



(19)  
Bundesrepublik Deutschland  
Deutsches Patent- und Markenamt

(10) **DE 603 16 068 T2** 2008.06.05

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 1 363 132 B1**

(21) Deutsches Aktenzeichen: **603 16 068.9**

(96) Europäisches Aktenzeichen: **03 010 587.8**

(96) Europäischer Anmeldetag: **12.05.2003**

(97) Erstveröffentlichung durch das EPA: **19.11.2003**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **05.09.2007**

(47) Veröffentlichungstag im Patentblatt: **05.06.2008**

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

**G11C 29/00** (2006.01)

**G06F 11/10** (2006.01)

(30) Unionspriorität:

**DE05502002 13.05.2002 IN**

(73) Patentinhaber:

**STMicroelectronics Pvt. Ltd., Uttar Pradesh, IN**

(74) Vertreter:

**WINTER, BRANDL, FÜRNISS, HÜBNER, RÖSS,  
KAISER, POLTE, Partnerschaft, 85354 Freising**

(84) Benannte Vertragsstaaten:

**DE, FR, GB, IT**

(72) Erfinder:

**Pathak, Shalini, Haryana - 122001, IN; Swami,  
Parvesh, New Delhi - 110021, IN**

(54) Bezeichnung: **Prüfverfahren und -gerät für Konfigurationsspeicherelemente in programmierbaren logischen Bauelementen (PLDS)**

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**

## GEBIET DER ERFINDUNG

**[0001]** Die Erfindung betrifft ein Verfahren und eine Vorrichtung zum Testen von Konfigurationsspeicherzellen in programmierbaren Logikvorrichtungen (PLDs).

## HINTERGRUND DER ERFINDUNG

**[0002]** Programmierbare Logikvorrichtungen (PLDs) sind Universal-Logikvorrichtungen, welche für verschiedene Funktionsweisen konfiguriert werden können. Derzeitige Entwicklungen in PLDs ermöglichen es, die Flexibilität und Komplexität von diesen Vorrichtungen zu verbessern. Viele PLDs sehen vollständige Systemtauglichkeiten vor, die auf dem Chip ausgeführte Speicher und eine große Vielfalt und Anzahl an Logikelementen beinhalten, die durch Konfigurieren von Konfigurationsspeicherzellen zum Vorsehen der erwünschten Funktionsweise elektrisch programmiert sind. Um ein vollständiges Realisieren einer Funktionalität sicherzustellen, ist es erwünscht, eine Einrichtung zum Testen der konfigurierten Konfigurationsspeicherzellen auf eine Richtigkeit einer Programmierung aufzuweisen.

**[0003]** Das US-Patent 6 237 124 beschreibt ein Verfahren zum Testen des konfigurierbaren Speichers unter Verwendung einer zyklischen Redundanzprüfung (CRC) gemäß dem Oberbegriff der unabhängigen Ansprüche. Bei diesem Verfahren wird der CRC-Mechanismus verwendet, um eine einmalige Störung (SEU) in dem Konfigurationsverfahren des statischen Direktzugriffsspeichers (SRAM) eines feldprogrammierbaren Gate-Array (FPGA) zu prüfen. In einer FPGA-Architektur auf SRAM-Basis gemäß dem in dieser Druckschrift beschriebenen Ausführungsbeispiel befindet sich ein löschbarer programmierbarer Nur-Lese-Speicher (EPROM) auf einer Leiterplatte in der Nähe des FPGA. Dieser EPROM speichert die Konfigurationsdaten zum Speichern des Konfigurations-SRAM für den FPGA-Kern. Eine EPROM-Steuereinrichtung auf dem FPGA richtet einen Datenstrom von dem EPROM in das FPGA. Die EPROM-Steuereinrichtung serialisiert den Datenstrom von dem EPROM in einen Datenstrom, der ein Bit breit ist. Dieser Datenstrom wird zu einem internen Takt synchronisiert und liefert Daten zu verschiedenen Abschnitten der FPGA-Architektur, die eine Steuerlogik, Reihen- und Spaltenzähler, eine CRC-Schaltung und den Konfigurations-SRAM für den FPGA-Kern beinhalten. Die CRC-Schaltung testet den Datenstrom von dem EPROM, um zu überprüfen, das dieser richtig ist. Ebenso kann die CRC-Schaltung, wie es in der Beschreibung offenbart ist, die Daten in dem Konfigurations-SRAM nach einem Laden testen. Ein Multiplexer wählt aus, ob die von der CRC-Schaltung zu prüfenden Daten der Ein-

gangsdatenstrom oder die Daten in dem Konfigurations-SRAM sind. Wenn unrichtige Konfigurationsdaten in dem Konfigurations-SRAM des FPGA-Kerns erfasst werden, kann die CRC-Schaltung der EPROM-Steuereinrichtung signalisieren, dass ein Fehler aufgetreten ist. Die EPROM-Steuereinrichtung kann diese Information verwenden, um ein Fehlersignal aus dem FPGA auszugeben und/oder ein erneutes Laden von dem EPROM einzuleiten.

**[0004]** Die zuvor beschriebene Konfigurationsschaltung ist zum Konfigurieren der herkömmlichen Speicheranordnung angemessen, aber beinhaltet keinen Mechanismus, durch welchen das vollständige Konfigurationsverfahren getestet werden könnte. Die CRC-(zyklische Redundanzprüfung)-Bits oder die Rahmenüberprüfungsfolgenbits, die nach jedem Rahmen von Daten geladen werden, ermöglichen es, auf irgendeinen Fehler zu überprüfen, der beim Laden der Datenrahmen in das interne Schieberegister auftreten kann, aber die Datenzeilen selbst könnten mit Bits fehlerhaft sein, die bei "1" hängen bleiben oder bei "0" hängen bleiben. In einem derartigen Fall kann es nicht sichergestellt werden, dass die Daten richtig an dem Ziel angekommen sind. Die Anforderung ist deshalb, jedes Bit von gespeicherten Daten zurückzuverfolgen.

**[0005]** Eine Speichervorrichtung mit Testmerkmalen ist in dem US-Patent 5 305 261 von Furutani et al. offenbart.

**[0006]** Ein FPGA mit einem eingebauten Selbsttest ist von Stroud C., Wijesuriya S., Hamilton C., Abramovici M., "Built-in self-test of FPGA interconnect", Proceedings of the International Test Conference 1998, Washington, DC, USA, 18. bis 23. Oktober 1998, Seiten 404 bis 411 beschrieben.

**[0007]** Ein Speicher, der eine Speicheranordnung beinhaltet, die ohne Unterbrechen des Betriebs der gesamten Anordnung erneut konfiguriert werden kann, ist in dem US-Patent 5 488 582 von Camarota offenbart.

## KURZFASSUNG DER ERFINDUNG

**[0008]** Die Aufgabe der Erfindung besteht darin, ein bitweises Verfolgen der Daten zu schaffen, die in der konfigurierbaren Datenverriegelungsanordnung verriegelt sind.

**[0009]** Eine weitere Aufgabe der Erfindung besteht darin, ein Testen des Speichers auf eine transparente Weise zuzulassen.

**[0010]** Gemäß der Erfindung ist eine programmierbare Logikvorrichtung (PLD) oder ein programmierbares Gate-Array (PGA) zum Testen der Konfigurationsspeicheranordnung gemäß Anspruch 1 offenbart.

Weiterhin ist ein Verfahren zum Freigeben des Testens der Konfigurationsspeicheranordnung einer programmierbaren Logikvorrichtung (PLD) oder eines programmierbaren Gate-Arrays (PGA) gemäß Anspruch 6 offenbart.

**[0011]** Um diese und andere Aufgaben zu lösen, schafft diese Erfindung eine programmierbare Logikvorrichtung (PLD), die die Fähigkeit beinhaltet, den Konfigurationsspeicher entweder unabhängig oder während einer Konfiguration zu testen, die eine Auswahlrichtung zum Auswählen einer besonderen Spalte oder Reihe der Konfigurationsspeicheranordnung, einen Eingabedatenspeicher zum Speichern von Konfigurationsdaten, von denen es erforderlich ist, das sie in der ausgewählten Spalte oder Reihe gespeichert sind, oder von Testdaten zum Testen der ausgewählten Spalte oder Reihe, einen Ausgabedatenspeicher zum Speichern der Ausgabe aus der ausgewählten Spalte oder Reihe und eine Testlogik aufweist, die Steuersignale zum Überprüfen des richtigen Betriebs der Datenleitungen der Konfigurationsspeicheranordnung ohne Stören der Daten liefert, die in der Speicheranordnung gespeichert sind. Die Testlogik wählt die Eingaben und Ausgaben der Speicherzellen in der Speicheranordnung von den Spalten oder Reihen der Speicheranordnung während eines Testens ab. Der richtige Betrieb der Datenleitungen der Speicheranordnung wird durch Anwenden von verschiedenen Testmustern, die "Alle 0"-, "Alle 1"-, "Wandernde 0"- und "Wandernde 1"-Testmuster beinhalten, und zurückschreiben und überprüfen der Ausgabe aus der Speicheranordnung überprüft.

#### KURZE BESCHREIBUNG DER FIGUREN

**[0012]** Die Aufgaben und Vorteile der Erfindung werden unter Bezugnahme auf die folgende Beschreibung und die beiliegende Zeichnung ersichtlicher, in welcher:

**[0013]** [Fig. 1A](#) eine vereinfachte Darstellung einer herkömmlichen konfigurierbaren Speicherverriegelungsanordnung zeigt.

**[0014]** [Fig. 1B](#) eine Struktur einer herkömmlichen Konfigurationsspeicherzelle zeigt.

**[0015]** [Fig. 2A](#) ein Blockschaltbild zeigt, dass das Konfigurationsverfahren gemäß einem der einfachsten Ausführungsbeispiele der vorliegenden Erfindung darstellt.

**[0016]** [Fig. 2B](#) einen Stromlaufplan zeigt, der ein Konfigurationsverfahren gemäß einem zweiten Ausführungsbeispiel der vorliegenden Erfindung darstellt.

**[0017]** [Fig. 2C](#) einen Stromlaufplan zeigt, der eine verbesserte Konfigurationsspeicherzelle für die kon-

figurierbare Datenverriegelungsanordnung gemäß der vorliegenden Erfindung darstellt.

**[0018]** [Fig. 2D](#) eine Darstellung einer Steuerschaltung zum Auswählen zwischen den Test- und Konfigurationsbetriebsraten in einem Ausführungsbeispiel der vorliegenden Erfindung zeigt.

#### DETAILLIERTE BESCHREIBUNG

**[0019]** Um die Zeichnung und die Beschreibung zu vereinfachen, wird die Erfindung unter Bezugnahme auf eine 8×8-Konfigurationsspeicherzellenanordnung zum Konfigurieren einer Anordnung von Logikmodulen und einer Verbindungsarchitektur beschrieben, jedoch werden Ausführungsbeispiele mit einer anderen komplexen Anordnungsstruktur für Fachleute ersichtlich. Die folgende Beschreibung der vorliegenden Erfindung ist lediglich veranschaulichend und nicht auf irgendeine Weise beschränkend.

**[0020]** Die konfigurierbaren Speicherzellen einer PLD sind typischerweise in einer Matrix angeordnet und werden unter Verwendung von seriellen Bitströmen von Konfigurationsdaten geladen. [Fig. 1A](#) stellt eine Konfiguration im Stand der Technik von Konfigurationsspeicheranordnungen dar. Eine Konfigurationsspeicheranordnung **100** weist Speicherzellen  $M_{x,y}$  auf, wobei  $x$  und  $y$  der Reihen- bzw. Spaltenstelle von jeder Konfigurationsspeicherzelle entsprechen.

**[0021]** Die Konfiguration von Daten in der Konfigurationsspeicheranordnung wird durch Schieben des Bitstroms von Konfigurationsdaten durch ein vertikales Schieberegister **101** erzielt, das durch einen Taktmechanismus **200** gesteuert wird, bis ein Rahmen von Daten (in diesem Beispiel 8 Bit breit) in das Bitregister **101** geschoben worden ist. Dieser Rahmen von Daten wird dann parallel durch horizontale Datenleitungen **100<sub>7</sub>** bis **100<sub>0</sub>** in einer Spalte von Konfigurationsspeicheranordnungen freigegeben, die durch ein Horizontalschieberegister **102** freigegeben werden. Die Spalte wird durch Schieben eines hohen/niedrigen Bits zu einer der ausgewählten vertikalen Leitungen **100<sub>8</sub>** bis **100<sub>15</sub>** über eine Flipflop-Anordnung des Horizontalschieberegisters **102** unter Verwendung einer Verschiebung pro Rahmen adressiert. Das Laden von Daten wird daher in einer Serial-In-Parallel-Out-(SIPO)-Betriebsart durchgeführt. Wenn das hohe/niedrige Bit nach äusserst rechts heraus geschoben wird, bewirkt es, dass das DONE-Signal (nicht gezeigt) der Konfigurationslogik hoch wird, was eine Anzeige ist, dass die Konfiguration beendet ist.

**[0022]** [Fig. 1B](#) zeigt den Stromlaufplan einer einzelnen Speicherzelle **103**. Die Speicherzelle **103** besteht aus einem Signalspeicher **103-2**, der durch zwei antiparallele Inverter **103-3** und **103-4** ausgebildet ist, der einen Bitwert speichert, der durch den

Durchgangstransistor **103-1** gegangen ist. Während einer Konfiguration wird, wenn das hohe Bit zu **103-5** verschoben wird, der Durchgangstransistor **103-1** freigegeben und wird die Information, die an der Leitung **103-6** vorhanden ist, in den Datensignalspeicher konfiguriert. Bei der Konfiguration sind Blockleitungen **103-5** und **103-6** einer der Ausgänge von Schieberegistern **102** bzw. Registern **101** von [Fig. 1A](#). Diese Anordnung sieht keinerlei Einrichtung zum Testen der konfigurierten Datensignalspeicher vor.

**[0023]** [Fig. 2A](#) zeigt eine vereinfachte Darstellung des Konfigurationsverfahrens gemäß der vorgeschlagenen Erfindung. Die Erfindung beinhaltet einen Mechanismus zum Freigeben des Tests von Konfigurationsdaten zu irgendeinem Zeitpunkt während oder nach einer Konfiguration ohne das Erfordernis einer zusätzlichen Schaltung. Der Speicher weist eine Konfigurationsspeicheranordnung **106**, das Vertikaleinschieberegister **104**, das Vertikalausschieberegister **107**, das Horizontalschieberegister **105** und den Konfigurationslogikblock **201** auf. Die Vertikalregister **104** und **107** sind über horizontale Leitungen **106<sub>0</sub>** bis **106<sub>7</sub>**, mit den Reihen der konfigurierbaren Zellen verbunden und die Ausgänge des Horizontalschieberegisters **105** sind durch vertikale Leitungen **106<sub>8</sub>** bis **106<sub>15</sub>** zu der Speicherzelle kaskadiert. Das Horizontalschieberegister **105** ist mit zwei gesonderten Bitrahmen versehen, welche als Pseudo-Bits verwendet werden.

**[0024]** Der Konfigurationslogikblock **201** beinhaltet allgemein Eingangsanschlüsse CONFIG, einen Datenbus DIN (0:7) und das Taktsignal CLK. Der Block **201** beinhaltet intern ebenso verschiedene Register usw.

**[0025]** Während einer Konfiguration wird der Datenrahmen über den Datenbus DIN in die Konfigurationslogik **201** geladen und in den verschiedenen Schieberegistern der Konfigurationslogik verriegelt. Eines der Schieberegister der Konfigurationslogik **201** sendet dann die Datenrahmen unter einem Steuern eines Takts VSR\_CLK, welcher von der Konfigurationslogik **201** erzeugt wird, zu dem Vertikaleinschieberegister **104**. Der Datenrahmen wird unter Verwendung dieses Taktmechanismus seriell in die Bitpositionen des Vertikaleinschieberegisters **104** geschoben. Ausgangsbits des Vertikaleinschieberegisters **104** sind mit Datenleitungen **106<sub>0</sub>** bis **106<sub>7</sub>** verbunden. Wenn der vollständige Datenrahmen in **104** geladen worden ist und die CRC-Schaltung keinen Fehler markiert wird ein Takt HSR\_CLK von der Konfigurationslogik **201** zum Freigeben des Horizontalschieberegisters erzeugt, um eine besondere Spalte der Konfigurationsspeicheranordnung durch Schieben eines hohen Bits **105<sub>0</sub>** bis **105<sub>9</sub>**, eine Verschiebung pro Datenrahmen, in dieses zu adressieren. Abhängig davon, welche der Ausgangsbits des Horizon-

talschieberegisters **105** hoch sind, wird eine der Datenleitungen **106<sub>8</sub>** bis **106<sub>15</sub>**, die mit dieser Bitposition verbunden ist, freigegeben, um dadurch zuzulassen, dass alle der Konfigurationsspeicherzellen, die in dieser Spalte vorhanden sind, die Datenrahmenbits verriegeln, die in dem Vertikaleinschieberegister **104** gespeichert sind.

**[0026]** Bits **105<sub>0</sub>** und **105<sub>1</sub>** des Horizontalschieberegisters **105** sind Pseudo-Bits und werden nicht verwendet, um irgendeine Spalte zu adressieren. Dies wird durchgeführt, da die ersten zwei Rahmen, die nach der Einleitung geladen werden, Testrahmen sind, um die Datenleitungen auf Störungen zu prüfen. Diese Rahmen werden nicht in den Konfigurationsspeicherzellen verriegelt und, um zu verhindern, dass dies auftritt, wird verhindert, dass Bits **105<sub>0</sub>** und **105<sub>1</sub>** irgendeine der Spalten erreichen, wenn das Horizontalschieberegister **105** bald nach dem Laden der Einleitung ein Arbeiten beginnt. Dies wird durch Freigeben des Speicherns von Daten in Konfigurationsspeicherzellen lediglich dann erzielt, wenn das Bit **105<sub>2</sub>** hoch wird. Um diese zusätzlichen Rahmen zu kompensieren, ist die letzte Rahmenadresse, die an dem Konfigurationsblock **201** vorgesehen wird (um das Ende des Konfigurationsverfahrens zu markieren) immer um zwei größer als die tatsächliche Adresse. HSR\_CLK steuert ebenso das parallele Laden von Daten in das Vertikalausschieberegister **107**. Immer dann, wenn HSR\_CLK hoch wird, werden die Inhalte des Vertikaleinschieberegisters **104** parallel über Datenleitungen **106<sub>0</sub>** bis **106<sub>7</sub>**, in das Vertikalausschieberegister **107** geladen. VSR\_CLK ist ebenso direkt mit dem Vertikalausschieberegister **107** verbunden und gibt das serielle Ausschleichen der vorherigen Daten aus einem Ausgangsanschluss frei, wenn der nächste Datenrahmen ein Laden in **104** beginnt.

**[0027]** Ein weiteres mögliches Ausführungsbeispiel der Erfindung ist in [Fig. 2B](#) gezeigt. Dieses Ausführungsbeispiel schafft eine Steuereinheit **108** und ein TEST-Signal. Die Zellen der Steuereinheit **108** sind mit Leitungen **106<sub>8</sub>** bis **106<sub>15</sub>** verbunden und weisen jeweils zwei Ausgänge  $E_i$  und  $T_i$  auf, die mit der Konfigurationsspeicheranordnung verbunden sind.  $I_i$  gibt eine besondere Spalte für jeden Dateneintrag frei, wohin gegen  $T_i$  ein Testen einer besonderen Spalte der Datenanordnung frei gibt.

**[0028]** [Fig. 2C](#) zeigt die interne Schaltung einer einzelnen Konfigurationsspeicherzellenstruktur  $M_{0,0}$  von [Fig. 2B](#). Anstatt eines direkten Verbindens des Datensignalspeichers mit der horizontalen Leitung **106<sub>0</sub>**, wird sie über einen Durchgangstransistor **109-5** und einen Inverter **109-2** mit drei Zuständen geleitet. Dies lässt **106<sub>0</sub>** während sowohl einer Konfiguration als auch eines Testens frei. Ein Signal  $E_0$  liegt an dem Gate-Anschluss des Durchgangstransistors **109-5** an, während  $T_0$  an der Freigabeleitung des Inverters

**109-2** mit 3 Zuständen anlegt. Die Konfigurations-speicherzelle **109-1** ist durch zwei antiparallele Inverter **109-3** und **109-4** ausgebildet.

**[0029]** [Fig. 2D](#) zeigt den Stromlaufplan für eine der Zellen **108<sub>0</sub>** der Steuerschaltung **108**. Die Steuerschaltung weist UND-Gatter **110-1** und **110-2** auf. Erste Eingangsanschlüsse von **110-1** und **110-2** sind über eine Leitung **106<sub>8</sub>** mit einem Bit **105<sub>2</sub>** verbunden. Das TEST-Signal, das von der Konfigurationslogik **201** kommt, ist der zweite Eingangsanschluss des UND-Gatters **110-2** und das invertierte TEST-Signal wird der zweite Eingangsanschluss von **110-1**.

**[0030]** Um die Konfiguration zu beginnen, wird das CONFIG-Signal für einen Taktzyklus hoch. Dies erzeugt ein internes RESET-Signal (nicht gezeigt), das die Register der Konfigurationslogik **201** zurücksetzt. Das Signal TEST ist zu der Zeit einer Konfiguration eine logische 0. Dies stört alle der Inverter mit drei Zuständen, da die Ti's zu 0 gebunden werden. Vor einem Laden der tatsächlichen Datenrahmen werden einige Testbits über den Dateneingangsbus DIN geladen. Das erste Testbyte ist überall 0. Die Konfigurationslogik **201** erzeugt VSR\_CLK und die Testbits starten ein serielles Schieben in das Vertikaleinschieberegister **104**. Wenn das letzte Bit des Rahmens Din7 zu der Bitposition **104<sub>7</sub>** gesendet wird, wird VSR\_CLK gesperrt und wird HSR\_CLK erzeugt, was das parallele Laden von Datenbits **104<sub>7</sub>** bis **104<sub>0</sub>** über Datenleitungen **106<sub>7</sub>** bis **106<sub>0</sub>** zu Bitpositionen **107<sub>7</sub>** bis **107<sub>0</sub>** bewirkt. Der gespeicherte Datenrahmen des Vertikalausschieberegisters **107** wird dann seriell aus dem Ausgangsanschluss ausgeschoben. Wenn die Datenbits, die von dem Vertikaleinschieberegister **107** empfangen werden, irgendein Bit als eine "1" enthalten, würde dies anzeigen, dass die Datenleitung, die mit dieser besonderen Bitposition verbunden ist, "bei 1 hängen bleibt" (da der Datenrahmen, der vorgesehen worden ist, überall 0 war). Das nächste Testbyte von überall 1 ist auf eine ähnliche Weise beim Erfassen von Fehlern eines "Hängenbleibens bei 0" nützlich. Dies beendet die erste Phase eines Testens, in welcher es sichergestellt wird, dass die Datenleitungen richtig arbeiten. Die Konfiguration der Daten beginnt nun und wenn das hohe Bit in das Horizontalschieberegister **105** geschoben wird, lässt das Bit **105<sub>2</sub>** das sich ergebende hohe Signal an dem Gate des Durchgangstransistors **109-5** auf Leitung **15**, zu, dass das Konfigurationsbit, das in **104** gespeichert ist, den Signalspeicher über Datenleitungen **106<sub>0</sub>** usw. erreicht. In dieser Betriebsart bleibt das Signal TEST niedrig, was dazu führt, dass der Ausgang **T<sub>i</sub>** dauerhaft niedrig bleibt, was bewirkt, dass der Inverter **109-2** mit drei Zuständen in dem Zustand einer hohen Impedanz ist. Das Ausgangssignal auf der Leitung **15** hängt von dem Wert ab, der in der entsprechenden Bitposition des Horizontalschieberegisters **105** gespeichert ist. Wenn das Bit "hoch" ist, wird die entsprechende Signalleitung **106**

"hoch", was an dem Ausgang **15** "hoch" ergibt. Das Gate des Durchgangstransistors **109-5** ist mit der Leitung **15** verbunden. Deshalb schaltet **109-5** ein, was zulässt, dass der Wert, der in dem entsprechenden Bit des Vertikaleinschieberegisters **104** gespeichert ist, über die Datenleitung **106<sub>0</sub>** usw. in **109-1** verriegelt wird. Während einer Konfiguration wird, wenn die Datenrahmen in einer nicht zugewiesenen Datensignalspeichergruppe verriegelt werden, ein Testen der konfigurierten Daten gleichzeitig wie folgt ausgeführt.

**[0031]** Nachdem der erste Rahmen in das Vertikaleinschieberegister **104** geladen worden ist, wird der Takt zu dem Horizontalschieberegisters **105**, das heisst HSR\_CLK, erzeugt. Das Dekodiersystem der Konfigurationslogik **201** bewirkt, dass ein Bit **105<sub>2</sub>** des Horizontalschieberegisters **105** "hoch" wird, was die erste Spalte der Speicheranleitung **106** über die Leitung **15** freigibt. Die Signalspeicher in dieser Spalte sowie das Vertikalausschieberegister **107** speichern die Bitwerte, die in Bits **104<sub>0</sub>** bis **104<sub>7</sub>** des Vertikaleinschieberegisters vorhanden sind. Das serielle Laden der zweiten Datenrahmen in das Vertikaleinschieberegister **104** und das serielle Ausschieben des ersten Rahmens aus dem Vertikalausschieberegister **107** treten gleichzeitig auf.

**[0032]** Dies lässt ein gleichzeitiges Überprüfen von jedem Datenrahmen zu. Dieses Verfahren schreitet fort, bis alle der Rahmen geladen worden sind. Dieser Ansatz bestätigt, dass die Konfigurationslogik **201** sowie das Vertikaleinschieberegister **104** funktionsbezogen richtig sind, da es immer eine Möglichkeit gibt, dass das CRC-Verfahren nicht irgendeinen Fehler in dem Konfigurationsverfahren zeigt, sondern der richtige Datenrahmen das Vertikaleinschieberegister **104** nicht erreichen kann oder das Letztere nicht im Stande sein kann, die Daten zu dem richtigen Ziel zu senden.

**[0033]** Nach einem Beenden der Konfiguration und des gleichzeitigen Testens können die Signalspeicher oder Spalten von Signalspeichern durch Anheben des TEST-Signals auf hoch weiterhin einzeln überprüft werden. Dies ist die tatsächliche Testbetriebsart, in welcher jeder verriegelte Wert zurückgelesen wird. Ein Setzen des Bits TEST auf hoch bewirkt, dass das Signal **16** hoch wird, was daher den Inverter **109-2** mit drei Zuständen freigibt, und das Bit, das in dem Signalspeicher **109-1** gespeichert ist, wird von dem Vertikalausschieberegister **107** zurückgelesen. Ebenso wird, da das Signal TEST hoch bleibt, das Signal **15** dauerhaft niedrig. Der Durchgangstransistor **109-5** bleibt daher ausgeschaltet und der Wert, der in dem Signalspeicher **109-1** gespeichert ist, kann nicht überschrieben werden. Der Signalwert von **16** wird hoch und der verriegelte Wert kann aus dem Ausgang von **109-2** ausgelesen werden. Der Konfigurationsblock **201** bleibt in dieser Betriebsart ebenso funktional, da Taktsignale VSR\_CLK

und HSR\_CLK auf die gleiche Weise wie in der Konfigurationsbetriebsart erzeugt werden. Das Taktsignal VSR\_CLK und HSR\_CLK gibt das Vertikalausschieberegister **107** sowie die Inverter mit drei Zuständen frei, die für die Testzwecke synchron aktiviert werden. Der Konfigurationsblock wirkt auf die übliche Weise und Bitströme von 0 können nur über den Eingangsbus DIN gesendet werden. Tatsächlich beeinträchtigt, egal welche Daten über den Eingangsbus gesendet werden können, dies nicht die bereits gespeicherten Bitwerte, da die Eingangsseite von jeder Speicherzelle  $N_{x,y}$  ausgeschaltet bleibt, da Signale Ei's, die auf 0 gebunden bleiben, ebenso aufgrund des hohen Einschaltsignals, wobei das TEST-Signal die Ausgänge des Vertikaleinschieberegisters **104** an drei Zuständen belässt.

**[0034]** Das Horizontalschieberegister **105** gibt die Spalten der Speicheranordnung eine nach der anderen zum Lesen über Datenleitungen Ti's frei. Die ausgewählte Spalte sendet ihre verriegelten Werte über die Inverter mit drei Zuständen zu Datenleitungen **106<sub>0</sub>** bis **106<sub>7</sub>**, welche dann parallel zu **107** geladen werden. Das serielle Ausschleichen des Bitstroms aus dem Vertikalausschieberegister **107** liefert die Werte, die in den einzelnen Signalspeichern der besonderen Spalte gespeichert sind.

**[0035]** Gemäß einem weiteren Ausführungsbeispiel der vorliegenden Erfindung (nicht gezeigt), kann ein weiteres Schieberegister an dem anderen Ende der Speicheranordnung parallel zu dem Horizontalschieberegister **105** verwendet werden. Dies ist zum Testen der Datenleitungen, die von dem Horizontalschieberegister **105** vertikal nach oben verlaufen, auch vor dem Beginn des tatsächlichen Konfigurationsverfahrens nützlich. Dieses Register kann ebenso zu der Zeit einer Konfiguration verwendet werden, um die Werte auszuschleichen, die in dem Horizontalschieberegister **105** gespeichert sind, was indirekt das Adressendekodiersystem der Konfigurationslogik testen wird. Die vorliegende Erfindung schafft daher ein wirksames Verfahren zum Testen durch Einbringen lediglich geringfügiger Änderungen in die verfügbaren Konfigurationsressourcen.

**[0036]** Es ist für Fachleute ersichtlich, dass das Vorgehende lediglich veranschaulichend als erschöpfend oder beschränkend gedacht ist, lediglich beispielhaft dargestellt worden ist und das verschiedene Ausgestaltungen innerhalb des Umfangs der vorhergehenden Erfindung durchgeführt werden können. Zum Beispiel könnte die Konfigurationsspeicheranordnung von irgendeiner erwünschten Abmessung sein. Weiterhin könnten das Vertikaleinschieberegister und das Vertikalausschieberegister mit Reihen anstatt Spalten der Speicheranordnung verbunden sein, während das Horizontalschieberegister mit Spalten anstatt Reihen verbunden ist.

**[0037]** Demgemäß ist diese Erfindung nicht auf die spezifischen Beispiele beschränkt, die zum Zwecke einer Offenbarung gewählt worden sind, sondern statt dessen als alle Änderungen und Ausgestaltungen bedeckend zu erachten, welche keine Abweichungen von dem zulässigen Umfang der Erfindung bilden, wie er durch die Ansprüche definiert ist.

## Patentansprüche

1. Programmierbare Logikvorrichtung, PLD, oder programmierbares Gate-Array, PGA, zum Testen der Konfigurationsspeicheranordnung (**106**) entweder unabhängig oder während einer Konfiguration, die aufweist:

eine Auswahleinrichtung (**105**) zum Auswählen einer besonderen Spalte oder Reihe der Konfigurationsspeicheranordnung,  
einen Eingabedatenspeicher (**104**) zum Speichern von Konfigurationsdaten, von denen es erforderlich ist, dass sie in der ausgewählten Spalte oder Reihe gespeichert werden, oder von Testdaten zum Testen der ausgewählten Spalte oder Reihe,  
einen Ausgabedatenspeicher (**107**) zum Speichern der Ausgabe aus der ausgewählten Spalte oder Reihe,  
eine Testlogik (**108**),

**dadurch gekennzeichnet, dass**

die Testlogik (**108**) Steuersignale zum Überprüfen des richtigen Betriebs von Durchgangs-Datenleitungen (**106<sub>1</sub>** bis **106<sub>7</sub>**) zwischen dem Eingabedatenspeicher (**104**) und dem Ausgabedatenspeicher (**107**) der Konfigurationsspeicheranordnung (**106**) ohne Stören oder Lesen der Daten liefert, die in Speicherzellen der Konfigurationsspeicheranordnung gespeichert sind.

2. Programmierbare Logikvorrichtung, PLD, oder programmierbares Gate-Array, PGA, nach Anspruch 1, wobei die Auswahleinrichtung (**105**) ein Seriell/Parallel-Wandler ist, der Freigabesignale (**106<sub>8</sub>** bis **106<sub>15</sub>**) zum Auswählen der erwünschten Spalte oder Reihe der Konfigurationsspeicheranordnung liefert.

3. Programmierbare Logikvorrichtung, PLD, oder programmierbares Gate-Array, PGA, nach Anspruch 1, wobei der Eingabedatenspeicher (**104**) ein Seriell/Parallel-Wandler ist, der entweder die Konfigurationsdaten, von denen es erforderlich ist, dass sie in der ausgewählten Spalte oder Reihe gespeichert werden, oder die Testdaten zum Überprüfen des Betriebs der Durchgangs-Datenleitungen (**106<sub>1</sub>** bis **106<sub>7</sub>**) der Konfigurationsspeicheranordnung speichert.

4. Programmierbare Logikvorrichtung, PLD, oder programmierbares Gate-Array, PGA, nach Anspruch 1, wobei der Eingabedatenspeicher (**104**) ein Seriell/Parallel-Wandler ist, der zurückgelesene Daten speichert und liefert, die von den Spalten oder Rei-

hen der Konfigurationsspeicheranordnung (**106**) empfangen werden.

5. Programmierbare Logikvorrichtung, PLD, oder programmierbares Gate-Array, PGA, nach Anspruch 1, wobei die Testlogik (**108**) eine Logikschaltung ist, die die Eingaben und Ausgaben der Speicherzellen in der Konfigurationsspeicheranordnung während eines Testens aus den Spalten oder Reihen der Konfigurationsspeicheranordnung abwählt.

6. Verfahren zum Freigeben des Testens der Konfigurationsspeicheranordnung einer programmierbaren Logikvorrichtung, PLD, oder eines programmierbaren Gate-Arrays, PGA, entweder unabhängig oder während einer Konfiguration, das die Schritte aufweist:

Auswählen einer besonderen Spalte oder Reihe der Konfigurationsspeicheranordnung (**106**),

Speichern von eingegebenen Konfigurationsdaten, von denen es erforderlich ist, dass sie in der ausgewählten Spalte oder Reihe gespeichert werden, oder von eingegebenen Testdaten zum Testen der ausgewählten Spalte oder Reihe,

Speichern der Ausgabe der ausgewählten Spalte oder Reihe und Liefern von zurückgelesenen Daten des Ausgabedaten, gekennzeichnet durch

Liefern von Steuersignalen (T0 bis T7) zum Überprüfen des richtigen Betriebs von Durchgangs-Datenleitungen (**106<sub>1</sub>** bis **106<sub>7</sub>**) der Konfigurationsspeicheranordnung zwischen den eingegebenen Konfigurationsdaten und der gespeicherten Ausgabe ohne Stören oder Lesen der Daten, die in den Speicherzellen der Konfigurationsspeicheranordnung gespeichert sind.

7. Verfahren zum Freigeben des Testens der Konfigurationsspeicheranordnung einer Logikvorrichtung nach Anspruch 6, wobei das Überprüfen des richtigen Betriebs der Durchgangs-Datenleitungen durch Speichern von Testmustern in den eingegebenen Testdaten und Bestätigen des richtigen zurückgelesenen Testmusters in den Ausgabedaten erzielt wird.

8. Verfahren zum Freigeben des Testens der Konfigurationsspeicheranordnung einer Logikvorrichtung nach Anspruch 7, wobei die Testmuster "Alle 0"-, "Alle 1"-, "Wandernde 0"- und "Wandernde 1"-Testmuster beinhalten.

Es folgen 6 Blatt Zeichnungen

Anhängende Zeichnungen

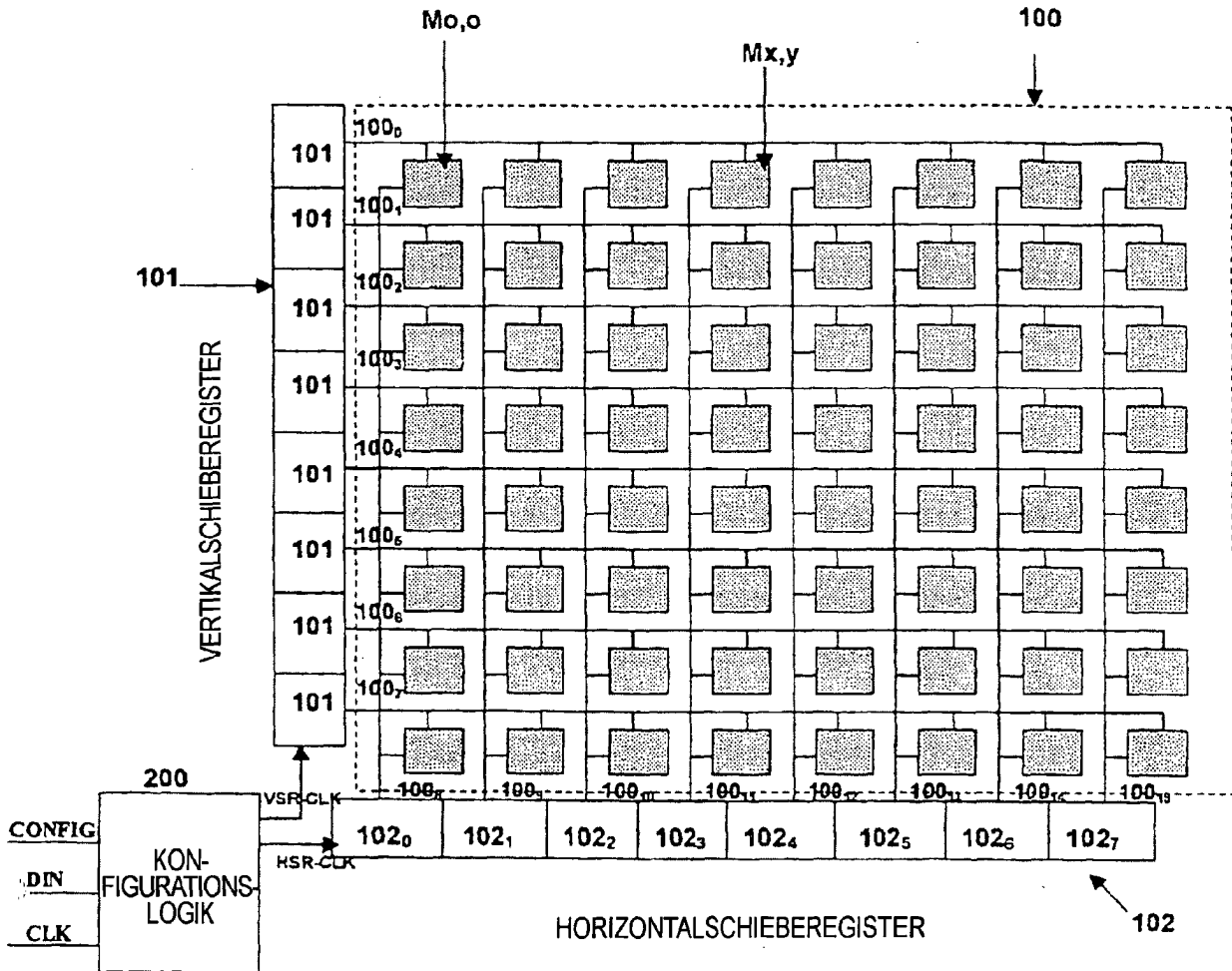


FIG 1A (STAND DER TECHNIK)



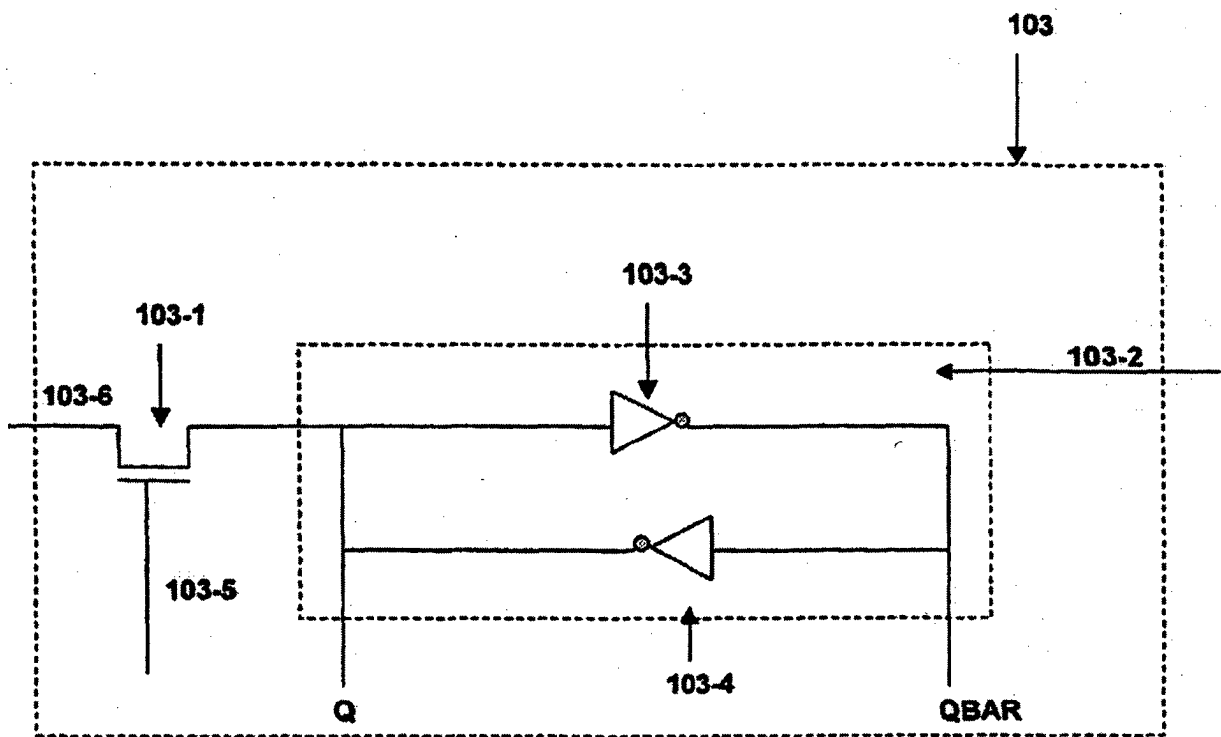


FIG 1B (STAND DER TECHNIK)

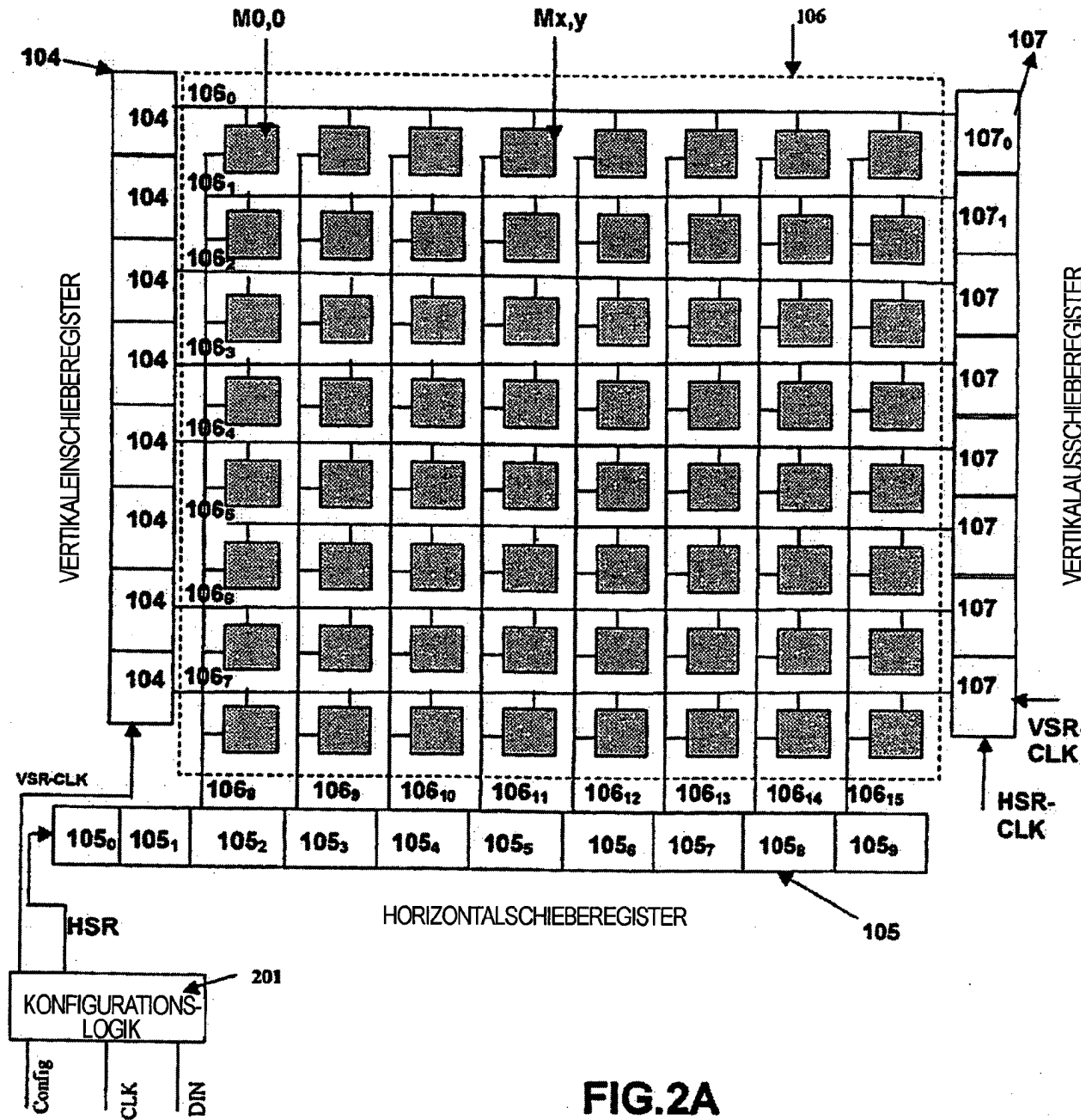


FIG.2A

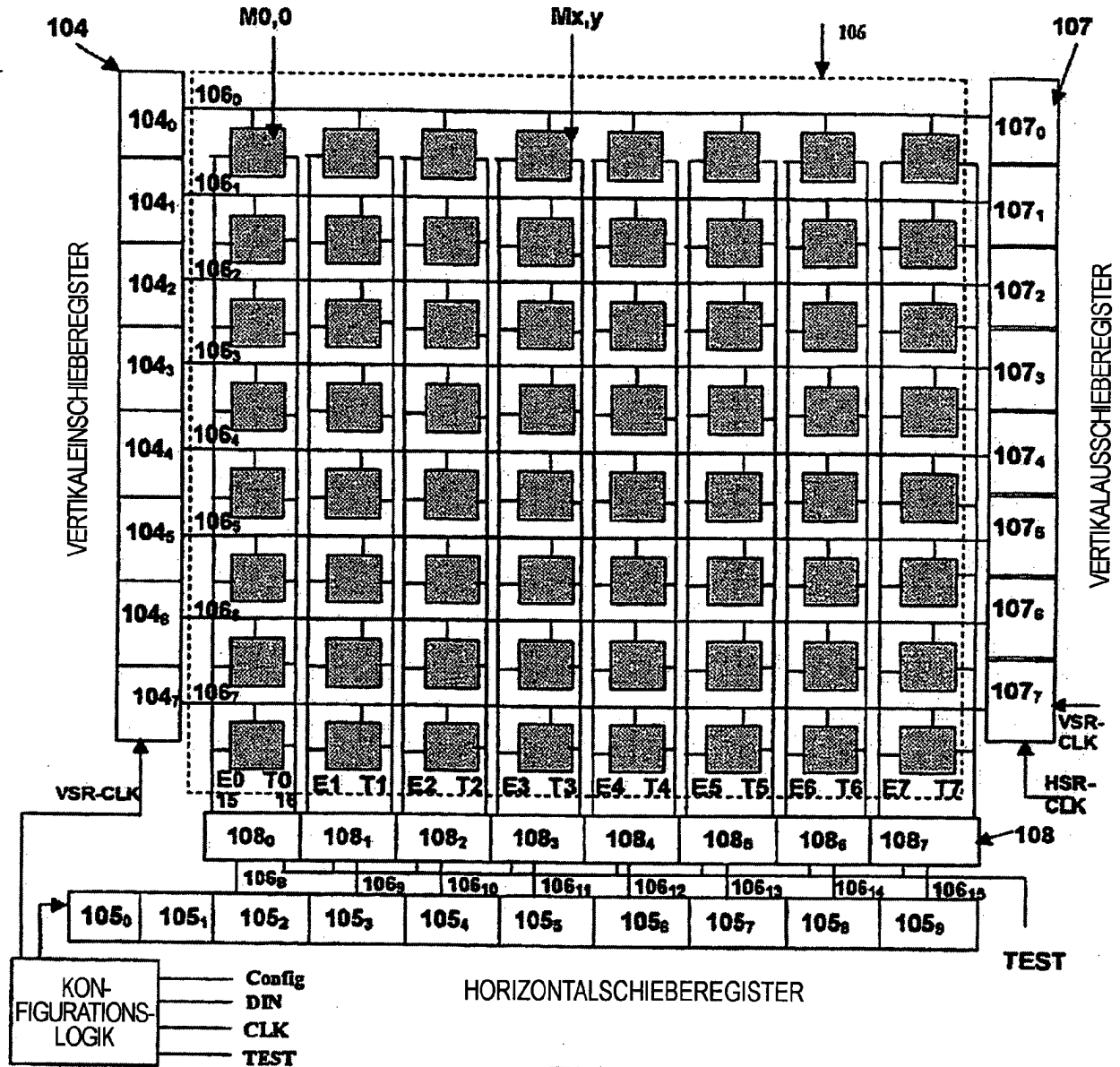
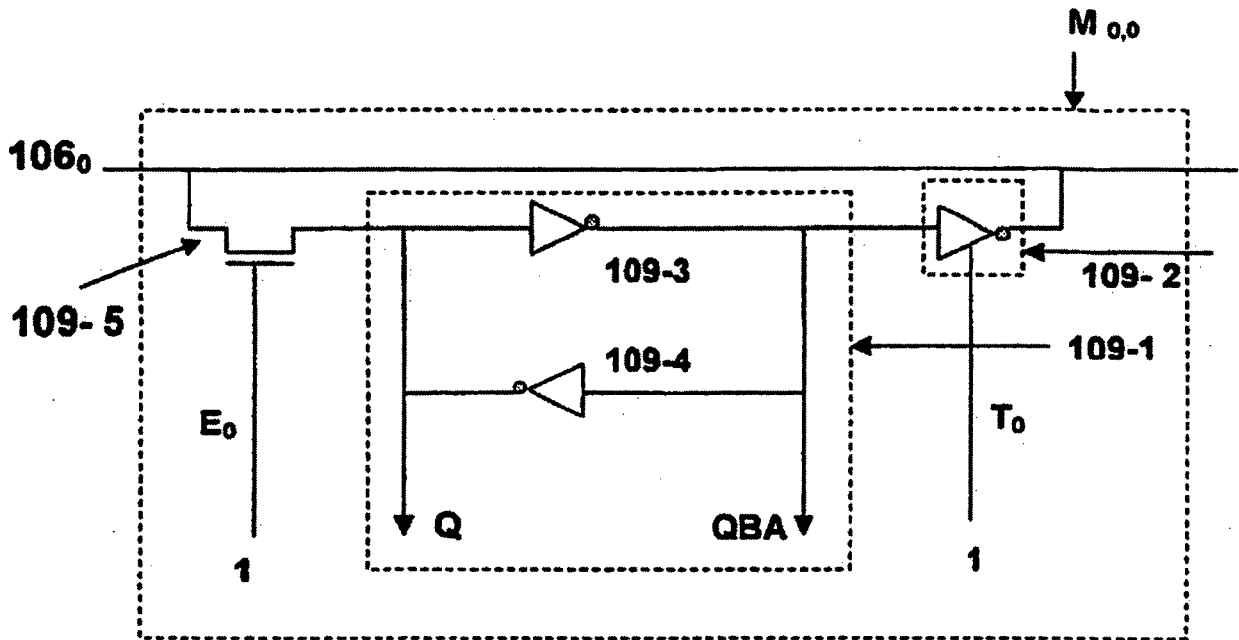
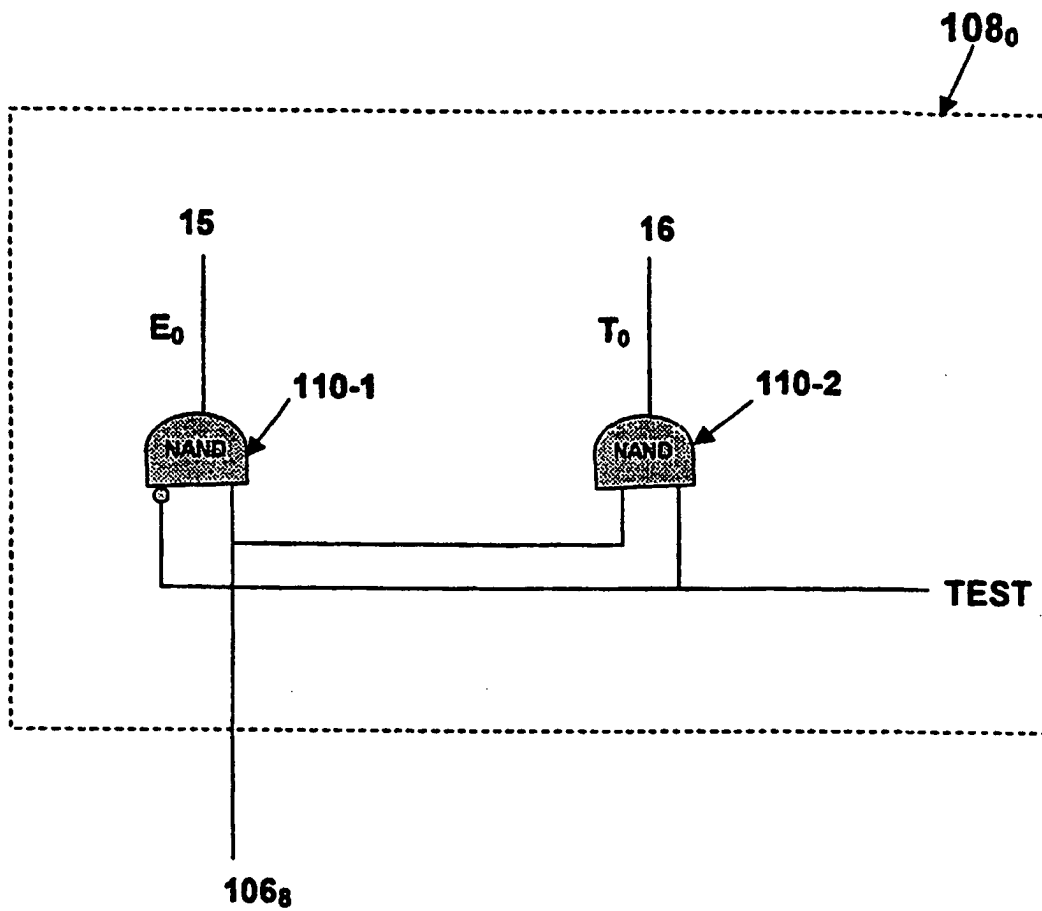


FIG.2B



**FIG 2C**



**FIG 2D**