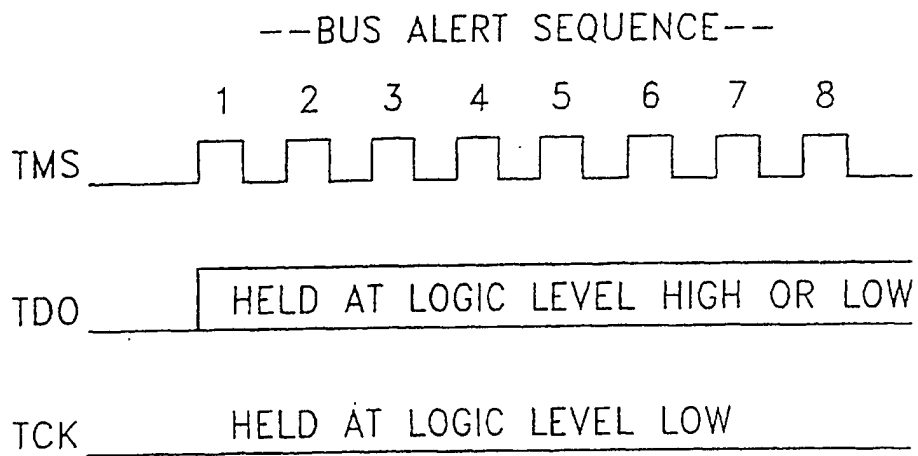


Fig. 4

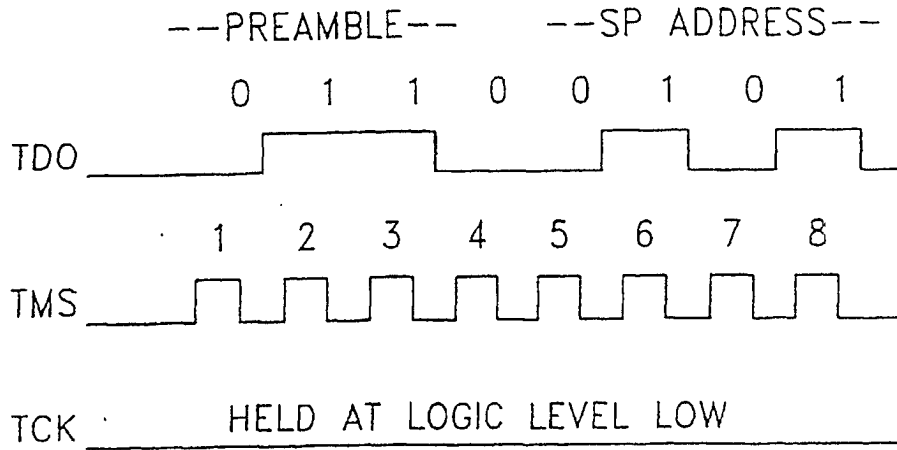


Fig. 5

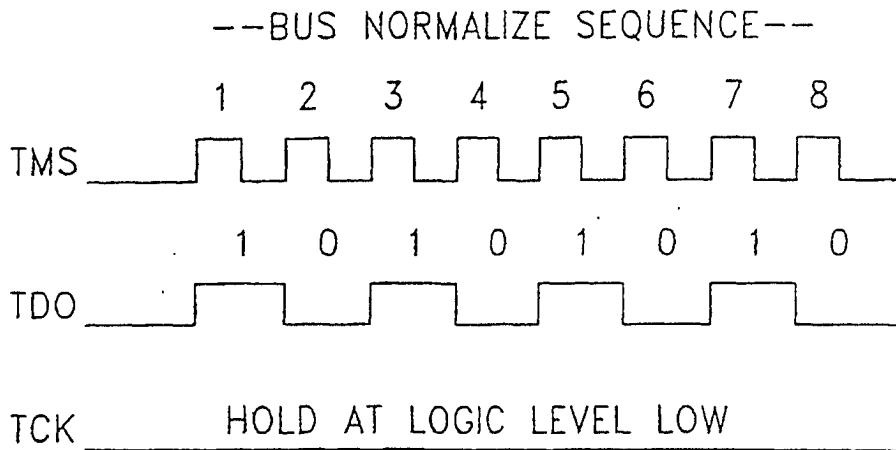


Fig. 6