



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 698 24 765 T2** 2005.07.21

(12)

Übersetzung der europäischen Patentschrift

(97) **EP 1 324 234 B1**

(51) Int Cl.7: **G06F 17/50**

(21) Deutsches Aktenzeichen: **698 24 765.5**

(96) Europäisches Aktenzeichen: **03 075 900.5**

(96) Europäischer Anmeldetag: **28.10.1998**

(97) Erstveröffentlichung durch das EPA: **02.07.2003**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **23.06.2004**

(47) Veröffentlichungstag im Patentblatt: **21.07.2005**

(30) Unionspriorität:

960715 30.10.1997 US

(84) Benannte Vertragsstaaten:

DE, FI, FR, GB, IT, SE

(73) Patentinhaber:

Chapman, David C., Santa Clara, Calif., US

(72) Erfinder:

Chapman, David C., Santa Clara, US

(74) Vertreter:

Weickmann & Weickmann, 81679 München

(54) Bezeichnung: **Polygondarstellung im Layout eines integrierten Schaltkreises**

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 die Konstruktion integrierter Schaltungen, und insbesondere ein Verfahren und eine Vorrichtung zum Darstellen von Polygonen in einem Integrierte-Schaltungs-Layout.

HINTERGRUND DER ERFINDUNG

[0002] Im Kontext der Konstruktion integrierter Schaltungen (IC) ist eine "Auslegung" bzw. ein "Layout" ein Satz geometrischer Muster, typischerweise in der Form von Polygonen, die die Größe und den Ort unterschiedlicher Materialtypen spezifizieren, die dazu benutzt werden, um Halbleitervorrichtungen und elektrische Verbindungen zwischen den Vorrichtungen während der Herstellung eines IC zu erzeugen. Z. B. kann ein Diffusionsfenster auf einem IC in einem Layout durch ein oder mehrere Polygone dargestellt werden, die durch eine Fabrikationseinrichtung mit der Bedeutung von "Diffusionsschichtgeometrie" interpretiert werden. Andere Materialschichten und Merkmale, wie etwa Kontakte und Bahnen, können auf ähnliche Weise in einem IC-Layout dargestellt werden. Die Polygone eines IC-Layouts müssen einen Satz von Konstruktionsregeln erfüllen, der die minimalen Größen für bestimmte Materialtypen sowie die minimalen Platzanforderungen zwischen den unterschiedlichen Materialtypen definiert. Der Satz der Konstruktionsregeln spezifiziert auch die Größe und die Abstandsanforderungen für andere Layoutmerkmale, wie etwa Kontakte.

[0003] Manchmal werden IC-Layouts mittels eines computerunterstützten Design (CAD)-Systems von Hand gezeichnet. Häufig ist dieser Ansatz für die Konstruktion IC-Layouts angemessen, die eine relativ kleine Anzahl von Vorrichtungen aufweisen. Jedoch erlaubt die moderne Herstellungstechnologie, dass mehrere Millionen Transistoren auf einem einzigen Siliciumsubstrat angeordnet werden. Das Zeichnen eines IC-Layouts von Hand, das Millionen von Transistoren enthält, nicht praktikabel, weil der Zeitaufwand übermäßig ist, auch wenn standardisierte Zellen dazu benutzt werden, die Anzahl einzelner Transistoren zu reduzieren, die gezeichnet werden müssen. Demzufolge sind verschiedene IC-Layoutkonstruktionswerkzeuge auf Computerbasis entwickelt worden, um das Generieren von IC-Layouts zu automatisieren. Zwei Typen von Layoutkonstruktionswerkzeugen enthalten Astzellen-Kompaktierungswerkzeuge und Astzellen-Synthesewerkzeuge. Im Kontext von IC-Layouts bezieht sich der Begriff "Astzelle" auf eine Gruppe von Transistoren, typischerweise von zwei bis mehreren Hundert, die zusammen eine spezifische Funktion haben, wie etwa eine logische NAND-Operation oder die Speicherung eines Informationsbits.

[0004] Astzellen-Kompaktierungswerkzeuge werden dazu benutzt, die Größe eines existierenden IC-Layouts zu reduzieren. Die meisten Astzellen-Kompaktierungswerkzeuge kompaktieren ein Layout entweder in der vertikalen oder horizontalen Richtung, indem Extraraum zwischen den Polygonen eliminiert wird. Eine "volle" Kompaktierung wird manchmal erreicht, indem zuerst ein Layout in der horizontalen Richtung komprimiert wird und dann das Layout in der vertikalen Richtung komprimiert wird. Jedoch sind die meisten Astzellen-Kompaktierungswerkzeuge auch in der Lage, die Koordinaten von Polygonpunkten in einem IC-Layout einzustellen, bis sie die Konstruktionsregeln für einen bestimmten Herstellungsprozess erfüllen, was diese hilfreich macht, um ein existierendes Layout von einem gegenwärtigen Satz von Konstruktionsregeln auf einen neuen Satz von Konstruktionsregeln zu übertragen.

[0005] Trotz der Vorteile der Astzellen-Kompaktierungswerkzeuge ist auch ein kompaktiertes Layout größer als vergleichbare handgezeichnete Zellen, weil die Kompaktierung allgemein ein eindimensionaler Prozess ist. Auch halten Kompaktierungswerkzeuge die existierende Layoutstruktur bei, obwohl durch nicht-lineare Änderungen in den Konstruktionsregeln (z. B. eine proportional vergrößerte Metallüberlappung um einen Kontakt) häufig bewirkt, dass menschliche Layoutkonstrukteure unterschiedliche Entscheidungen treffen, wenn sie das Layout in diesen neuen Konstruktionsregeln zeichnen.

[0006] Im Gegensatz zu Astzellen-Kompaktierungswerkzeugen haben Astzellen-Synthesewerkzeuge die Fähigkeit zum Erzeugen eines neuen Astzellenlayouts auf der Basis einer Netzliste auf Transistorebene, die die Größe von Transistoren und der elektrischen Verbindungen zwischen den Transistoren spezifiziert. Weil das Layoutsynthesystem eine existierende Layoutstruktur nicht beibehalten braucht, kann sie den vollen Vorteil der Konstruktionsregeln oder Schaltungsanforderungen nutzen, z. B. durch Umordnen von Transistoren, Platzieren von Transistoren in abwechselnden Reihen oder Umordnen der Lenkung dort, wo Betriebsmittel verfügbar werden. Viele Layoutsynthesysteme verwenden die Kompaktierung als letztendlichen Schritt bei der Erzeugung eines Layouts, aber sie können noch kompaktere Layouts erzeugen als ihre Gegenstücke, die nur existierende Layouts kompaktieren.

[0007] Die meisten IC-Layoutkonstruktionswerkzeuge erzeugen eine interne Darstellung der in einem Layout enthaltenen Polygone. Zwei Ansätze zum Darstellen von Polygonen in einem IC-Layout beinhalten den allgemeinen Polygonansatz und den Ader- oder Pfadansatz.

[0008] Der allgemeine Polygonansatz verwendet eine einzige Liste von Punkten, die Rechtecke oder

Trapeze repräsentieren, die zum Konstruieren der in einem IC-Layout enthaltenen Polygone verwendet werden. Jedes Rechteck oder Trapez hat parallele Seiten mit der gleichen Orientierung und ist durch vier Punkte definiert. Komplexe Polygone werden dargestellt, indem mehrere Trapeze einander benachbart angeordnet werden. Der allgemeine Polygonansatz ist beliebt und sorgt für die Flexibilität zur Darstellung jeder Layoutgeometrie. Es gibt keine bevorzugte Orientierung oder "Abwicklungsrichtung" in den Trapezen. D. h. ein Trapez wird nicht in Bezug auf eine bestimmte Ecke referenziert, und die Seiten eines Trapezes sind nicht geordnet (z. B. in Bezug auf den Stromfluss). Jedoch macht es die beliebige Natur schwierig, dass der allgemeine Polygonansatz in Software implementiert wird. Weil die meiste Synthesoftware die Anzahl von Punkten begrenzt, die in einem einzigen Polygon enthalten sein können, müssen einige komplexe Polygone, die zahlreiche Trapeze enthalten, in kleinere Polygone aufgeteilt werden. Das Erfordernis, ein Polygon in viele Trapeze oder Rechtecke aufzuteilen, überdeckt ihren Zweck und es ist schwierig, nützliche Eigenschaften, wie etwa die Größe eines Widerstands oder einer Transistorvorrichtung zu bestimmen. Auch können relativ einfache Änderungen in den Polygonen dazu führen, dass die Trapezdarstellung insgesamt restrukturiert wird. Viele Werkzeuge verwenden nur Rechtecke, weil die Algorithmen, die zur Handhabung aus Trapezen gebauter, nicht orthogonaler Geometrie erforderlich sind, kompliziert sind. Es sind große Mengen an Trapezen oder Rechtecken erforderlich, um alle außer die einfachsten Polygone darzustellen, und nützliche Operationen, wie das Einfügen einer Biegung in einen Ader, sind schwierig, weil sie keinen direkten Analog in einer Liste verketteter Trapeze haben. Schließlich repräsentieren einige Implementierungen des allgemeinen Polygonansatzes den gesamten Konstruktionsraum, einschließlich eines etwa ungenutzten Raums zwischen den Polygonen, was die Anzahl der Polygone erhöht und mehr Daten erfordert, um das Layout darzustellen.

[0009] Der Ader- oder Pfadansatz repräsentiert Polygone unter Verwendung einer Mittellinie und einer Polygonbreite für jeden Punkt entlang der Mittellinie. Die Einfachheit dieses Ansatzes lässt sich in Synthesewerkzeugen leicht implementieren und kann das Lenken (Rooting) zwischen Diffusionsinseln vereinfacht. Auch können die Breite und der Widerstand eines Pfads leicht errechnet werden. Die meiste Layoutgeometrie kann dargestellt werden, ohne Adern zu verketteten oder Adern nächst beieinander anzuordnen, und dieser Ansatz wird in vielen Rooting-Werkzeugen verwendet. Jedoch erfordert der Pfadansatz, dass Polygone parallele Seiten haben, was ihn für komplexere Formen ungeeignet macht, wie etwa achteckige Kontakte, unregelmäßig geformte Diffusionsinseln oder -adern mit veränderlicher Breite.

[0010] Im Hinblick auf das Erfordernis, Polygone in IC-Layoutsynthesewerkzeugen darzustellen und die Einschränkungen in existierenden Ansätzen ist ein verbessertes Verfahren zum Vorbestimmen der kanonischen Richtung aus einem gerichteten Liniensegment zu einem beliebigen Punkt in einer integrierten Schaltungsauslegung besonders erwünscht.

ZUSAMMENFASSUNG DER ERFINDUNG

[0011] Gemäß einem Aspekt der Erfindung, wie er in den Ansprüchen 1 bis 3 ausgeführt ist, wird im Wesentlichen ein Verfahren zum Vorbestimmen der kanonischen Richtung aus einem gerichteten Liniensegment zu einem beliebigen Punkt in einer integrierten Schaltungsauslegung vorgesehen, wobei das Verfahren die Schritte umfasst: a) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen Punkt auf dem gerichteten Liniensegment; b) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen ersten Endpunkt des gerichteten Liniensegments; und c) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen zweiten Endpunkt des gerichteten Liniensegments; und d) auf der Basis der Ergebnisse der Schritte a) bis c), Zuweisen einer kanonischen Richtung von dem gerichteten Liniensegment zu dem beliebigen Punkt.

KURZBESCHREIBUNG DER ZEICHNUNGEN

[0012] Ausführungen der Erfindung werden als Beispiel und keineswegs als Einschränkung in den Figuren der beigefügten Zeichnungen dargestellt, worin gleiche Bezugszahlen ähnliche Elemente betreffen, und worin:

[0013] [Fig. 1](#) zeigt eine kanonische Orientierung gemäß einer Ausführung der Erfindung;

[0014] [Fig. 2](#) zeigt ein gerichtetes Liniensegment und einen benachbarten Punkt gemäß einer Ausführung der Erfindung;

[0015] [Fig. 3](#) zeigt eine Ader gemäß einer Ausführung der Erfindung;

[0016] [Fig. 4A](#) zeigt eine Punktdatenstruktur gemäß einer Ausführung der Erfindung;

[0017] [Fig. 4B](#) zeigt eine Gerichtetes-Liniensegment-Datenstruktur gemäß einer Ausführung der Erfindung;

[0018] [Fig. 4C](#) zeigt eine Adersegmentdatenstruktur gemäß einer Ausführung der Erfindung;

[0019] [Fig. 4D](#) zeigt eine Aderdatenstruktur gemäß einer Ausführung der Erfindung;

[0020] [Fig. 4E](#) zeigt eine Polygondatenstruktur ge-

mäß einer Ausführung der Erfindung;

[0021] [Fig. 4F](#) zeigt eine Zweigdatenstruktur gemäß einer Ausführung der Erfindung;

[0022] [Fig. 4G](#) zeigt eine Seitenzweigdatenstruktur gemäß einer Ausführung der Erfindung;

[0023] [Fig. 5](#) zeigt ein aus Adersegmenten aufgebautes Polygon gemäß einer Ausführung der Erfindung;

[0024] [Fig. 6A–Fig. 6E](#) zeigen einen Satz von Datenstrukturen, die dem Polygon von [Fig. 5](#) zugeordnet sind, gemäß einer Ausführung der Erfindung;

[0025] [Fig. 7](#) zeigt ein aus Adersegmenten aufgebautes zweites Polygon gemäß einer Ausführung der Erfindung;

[0026] [Fig. 8](#) zeigt eine aus Adersegmenten aufgebaute Diffusionsinsel gemäß einer Ausführung der Erfindung;

[0027] [Fig. 9A](#) zeigt ein gerichtetes Liniensegment und einen Orientierungsschlüssel gemäß einer Ausführung der Erfindung;

[0028] [Fig. 9B](#) zeigt ein Adersegment und einen Orientierungsschlüssel gemäß einer Ausführung der Erfindung;

[0029] [Fig. 10](#) zeigt ein mittels Adersegmenten aufgebauten Abschnitt eines IC-Layouts gemäß einer Ausführung der Erfindung;

[0030] [Fig. 11](#) zeigt die Verwendung einer Ausschlusszone zur Bestimmung des Polygonabstands gemäß einer Ausführung der Erfindung;

[0031] [Fig. 12](#) ist ein Flussdiagramm mit Darstellung eines Verfahrens zur Durchführung einer Abstandsprüfung zwischen Polygonen gemäß einer Ausführung der Erfindung;

[0032] [Fig. 13A](#) zeigt ein mittels Adersegmenten aufgebaute Ader gemäß einer Ausführung der Erfindung;

[0033] [Fig. 13B](#) zeigt eine Polygondarstellung der Ader von [Fig. 13A](#) gemäß einer Ausführung der Erfindung; und

[0034] [Fig. 14](#) ist ein Blockdiagramm eines Computersystems, an dem Ausführungen der Erfindung implementiert werden können.

DETAILLIERTE BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNG

[0035] Es wird ein neuartiger Ansatz zur Darstellung von Polygonen in einem IC-Layout beschrieben. In der folgenden Beschreibung werden zu Erläuterungszwecken spezifische Details aufgeführt, um für ein gründliches Verständnis der Erfindung zu sorgen. Jedoch wird es ersichtlich, dass die Erfindung ohne diese spezifischen Details praktiziert werden kann. In anderen Umständen sind gutbekannte Strukturen und Vorrichtungen in Blockdiagrammform abgebildet, um ein unnötiges Überdecken der Erfindung zu vermeiden.

FUNKTIONELLE ÜBERSICHT

[0036] Allgemein werden gemäß einer Ausführung der Erfindung Polygone eines IC-Layouts durch ein oder mehrere Adern dargestellt, die jeweils durch ein oder mehrere Adersegmente dargestellt werden. Jedes Adersegment ist aus einem Paar gerichteter Liniensegmente gebildet. Eine neuartige Datenstrukturhierarchie wird vorgesehen, um Daten beizubehalten, die Adern, Adersegmente und gerichtete Liniensegmente darstellen. Die Datenstrukturhierarchie wird während der IC-Layoutsynthese benutzt, um an dem IC-Layout Gültigkeitsprüfungen durchzuführen.

DETAILLIERTE BESCHREIBUNG

1) Kanonische Orientierung

[0037] Wie in [Fig. 1](#) dargestellt, bezieht sich die kanonische Orientierung allgemein auf die Richtung der IC-Layoutgeometrie in Bezug auf einen vorbestimmten Orientierungsschlüssel **100**. Gemäß einer Ausführung der Erfindung kann die IC-Layoutgeometrie in einer von acht kanonischen Orientierungen orientiert werden, wobei jede kanonische Orientierung ein Mehrfaches von 45° darstellt. Beginnt man an der Oberseite des Orientierungsschlüssels **100** und schreitet im Uhrzeigersinn fort, enthalten die kanonischen Orientierungen (oben), (oben rechts), (rechts), (unten rechts), (unten), (unten links), (links) und (oben links).

[0038] Zusätzlich zu der in einer kanonischen Orientierung orientierten IC-Layoutgeometrie wird die Beziehung zwischen IC-Layoutmerkmalen auch im Hinblick auf eine kanonische Richtung beschrieben. Z. B. ist, wie in [Fig. 2](#) dargestellt, ein Liniensegment **100** durch Endpunkte **202** und **204** definiert. Gemäß einer Ausführung der Erfindung beruht die kanonische Orientierung des Liniensegments **200** auf der Position des Endpunkts **204** in Bezug auf den Endpunkt **202**. Somit ist das Liniensegment **200** in Richtung (oben rechts) orientiert oder gerichtet, wie mit dem Richtungspfeil **205** angegeben. Es gibt keine Einschränkungen auf die Richtung, in der ein Liniensegment

zeigen kann. Wenn z. B. der Endpunkt **204** unterhalb und links des Endpunkts **202** ist, dann wäre die kanonische Orientierung des Liniensegments **200** (unten links). Zusätzlich ist der Punkt **206** links (links) des Liniensegments **200** angeordnet, während der Punkt **208** oberhalb und rechts (oben rechts) des Liniensegments **200** angeordnet ist. Dieser Ansatz ist zur Bestimmung nützlich, ob der Abstand zwischen dem Liniensegment **200** und dem Punkt **206** einem Satz vorbestimmter Abstandskriterien genügt, wie nachfolgend beschrieben wird.

[0039] Wie zuvor eingeführt, werden IC-Layoutpolygone durch ein oder mehrere Adern dargestellt, und jede Ader wird durch ein oder mehrere Adersegmente dargestellt. Gemäß einer Ausführung der Erfindung hat jedes Adersegment vier Seiten, zwei seitlichen Seiten (links und rechts) und zwei Endseiten (erste und zweite). Die seitlichen Seiten sind durch ein Paar gerichteter Ränder ausgedrückt, die im Wesentlichen in derselben Richtung orientiert sind, aber die nicht notwendigerweise parallel sind oder zusammenfallende Endpunkte aufweisen. Gemäß einer Ausführung der Erfindung müssen die linken und rechten Seiten jeweils ein Mehrfaches eines 45° Winkels in Bezug aufeinander haben, wobei aber die Endseiten jeden Winkel in Bezug zueinander haben können. Allgemein haben die Endseiten ein Mehrfaches von 45° an den freien Enden der Ader, während "interne" Endseiten jeden Winkel haben können. Wie bei den Adersegmenten können die Adersegmente in jede Richtung zeigen, wie zuvor in Bezug auf [Fig. 1](#) beschrieben. Gemäß einer Ausführung der Erfindung brauchen nicht alle Punkte eines Adersegments zusammenfallen, um sicherzustellen, dass jede Seite eine Nicht-null-Länge und eine definierte Richtung hat.

[0040] [Fig. 3](#) zeigt eine Ader **300**, die aus einem Adersegment **302** und einem Adersegment **304** aufgebaut ist, die zu einem Adersegment **302** mit einem 45° Winkel verknüpft sind, gemäß einer Ausführung der Erfindung. Das Adersegment **302** ist aufgebaut aus einem ersten gerichteten Rand **306**, der die linke Seite des Adersegments **302** bildet, und einem zweiten gerichteten Rand **308**, der die rechte Seite des Adersegments **302** bildet. Das Adersegment **302** enthält auch eine erste Endseite **310** und eine zweite Endseite **312**. Beide gerichteten Ränder **306** und **308** zeigen in der (unten) Richtung. Daher zeigt auch das Segment **302** in der (unten) Richtung.

[0041] Das Adersegment **304** ist aus einem ersten gerichteten Rand **314** und einem zweiten gerichteten Rand **416** aufgebaut. Beide gerichteten Ränder **314**, **316** sind in der (unten, rechts) Richtung orientiert, so dass das Adersegment **304** in der (unten, rechts) Richtung orientiert wird. Das Adersegment **304** enthält eine erste Endseite **312**, die mit der zweiten Endseite des Adersegments **302** zusammenfällt, und

eine zweite Endseite **318**.

2) Datenstrukturen

[0042] Gemäß einer Ausführung der Erfindung wird eine neuartige Datenstrukturarchitektur angegeben, um Polygone in einem IC-Layout darzustellen, wie in den [Fig. 4A-Fig. 4G](#) gezeigt. Gemäß einer Ausführung der Erfindung wird jedes Polygon in einem IC-Layout durch ein oder mehrere Adern dargestellt. Jede Ader wird durch ein oder mehrere Adersegmente dargestellt, die jeweils durch ein Paar gerichteter Liniensegmente dargestellt werden. Jedes Adersegment referenziert die Ader, in der es enthalten ist, und jede Ader referenziert ein Polygon, in der sie enthalten ist. Es kann bei gegebener Referenz auf ein einziges Adersegment ein gesamtes Polygon lokalisiert und manipuliert werden.

[0043] [Fig. 4A](#) zeigt den Kontext einer Punktdatenstruktur **400** gemäß einer Ausführung der Erfindung. Die Punktdatenstruktur **400** spezifiziert den Ort eines Punkts in einem IC-Layout, indem die X- und Y-Koordinaten (X KOORDINATE, Y KOORDINATE) für einen Punkt in einem IC-Layout spezifiziert werden.

[0044] [Fig. 4B](#) zeigt die Inhalte einer Gerichtetes-Liniensegment-Datenstruktur **410**. Die Gerichtetes-Liniensegment-Datenstruktur **410** spezifiziert sowohl einen ersten Endpunkt (ENDE 1) als auch einen zweiten Endpunkt (ENDE 2) des entsprechenden gerichteten Liniensegments. Die Gerichtetes-Liniensegment-Datenstruktur **410** spezifiziert auch eine Richtung (RICHTUNG) eines gerichteten Liniensegments. Wie zuvor beschrieben, kann ein gerichtetes Liniensegment in einer von acht vordefinierten Richtungen gerichtet sein, wie in [Fig. 1](#) gezeigt.

[0045] Die Gerichtetes-Liniensegment-Datenstruktur **410** spezifiziert auch ein Orthoflag (ORTHO) und ein Oktantflag (OKTANT). Das Orthoflag ist ein BOOL'SCHES Flag, das angibt, ob ein gerichtetes Liniensegment mit einem Winkel ausgerichtet ist, der ein Mehrfaches von 90° beträgt, während das Oktantflag ein BOOL'SCHES Flag ist, das spezifiziert, ob ein gerichtetes Liniensegment in einem Winkel ausgerichtet ist, der ein Mehrfaches von 45° beträgt. Wenn, gemäß einer Ausführung der Erfindung, weder das Ortho- noch das Oktantflag gesetzt ist, dann ist das gerichtete Liniensegment mit einem "schiefen" Winkel orientiert, d. h. einem anderen Winkel als 45° oder 90° , der unter gewissen Umständen eine Gleitpunktarithmetik erfordert, um Koordinaten des gerichteten Liniensegments zu berechnen.

[0046] [Fig. 4C](#) zeigt die Inhalte einer Adersegmentdatenstruktur **420**, der einen linken Rand (linken RAND) und einen rechten Rand (rechten RAND) eines Adersegments spezifiziert. Sowohl der linke Rand als auch der rechte Rand referenzieren jeweils

eine Gerichtetes-Liniensegment-Datenstruktur **410**. Die Adersegmentdatenstruktur **420** enthält auch eine Referenz auf das nächste Adersegment (NÄCHSTES ADERSEGMENT) und das vorherige Adersegment (VORHERIGES ADERSEGMENT) in der Ader, unter Bildung einer doppelt verknüpften Liste. Die Adersegmentdatenstruktur für das erste Adersegment in einer Ader spezifiziert einen vorbestimmten Wert, um anzugeben, dass sich das Adersegment nicht auf ein vorheriges Adersegment bezieht. Gemäß einer Ausführung der Erfindung wird als der vorbestimmte Wert ein NULL-Wert verwendet. Ähnlich spezifiziert die Adersegmentdatenstruktur für das letzte Adersegment in einer Ader einen vorbestimmten Wert zur Angabe, dass das Adersegment sich nicht auf ein nächstes Adersegment bezieht. Obwohl eine Ausführung der Aderdatenstruktur **420** im Kontext einer doppelt verknüpften Liste beschrieben, ist keine doppelt verknüpfte Liste erforderlich.

[0047] Die Adersegmentdatenstruktur **420** enthält auch eine Referenz auf eine Stammader (STAMMADER), die das Adersegment enthält, sowie einen Bezug (ABZWEIGUNG) auf eine Zweigdatenstruktur, die relevant wird, wenn eine freiliegende Endseite des gegenwärtigen Adersegments eine SeitenSeite einer anderen Ader berührt, zur Bildung einer Verzweigung. Inhalte einer Zweigdatenstruktur werden nachfolgend beschrieben. Zweigdatenstrukturen sind nicht Adersegmenten zugeordnet, die keine Verzweigung aufweisen, um Speicher einzusparen.

[0048] [Fig. 4D](#) zeigt die Inhalte einer Aderdatenstruktur **430**. Die Aderdatenstruktur **430** spezifiziert einen Bezug auf ein Stammpolygon (STAMMPOLYGON) sowie eine Zweigzahl (ZWEIGZAHL), die die Ader identifiziert. Die Aderdatenstruktur **430** spezifiziert auch einen Begrenzungskasten (BBX), der alle Adersegmente der Ader umschließt. Ein Begrenzungskasten erlaubt schnelle Bereichsabfrageprüfungen, sodass auch dann, wenn eine Ader durch eine Bereichsabfrage zurückgebracht wird, wenn kein Adersegment dem Anfragepunkt nahe genug ist, dann kein Segment in der Ader geprüft zu werden braucht. Gemäß einer Ausführung der Erfindung wird ein Begrenzungskasten (BBX) als vier ganze Zahlen entsprechend den linken, unteren, rechten und oberen Koordinaten des Begrenzungskastens implementiert.

[0049] Die Aderdatenstruktur **430** enthält auch einen Bezug auf das erste Adersegment (ERSTES ADERSEGMENT) und das letzte Adersegment (LETZTES ADERSEGMENT) in der Ader sowie einen Bezug auf ein an dem ersten Ende angebrachte Ader (AM ERSTEN ENDE ANGEBRACHTE ADER) und eine am zweiten Ende angebrachte Ader (AM ZWEITEN ENDE ANGEBRACHTE ADER).

[0050] [Fig. 4E](#) zeigt die Inhalte einer Polygondaten-

struktur **440**, die einen Begrenzungskasten (BBX) in einem Feld von Zweigzahlen (FELD VON ZWEIGZAHLN) spezifiziert, das alle Adersegmente eines Polygons durch eine Zweigzahl spezifiziert, wie zuvor in Bezug auf die Aderdatenstruktur **430** beschrieben. Gemäß einer Ausführung der Erfindung enthält die Polygondatenstruktur **440** auch einen Bezug auf die entsprechende Prozessschicht (SCHICHT) für das Polygon sowie die Polygonzahl (POLYGONZAHL), die zum eindeutigen Identifizieren des Polygons verwendet wird. Z. B. kann diese Information mit einer Liste verwendet werden, die alle Polygone für eine bestimmte Prozessschicht spezifiziert.

[0051] [Fig. 4F](#) zeigt die Inhalte einer Zweigdatenstruktur **450**, die verwendet wird, wenn eine Abzweigung in einem Polygon gebildet wird, gemäß einer Ausführung der Erfindung. Eine Abzweigung wird gebildet, wenn eine freie Endseite eines Adersegments eine seitliche Seite einer anderen Ader berührt. Die Zweigdatenstruktur **450** spezifiziert eine Referenz auf ein Stammadersegment (STAMMADERSEGMENT) sowie Referenzen auf andere Adersegmente, die an dem ersten Ende (ZWEIGENDE 1) bzw. dem zweiten Ende (ZWEIGENDE 2) des Adersegments angebracht sind. Die Zweigstruktur **450** enthält einen Bezug auf erste und letzte Zweige (ERSTER linker ZWEIG, LETZTER linker ZWEIG) und erste und letzte rechte Zweige (ERSTER rechter ZWEIG, LETZTER rechter ZWEIG) aus einem Adersegment. D. h. diese Datenstrukturelemente referenzieren eine Seitenzweigstruktur, wenn ein bestimmtes Adersegment andere Adersegmente aufweist, die sich mit dem Adersegment an einem seiner Seiten schneiden. Gemäß einer Ausführung der Erfindung kann jede Anzahl von Zweigen an einer seitlichen Seite eines Adersegments angebracht sein, wobei aber auch die Endseite einer Ader an nur einem anderen Adersegment angebracht sein kann.

[0052] [Fig. 4G](#) zeigt die Inhalt einer Seitenzweigstruktur **460**, die eine Referenz auf ein Adersegment (ANDERES ADERSEGMENT) spezifiziert, das ein Teil des an dem Stammadersegment angebrachten Zweigs ist, sowie das Liniensegment (STAMMLINIENSEGMENT) in dem Stammadersegment, an dem das andere Adersegment angebracht ist. Die Seitenzweigstruktur **460** spezifiziert, ob ein anderes Adersegment an der linken Seite des Stammadersegments angebracht ist, auf der Basis eines BOOL'SCHEN Flags (AM linkem RAND). Die Seitenzweigstruktur **460** enthält auch eine Referenz auf die Datenstruktur des nächsten Seitenzweigs (NÄCHSTER SEITENZWEIG) und des vorherigen Seitenzweigs (VORHERIGER SEITENZWEIG) sowie den ersten Seitenpunkt (ERSTER SEITENPUNKT) und den zweiten Seitenpunkt (ZWEITER SEITENPUNKT), die die Orte auf dem Stammadersegment spezifizieren, wo die anderen Adersegmente angebracht sind. Die Seitenpunkte unterstützen

die Sortierung der Seitenzweiglisten; der erste Seitenpunkt ist dem Ende1 des Stammadersegments näher.

[0053] Die vorgenannten Datenstrukturen werden nun in Bezug auf [Fig. 5](#) und ein Polygon **500** beschrieben, das aus drei Adern, ADER A, ADER B und ADER C, aufgebaut ist, gemäß einer Ausführung der Erfindung. Jede Ader, ADER A, ADER B und ADER C, besteht aus einem einzigen Adersegment, identifiziert als ADERSEGMENT A, ADERSEGMENT B und ADERSEGMENT C.

[0054] Das ADERSEGMENT A ist aus zwei gerichteten Liniensegmenten a_1 und a_2 aufgebaut, die beide in der (oben) Richtung ausgerichtet sind, wie durch die Richtungspfeile angegeben. Das ADERSEGMENT B ist aus gerichteten Liniensegmenten b_1 und b_2 aufgebaut, die beide in der (linken) Richtung ausgerichtet sind. Die ADER B ist mit der ADER A an dem gerichteten Liniensegment a_1 an einem Ort verknüpft, der durch die Punkte d_1 und d_2 definiert ist. Das ADERSEGMENT C ist aus gerichteten Liniensegmenten c_1 und c_2 aufgebaut, die beide in der (rechts) Richtung ausgerichtet sind. Die ADER C verbindet sich mit ADER A auf dem gerichteten Liniensegment a_1 an einem Ort, der durch die Punkte e_1 und e_2 definiert ist. Begrenzungskästen BBX_A , BBX_B , BBX_C definieren die Inhalte von ADER A, ADER B bzw. ADER C, und der Begrenzungskasten BBX_P definiert die Inhalte des Polygons **500**.

[0055] Gemäß einer Ausführung der Erfindung teilen sich alle Adern eines Polygons eine logische Zuordnung. Z. B. können für ein bestimmtes Polygon alle Adern physikalisch durchgängig sein und sich elektrische Eigenschaften teilen. Für ein anderes Polygon brauchen sich die Adern keine physikalische Zuordnung teilen, sondern können elektrisch verbunden sein, z. B. im Falle einer Gruppe von Kontakten. Andererseits kann ein Satz von Adern logisch in ein Polygon gruppiert sein, weil es zweckdienlich ist, dass sich diese Adern ein einziges Polygon teilen. Obwohl z. B. nicht physikalisch oder elektrisch zugeordnet, können alle Polysiliciumgatter auf einer Diffusionsinsel in einem einzigen Polygon enthalten sein, weil es zweckdienlich ist, alle Polysiliciumgatter auf einer Diffusionsinsel zusammen zu manipulieren.

[0056] Der Satz von Datenstrukturen, die zur Darstellung des Polygons **500** gemäß einer Ausführung der Erfindung erzeugt sind, ist in den [Fig. 6A–Fig. 6E](#) gezeigt. [Fig. 6A](#) zeigt eine Polygondatenstruktur **600** für das Polygon **500**. Die Polygondatenstruktur **600** spezifiziert, dass das Polygon **500** drei Adern enthält: ADER A, ADER B und ADER C. Die Polygondatenstruktur **600** spezifiziert auch, dass das Polygon **500** eine Diffusionsschichtgeometrie repräsentiert und eine Polygonzahl von 1 hat.

[0057] Eine Aderdatenstruktur **600** wird für ADER A erzeugt. Ähnlich werden Aderdatenstrukturen **620**, **630** für ADER B bzw. ADER C erzeugt. Eine Adersegmentdatenstruktur **640** wird für das ADERSEGMENT A erzeugt. Eine Adersegmentdatenstruktur **650** wird für das ADERSEGMENT B erzeugt. Schließlich wird eine Aderdatenstruktur **660** für das ADERSEGMENT C erzeugt. Ähnlich werden Zweigdatenstrukturen **660**, **670** und **680** für jedes Adersegment erzeugt. Zusätzlich werden zwei Seitenzweigdatenstrukturen **690** und **695** für das ADERSEGMENT A erzeugt, da ADER B und ADER C mit ADER A entlang einem gerichteten Rand a_1 von ADER A verbunden sind.

[0058] [Fig. 7](#) zeigt ein Polygon **700**, das aus zwei Adern **702** und **704** aufgebaut ist, gemäß einer Ausführung der Erfindung. Die Adern **702** und **704** sind aus einem Satz von Adersegmenten aufgebaut, die jeweils aus einem Paar gerichteter Liniensegmente aufgebaut sind. Die Richtung der gerichteten Liniensegmente ist durch die Pfeile **706** angegeben.

[0059] Das Polygon **700** enthält mehrere unterschiedliche Typen von IC-Layoutgeometrien, die mittels Adersegmenten aufgebaut sind, die, gemäß einer Ausführung der Erfindung, entweder eine Polysilicium- oder Metallschichtgeometrie repräsentieren. Eine Kontakteinfassung **708** ist mit der Ader **704** über einen geknickten Pfad **710** verbunden. Die Ader **704** enthält eine achteckige Kontakteinfassung **712**, die aus drei Adersegmenten **714**, **715** und **716** aufgebaut ist. Gemäß einer Ausführung der Erfindung, und wie durch die Adersegmente **714** und **716** dargestellt, brauchen die zur Bildung eines Adersegments verwendeten gerichteten Liniensegmente nicht parallel sein oder Endpunkte aufweisen, die mit benachbarten Segmenten zusammenfallen. Dieser Aspekt der Erfindung erlaubt eine weite Vielzahl von IC-Layoutgeometrien, die mittels Adersegmenten konstruiert werden können.

[0060] [Fig. 8](#) zeigt eine IC-Layoutdiffusionsinsel **800**, die aus Adersegmenten aufgebaut ist, gemäß einer Ausführung der Erfindung. Die Diffusionsinsel **800** enthält Diffusionsflächen **802**, die eine Diffusionsgeometrie darstellen. Die Diffusionsinsel **800** enthält auch Polysiliciumflächen **804**, die eine Polysiliciumgeometrie darstellen. Die Diffusionsinsel **800** enthält auch eine Serie von Kontakten **806**. Schließlich enthält die Diffusionsinsel **800** eine Verbindungswanne **808**, die einen Kontakt **810** enthält.

[0061] Die Diffusionsinsel **800** wird durch drei Polygone dargestellt. Trotz der unregelmäßigen Form der Diffusionsinsel **800** enthält ein Polygon alle Diffusionsflächen **802** und ist aufgebaut aus einem Adersegment **812**, einem Adersegment **814** und einem Adersegment **816**. Ein anderes Polygon enthält alle Polysiliciumflächen **804** und ist aufgebaut aus vier

Adern, einer für jedes Polysiliciumgatter. Ein anderes Polygon enthält alle Kontakte **806** sowie den in der Verbindungswanne **808** enthaltenen Kontakt **810**. Die Kontakte **806** können, aufgrund ihrer gemeinsamen elektrischen Charakteristika, in einem einzigen Polygon definiert werden, oder können alternativ als separate Polygone definiert werden. Wenn z. B. ein Polysiliciumgatter an einer Seite mehr als einen Kontakt aufweist, können die Kontakte an dieser Seite als ein Polygon dargestellt werden, mit einer Ader pro Kontaktbereich (Rechteck/Achteck etc.).

3) Verwendung von Datenstrukturen während der IC-Layoutsynthese

[0062] Gemäß einer Ausführung der Erfindung werden die oben beschriebenen Strukturen während der IC-Layoutsynthese verwendet, um zu verifizieren, dass bestimmte Charakteristika eines Layouts, wie etwa Polygonbreiten, Abstände und Kontakteinfassung, vorbestimmte Konstruktionsregeln erfüllen. Das Konzept der kanonischen Richtung, wie hierin beschrieben, ist ein wichtiger Aspekt dieses Typs der Verifizierung, weil sie dazu beiträgt, die Anzahl der im Layout erforderlichen vollständigen Prüfungen zu minimieren. Im Kontext der IC-Layoutverifizierung bezieht sich eine Vollprüfung auf den Prozess der Prüfung aller Seiten eines Polygons in Bezug auf ein in dem IC-Layout enthaltenes Objekt, differenziert z. B. aus der Prüfung nur der dem anderen Objekt nächsten Seite des Polygons.

a. Bestimmung der kanonischen Richtung von einem Liniensegment zu einem Punkt

[0063] Gemäß einer Ausführung der Erfindung werden bis zu drei Halbebenenprüfungen verwendet, um die kanonische Richtung von einem gerichteten Liniensegment zu einem beliebigen Punkt zu bestimmen. Die erste Halbebenenprüfung bestimmt, ob der beliebige Punkt links, rechts oder an der Oberseite des gerichteten Liniensegments ist. Die zweite Halbebenenprüfung bestimmt, ob der beliebige Punkt über oder unter dem ersten Endpunkt des gerichteten Liniensegments ist. Die dritte Halbebenenprüfung bestimmt, ob der beliebige Punkt über oder unter dem zweiten Endpunkt des gerichteten Liniensegments ist. Durch Auswerten der Ergebnisse der drei Halbebenenprüfungen wird eine etwaige Mehrdeutigkeit aufgelöst, die dem Ort des beliebigen Punkts in Bezug auf das gerichtete Liniensegment zuordenbar ist.

[0064] Jede Halbebenenprüfung beinhaltet die Berechnung des Punktprodukts des beliebigen Punkts in Bezug auf das gerichtete Liniensegment. Ein Liniensegment, das die Grenze der Halbebene darstellt, wird um 90° im Gegenuhrzeigersinn gedreht, und ein beliebiger Punkt wird auf das neue Liniensegment projiziert. Das Vorzeichen des Punktprodukts be-

stimmt dann, ob sich der beliebige Punkt links der Halbebenengrenze (+1), auf der Grenze (0) oder rechts der Grenze (-1) befindet. Die drei Halbebenengrenzen sind jeweils das ursprüngliche Liniensegment, ein Liniensegment, das durch den ersten Endpunkt hindurchgeht und um 90° gedreht ist, sowie ein Liniensegment, das durch den zweiten Endpunkt hindurchgeht und um 90° gedreht ist.

[0065] Z. B. zeigt [Fig. 9A](#) ein gerichtetes Liniensegment **900**, das durch einen ersten Endpunkt **902** und einen zweiten Endpunkt **904** definiert ist. Die Flächen um das gerichtete Liniensegment **900** herum entsprechen dem Orientierungsschlüssel **100** von [Fig. 1](#). Insbesondere definiert die Fläche **906** eine Fläche, die in Bezug auf das gerichtete Liniensegment **900** oben ist (oben). Die Fläche **908** definiert eine Fläche, die über und rechts (oben, rechts) des gerichteten Liniensegments **900** ist. Die Fläche **910** definiert eine Fläche, die rechts (rechts) in Bezug auf das gerichtete Liniensegment **900** ist. Die Fläche **912** definiert eine Fläche, die unten und rechts (unten, rechts) des gerichteten Liniensegments **900** ist. Die Fläche **914** definiert eine Fläche, die unten (unten) in Bezug auf das gerichtete Liniensegment **900** ist. Die Fläche **916** definiert eine Fläche, die unten und links (unten, links) des gerichteten Liniensegments **900** ist. Die Fläche **918** definiert eine Fläche, die sich links des gerichteten Liniensegments **900** befindet. Schließlich definiert die Fläche **920** eine Fläche, die oben und links (oben, links) des gerichteten Liniensegments **900** ist.

[0066] Um die kanonische Richtung zwischen einem beliebigen Punkt **925** und dem gerichteten Liniensegment **900** zu bestimmen, werden bis zu drei Halbebenenprüfungen verwendet. Die erste Halbebenenprüfung bestimmt, ob sich der Punkt **925** links, rechts oder an der Oberseite des gerichteten Liniensegments **900** befindet. Wie in [Fig. 9A](#) gezeigt, liegt der Punkt **925** links des gerichteten Liniensegments **900**. Dies bedeutet, dass bei Abschluss der ersten Halbebenenprüfung der Punkt **925** in den Flächen **916**, **918** oder **920** liegen könnte.

[0067] Die zweite Halbebenenprüfung bestimmt, ob der Punkt **925** über oder unter dem ersten Endpunkt **902** des gerichteten Liniensegments **900** liegt. Das Ergebnis der zweiten Halbebenenprüfung zeigt an, dass sich der Punkt **925** über dem ersten Punkt **902** befindet, was bedeutet, dass der Punkt **925** nicht in der Fläche **916** liegen kann und daher in entweder der Fläche **918** oder der Fläche **920** liegen muss. Die dritte Halbebenenprüfung bestimmt, ob sich der Punkt **925** über oder unter dem zweiten Endpunkt **904** befindet. Wie dargestellt, zeigt die zweite Halbebenenprüfung an, dass der Punkt **925** unter dem zweiten Endpunkt **904** liegt, was bedeutet, dass der Punkt **925** in der Fläche **918** liegen muss oder links des gerichteten Liniensegments **900**.

b. Bestimmung der kanonischen Richtung aus einem Adersegment zu einem Punkt

[0068] Zur Durchführung von Abstandsprüfungen während der IC-Layoutverifizierung müssen Abstände zwischen Polygonen bestimmt und dann mit der minimalen Trennung verglichen werden, die durch die geltenden Designregeln gefordert werden. Typischerweise wird eine Abstandsprüfung zwischen einem Polygon und einem gewählten Punkt an einem anderen Polygon durchgeführt.

[0069] Gemäß einer Ausführung der Erfindung wird ein Verfahren angegeben, um die kanonische Richtung von einem Adersegment zu einem Punkt zu bestimmen, sodass höchstens zwei Seiten eines Polygons gegenüber dem Punkt geprüft werden müssen. Bis zu vier Prüfungen der kanonischen Richtung von Segmentseiten zu dem beliebigen Punkt werden durchgeführt, um die kanonische Richtung von einem Adersegment zu einem Punkt zu bestimmen und eine etwaige Mehrdeutigkeit aufzulösen, die dem zuzuordnen ist, ob sich der Punkt innerhalb oder außerhalb des Adersegments befindet.

[0070] [Fig. 9B](#) zeigt ein Adersegment **952**, das durch eine linke Seite **956**, eine rechte Seite **958**, eine erste Endseite **960** und eine zweite Endseite **962** definiert ist. Die Flächen um das Adersegment **952** herum entsprechen dem Orientierungsschlüssel **100** von [Fig. 1](#). Insbesondere definiert die Fläche **964** eine Fläche, die in Bezug auf das Adersegment **952** oben liegt (oben). Die Fläche **966** definiert eine Fläche, die in Bezug auf das Adersegment **952** oben und rechts (oben, rechts) liegt. Die Fläche **968** definiert eine Fläche, die in Bezug auf das Adersegment **902** rechts (rechts) liegt. Die Fläche **920** definiert eine Fläche, die in Bezug auf das Adersegment **952** unten und rechts (unten, rechts) liegt. Die Fläche **972** definiert eine Fläche, die in Bezug auf das Adersegment **952** unten (unten) liegt. Die Fläche **974** definiert eine Fläche, die in Bezug auf das Adersegment **952** unten und links (unten, links) liegt. Schließlich definiert die Fläche **976** eine Fläche, die in Bezug auf das Adersegment **952** links (links) liegt. Die Fläche **978** definiert eine Fläche, die in Bezug auf das Adersegment **952** oben und links (oben, links) liegt.

[0071] Man nehme z. B. an, dass die kanonische Richtung von dem Adersegment **952** zu einem Punkt **980** bestimmt werden soll. Eine erste kanonische Richtungsprüfung des Punkts **980** in Bezug auf die linke Seite **956** wird durchgeführt, um zu bestimmen, ob der Punkt **980** links, rechts oder an der Oberseite der linken Seite **956** liegt. In dieser Situation zeigt die erste kanonische Richtungsprüfung an, dass der Punkt **980** rechts der linken Seite **956** liegt. Somit muss auf der Basis der Ergebnisse der ersten kanonischen Richtungsprüfung der Punkt **980** in der Fläche **964**, **966**, **968**, **970**, **972** oder innerhalb des Ader-

segments **952** angeordnet sein.

[0072] Eine zweite kanonische Richtungsprüfung des Punkt **980** in Bezug auf die rechte Seite **958** wird durchgeführt, um zu bestimmen, ob der Punkt **980** links, rechts oder an der Oberseite der rechten Seite **958** liegt. In dieser Situation zeigt die zweite kanonische Richtungsprüfung an, dass der Punkt **980** links der rechten Seite **958** liegt. Somit muss auf der Basis der Ergebnisse beider durchgeführten ersten und zweiten kanonischen Richtungsprüfungen der Punkt **980** in den Flächen **964**, **972** oder innerhalb des Adersegments **952** liegen.

[0073] Eine dritte kanonische Richtungsprüfung von Punkt **980** in Bezug auf die erste Fläche **960** wird durchgeführt, um zu bestimmen, ob der Punkt **980** links, rechts oder an der Oberseite der ersten Endseite **960** liegt. In dieser Situation zeigt die dritte kanonische Richtungsprüfung an, dass der Punkt **980** links der ersten Endseite **960** liegt. Daher muss, auf der Basis der Ergebnisse der drei kanonischen Richtungsprüfungen, der Punkt **980** in der Fläche **972** liegen, oder (unten) mit Bezug auf das Adersegment **952**.

[0074] Für das gerade beschriebene bestimmte Szenario waren nur drei kanonische Richtungsprüfungen erforderlich, um die kanonische Richtung von einem Adersegment zu einem Punkt unzweideutig zu bestimmen. Jedoch ist in einigen Situationen eine vierte kanonische Richtungsprüfung erforderlich, um die kanonische Richtung zwischen einem Adersegment und einem Punkt definitiv zu bestimmen. Wo z. B. die dritte kanonische Richtungsprüfung von Punkt **980** in Bezug auf die erste Endseite **960** anzeigt, dass der Punkt **980** rechts der ersten Endseite **960** liegt, dann ist eine vierte kanonische Richtungsprüfung des Punkts **980** in Bezug auf die zweite Endseite **962** erforderlich, um zu bestimmen, ob der Punkt **970** in der Fläche **964** angeordnet ist, oder innerhalb oder oberhalb des Adersegment **952** angeordnet ist.

[0075] Die kanonische Richtung von den Adersegmentseiten zu einem beliebigen Punkt **980** wird verwendet, um sicherzustellen, dass die Flächen **964**, **966**, **698**, **970**, **972**, **974**, **976** und **978** nicht enger werden, wenn der Abstand von dem beliebigen Punkt **980** zu dem Adersegment **852** zunimmt. Dies kann immer dann passieren, wenn der Winkel zwischen zwei benachbarten Seiten größer als 90° ist.

c. Abstandsprüfungen

[0076] Gemäß einer Ausführung der Erfindung wird ein Ansatz zur Durchführung von Abstandsprüfungen zwischen Polygonen, die mittels Adersegmenten und kanonischer Richtung konstruiert sind, vorgesehen. [Fig. 10](#) zeigt ein IC-Layout **1000**, das eine Diffusionsinsel **1002** und eine Diffusionsinsel **1004** enthält. Die

Diffusionsinsel **1002** enthält ein Polygon **1005**, das eine Diffusionsgeometrie darstellt und aus einem einzigen Adersegment aufgebaut ist. Ähnlich enthält die Diffusionsinsel **1004** ein Polygon **1009**, das eine Diffusionsgeometrie darstellt und aus einem einzigen Adersegment aufgebaut ist.

[**0077**] Gemäß einer Ausführung der Erfindung wird die Durchführung einer Abstandsprüfung zwischen parallelen überlappenden Rändern von Adersegmenten erreicht durch einen Vergleich des senkrechten Abstands zwischen den Adern mit dem minimalen Platz, der durch die geltenden Konstruktionsregeln gefordert wird. Z. B. beinhaltet die Durchführung einer Abstandsprüfung zwischen dem Polygon **1005** und dem Polygon **1009** die Berechnung eines orthogonalen Abstands zwischen den Punkten a_1 und a_2 auf dem gerichteten Liniensegment A und den Punkten b_1 und b_2 auf dem Liniensegment B, und dann Vergleichen des berechneten Abstands mit dem durch die Konstruktionsregeln spezifizierten minimalen Abstand. Wenn der berechnete Abstand dem durch die Konstruktionsregeln spezifizierten minimalen Abstand nicht genügt, dann muss eines der Polygone **1005** oder **1009** bewegt werden. Wenn z. B. der Abstand zwischen dem Polygon **1005** und dem Polygon **1009** dem durch die Konstruktionsregeln spezifizierten minimalen Abstand nicht genügt, dann könnte das Polygon **1005** in der Abwärtsrichtung bewegt werden, um den Abstand zwischen dem Polygon **1005** und dem Polygon **1009** zu vergrößern. Alternativ könnte das Polygon **1009** in der Aufwärtsrichtung bewegt werden.

[**0078**] Zur Berechnung des Abstands zwischen nicht überlappenden Adern werden die nächsten Ecken zwischen den zwei Adern verwendet, und anstatt eines orthogonalen Abstands wird ein nicht orthogonaler oder Manhattanabstand verwendet. Der Begriff Manhattanabstand bezieht sich auf einen Abstand, wo jede Einheit in der X-Richtung einer Einheit darüber und in der Y-Richtung eine Einheit darüber liegt. Der resultierende Richtungsvektor ist dann einer von (oben, links), (oben, rechts), (unten, rechts) oder (unten, links). Wenn der Abstand dem durch die Konstruktionsregeln spezifizierten minimalen Abstand nicht genügt, dann wird eines der Polygone in einer Richtung bewegt, die den zwei gegenüberliegenden Seiten einer der Ecken gegenüberliegt.

[**0079**] Man nehme z. B. in **Fig. 10A** an, dass eine Diffusion zu Polysiliciumabstandsprüfung zwischen dem Punkt b_1 auf dem Polygon **1009** und dem Punkt c_1 auf dem Polygon **1010** durchgeführt wird. Wenn der Abstand zwischen den Punkten b_1 und c_1 dem durch die Konstruktionsregeln spezifizierten minimalen Abstand nicht genügt, dann wird das Polygon **1010** in Richtung (unten, links) bewegt, da eine der nächsten Seiten **1014** des Polygons **1010** nach oben gerichtet ist, und die andere der nächsten Seiten

1016 nach rechts gerichtet ist. Auf ähnliche Weise könnte das Polygon **1009** in der Richtung (oben, rechts) bewegt werden.

[**0080**] Sobald eine Abstandsverletzung identifiziert worden ist und die Richtung, um die eines der Polygone zur Erfüllung der Konstruktionsregeln bewegt werden muss, bestimmt worden ist, muss der Abstand, um den das Polygon zur Erfüllung der minimalen Abstandserfordernisse bewegt werden muss, bestimmt werden. Gemäß einer Ausführung der Erfindung wird ein neuartiger Ansatz vorgesehen, um den Abstand zur Umordnung der Layoutgeometrie zu bestimmen, um ein Abstandserfordernis zu erfüllen. Allgemein beinhaltet der Ansatz die Konstruktion einer "Ausschlusszone" um die Layoutgeometrie herum, um die minimale Trennung zu definieren, die zwischen dieser Layoutgeometrie und einer anderen Layoutgeometrie eingehalten werden, um den durch die Konstruktionsregeln geforderten minimalen Abstand zu erfüllen. Dieser Ansatz ist nicht von irgendeiner bestimmten Datenstrukturarchitektur abhängig. Wenn ein Punkt, der ein Teil der anderen Geometrie ist, innerhalb der Ausschlusszone liegt, dann liegt eine Abstandsverletzung vor.

[**0081**] Wie in **Fig. 11** gezeigt, ist ein Adersegment **1100** zumindest teilweise aus zwei gerichteten Liniensegmenten **1102** und **1104** aufgebaut, die sich an einem Punkt **1105** treffen. Zur Konstruktion der Ausschlusszone werden zwei Linien **1106** und **1108** entlang der Richtungsvektoren der Liniensegmente **1102** und **1104** an einer Trennung von den gerichteten Liniensegmenten **1102**, **1104** gezogen, um die Konstruktionsregeln zu erfüllen. Die Linien **1106** und **1108** werden gebildet durch Auswärtsbewegung von einem Punkt **1105** in Richtung senkrecht zur Richtung der gerichteten Liniensegmente **1102** und **1104**. Wenn die gerichteten Liniensegmente **1102** und **1104** zueinander senkrecht, dann wird eine dritte Linie **1110** gezogen, die die Linien **1106** und **1108** verbindet. Die Linien **1106**, **1108** und **1110** bilden eine Ausschlusszone **1112** um das Adersegment **1100** herum. Wenn die gerichteten Liniensegmente **1102** und **1104** orthogonal sind, beruht die Platzierung der Linien **1106** und **1108** in Bezug auf die gerichteten Liniensegmente **1102** und **1104** auf dem Kartesischen Breiten-Abstands- oder Umfangswert. Die Platzierung der dritten Linie **1110** beruht auf der Manhattanbreite, dem Abstand oder den Umfängen, wie sie durch die Konstruktionsregeln definiert sind. Da die Manhattanbreiten, Abstands- oder Umfangsregeln relativ zur Kartesischen Regel größer werden, bewegt sich die Linie **1110** von dem Adersegment **1100** auswärts, während die Linien **1106** und **1108** am Ort verbleiben. Innerhalb der Grenze kann die Linie **1110** bis zu den Kreuzungen der Linien **1106** und **1108** nach außen bewegt werden.

[**0082**] Wenn andererseits die gerichteten Linien-

segmente **1102** und **1104** nicht orthogonal sind, dann beruht die Platzierung der Linien **1106** und **1108** in Bezug auf die gerichteten Liniensegmente **1102** und **1104** auf dem Kartesischen Breiten-, Abstands- oder Umfangswert, geteilt durch die Quadratwurzel von zwei und aufgerundet zum nächsten ganzzahligen Wert, um den Manhattanabstand bereitzustellen, der zum Erhalten des erforderlichen Kartesischen Abstands erforderlich ist.

[0083] Der Ansatz zum Umordnen eines Adersegments, um einem Abstandserfordernis zu erfüllen, gemäß einer Ausführung der Erfindung, wird nun in Bezug auf das Flussdiagramm von [Fig. 12](#) beschrieben. Nach dem Start in Schritt **1200** werden in Schritt **S1202** zwei Linien **1106** und **1108** entlang den Richtungsvektoren der zwei Liniensegmente konstruiert, die das Adersegment aufbauen. In Schritt **1204** wird bestimmt, ob die gerichteten Liniensegmente **1102**, **1104** senkrecht sind. Falls ja, dann wird in Schritt **1206** bestimmt, ob die gerichteten Liniensegmente **1102**, **1104** orthogonal sind. Wenn die gerichteten Liniensegmente **1102**, **1104** beide senkrecht und orthogonal sind, dann wird in Schritt **1208** der Manhattanabstand benutzt, um eine dritte Linie **1110** in Bezug auf die zwei konstruierten Linien zu positionieren.

[0084] Wenn andererseits in Schritt **1204** bestimmt wird, dass die gerichteten Liniensegmente **1102**, **1104** nicht senkrecht sind, dann umfasst die Ausschlusszone die gerichteten Liniensegmente **1106** und **1108**, und der Prozess wird in Schritt **1214** abgeschlossen. Wenn jedoch in Schritt **1206** bestimmt wird, dass die gerichteten Liniensegmente **1102**, **1104** nicht orthogonal sind, dann basiert der Ort der dritten Linie **1110** in Bezug auf die ersten zwei konstruierten Linien auf dem minimalen Kartesischen Abstand (Produkt der Quadratwurzel von zwei und des Manhattanabstands, aufgerundet), der durch die Konstruktionsregeln spezifiziert ist.

[0085] Dann wird in Schritt **1212** die dritte Linie **1110** konstruiert und zwischen die gerichteten Liniensegmente **1102** und **1104** gelegt. Der Prozess ist in Schritt **1214** abgeschlossen. Dieser Ansatz kann auch dazu benutzt werden, die Polygonbreiten und Kontaktumfassungen zu prüfen und zu korrigieren.

d. Erzeugen der Polygonausgabe

[0086] Gemäß einer Ausführung der Erfindung wird hierin die Fähigkeit angegeben, eine Polygonausgabe auf der Basis der hierin beschriebenen Datenstrukturhierarchie zu generieren. Dieser Ansatz erzeugt weniger Polygone als der alternative Ansatz zum Erzeugen eines Polygons für jedes Adersegment. Das Erzeugen weniger Polygone reduziert die Zeit, die zum Durchlauf der Analyseprogramme erforderlich ist, wie etwa Konstruktionsregelprüfungen, Schaltungsextraktion und Maskenherstellung. Im

Kontext von computerunterstützten Designpaketen können weniger Polygone auch den Überblick über das Layout leichter machen, weil weniger Polygone auf dem Schirm sind. Allgemein beinhaltet der Ansatz das Auswerten von Adersegmentdatenstrukturen für jede Ader, um kollineare Punktsätze zu identifizieren, die dann kombiniert werden, um die Anzahl der Polygone zu reduzieren. Gemäß einer Ausführung der Erfindung werden linke Seiten durchgehender Adersegmente ausgewertet und werden dann die rechten Seiten derselben durchgehenden Adersegmente ausgewertet.

[0087] Z. B. zeigt [Fig. 13A](#) eine Ader **1300**, die aus drei durchgehenden Adersegmenten **1302**, **1304** und **1306** aufgebaut ist, gemäß einer Ausführung der Erfindung. Die linke Seite des Adersegments **1302** ist durch ein gerichtetes Liniensegment **a** definiert, das wiederum durch Endpunkte a_1 und a_2 definiert ist. Die linke Seite des Adersegments **1304** ist durch ein gerichtetes Liniensegment **b** definiert, das wiederum durch Endpunkte b_1 und b_2 definiert ist. Die linke Seite des Adersegments **1306** ist durch ein gerichtetes Liniensegment **c** definiert, das wiederum durch Endpunkte c_1 und c_2 definiert ist. Zum Erzeugen einer Polygondarstellung der Ader **1300** gemäß einer Ausführung der Erfindung werden kollineare Punktsätze entlang linken und rechten Seiten von Adersegmenten **1302**, **1304** und **1306** kombiniert. Da das gerichtete Liniensegment **a** kollinear zu dem gerichteten Liniensegment **b** ist, können die Punkte a_2 und b_1 eliminiert werden, was effektiv eine einzige linke Seite entlang den Adersegmenten **1302** und **1304** erzeugt. Da die Punkte b_2 und c_1 zusammenfallen, können sie ähnlich durch einen einzigen Punkt ersetzt werden. [Fig. 13B](#) zeigt eine Polygondarstellung **1308** der Ader **1300** von [Fig. 13A](#), nachdem kollineare Punktsätze kombiniert worden sind, gemäß einer Ausführung der Erfindung.

[0088] Einige der Polygonmanipulationsprozesse definieren eine maximale Anzahl von Polygonpunkten pro Polygon. In dieser Situation ist die Anzahl der Segmente, die gemäß diesem Ansatz verfolgt werden können, auf die maximale Anzahl zulässiger Polygone, geteilt durch vier, begrenzt. Wenn die Anzahl der Adersegmente diese Zahl für eine gegebene Ader überschreitet, dann werden zusätzliche Polygone benutzt, um diese Ader darzustellen.

HARDWARE-ÜBERBLICK

[0089] [Fig. 14](#) ist ein Blockdiagramm, das ein Computersystem **1400** zeigt, auf dem eine Ausführung der Erfindung implementiert werden kann. Das Computersystem **1400** enthält einen Bus **1402** oder einen anderen Kommunikationsmechanismus zum Kommunizieren von Information sowie einen mit dem Bus **1402** gekoppelten Prozessor **1404** zur Informationsverarbeitung. Das Computersystem **1400** enthält

auch einen Hauptspeicher **1406**, wie etwa einen Direktzugriffsspeicher (RAM) oder eine andere dynamische Speichervorrichtung, die mit dem Bus **1402** gekoppelt ist, zur Speicherung von Information und Anweisung zur Ausführung durch den Prozessor **1404**. Der Hauptspeicher **1406** kann auch zum Speichern temporärer Variablen oder anderer intermediärer Information während der Ausführung von Anweisungen zur Ausführung durch den Prozessor **1404** verwendet werden. Das Computersystem **1400** enthält ferner einen Festwertspeicher (ROM) **1408** oder eine andere statische Speichervorrichtung, die mit dem Bus **1402** gekoppelt ist, zum Speichern von statischer Information und Anweisungen für den Prozessor **1404**. Eine Speichervorrichtung **1410**, wie etwa eine Magnetplatte oder eine Optikplatte, ist vorgesehen und mit dem Bus **1402** gekoppelt, zum Speichern von Information und Anweisungen.

[0090] Das Computersystem **1400** kann über den Bus **1402** mit einer Anzeige **1412**, wie etwa einer Kathodenstrahlröhre (CRT) gekoppelt sein, um einem Computeranwender Information anzuzeigen. Eine Eingabevorrichtung **1414**, einschließlich alphanumerischen und anderen Tasten, ist mit dem Bus **1402** gekoppelt, zur Kommunikation von Informations- und Befehlsauswahlen an den Prozessor **1404**. Ein anderer Typ der Verwendereingabevorrichtung ist eine Cursorsteuerung **1416**, wie etwa eine Maus, ein Trackball oder Cursorrichtungstasten zur Kommunikation von Richtungsinformation und Befehlsauswahlen an den Prozessor **1404** und zum Steuern der Cursorbewegung auf der Anzeige **1412**. Diese Eingabevorrichtung hat typischerweise zwei Freiheitsgrade in zwei Achsen, einer ersten Achse (z. B. X) und einer zweiten Achse (z. B. Y), die es erlauben, dass die Vorrichtung Positionen in einer Ebene spezifiziert.

[0091] Die Erfindung bezieht sich auf die Verwendung des Computersystems **1400** zur Darstellung von Polygonen in einem IC-Layout. Gemäß einer Ausführung der Erfindung wird die Darstellung von Polygonen in einem IC-Layout durch das Computersystem **1400** in Antwort darauf vorgesehen, dass der Prozessor **1404** in dem Hauptspeicher **1406** enthaltene Anweisungssequenzen ausführt. Diese Anweisungen können in den Hauptspeicher **1406** aus einem anderen computerlesbaren Medium eingelesen werden, wie etwa einer Speichervorrichtung **1410**. Jedoch ist das computerlesbare Medium nicht auf solche Vorrichtungen, wie etwa die Speichervorrichtung **1410**, beschränkt. Z. B. kann das computerlesbare Medium eine Floppy Disk, eine Flexi Disk, eine Festplatte, ein Magnetband oder irgendein anderes magnetisches Medium enthalten, eine CD-ROM, irgendein anderes optisches Medium, Lochkarten, Papierband oder irgendein anderes physikalisches Medium mit Lochmuster, ein RAM, ein PROM sowie ein EPROM und ein FLASH-EPROM, irgendeinen ande-

ren Speicherchip und eine Kassette, eine Trägerwelle, wie nachfolgend beschrieben, oder irgendein anderes Medium sein, aus dem ein Computer lesen kann.

[0092] Die Ausführung der in dem Hauptspeicher **1406** enthaltenen Anweisungssequenzen bewirkt, dass der Prozessor **1404** die zuvor beschriebenen Prozessschritte ausführt. In alternativen Ausführungen kann eine hartverdrahtete Schaltung anstelle von oder in Kombination mit Softwareanweisungen verwendet werden, um die Erfindung zu implementieren. Daher sind die Ausführungen der Erfindung nicht auf irgendeine bestimmte Kombination von Hardware-schaltung und Software beschränkt.

[0093] Das Computersystem **1400** enthält auch eine Kommunikationsschnittstelle **1418**, die mit dem Bus **1402** gekoppelt ist. Die Kommunikationsschnittstelle **1418** bietet eine Zweiwegedaten-Kommunikationskupplung mit einer Netzwerkverknüpfung **1420**, die mit einem lokalen Netzwerk **1422** verbunden ist. Z. B. kann die Kommunikationsschnittstelle **1418** eine integrierte digitale Dienstnetzwerk (ISDN)-Karte oder ein Modem sein, um für eine Datenkommunikationsverbindung mit einem entsprechenden Telefonleitungstyp zu sorgen. Als ein anderes Beispiel kann die Kommunikationsschnittstelle **1418** eine Lokalbereichnetzwerk (LAN)-Karte sein, um eine Datenkommunikationsverbindung mit einem kompatiblen LAN vorzusehen. Es können auch drahtlose Verknüpfungen implementiert werden. In jeder solchen Implementierung sendet und empfängt die Kommunikationsschnittstelle **1418** elektrische, elektromagnetische oder optische Signale, die digitale Datenströme tragen, welche verschiedene Informationstypen repräsentieren.

[0094] Die Netzwerkverknüpfung **1420** liefert typischerweise eine Datenkommunikation durch ein oder mehrere Netzwerke zu anderen Datenvorrichtungen. Z. B. kann die Netzwerkverknüpfung **1420** eine Verbindung durch ein lokales Netzwerk **1422** zu einem Hostecomputer **1424** oder zu einem Datengerät vorzusehen, das durch einen Internet-Serviceprovider (ISP) **1426** betrieben wird. Das ISP **1426** liefert wiederum Datenkommunikationsdienste durch ein weltweites Paketdaten-Kommunikationsnetzwerk, das nun allgemein als "Internet" **1428** bezeichnet wird. Das lokale Netzwerk **1422** und das Internet **1428** verwenden beide elektrische, elektromagnetische oder optische Signale, die digitale Datenströme tragen. Die Signale durch die verschiedenen Netzwerke und die Signale auf der Netzwerkverknüpfung **1420** und durch die Kommunikationsschnittstelle **1418**, welche die digitalen Daten zu und von dem Computersystem **1400** tragen, sind beispielhafte Formen von Trägerwellen, welche die Information transportieren.

[0095] Das Computersystem **1400** kann Meldungen

senden und Daten empfangen, einschließlich einem Programmcode, und zwar durch das oder die Netzwerke, die Netzwerkverknüpfung **1420** und die Kommunikationsschnittstelle **1418**. Im Beispiel des Internets könnte ein Server **1430** einen angeforderten Code für ein Anwenderprogramm durch das Internet **1428**, das ISP **1426**, das lokale Netzwerk **1422** und die Kommunikationsschnittstelle **1418** übertragen. Gemäß der Erfindung sorgt eine solche heruntergeladene Anwendung für die Darstellung von Polygonen in einem IC-Layout, wie hierin beschrieben.

[0096] Der empfangene Code kann durch den Prozessor **1414** ausgeführt, wenn er empfangen wird, und/oder in der Speichervorrichtung **1410** oder einer anderen nicht-flüchtigen Speicherung zur späteren Ausführung gespeichert werden. Auf diese Weise kann das Computersystem **1400** einen Anwendercode in der Form einer Trägerwelle erhalten.

[0097] Obwohl Ausführungen der Erfindung zu Illustrationszwecken im Kontext von Transistoren enthaltenden IC-Layouts beschrieben worden sind, ist die Erfindung auch auf IC-Layouts anwendbar, die irgendeinen typ integrierter Schaltungskomponenten oder -vorrichtungen enthält, wie etwa Widerstände, Kondensatoren, Logikgatter oder irgendeinen anderen Typ von IC-Vorrichtungen.

[0098] Die Erfindung bietet verschiedene Vorteile gegenüber früheren Ansätzen zur Darstellung von Polygonen in einem IC-Layout. Erstens kann auch die höchstkomplizierte Geometrie durch Adersegmente dargestellt werden, einschließlich achteckigen Kontaktumfängen und unregelmäßig geformten Diffusionsinseln. Auch erleichtert die Polygondarstellung mittels Adersegmenten die geometrische Analyse, wie etwa Abstands- und Umfangsprüfung. Die Adersegmente sind für die geometrische Analyse zweckdienlich, weil die meisten freiliegenden Polygonränder bekannt sind und die Breiten, Abstands- und Umfangsprüfungen nur dann gültig sind, wenn sie an freiliegenden Polygonrändern angewendet werden. Viele horizontale Ränder in einer Datenbank von Trapezen befinden sich tatsächlich innerhalb der Polygone, und die Analysesoftware muss signifikanten Aufwand betreiben, um diese Außenräder zu finden. Adersegmente können auch die meisten üblichen Layoutformen direkt darstellen, was eine leichte Implementierung komplizierterer Regeln erlaubt (wie etwa die Ausschlusszone). Z. B. könnte ein Polysiliziumgatter erfordern, dass seine Breite größer wird, wenn das Gatterpolygon mit einem Winkel quert. Weil das Gatterpolygon direkt aus Adersegmenten aufgebaut ist, ist es leicht, diese Regel auf segmentweiser Basis in Kraft zu setzen, durch Prüfen der Liniensegmentrichtungen und Sicherstellen, dass ihr Abstand größer ist, wenn ihre Winkel nicht orthogonal sind. Der Adersegmentansatz erleichtert auch eine relativ leichte elektrische Analyse, wie etwa Wider-

standsprüfung, weil die Verwendung von Adersegmenten erlaubt, dass der Widerstand einer Ader schnell berechnet wird. Der Adersegmentansatz ist auch zur Implementierung in Computersoftware gut geeignet, weil sich Adersegmente in Datenstrukturen leicht ausdrücken lassen. Der Vorteil von Adersegmenten ist, dass die Form allgemein verwendeter Layoutstrukturen viel näher sind, sodass viel weniger von diesen erforderlich sind, um diese Strukturen darzustellen, und es viel weniger wahrscheinlich ist, dass die Form eines Objekts nicht offensichtlich ist, nachdem nur ein oder zwei Datenstrukturen inspiziert sind. Mit Trapezen und Rechtecken kann eine kleinere Änderung zur Polygonform bewirken, dass die Trapezdatenstrukturen auf dem gesamten Weg an der anderen Seite des Polygons modifiziert werden, was es schwierig macht, mit einem Polygon als Einheit zu arbeiten.

Patentansprüche

1. Computerimplementiertes Verfahren der Bestimmung der kanonischen Richtung von einem gerichteten Liniensegment zu einem beliebigen Punkt in einer integrierten Schaltungsauslegung, wobei das Verfahren die Schritte umfasst:

- a) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen Punkt auf dem gerichteten Liniensegment;
- b) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen ersten Endpunkt des gerichteten Liniensegments; und
- c) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen zweiten Endpunkt des gerichteten Liniensegments; und
- d) auf der Basis der Ergebnisse der Schritte a) bis c), Zuweisen einer kanonischen Richtung von dem gerichteten Liniensegment zu dem beliebigen Punkt.

2. Computerlesbares Medium zum Bestimmen der kanonischen Richtung von einem gerichteten Liniensegment zu einem beliebigen Punkt in einer integrierten Schaltungsauslegung, wobei das computerlesbare Medium computerlesbare Codes aufweist, die, wenn sie durch einen oder mehrere Prozesse ausgeführt werden, sämtliche Verfahrensschritte von Anspruch 1 ausführen.

3. Computersystem zum Bestimmen der kanonischen Richtung von einem gerichteten Liniensegment zu einem beliebigen Punkt in einer integrierten Schaltungsauslegung, wobei das Computersystem umfasst:

- ein oder mehrere Prozessoren; und
- einen Speicher, der mit dem einen oder mehreren Prozessoren kommunikativ gekoppelt ist, wobei der Speicher computerlesbare Codes aufweist, die, wenn sie durch den einen oder die mehreren Prozessoren ausgeführt werden, bewirken, dass der eine oder die mehreren Prozessoren die Funktionen aus-

führen von:

- a) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen Punkt auf dem gerichteten Liniensegment;
- b) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen ersten Endpunkt des gerichteten Liniensegments; und
- c) Bestimmen des Orts des beliebigen Punkts in Bezug auf einen zweiten Endpunkt des gerichteten Liniensegments; und
- d) auf der Basis der Ergebnisse der Schritte a) bis c), Zuweisen einer kanonischen Richtung von dem gerichteten Liniensegment zu dem beliebigen Punkt.

Es folgen 16 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1

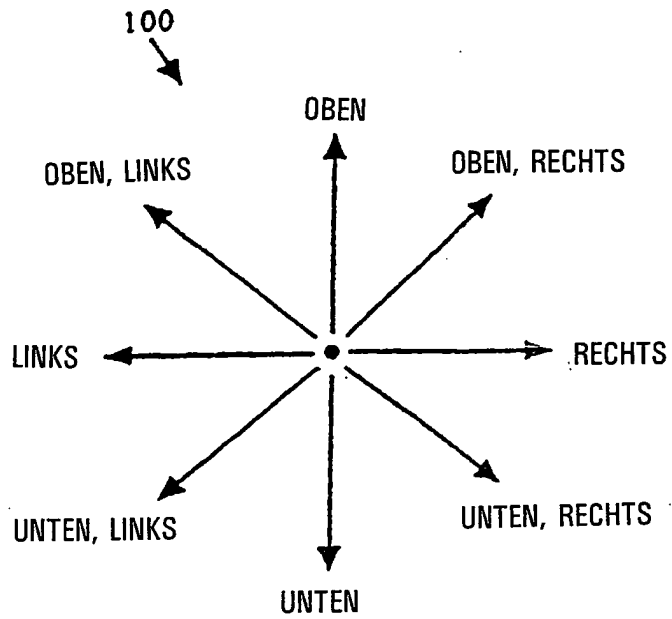


FIG. 2

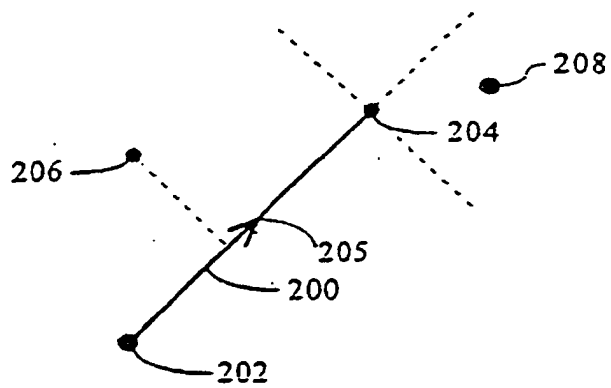


FIG. 3

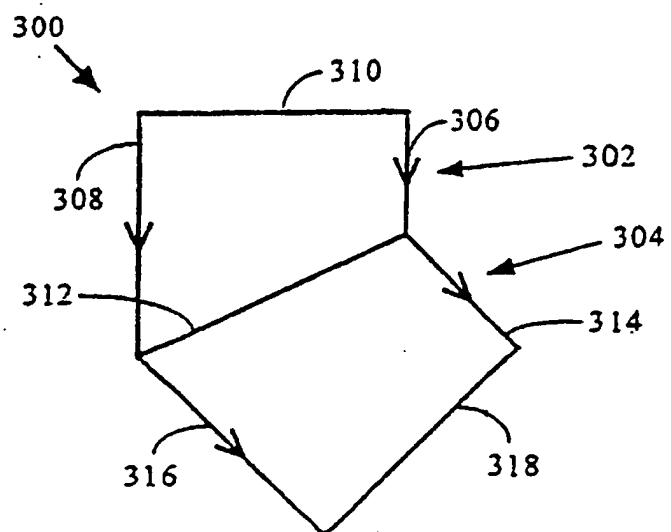


FIG. 4A

400 →

PUNKT	
X KOORDINATE	
Y KOORDINATE	

FIG. 4B

410 →

GERICHTETES LINIENSEGMENT	
ERSTER ENDPUNKT (ENDE1)	
ZWEITER ENDPUNKT (ENDE2)	
RICHTUNG	
ORTHO	
OKTANT	

FIG. 4C

420 →

ADERSEGMENT	
LINKER RAND	
RECHTER RAND	
NÄCHSTES ADERSEGMENT	
VORHERIGES ADER- SEGMENT	
STAMMADER	
ABZWEIGUNG	

FIG. 4D

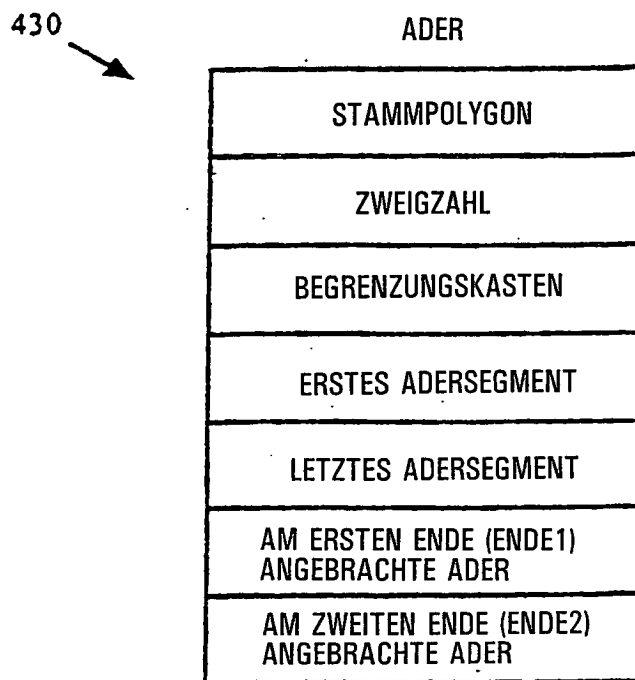


FIG. 4E

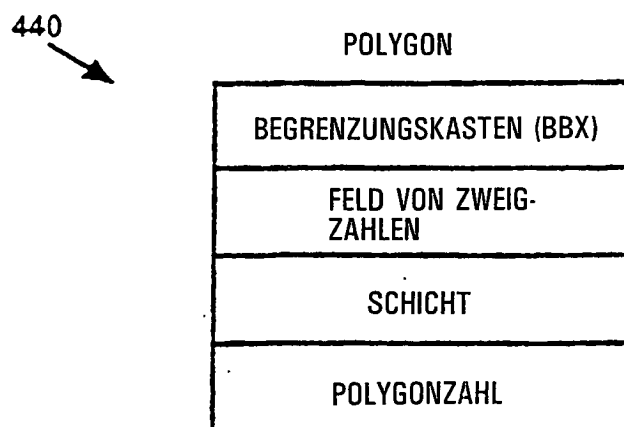


FIG. 4F

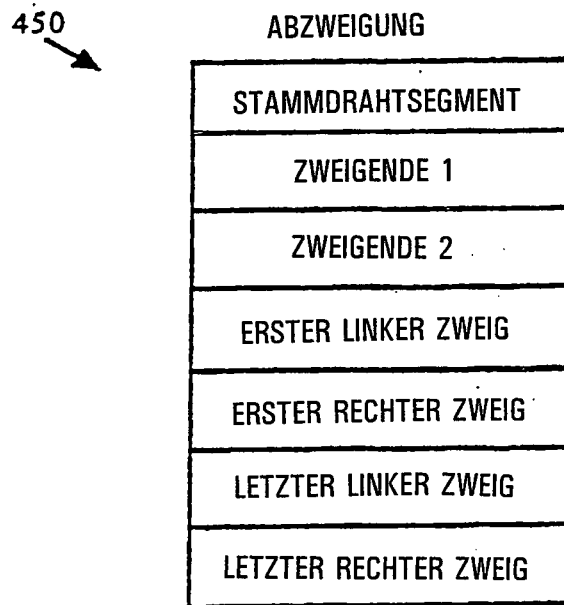


FIG. 4G

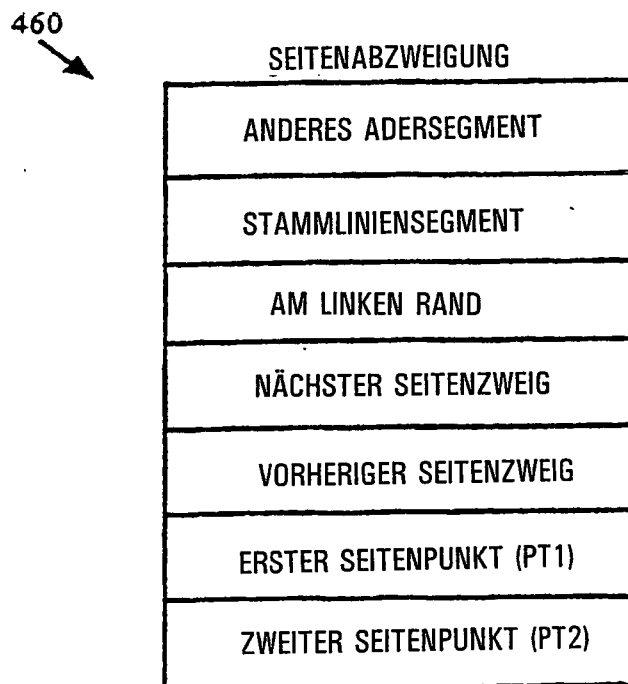


FIG. 5

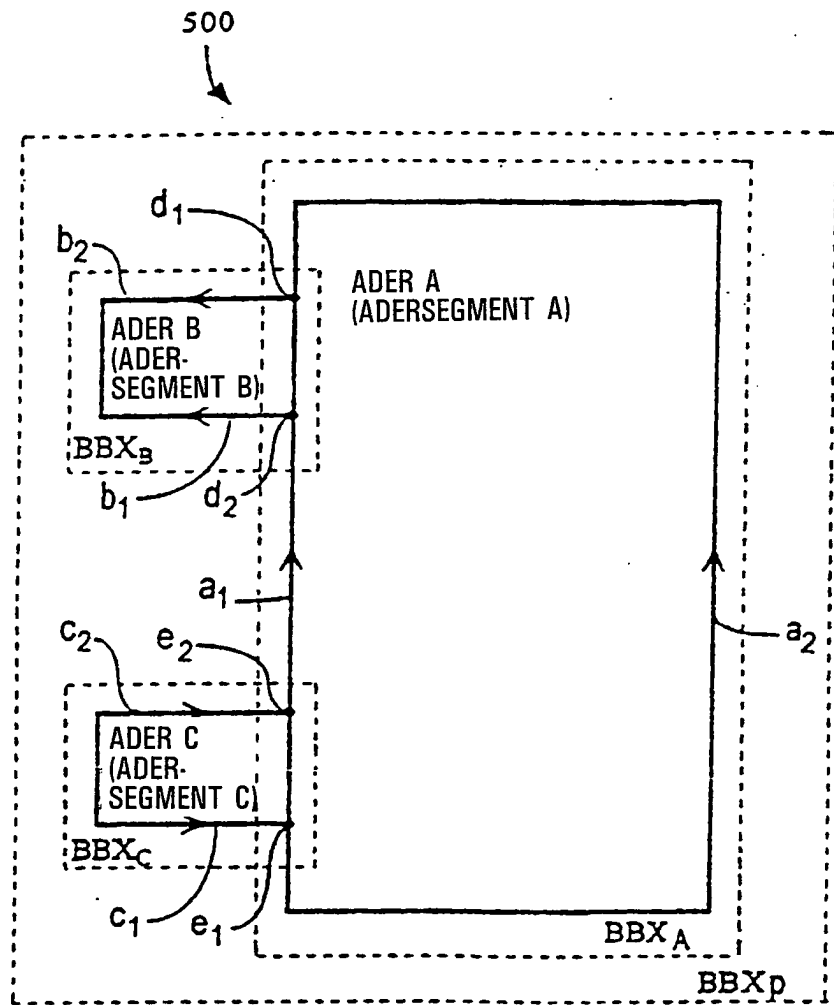


FIG. 6A

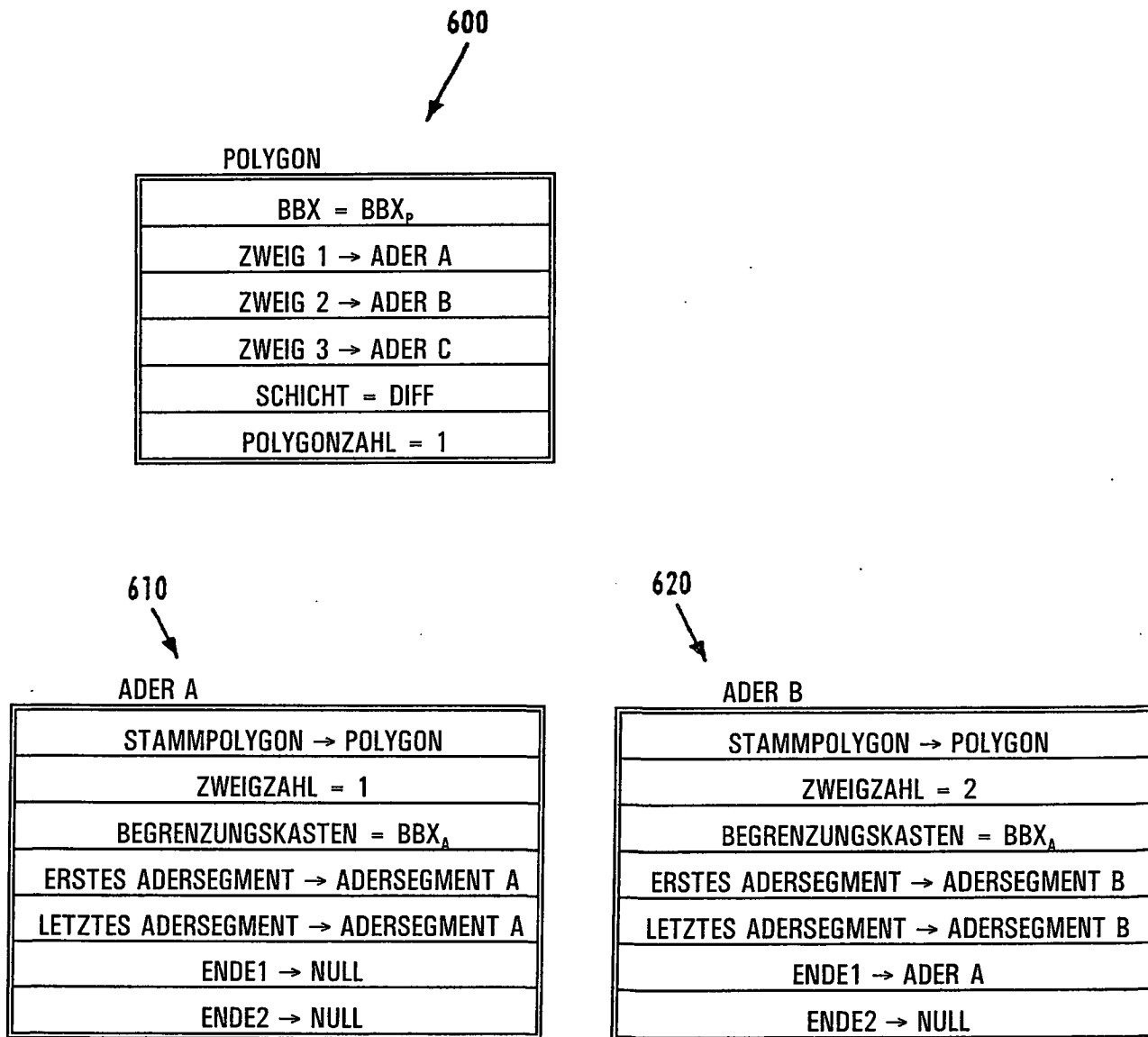


FIG. 6B

630
↓

ADER C	
STAMMPOLYGON	→ POLYGON
ZWEIGZAHL	= 3
BEGRENZUNGSKASTEN	= BBX_c
ERSTES ADERSEGMENT	→ ADERSEGMENT C
LETZTES ADERSEGMENT	→ ADERSEGMENT C
ENDE1	→ NULL
ENDE2	→ ADER A

640
↓

ADERSEGMENT A	
LINKER RAND	= a_1
RECHTER RAND	= a_2
NÄCHSTES ADERSEGMENT	→ NULL
VORHERIGES ADERSEGMENT	→ NULL
STAMMADER	→ ADER A
ABZWEIGUNG	→ ABZWEIGUNG A

FIG. 6C

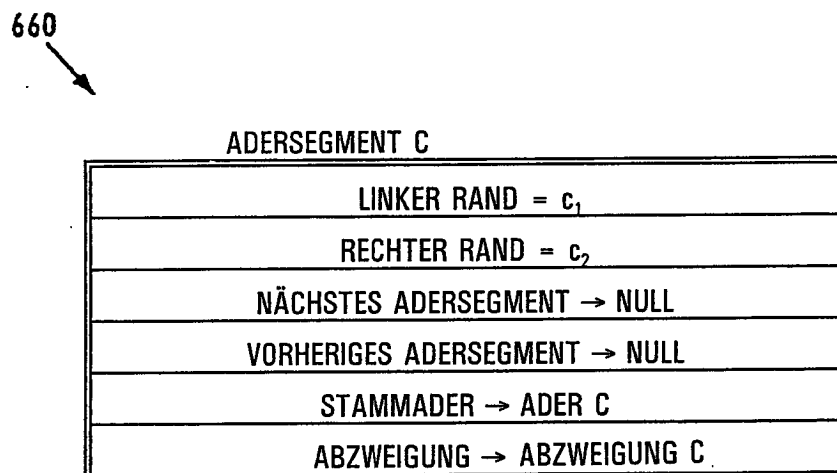
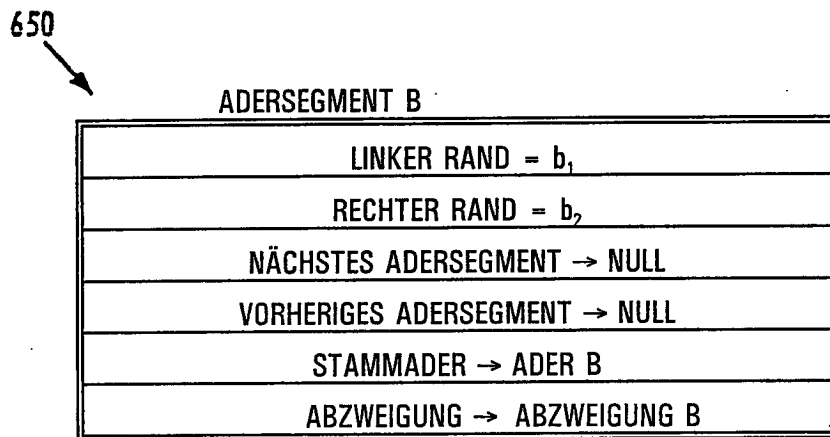


FIG. 6D

660

ABZWEIGUNG A

STAMMADERSEGMENT → ADERSEGMENT A
ZWEIGENDE 1 → NULL
ZWEIGENDE 2 → NULL
ERSTER LINKER ZWEIG → SEITENZWEIG C
ERSTER RECHTER ZWEIG → NULL
LETZTER LINKER ZWEIG → SEITENZWEIG B
LETZTER RECHTER ZWEIG → NULL

670

ABZWEIGUNG B

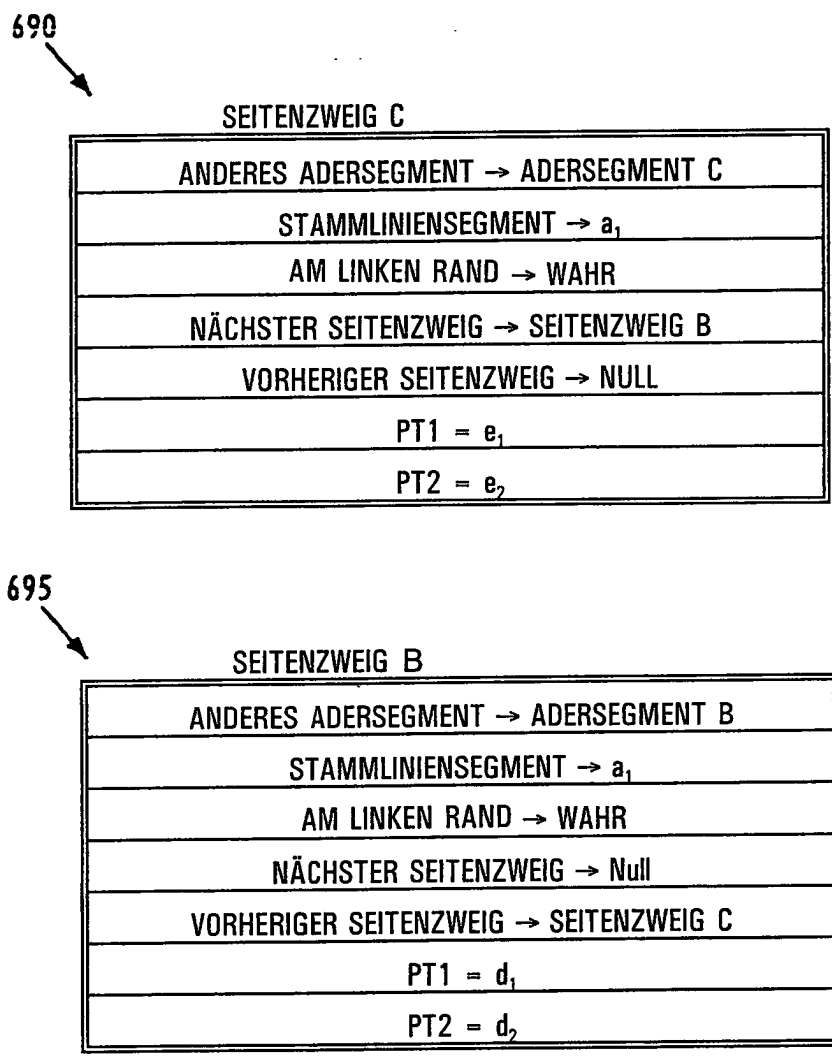
STAMMADERSEGMENT → ADERSEGMENT B
ZWEIGENDE 1 → ADERSEGMENT A
ZWEIGENDE 2 → NULL
ERSTER LINKER ZWEIG → NULL
ERSTER RECHTER ZWEIG → NULL
LETZTER LINKER ZWEIG → NULL
LETZTER RECHTER ZWEIG → NULL

680

ABZWEIGUNG C

STAMMADERSEGMENT → ADERSEGMENT C
ZWEIGENDE 1 → NULL
ZWEIGENDE 2 → ADERSEGMENT A
ERSTER LINKER ZWEIG → NULL
ERSTER RECHTER ZWEIG → NULL
LETZTER LINKER ZWEIG → NULL
LETZTER RECHTER ZWEIG → NULL

FIG. 6E



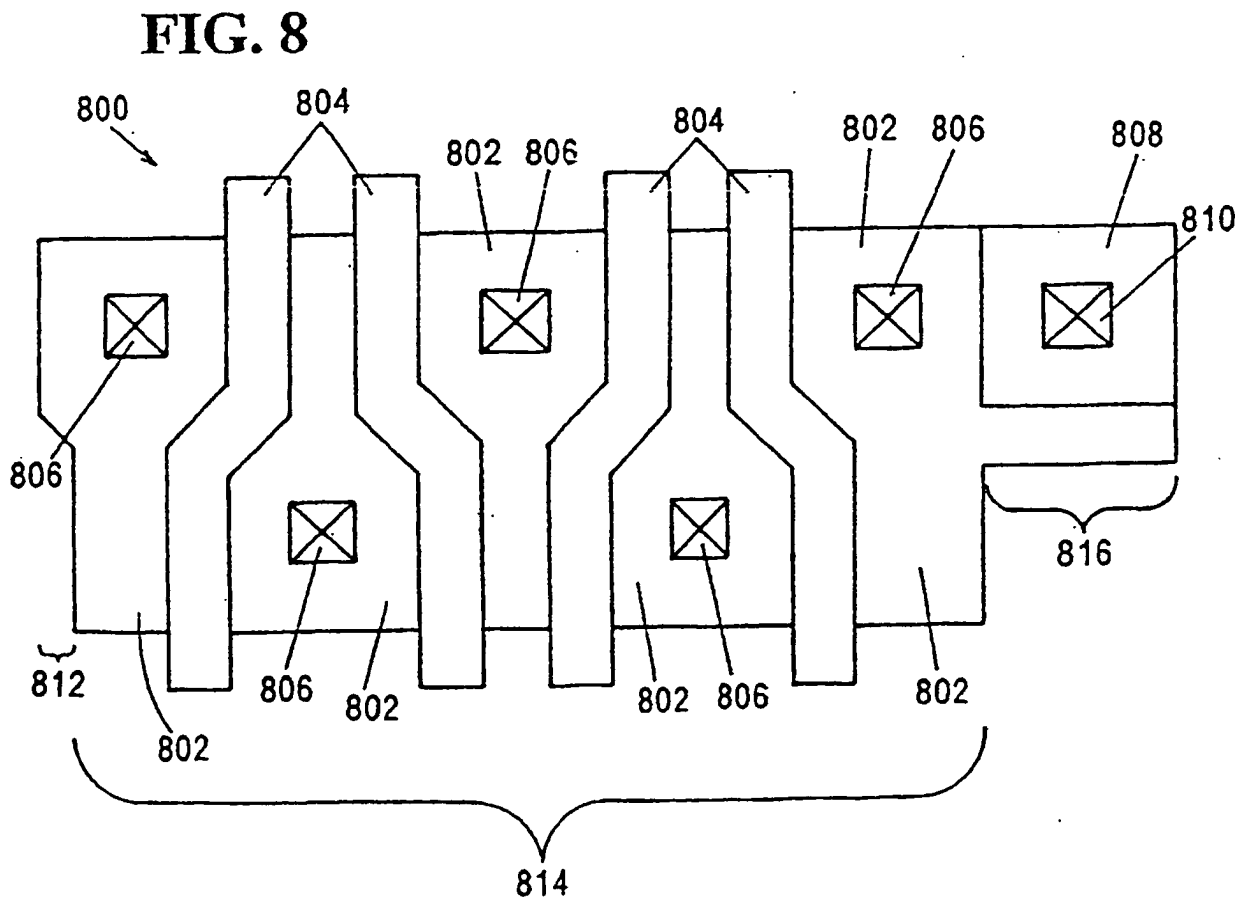
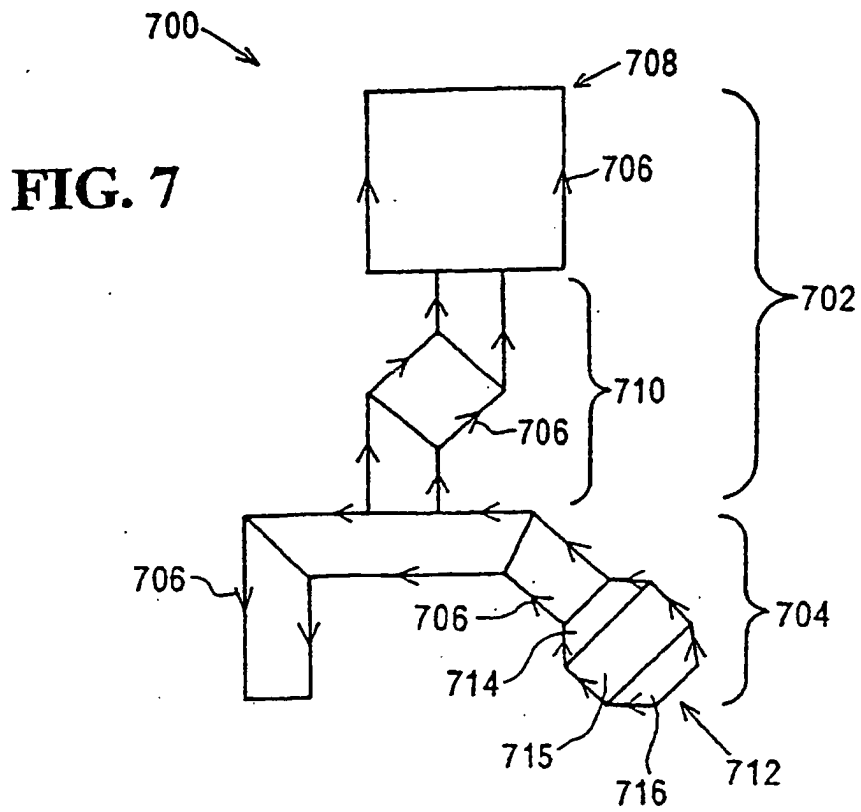


FIG. 9A

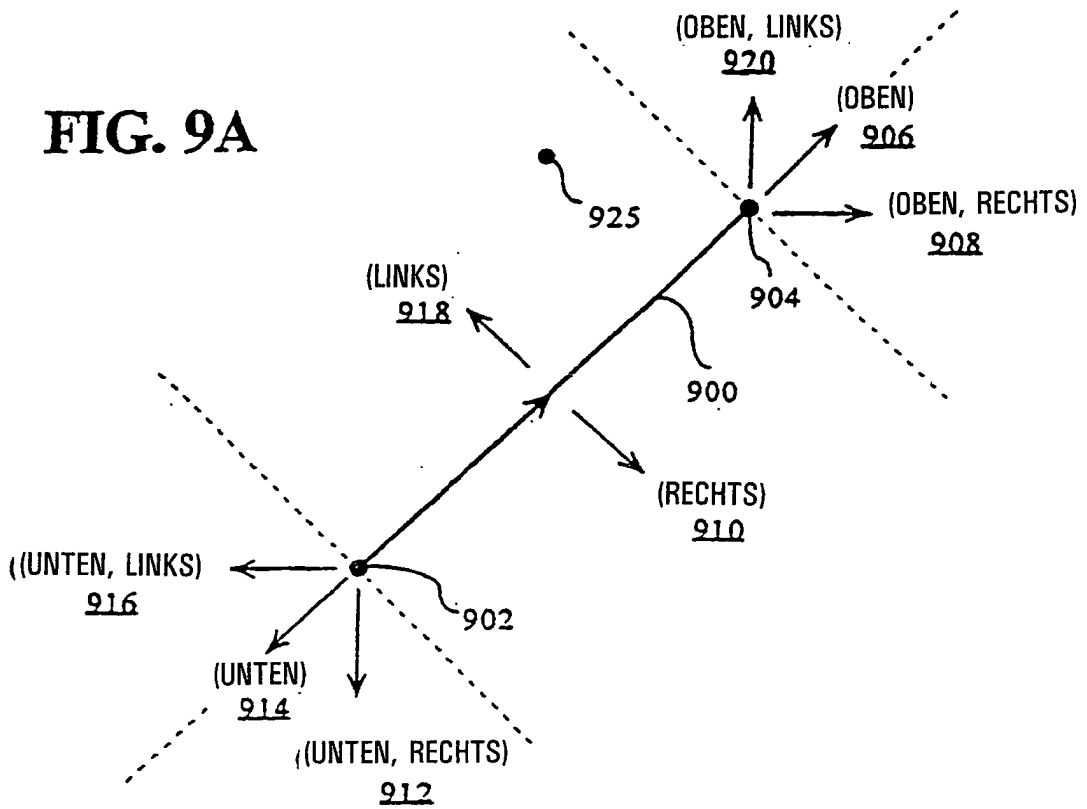


FIG. 9B

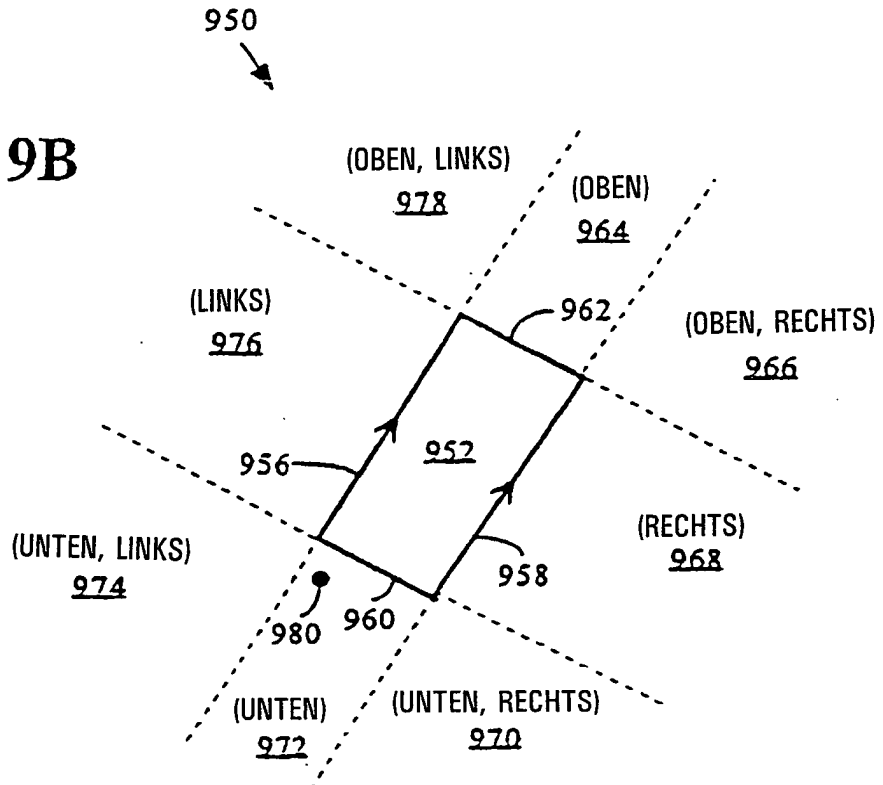


FIG. 10

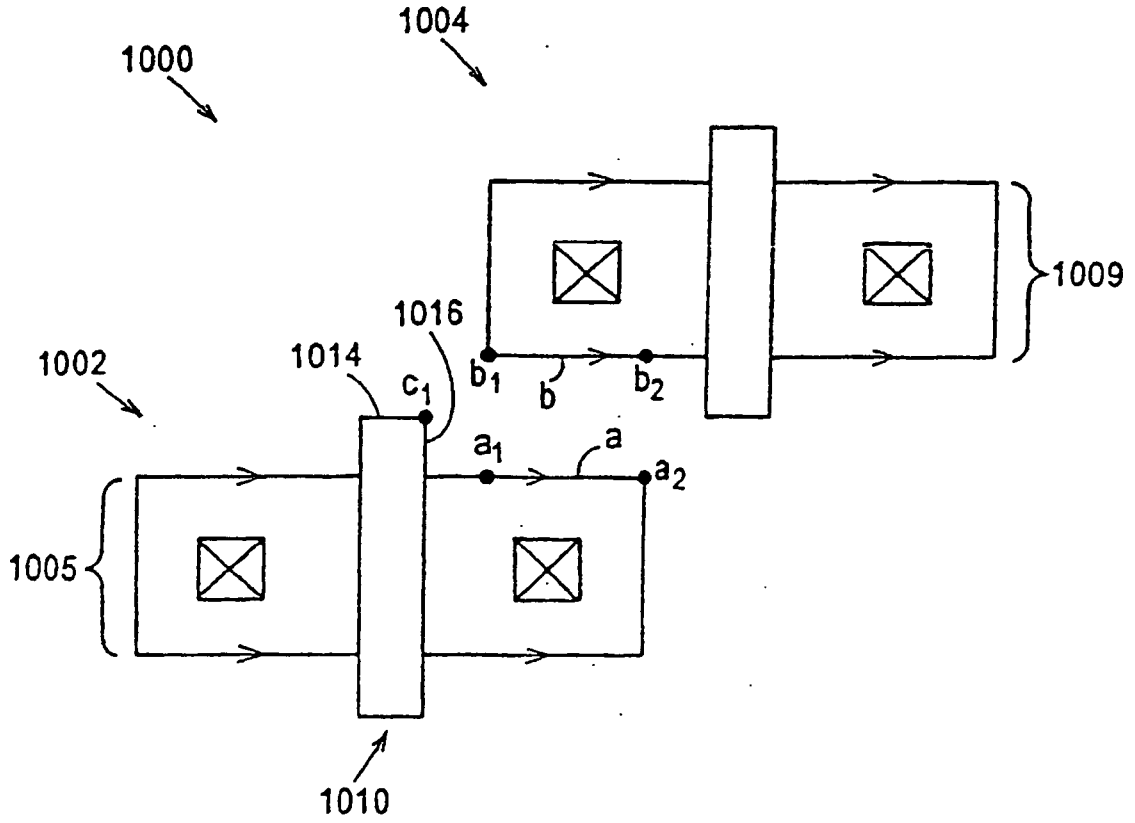


FIG. 11

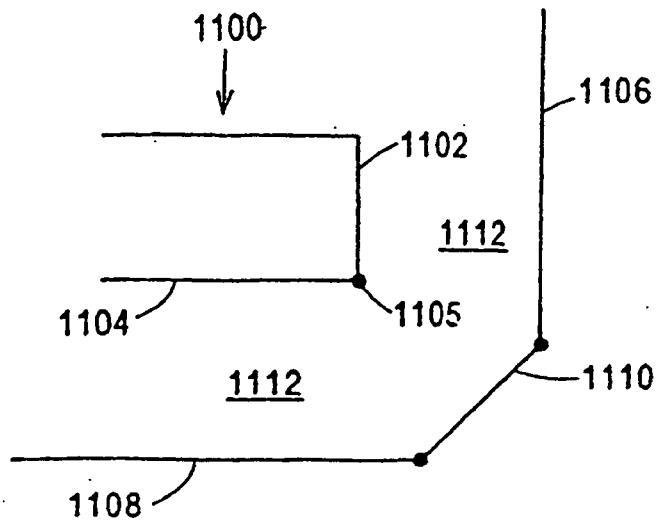


FIG. 12

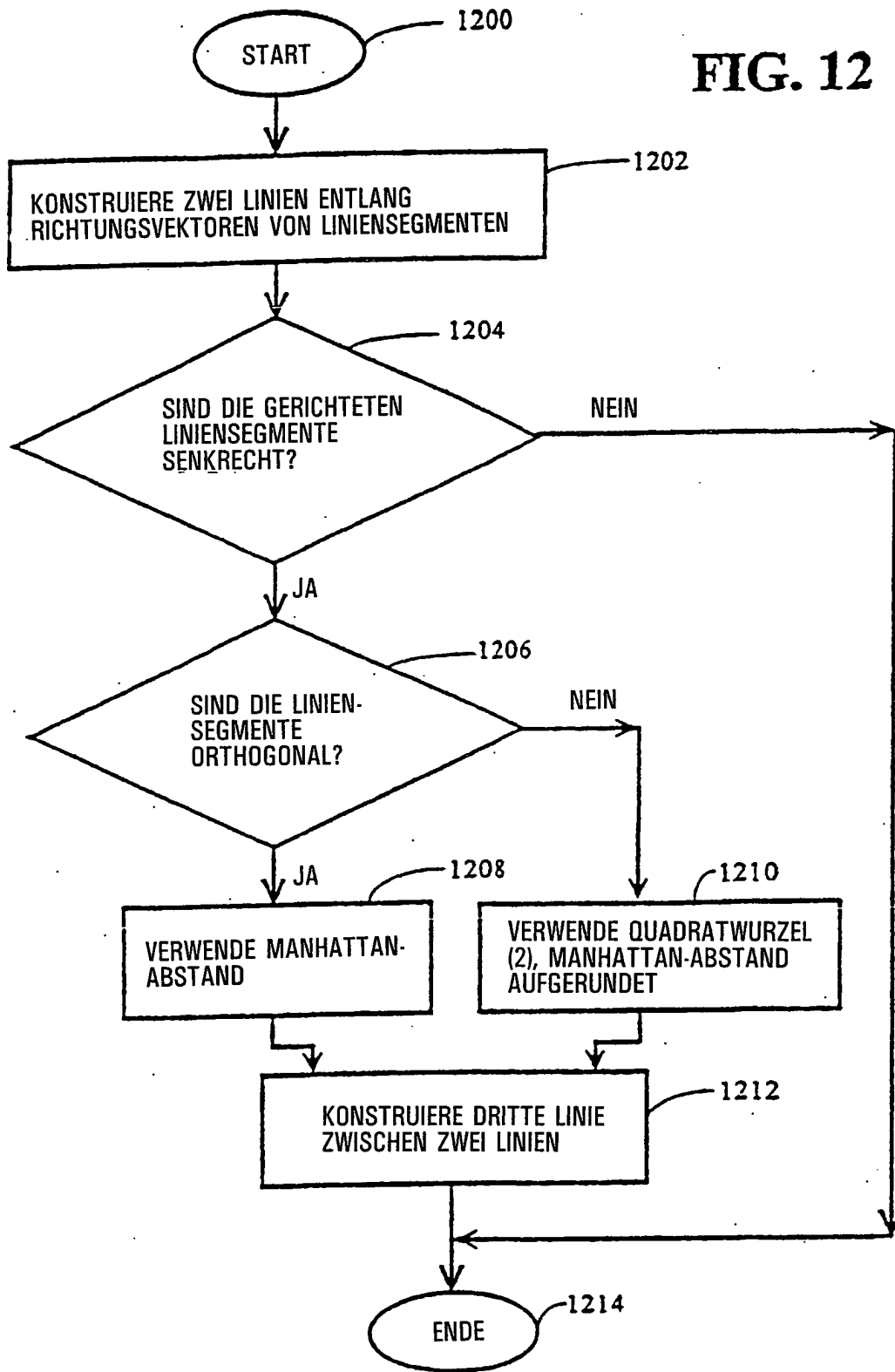


FIG. 13A

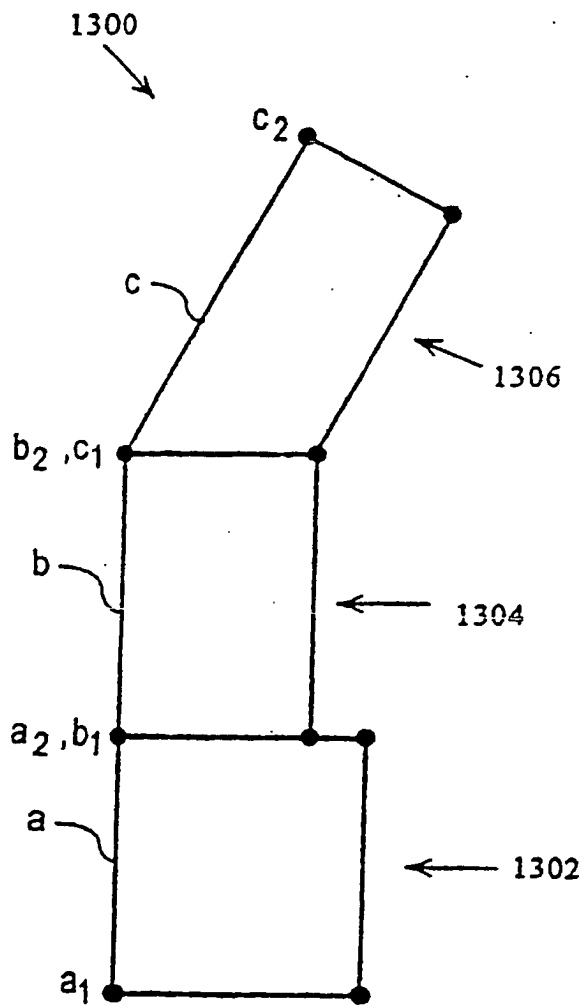


FIG. 13B

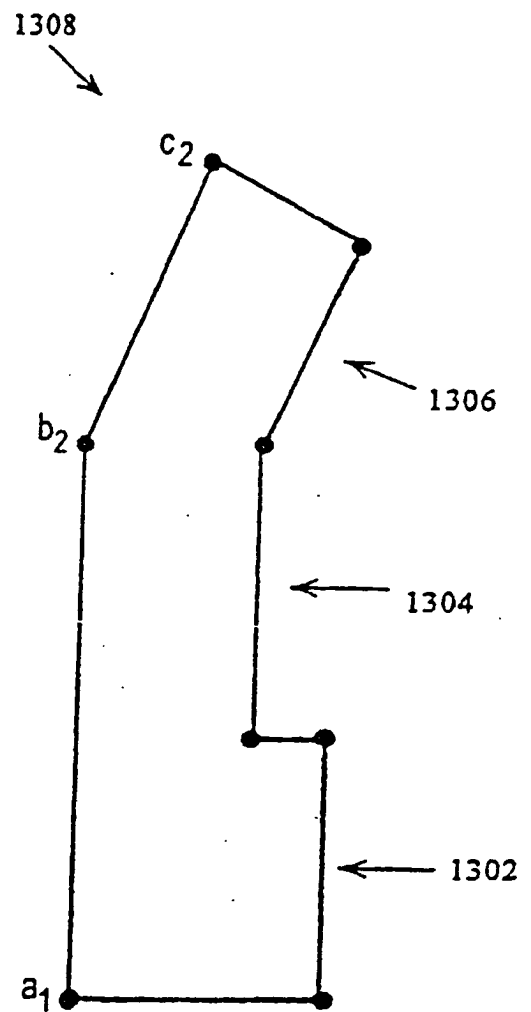


FIG. 14

