



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) DE 100 85 387 T5 2004.07.01

(12)

Veröffentlichung

der internationalen Anmeldung mit der

(87) Veröffentlichungs-Nr.: **WO 01/043400**

in deutscher Übersetzung (Art. III § 8 Abs. 2 IntPatÜG)

(21) Deutsches Aktenzeichen: **100 85 387.0**

(86) PCT-Aktenzeichen: **PCT/CA00/01441**

(86) PCT-Anmeldetag: **08.12.2000**

(87) PCT-Veröffentlichungstag: **14.06.2001**

(43) Veröffentlichungstag der PCT Anmeldung
in deutscher Übersetzung: **01.07.2004**

(51) Int Cl.⁷: **H04L 29/12**

(30) Unionspriorität:

60/170,232	10.12.1999	US
60/212,966	21.06.2000	US

(74) Vertreter:

**LENZING GERBER Patentanwälte, 40212
Düsseldorf**

(71) Anmelder:

**Mosaic Technologies Incorporated, Kanata,
Ontario, CA**

(72) Erfinder:

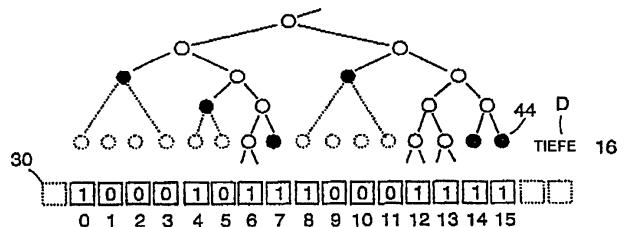
Brown, David A., Carp, Ontario, CA

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: **Verfahren und Vorrichtung zur Adresssuche längster Übereinstimmung**

(57) Hauptanspruch: Nachschlagetabelle, umfassend:

eine erste Nachschlageeinheit, welche einen Schlüssel empfängt und ein erstes Ergebnis durch eine Mehrstufen-suche in der ersten Nachschlageeinheit bereitstellt; und eine zweite Nachschlageeinheit, welche den Schlüssel empfängt und ein zweites Ergebnis durch eine Mehrstufen-suche in der zweiten Nachschlageeinheit bereitstellt, wobei sowohl die erste Nachschlageeinheit als auch die zweite Nachschlageeinheit den Schlüssel parallel empfangen und nur eine der Einheiten ein Endergebnis für den Schlüssel bereitstellt, welches nur in einer der Nachschlageeinheiten in Abhängigkeit von dem ersten Ergebnis und dem zweiten Ergebnis gespeichert wird.



Beschreibung**HINTERGRUND DER ERFINDUNG**

[0001] Das Internet ist eine Reihe von Netzwerken, welche durch Router verbunden sind. Ein Router unterhält eine Routentabelle, welche für jedes mögliche Zielnetzwerk den nächsten Sprung angeibt, zu welchem ein empfangenes Datenpaket weitergeleitet werden soll. Der nächste Sprung kann ein anderer Router oder das Endziel sein.

[0002] Ein Internetprotokoll ("IP")-Datenpaket, welches an einem Port in einem Router empfangen wird, enthält eine IP-Zieladresse. Die IP-Zieladresse ist das Endziel des IP-Datenpakets. Gegenwärtig gibt es zwei Versionen von IP, und zwar IP-Version 4 ("IPv4") und IP-Version 6 ("IPv6"). Die IPv4 stellt ein 32-Bit-Feld in einem IP-Header, welcher im Datenpaket enthalten ist, zum Speichern der IP-Zieladresse bereit. Der Router leitet ein empfangenes Datenpaket verbunden an einen Folgeschleifen-Router oder die Zieladresse weiter, wenn das Ziel das örtliche Netzwerk ist, in Abhängigkeit von der IP-Zieladresse, welche im IP-Header gespeichert ist.

[0003] Eine 32-Bit-IPv4-Zieladresse stellt 4 Milliarden mögliche Routen bereit. Ein Internet-Router speichert normalerweise 50.000 der 4 Milliarden möglichen Routen. Die Anzahl von gespeicherten Routen nimmt jedoch mit dem Wachstum des Internets und der weitverbreiteten Verwendung von IPv6 zu.

[0004] Ursprünglich war der IP-Adressraum in drei Klassen von IP-Adressen unterteilt; und zwar A, B und C. Jeder IP-Adressraum war in eine Netzwerkadresse und eine Host-Adresse unterteilt. Klasse A berücksichtigte 126 Netzwerke und 16 Milliarden Hosts pro Netzwerk. Klasse B berücksichtigte 16.382 Netzwerke mit 64.000 Hosts pro Netzwerk und Klasse C berücksichtigte 2 Millionen Netzwerke mit 256 Hosts pro Netzwerk. Das Unterteilen des IP-Adressraums in verschiedene Klassen verringerte jedoch die Anzahl von verfügbaren IP-Adressen. Klasse C erlaubte nur ein Maximum von 256 Hosts pro Netzwerk, was für die meisten Organisationen zu wenig ist. Daher wurde den meisten Organisationen eine Adresse der Klasse B zugeteilt, welche 64.000 Host-Adressen aufnahm, die nicht durch andere Organisationen verwendet werden konnten, selbst wenn sie durch die Organisation, der sie zugeteilt waren, nicht verwendet wurden. Hosts in einer Organisation mit einer IP-Adresse der Klasse B speichern alle dieselbe Netzwerkadresse in den 16 höchswertigen Bits ("MBSs", nach Engl. Most Significant Bits), zum Beispiel 27.32.xx.xx.

[0005] Klassenloses InterDomain-Routing ("CIDR") wurde eingeführt, um nicht verwendete IP-Hostadressen verfügbar zu machen. Die restlichen unbenutzten Netzwerke werden Organisationen in Blöcken veränderlicher Größe zugeordnet. Eine Organisation, welche 500 Adressen benötigt, bekommt 500 kontinuierliche Adressen. Zum Beispiel können einer Organisation 500 verfügbare Adressen zugeordnet werden, welche bei 128.32.xx beginnen. Die Anzahl von Routen, welche durch einen Router gespeichert werden, nahm seit der Einführung des klassenlosen InterDomain-Routings zu. Klassenloses InterDomain-Routing erfordert längste Präfixübereinstimmung, um die entsprechende Route zu finden, anstelle des Suchens nach einer übereinstimmenden Netzwerkadresse, um den entsprechenden nächsten Sprung für die IP-Zieladresse zu finden. Zum Beispiel kann eine Suche nach den 16 MBSs einer IP-Adresse der Klasse B, zum Beispiel 128.xx.xx, nicht mehr eingestellt werden, da 128.32.4.xx einer anderen Organisation zugeteilt sein kann, welche einen anderen nächsten Sprung erfordert.

[0006] Ein Verfahren zur Suche nach einer längsten Präfixübereinstimmung für einen Schlüssel ist durch die Verwendung einer Binärbaumsuche. Eine Binärbaumsuche gleicht eine 32-Bit-Eingabe bitweise mit 32 Stufen ab, wobei 32 Suchen erforderlich sind, um die Eingabe zu finden, welche mit dem 32-Bit-Schlüssel übereinstimmt. Ein anderes Verfahren zur Suche nach einer Übereinstimmung ist durch die Verwendung eines Patricia-Baumes. Ein Patricia-Baum verringert die Anzahl von Suchen, welche erforderlich sind, wenn es keine Eingaben unter einem Blatt des Binärbaums gibt.

[0007] Ein weiteres Verfahren zum wirksamen Suchen nach einem nächsten Sprung, welcher mit einer IP-Zieladresse verbunden ist, wird in der PCT-Anmeldung Serie Nummer PCT/SE98/00854 mit dem Titel "Method and Systems for Fast Routing Lookups" durch Brodnick et al., eingereicht am 11. Mai 1998, beschrieben. Das Verfahren, welches durch Brodnick beschrieben wird, verringert die Anzahl von nächsten Sprüngen, welche gespeichert werden, durch Nichtspeichern von Duplikatrouten. Durch Verringern der Anzahl von nächsten Sprüngen wird der Speicherplatzbedarf verringert, so daß eine Routen-Nachschatgetabelle in einem Cache-Schnellspeicher gespeichert werden kann.

[0008] Brodnick et al. unterteilt den Binärbaum in drei Stufen. Das Unterteilen des Binärbaums in 3 Stufen verringert die Anzahl von Suchen auf drei. Die indexierte Eingabe auf der ersten Stufe zeigt an, ob die Suche auf der ersten Stufe enden kann, wobei die Route von der Eingabe genommen wird, oder ob die Suche auf einer nachfolgenden Stufe unter Verwendung eines weiteren Abschnitts der IP-Zieladresse fortgesetzt werden muß.

[0009] **Fig. 1A** veranschaulicht eine 64-KB (65536)-Bitmap nach dem Stand der Technik, welche die erste Stufe eines Binärbaums darstellt. Eine 64-KB-Bitmap 30 stellt die Blätter oder Knoten **44** des Binärbaums bei Tiefe **16** mit einem Bit pro Knoten **44** dar. Die Bitmap ist in Bitmasken mit einer Länge von **16** unterteilt. Es sind

$2^{12} = 4.096$ Bitmasken in der 64-KB-Bitmap vorhanden. Eine Bitmaske ist in **Fig. 1A** dargestellt. Ein Bit in der Bitmap **30** wird auf '1' gesetzt, wenn ein Teilbaum oder ein Routenindex in einem Feld von Zeigern, welche dem Knoten **44** entsprechen, gespeichert wird. Ein Bit in der Bitmap **30** wird auf '0' gesetzt, wenn der Knoten sich eine Routeneingabe mit einem vorhergehenden Knoten **44** teilt.

[0010] **Fig. 1B** veranschaulicht eine Nachschlagetabelle nach dem Stand der Technik, welche im Cache-Speicher realisiert ist. Die Nachschlagetabelle umfaßt ein Feld von Codewörtern **36**, ein Feld von Basisindizes **34** und eine Map-Tabelle **40**. Eine 32-Bit-IP-Adresse **38** ist ebenfalls in **Fig. 1B** dargestellt. Ein Codewort **46** wird in dem Feld von Codewörtern **36** für jede Bitmaske in der Bitmap **30** (**Fig. 1A**) gespeichert. Das Codewort **46** umfaßt einen Sechs-Bit-Wert **46a** und ein 10-Bit-Offset **46b**. Ein Basisindex **42** wird in dem Feld von Basisindizes **34** für jede vier Codewörter **46** in dem Feld von Codewörtern **36** gespeichert.

[0011] Das Feld von Codewörtern **36**, das Feld von Basisindizes **34** und die Map-Tabelle **40** werden verwendet, um einen Zeiger in einem Feld von Zeigern (nicht dargestellt) auszuwählen. Der Zeiger speichert einen Routenindex oder einen Index, um eine weitere Suche durchzuführen.

[0012] Eine Gruppe von Zeigern in dem Feld von Zeigern wird durch auswählen eines Codewortes **46** in dem Feld von Codewörtern **36** und eines Basisindexes **42** in dem Feld von Basisindizes **34** ausgewählt. Das Codewort **46** wird unter Verwendung der ersten 12 Bits **50** der IP-Adresse **38** ausgewählt. Der Basisindex **42** wird unter Verwendung der ersten 10 Bits **48** der IP-Adresse **38** ausgewählt. Der richtige Zeiger in der Gruppe von Zeigern wird unter Verwendung der Map-Tabelle 32 ausgewählt.

[0013] Der 10-Bit-Wert **46b** in dem ausgewählten Codewort **36** ist ein Index zur Map-Tabelle 32. Die Map-Tabelle **32** bildet Bitzahlen innerhalb einer Bitmaske zu 4-Bit-Offsets ab. Das Offset spezifiziert den Zeiger innerhalb der ausgewählten Gruppe von Zeigern in dem Feld von Zeigern. Der 10-Bit-Wert **46b** wählt die Reihe in der Map-Tabelle 32 aus und Bits 19:16 der IP-Adresse **52** wählt das 4-Bit-Offset **54** aus.

[0014] Infolgedessen erfordert eine Suche nach einem Zeiger die folgenden Cache-Speicherzugriffe: (1) ein 16-Bit-Codewort **46** lesen, (2) eine 16-Bit-Basisadresse **42** lesen; (3) ein 4-Bit-Offset **54** von der Map-Tabelle **32** ablesen; (4) einen Zeiger bei einem Zeigerindex lesen, wobei der Zeigerindex die Summe der Basisadresse **42**, des Codewort-Offsets **46a** und des 4-Bit-Offsets **54** ist.

[0015] Dieselben Speicherzugriffe sind für jede Stufe des Binärbaumes erforderlich. Infolgedessen erfordert ein Absuchen von drei Stufen **12** Speicherzugriffe.

ZUSAMMENFASSUNG DER ERFINDUNG

[0016] Anwaltverzeichnis Nummer 2037.2004-001 mit dem Titel "Verfahren und Vorrichtung zur Adressuche längster Übereinstimmung" durch David A. Brown beschreibt eine Nachschlagetabelle zum Durchführen von Mehrstufensuchen nach einem Wert, welcher einem Schlüssel entspricht. Die Anzahl von Routenindizes, welche in der Nachschlagetabelle gespeichert werden können, ist durch die Anzahl von Speicherstellen in der Nachschlagetabelle beschränkt.

[0017] Gemäß den Prinzipien der Erfindung wird eine Nachschlagetabelle vorgestellt, welche eine erste Nachschlageeinheit und eine zweite Nachschlageeinheit umfaßt. Die erste Nachschlageeinheit empfängt einen Schlüssel und stellt ein erstes Ergebnis durch eine Mehrstufensuche in der ersten Nachschlageeinheit bereit. Die zweite Nachschlageeinheit empfängt den Schlüssel und stellt ein zweites Ergebnis durch eine Mehrstufensuche in der zweiten Nachschlageeinheit bereit. Sowohl die erste Nachschlageeinheit als auch die zweite Nachschlageeinheit empfangen den Schlüssel parallel und nur eine der Einheiten stellt ein Endergebnis für den Schlüssel bereit, welcher nur in einer der Nachschlageeinheiten gespeichert wird, in Abhängigkeit von dem ersten Ergebnis und dem zweiten Ergebnis.

[0018] Die Endergebnisse werden unter den Tabellen in Abhängigkeit von der Gesamtzahl von Endergebnissen, welche in der Nachschlagetabelle zu speichern sind, verteilt. Der Schlüssel kann eine IP-Adresse sein und das Endergebnis ist ein Routenindex für die IP-Adresse.

[0019] Eine erste Speicherstelle in der ersten Nachschlageeinheit, welche durch den Schlüssel indexiert wird, speichert das Endergebnis und eine zweite Speicherstelle in der zweiten Nachschlageeinheit, welche durch den Schlüssel indexiert wird, speichert eine Anzeige, daß das Endergebnis in der ersten Nachschlageeinheit gespeichert wird. Jede der ersten Nachschlageeinheit und der zweiten Nachschlageeinheit umfaßt eine Mehrzahl von Mapper-Stufen und jede der Mapper-Stufen umfaßt Speicherstellen zum Speichern von Ergebnissen.

[0020] In einem Ausführungsbeispiel wird das Endergebnis, welches dem Schlüssel entspricht, der an einer Stelle in irgendeiner der Mehrzahl von Mapper-Stufen in der ersten Nachschlageeinheit gespeichert wird, an die zweite Nachschlageeinheit weitergeleitet und durch die Nachschlageeinheit bereitgestellt.

[0021] In einem alternativen Ausführungsbeispiel wird das Endergebnis, welches dem Schlüssel entspricht, durch die Nachschlageeinheit bereitgestellt, in welcher das Endergebnis gespeichert wird.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

- [0022] **Fig.** 1A veranschaulicht eine Bitmap nach dem Stand der Technik zur Darstellung der ersten Stufe eines Binärbaumes;
- [0023] **Fig.** 1B veranschaulicht eine Nachschlagetabelle nach dem Stand der Technik, welche im Cache-Speicher realisiert ist;
- [0024] **Fig.** 2A veranschaulicht eine Präfixnachschlagetabelle längster Übereinstimmung gemäß den Prinzipien der vorliegenden Erfindung;
- [0025] **Fig.** 2B veranschaulicht eine Binärbaumdarstellung der Routenindizes, welche in der Nachschlagetabelle, die in **Fig.** 2A dargestellt ist, gespeichert werden;
- [0026] **Fig.** 3 veranschaulicht eine Präfixnachschlagetabelle längster Übereinstimmung für einen 40-Bit-Schlüssel gemäß den Prinzipien der vorliegenden Erfindung;
- [0027] **Fig.** 4 veranschaulicht die Typen von Mapper-Eingaben, welche in dein direkt abgebildeten Mapper, der in **Fig.** 2A dargestellt ist, gespeichert werden können;
- [0028] **Fig.** 5 veranschaulicht den Mapper, welcher den Knoten auf der Mapper-Stufe_2 112b, die in **Fig.** 2B dargestellt ist, entspricht;
- [0029] **Fig.** 6A ist eine Binärbaumdarstellung eines Teilbaumes;
- [0030] **Fig.** 6B veranschaulicht den dichtbesiedelten Teilbaum-Deskriptor, welcher in dem Datenfeld gespeichert wird, das in **Fig.** 5 dargestellt ist, und den Knoten auf der unteren Stufe des Teilbaumes entspricht, welcher in **Fig.** 6A dargestellt ist;
- [0031] **Fig.** 7 veranschaulicht das Zeigerfeld, welches in **Fig.** 5 dargestellt ist;
- [0032] **Fig.** 8 veranschaulicht die Mapper-Adreßlogik, welche in **Fig.** 5 dargestellt ist;
- [0033] **Fig.** 9 ist ein Ablaufdiagramm, welches die Schritte zur Suche nach der längsten Übereinstimmung veranschaulicht;
- [0034] **Fig.** 10A veranschaulicht ein Ausführungsbeispiel zur Bereitstellung von Tiefenerweiterung;
- [0035] **Fig.** 10B veranschaulicht eine von Nachschlagetabellen in dem Ausführungsbeispiel, welches in **Fig.** 10A dargestellt ist;
- [0036] **Fig.** 10C veranschaulicht ein anderes Ausführungsbeispiel zur Bereitstellung von Tiefenerweiterung, um die Anzahl von Mapper-Eingaben, welche zum Speichern von Werten verfügbar sind, zu erhöhen;
- [0037] **Fig.** 10D veranschaulicht die Nebennachschlagetabelle in dem Ausführungsbeispiel, welches in **Fig.** 10C dargestellt ist;
- [0038] **Fig.** 11-B veranschaulichen eine Binärbaumdarstellung der Verteilung der Knoten in der Binärbaumdarstellung von Routenindizes, welche in **Fig.** 2B dargestellt sind, unter den Mapper-Eingaben in den Nachschlagetabellen, welche in **Fig.** 10A und 10C dargestellt sind;
- [0039] **Fig.** 12 ist ein Ablaufdiagramm, welches ein Verfahren zur Verteilung von Werten unter den Mapper-Eingaben in den Nachschlagetabellen, die in **Fig.** 10A und 10C dargestellt sind, veranschaulicht;
- [0040] **Fig.** 13 ist ein Ablaufdiagramm, welches ein Verfahren zur Suche nach einem Wert veranschaulicht, der einem Suchschlüssel entspricht, welcher in einer von Mapper-Eingaben in den Nachschlagetabellen gespeichert ist, die in **Fig.** 10C dargestellt sind;
- [0041] **Fig.** 14 ist ein Ablaufdiagramm, welches ein Verfahren zur Suche nach einem Wert veranschaulicht, der einem Suchschlüssel entspricht, welcher in einer der Mapper-Eingaben in den Nachschlagetabellen gespeichert ist, die in **Fig.** 10A dargestellt sind;
- [0042] **Fig.** 15 veranschaulicht eine Binärbaumdarstellung eines dünnbesiedelten Teilbaumes und eines dichtbesiedelten Teilbaumes auf der zweiten Mapper-Stufe, welche durch die erste Mapper-Stufe indexiert wird;
- [0043] **Fig.** 16A–C veranschaulichen die Abänderung des Datenfeldes und des Zeigerfeldes in der Teilbaumeingabe, welche in **Fig.** 5 dargestellt ist, und der Teilbaumeingabe, welche in **Fig.** 4 dargestellt ist, um die Speicherung einer Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren in der Teilbaumeingabe zu erlauben;
- [0044] **Fig.** 17 veranschaulicht die Teilbaum-Logik des dünnbesiedelten Modus in der Offset-Logik, welche in **Fig.** 8 dargestellt ist, um ein Block-Offset für einen Knoten in einem dünnbesiedelten Teilbaum auszuwählen;
- [0045] **Fig.** 18 veranschaulicht die Logik des dünnbesiedelten Modus, welche in der Offset-Logik in **Fig.** 17 dargestellt ist;
- [0046] **Fig.** 19A–D veranschaulichen die Auswahl eines Block-Offsets für einen Knoten in einem dünnbesiedelten Teilbaum;
- [0047] **Fig.** 20 ist ein Blockdiagramm, welches die Basisauswahl-Logik des dünnbesiedelten Modus in der Zeigerlogik, die in **Fig.** 8 dargestellt ist, veranschaulicht;
- [0048] **Fig.** 21 veranschaulicht einen dichtbesiedelten Teilbaum-Deskriptor und einen dünnbesiedelten Teilbaum-Deskriptor, welche im Teilbaumspeicher gespeichert sind;
- [0049] **Fig.** 22 ist ein Ablaufdiagramm, welches ein Verfahren zur Bereitstellung einer Mapper-Adresse für eine Mapper-Eingabe in einem Teilbaum-Mapper veranschaulicht, welcher eine Route für einen Knoten in ei-

nem dünnbesiedelten Teilbaum und einem dichtbesiedelten Teilbaum speichert;

[0050] **Fig.** 23 veranschaulicht eine Binärbaumdarstellung einer neuen Route, welche zur Nachschlagetabelle zu addieren ist;

[0051] **Fig.** 24 veranschaulicht Aktualisierungsrouten, welche in einem Prozessorspeicher gespeichert werden;

[0052] **Fig.** 25 veranschaulicht die neue Route, welche in **Fig.** 23 dargestellt ist und in der Nachschlagetabelle gespeichert wird;

[0053] **Fig.** 26 ist ein Ablaufgramm, welches die Schritte zum Addieren der neuen Route zur Nachschlagetabelle, welche in **Fig.** 25 dargestellt ist, veranschaulicht.

[0054] Die vorhergehenden und andere Aufgaben, Merkmale und Vorteile der Erfindung sind ersichtlich aus der anschließenden ausführlicheren Beschreibung von bevorzugten Ausführungsbeispielen der Erfindung, wie in den beiliegenden Zeichnungen veranschaulicht, in welchen ähnliche Bezugszeichen dieselben Teile durch die verschiedenen Ansichten hindurch bezeichnen. Die Zeichnungen sind nicht unbedingt maßstabsgerecht, sondern das Hauptaugenmerk wird auf die Veranschaulichung der Prinzipien der Erfindung gelegt.

AUSFÜHRLICHE BESCHREIBUNG DER ERFINDUNG

[0055] Es folgt eine Beschreibung von bevorzugten Ausführungsbeispielen der Erfindung.

Dichtbesiedelter Modus

[0056] **Fig.** 2A veranschaulicht eine Präfixnachschlagetabelle **100** längster Übereinstimmung gemäß den Prinzipien der vorliegenden Erfindung. Die Nachschlagetabelle **100** stellt einen Routenindex **102** für einen Schlüssel **104** bereit. Der Routenindex **102** wird verwendet, um auf den nächsten Sprung für die IP-Zieladresse zuzugreifen. In dem Ausführungsbeispiel, welches in **Fig.** 2A dargestellt ist, ist der Schlüssel **104** zwar 32 Bits breit, ist der Schlüssel **104** jedoch nicht auf 32 Bits beschränkt. Die Nachschlagetabelle **100** umfaßt drei Mapper **106a–c**. Jeder Mapper **106a–c** umfaßt einen getrennt adressierbaren Speicher. Der Routenindex **102** oder ein Standardroutenindex, welcher dem Schlüssel **104** entspricht, wird an einer Stelle in einem der Mapper **106a–c** gespeichert. Die Mapper-Ausgaben **110a–c** von jedem Mapper werden in einem Verzögerungsspeicher **150a–c** gespeichert, bis auf alle Mapper **106a–c** für den Schlüssel zugegriffen wurde, wenn mehrfache Mapper abgesucht werden müssen.

[0057] Der Multiplexer **108** wählt eine der Mapper-Ausgaben **110a–c**, welche zu den Eingängen des Multiplexers **108** weitergeleitet werden, als den Routenindex **102** aus. Die Mapper-Ausgabe **110a–c** wird in Abhängigkeit vom höchstwertigen Bit ("MSB", nach Engl. Most Important Bit) der Mapper-Ausgabe **110a–c** ausgewählt. Das MSB der Mapper-Ausgabe **110a–c** ist nur dann '1', wenn die Mapper-Ausgabe **110a–c** einen Routenindex **102** beinhaltet.

[0058] **Fig.** 2B veranschaulicht eine Binärbaumdarstellung der Eingaben, welche in den Mappern **106a–c** in der Nachschlagetabelle **100**, welche in **Fig.** 2A dargestellt ist, gespeichert werden. **Fig.** 2B wird in Verbindung mit **Fig.** 2A beschrieben. Der 32-Bit-Schlüssel **104** kann als ein Binärbaum mit 32 Stufen dargestellt werden. Eine Binärbaumrealisierung erfordert 32 Suchen, um bitweise 32 Stufen abzusuchen. Um die Anzahl von Suchen zu verringern, werden die 32 Stufen des Binärbaumes in drei Mapper-Stufen **112a–c** unterteilt, wobei jede Mapper-Stufe **112a–c** einem Mapper **106a–c** (**Fig.** 2A) entspricht. Die Mapper-StufeM1 **112a** enthält die ersten 16 der 32 Stufen des Binärbaumes. Der Einfachheit halber sind jedoch nur 5 der 16 Stufen in **Fig.** 2B dargestellt. Die Mapper-Stufe_2 **112b** enthält die nächsten 8 Stufen des 32-stufigen Binärbaumes, wobei drei der acht Stufen in **Fig.** 2B dargestellt sind. Die Mapper-Stufe_3 **112c** enthält die letzten 8 Stufen des 32-stufigen Binärbaumes, wobei drei der acht Stufen in **Fig.** 1B dargestellt sind. Jede Mapper-Stufe **112a–c** umfaßt eine Mehrzahl von Knoten. Die 32 Stufen so zu unterteilen, daß 16 Stufen (16 MSBs des Schlüssels **104**) auf der Mapper-Stufe_1 **112a**, 8 Stufen auf der Mapper-Stufe_2 **112b** und 8 Stufen auf der Mapper-Stufe_3 **112c** sind, scheint in der gegenwärtigen Speichertechnologie optimal zu sein; die Erfindung ist jedoch nicht auf diese Konfiguration beschränkt.

[0059] Anstatt 16 getrennte bitweise Suchen nach den ersten 16 Bits des Schlüssels **104** durchzuführen, werden die Routenindizes **102**, welche mit den ersten 16 Bits des Schlüssels **104** verbunden sind, im Mapper **106a** (**Fig.** 2A) gespeichert. Der Mapper **106a** (**Fig.** 2A) wird mit den ersten 16 MSBs-Bits des Schlüssels **104** direkt indexiert. Der nächste Mapper **106b** wird abgesucht in Abhängigkeit davon, ob der vorhergehende Mapper **106a** den Routenindex **102** speichert, welcher verwendet wird, um auf die nächste Sprunginformation, die mit dem Schlüssel **104** verbunden ist, zuzugreifen.

[0060] Wie in **Fig.** 2B dargestellt, umfassen die Knoten oder Blätter, welche auf der Mapper-Stufe_1 **112a** dargestellt sind, zwei Routen **114**, **116**, welche mit r0 bzw. s1 gekennzeichnet sind, und zwei Zeiger **130⁴** und **130²³** zu Mapper-Stufe_2 **112b**, welche mit s0 bzw. s1 gekennzeichnet sind. Ein Routenindex **102** für jede Route **114**, **116** wird im L1-Mapper **106a** gespeichert. Ebenso wird ein Adresszeiger **120** für den L2-Mapper **106b** für den

Teilbaumindex **130⁴** gespeichert und wird ein Adresszeiger (nicht dargestellt) für den L2-Mapper **106b** für den Teilbaum **130²³** gespeichert. Ein Adresszeiger **120**, welcher für den Teilbaumindex **130⁴** in der Mapper-Eingabe **140⁴** im Mapper **106a** gespeichert wird, zeigt an, daß ein Absuchen der nächsten Stufe notwendig ist, um einen Routenindex **102**, welcher mit dem Schlüssel **104** verbunden ist, zu finden.

[0061] Der Wert irgendeines Knotens im Baum kann durch Verfolgen eines Pfades von der Wurzel **114** bestimmt werden. Jeder Knoten im Binärbaum ist mit zwei Nachfolgern dargestellt, einem rechten Nachfolger und einem linken Nachfolger. Der rechte Nachfolger wird gewählt, wenn der Vorgängerknoten '1' ist. Der linke Nachfolger wird gewählt, wenn der Vorgängerknoten '0' ist. Bei Verfolgen des Pfades von der Wurzel **114** zum Knoten wird **r1** als der Routenindex **102** im L1-Mapper **106a** für alle Schlüssel mit den MSBs auf '010' gesetzt gespeichert. Bei Verfolgen des Pfades vom Wurzelknoten **114** zum s0-Knoten **130⁴** wird **s0** im L1-Mapper **106a** für alle Schlüssel mit den MSBs auf '00011' gesetzt gespeichert.

[0062] Der L1-Mapper **106a** ist ein direkt abgebildeter Mapper und speichert einen Routenindex **102** für jeden/s Unterstufen-Knoten oder -Blatt auf der unteren Stufe der Mapper-Stufe_1 **112a**. Die untere Stufe der Mapper-Stufe_1 **112a** ist die sechzehnte Stufe des 32-stufigen Binärbaumes. Die sechzehnte Stufe weist 64-KB-Knoten auf. Zum Zwecke der Veranschaulichung ist die untere Stufe der Mapper-Stufe_1 **112a** jedoch als Stufe **5** des 32-stufigen Binärbaumes dargestellt. Die Routenindizes **102**, welche im L1-Mapper **106a** dargestellt sind, entsprechen den Knoten **130¹–130³²** von Stufe **5** der Mapper-Stufe_1 **112a**. Bei Verfolgen des Pfades vom Wurzelknoten **114** zu den Knoten **130¹, 130², 130³** der Stufe **5** ist der Routenindex **102 r0**. Infolgedessen wird **r0** an den Stellen **140¹, 140², 140³** des L1-Mappers **106a** gespeichert; das heißt, bei Index 00000, 00001 und 00010. Knoten **130⁴** speichert einen Teilbaumindex **s0**, weshalb **s0** an der Stelle **140⁴** im L1-Mapper **106a** bei Adresse **00011** gespeichert wird. Auf ähnliche Weise ist der Routenindex **102** für die Knoten **130⁵–130⁸** Stufe **5 r0**, weshalb die Stellen **140⁵, 140⁶, 140⁷, 140⁸** bei den Adressen **00100, 00101, 00110** und **00111** im L1-Mapper **106a r0** speichern. Der Routenindex **102** für die Knoten **130⁹–130¹²** der Stufe **5** ist **r1**, weshalb die Stellen **140⁹, 140¹⁰** bei den Adressen **01000** und **010001** im L1-Mapper **106a r1** speichern.

[0063] Jede Stelle im L1-Mapper **106a** speichert einen Routenindex **102**, welcher dem Knoten **300¹–300³²** der Stufe **5** direkt oder durch einen Vorgänger des Knotens **300¹–32** der Stufe **5** oder einen Adresszeiger zum nächsten Mapper **106b–c** zugewiesen ist. Die Mapper-Stufe_3 **106c** umfaßt zwei Host-Knoten **h0** bei Knoten **138** und **h1** bei Knoten **140** auf der unteren Stufe des 32-stufigen Binärbaumes. Eine Suche nach einem Host-Knoten erfordert ein Absuchen aller Bits des Schlüssels **104**. Wie in Fig. 2A dargestellt, wird der Routenindex **102** für **h0** an der Stelle **146⁴** im L3-Mapper **106c** gespeichert. Im Gegensatz zum L1-Mapper **106a** werden der L2-Mapper **106b** und der L3-Mapper **106c** nicht direkt abgebildet.

[0064] In den Mappern **106b** und **106c** wird ein Routenindex **102** nicht für jede mögliche Eingabe gespeichert. Ein Routenindex **102** wird nur gespeichert, wenn der Routenindex **102** für den Knoten sich von dem zuvor gespeicherten Routenindex **102** im Mapper **106b–c** unterscheidet. Bei Betrachten der Knoten der Stufe 3 im ersten Teilbaum A, welcher auf der Mapper-Stufe_2 **112b** dargestellt ist, ist der Routenindex **102** für den Knoten **132¹** und Knoten **132² r0**, weshalb ein Routenindex für **r0** an der Stelle **142¹** sowohl für den Knoten **132¹** als auch den Knoten **132²** im L2-Mapper **106b** gespeichert wird. Ein Teilbaumindex **s0** für den Knoten **132²** wird an der Stelle **142²** gespeichert. Der Routenindex **102**, welcher mit dem Knoten **132⁴** der Stufe 3 und den Knoten **132⁵** und **132⁶** der Stufe 3 verbinden ist, ist **r0**, welcher sich vom **s0**, welcher für den vorhergehenden Knoten **132²** gespeichert wurde, unterscheidet, weshalb **r0** an der nächsten Stelle **142³** im L2-Mapper **106b** gespeichert wird. Die Route **r2** wird an der Stelle **142⁴** im L2-Mapper **106b** für den Knoten **132⁷** gespeichert, da der Knoten **132⁷** nicht dieselbe Route benutzt wie der vorhergehende Knoten **132⁶**. Der Teilbaumindex **s3** wird für den nächsten Knoten **132⁸** der Stufe 3 gespeichert, weshalb **s3** an der Stelle **142⁵** im L2-Mapper **106b** gespeichert wird. Dadurch, daß ein Routenindex nur gespeichert wird, wenn es eine Routenänderung von dem vorhergehenden Knoten gibt, wird der Speicher, welcher zum Speichern von Routenindizes **102** erforderlich ist, verringert. Wie dargestellt, werden nur fünf Stellen im L2-Mapper **106b** zum Speichern von Routenindizes für die acht Knoten **132¹–132⁸** der Stufe 3 im ersten Teilbaum A auf der Mapper-Stufe_2 **112b** benötigt. Der nicht direkt abgebildete Mapper **106b, 106c** wird später in Verbindung mit Fig. 5 ausführlicher beschrieben.

[0065] Fig. 3 veranschaulicht eine Präfixnachschlagetabelle **200** längster Übereinstimmung für einen 40-Bit-Schlüssel **210** gemäß den Prinzipien der vorliegenden Erfindung. In einem Ausführungsbeispiel umfaßt der 40-Bit-Schlüssel ein 8-Bit-Präfix und eine 32-Bit-IP-Adresse. Das 8-Bit-Präfix kann ein Identifikator für ein virtuelles Privatnetz ("VPN") sein, welcher mit der 32-Bit-IP-Adresse verbunden ist. Die Nachschlagetabelle **200** umfaßt vier Mapper **106a–d**. Der Mapper **106a** ist ein direkt abgebildeter Mapper, wie in Verbindung mit Fig. 2A beschrieben. Die Mapper **106a–d** sind indirekte Mapper. Der Mapper **106a** speichert einen Routenindex **102** oder einen Teilbaumindex für den L2-Mapper **106b**, welcher den 16 MSBs des 40-Bit-Schlüssels **210** entspricht. Infolgedessen weist der L1-Mapper 64 KB möglicher Stellen, eine für jeden 64-KB-Knoten auf der ersten Mapper-Stufe **112a** (Fig. 2B), auf. Die Eingabedaten **220a** des L1-Mappers, welche an der entsprechenden Stelle im L1-Mapper **106a** gespeichert werden, werden an ein Fließband **208** und den indirekten L2-Mapper **106b** weitergeleitet. Wenn die Eingabedaten **220a** des L1-Mappers anzeigen, daß ein Absuchen der nächsten Stufe unter Verwendung der nächsten acht Bits des Schlüssels **210b** erforderlich ist, wird eine

Suche im indirekten L2-Mapper **106b** in Abhängigkeit von den nächsten acht Bits des Schlüssels **210b** und der Eingabedaten **220a** des L1-Mappers durchgeführt.

[0066] Das Ergebnis der Suche auf der zweiten Stufe wird auf den Eingabedaten **220b** des L2-Mappers bereitgestellt, welche an das Fließband **208** und den indirekten L3-Mapper **106c** weitergeleitet werden. Eine Suche auf der dritten Stufe wird im indirekten L3-Mapper **106c** in Abhängigkeit von den nächsten acht Bits des Schlüssels **210c** und den Eingabedaten **220b** des L2-Mappers durchgeführt.

[0067] Das Ergebnis der Suche wird auf den Eingabedaten **220c** des L3-Mappers bereitgestellt, welche an das Fließband **208** und den indirekten L4-Mapper **106d** weitergeleitet werden. Die Eingabedaten **220c** des L3-Mappers bestimmen, ob eine vierte Suche im indirekten L4-Mapper **106d** in Abhängigkeit von den letzten acht Bits des Schlüssels **210d** und den Eingabedaten **220c** des L3-Mappers durchgeführt wird.

[0068] Das Ergebnis der vierten Suche wird auf den Eingabedaten **220d** des L4-Mappers bereitgestellt. Der Routenindex **102**, welcher mit dem Präfix längster Übereinstimmung für den Schlüssel **210** verbunden ist, wird nur an einer Stelle in einem der Mapper **106a-d** gespeichert. Infolgedessen ist der Routenindex **102**, welcher an das Fließband **208** weitergeleitet, nur in einer der Mapper-Eingabedaten **220a-d** enthalten. Wenn der Routenindex **102** in einem der Mapper **106a-d**, zum Beispiel Mapper **106b**, gefunden wird, ist ein Absuchen der restlichen Mapper **106c-d** nicht notwendig, und es wird nicht auf die Mapper **106c-d** zugegriffen. Das Fließband **208** umfaßt einen Multiplexer **108** (**Fig. 2a**) zum Auswählen des Routenindexes **102**, welcher in einer der Mapper-Eingabedaten **220a-d** enthalten ist. Zum Beispiel kann das MSB der Mapper-Eingabedaten **220a-d** eine Anzeige, ob ein Routenindex enthalten ist, bereitstellen.

[0069] Durch Verwenden eines Fließbandes **208** in Verbindung mit den Mappern **106a-d** kann mehrfaches Absuchen der Präfixtabelle **20** längster Übereinstimmung mit verschiedenen Schlüsseln **210** parallel durchgeführt werden. Das Fließband **208** erlaubt es, daß mehrfaches Absuchen der 40-Bit-Nachschlagetabelle **200** parallel stattfindet, indem die Mapper-Eingabedaten **220a-d** für jeden Mapper **106a-d**, welcher mit dem 40-Bit-Schlüssel verbunden ist, gespeichert werden, bis ein Absuchen jedes der anderen Mapper **106a-d** abgeschlossen ist, falls notwendig, um den Routenindex zu finden, welcher dem 40-Bit-Schlüssel **210** entspricht. Infolgedessen wird eine Suchanforderung nach einem Routenindex, welcher einer empfangenen IP-Adresse entspricht, durch Durchführen eines einzelnen Speicherzugriffs auf den direkt abgebildeten Mapper **106a** an die Nachschlagetabelle **200** ausgegeben. Eine anschließende Suche nach einem Routenindex, welcher einem anderen Schlüssel entspricht, kann an die Nachschlagetabelle **200** im nächsten Speicherzugriffzyklus für den direkt abgebildeten Mapper **106a** ausgegeben werden.

[0070] **Fig. 4** veranschaulicht die Typen von Mapper-Eingaben, welche in den direkt abgebildeten Mapper **106a**, welcher in **Fig. 3** dargestellt ist, gespeichert werden können. Eine Mapper-Eingabe für irgendeinen Knoten im Binärbaum, welcher in **Fig. 2B** dargestellt ist, kann eine Leereingabe **300**, eine Routeneingabe **302** oder einen Teilbaumeingaben-Deskriptor **304** speichern. Jeder Typ von Mapper-Eingabe **300, 302, 304** umfaßt ein Teilbaum-Flag **306**. Der Zustand des Teilbaum-Flags **306** zeigt an, ob die Mapper-Eingabe ein Teilbaumeingaben-Deskriptor **304** ist. Wenn das Teilbaum-Flag **306** auf '1' gesetzt wird, ist die Mapper-Eingabe ein Teilbaumeingaben-Deskriptor **304** und enthält einen Teilbaumindex **312**. Der Teilbaumindex **312** ist die Adresse des Teilbaumeingaben-Deskriptors **304**, welcher im nächsten nicht direkt abgebildeten Mapper **106b-d** gespeichert wird. Die Teilbaumeingabe wird später in Verbindung mit **Fig. 4** beschrieben. Wenn das Teilbaum-Flag **306** '0' ist, wird das Leereingabe-Flag **314** überprüft, um festzustellen, ob die Mapper-Eingabe eine Leereingabe **300** oder eine Routeneingabe **302** ist. Wenn das Leereingabe-Flag **314** '0' ist, ist die Eingabe eine Leereingabe **300**. Wenn das Leereingabe-Flag **314** '1' ist, ist die Eingabe eine Routeneingabe **302** und speichert den Routenindex **102** (**Fig. 3**), welcher mit dem Schlüssel **104** im Routenindexfeld **310** verbunden ist. Das Teilbaum-Flag **306** wird durch den Multiplexer **108** (**Fig. 2A**) verwendet, um die Mapper-Eingabedaten **220a-d**, welche den Routenindex **102** (**Fig. 3**) enthalten, auszuwählen.

[0071] **Fig. 5** veranschaulicht den Mapper **106b**, welcher den Knoten auf der Mapper-Stufe_2 **112b**, welche in **Fig. 2B** dargestellt sind, entspricht. Der Mapper **106b** umfaßt einen Teilbaumspeicher **400**, eine Mapper-Adreßlogik **402** und einen Teilbaum-Mapper **418**. Der Teilbaumindex **312**, welcher durch den ersten Abschnitt des Schlüssels **210a**, welcher im Mapper **106a** gespeichert wird, ausgewählt wird, wird an den Teilbaumspeicher **400** weitergeleitet. Der Teilbaumspeicher **400** enthält eine Teilbaumeingabe **404**, welche durch den Teilbaumindex **312** ausgewählt wird. Die Teilbaumeingabe **404** umfaßt ein Datenfeld **406** und ein Zeigerfeld **408**.

[0072] Rückblickend auf **Fig. 2B** entspricht die Teilbaumeingabe **404** der unteren Stufe eines der Teilbäume, welche auf der Mapper-Stufe_2 **112b** dargestellt sind. Wenn die Mapper-Stufe_2 **112b** acht Stufen aufweist, weist die untere Stufe jedes Teilbaumes (nicht dargestellt) ein Maximum von 256 Routen, eine für jeden Knoten, auf.

[0073] Um wieder mit **Fig. 5** fortzufahren, stellt die Teilbaumeingabe **404** Zugriff auf 256 mögliche Routenindizes **102** (**Fig. 3**) bereit, welche jedem Knoten auf der unteren Stufe des Teilbaumes entsprechen. Die Routenindizes **102** (**Fig. 3**) werden in dem Teilbaum-Mapper **418** gespeichert. Um Zugriff auf 256 mögliche Routenindizes bereitzustellen, wird ein dichtbesiedelter Teilbaum-Deskriptor in dem Datenfeld **406** gespeichert.

Das Datenfeld **406** ist 256 Bits breit und stellt ein Bit für jeden Knoten auf der unteren Stufe des Teilbaumes bereit. Das Datenfeld **406** wird später in Verbindung mit **Fig. 6A** und **Fig. 6B** ausführlicher beschrieben. Das Zeigerfeld **408** ist 256 Bits breit, um die Speicherung von sechzehn 16-Bit-Zeigern zu berücksichtigen, wobei jeder Zeiger die Basisadresse für 16 benachbarte Mapper-Eingaben im Teilbaum-Mapper **418** speichert, um Zugriff auf 256 Routenindizes bereitzustellen. Infolgedessen kann das Zeigerfeld **408** indirekt einen Zeiger zu einer Mapper-Eingabe im Teilbaum-Mapper **418** für jeden Knoten auf der unteren Stufe des Teilbaumes bereitstellen. Das Zeigerfeld **408** wird in Verbindung mit **Fig. 6** ausführlicher beschrieben.

[0074] Die Teilbaumdaten **412**, welche im dichtbesiedelten Teilbaum-Deskriptor im Datenfeld **406** gespeichert werden, und der Teilbaumzeiger **414**, welcher im Zeigerfeld **408** gespeichert wird, werden an die Mapper-Adreßlogik **402** weitergeleitet. Die Mapper-Adreßlogik **402** empfängt auch den nächsten Abschnitt des Schlüssels **210b** (die nächsten acht Bits).

[0075] Die Mapper-Adreßlogik **402** bestimmt die Mapper-Adresse **416** der Mapper-Eingabe, welche mit dem Knoten auf der unteren Stufe des Teilbaumes verbinden ist, in Abhängigkeit von den nächsten acht Bits des Schlüssels **212b**, den Teilbaumdaten **412** und den Teilbaumzeigern **414**, welche mit dem Teilbaum verbunden sind. Die Mapper-Adresse **416** wählt die Mapper-Eingabe im Teilbaum-Mapper **418** aus. Der Teilbaum-Mapper **418** beinhaltet dieselben Typen von Mapper-Eingaben, wie in Verbindung mit **Fig. 4** für den direkt abgebildeten Mapper **106a** beschrieben. Der Inhalt der Mapper-Dateneingabe **220b** bestimmt, ob eine nachfolgende Suche notwendig ist. Eine nachfolgende Suche ist erforderlich, wenn die Mapper-Eingabedaten **220b** einen Teilbaumindex **312** (**Fig. 4**) enthalten, welcher anzeigen, daß es eine andere Teilbaumeingabe **404** auf der nächsten Mapper-Stufe **112c** (**Fig. 2B**) gibt.

[0076] Der zweite Abschnitt des Schlüssels **210b** wählt den Knoten auf der unteren Stufe des ausgewählten Teilbaumes aus. Der Teilbaumzeiger **414** wählt die Basisadresse aus, welche mit dem Knoten in dem Teilbaum verbunden ist, und die Teilbaumdaten **412** wählen das Offset innerhalb des Blocks von Mapper-Eingaben, welche mit der Basisadresse verbunden sind. Die Mapper-Adreßlogik **401** wird später in Verbindung mit **Fig. 7** beschrieben.

[0077] **Fig. 6A** ist eine Binärbaumdarstellung eines Teilbaumes. Der dargestellte Teilbaum umfaßt 5 Stufen. Der Teilbaum umfaßt drei Routenindizes **r1**, **r2** und **r3** und zwei Teilbaumindizes **s0** und **s1**. Es gibt 32 Knoten **500¹–500³²** auf der unteren Stufe des Teilbaumes. Der Routenindex oder Teilbaumindex, welcher mit jedem Knoten **500¹–500³²** auf der unteren Stufe verbunden ist, ist in der folgenden Tabelle dargestellt.

Teilbaum-Bits	Route/Teilbaum
00000	r0
00001	r0
00010	r0
00011	r0
00100	r1
00101	r1
00110	r0
00111	r0
01000	r2
01001	s0
01010	r2
01011	r2
01100	r2
01110	r2
01111	r2
1xxxx	r3

Tabelle 1

[0078] **Fig. 6B** veranschaulicht den dichtbesiedelten Teilbaum-Deskriptor, welcher im Datenfeld **406** gespeichert wird, das in **Fig. 5** dargestellt ist und den Knoten auf der unteren Stufe des Teilbaumes, welcher in

Fig. 6A dargestellt sind, entspricht. Das Datenfeld **406** enthält 32 Bits, ein Bit für jeden Knoten **500** auf der unteren Stufe des Teilbaumes, welcher in **Fig. 6A** dargestellt ist. Die Bits **500¹–500³²** in dem Datenfeld **406** werden folgendemnäßen zugeteilt: ein Bit im Datenfeld **406** wird auf '0' gesetzt, wenn der Routenindex für den vorhergehenden Knoten zu verwenden ist, und wird auf '1' gesetzt, um die nächste Mapper-Adresseingabe zu erhöhen, wenn der nächste Routenindex, welcher im Teilbaum-Mapper **418** (**Fig. 5**) gespeichert ist, zu verwenden ist. Das erste Bit im Datenfeld **402** wählt die Standardroute r0, welche in der Mapper-Eingabe **504¹** gespeichert wird, sofern die Route nicht spezifiziert wird. Infolgedessen wird das Bit **502¹** auf '0' gesetzt, um die Standardroute auszuwählen, da keine Route spezifiziert ist. Die Standardroute r0, welche in der Mapper-Eingabe **504¹** gespeichert wird, wird für die nächsten drei Knoten **500²–500⁴** ausgewählt, weshalb die entsprechenden Bits **502¹–502⁴** im Datenfeld **406** auf '0' gesetzt werden, um den vorhergehenden Routenindex zu verwenden, welcher durch **502¹** verwendet wurde. Es erfolgt eine Routenänderung bei Knoten **500⁵**.

[0079] Die Route r1, die für den Knoten **500⁵** verwendet wird, welcher in der Mapper-Eingabe **504¹** gespeichert wird, wird auch durch den Knoten **500⁶** benutzt. Daher ist das Bit **502⁵** '1' und zeigt eine Routenänderung an, um die Mapper-Eingabe **504²** im Teilbaum-Mapper **418** (**Fig. 5**) auszuwählen. Das Bit **502⁶** ist '0' und zeigt an, daß der Routenindex, welcher in **502⁵** gespeichert wird, für diesen Knoten zu verwenden ist. Keine Route wird für den Knoten **500** bereitgestellt, weshalb eine Routenänderung erfolgt und ein '1' in Bit **502¹** gespeichert wird, wodurch eine Mapper-Eingabe **504³** im Teilbaum-Mapper **418** (**Fig. 5**) notwendig ist, welche die Standardroute r0 speichert.

[0080] Der Knoten **500¹⁰** weist eine Route auf welche sich vom vorhergehenden Knoten **500⁹** unterscheidet, eine neue Routeneingabe ist im Teilbaum-Mapper **418** (**Fig. 5**) erforderlich. Das Bit **502¹⁰**, welches dem Knoten **500¹⁰** entspricht, wird auf '1' gesetzt und die Mapper-Eingabe **504⁵**, welche s0 speichert, wird an der nächsten benachbarten Speicherstelle zum Teilbaum-Mapper **418** (**Fig. 5**) addiert.

[0081] Der Knoten **500¹¹** weist eine Route auf welche sich vom vorhergehenden Knoten **500¹⁰** unterscheidet, eine neue Mapper-Eingabe ist im Teilbaum-Mapper **418** (**Fig. 5**) erforderlich. Das Bit **502¹¹**, welches dem Knoten **500¹¹** entspricht, wird auf '1' gesetzt und die Mapper-Eingabe **504⁶**, welche r2 speichert, wird an der nächsten benachbarten Speicherstelle zum Teilbaum-Mapper **418** (**Fig. 5**) addiert.

[0082] Die Knoten **500¹²** und **500¹³** benutzen dieselbe Route wie der vorhergehende Knoten **500¹¹**, eine neue Mapper-Eingabe ist im Teilbaum-Mapper **418** (**Fig. 5**) nicht erforderlich. Das Bit **502¹²**, welches dem Knoten **500¹²** entspricht, und das Bit **502¹³**, welches dem Knoten **500¹³** entspricht, werden im Datenfeld **406** auf '0' gesetzt.

[0083] Der Knoten **500¹⁴** weist eine Route auf, welche sich vom vorhergehenden Knoten **500¹³** unterscheidet, eine neue Mapper-Eingabe ist im Teilbaum-Mapper **418** (**Fig. 5**) erforderlich. Das Bit **502¹⁴** im Datenfeld **406**, welches dem Knoten **500¹⁴** entspricht, wird auf '1' gesetzt und die Mapper-Eingabe **504¹**, welche s1 speichert, wird zum Teilbaum-Mapper **418** (**Fig. 5**) addiert. Der Knoten **500¹⁵** weist eine Route auf, welche sich vom vorhergehenden Knoten **500¹⁴** unterscheidet, eine neue Mapper-Eingabe ist im Teilbaum-Mapper **418** (**Fig. 5**) erforderlich. Das Bit **502¹⁵** im Datenfeld **406**, welches dem Knoten **500¹⁵** entspricht, wird auf '1' gesetzt und die Mapper-Eingabe **504⁸**, welche r2 speichert, wird zum Teilbaum-Mapper **418** (**Fig. 5**) addiert. Der Knoten **500¹⁶** benutzt dieselbe Route wie der vorhergehende Knoten **500¹⁵**, eine neue Mapper-Eingabe im Teilbaum-Mapper **418** (**Fig. 5**) ist nicht erforderlich.

[0084] Der Knoten **500¹⁷** weist eine Route auf, welche sich vom vorhergehenden Knoten **500¹⁶** unterscheidet, eine neue Mapper-Eingabe ist im Teilbaum-Mapper **418** (**Fig. 5**) erforderlich. Das Bit **502¹⁷** im Datenfeld **406**, welches dem Knoten **500¹⁷** entspricht, wird auf '1' gesetzt und die Mapper-Eingabe **504⁹**, welche r3 speichert, wird zum Teilbaum-Mapper **418** (**Fig. 5**) addiert.

[0085] Die Knoten **500¹⁸–500³²** benutzen alle dieselbe Route wie der Knoten **500¹⁷**, eine neue Mapper-Eingabe ist im Teilbaum-Mapper **418** (**Fig. 5**) nicht erforderlich. Die entsprechenden Bits **502¹⁸–502³²** werden auf '0' gesetzt. Infolgedessen sind neun Mapper-Eingaben **504^{1–9}** notwendig, um die Routeneingaben **302** (**Fig. 4**) oder die Teilbaumeingaben **304** (**Fig. 4**) für die 32 Knoten **500¹–500³²** zu speichern.

[0086] Eine Mapper-Eingabe **504¹–504⁹**, welche einem Knoten **500¹–500³²** entspricht, wird im Teilbaum-Mapper **418** (**Fig. 5**) indexiert durch Berechnen der Anzahl von '1', die im dichtbesiedelten Teilbaum-Deskriptor gespeichert werden, welcher im Datenfeld **406** gespeichert wird. Um zum Beispiel die Mapper-Eingabe **504¹–504⁹**, welche dem Knoten **500²⁸** entspricht, zu finden, wird die Anzahl von '1', welche in den Bits **502–502²⁸** des Datenfeldes **406** gespeichert werden, gezählt. Die Anzahl von '1' beträgt 8 und die entsprechende Mapper-Eingabe ist die achte Stelle von der Standardroute aus; d.h., Mapper-Eingabe **504⁹**.

[0087] Mapper-Eingaben nur zu speichern, wenn eine Routenänderung erfolgt, verringert die Anzahl von Mapper-Eingaben **504¹–504⁹** pro Teilbaum im Teilbaum-Mapper **418** (**Fig. 5**).

[0088] **Fig. 7** veranschaulicht das Zeigerfeld **408**, welches in **Fig. 5** dargestellt ist. Das Zeigerfeld **408** umfaßt die Blockbasisadrefsfelder **600¹**, **600²** zum Speichern von Basisadressen für Blöcke von 16 benachbarten Mapper-Eingabestellen **504¹–504¹⁶** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**). Speicher wird im Teilbaum-Mapper **418** (**Fig. 5**) in Blöcken **602¹**, **602²** von 16 benachbarten Mapper-Eingaben zugeordnet. Ein 8-stufiger Teilbaum kann bis zu 256 verschiedene Routen aufweisen, welche 16 Blöcke **602¹**, **602²** erfordern, um alle 256 Routen

zu speichern. Die Anzahl von Blöcken **602**, welche benötigt werden, hängt von der Anzahl von Routen für den Teilbaum ab. Ein Block **602** wird einem konkreten Teilbaum durch Entfernen der Blockbasisadresse **602¹**, **602²** von einer freien Liste von Blockbasisadressen (nicht dargestellt) zugeordnet. Verfahren zur Bereitstellung einer freien Liste von Adressen für einen Speicher sind in der Technik wohlbekannt.

[0089] Durch Zuordnen von Speicherblöcken von 16 Mapper-Eingaben **504¹⁻¹⁶** ist der Speicher im Teilbaum-Mapper **418** (**Fig. 5**) leichter zu verwalten, da die zugeordneten 16 Stellen benachbart sind.

[0090] **Fig. 8** veranschaulicht die Mapper-Adreßlogik **402**, welche in **Fig. 5** dargestellt ist. Die Mapper-Adreßlogik **402** umfaßt die Offset-Logik **700**, die Zeiger-Logik **702** und die Addierer-Logik **704**.

[0091] Die Offset-Logik **700** umfaßt die Knotenauswahl-Logik **706** und Einserzähl-Logik **708**. Die Zeiger-Logik umfaßt die Basisadreßauswahl-Logik **710**.

[0092] Die Knotenauswahl-Logik **706** wählt den Knoten **500** (**Fig. 6B**) auf den Teilbaumdaten **412**, welche den acht Bits des Schlüssels **210b** entsprechen, aus. Die entsprechende Knotenzahl wird auf der Knotenauswahl **718** an die Einserzähl-Logik **708** weitergeleitet. Die Einserzähl-Logik **708** zählt die Anzahl von '1', welche im Teilbaum-Datenfeld **406** gespeichert werden, bis zu dem Bit, welches dem ausgewählten Knoten **500** entspricht. Die Anzahl von Einsen wird auf der Blockauswahl **712** an die Zeiger-Logik **702** und Block-Offset **714** an die Addierer-Logik **704** weitergeleitet.

[0093] Es kann ein Maximum von 256 '1' in einem Teilbaum-Datenfeld **406** von 256 Bits gespeichert werden, was ein 8 Bit breites Zählfeld erfordert. Das 8-Bit-Zählfeld wird in zwei Felder unterteilt, wobei die 4 MSBs die Blockauswahl **712** bereitstellen und die 4 niedrigstwertigen Bits ("LSBs", nach Engl. Least Significant Bits) das Block-Offset **714** bereitstellen.

[0094] Wenn zum Beispiel der Acht-Bit-Schlüssel **210b** '0100 0100' ist, um die Knotenzahl **68** zu wählen, und 27 '1' in den ersten 68 Bits der Teilbaumeingaben **412** gespeichert werden, ist die Zählung 1C Hex (0001 1100), die MSBs (0001); das heißt, Blockauswahl **714**, wählen Block **602¹** (**Fig. 6**) und die LSBs (1100), das heißt, das Basisblock-Offset, wählen die Mapper-Eingabe **504¹¹** (**Fig. 6**); das heißt, die zwölft Eingabe in Block **502¹**.

[0095] Die Basisadreßauswahl-Logik **710** wählt die Basisadresse **716** vom Teilbaumzeiger **414** in Abhängigkeit von der Blockauswahl **712**, welche von der Offset-Logik **700** weitergeleitet wird, aus. Die Addierer-Logik **704** addiert das Block-Offset **714**, welches von der Offset-Logik **700** weitergeleitet wurde, zur Basisadresse **716** und stellt die Mapper-Adresse **416** bereit. Die Mapper-Adresse **416** ist der Index der Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106b-d**.

[0096] **Fig. 9** ist ein Ablaufdiagramm, welches die Schritte zur Suche nach dem Präfix längster Übereinstimmung für einen Schlüssel **210** (**Fig. 3**) in der Nachschlagetabelle **200** gemäß den Prinzipien der vorliegenden Erfindung veranschaulicht.

[0097] Bei Schritt **800** wird der erste Abschnitt des Schlüssels **210a** (**Fig. 3**) als der Index an den Mapper **106a** weitergeleitet. Die Verarbeitung wird bei Schritt **802** fortgesetzt.

[0098] Bei Schritt **802** bestimmen die Mapper-Eingabedaten **220a** (**Fig. 3**), welche in der Mapper-Eingabe **504** (**Fig. 6B**) im Mapper der ersten Stufe gespeichert werden, der durch den ersten Abschnitt des Schlüssels **210a** (**Fig. 3**) indexiert wird, ob ein weiteres Absuchen der nächsten Stufe notwendig ist. Wenn dem so ist, wird die Verarbeitung bei Schritt **804** fortgesetzt. Wenn dem nicht so ist, speichert die Routeneingabe **302** (**Fig. 4**) in der indexierten Mapper-Eingabe **504** (**Fig. 6B**) im Mapper der ersten Stufe die entsprechende längste Präfixroute für den Schlüssel und die Verarbeitung wird bei Schritt **808** fortgesetzt.

[0099] Bei Schritt **804** wird der Mapper **106b-d** der nächsten Stufe abgesucht. Der Index für den Mapper der nächsten Stufe hängt vom Teilbaumindex **312**, welcher im Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) in der indexierten Mapper-Eingabe **504** (**Fig. 6B**) im Mapper der vorhergehenden Stufe gespeichert wird, und dem nächsten Abschnitt des Schlüssels **210b -d** ab. Die Verarbeitung wird bei Schritt **806** fortgesetzt.

[0100] Bei Schritt **806** speichert die indexierte Mapper-Eingabe **504** (**Fig. 6B**) im Mapper der nächsten Stufe den entsprechenden längsten Präfixroutenindex für den Schlüssel oder einen Teilbaumindex, welcher anzeigt, daß eine weitere Suche notwendig ist. Wenn eine weitere Suche erforderlich ist, wird die Verarbeitung bei Schritt **804** fortgesetzt. Wenn dem nicht so ist, wird die Verarbeitung bei Schritt **808** fortgesetzt.

[0101] Bei Schritt **808** wird der Routenindex **102** (**Fig. 3**), welcher in einer Mapper-Eingabe **504** (**Fig. 6B**) in einem der Mapper **106a-d** gespeichert wird, von der Nachschlagetabelle **200** als der Routenindex **102** (**Fig. 3**) weitergeleitet. Die Verarbeitung ist abgeschlossen.

Tiefenerweiterung

[0102] Die Anzahl von Routenindizes **102** (**Fig. 3**), welche in der Nachschlagetabelle **200**, welche in **Fig. 3** dargestellt ist, gespeichert werden können, wird durch die Anzahl von verfügbaren Mapper-Eingaben **504** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**) eingeschränkt. Wenn zum Beispiel jeder Teilbaum-Mapper **418** (**Fig. 5**) 128-KB-Mapper-Eingaben enthält und es zwei Teilbaum-Mapper **418** (**Fig. 5**) in der Nachschlagetabelle gibt, kann ein Maximum von 256-KB-Routenindizes **102** (**Fig. 3**) in der Nachschlagetabelle **200** gespeichert werden. Ein Teilbaum-Mapper **418** (**Fig. 5**) mit 128-KB-Mapper-Eingaben erfordert einen 17-Bit-Index. Ein Teil-

baum-Mapper **418** (**Fig. 5**) mit 512-KB-Mapper-Eingaben erfordert einen 19-Bit-Index. Zwei 512-KB-Teilbaum-Mapper **418** (**Fig. 5**) in der Nachschlagetabelle **200** stellen Speicherung für 1 Million von möglichen 4 Milliarden Routenindizes **102** (**Fig. 3**) für eine 32-Bit-IPv4-Zieladresse bereit.

[0103] Die Anzahl von Mapper-Eingaben **504** (**Fig. 6B**) zum Speichern von Routenindizes **102** (**Fig. 3**) kann durch die Bereitstellung einer Mehrzahl von Nachschlagetabellen **200** erhöht werden. Die Mehrzahl von Nachschlagetabellen werden parallel nach dem Wert abgesucht, welcher einem Suchschlüssel **210** entspricht, der in einer Mapper-Eingabe **504** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**) in einer der Nachschlagetabellen **200** gespeichert wird.

[0104] **Fig. 10A** veranschaulicht ein Ausführungsbeispiel für Tiefenerweiterung. Zwei Nachschlagetabellen sind dargestellt, und zwar eine Hauptnachschlagetabelle **200a** und eine Nebennachschlagetabelle **200b**. Die Anzahl von Nachschlagetabellen ist jedoch nicht auf die beiden dargestellten beschränkt, sondern es können mehr als eine Nebennachschlagetabelle **200b** addiert werden.

[0105] Jede der Nachschlagetabellen **200a–b** wird mit demselben Suchschlüssel **210** parallel abgesucht. Der Routenindex **102** (**Fig. 3**), welcher dem Suchschlüssel **210** entspricht, wird in einem Teilbaum-Mapper **418** (**Fig. 5**) in einer der Nachschlagetabellen **200a–b** oder im direkten Mapper **106a** in beiden Nachschlagetabellen **200a**, **200b** gespeichert. Der Endroutenindex **900** wird nach dem parallelen Absuchen der beiden Nachschlagetabellen **200a**, **200b** gefunden.

[0106] **Fig. 10B** veranschaulicht eine der Nachschlagetabellen **200a** in der Ausführungsform, welche in **Fig. 10A** dargestellt ist. Jede der Nachschlagetabellen **200a–b** umfaßt Mapper **106a–d** und ein Fließband **208**, wie bereits für die Nachschlagetabelle **200** in Verbindung mit **Fig. 3** beschrieben wurde, und eine Treiber-Logik **902**. Die Nachschlagetabelle **200a** führt eine Mehrstufensuche in den Mappern **106a–d** nach einem Routenindex, welcher dem Suchschlüssel entspricht, durch. Das Ergebnis jeder Stufensuche wird auf den Mapper-Eingabedaten **220a–b** an das Fließband **208** weitergeleitet. Das Fließband **208** leitet das Ergebnis der Suche **904** an die Treiber-Logik **902** weiter. Die Mapper-Eingaben **504** (**Fig. 6B**) im Mapper **106a** in jeder der Nachschlagetabellen **200a–b** speichern die Teilbaumeingaben **304** (**Fig. 4**), aber die Routeneingaben **302** (**Fig. 4**) werden nur im Mapper **106a** in der Hauptnachschlagetabelle **200a** gespeichert. Die Leereingabe **300** (**Fig. 4**) wird in den Mapper-Eingaben **504** (**Fig. 6B**) im Mapper **106a** in der Nebennachschlagetabelle **200b** anstelle der Routeneingaben **302** gespeichert. Nur eine Nachschlagetabelle mit den Routenindizes im Mapper **106a** zu bestücken, vermeidet das auswählen einer Nachschlagetabelle, um den Endroutenindex **900** bereitzustellen. Dies führt zu 64 KB von Speicher in der Nebennachschlagetabelle **200b**, welcher zwar nicht verwendet werden kann, um Routenindizes zu speichern, es aber ermöglicht, daß dieselbe Nachschlagetabelle als eine Hauptnachschlagetabelle oder eine Nebennachschlagetabelle konfiguriert wird, wie in Verbindung mit **Fig. 3** beschrieben. In einem alternativen Ausführungsbeispiel kann ein Nebennachschlagegerät ohne den Mapper **106a** bereitgestellt werden. Eine Suche endet im Mapper **106a** in der Hauptnachschlagetabelle **200a**, wenn ein Routenindex **102** (**Fig. 3**) in einer Routeneingabe **302** (**Fig. 4**) im Mapper **106a** gespeichert wird.

[0107] Wie in **Fig. 10A** dargestellt, teilen sich die Hauptnachschlagetabelle **200a** und die Nebennachschlagetabelle **200b** den Endroutenindex **900**. Die Nachschlagetabelle **200a**, **200b**, in welcher der Endroutenindex **900** gespeichert wird, stellt den Routenindex **102** (**Fig. 3**) bereit. Wenn jede der Nachschlagetabellen **200a**, **200b** ein getrenntes Gerät ist, verringert das Teilen des Endroutenindexes **900** die Summe der externen Anschlüsse jedes Geräts. Nur eine der Nachschlagetabellen **200a,b** treibt den Endroutenindex **900** zu jeder Zeit an.

[0108] Um einen Fehlerzustand zu vermeiden, bei welchem der Routenindex, welcher mit dem Suchschlüssel **210** verbunden ist, in beiden Nachschlagetabellen **200a**, **200b** gespeichert wird, wobei beide Nachschlagetabellen den Endroutenindex **900** gleichzeitig antreiben würden, speichert jede Nachschlagetabelle **200a,b** einen Gerätecode **906**. Ein 3-Bit-Gerätecode erlaubt der erweiterten Nachschlagetabelle, 8 Geräte zu umfassen.

[0109] Die Treiber-Logik **902** bestimmt, ob das Ergebnis der Suche **904** einen Routenindex **102** (**Fig. 3**) enthält. Wenn dem so ist, signalisiert die Treiber-Logik **902** in der Nachschlagetabelle **200a** einen Versuch, den Endroutenindex **900** auf einem Busanforderungssignal (nicht dargestellt) anzutreiben. Wenn zwei oder mehr Nachschlagetabellen **200a**, **200b** den Versuch signalisieren, die Routenindexsignale gleichzeitig anzutreiben, wird der Routenindex durch die Nachschlagetabelle **200a**, **200b** mit dem niedrigsten Gerätecode bereitgestellt. Verfahren zum Lösen von Buskonflikten durch die Verwendung eines Busanforderungssignals sind in der Technik wohlbekannt.

[0110] **Fig. 10C** veranschaulicht ein anderes Ausführungsbeispiel zur Bereitstellung von Tiefenerweiterung, um die Anzahl von Mapper-Eingaben zu erhöhen, welche zum Speichern eines Wertes verfügbar sind, der einem Suchschlüssel **210** entspricht. In dem Ausführungsbeispiel, welches in **Fig. 10C** dargestellt ist, werden zwei Nachschlagetabellen **200a–b** zum Speichern von Werten bereitgestellt, und zwar eine Hauptnachschlagetabelle **200c** und eine Nebennachschlagetabelle **200d**. Die Anzahl von Nachschlagetabellen ist jedoch nicht auf die beiden dargestellten beschränkt, die Anzahl von Mapper-Eingaben kann durch Addieren von mehr Nebennachschlagetabellen **200d** erhöht werden. Eine Suche nach dem Wert, welcher in einer Mapper-Eingabe in einer der Nachschlagetabellen **200c–b** gespeichert wird und dem Suchschlüssel [39:0] **210** entspricht, wird in den Nachschlagetabellen **200c–b** parallel durchgeführt.

[0111] **Fig. 10D** veranschaulicht die Nebennachschlagetabelle **200d** in der Ausführungsform, welche in **Fig. 10C** dargestellt ist. Jede Nachschlagetabelle umfaßt Mapper **106a–b**, wie für die Nachschlagetabelle **200** in Verbindung mit **Fig. 3** beschrieben. Mapper-Eingaben im Mapper **106a** in jeder der Nachschlagetabellen **200c–b** speichern Teilbaumeingaben **304** (**Fig. 4**). Jede Nachschlagetabelle **200c–b** leitet einen Teilbaumindex **312**, welcher in einem Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) gespeichert wird, der in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** gespeichert wird, an den nächsten Mapper **106b–b** weiter. Ein Routenindex **102** (**Fig. 3**) wird jedoch nur im Mapper **106a** in einer Hauptnachschlagetabelle **200c** gespeichert. Eine Leereingabe wird im Mapper **106a** in einer Nebennachschlagetabelle **200d** gespeichert, um das Speichern eines Routenindexes, welcher dem Schlüssel entspricht, in mehr als einer Nachschlagetabelle **200b**, **200d** zu vermeiden.

[0112] Das Ergebnis der Mehrstufensuche **904** in der Nebennachschlagetabelle **200d** wird an die Endindex-Logik **1004** weitergeleitet. Die Endindex-Logik **1004** leitet das Ergebnis der Mehrstufensuche **904** oder das ankommende Ergebnis **1000a**, welches von der Hauptnachschlagetabelle **200c** weitergeleitet wird, als das abgehende Ergebnis **1002a** weiter. Wenn ein Routenindex **102** (**Fig. 3**) im Ergebnis der Mehrstufensuche **904** enthalten ist, wird das Ergebnis der Mehrstufensuche als das abgehende Ergebnis **1002a** weitergeleitet. Wenn der Routenindex **102** (**Fig. 3**) in dem ankommenden Ergebnis **1000a** enthalten ist, wird das ankommende Ergebnis **1000a** als das abgehende Ergebnis **1002a** weitergeleitet. Wenn der Routenindex **102** (**Fig. 3**) weder im ankommenden Ergebnis **1000a** noch dem Ergebnis der Mehrstufensuche **904** enthalten ist, wird das Ergebnis der Mehrstufensuche **904** als das abgehende Ergebnis **1002a** weitergeleitet.

[0113] Wie in **Fig. 10C** dargestellt, sind die Hauptnachschlagetabelle **200c** und die Nebennachschlagetabelle **200d** durch einen gemeinsamen Bus, welcher als ankommendes Ergebnis **1000a** gekennzeichnet ist, verbunden. Der Routenindex **102** (**Fig. 3**) wird von der Nebennachschlagetabelle **200d** nur auf dem abgehenden Ergebnis **1002a** weitergeleitet. Wenn es mehr als eine Nebennachschlagetabelle **200d** gibt, wird der Routenindex **102** (**Fig. 3**) für die erweiterte Nachschlagetabelle durch die letzte Nebennachschlagetabelle bereitgestellt. Diese Ausführungsform vermeidet zwar die Realisierung des Mehr-Treiber-Endroutenindexes **900**, welcher in Verbindung mit **Fig. 10A** beschrieben wurde, erfordert aber mehr externe Gerätanschlüsse für das ankommende Ergebnis **1000a**.

[0114] **Fig. 11A–B** veranschaulichen eine Binärbaumdarstellung der Verteilung der Routen, welche in **Fig. 2B** dargestellt sind, unter den Nachschlagetabellen **200a–b** (**Fig. 10A**) oder **200c–b** (**Fig. 10C**).

[0115] **Fig. 11A** veranschaulicht eine Binärbaumdarstellung der Routen, welche in der Hauptnachschlagetabelle **200a** (**Fig. 10A**) oder **200c** (**Fig. 10C**) gespeichert werden. Teilbaum B, welcher in der Binärbaumdarstellung der Routen dargestellt ist, welche in **Fig. 2B** dargestellt sind, ist in der Hauptnachschlagetabelle **200a** nicht enthalten. Die Knoten **130¹–130²²** und **130^{24–32}** werden im Mapper **106a** in der Nachschlagetabelle **200a** codiert, wie in Verbindung mit **Fig. 3** beschrieben. Der Knoten, an welchem der Teilbaum B indexiert wäre, wenn er in der Hauptnachschlagetabelle **200a** gespeichert werden würde, ist grafisch durch ein X dargestellt, welches einen ausgeästeten Teilbaum anzeigt. Die Mapper-Eingabe **504** (**Fig. 6B**), welche dem Knoten **130²³** in der Hauptnachschlagetabelle **200a** entspricht, speichert keinen Teilbaumindex **312** (**Fig. 4**) zu Teilbaum B mehr. Stattdessen wird die Leereingabe **300** (**Fig. 4**) in der Mapper-Eingabe **504** (**Fig. 6B**), welche dem Knoten **130²³** entspricht, in der Hauptnachschlagetabelle **200a** gespeichert und zeigt an, daß die Mapper-Eingabe, welche dem Knoten **130²³** entspricht, in einem Teilbaum-Mapper **418** (**Fig. 5**) in einer anderen Nebennachschlagetabelle **200b** gespeichert wird.

[0116] **Fig. 11B** veranschaulicht eine Binärbaumdarstellung der Routen, welche in den Mapper-Eingaben **504** (**Fig. 6B**) in den Teilbaum-Mappern **418** (**Fig. 5**) in der Nebennachschlagetabelle **200b** (**Fig. 10A**) oder **200d** (**Fig. 10C**) gespeichert werden. Die Binärbaumdarstellung der Routen, welche in der Nebennachschlagetabelle **200b** gespeichert sind, unterscheidet sich von der Binärbaumdarstellung, welche in **Fig. 2B** dargestellt ist, darin, daß der Teilbaum A nicht enthalten ist. Infolgedessen sind die Knoten **130¹–130³** und **130⁵–130³²** codiert, wie in Verbindung mit **Fig. 2B** beschrieben. Die Mapper-Eingabe **504** (**Fig. 6B**), welche dem Knoten **130⁴** in der Nebennachschlagetabelle **200b** entspricht, speichert keinen Teilbaumindex **312**

[0117] (**Fig. 4**) zu Teilbaum A mehr. Stattdessen speichert die Mapper-Eingabe, welche dem Knoten **130⁴** in der Nebennachschlagetabelle **200b** entspricht, die Leereingabe **300** (**Fig. 4**), welche anzeigt, daß die Mapper-Eingabe, welche dem Knoten **130⁴** entspricht, in einer anderen Nachschlagetabelle gespeichert wird. Der Teilbaumindex für den Teilbaum A und infolgedessen der Routenindex für den Host **138** (**Fig. 11A**) wird in der Hauptnachschlagetabelle **200a** gespeichert und der Teilbaumindex für den Teilbaum B und infolgedessen der Routenindex für den Host **140** wird in der Nebennachschlagetabelle **200b** gespeichert. Die Nebennachschlagetabelle **200b**, **200d** speichern nur Ergebnisse für einen Teilbaum; das heißt, eine Nebennachschlagetabelle **200b**, **200d** speichert kein Ergebnis im Mapper **106a** der ersten Stufe.

[0118] Unter Bezugnahme auf **Fig. 11A** und **Fig. 11B** führt eine Suche nach irgendeinem der Knoten **130–130¹²** auf der Hauptmapper-Stufe_1 **1102a** (**Fig. 3**) oder Nebenmapper-Stufe_1 **104a** mit einem ersten Abschnitt des Schlüssels **210a** zu **r1 116**, welcher in einer Routeneingabe **302** (**Fig. 4**) in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** in der Hauptnachschlagetabelle **200a**, **200c** gespeichert wird, und einer

Leereingabe **300** (**Fig. 4**), welche in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** in der Nebennachschatgetabelle **200b**, **200d** gespeichert wird. Die Routeneingabe **302** (**Fig. 4**), welche in der Hauptnachschatgetabelle **200a**, **200c** gespeichert wird, wird auf dem ankommenden Ergebnis **1000a** an die Nebennachschatgetabelle **200b**, **200d** weitergeleitet und durch die Nebennachschatgetabellen **200b**, **200d** auf dem abgehenden Ergebnis **1002a** weitergeleitet.

[0119] Eine Suche nach dem Knoten **130⁴** mit einem ersten Abschnitt des Schlüssels **210a** führt zu einem Teilbaumindex **312** (**Fig. 4**) für den Teilbaum A, welcher in einem Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** in der Hauptnachschatgetabelle **200a** gespeichert wird. Der Teilbaumindex **312** wird an den Mapper **106b** in der Hauptnachschatgetabelle **200a** weitergeleitet, um die Suche nach der Routeneingabe **302** (**Fig. 4**), welche in der Hauptnachschatgetabelle **200a** gespeichert wird, fortzusetzen.

[0120] Eine Suche nach dem Knoten **130²³** mit einem ersten Abschnitt des Schlüssels **210a** führt zu einer Leereingabe **300** (**Fig. 4**), welche in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** in der Hauptnachschatgetabelle **200a** gespeichert wird, und einem Teilbaumeingaben-Deskriptor **304** (**Fig. 4**), welcher in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** in der Nebennachschatgetabelle **200b** gespeichert wird. Infolgedessen geht die Suche nach der Routeneingabe **302** (**Fig. 4**) mit dem nächsten Abschnitt des Schlüssels **210b** im Mapper **106b** in der Nebennachschatgetabelle **200b** weiter.

[0121] **Fig. 12** ist ein Ablaufdiagramm, welches veranschaulicht ein Verfahren zur Verteilung von Routeneingaben **320** (**Fig. 4**), welche in Mapper-Eingaben **504** (**Fig. 6B**) zu speichern sind, unter den Nachschlagetabellen **200a–b**, welche in **Fig. 10A** dargestellt sind. Dasselbe Verfahren gilt für die Nachschlagetabellen **200c–b**, welche in **Fig. 10C** dargestellt sind. Die Routeneingaben **320** (**Fig. 4**), welche in den Mapper-Eingaben zu speichern sind, werden anfänglich durch einen Prozessor (nicht dargestellt) in einem Speicher gespeichert, bevor sie in den Nachschlagetabellen **200a–b** gespeichert werden.

[0122] Während die Routeneingaben **302** (**Fig. 4**) im Speicher gespeichert werden, wird die Anzahl der Routeneingaben **302** (**Fig. 4**), welche in jeder der Nachschlagetabellen **200a–b** (**Fig. 10A**) zu speichern sind, gezählt. Die Routeneingaben **302** (**Fig. 4**) für die Mapper-Stufe_1 **1104a** (**Fig. 11B**) werden im Mapper **106a** in der Nachschlagetabelle **200a** gespeichert. Die Teilbaumeingaben **304** (**Fig. 4**) für die Mapper-Stufe_1 **1104a** (**Fig. 11B**) werden im Mapper **106a** in jeder der Nachschlagetabellen **200a–200b** gespeichert.

[0123] Bei Schritt **1200** wird die Anzahl von Routeneingaben **302** (**Fig. 4**), welche für jede der Teilbaumeingaben **304** (**Fig. 4**) im Mapper **106a** in jeder der Nachschlagetabellen **200a–b** (**Fig. 10A**) zu speichern sind, berechnet, um zu bestimmen, wie die Routeneingaben **302** (**Fig. 4**) unter den Nachschlagetabellen **200a–b** (**Fig. 10A**) zu verteilen. Nachdem die Gesamtzahl von Mapper-Eingaben **504** (**Fig. 6B**), welche erforderlich sind, um die Routeneingaben **302** (**Fig. 4**) zu speichern, bestimmt wurde, wird die Verarbeitung bei Schritt **1202** fortgesetzt.

[0124] Bei Schritt **1202** wird die Gesamtzahl von Mapper-Eingaben **504** (**Fig. 6B**), welche für die Teilbäume zu speichern sind, durch die Anzahl von Nachschlagetabellen **200a–b** (**Fig. 10A**) geteilt, um die Anzahl von Routeneingaben **302** (**Fig. 4**) zu bestimmen, welche in jeder Nachschlagetabelle **200a–b** (**Fig. 10A**) zu speichern sind. Die Verarbeitung wird bei Schritt **1204** fortgesetzt.

[0125] Bei Schritt **1204** wird eine Routeneingabe **302** (**Fig. 4**) in einer Mapper-Eingabe **504** (**Fig. 6B**) in einem Teilbaum-Mapper **418** (**Fig. 5**) in der ausgewählten Nachschlagetabelle **200a–b** gespeichert. Die Verarbeitung wird bei Schritt **1206** fortgesetzt.

[0126] Wenn bei Schritt **1206** die Anzahl von Routeneingaben, welche in den Mapper-Eingaben **504** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**) in der ausgewählten Nachschlagetabelle **200a–b** (**Fig. 10A**) gespeichert werden, kleiner als $1/n$ ist, wobei n die Anzahl von verfügbaren Nachschlagetabellen **200a–b** (**Fig. 10A**) ist, wird die Verarbeitung bei Schritt **1204** fortgesetzt.

[0127] Wenn dem nicht so ist, speichert die ausgewählte Nachschlagetabelle **200a–b** bereits $1/n$ der Gesamtzahl von Mapper-Eingaben und wird die Verarbeitung bei Schritt **1208** fortgesetzt.

[0128] Bei Schritt **1208** speichert die ausgewählte Nachschlagetabelle **200a–b** $1/n$ der Gesamtzahl von Mapper-Eingaben und wird eine Leereingabe **300** (**Fig. 4**) für jeden restlichen Teilbaumknoten in der ausgewählten Nachschlagetabelle **200a–b** gespeichert, da Routenindizes für den jeweiligen Teilbaum nicht in der gegenwärtig ausgewählten Nachschlagetabelle gespeichert werden. Die Verarbeitung wird bei Schritt **1210** fortgesetzt.

[0129] Bei Schritt **1210** ist die Verarbeitung abgeschlossen, wenn alle Routeneingaben gespeichert wurden. Wenn dem nicht so ist, wird die Verarbeitung bei Schritt **1212** fortgesetzt.

[0130] Bei Schritt **1212** wird die nächste Nachschlagetabelle **200a–b** (**Fig. 10A**) ausgewählt. Die Verarbeitung wird bei Schritt **1204** fortgesetzt.

[0131] Die Routeneingaben werden unter den Nachschlagetabellen **200a–b** (**Fig. 10A**) verteilt, bevor nach einem Routenindex gesucht wird, welcher einer IP-Adresse entspricht. Diese Suche wird in jeder der Nachschlagetabellen **200a–b** (**Fig. 10A**) parallel durchgeführt. Das Verfahren zur Suche, welche in jeder der Nachschlagetabellen parallel realisiert wird, wird für eine der Nachschlagetabellen **200a–b** (**Fig. 10A**) beschrieben.

[0132] **Fig. 13** ist ein Ablaufdiagramm, welches ein Verfahren zur Suche mit einem Suchschlüssel nach ei-

nem Wert veranschaulicht, welcher dem Suchschlüssel entspricht, der in irgendeiner der Nachschlagetabellen **200c-d**, welche in **Fig. 10C** dargestellt sind, gespeichert wird.

[0133] Bei Schritt **1300** empfängt jede der Nachschlagetabellen **200c-d** (**Fig. 10C**) einen Suchschlüssel **210**. Der Mapper **106a** in jeder der Nachschlagetabellen **200c-d** wird nach einem Wert abgesucht, welcher dem ersten Abschnitt des Schlüssels **210a** entspricht. Die Verarbeitung wird bei Schritt **1302** fortgesetzt.

[0134] Bei Schritt **1302** wird die Eingabe gelesen, welche in der Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** gespeichert wird. Die Mapper-Eingabe **504** (**Fig. 6B**) in der Hauptnachschlagetabelle **200c** kann die Leereingabe **300** (**Fig. 4**), eine Routeneingabe **302** (**Fig. 4**) oder einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichern. Die Mapper-Eingabe **504** (**Fig. 6B**) in der Nebennachschlagetabelle **200d** kann die Leereingabe **300** (**Fig. 4**) und den Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichern. Wenn die Mapper-Eingabe in der jeweiligen Nachschlagetabelle **200** eine Routeneingabe **302** (**Fig. 4**) speichert, ist die Eingabe ein gültiger Wert, kein weiteres Absuchen der nachfolgenden Mapper **106b-d** in der Nachschlagetabelle **200c-d** ist erforderlich und die Verarbeitung wird bei Schritt **1310** fortgesetzt. Wenn dem nicht so ist, wird die Verarbeitung bei Schritt **1304** fortgesetzt.

[0135] Wenn bei Schritt **1304** die Eingabe einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichert, ist eine weitere Suche in der Nachschlagetabelle **200c-d** erforderlich und die Verarbeitung wird bei Schritt **1306** fortgesetzt. Wenn dem nicht so ist, speichert die Eingabe die Leereingabe, welche anzeigen, daß eine weitere Suche nicht notwendig ist, und die Verarbeitung wird bei Schritt **1310** fortgesetzt.

[0136] Bei Schritt **1306** wird die Suche in den ausgewählten Teilbaum fortgesetzt. Der Mapper **106b-d** (**Fig. 3**) der nächsten Stufe wird in Abhängigkeit von einem nächsten Abschnitt des Schlüssels **210b-d** und einem Teilbaumindex **312** (**Fig. 4**), welcher aus einer Suche auf der vorhergehenden Stufe resultiert, abgesucht. Die Verarbeitung wird bei Schritt **1308** fortgesetzt.

[0137] Bei Schritt **1308** wird bestimmt, ob die Suche in Abhängigkeit von der Mapper-Eingabe, welche aus der Suche im Mapper **106b-d** auf der gegenwärtigen Stufe resultiert, fortgesetzt wird. Wenn die Mapper-Eingabe **504** (**Fig. 6B**) einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichert, wird die Suche mit dem Mapper **106b-d** der nächsten Stufe fortgesetzt und wird die Verarbeitung bei Schritt **1306** fortgesetzt. Wenn die Mapper-Eingabe **504** (**Fig. 6B**) keinen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichert, ist eine weitere Suche nicht erforderlich und die Verarbeitung wird bei Schritt **1310** fortgesetzt.

[0138] Bei Schritt **1310** wird das Ergebnis der Suche mit dem ankommenden Ergebnis **1000a**, welches von einer anderen Nachschlagetabelle empfangen wird, verglichen. Wenn zum Beispiel die Nachschlagetabelle Nebennachschlagetabelle **200d** ist, wird das ankommende Ergebnis von der Suche in der Hauptnachschlagetabelle **200c** auf dem ankommenden Ergebnis **1000a** an die Nachschlagetabelle **200d** weitergeleitet und mit dem Ergebnis der Suche in der Nebennachschlagetabelle **200d** verglichen. Die Verarbeitung wird bei Schritt **1312** fortgesetzt.

[0139] Wenn bei Schritt **1312** das ankommenden Ergebnis **1000a** und das Ergebnis der Suche in der gegenwärtigen Nachschlagetabelle **200d** verschieden sind, wird die Verarbeitung bei Schritt **1314** fortgesetzt. Wenn das ankommende Ergebnis **1000a** und das Ergebnis der Suche in der gegenwärtigen Nachschlagetabelle **200d** gleich sind, wurden zwei gültige Ergebnisse in den Mapper-Eingaben **504** (**Fig. 6B**) in getrennten Nachschlagetabellen **200c-d** gespeichert. Zwei gültige Ergebnisse sollten für denselben Schlüssel **210** nicht gespeichert werden, die Verarbeitung wird bei Schritt **1316** fortgesetzt.

[0140] Bei Schritt **1314** wird das ankommende Ergebnis **1000a** überprüft, um festzustellen, ob es gültig ist. Ein ankommendes Ergebnis **1000a** ist ungültig, wenn es eine Routeneingabe **302** (**Fig. 4**) ist. Das ankommende Ergebnis **1000a** ist ungültig, wenn es eine Leereingabe **300** (**Fig. 4**) oder ein Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) ist. Teilbaumeingaben-Deskriptor **304** (**Fig. 4**), Routeneingabe **302** (**Fig. 4**) und Leereingabe **300** (**Fig. 4**) wurden bereits in Verbindung mit **Fig. 4** beschrieben. Wenn das ankommende Ergebnis **1000a** ungültig ist, wird die Verarbeitung bei Schritt **1318** fortgesetzt. Wenn dem nicht so ist, wird die Verarbeitung bei Schritt **1320** fortgesetzt.

[0141] Bei Schritt **1318** ist das ankommende Ergebnis **1000a** gültig und das Ergebnis, welches aus der Suche in der gegenwärtigen Nachschlagetabelle **200d** resultiert, ist ungültig. Das ankommende Ergebnis **1000a** wird auf dem abgehenden Ergebnis **1002a** von der gegenwärtigen Nachschlagetabelle **200d** weitergeleitet. Das ankommende Ergebnis **1000a** wird als der Routenindex **102** (**Fig. 3**) weitergeleitet, wenn die gegenwärtige Nachschlagetabelle **200d** die letzte Nachschlagetabelle ist, oder als das ankommende Ergebnis **1000a** zur nächsten Nachschlagetabelle weitergeleitet. Die Verarbeitung ist abgeschlossen.

[0142] Bei Schritt **1316** werden zwei gültige Ergebnisse für den Schlüssel in verschiedenen Nachschlagetabellen gespeichert. Ein Fehler ist während des Speicherns der Routeneingaben in den Nachschlagetabellen **200c-d** aufgetreten. Ein Fehlercode wird erzeugt, so daß der Fehler korrigiert werden kann. Die Verarbeitung ist abgeschlossen.

[0143] Bei Schritt **1320** ist weder das Ergebnis aus einem Absuchen der gegenwärtigen Nachschlagetabelle **200d** noch das ankommende Ergebnis **1000a** gültig. Obwohl es ungültig ist, wird das Ergebnis der Suche in der gegenwärtigen Nachschlagetabelle **200d** als das ankommende Ergebnis **1000a** an die nächste Nachschla-

getabelle weitergeleitet. Die Verarbeitung ist abgeschlossen.

[0144] **Fig. 14** ist ein Ablaufdiagramm, welches ein Verfahren zur Suche nach einem Wert veranschaulicht, welcher einem Suchschlüssel entspricht, der in einer der Nachschlagetabellen **200a–b** gespeichert wird, welche in **Fig. 10A** dargestellt sind.

[0145] Bei Schritt **1340** wird der Mapper **106a** der ersten Stufe in beiden Nachschlagetabellen **200a–b** nach einem Wert abgesucht, welcher einem ersten Abschnitt eines Schlüssels **210a** entspricht. Die Verarbeitung wird bei Schritt **1342** fortgesetzt.

[0146] Wenn bei Schritt **1342** ein gültiger Ergebniswert gefunden wird, nachdem der Mapper **106a** der ersten Stufe mit dem ersten Abschnitt des Schlüssels **210a** abgesucht wurde, wird die Verarbeitung bei Schritt **1352** fortgesetzt. Wenn dem nicht so ist, wird die Verarbeitung bei Schritt **1344** fortgesetzt.

[0147] Wenn bei Schritt **1344** der Wert, welcher aus dem Absuchen des Mappers **106a** der ersten Stufe mit dem ersten Abschnitt des Schlüssels **210a** resultiert, ein Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) ist, wird die Verarbeitung bei Schritt **1346** fortgesetzt. Wenn dem nicht so ist, wird der gültige Wert für den Schlüssel nicht in der gegenwärtigen Nachschlagetabelle gespeichert und ist die Verarbeitung abgeschlossen.

[0148] Bei Schritt **1346** wird die Suche nach einem gültigen Wert in dem Teilbaum fortgesetzt, welcher im Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) identifiziert wurde, welcher während des Absuchens des Mappers der vorhergehenden Stufe gefunden wurde. Der Mapper der nächsten Stufe wird in Abhängigkeit von einem nächsten Abschnitt des Schlüssels **210b–c** und einer Teilbaumauswahl, welche aus einem Absuchen der nächsten Stufe resultiert, nach einem Wert abgesucht. Die Verarbeitung wird bei Schritt **1348** fortgesetzt.

[0149] Bei Schritt **1348** bestimmt das Ergebnis der Suche, ob ein Absuchen eines Mappers der nächsten Stufe notwendig ist. Die Eingabe, welche aus der gegenwärtigen Suche resultiert, kann eine Routeneingabe **302** (**Fig. 4**), eine Leereingabe **300** (**Fig. 4**) oder einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichern. Wenn die Eingabe einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichert, ist eine weitere Suche erforderlich und wird die Verarbeitung bei Schritt **1346** fortgesetzt. Wenn die Eingabe keinen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichert, wird die Verarbeitung bei Schritt **1350** fortgesetzt.

[0150] Wenn bei Schritt **1350** die Eingabe einen Routenindex **102** (**Fig. 3**) speichert, wird die Verarbeitung bei Schritt **1352** fortgesetzt. Wenn dem nicht so ist, wird die Eingabe in einer anderen Nachschlagetabelle gespeichert. Die Verarbeitung ist abgeschlossen.

[0151] Bei Schritt **1352** wird ein gültiger Wert, welcher dem Schlüssel entspricht, in der gegenwärtigen Nachschlagetabelle gespeichert. Der gültige Wert wird als der Routenindex **102** (**Fig. 3**), welcher dem Schlüssel entspricht, weitergeleitet. Die Verarbeitung ist abgeschlossen.

[0152] Dünnbesiedelter Modus Um wieder auf **Fig. 5** zurückzukommen, sorgt die Teilbaumeingabe **404** für den Zugriff auf bis zu 256 mögliche Routenindizes, einen pro Knoten in dem Teilbaum mit 256 Knoten. Die Routenindizes werden in den Mapper-Eingaben **504¹–504¹¹** im Teibaum-Mapper **418** (**Fig. 5**) gespeichert (**Fig. 5**). Die Mapper-Adresse **416** für eine Mapper-Eingabe **504** (**Fig. 6B**) im Teibaum-Mapper **418** (**Fig. 5**) wird in Abhängigkeit von einem dichtbesiedelten Teilbaum-Deskriptor, welcher im Datenfeld **406** gespeichert ist, und einem Teilbaumzeiger, welcher im Zeigerfeld **408** in der Teilbaumeingabe **404** gespeichert ist, bestimmt. Das Format des dichtbesiedelten Teilbaum-Deskriptors wurde bereits in Verbindung mit **Fig. 6A–6B** beschrieben. Der dichtbesiedelte Teilbaum-Deskriptor speichert ein Knoten-Bit **502** (**Fig. 6B**) für jeden Knoten in dem 256-Knoten-Teibaum. Alle Teilbäume weisen jedoch einen verschiedenen Routenindex für jeden der 256 Knoten auf, so kann zum Beispiel ein Teilbaum nur einen Routenindex haben.

[0153] **Fig. 15** veranschaulicht eine Binärbaumdarstellung eines dünnbesiedelten Teilbaumes A und eines dichtbesiedelten Teilbaumes B auf der zweiten Mapper-Stufe **112b**, welche durch Teilbaumeingaben **304** (**Fig. 4**) auf der ersten Mapper-Stufe **112a** indexiert sind. Ein Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) für s1 im Mapper **106a** speichert einen Teilbaumindex **312** für die Teilbaumeingabe **404** für den Teilbaum A. Ein Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) für s0 im Mapper **106a** speichert einen Teilbaumindex **312** für die Teilbaumeingabe **404** für den Teilbaum B.

[0154] Der dichtbesiedelte Teilbaum B weist elf Routenindizes, und zwar r6–r16, und sechs Teilbaumeingaben, und zwar s2–s7, auf. Die Mapper-Adressen **416**, welche den Mapper-Eingaben **504** (**Fig. 6B**) entsprechen, welche die Routeneingaben **302** (**Fig. 4**) und die Teilbaumeingaben **304** (**Fig. 4**) für den Teilbaum B speichern, sind in einem dichtbesiedelten Teilbaum-Deskriptor codiert, wie bereits in Verbindung mit **Fig. 6B** beschrieben wurde.

[0155] Der dünnbesiedelte Teilbaum A speichert zwei Routenindizes, und zwar r1 und r2. Wenn sie in einem dichtbesiedelten Teilbaum-Deskriptor gespeichert werden, wird eine ganze Teilbaumeingabe **404** verwendet, um drei Mapper-Adressen **416** für die Mapper-Eingaben **504** (**Fig. 6B**) bereitzustellen, und zwar r0, r1 und r2.

[0156] Die Anzahl von Routen, welche in der Nachschlagetabelle **200** gespeichert werden, kann durch Codieren eines dünnbesiedelten Teilbaums in einem von einer Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren und Codieren eines dichtbesiedelten Teilbaums in einem dichtbesiedelten Teilbaum-Deskriptor in einer Teilbaumeingabe **404** erhöht werden.

[0157] Ein dichtbesiedelter Teilbaum weist 16 oder mehr Mapper-Eingaben **504** (**Fig. 6B**) auf; das Datenfeld

406 in der Teilbaumeingabe **404** speichert einen dichtbesiedelten Teilbaum-Deskriptor, wie in Verbindung mit **Fig. 6A-6B** beschrieben. Ein dünnbesiedelter Teilbaum weist 15 oder weniger Mapper-Eingaben **504** (**Fig. 6B**) auf das Datenfeld in der Teilbaumeingabe **404** speichert eine Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren. Durch Bereitstellen der Fähigkeit, dünnbesiedelte Teilbäume in dünnbesiedelten Teilbaum-Deskriptoren zu speichern, können mehr Teilbäume im Teilbaumspeicher **400** gespeichert werden und können infolgedessen mehr Routeneingaben in der Nachschlagetabelle **200** gespeichert werden.

[0158] **Fig. 16A-C** veranschaulichen die Abänderungen des Datenfeldes **406** und des Zeigerfeldes **408** in der Teilbaumeingabe **404**, welche in **Fig. 5** dargestellt ist, und des Teilbaumeingaben-Deskriptors **304** (**Fig. 4**), welcher in **Fig. 4** dargestellt ist, um die Speicherung einer Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren in der Teilbaumeingabe **404** zu erlauben.

[0159] Unter Zuwendung zu **Fig. 16A** umfaßt das Datenfeld **406** in einer Teilbaumeingabe **404**, welche im dünnbesiedelten Modus konfiguriert ist, eine Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren **1400¹-1400¹¹** anstelle des dichtbesiedelten Teilbaum-Deskriptors mit einem Bit pro Knoten des Teilbaumes, welcher in Verbindung mit **Fig. 6B** beschrieben wurde. Jeder dichtbesiedelte Teilbaum-Deskriptor **1400¹-1400¹¹** umfaßt einen Knoten-Deskriptor **1402¹-1402¹¹**. Ein Knoten-Deskriptor **1402¹-1402¹¹** ist ein 9-Bit-Wert, welcher eine vollcodierte Route in dem Teilbaum darstellt. Der Knoten-Deskriptor **1402¹-1402¹¹** beschreibt einen einzelnen Knoten oder eine Mehrzahl von Knoten im Teilbaum.

[0160] Unter Zuwendung zu **Fig. 16B** wird, um die Speicherung von dünnbesiedelten Teilbaum-Deskriptoren zu unterstützen, ein Modusfeld **1404** zum Zeigerfeld **408** in der Teilbaumeingabe **404** addiert. Das Zeigerfeld **408** speichert auch die Blockbasisadresse **600¹** und die Blockbasisadresse **600²**, wobei jeder Block **16** zugeordnete Mapper-Adressen **416** umfaßt, welche insgesamt 32 Mapper-Adressen **416** pro Teilbaumeingabe **404** bereitstellen. Das Modusfeld **1404** speichert einen Moduswert.

[0161] Der Moduswert, welcher im Modusfeld **1404** gespeichert wird, zeigt die Anzahl von dünnbesiedelten Teilbaum-Deskriptoren **1400¹-1400¹¹**, welche in der Teilbaumeingabe **404** gespeichert werden, und die Anzahl von Knoten-Deskriptoren **1402¹-1402¹¹**, welche in jedem dünnbesiedelten Teilbaum-Deskriptoren **1400¹-1400¹¹** gespeichert werden, an.

[0162] Tabelle 2 veranschaulicht die Konfiguration der Teilbaum-Eingabe **404** für jeden Modus.

	Knoten / Teilbaum		Bits	Teilbäume / Teilbaum-eingabe	Routen / Teilbaum-eingabe		Unbe-nutzt	Mapper-Eingaben, die für eine Teilbaumeingabe notwendig sind		
Modus	Max.	Min.			Max.	Min.		Max.	Min.	
0	1	1	9		16	16	16	112	32	32
1	2	2	18		10	20	20	76	30	30
2	3	3	27		8	24	24	40	32	32
3	4	4	36		6	24	24	40	30	30
4	7	5	63		4	28	20	4	32	24
5	15	8	135		2	30	16	- 14	32	18

Tabelle 2

[0163] Wenn zum Beispiel unter Bezugnahme auf Tabelle 2 der Moduswert, welcher im Modusfeld **1404** im Zeigerfeld **408** in der Teilbaumeingabe **404** gespeichert wird, auf '4' gesetzt wird, speichert jeder dünnbesiedelte Teilbaum-Deskriptor **1400** in der Teilbaumeingabe **404** zwischen 5 und 7 Knoten-Deskriptoren **1402¹-1402¹¹**. Jeder Knoten-Deskriptor **1402¹-1402¹¹** speichert 9 Bits. Die Gesamtzahl von Bits, welche im dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden, wird durch Multiplizieren der Anzahl der Knoten-Deskriptoren **1402¹-1402¹¹** pro dünnbesiedelten Teilbaum-Deskriptor **1400¹-1400¹¹** mit 9 (der Anzahl von Bits pro Knoten-Deskriptor **1402¹-1402¹¹**) berechnet. Bei Berechnung der Anzahl von Bits pro dünnbesiedelten Teilbaum-Deskriptor **1400** für Modus **4** speichert ein dünnbesiedelter Teilbaum-Deskriptor **1400** mit 7 Knoten-Deskriptoren **1402** 63 Bits (7 Knoten-Deskriptoren * 9 Bits = 63).

[0164] Die Anzahl der dünnbesiedelten Teilbaum-Deskriptoren **1400** pro Teilbaumeingabe **404** wird durch Dividieren der Anzahl von Bits im Datenfeld **406** durch die Anzahl von Bits im dünnbesiedelten Teilbaum-Deskriptor **1400¹-1400¹¹** berechnet. Für Modus **4** ist die Anzahl von Bits im Datenfeld **406** 256 und ist die Anzahl

von Bits im dünnbesiedelten Teilbaum-Deskriptor **63**. Infolgedessen ist die Anzahl von dünnbesiedelten Teilbaum-Deskriptoren **1400¹-1400¹¹** 4 (int(256/63) = 4).

[0165] Die Gesamtzahl von Knoten-Deskriptoren **1402¹-1402¹¹** pro Teilbaumeingabe **404** ist die Anzahl der Knoten pro Teilbaum multipliziert mit der Anzahl von Teilbäumen pro Teilbaumeingabe **404**. Bei der Berechnung fair Modus **4** ist die Gesamtzahl der Knoten-Deskriptoren **1402** pro Teilbaumeingabe **404** 28, wenn 7 Knoten-Deskriptoren **1402¹-1402¹¹** in einem dünnbesiedelten Teilbaum-Deskriptor **1400¹-1400¹¹** gespeichert werden ($7 \cdot 4 = 28$), und 20, wenn 5 Knoten-Deskriptoren **1402** pro dünnbesiedelten Teilbaum-Deskriptor **1400¹-1400¹¹** gespeichert werden ($5 \cdot 4 = 20$).

[0166] Die Spalte der Tabelle 2 mit den Mapper-Eingaben zeigt an, wie viele Mapper-Eingaben **504** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**) durch die Teilbaumeingabe **404** verwendet werden. Der Mapper-Wert wird durch Erhöhen der Knoten pro Teilbaum um eins und durch Multiplizieren mit der Anzahl von Teilbäumen im dünnbesiedelten Teilbaum-Deskriptor berechnet. Die Knoten pro Teilbaum werden um eins erhöht, da eine Mapper-Eingabe mehr als die Anzahl von Knoten pro Teilbaum erforderlich ist, um die Standardeingabe für den Teilbaum speichern.

[0167] Unter Bezugnahme auf die Reihe von Modus **4** in Tabelle 2 sind 32 ($(7 + 1) \cdot 4 = 32$) Mapper-Eingaben pro Teilbaumeingabe **404** erforderlich, wenn sieben Knoten-Deskriptoren **1402** pro dünnbesiedelten Teilbaum-Deskriptor **1400** vorhanden sind, und sind 24 ($((5 + 1) \cdot 4 = 24$) Knoten-Deskriptoren **1402** pro dünnbesiedelten Teilbaum-Deskriptor **1400** erforderlich, wenn fünf Knoten-Deskriptoren **1402** pro dünnbesiedelten Teilbaum-Deskriptor **1400** vorhanden sind. Die Anzahl von Knoten pro Teilbaum und Teilbäumen pro Teilbaumeingabe **404** wird so ausgewählt, daß die größtmögliche Anzahl von Knoten-Deskriptoren pro Teilbaumeingabe **404** 30 nicht überschreitet, da die Mapper-Adressen **416** im Teilbaum-Mapper **418** (**Fig. 5**) in 16 Blockzunahmen zugeordnet werden. Die 32 Mapper-Eingaben **416** werden durch Speichern von zwei Blockbasisadressen **600¹, 600²**, welche im Zeigerfeld **408** gespeichert werden, bereitgestellt.

[0168] Unter Zuwendung zu **Fig. 16C** kann jede Teilbaumeingabe **404** im Teilbaumspeicher 400 im dichtbesiedelten Modus konfiguriert werden, wie in Verbindung mit **Fig. 6B** beschrieben, oder im dünnbesiedelten Modus. Der Teilbaumeingaben-Deskriptor **304** (**Fig. 4**), welcher im Teilbaum-Mapper **418** (**Fig. 5**) gespeichert wird und welcher in Verbindung mit **Fig. 4** für den dichtbesiedelten Modus beschrieben wurde, wird abgeändert, um durch Bereitstellen einer Anzeige, ob die Teilbaumeingabe **404** im dichtbesiedelten Modus oder dünnbesiedelten Modus ist, den dünnbesiedelten Modus zu erlauben. Der Anzeiger wird durch das Typfeld **1406** bereitgestellt.

[0169] Der Zustand des Typfelds **1406** zeigt an, ob die Teilbaumeingabe **4004** im dichtbesiedelten Modus oder im dünnbesiedelten Modus konfiguriert ist. Wenn die Teilbaumeingabe **404** im dünnbesiedelten Modus konfiguriert ist, werden die Werte, welche im Auswahlfeld **1408** des dünnbesiedelten Teilbaum-Deskriptors gespeichert werden, und der Teilbaumindex **312** verwendet, um einen dünnbesiedelten Teilbaum-Deskriptor **1400** auszuwählen. Die Auswahl **1408** des dünnbesiedelten Teilbaum-Deskriptors wird später in Verbindung mit **Fig. 16** ausführlicher beschrieben.

[0170] **Fig. 17** veranschaulicht die Logik **1502** des dünnbesiedelten Modus in der Offset-Logik **700**, welche in **Fig. 8** dargestellt ist, zum Bereitstellen eines Block-Offsets **714**, um eine Mapper-Eingabe **504** (**Fig. 6B**) für einen Knoten in einem dünnbesiedelten Teilbaum auszuwählen. Die Logik **1502** des dünnbesiedelten Modus stellt das Block-Offset **714** in Abhängigkeit von einem Knoten-Deskriptor **1420** bereit, welcher in einem dünnbesiedelten Teilbaum-Deskriptor **1400** in einer Teilbaumeingabe **404** gespeichert wird. Die Offset-Logik **700** umfaßt auch die Logik **1500** des dichtbesiedelten Modus. Die Logik **1500** des dichtbesiedelten Modus umfaßt die Knotenauswahl **706** und die Einserzähl-Logik **708** zum Bereitstellen eines Block-Offsets **714** für eine Route in einem dichtbesiedelten Teilbaum. Die Logik **1500** des dichtbesiedelten Modus wurde bereits in Verbindung mit **Fig. 8** beschrieben.

[0171] Wenn der Zustand des Typfeldes **1406** anzeigt, daß die Teilbaumeingabe **404** im dünnbesiedelten Modus konfiguriert ist, werden die Teilbaumdaten **412** von der Teilbaumeingabe **404** an die Logik **1502** des dünnbesiedelten Modus weitergeleitet. Die Logik **1502** des dünnbesiedelten Modus wird in Verbindung mit **Fig. 18** beschrieben.

[0172] **Fig. 18** veranschaulicht die Logik **1502** des dünnbesiedelten Modus, welche in der Offset-Logik **700** in **Fig. 17** dargestellt ist. Die Logik **1502** des dünnbesiedelten Modus umfaßt eine Teilbaumauswahl-Logik **1600**, einen Multiplexer **1602**, einen Assoziativspeicher ("CAM", nach Engl. Content Addressable Memory) **1606** und eine Umwandlungslogik **1604**. Die dünnbesiedelten Teilbaum-Deskriptoren **1400¹-1400¹¹**, welche im Datenfeld **406** in der ausgewählten Teilbaumeingabe **404** gespeichert werden, werden auf den Teilbaumdaten **412** an die Offset-Logik **700** weitergeleitet. Die Offset-Logik **700** leitet die dünnbesiedelten Teilbaum-Deskriptoren **1400¹-1400¹¹** an den Multiplexer **1602** in der Logik **1502** des dünnbesiedelten Modus weiter. Einer der dünnbesiedelten Teilbaum-Deskriptoren **1400¹** in den Teilbaumdaten **412** wird durch die Auswahl **1614**, welche durch die Teilbaumauswahl-Logik **1600** erzeugt wird, ausgewählt.

[0173] Die Teilbaumauswahl-Logik **1600** erzeugt die Auswahl **1614**, um den dünnbesiedelten Teilbaum-Deskriptor **1400¹** auszuwählen in Abhängigkeit von der Auswahl **1408** dünnbesiedelter Teilbaum-Deskriptoren,

welche von der Mapper-Eingabe weitergeleitet wird, die auf der vorhergehenden Mapper-Stufe ausgewählt wurde, und dem Modus **1404**, welcher im Zeigerfeld **408** in der ausgewählten Teilbaumeingabe **404** gespeichert wird. Tabelle 3 veranschaulicht den ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1400¹** und die jeweiligen Teilbaumdatenbits **412**, welche auf dem ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1610** vom Multiplexer **1602** für eine Teilbaumeingabe **404** des Modus **4** weitergeleitet werden. Unter Bezugnahme auf die Reihe von Modus **4** in Tabelle 2 können in der Teilbaumeingabe **404** des Modus **4** vier dünnbesiedelte Teilbaum-Deskriptoren gespeichert werden. Jeder der vier dünnbesiedelten Teilbaum-Deskriptoren **1400** ist 63 Bits und kann von sieben bis fünf Knoten-Deskriptoren **1402** speichern. Infolgedessen beginnt jeder der vier dünnbesiedelten Teilbaum-Deskriptoren **1400** bei einer 63-Bit-Grenze. Der erste dünnbesiedelte Teilbaum-Deskriptor **1400¹** wird bei 62:0 im Datenfeld **406** gespeichert. Der zweite dünnbesiedelte Teilbaum-Deskriptor **1400²** wird bei 125:63 im Datenfeld **406** gespeichert. Der dritte dünnbesiedelte Teilbaum-Deskriptor **1400³** wird bei 188:126 im Datenfeld **406** gespeichert und der vierte dünnbesiedelte Teilbaum-Deskriptor **1400⁴** wird bei 251:189 Bits im Datenfeld gespeichert. Die jeweiligen Bits im Datenfeld **406** werden durch die Auswahl **1408** dünnbesiedelter Teilbaum-Deskriptoren ausgewählt. Wenn zum Beispiel bei Betrachtung von Tabelle 3 die Auswahl **1408** dünnbesiedelter Teilbaum-Deskriptoren '0001' ist, wird der zweite dünnbesiedelte Teilbaum-Deskriptor **1400²** ausgewählt und werden die Bits 125:63 der 256-Bit-Teilbaumdaten **412** durch den Multiplexer **1602** auf dem ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1610** an die Umwandlungslogik **1604** weitergeleitet.

Auswahl dünnbesiedelter Teilbaum-Deskriptoren	Teilbaumdaten	Dünnbesiedelter Teilbaum-Deskriptor
0000	Bits 62:0	1
0001	Bits 125:63	2
0010	Bits 188:126	3
0011	Bits 251:189	4

Tabelle 3

[0174] Jede Teilbaumeingabe **404** im Teilbaumspeicher **400** kann im dünnbesiedelten Modus oder im dichtbesiedelten Modus konfiguriert werden. Jede Teilbaumeingabe **404**, welche im dünnbesiedelten Modus konfiguriert ist, kann so konfiguriert werden, daß sie eine verschiedene Anzahl von Knoten-Deskriptoren **1402** pro dünnbesiedelten Teilbaum-Deskriptor **1400** durch Modus **1404** speichert. Alle dünnbesiedelten Teilbaum-Deskriptoren **1400** in der Teilbaumeingabe **404**, welche im dünnbesiedelten Modus konfiguriert ist, speichern dieselbe Anzahl von Knoten-Deskriptoren **1402** pro dünnbesiedelten Teilbaum-Deskriptor **1400**.

[0175] Ein Knoten-Deskriptor **1402** kann codiert werden, um mehrfache Knoten im Teilbaum darzustellen. Mehrfache Acht-Bit-Knoten, welche durch den Knoten-Deskriptor **1402** dargestellt werden, werden durch Maskieren von einigen der acht Bits identifiziert. Anstelle des Speicherns von Maskenbits bei jedem Knoten-Deskriptor **1402** wird ein Neun-Bit-Knoten-Deskriptor **1402** verwendet, um die acht Bit breiten Knoten, welche durch den Knoten-Deskriptor **1402** dargestellt werden, voll zu codieren. Der acht Bit breite Knoten wird in einem neun Bit breiten Knoten-Deskriptor **1402** unter Verwendung der Bit-Lauflängencodierung codiert. Die Bit-Lauflängencodierung erlaubt es, zu identifizieren, welche der acht Bits des Knoten maskiert sind.

[0176] Die Umwandlungslogik **1604** wandelt die neun Bit breiten Knoten-Deskriptoren **1402¹–1402⁴**, welche im ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden, in Acht-Bit-CAM-Werte **1612** um, welche Bits enthalten, die auf 'X' (beliebig) gesetzt sind, und lädt die Acht-Bit-CAM-Werte **1612** in den CAM **1606**. Ein Beispiel der Umwandlung der 9-Bit-Knoten-Deskriptoren **1402** in 8-Bit-CAM-Werte **1612** durch die Umwandlungslogik **1604** ist in der folgenden Tabelle 4 dargestellt.

8-Bit-Wert	9-Bit-Code
101100XX	101100100
100XXXXX	100100000
001111XX	001111100
0011XXXX	001110000

Tabelle 4

[0177] Die Spalte mit den 9-Bit-Codes veranschaulicht Werte, welche in den Knoten-Deskriptoren **1402** gespeichert werden. Bei Betrachtung der ersten Reihe in Tabelle 4 ist der 9-Bit-Code, welcher im Knoten-Deskriptor **1402** gespeichert wird, '101100100' und wird der entsprechende 8-Bit-Wert '101100XX' im CAM **1606** gespeichert. Die Umwandlungslogik **1604** wandelt den 9-Bit-Code durch Absuchen des 9-Bit-Codes von rechts nach links nach dem ersten Bit, welches auf '1' gesetzt ist, um. Bei Betrachtung der Bits im 9-Bit-Code von rechts nach links sind die ersten beiden Bits auf '0' gesetzt und ist das dritte Bit auf '1' gesetzt. Die Umwandlungslogik **1604** wandelt die '100' in zwei Beliebige ('XX') um, da zwei '0' rechts vom ersten '1' stehen. Der erste '1' wird ignoriert und die restlichen Bits werden direkt in die nächsten Bits des 8-Bit-Wertes kopiert.

[0178] Bei Betrachtung der zweiten Reihe in Tabelle 4 ist der 9-Bit-Code, welcher im Knoten-Deskriptor **1402** gespeichert ist, '100100000'. Die Umwandlungslogik **1604** wandelt den 9-Bit-Code durch Absuchen des 9-Bit-Codes von rechts nach links nach dem ersten '1' um. Die fünfte Ziffer speichert einen '1'. Der 9-Bit-Code wird in einen 8-Bit-Wert umgewandelt, wobei die fünf niedrigstwertigen Bits ("LSBs", nach Engl. Least Significant Bits) auf "beliebig" ("X") gesetzt werden. Durch Speichern der Knoten-Deskriptoren **1402** unter Verwendung der 9-Bit-Bit-Lauflängencodierung wird die Anzahl von Bits, welche pro Knoten-Deskriptor **1402** erforderlich sind, auf ein Minimum herabgesetzt, wodurch die Anzahl von Knoten-Deskriptoren **1402**, welche in der Nachschlagetabelle **200** gespeichert werden können, erhöht wird.

[0179] Nach der Umwandlung der 9-Bit-Knoten-Deskriptoren **1402** in 8-Bit-Werte lädt die Umwandlungslogik **1604** die 8-Bit-Werte in den CAM **1606**. Die 8-Bit-Werte werden in derselben Reihenfolge in den CAM **1606** geladen, wie die Knoten-Deskriptoren **1402** im ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden; das heißt, von der kürzesten zur längsten Übereinstimmung. Der CAM **1606** stellt Speicherung zum Speichern der größtmöglichen Anzahl von Knoten-Deskriptoren **1402** pro dünnbesiedelten Teilbaum-Deskriptor **1400** bereit. Infolgedessen ist der CAM **8** Bits breit mal sechzehn Eingaben tief, um fünfzehn Eingaben bereitzustellen, um die größtmögliche Anzahl von Knoten-Deskriptoren **1402** für einen dünnbesiedelten Teilbaum-Deskriptor **1400** des Modus **5** und eine Mapper-Standardadresse zu speichern. Der CAM **1606** hat einen Resolver für Ternär-Fähigkeit und Mehrfachübereinstimmung eingebaut. Ein CAM **1606** kleiner Größe kann im Gegensatz zur Bereitstellung eines echten Assoziativspeichers in Gattern realisiert werden; das heißt, der CAM **1606** kann in einer Hardware-Schaltungsanordnung realisiert werden, welche einen CAM emuliert.

[0180] Die Anzahl von Knoten-Deskriptoren **1402**, welche in einem dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden, bestimmt die Teilbaumeingabe **402**, in welcher der dünnbesiedelte Teilbaum-Deskriptor **1400** gespeichert wird. Dünnbesiedelte Teilbaum-Deskriptoren **1400**, welche Knoten-Deskriptoren **1402** innerhalb des Bereichs für einen bestimmten Modus speichern, werden in derselben Teilbaumeingabe **404** gespeichert. Eine Mapper-Standardadresse wird für eine Standardroute für jeden Teilbaum berechnet. Ein 8-Bit-Standardwert wird an der ersten Stelle im CAM **1606** dauerhaft gespeichert, um die Mapper-Standardadresse zu berechnen.

[0181] Nachdem die 8-Bit-Werte für den ausgewählten dünnbesiedelten Teilbaum **1400** in den CAM **1606** geladen wurden, wird der CAM **1606** mit dem nächsten Abschnitt des Schlüssels **210b** abgesucht. Die Eingabe im CAM **1606**, welche mit der größten Anzahl von Bits im nächsten Abschnitt des Schlüssels **210b** übereinstimmt, wird ausgewählt. Die Übereinstimmungsadresse, welche aus dem Absuchen des CAMs resultiert, wird als das Block-Offset **714** weitergeleitet. Das Block-Offset **714** wird verwendet, um die Mapper-Adresse **416** für die Mapper-Eingabe zu bestimmen, welche der Route entspricht, die im Teilbaum-Mapper **418** (Fig. 5) gespeichert wird.

[0182] Fig. 19A-D veranschaulichen die Auswahl eines Block-Offsets **714** für einen Knoten in einem dünnbesiedelten Teilbaum **1700**. Fig. 19A ist eine grafische Darstellung der Routen in dem dünnbesiedelten Teilbaum **1700**. Knoten im Teilbaum **1700** entsprechen einer von drei Routen r0, r1 und r2, wobei r0 die Standardroute für den Teilbaum **1700** ist. Zwei Routen r1, r2 werden in den Knoten-Deskriptoren **1402¹** und **1402²** in einem dünnbesiedelten Teilbaum-Deskriptor **1400** codiert. Ein Wert für die Standardroute r0 wird in der ersten Eingabe **1702** im CAM **1606** dauerhaft gespeichert. Unter Bezugnahme auf Tabelle 2 wird ein dünnbesiedelter Teilbaum-Deskriptor **1400** mit zwei Knoten-Deskriptoren **1402** in der Teilbaumeingabe **404** gespeichert, wobei das Modusfeld **1404** auf '1' gesetzt wird.

[0183] Bei Betrachtung des Teilbaums **1700** entspricht r2 allen Knoten, welche mit 10xxxxxx übereinstimmen, und entspricht r1 allen Knoten, welche mit 010xxxxx übereinstimmen. Um die Anzahl von Bits, welche durch jeden Knoten-Deskriptor **1402¹**, **1402²** benötigt werden, um jede Route im dünnbesiedelten Teilbaum-Deskriptor **1400** zu beschreiben, auf ein Minimum herabzusetzen, wird der Knoten-Deskriptor **1402¹**, **1402²** unter Verwendung der Bit-Lauflängencodierung codiert. Das Codierungsverfahren verwendet ein Bit mehr als die Anzahl von Bits, welche zum Vollcodieren des Knotens verwendet werden. Ein '1' wird an der Stelle des ersten 'X' ('beliebig') eingefügt und die restlichen X werden als 0 codiert. Infolgedessen wird die Route 10xxxxxx in 101000000 übersetzt und wird 010xxxxx in 010100000 übersetzt.

[0184] Fig. 19B veranschaulicht die Speicherung der Knoten-Deskriptoren **1402¹** und **1402²** im dünnbesiedelten Teilbaum-Deskriptor **1400**. Die Knoten-Deskriptoren **1402¹** und **1402²** werden in einer Teilbaumeingabe

404 gespeichert, wobei das Modusfeld **1404** auf '1' gesetzt wird, da zwei Knoten-Deskriptoren **1402¹** und **1402²** in dem dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden. Die längste Übereinstimmung für den Teilbaum ist r1, da r1 eine Übereinstimmung der ersten drei Bits erfordert und r2 eine Übereinstimmung der ersten zwei Bits erfordert. Die Knoten-Deskriptoren **1402¹** und **1402²** werden in der Reihenfolge von der kürzesten zur längsten Übereinstimmung im dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert, wobei der Knoten-Deskriptor **1402¹** für r2 zuerst gespeichert wird und der Knoten-Deskriptor **1402²** für r1 anschließend gespeichert wird.

[0185] **Fig. 19C** veranschaulicht die Umwandlung des Knoten-Deskriptors **1402²** in einen 8-Bit-Maskenwert **1706**. Bei Betrachtung der Knoten-Deskriptor-Bits **1708¹–1708⁹** von links nach rechts wird der erste '1' in Bit **1708⁶** gespeichert, was das Ende des Maskenbits für den 8-Bit-Maskenwert **1706** markiert. Um den Knoten-Deskriptor **1402²** in einen 8-Bit-Maskenwert **1706** umzuwandeln, werden die folgenden Bit-Umwandlungen durchgeführt. Der '0', welcher im Knoten-Deskriptor-Bit **1708¹** gespeichert wird, wird in 'X' umgewandelt und im 8-Bit-Maskenwert-Bit **1710¹** gespeichert. Der '0', welcher im Knoten-Deskriptor-Bit **1708²** gespeichert wird, wird in 'X' umgewandelt und im 8-Bit-Maskenwert-Bit **1710²** gespeichert. Der '0', welcher im Knoten-Deskriptor-Bit **1708³** gespeichert wird, wird in 'X' umgewandelt und im 8-Bit-Maskenwert-Bit **1710³** gespeichert. Der '0', welcher im Knoten-Deskriptor-Bit **1708⁴** gespeichert wird, wird in 'X' umgewandelt und im 8-Bit-Maskenwert-Bit **1710⁴** gespeichert. Der '0', welcher im Knoten-Deskriptor-Bit **1708⁵** gespeichert wird, wird in 'X' umgewandelt und im 8-Bit-Maskenwert-Bit **1710⁵** gespeichert. Der '1', welcher im Knoten-Deskriptor-Bit **1708⁶** gespeichert wird, wird ignoriert. Der '0', welcher im Knoten-Deskriptor-Bit **1708⁷** gespeichert wird, wird im 8-Bit-Maskenwert-Bit **1710⁶** gespeichert. Der '1', welcher im Knoten-Deskriptor-Bit **1708⁸** gespeichert wird, wird im 8-Bit-Maskenwert-Bit **1710⁷** gespeichert. Der '0', welcher im Knoten-Deskriptor-Bit **1708⁹** gespeichert wird, wird im 8-Bit-Maskenwert-Bit **1710⁸** gespeichert.

[0186] **Fig. 19D** veranschaulicht die Speicherung der Knoten-Deskriptoren **1402¹** und **1402²** im CAM **1606** und die entsprechenden Mapper-Eingaben **504¹–504³**, welche im Teilbaum-Mappen **418** (**Fig. 5**) für den ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden. Die 9-Bit-Knoten-Deskriptoren **1402¹** und **1402²**, welche in dem ausgewählten Teilbaum-Deskriptor **1400** gespeichert werden, werden in der Umwandlungslogik **1604** (**Fig. 18**) umgewandelt und in den CAM **1606** geladen. Die erste Eingabe **1702** im CAM **1606** ist die Standardeingabe für r0, welche in Teilbaum **1700** in **Fig. 19A** dargestellt ist. Die zweite Eingabe **1704** wird vom ersten Knoten-Deskriptor **1402¹** umgewandelt, welcher in dem ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert wird. Die zweite Eingabe **1704** ist die kürzeste Übereinstimmung, welche für r2 umgewandelt wird. Der zweite Knoten-Deskriptor **1402²**, welcher in dem ausgewählten Teilbaum-Deskriptor **1400** gespeichert wird, wird von 010100000 in 010XXXXX umgewandelt und in der dritten Eingabe **1706** im CAM **1606** gespeichert.

[0187] Ein Absuchen des CAM **1606** führt zum Block-Offset **714** (**Fig. 18**). Das Block-Offset **714** wird verwendet, um die Mapper-Adresse **416** für die Mapper-Eingabe **504¹–504³**, welche im Teilbaum-Mapper **418** (**Fig. 5**) gespeichert ist, zu bestimmen. Der CAM **1606** wird mit zweitem Abschnitt des Schlüssels **210b** nach der Eingabe **1702, 1704, 1706**, welche die längste Übereinstimmung speichert, abgesucht. Das Block-Offset **714**, welche durch den CAM **1606** bereitgestellt wird, wird in Abhängigkeit von einer der Blockbasisadressen **600¹**, **600²**, welche im Zeigerfeld **408** in der ausgewählten Teilbaumeingabe **404** gespeichert sind, mit einer Teilbaum-Basisadresse kombiniert.

[0188] **Fig. 20** ist ein Bockdiagramm, welches die Basisauswahl-Logik **1800** des dünnbesiedelten Modus in der Zeiger-Logik **702**, welche in **Fig. 8** dargestellt ist, veranschaulicht. Die Zeiger-Logik **702** wählt die Basisadresse **716** aus, welche verwendet wird, um die Mapper-Adresse **416** für die Mapper-Eingabe **504** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**) zu berechnen. Die Zeiger-Logik **701** umfaßt die Basisauswahl-Logik **710** des dichtbesiedelten Modus und die Basisauswahl-Logik **1800** des dünnbesiedelten Modus, von welchen eine ausgewählt wird in Abhängigkeit des Zustandes von Typ **1406**, welcher im Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) gespeichert wird, welcher von der vorhergehenden Mapper-Stufe weitergeleitet wurde. Wie bereits beschrieben wurde, zeigt der Zustand von Typ **1406** an, ob die Teilbaumeingabe **404** im dichtbesiedelten Modus konfiguriert ist.

[0189] Die Basisauswahl-Logik **1800** des dünnbesiedelten Modus berechnet die Basisadresse **716** für den dünnbesiedelten Teilbaum-Deskriptor **1400**, wenn die Teilbaumeingabe **404** eine Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren **1400** speichert. Die Basisauswahl-Logik **1800** des dünnbesiedelten Modus berechnet die Basisadresse **716** unter Verwendung des Moduswertes **1608**, welcher im Modusfeld **1404** gespeichert ist, und des Teilbaumzeigers **414**, welcher im Blockbasisadreßfeld **600¹**, **600²** in der Teilbaumeingabe **404** gespeichert ist, und der Auswahl **1408** dünnbesiedelter Teilbaum-Deskriptoren, welche im Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) gespeichert wird, welcher von der vorhergehenden Mapper-Stufe weitergeleitet wurde. Die Basisadresse **716** wird folgendermaßen berechnet:

Basisadresse (für den dünnbesiedelten Teilbaum-Deskriptor) = Blockbasisadresse + Basis-Offset, wobei Basis-Offset = ((1 + Knoten/Teilbaum)·Auswahl dünnbesiedelter Teilbaum-Deskriptoren).

[0190] Um zum Beispiel die Basisadresse **716** für den Beginn von Teilbaumzahl **2** in einer Teilbaumeingabe **404**, welche im dünnbesiedelten Modus **4** konfiguriert ist, zu finden, wird zuerst das Basis-Offset berechnet. Die Auswahl **1408** dünnbesiedelter Teilbaum-Deskriptoren für die Teilbaumzahl **2** ist '1' und die Anzahl von Knoten/Teilbäumen ist 7 (siehe Tabelle 2). Das Basis-Offset ist $8((1+7)\cdot1)$. Jede Blockbasisadresse **600¹**, **600²** ist die Basisadresse für einen Block von 16 Mapper-Adressen, welche für die Teilbaumeingabe **404** zugeordnet werden.

[0191] Das Basis-Offset für die Teilbaumzahl **2** ist 8, was weniger als 16 ist, weshalb die Blockadresse für Teilbaum **2** die Blockbasisadresse **600¹** ist und die Basisadresse **716** für den dünnbesiedelten Teilbaum-Deskriptor die Blockbasisadresse **600¹** + 8 ist. Die folgende Tabelle 5 veranschaulicht die Teilbaum-Basisadresse für jeden der vier Teilbäume in einer Teilbaumeingabe **404**, welche im Modus **4** konfiguriert ist.

Teilbaum-Basisadresse	Teilbaum
Blockbasisadresse 1 + 0	1
Blockbasisadresse 2 + 8	2
Blockbasisadresse 2 + 0	3
Blockbasisadresse 2 + 8	4

Tabelle 5

[0192] **Fig. 21** veranschaulicht einen dichtbesiedelten Teilbaum-Deskriptor und einen dünnbesiedelten Teilbaum-Deskriptor, welche im Teilbaumspeicher **400** gespeichert werden. **Fig. 21** wird in Verbindung mit **Fig. 15** beschrieben. Ein dichtbesiedelter Teilbaum-Deskriptor für Teilbaum B (**Fig. 21**) wird im Datenfeld **406¹** in der Teilbaumeingabe **404¹** gespeichert. Ein dünnbesiedelter Teilbaum-Deskriptor **1400¹** für Teilbaum A (**Fig. 21**) wird im Datenfeld **406²** in der Teibaumeingabe **404²** gespeichert. Der dichtbesiedelte Teilbaum-Deskriptor speichert ein Knoten-Bit für jeden Knoten auf der unteren Stufe von Teilbaum B, wie bereits in Verbindung mit **Fig. 6B** beschrieben wurde. Der dünnbesiedelte Teilbaum-Deskriptor **1400¹** umfaßt die Knoten-Deskriptoren **1402¹** und **1402²**, welche den Routen r4 und r5 entsprechen, wie in Verbindung mit **Fig. 19B** beschrieben wurde. Der Teilbaumindex **312** wählt die Teilbaumeingabe **404¹**, **404²** aus.

[0193] Der Teilbaumindex **312**, welcher in einem Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** für s0 (**Fig. 15**) gespeichert wird, wählt die Teilbaumeingabe **404¹** aus. Der Teilbaumindex **312**, welcher in einem Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) in einer Mapper-Eingabe **504** (**Fig. 6B**) im Mapper **106a** für s1 (**Fig. 15**) gespeichert wird, wählt die Teilbaumeingabe **404²** aus. Infolgedessen kann der Teilbaumspeicher **400** Teilbaumeingaben **404¹**, **404²** für dünnbesiedelte Teilbäume und dichtbesiedelte Teilbäume speichern.

[0194] **Fig. 22** ist ein Ablaufdiagramm, welches ein Verfahren zur Bereitstellung einer Mapper-Adresse **416** (**Fig. 5**) für eine Mapper-Eingabe **504** (**Fig. 6B**) in einem Teilbaum-Mapper **418** (**Fig. 5**) veranschaulicht, welche eine Route für einen Knoten in einem dünnbesiedelten Teilbaum und einem dichtbesiedelten Teilbaum speichert. Jede Teilbaumeingabe **404** kann eine Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren oder einen einzelnen dichtbesiedelten Teilbaum-Deskriptor speichern. Eine Kombination von dünnbesiedelten Teilbaum-Deskriptoren und dichtbesiedelten Teilbaum-Deskriptoren ist möglich in Abhängigkeit davon, wie die Routen im Binärbaum verteilt werden. Die Flexibilität, dünnbesiedelten Modus und dichtbesiedelte Teilbaum-Deskriptoren in Teilbaumeingaben **404** im Teilbaumspeicher **400** zu mischen und übereinzustimmen, erlaubt einen besseren Gebrauch des Teilbaumspeichers **400**.

[0195] Bei Schritt **1900** wird die Konfiguration der ausgewählten Teilbaumeingabe **404** vom Zustand von Typ **1406** (**Fig. 16C**) bestimmt, welcher im Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) gespeichert wird, welcher auf der vorhergehenden Mapper-Stufe ausgewählt wurde. Wenn der Typ der Teilbaumeingabe **404** im dünnbesiedelten Modus konfiguriert ist, wird die Verarbeitung bei Schritt **1902** fortgesetzt. Wenn dem nicht so ist, wird die Verarbeitung bei Schritt **1914** fortgesetzt.

[0196] Bei Schritt **1902** wird die Teileingabe **404** im dünnbesiedelten Modus konfiguriert. Eine Teilbaumeingabe **404**, welche im dünnbesiedelten Modus konfiguriert ist, speichert eine Mehrzahl von dünnbesiedelten Teilbaum-Deskriptoren **1400**. Die Anzahl von dünnbesiedelten Teilbaum-Deskriptoren **1400**, welche in der Teilbaumeingabe **404** gespeichert werden, hängt vom Zustand des Modusfeldes **1404** ab. Die Logik **1502** des dünnbesiedelten Modus in der Offset-Logik **700** wählt den dünnbesiedelten Teilbaum-Deskriptor **1400** aus der Teilbaumeingabe **404** in Abhängigkeit von der Auswahl **1408** dünnbesiedelter Teilbaum-Deskriptoren, welche im Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) gespeichert wird, welcher von der vorhergehende Mapper-Stufe weitergeleitet wurde, und dem Inhalt des Modusfeldes **1404** aus, wie bereits in Verbindung mit **Fig. 14** beschrieben wurde. Die Verarbeitung wird bei Schritt **1904** fortgesetzt.

[0197] Bei Schritt **1904** werden die codierten 9-Bit-Werte, welche in den Knoten-Deskriptoren **1402** im ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden, in 8-Bit-Werte umgewandelt und in der Reihenfolge von der kürzesten zur längsten Übereinstimmung im CAM **1606** gespeichert. Die Verarbeitung wird bei Schritt **1906** fortgesetzt.

[0198] Bei Schritt **1906** wird der CAM **1606** mit dem nächsten Abschnitt des Schlüssels **2106** nach der CAM-Eingabe, welche die längste Übereinstimmung speichert, abgesucht. Die Verarbeitung wird bei Schritt **1908** fortgesetzt.

[0199] Bei Schritt **1908** wird die Adresse der Stelle im CAM **1606**, welche die längste Übereinstimmung für den nächsten Abschnitt des Schlüssels **210b** speichert, als das Block-Offset **714** weitergeleitet. Das Block-Offset **714** wird verwendet, um die Mapper-Adresse **416** (**Fig. 5**) der Mapper-Eingabe **504** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**) zu berechnen. Die Verarbeitung wird bei Schritt **1910** fortgesetzt.

[0200] Bei Schritt **1910** wird die Basisadresse **716** (**Fig. 20**) für den ausgewählten dünnbesiedelten Teilbaum-Deskriptor **1400** berechnet in Abhängigkeit von der Auswahl **1408** dünnbesiedelter Teilbaum-Deskriptoren, welche im Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) gespeichert wird, welcher von der vorhergehenden Mapper-Stufe weitergeleitet wurde, und dem Inhalt des Modusfeldes **1404**, welches in der ausgewählten Teilbaumeingabe **404** gespeichert wird. Die Verarbeitung wird bei Schritt **1912** fortgesetzt.

[0201] Bei Schritt **1912** wird die Mapper-Adresse **416** durch Addieren des Block-Offsets **714** und der Basisadresse **716** in der Addierer-Logik **704** (**Fig. 8**) berechnet. Die Mapper-Eingabe **504** (**Fig. 6B**), welche durch die Mapper-Adresse **416** im Teilbaum-Mapper **418** (**Fig. 5**) identifiziert wird, speichert entweder eine Routeneingabe **302** (**Fig. 4**) oder einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**). Wenn die Mapper-Eingabe **504** (**Fig. 6B**) eine Routeneingabe **302** (**Fig. 4**) speichert, ist die Suche abgeschlossen. Wenn die Mapper-Eingabe **504** (**Fig. 6B**) einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) speichert, wird die Suche nach dem Wert, welcher dem Schlüssel **210** entspricht, auf der nächsten Mapper-Stufe fortgesetzt.

[0202] Bei Schritt **1914** wird die Teilbaumeingabe **404** im dichtbesiedelten Modus konfiguriert und speichert einen einzelnen dichtbesiedelten Teilbaum-Deskriptor im Datenfeld **406**. Das Block-Offset **714** wird durch Zählen der Anzahl von '1', welche im dichtbesiedelten Teilbaum-Deskriptor gespeichert werden, welcher im Datenfeld **406** in der Teilbaumeingabe **404** gespeichert wird, berechnet, wie bereits in Verbindung mit **Fig. 6B** beschrieben. Die Verarbeitung wird bei Schritt **1916** fortgesetzt.

[0203] Bei Schritt **1916** speichert die Teilbaumeingabe **404** sechzehn Blockbasisadressen **600** im Zeigerfeld **408** in der Teilbaumeingabe **404**. Einer der Blockbasiszeiger **600** wird durch die Basisauswahl-Logik **710** des dichtbesiedelten Modus in der Zeigerlogik **702** ausgewählt, welche in Verbindung mit **Fig. 8** bereits beschrieben wurde. Die Verarbeitung wird bei Schritt **1912** fortgesetzt.

Schrittweise Aktualisierung

[0204] **Fig. 23** veranschaulicht eine Binärbaumdarstellung einer neuen Route, welche zur Nachschlagetabelle **200** zu addieren ist. Der Binärbaum veranschaulicht Routen, welche in der Nachschlagetabelle **200** für Mapper-Stufe_1 **2000**, Mapper-Stufe_2 **2002** und Mapper-Stufe_3 **2004** gespeichert werden. Die Mapper-Stufe_2 **2002** speichert Routen für die Teilbäume A und B. Die Mapper-Stufe_3 **2004** speichert Routen für die Teilbäume A_1, A_2, B_1 und B_2 . $s5$ stellt einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**), welcher in einem Teilbaum-Mapper **418** (**Fig. 5**) gespeichert wird, dar. Der Teilbaumeingaben-Deskriptor **304** (**Fig. 5**) für $s5$ speichert einen Zeiger zu Teilbaum B_2 , welcher es erlaubt, die Suche nach einer Route längster Übereinstimmung für einen Schlüssel **210** auf der Mapper-Stufe_3 **2004** fortzusetzen.

[0205] Teilbaum B_2 **2006** ist ein dünnbesiedelter Teilbaum, da er nur zwei Routen, $r6$ und $h1$, aufweist. Infolgedessen werden die Knoten-Deskriptoren **1402** (**Fig. 16A**) für die Knoten $r6$ und $h1$ in einem dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert, wie bereits in Verbindung mit **Fig. 14A** beschrieben wurde. Der dünnbesiedelte Teilbaum-Deskriptor **1400** für den Teilbaum B_2 **2006** wird in einer Teilbaumeingabe **404** gespeichert, wobei das Modusfeld **1404** im Teilbaumspeicher **400** auf '1' gesetzt wird, da zwei Knoten-Deskriptoren **1402** vorhanden sind, welche im dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden.

[0206] Eine neue Route $h2$, welche im Teilbaum B_2 **2008** dargestellt ist, ist zu Nachschlagetabelle **200** zu addieren. Die neue Route $h1$ kann nicht direkt zum Teilbaum B_2 **2006** in der Nachschlagetabelle addiert werden, da die Addition einer Route zum Teilbaum B_2 **2006** die Anzahl von Knoten-Deskriptoren **1402**, welche im dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden, von 2 auf 3 erhöht. Die Addition eines Knoten-Deskriptors **1402** zum dünnbesiedelten Teilbaum-Deskriptor **1400** erfordert die Zuordnung eines neuen dünnbesiedelten Teilbaum-Deskriptors **1400** in einer Teilbaumeingabe **404**, wobei das Modusfeld **1404** auf '2' gesetzt wird. Infolgedessen erfordert die Addition der neuen Route $h1$ die Ersetzung des Teilbaumes B_2 **2006** durch den Teilbaum B_2 **2008**.

[0207] **Fig. 24** veranschaulicht Aktualisierungsrouten, welche in einem Prozessorspeicher **2400** gespeichert werden. Eine Kopie des Binärbaumes, welcher in der Nachschlagetabelle **200** gespeichert wird, wird auch im Prozessorspeicher **2400** getrennt von der Nachschlagetabelle **200** gespeichert. Die Routen, welche für den

Teilbaum B_2 **2006** gespeichert werden, werden auf den Teilbaum B_2' **2008** im Prozessorspeicher **2400** kopiert und die neue Route h2 wird zum Teilbaum B_2' **2008** addiert.

[0208] Eine Routen-Aktualisierungsroutine **2402** erzeugt eine Folge von Routen-Aktualisierungsbefehlen **2404**, um den Teilbaum B_2' **2008** zu Nachschlagetabelle **200** zu addieren, und leitet die Routenaktualisierungen **2404** an die Tabellen-Aktualisierungsroutine **2406** weiter. Die Tabellen-Aktualisierungsroutine **2406** erzeugt Tabellenaktualisierungen **2410** für die Routenaktualisierungen **2402** und leitet die Aktualisierungszyklen **2412** weiter, um die Nachschlagetabelle **200** mit den Routenaktualisierungen **2404** zu aktualisieren. Die Aktualisierungszyklen **2412** schreiben die Routenaktualisierungen in die geeigneten Speicherstellen in dem Teilbaumspeicher **400** (**Fig. 5**) und dem Teilbaum-Mapper **418** (**Fig. 5**).

[0209] Um auf **Fig. 23** zurückzukommen, umfassen die Aktualisierungszyklen **2412** Befehle, um einen Abschnitt des Teilbaum-Mappers **418** (**Fig. 5**) zuzuordnen, um die Routen für den neuen Teilbaum B_2' **2008** in den Mapper-Eingaben **504** (**Fig. 6B**) zu speichern. Der Teilbaum B_2' **2008** enthält Routeneingaben, welche in den Mapper-Eingaben **504** (**Fig. 6B**) für die Routen h1 und r6 und die neue Route h2 gespeichert werden. Nachdem die Routeneingaben für den Teilbaum B_2' **2008** in den Mapper-Eingaben **504** (**Fig. 6B**) im Teilbaum-Mapper **418** (**Fig. 5**) gespeichert wurden, werden die Knoten-Deskriptoren **1402** geschaffen und in einem dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert. Der dünnbesiedelte Teilbaum-Deskriptor **1400** wird in der Teilbaumeingabe **404** gespeichert. Der Modus **1404** der Teilbaumeingabe **404** hängt mit der Anzahl von Knoten-Deskriptoren **1402**, welche im dünnbesiedelten Teilbaum-Deskriptor **1400** gespeichert werden, zusammen.

[0210] Nachdem der dünnbesiedelte Teilbaum-Deskriptor **1400** für den Teilbaum B_2' **2008** in einer Teilbaumeingabe **404** im Teilbaumspeicher **400** in der Nachschlagetabelle **200** gespeichert wurde, wird der Teilbaumeingaben-Deskriptor **304** (**Fig. 4**), welcher durch s5 dargestellt ist, abgeändert, um statt auf den Teilbaum B_2 **2006** auf den Teilbaum B_2' **2008** zu zeigen. Während der Teilbaum B_2' **2008** zu Nachschlagetabelle addiert wird, kann auf die Routen r6 und h1, welche im Teilbaum B_2 **2006** gespeichert werden, durch s5 zugegriffen werden. Danach wird Teilbaum B_2' **2008** in der Nachschlagetabelle gespeichert und wird s5 abgeändert, um auf den Teilbaum B_2 **2006** zu zeigen, und es kann auf die Routen r6, h1 und die neue Route h2 zugegriffen werden. Infolgedessen kann der Teilbaum B_2 **2006** weiterhin nach Routenindizes abgesucht werden, welche Route r6 und h1 entsprechen, während die neue Route h2 zur Nachschlagetabelle **200** addiert wird.

[0211] **Fig. 25** veranschaulicht die neue Route h2, welche in **Fig. 23** dargestellt ist, gespeichert in der Mapper-Eingabe **504c⁴** im Teilbaum-Mapper **418b** in der Nachschlagetabelle **200**. **Fig. 25** wird in Verbindung mit der Binärbaumdarstellung beschrieben, welche in **Fig. 24** dargestellt ist.

[0212] Teilbaum B auf der Mapper-Stufe_2 **2002** weist drei Routen auf; und zwar r3, s4 und s5. Teilbaum B ist ein dünnbesiedelter Teilbaum, da er weniger als sechzehn Routen aufweist. Die Knoten-Deskriptoren **1402a¹–1402a³** für den Teilbaum B r3, s4 und s5 werden in einem dünnbesiedelten Teilbaum-Deskriptor **1400a¹** in Teilbaumeingabe **404a** im Teilbaumspeicher **400a** gespeichert. Eine Mapper-Eingabe **504a²–504a⁴** wird im Teilbaum-Mapper **418a** für jede Route in Teilbaum B gespeichert. Eine Standardroute für den Teilbaum B wird in der Mapper-Eingabe **504a¹** im Teilbaum-Mapper **418a** gespeichert. Jede Mapper-Eingabe **504a²–504a⁴** speichert eine Routeneingabe **302** (**Fig. 4**) oder einen Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) für den Knoten. Ein Teilbaumeingaben-Deskriptor **304** (**Fig. 4**) wird für Route s4 in **504a³** und s5 in **504a⁴** gespeichert. Der Teilbaumeingaben-Deskriptor **304** (**Fig. 4**), welcher in der Mapper-Eingabe **504a⁴** für s5 gespeichert wird, stellt den Teilbaumindex **312b** für den Teilbaumspeicher **400b** bereit, um die Suche auf der nächsten Stufe zu beginnen; das heißt, für Mapper-Stufe_3 2004.

[0213] Der Teilbaum B_2 ist ebenfalls ein dünnbesiedelter Teilbaum, da er zwei Routen aufweist; und zwar h1 und r6. Die Knoten-Deskriptoren **1402b¹–1402b²** werden in dem dünnbesiedelten Teilbaum-Deskriptor **1400b¹** in der Teilbaumeingabe **404b** im Teilbaumspeicher **400b** gespeichert. Jede Route im Teilbaum B_2 wird in einer Mapper-Eingabe **504b²–504b³** gespeichert und die Standardroute für den Teilbaum B_2 wird in der Mapper-Eingabe **504b¹** gespeichert.

[0214] Um im Teilbaum B_2 **2006** nach Route h1 zu suchen, wird die Adresse der Teilbaumeingabe **404a**, welche den dünnbesiedelten Teilbaum-Deskriptor **1400a** speichert, welcher den Knoten-Deskriptor **1402** für Route s5 speichert, auf dem Teilbaumindex **312a** an den Teilbaumspeicher **400a** weitergeleitet. Das Datenfeld **406** und das Zeigerfeld **408**, welche in der ausgewählten Teilbaumeingabe **404a** gespeichert werden, werden auf den Teilbaumdaten **412a** und den Teilbaumzeigern **414a** an die Mapper-Adreßlogik **402a** weitergeleitet. Die Mapper-Adreßlogik **402a** erzeugt die Mapper-Adresse **416a** für die Mapper-Eingabe **504a⁴**, welche die Teilbaumeingabe für s5 speichert. Die Mapper-Adresse **416a** hängt von den Teilbaumdaten **412a**, den Teilbaumzeigern **414a** und einem nächsten Abschnitt des Schlüssels **210b** ab. Die Teilbaumeingabe für s5 wird auf dem Teilbaumindex **312b** an den Teilbaumspeicher **400b** weitergeleitet.

[0215] Der Teilbaumspeicher **400b** speichert die Knoten-Deskriptoren **1402b², 1402b¹** für den Teilbaum B_2 **2006**. Der dünnbesiedelte Teilbaum-Deskriptor **1400b¹** für B_2 wird in der Teilbaumeingabe **404b** gespeichert. Das Datenfeld **406** und das Zeigerfeld **408**, welche in der Teilbaumeingabe **404b** gespeichert werden, werden auf den Teilbaumdaten **412b** und den Teilbaumzeigern **414b** an die Mapper-Adreßlogik **402b** weitergeleitet.

Die Mapper-Adresslogik **402b** erzeugt die Mapper-Adresse **416b** für die Mapper-Eingabe **504b³**, welche die Routeneingabe für h1 speichert. Die Mapper-Adresse **416b** hängt von den Teilbaumdaten **412b**, den Teilbaumzeigern **414b** und einem nächsten Abschnitt des Schlüssels **210c** ab.

[0216] Um Route h2 zum Teilbaum B₂ **2006** zu addieren, wird ein Block von zuvor unbgenutzten Mapper-Eingaben **602c** im Teilbaum-Mapper **418b** zugeordnet, um die Mapper-Eingaben **504c²–504c⁴** zu speichern, welche die Routen r6, h1 und h2 für den Teilbaum B₂' **2008** speichern. Die Mapper-Eingabe **504c¹** speichert die Standardeingabe für den Teilbaum B₂' **2008**; das heißt, denselben Wert, welcher in der Mapper-Eingabe **504b¹** gespeichert wird. Die Mapper-Eingabe **504c²** speichert die Routeneingabe für Route r6, das heißt, denselben Wert, welcher in der Mapper-Eingabe **504b²** gespeichert wird. Die Mapper-Eingabe **504c³** speichert die Routeneingabe für Route h1, das heißt, denselben Wert, welcher in der Mapper-Eingabe **504b³** gespeichert wird. Die Mapper-Eingabe **504c⁴** speichert die Routeneingabe für die neue Route h2. Während der Block von Mapper-Eingaben **504c^{1–4}** geschrieben wird, kann auf die Routeneingaben, welche in den Mapper-Eingaben **504b^{1–504b³}** gespeichert werden, durch die Teilbaumeingabe, welche für die Route s5 in **504a⁴** im Teilbaum-Mappert **418a** gespeichert wird, zugegriffen werden.

[0217] Nach Speichern der Mapper-Eingaben **504c^{1–4}** für den Teilbaum B₂' **2008** im Teilbaum-Mapper **418b** wird ein dünnbesiedelter Teilbaum-Deskriptor **1400c¹** zum Teilbaumspeicher **400b** addiert. Die Anzahl von Knoten-Deskriptoren **1402c^{1–3}** ist kleiner als sechzehn, weshalb die Knoten-Deskriptoren **1402c^{1–3}** in einem dünnbesiedelten Teilbaum-Deskriptor **1400c¹** gespeichert werden. Die Stelle des Teilbaum-Deskriptors **1400¹** im Teilbaumspeicher **400b** hängt von der Anzahl von Knoten-Deskriptoren **1402c^{1–3}** ab, welche mit dem dünnbesiedelten Teilbaum-Deskriptor **1400c¹** verbunden sind. Durch Addieren einer neuen Route zum Teilbaum B₂ **2006** erhöhte sich die Anzahl von Knoten-Deskriptoren **1402c^{1–1402c³}**, welche für den dünnbesiedelten Teilbaum-Deskriptor **1400c¹** zu speichern sind, von zwei auf drei. Der dünnbesiedelte Teilbaum-Deskriptor **1400c¹** wird in einer Teilbaumeingabe **404c** gespeichert mit drei Knoten-Deskriptoren pro dünnbesiedelten Teilbaum-Deskriptor und dem Modusfeld **1404** auf '2' gesetzt. Der dünnbesiedelte Teilbaum-Deskriptor **1400c¹** wird in einer gegenwärtigen Teilbaumeingabe **404c** des Modus **3** gespeichert, wenn Platz verfügbar ist, oder es wird eine Teilbaumeingabe des Modus **3** zugeordnet. Knoten-Deskriptoren für Routen in B₂' **2008** werden in den Knoten-Deskriptoren **1402c^{1–3}** im dünnbesiedelten Teilbaum-Deskriptor **1400c¹** in der Teilbaumeingabe **404c** des Modus **3** gespeichert.

[0218] Nachdem der dünnbesiedelte Teilbaum-Deskriptor **1400c¹** und die Knoten-Deskriptoren **1402c^{1–3}** im Teilbaumspeicher **400b** gespeichert wurden, kann auf den Teilbaum B₂' **2008** zugegriffen werden. Um Zugriff auf B₂' **2008** bereitzustellen, wird die Teilbaumeingabe **504a⁴** abgeändert, um den dünnbesiedelten Teilbaum-Deskriptor **1400c¹** in der Teilbaumeingabe **404c** statt des dünnbesiedelten Teilbaum-Deskriptors **1400b¹** in der Teilbaumeingabe **404b** zu indexieren. Es kann auf die Routeneingabe für Route h2, welche in der Mapper-Eingabe **504c⁴** gespeichert wird, und die Routen r6 und h1, welche in den Mapper-Eingaben **504c²** bzw. **504c³** gespeichert werden, zugegriffen werden.

[0219] Auf die Mapper-Eingaben **504b^{1–504b³}** kann nicht mehr zugegriffen werden und sie werden freigegeben und zur künftigen Zuordnung auf eine frei Liste (nicht dargestellt) gesetzt. Auch auf den dünnbesiedelten Teilbaum-Deskriptor **1400b¹** kann nicht mehr zugegriffen werden. Infolgedessen wird der dünnbesiedelte Teilbaum-Deskriptor **1400b¹** freigegeben und zur künftigen Zuordnung auf eine frei Liste (nicht dargestellt) gesetzt.

[0220] Die Addition einer Route zu einem dünnbesiedelten Teilbaum wurde beschrieben. Eine Route kann auch zu einem dichtbesiedelten Teilbaum addiert werden durch Speichern eines neuen dichtbesiedelten Teilbaum-Deskriptors in einer neu zugeordneten Teilbaumeingabe **404** und der entsprechenden Mapper-Eingaben im Teilbaum-Mapper **418** und Ändern der Teilbaumeingabe, welche in der Mapper-Eingabe **504a⁴** gespeichert wird, um die neu zugeordnete Teilbaumeingabe **404** zu indexieren.

[0221] **Fig. 26** ist ein Ablaufdiagramm, welches die Schritte zur Durchführung einer schrittweisen Aktualisierung veranschaulicht, um eine Route zur Nachschlagetabelle **200** zu addieren, welche in **Fig. 25** dargestellt ist.

[0222] Bei Schritt **2200** wird die Anzahl von Routen pro Teilbaum berechnet, um zu bestimmen, ob die Routenaktualisierung zu einem dünnbesiedelten oder dichtbesiedelten Teilbaum führt. Wenn der Teilbaum nach der Routenaktualisierung dichtbesiedelt ist, wird die Verarbeitung bei Schritt **2218** fortgesetzt. Wenn der Teilbaum nach der Routenaktualisierung dünnbesiedelt ist, wird die Verarbeitung bei Schritt **2202** fortgesetzt.

[0223] Bei Schritt **2202** ist der Teilbaum dünnbesiedelt. Der dünnbesiedelte Teilbaum-Modus wird festgelegt. Die Verarbeitung wird bei Schritt **2204** fortgesetzt.

[0224] Bei Schritt **2204** wird eine Liste von teilweise gefüllten Teilbaumeingaben **404**, welche im Teilbaum-Mapper **418** (**Fig. 5**) gespeichert sind, abgesucht, um zu bestimmen, ob der neue dünnbesiedelte Teilbaum-Deskriptor **1400c¹** in einer zuvor zugeordneten Teilbaumeingabe **404** gespeichert werden kann. Zum Beispiel können vier dünnbesiedelte Teilbaum-Deskriptoren **1400c^{1–1400c⁴}** in einer Teilbaumeingabe **404** des Modus **4** gespeichert werden. Wenn nur drei gespeichert werden, ist die Teilbaumeingabe **404** teilweise gefüllt und wird auf einer Liste von teilweise gefüllten Teilbaumeingaben **404** gespeichert. Wenn eine teilweise gefüllte Teilbaumeingabe **404** verfügbar ist, wird die Verarbeitung bei Schritt **2208** fortgesetzt. Wenn dem nicht so ist, wird die Verarbeitung bei Schritt **2206** fortgesetzt.

[0225] Bei Schritt 2206 wird eine neue Teilbaumeingabe 404c zum Speichern des dünnbesiedelten Teilbaum-Deskriptors 1400c¹ zugeordnet und die Mapper-Eingaben 504c¹–504c⁴ werden zugeordnet im Teilbaum-Mapper zum Speichern der Mapper-Eingaben 504 (Fig. 6B) für die Knoten-Deskriptoren 1402c¹–1402c³, welche im dünnbesiedelten Teilbaum-Deskriptor 1400c¹ in der neu zugeordneten Teilbaumeingabe 404c gespeichert werden. Ein Zeiger zu einem zugeordneten Block von Mapper-Eingaben 504c¹–504c⁴ im Teilbaum-Mapper 418 (Fig. 5) wird im Zeigerfeld 408 in der neuen Teilbaumeingabe 404c gespeichert. Die Verarbeitung wird bei Schritt 2208 fortgesetzt.

[0226] Bei Schritt 2208 wird die Stelle der ersten Mapper-Eingabe 504c¹ im Teilbaum-Mapper für den dünnbesiedelten Teilbaum-Deskriptor 1400c¹ bestimmt von Zeigern, welche im Zeigerfeld 408 in der Teilbaumeingabe 404c gespeichert werden, und dem Modus, welcher im Modusfeld 1404 in der Teilbaumeingabe 404c gespeichert wird. Die Verarbeitung wird bei Schritt 2210 fortgesetzt.

[0227] Bei Schritt 2210 werden die Routeneingaben für den dünnbesiedelten Teilbaum in den Mapper-Eingaben 504c¹–504c⁴ im Teilbaum-Mapper 418b gespeichert. Die Verarbeitung wird bei Schritt 2212 fortgesetzt.

[0228] Bei Schritt 2212 wird der dünnbesiedelte Teilbaum-Deskriptor 1400c¹, welcher die Knoten-Deskriptoren 1402c¹–1402c³ speichert, in der Teilbaumeingabe 404c gespeichert. Die Verarbeitung wird bei Schritt 2214 fortgesetzt.

[0229] Bei Schritt 2214 wird der Teilbaumeingaben-Deskriptor 304 (Fig. 4), welcher in der Mapper-Eingabe 504a⁴ gespeichert wird, abgeändert, um den neuen dünnbesiedelten Teilbaum-Deskriptor 1400c¹, welcher in der Teilbaumeingabe 404c gespeichert wird, zu indexieren. Die Verarbeitung wird bei Schritt 2216 fortgesetzt.

[0230] Bei Schritt 2216 kann auf die Mapper-Eingaben 504b¹–504b³ und den dünnbesiedelten Teilbaum-Deskriptor 1400b nicht mehr zugegriffen werden. Die Mapper-Eingaben 504b¹–504b³ werden auf eine freie Liste von Mapper-Eingaben 504 (Fig. 6B) für den Teilbaum-Mapper 418b gesetzt und können zum Speichern anderer Routen zugeordnet werden. Die erste verfügbare Stelle in der Teilbaumeingabe 404b wird in der Liste von teilweise gefüllten Teilbaumeingaben aktualisiert. Die Verarbeitung ist abgeschlossen.

[0231] Bei Schritt 2218 wird eine neue Teilbaumeingabe 404 aus einer Liste von freien Teilbaumeingaben 404, welche im Prozessorspeicher 2400 (Fig. 24) gespeichert werden, zugeordnet. Die neue Teilbaumeingabe 404 wird zum Speichern eines neuen dichtbesiedelten Teilbaum-Deskriptors zugeordnet. Blöcke von Mapper-Eingaben 504 (Fig. 6B) im Teilbaum-Mapper 418b werden zum Speichern der Routen zugeordnet. Die Zeiger zu den Blöcken von zugeordneten Mapper-Eingaben 504 (Fig. 6B) werden im Zeigerfeld 408 (Fig. 7) in der Teilbaumeingabe 404 (Fig. 5) gespeichert. Die Verarbeitung wird bei Schritt 2220 fortgesetzt.

[0232] Bei Schritt 2220 wird der neue dichtbesiedelte Teilbaum-Deskriptor in das Datenfeld 406 in der neuen Teilbaumeingabe 404 geschrieben, wie bereits in Verbindung mit Fig. 6A–B beschrieben wurde. Die Verarbeitung wird bei Schritt 2222 fortgesetzt.

[0233] Bei Schritt 2222 werden die Routeneingaben für den dichtbesiedelten Teilbaum in den Mapper-Eingaben 504 (Fig. 6B) im Teilbaum-Mapper 418 (Fig. 5) gespeichert, welcher durch die Zeiger identifiziert wird, welche im Zeigerfeld 408 in der Mapper-Eingabe 404 gespeichert werden. Die Verarbeitung wird bei Schritt 2224 fortgesetzt.

[0234] Bei Schritt 2224 wird der Teilbaumeingaben-Deskriptor 304 (Fig. 4), welcher in der Mapper-Eingabe 504a⁴ gespeichert wird, abgeändert, um den neuen dichtbesiedelten Teilbaum-Deskriptor, welcher in der neuen Teilbaumeingabe 404c gespeichert wird, zu indexieren. Es kann nun auf Routeneingabe für h2, welche in der Mapper-Eingabe 504c⁴ gespeichert wird, zugegriffen werden. Die Verarbeitung wird bei Schritt 2226 fortgesetzt.

[0235] Bei Schritt 2226 werden die Mapper-Eingaben 504 (Fig. 6B), welche durch die Zeiger indexiert werden, welche im Zeigerfeld 408 in der alten Teilbaumeingabe 404 gespeichert werden, an die freie Liste von Mapper-Eingaben zurückgegeben, welche im Prozessorspeicher 2400 (Fig. 24) gespeichert werden. Die alte Teilbaumeingabe 404b wird zu der freien Liste von Teilbaumeingaben addiert, welche im Prozessorspeicher 2400 (Fig. 24) gespeichert werden.

[0236] Der Prozeß wurde für die Addition einer Route zur Nachschlagetabelle beschrieben. Ein ähnlicher Prozeß wird durchgeführt, um eine Route von der Nachschlagetabelle zu löschen. Um zum Beispiel h2 504c⁴ vom Teilbaum B₂' zu löschen, ist erforderlich Speichern eines neuen dünnbesiedelten Teilbaum-Deskriptors mit zwei Knoten-Deskriptoren für die Routen r6 und h1, Speichern des dünnbesiedelten Teilbaum-Deskriptors in einer Teilbaumeingabe des Modus 2, Aktualisieren des entsprechenden Teilbaum-Mappers und Abändern des Teilbaumeingaben-Deskriptors 304 (Fig. 4), welcher in der Mapper-Eingabe 504a⁴ gespeichert wird, um den aktualisierten Teilbaumeingaben-Deskriptor zu indexieren, welcher in der neuen Teilbaumeingabe 404 gespeichert wird.

[0237] Obwohl diese Erfindung insbesondere unter Bezugnahmen auf bevorzugte Ausführungsbeispiele davon dargestellt und beschrieben wurde, ist für den Durchschnittsfachmann leicht erkennbar, daß verschiedene Änderungen in Form und Einzelheiten daran vorgenommen werden können, ohne sich vom Rahmen der Erfindung, welcher durch die beigefügten Patentansprüche umfaßt wird, zu entfernen.

[0238] Zusammenfassung: Wir stellen ein Verfahren und eine Vorrichtung zum Steigern der Tiefe einer Nach-

schlagetabelle vor. Die Nachschlagetabelle umfaßt wenigstens zwei Nachschlageeinheiten. Eine Suchanforderung nach einem Endergebniswert, welcher dem Schlüssel entspricht, wird an jede der Nachschlageeinheiten parallel ausgegeben. Jede der Nachschlageeinheiten führt parallel eine Mehrstufensuche nach dem Endergebniswert durch. Der Endergebniswert, welcher nur in einer der Nachschlageeinheiten gespeichert wird, wird durch eine der Nachschlageeinheiten bereitgestellt.

Patentansprüche

1. Nachschlagetabelle, umfassend:

eine erste Nachschlageeinheit, welche einen Schlüssel empfängt und ein erstes Ergebnis durch eine Mehrstufensuche in der ersten Nachschlageeinheit bereitstellt; und

eine zweite Nachschlageeinheit, welche den Schlüssel empfängt und ein zweites Ergebnis durch eine Mehrstufensuche in der zweiten Nachschlageeinheit bereitstellt, wobei sowohl die erste Nachschlageeinheit als auch die zweite Nachschlageeinheit den Schlüssel parallel empfangen und nur eine der Einheiten ein Endergebnis für den Schlüssel bereitstellt, welches nur in einer der Nachschlageeinheiten in Abhängigkeit von dem ersten Ergebnis und dem zweiten Ergebnis gespeichert wird.

2. Nachschlagetabelle nach Anspruch 1, wobei Endergebnisse für den Schlüssel unter den Nachschlageeinheiten in Abhängigkeit von der Gesamtzahl von Endergebnissen, welche in der Nachschlagetabelle zu speichern sind, verteilt werden.

3. Nachschlagetabelle nach Anspruch 1, wobei der Schlüssel eine IP-Adresse ist und das Endergebnis ein Routenindex ist.

4. Nachschlagetabelle nach Anspruch 1, wobei eine erste Speicherstelle in der ersten Nachschlageeinheit, welche durch den Schlüssel indexiert wird, das Endergebnis speichert und eine zweite Speicherstelle in der zweiten Nachschlageeinheit, welche durch den Schlüssel indexiert wird, eine Anzeige speichert, daß das Endergebnis in der ersten Nachschlageeinheit gespeichert wird.

5. Nachschlagetabelle nach Anspruch 1, wobei jede der ersten Nachschlageeinheit und der zweiten Nachschlageeinheit eine Mehrzahl von Mapper-Stufen umfaßt und jede der Mapper-Stufen Speicherstellen zum Speichern von Ergebnissen umfaßt.

6. Nachschlagetabelle nach Anspruch 4, wobei das Endergebnis, welches dem Schlüssel entspricht, welcher an einer Stelle auf irgendeiner der Mehrzahl von Mapper-Stufen in der ersten Nachschlageeinheit gespeichert wird, an die zweite Nachschlageeinheit weitergeleitet wird und durch die zweite Nachschlageeinheit bereitgestellt wird.

7. Nachschlagetabelle nach Anspruch 1, wobei das Endergebnis, welches dem Schlüssel entspricht, durch die Nachschlageeinheit, in welcher das Endergebnis gespeichert wird, bereitgestellt wird.

8. Verfahren zur Bereitstellung eines Endergebnisses für einen Schlüssel, umfassend die folgenden Schritte:

Paralleles Empfangen des Schlüssels durch eine erste Nachschlageeinheit und eine zweite Nachschlageeinheit;

Durchführen einer Mehrstufensuche durch die erste Nachschlageeinheit, welche zu einem ersten Ergebnis führt;

durchführen einer Mehrstufensuche durch die zweite Nachschlageeinheit, welche zu einem zweiten Ergebnis führt; und

Bereitstellen des Endergebnisses für den Schlüssel durch eine der Einheiten in Abhängigkeit von dem ersten Ergebnis und dem zweiten Ergebnis.

9. Verfahren nach Anspruch 8, welches die folgenden Schritte umfaßt:

Verteilen der Endergebnisse unter den Einheiten in Abhängigkeit von der Gesamtzahl von Endergebnissen, welche in den Nachschlagetabellen zu speichern sind.

10. Verfahren nach Anspruch 8, wobei der Schlüssel eine IP-Adresse ist und das Endergebnis ein Routenindex ist.

11. Verfahren nach Anspruch 8, wobei eine erste Speicherstelle in der ersten Nachschlageeinheit, welche

durch einen Schlüssel indexiert wird, das Endergebnis speichert und eine zweite Speicherstelle in der zweiten Nachschlageeinheit, welche durch den Schlüssel indexiert wird, eine Anzeige speichert, daß das Endergebnis in der ersten Nachschlageeinheit gespeichert wird.

12. Verfahren nach Anspruch 8, wobei jede der ersten Nachschlageeinheit und der zweiten Nachschlageeinheit eine Mehrzahl von Mapper-Stufen umfaßt und jede der Mapper-Stufen Speicherstellen zum Speichern von Ergebnissen umfaßt.

13. Verfahren nach Anspruch 12, wobei das Ergebnis, welches dem Schlüssel entspricht, welcher an einer Stelle auf einer der Mehrzahl von Mapper-Stufen in der ersten Nachschlageeinheit gespeichert wird, an die zweite Nachschlageeinheit weitergeleitet wird und durch die zweite Nachschlageeinheit bereitgestellt wird.

14. Verfahren nach Anspruch 8, wobei das Endergebnis, welches dem Schlüssel entspricht, durch die Nachschlageeinheit, in welcher das Endergebnis gespeichert wird, bereitgestellt wird.

15. Nachschlagetabelle, umfassend:

eine erste Nachschlageeinheit, welche einen Schlüssel empfängt und ein erstes Ergebnis durch eine Mehrstufigensuche in der ersten Nachschlageeinheit bereitstellt;
eine zweite Nachschlageeinheit, welche den Schlüssel empfängt und ein zweites Ergebnis durch eine Mehrstufigensuche in der zweiten Nachschlageeinheit bereitstellt, und
Mittel zum Bereitstellen eines Endergebnisses, welches nur in einer der Nachschlageeinheiten gespeichert wird, für den Schlüssel, welcher sowohl von der ersten Nachschlageeinheit als auch der zweiten Nachschlageeinheit parallel empfangen wird, in Abhängigkeit von dem ersten Ergebnis und dem zweiten Ergebnis.

16. Nachschlagetabelle nach Anspruch 15, wobei die Endergebnisse für den Schlüssel unter den Nachschlageeinheiten in Abhängigkeit von der Gesamtzahl von Endergebnissen, welche in der Nachschlagetabelle zu speichern sind, verteilt werden.

17. Nachschlagetabelle nach Anspruch 15, wobei der Schlüssel eine IP-Adresse ist und das Endergebnis ein Routenindex ist.

18. Nachschlagetabelle nach Anspruch 15, wobei eine erste Speicherstelle in der ersten Nachschlageeinheit, welche durch den Schlüssel indexiert wird, das Ergebnis speichert und eine zweite Speicherstelle in der zweiten Nachschlageeinheit, welche durch den Schlüssel indexiert wird, eine Anzeige speichert, daß das Ergebnis in der ersten Nachschlageeinheit gespeichert wird.

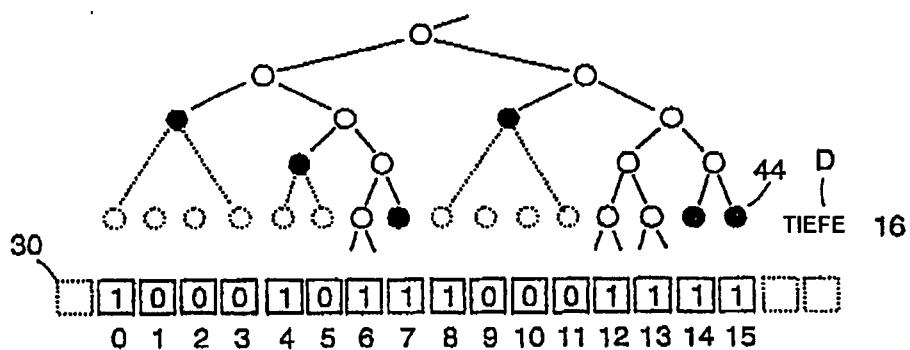
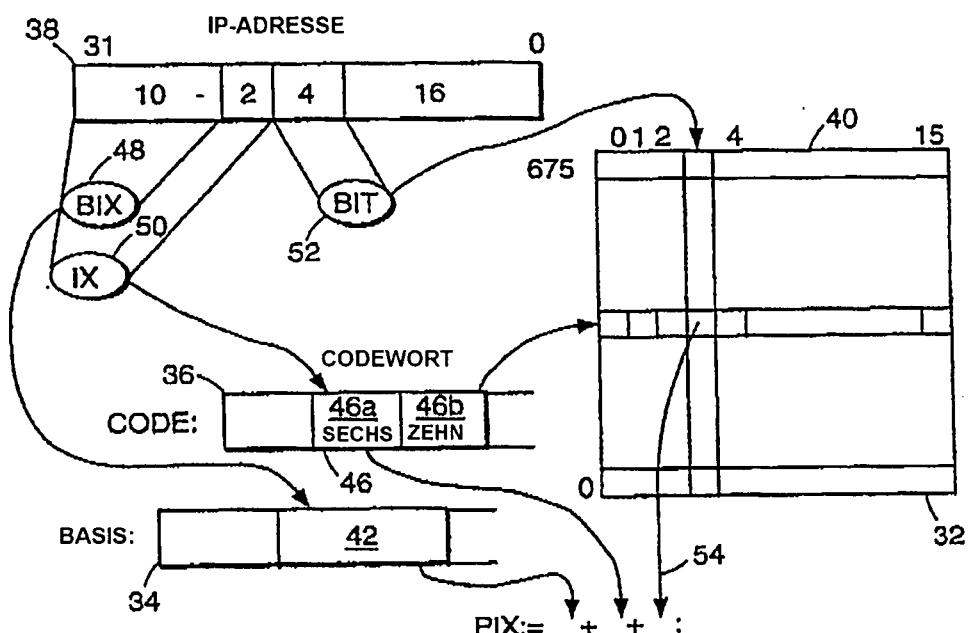
19. Nachschlagetabelle nach Anspruch 15, wobei jede der ersten Nachschlageeinheit und der zweiten Nachschlageeinheit eine Mehrzahl von Mapper-Stufen umfaßt und jede der Mapper-Stufen Speicherstellen zum Speichern umfaßt.

20. Nachschlagetabelle nach Anspruch 19, wobei das Endergebnis, welches dem Schlüssel entspricht, welcher an einer Stelle auf einer der Mehrzahl von Mapper-Stufen in der ersten Nachschlageeinheit gespeichert wird, an die zweite Nachschlageeinheit weitergeleitet wird und durch die zweite Nachschlageeinheit bereitgestellt wird.

21. Nachschlagetabelle nach Anspruch 15, wobei das Endergebnis, welches dem Schlüssel entspricht, durch die Nachschlageeinheit, in welcher das Endergebnis gespeichert wird, bereitgestellt wird.

Es folgen 34 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1A
STAND DER TECHNIKFIG. 1B
STAND DER TECHNIK

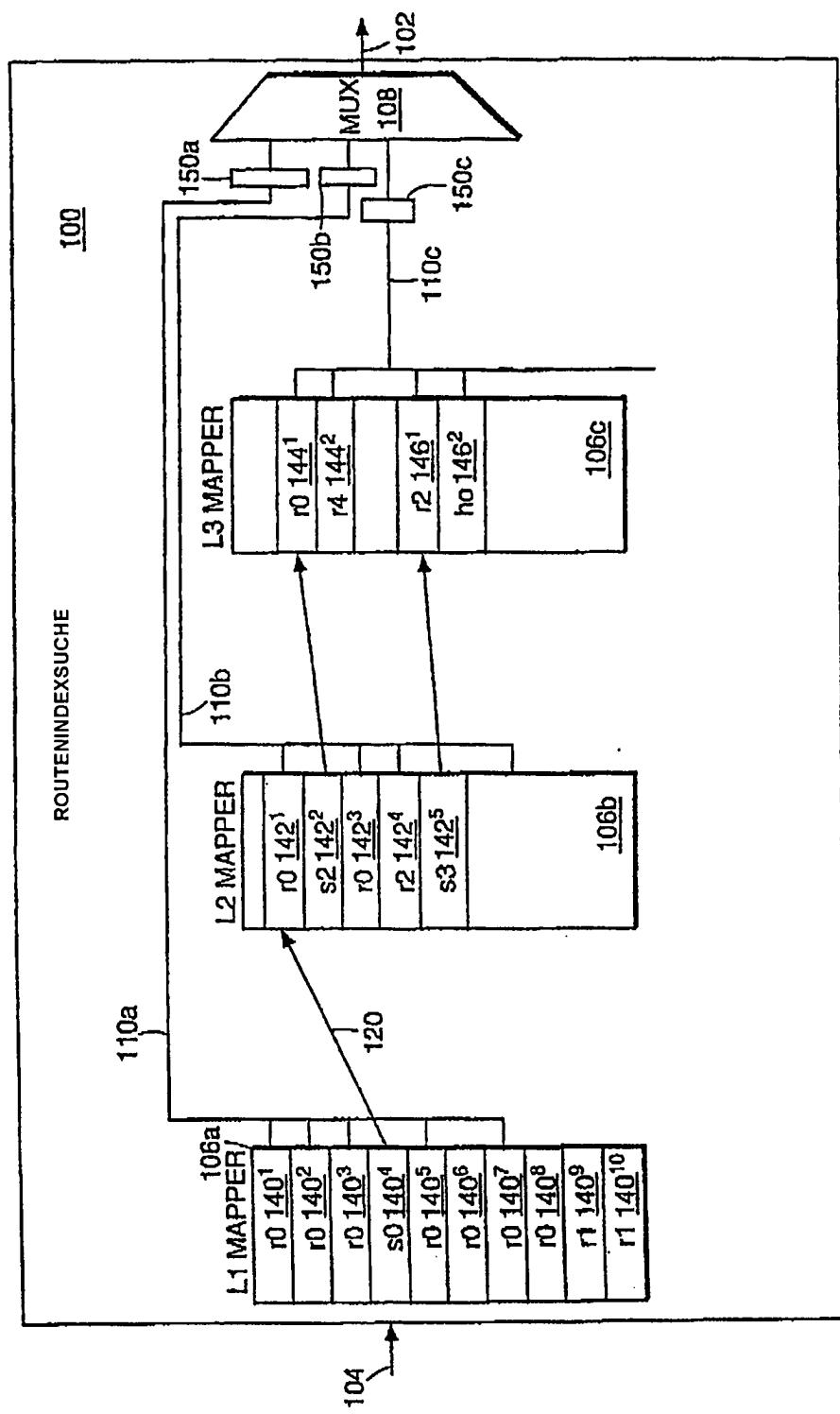


FIG. 2A

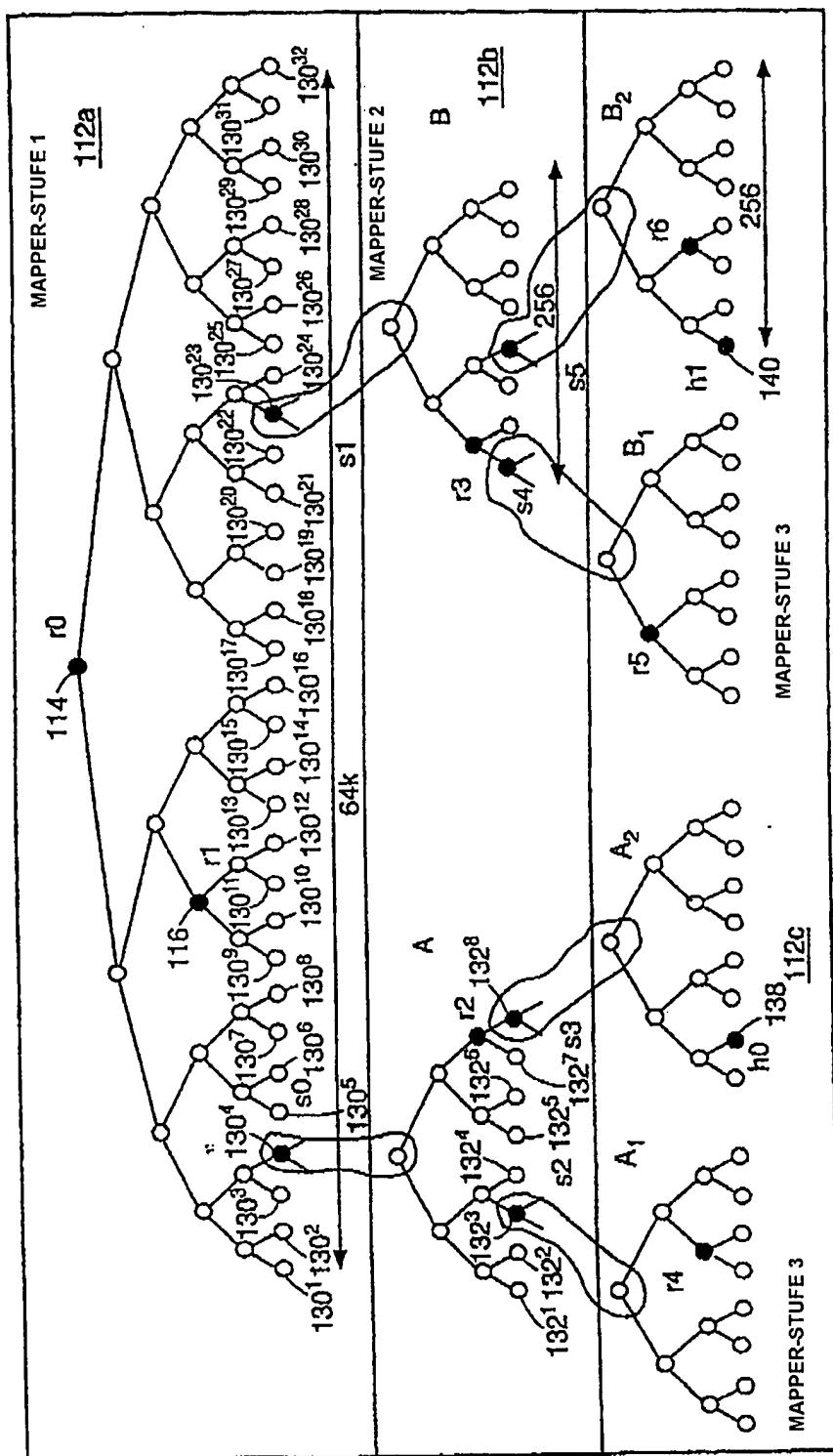


FIG. 2B

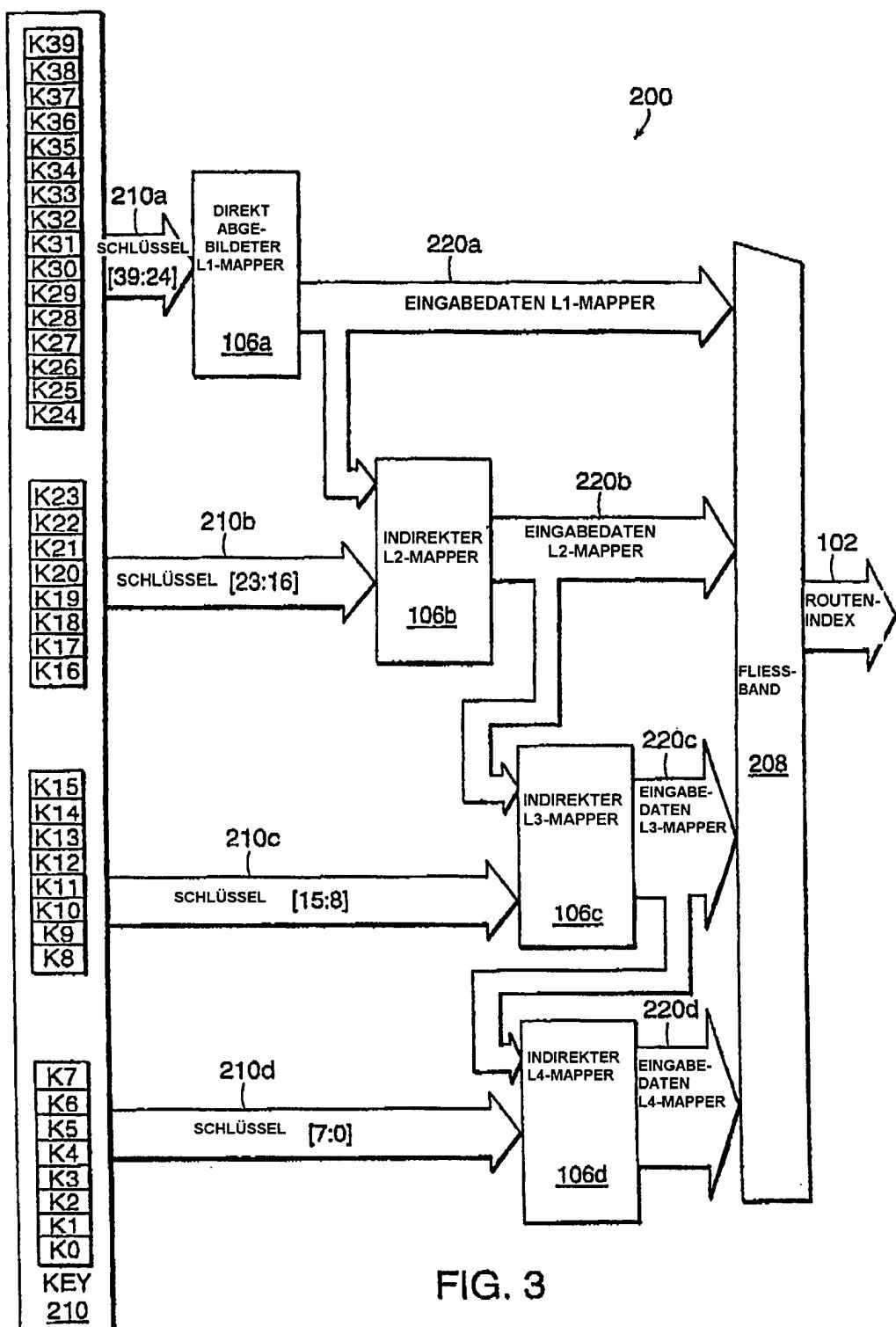


FIG. 3

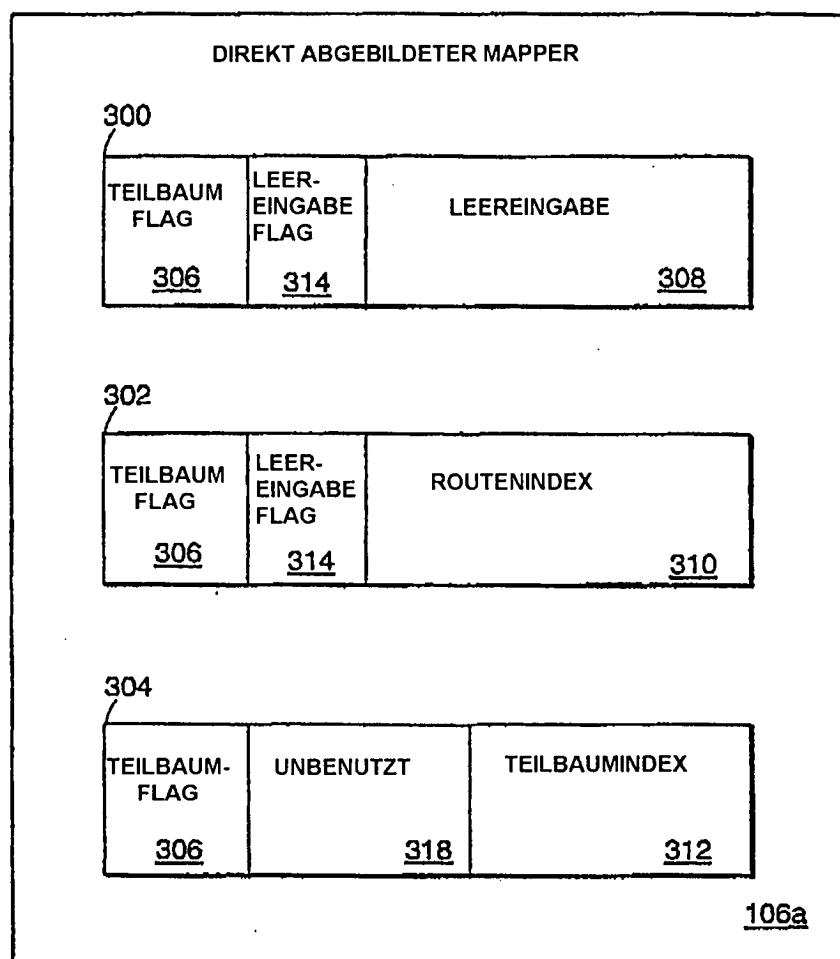


FIG. 4

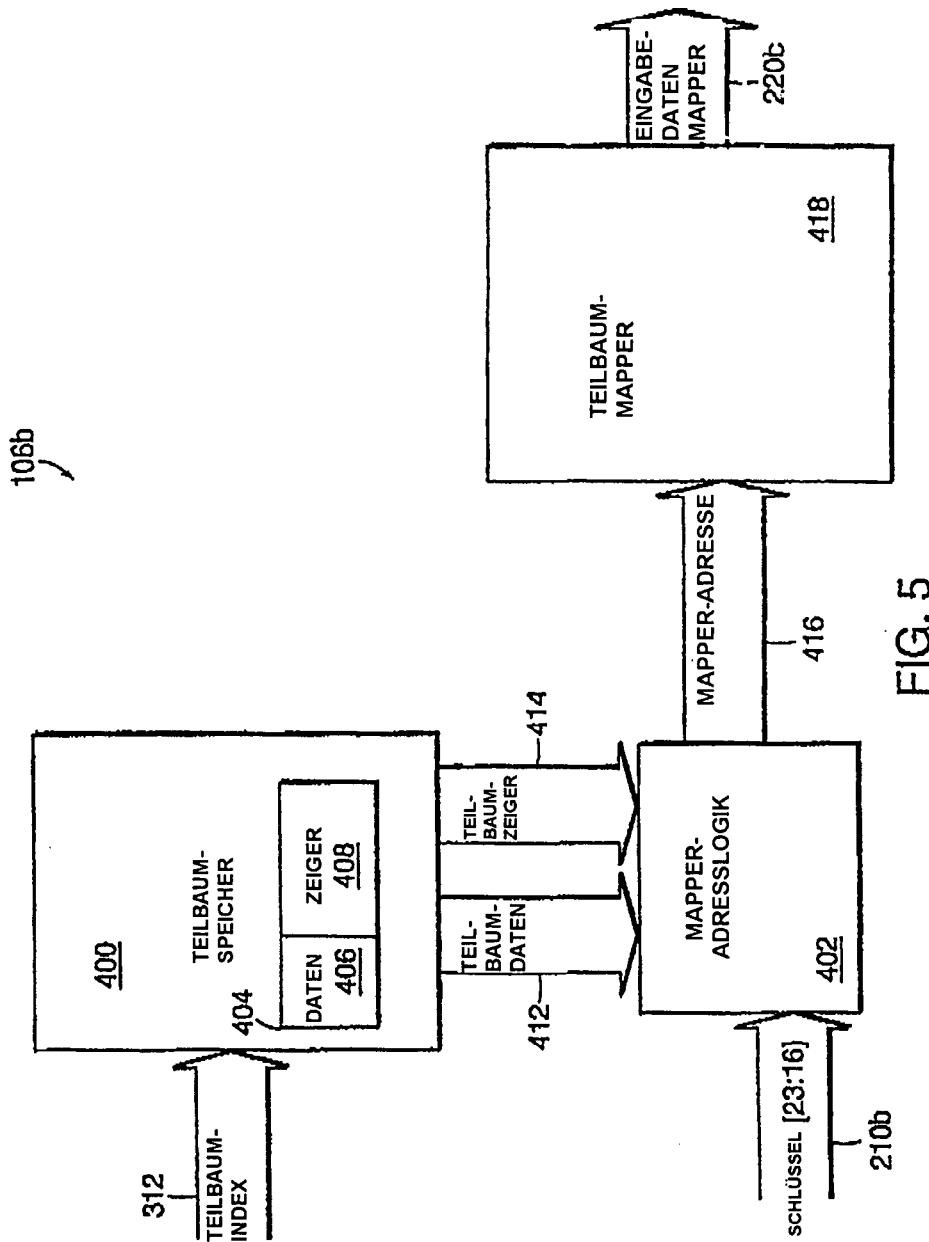


FIG. 5

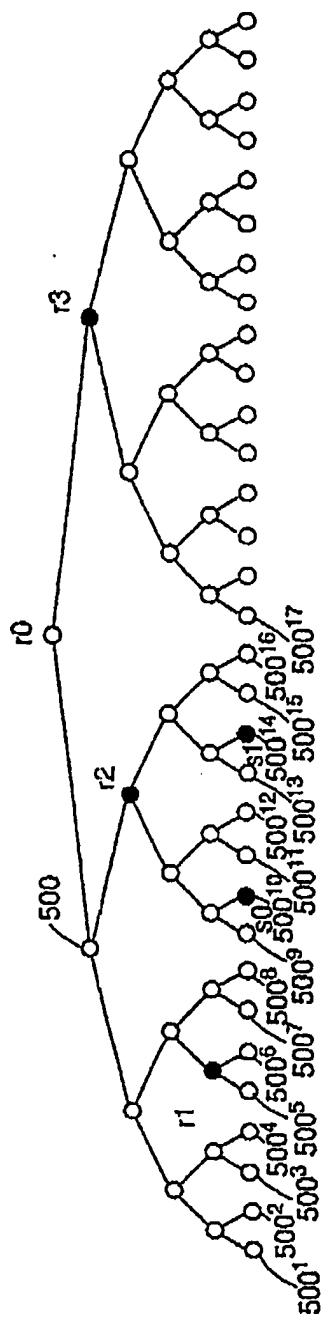


FIG. 6A

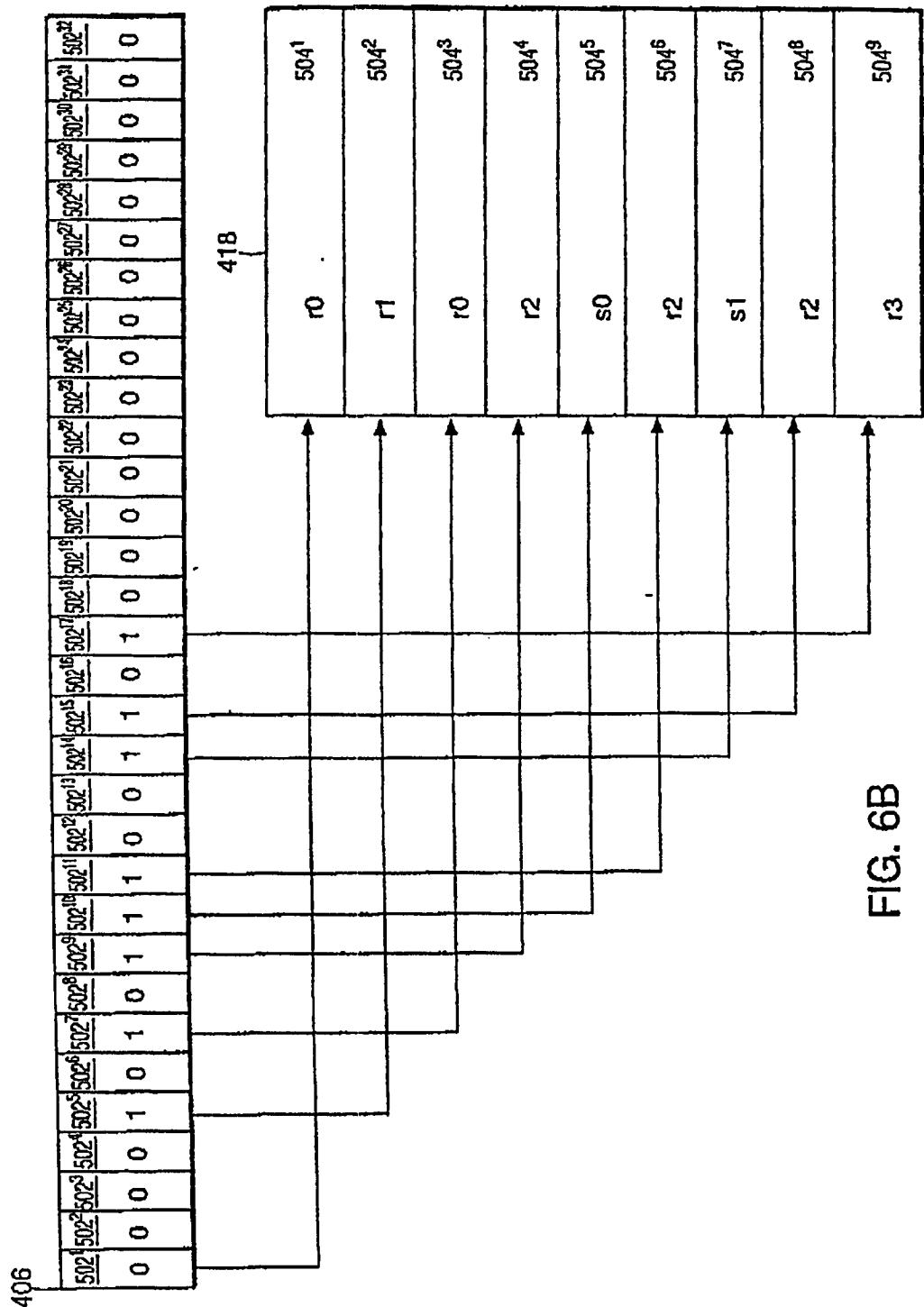


FIG. 6B

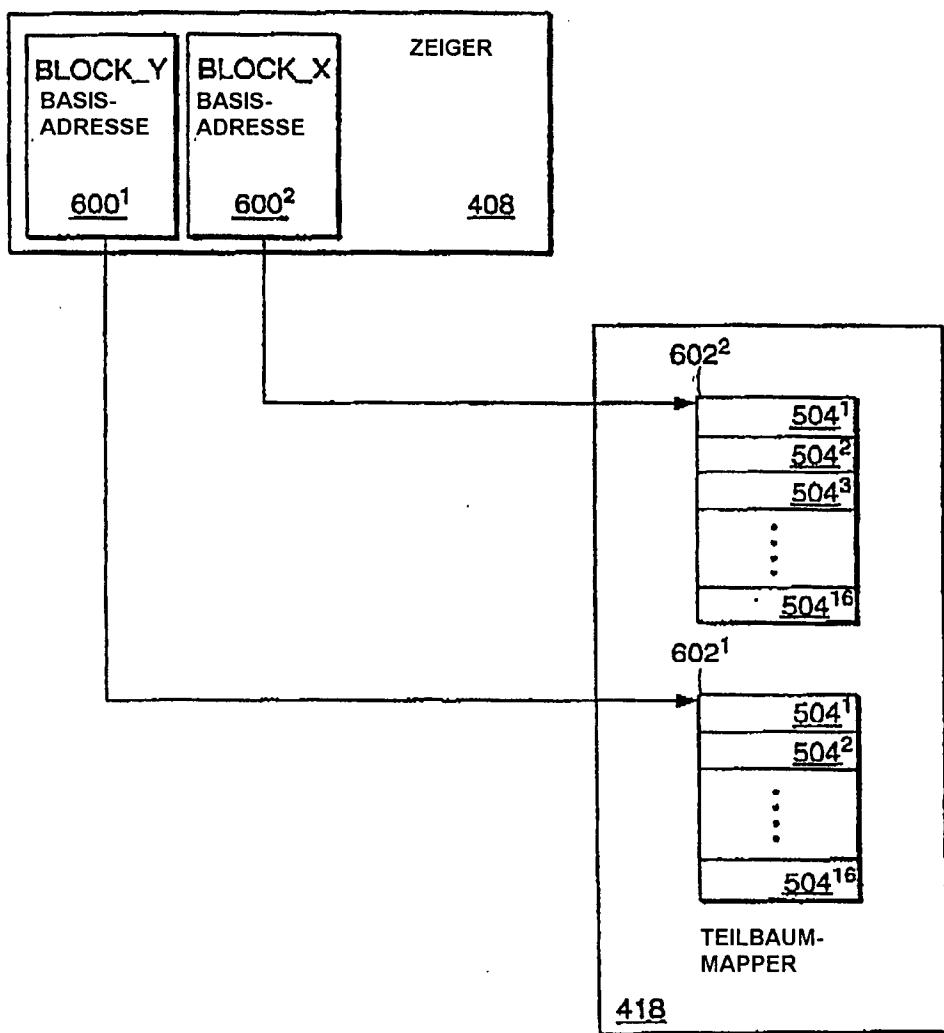


FIG. 7

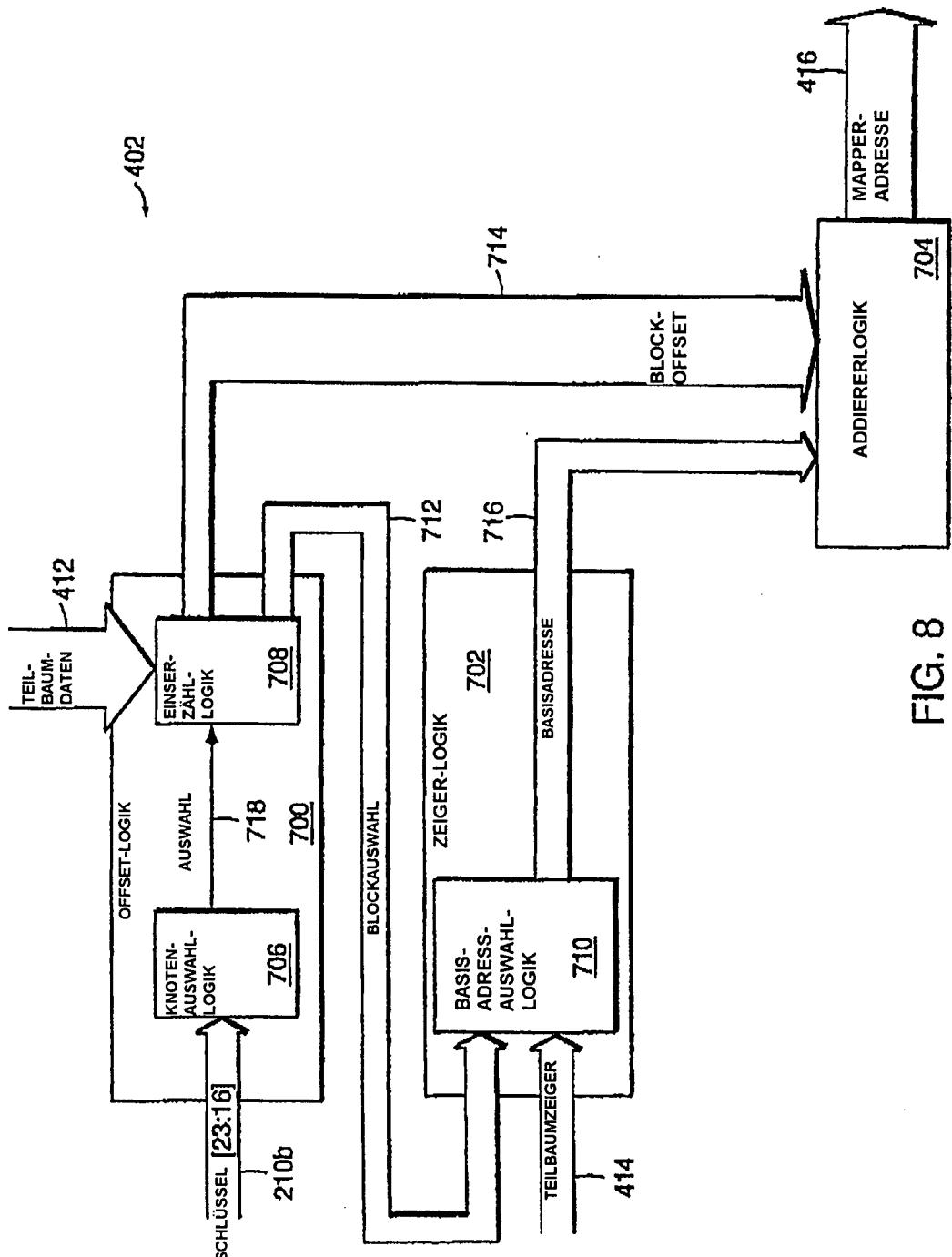


FIG. 8

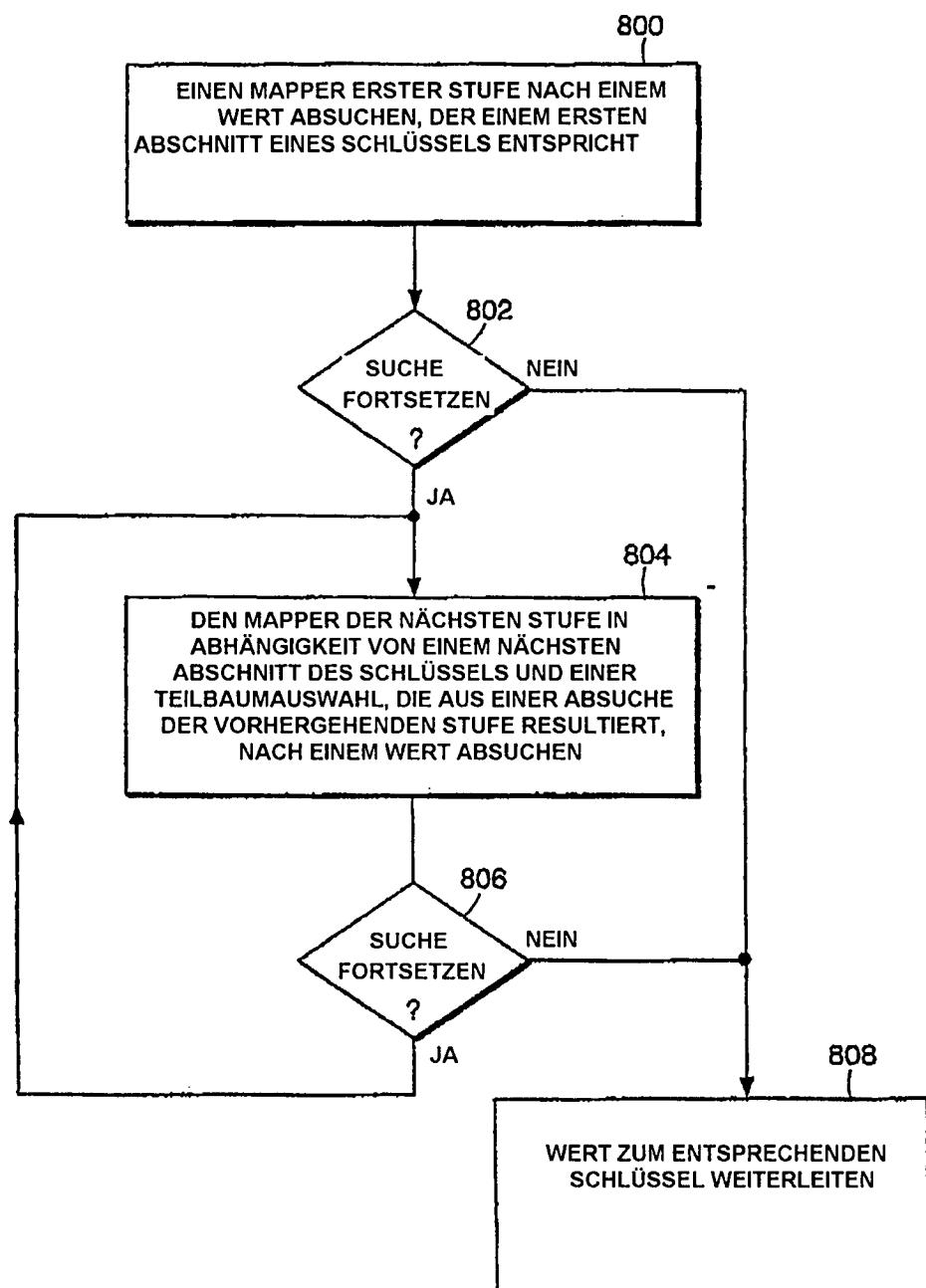


FIG. 9

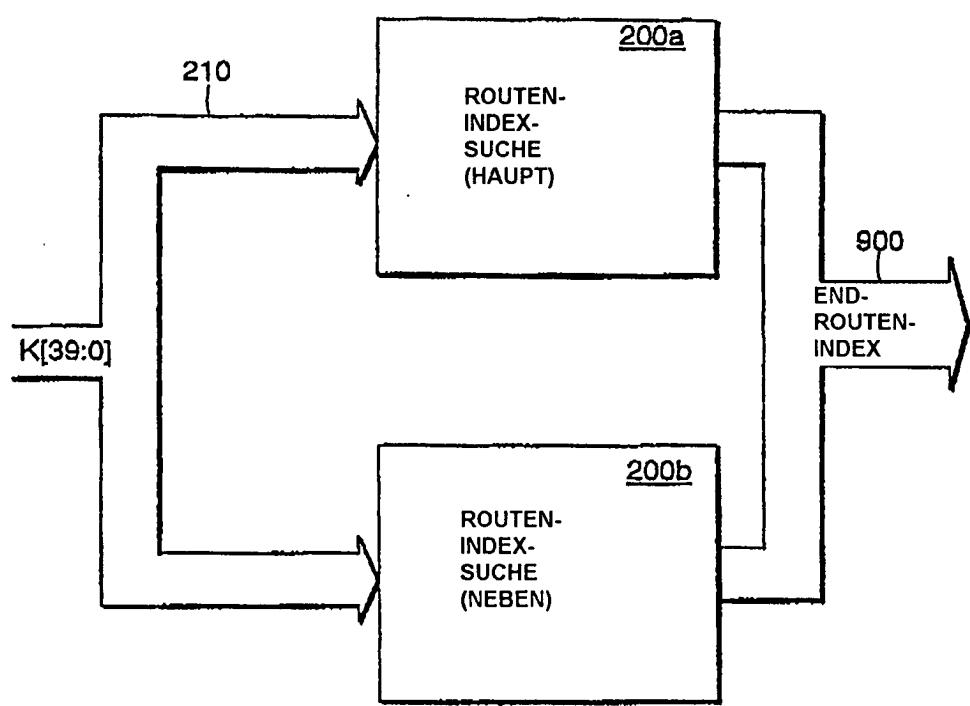


FIG. 10A

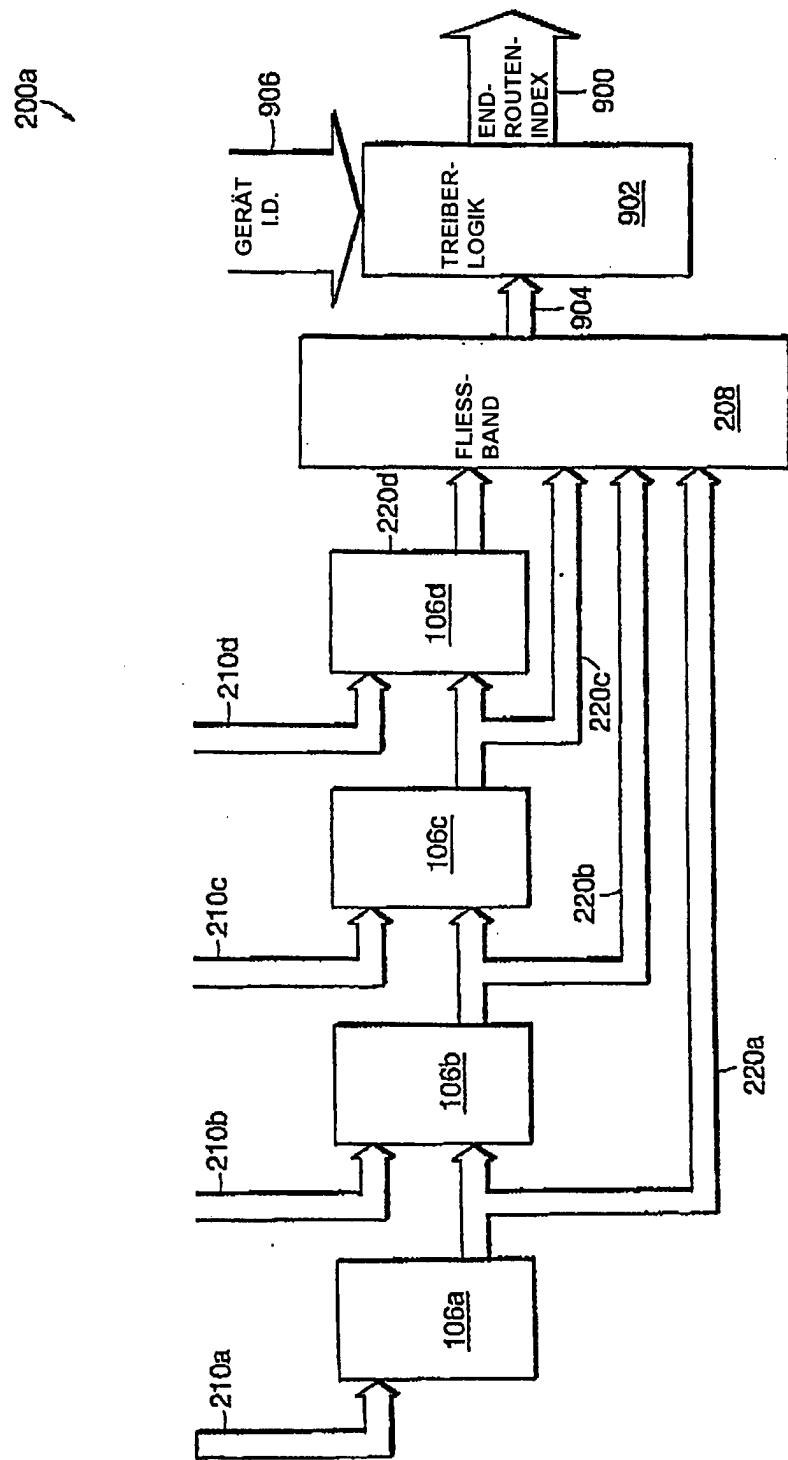


FIG. 10B

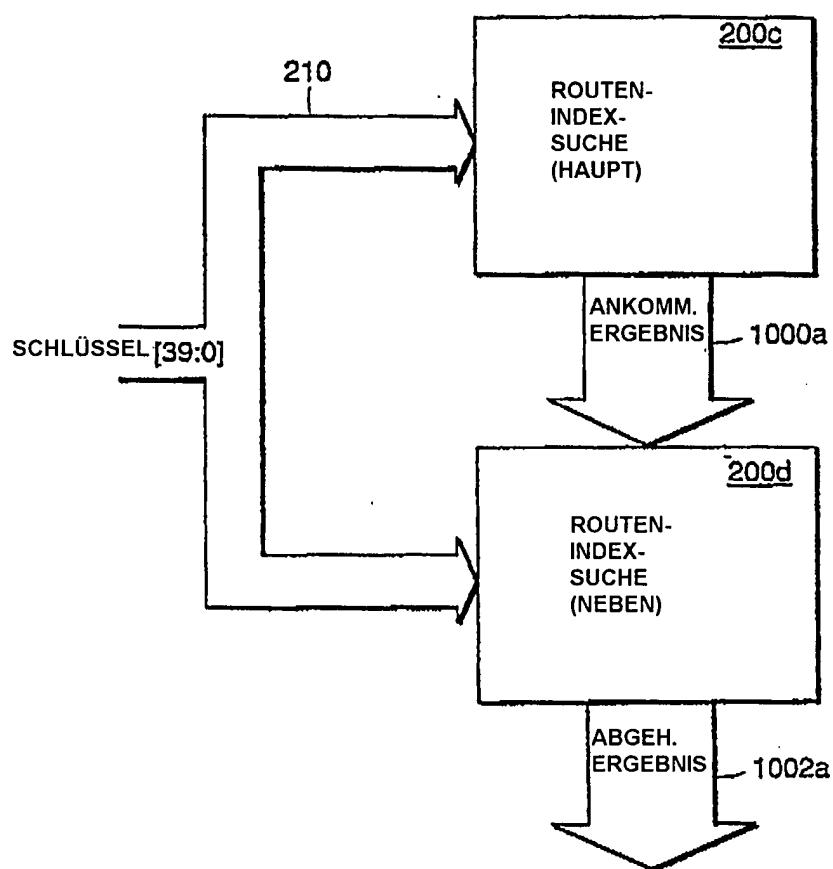


FIG. 10C

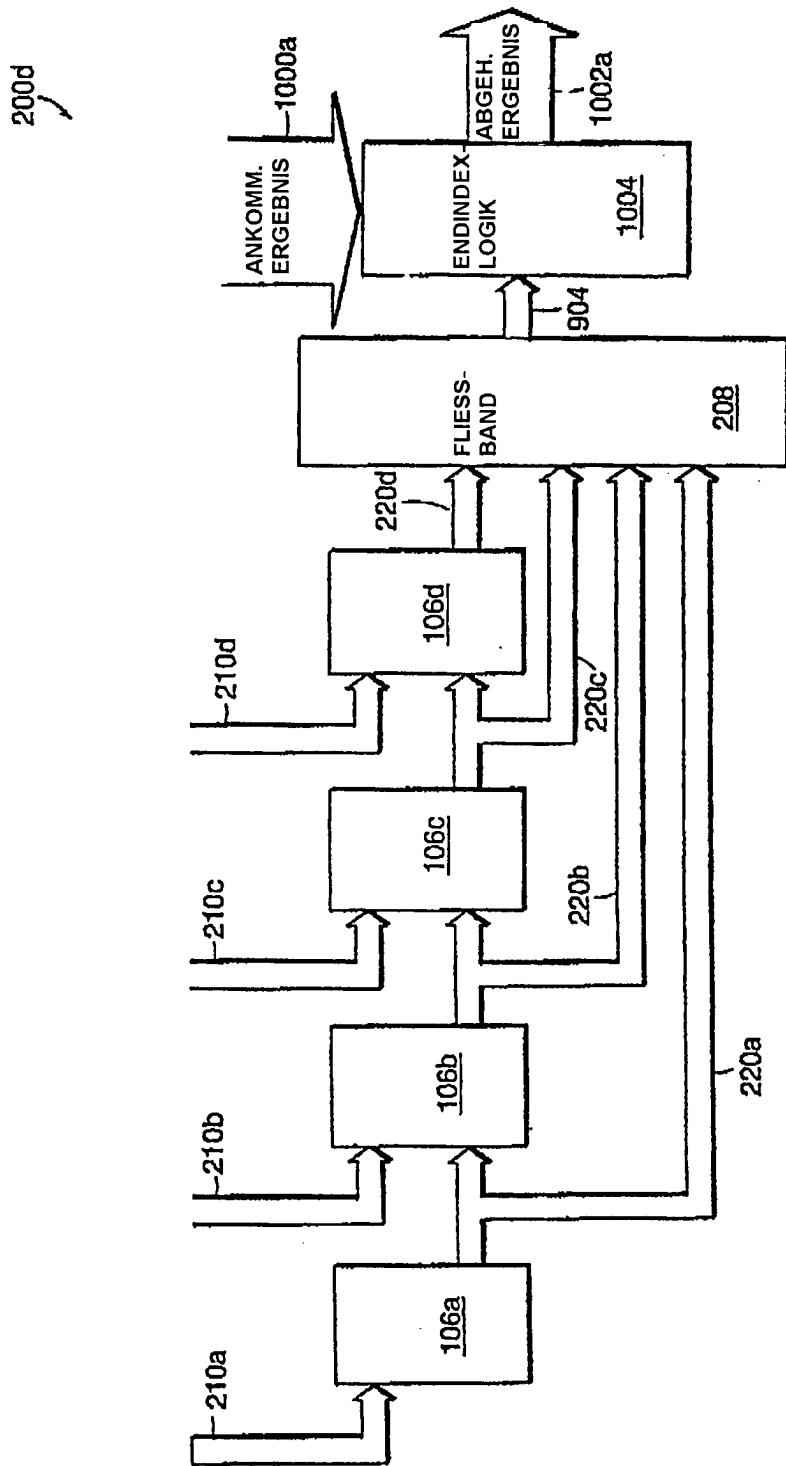


FIG. 10D

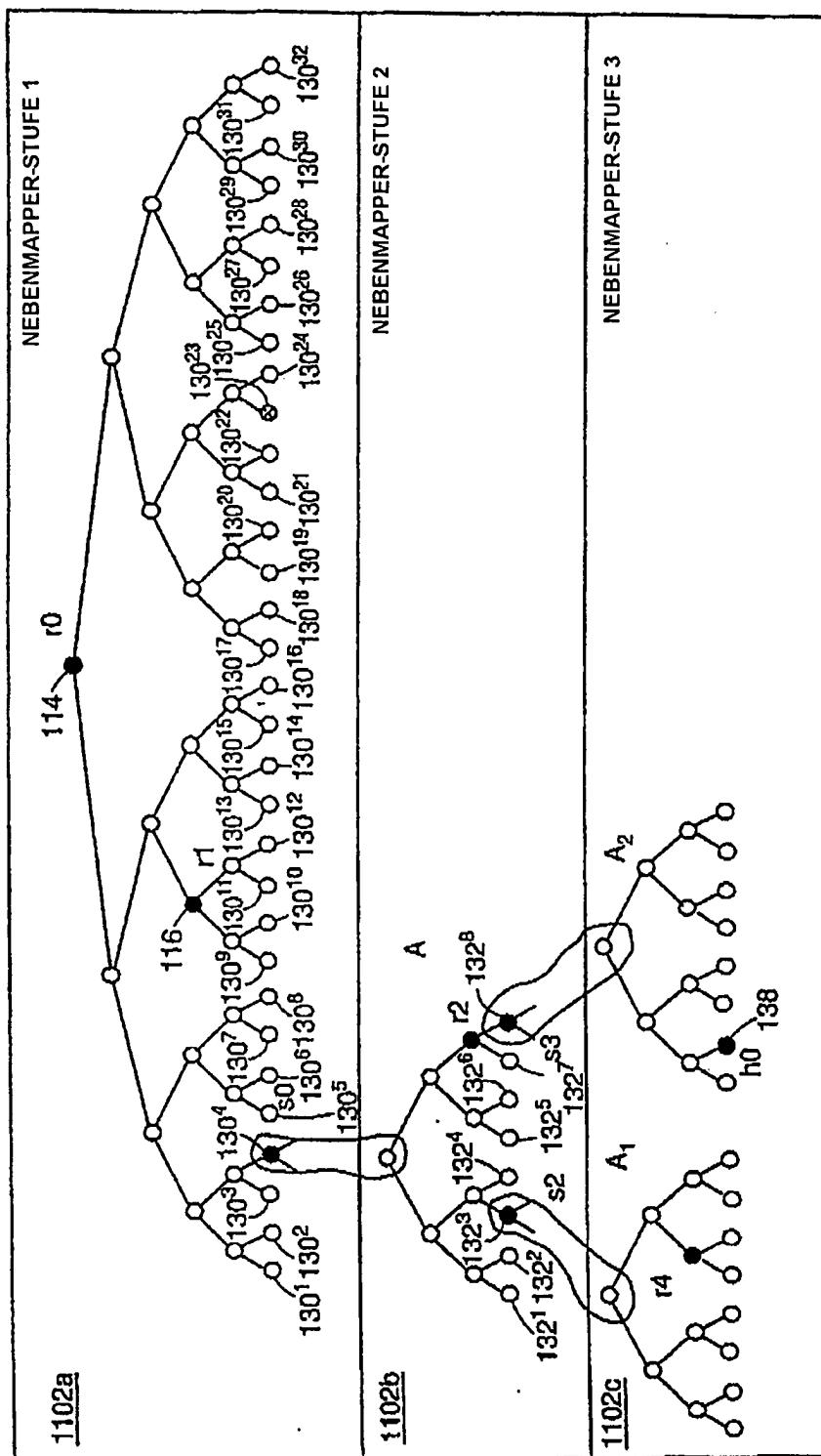


FIG. 11A

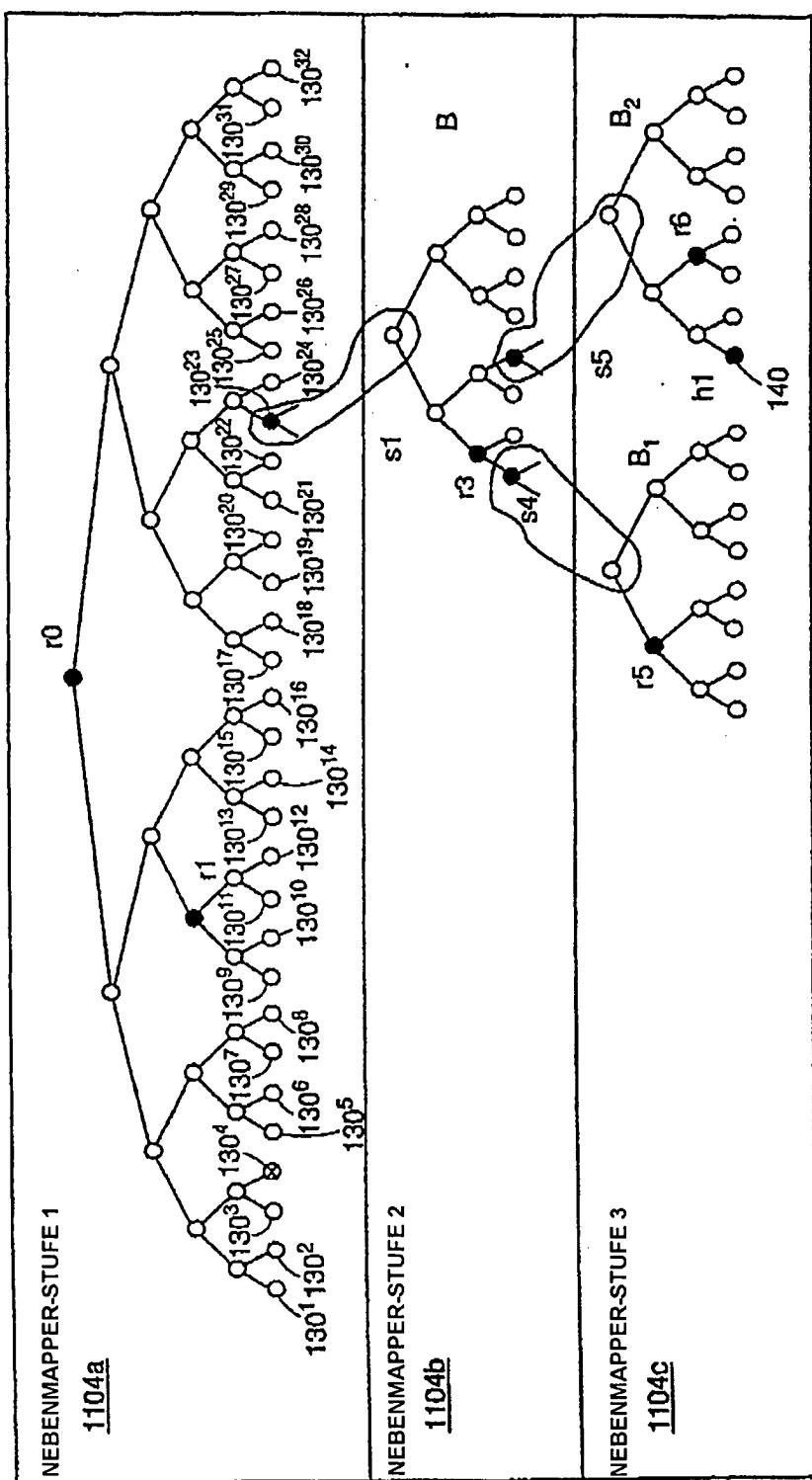


FIG. 11B

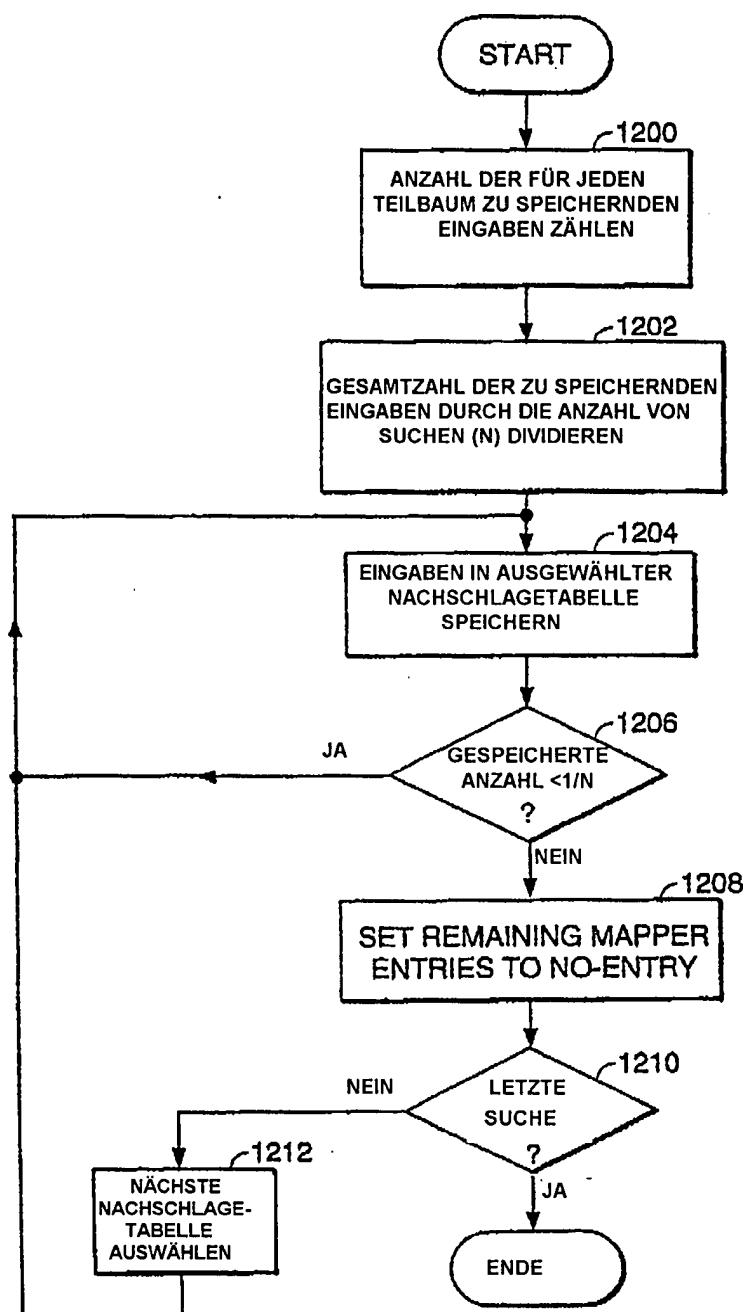


FIG. 12

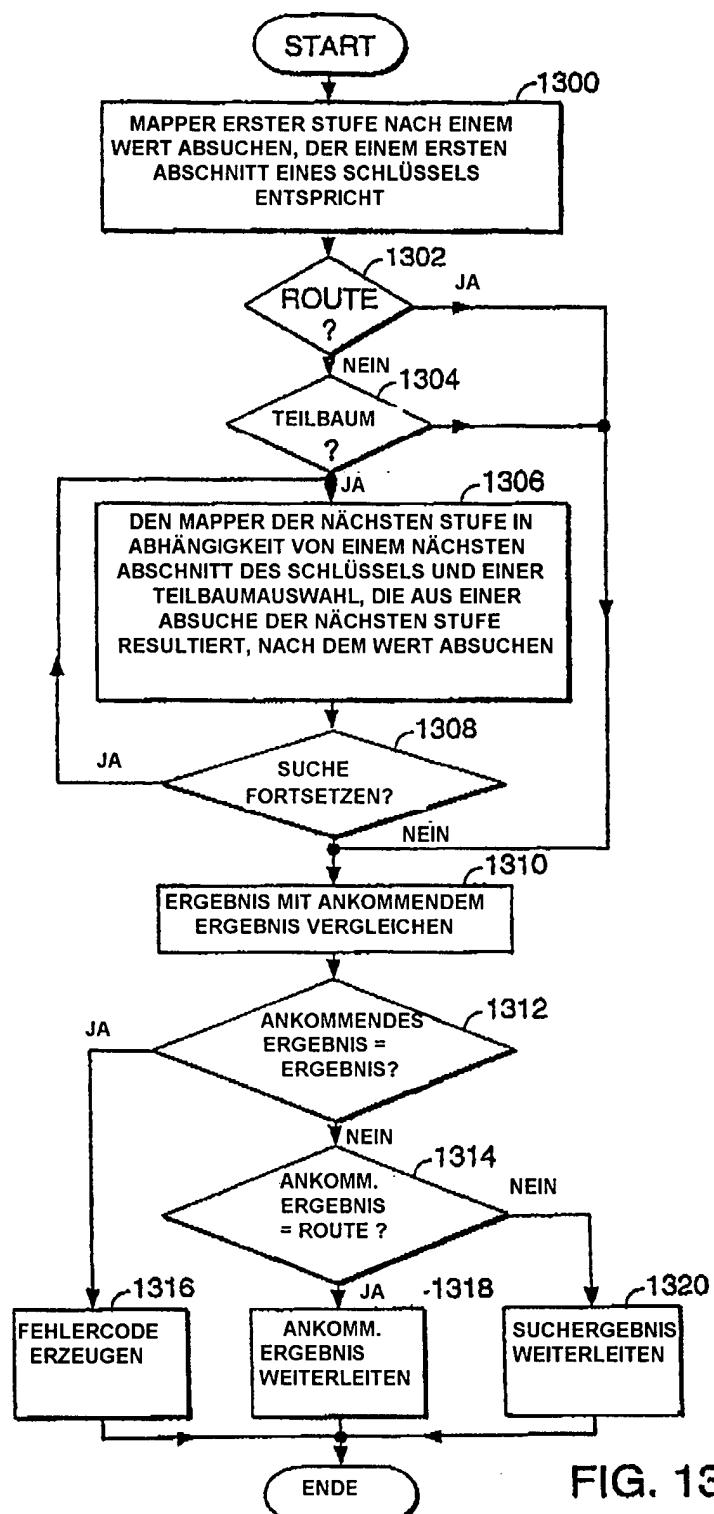


FIG. 13

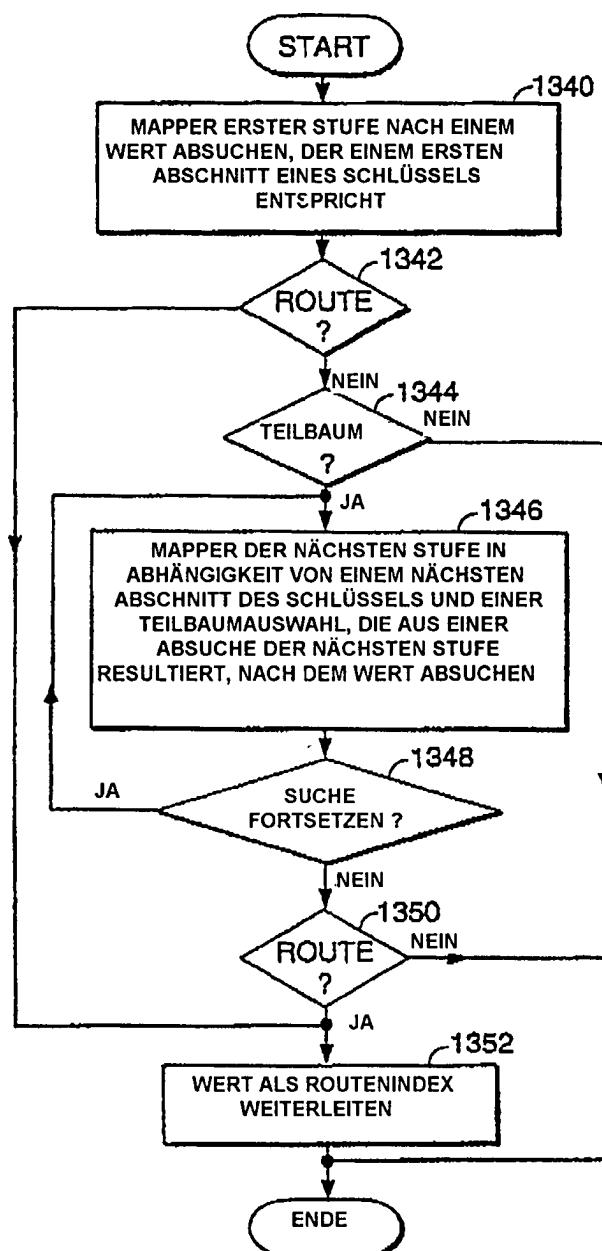


FIG. 14

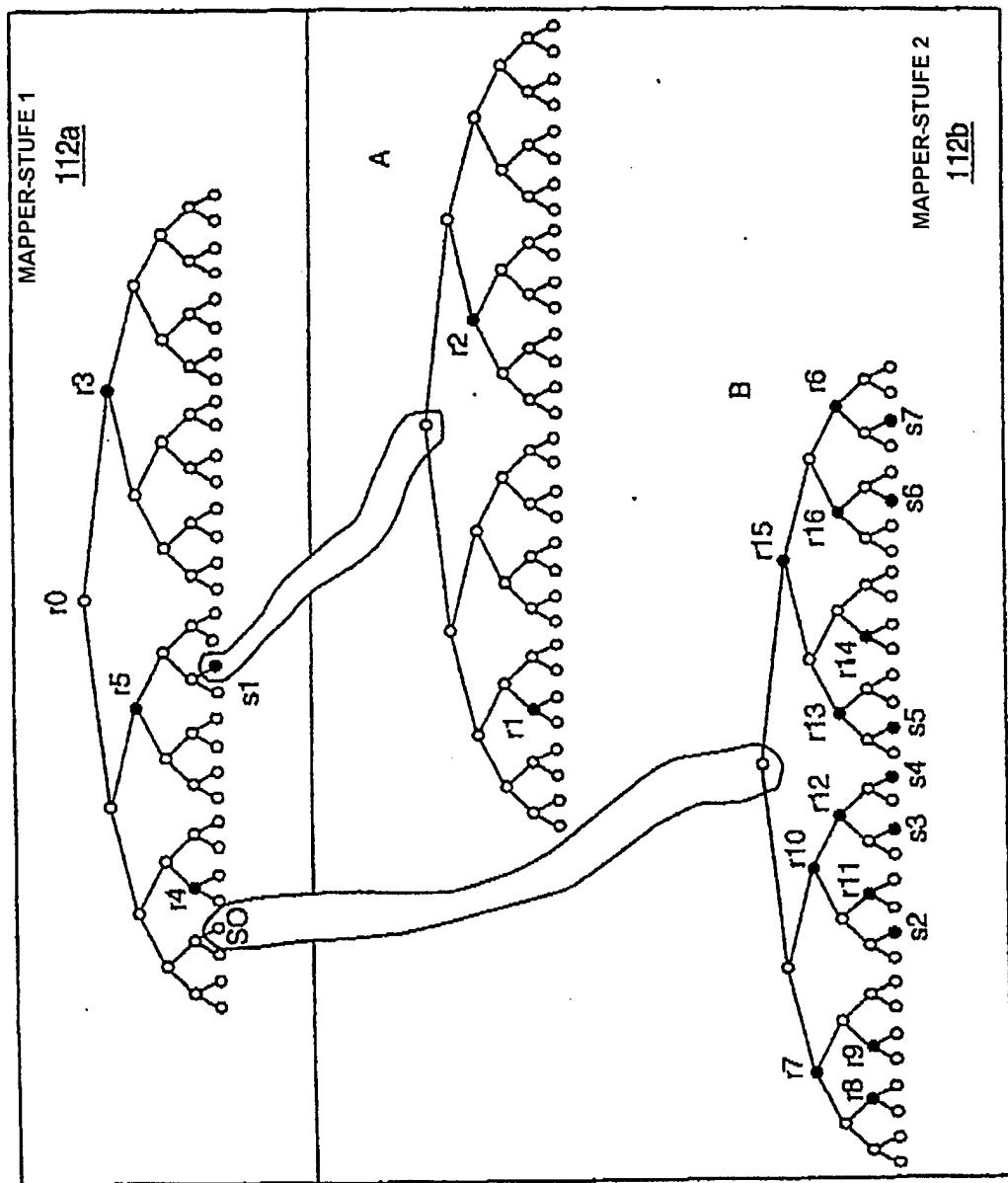


FIG. 15

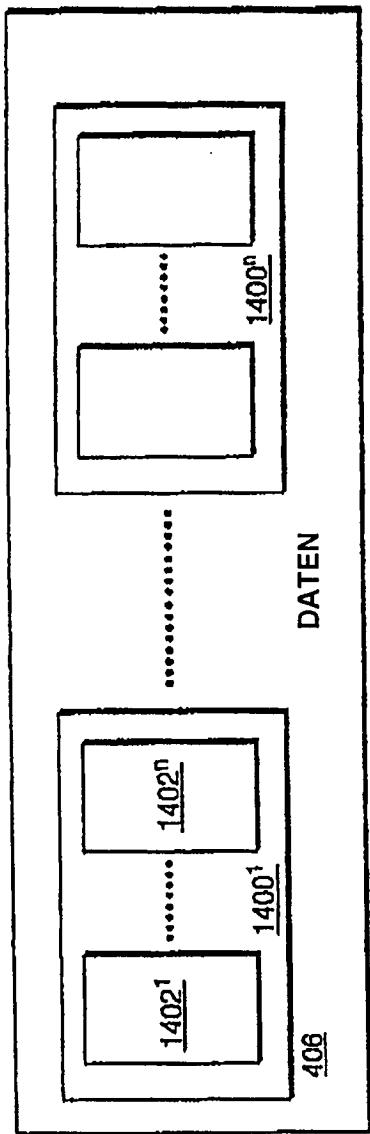


FIG. 16A

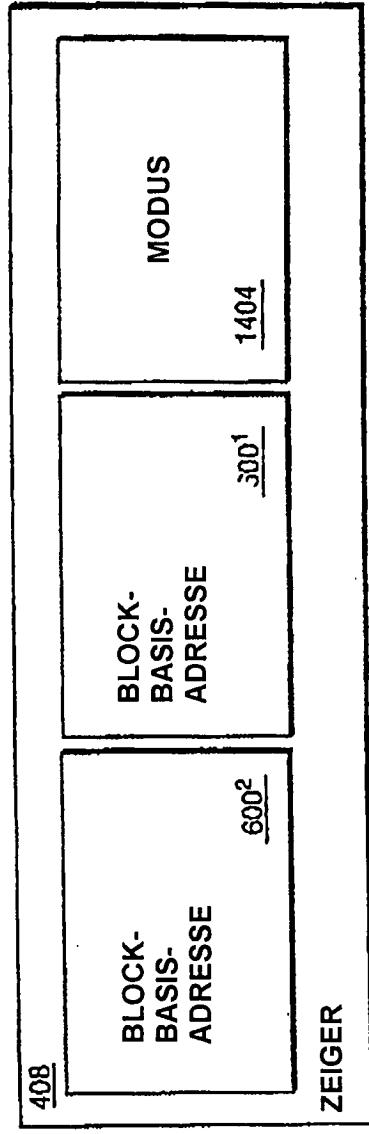


FIG. 16B

304	318			
TEILBAUM-FLAG	TYP	AUSWAHL DÜNNBESIED. TEILBAUM-DESKRIPTOR	TEILBAUM-INDEX	
<u>306</u>	<u>1406</u>	<u>1408</u>	<u>312</u>	

FIG. 16C

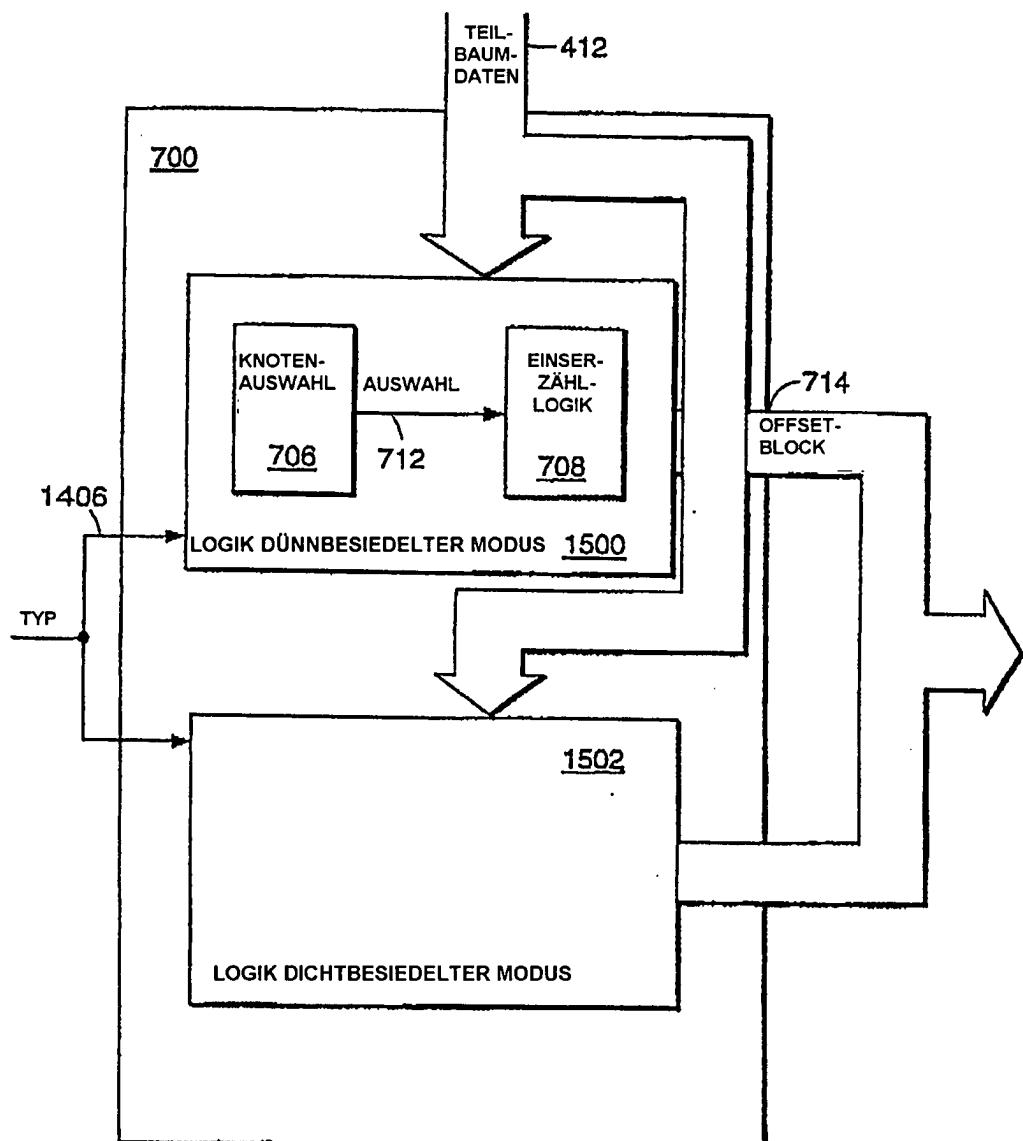


FIG. 17

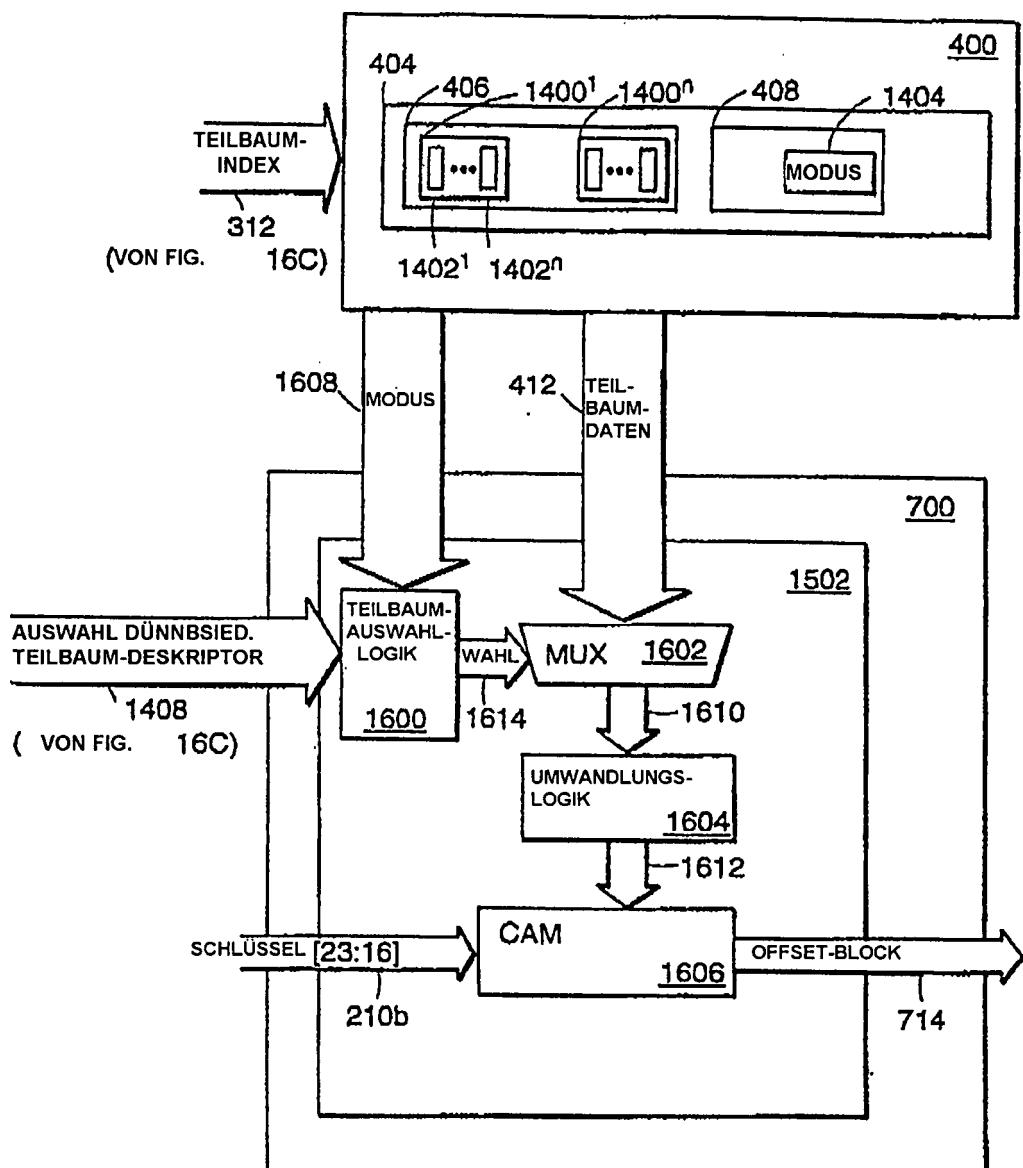


FIG. 18

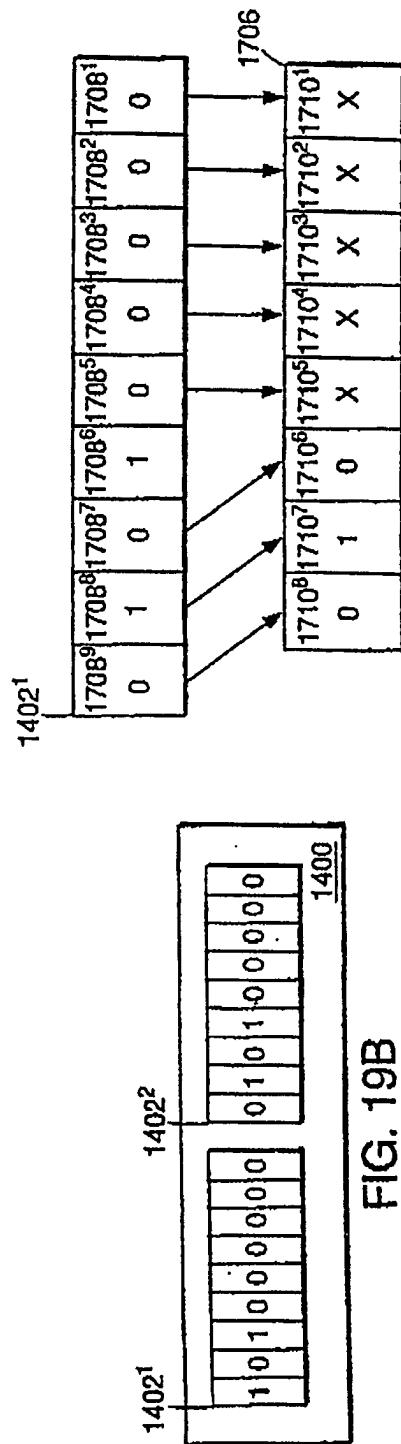
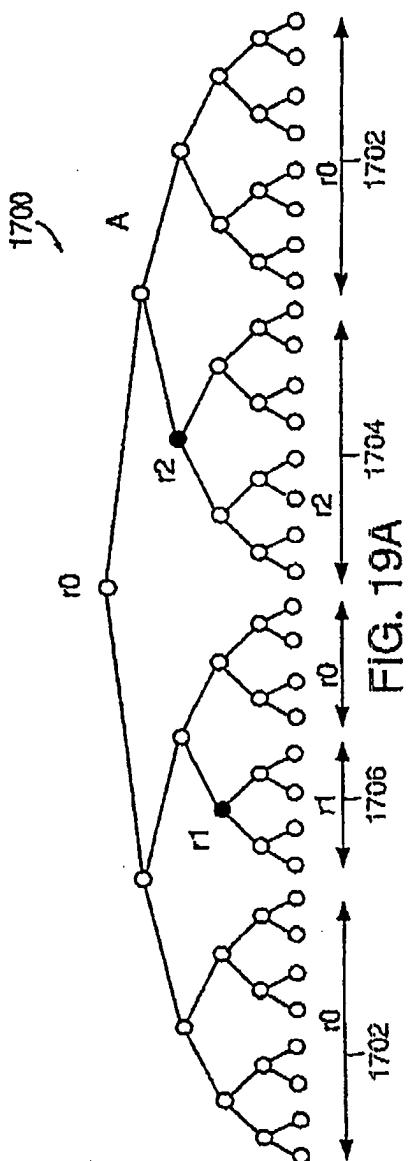


FIG. 19C

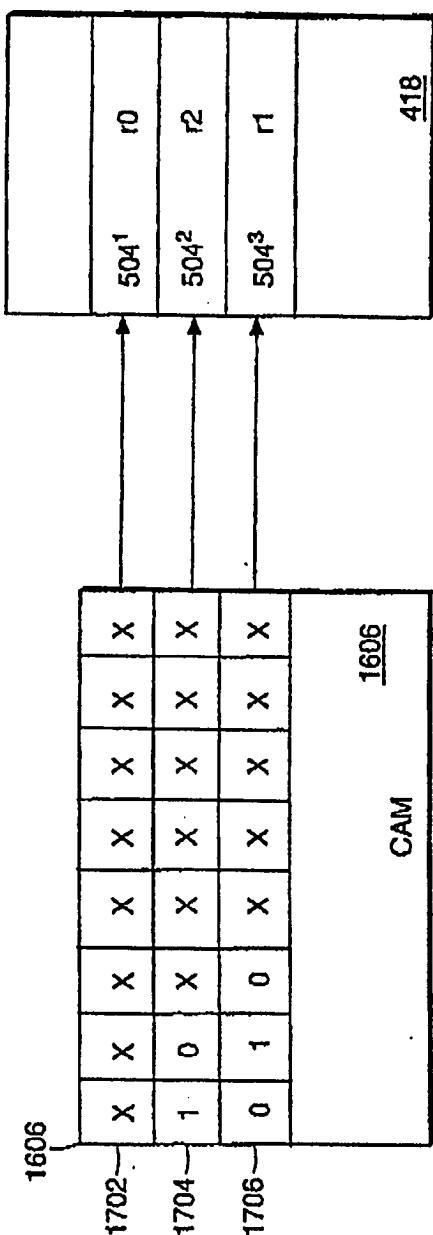


FIG. 19D

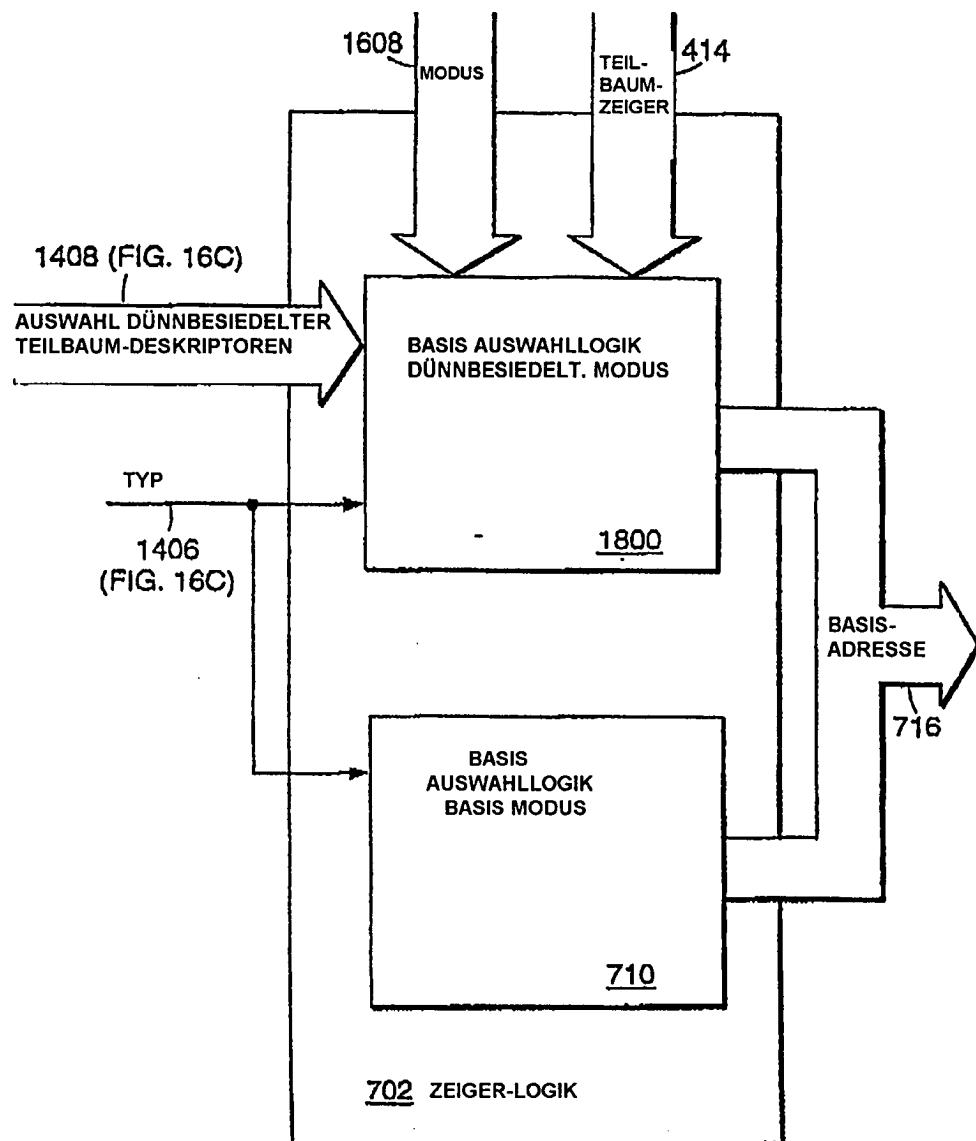


FIG. 20

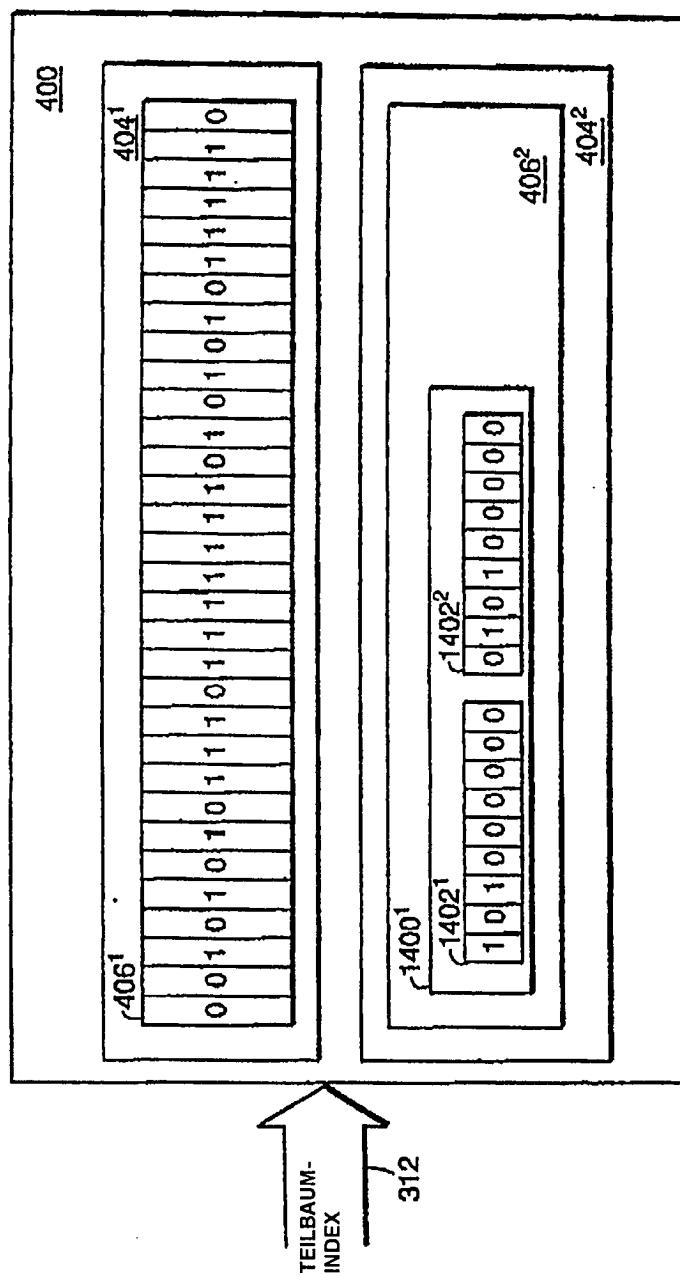
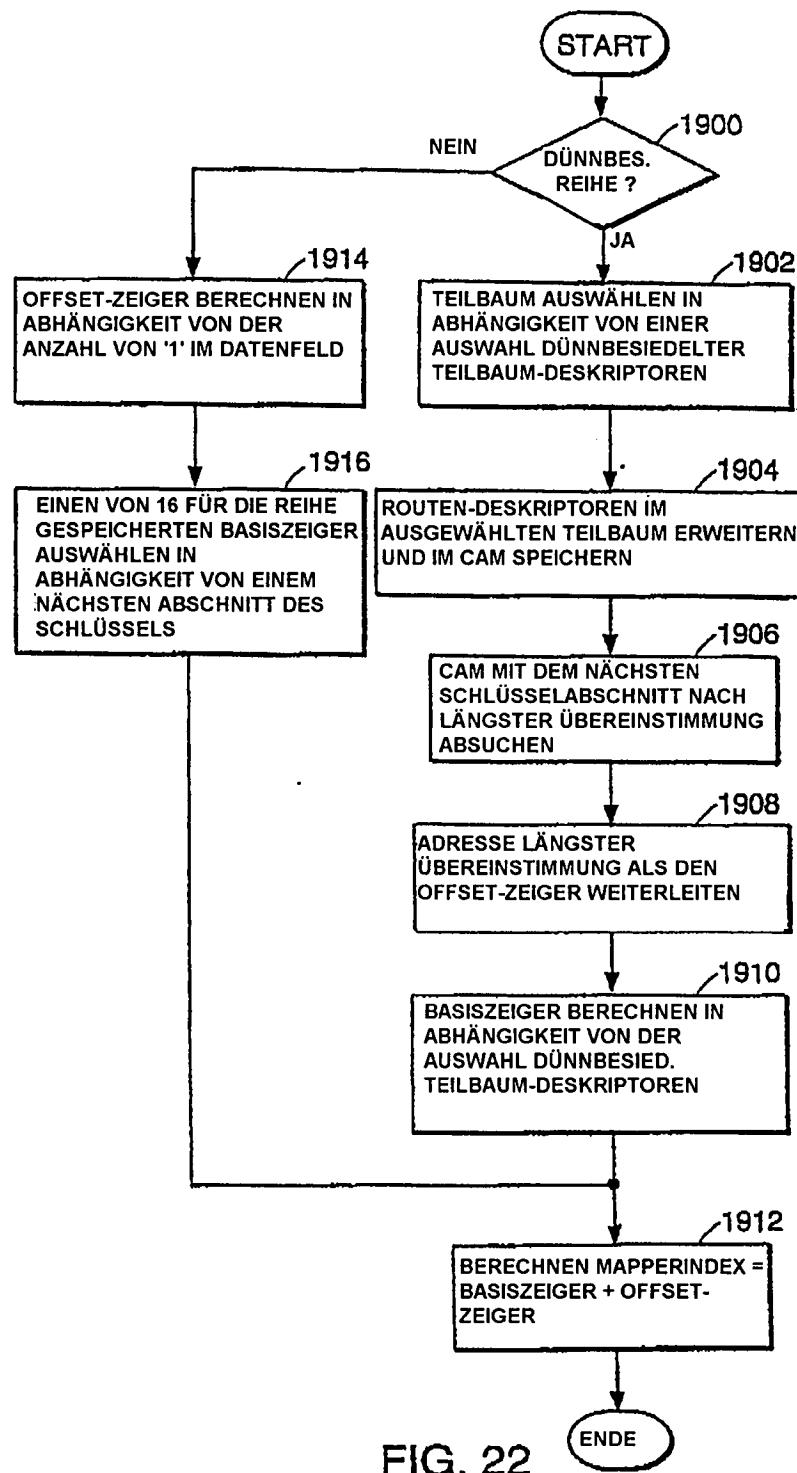


FIG. 21



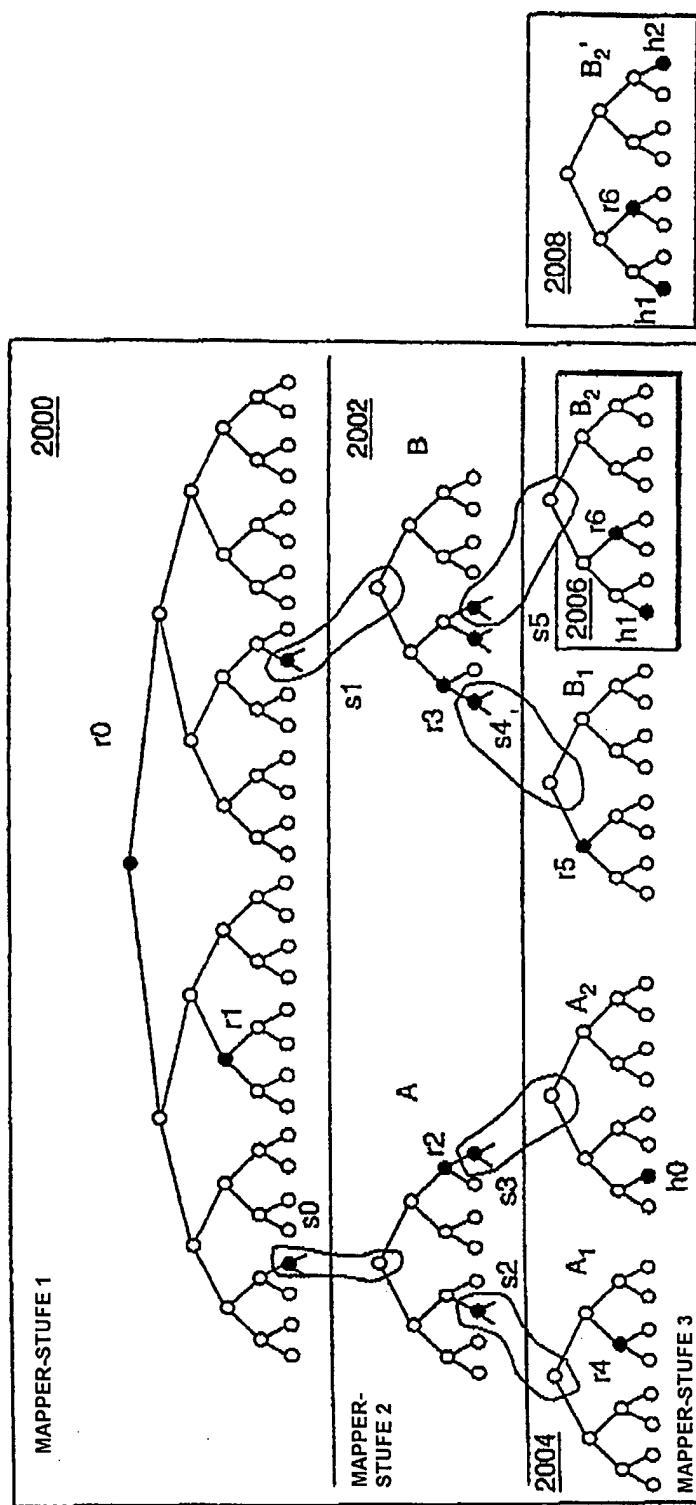


FIG. 23

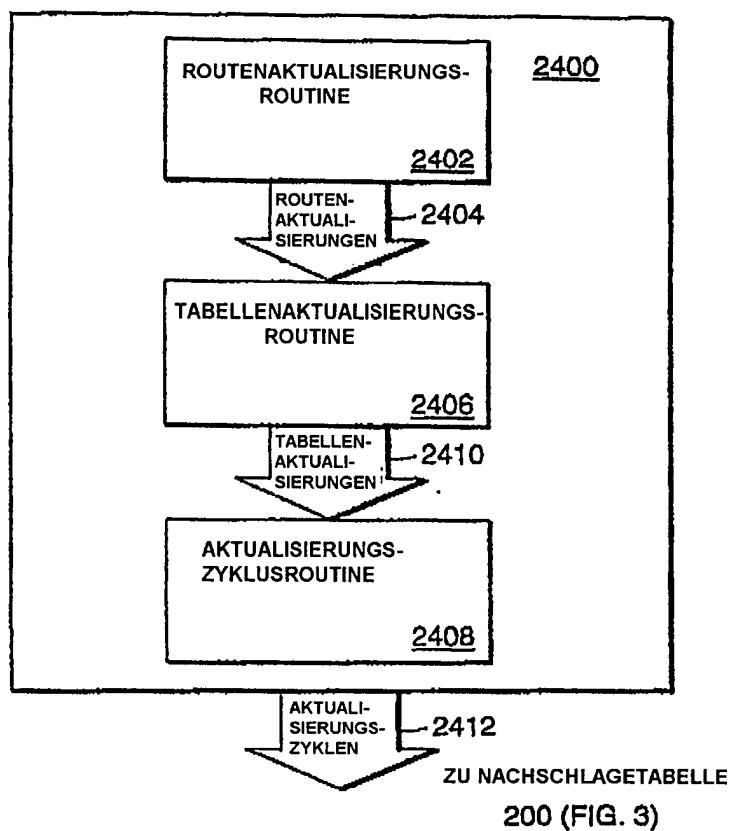


FIG. 24

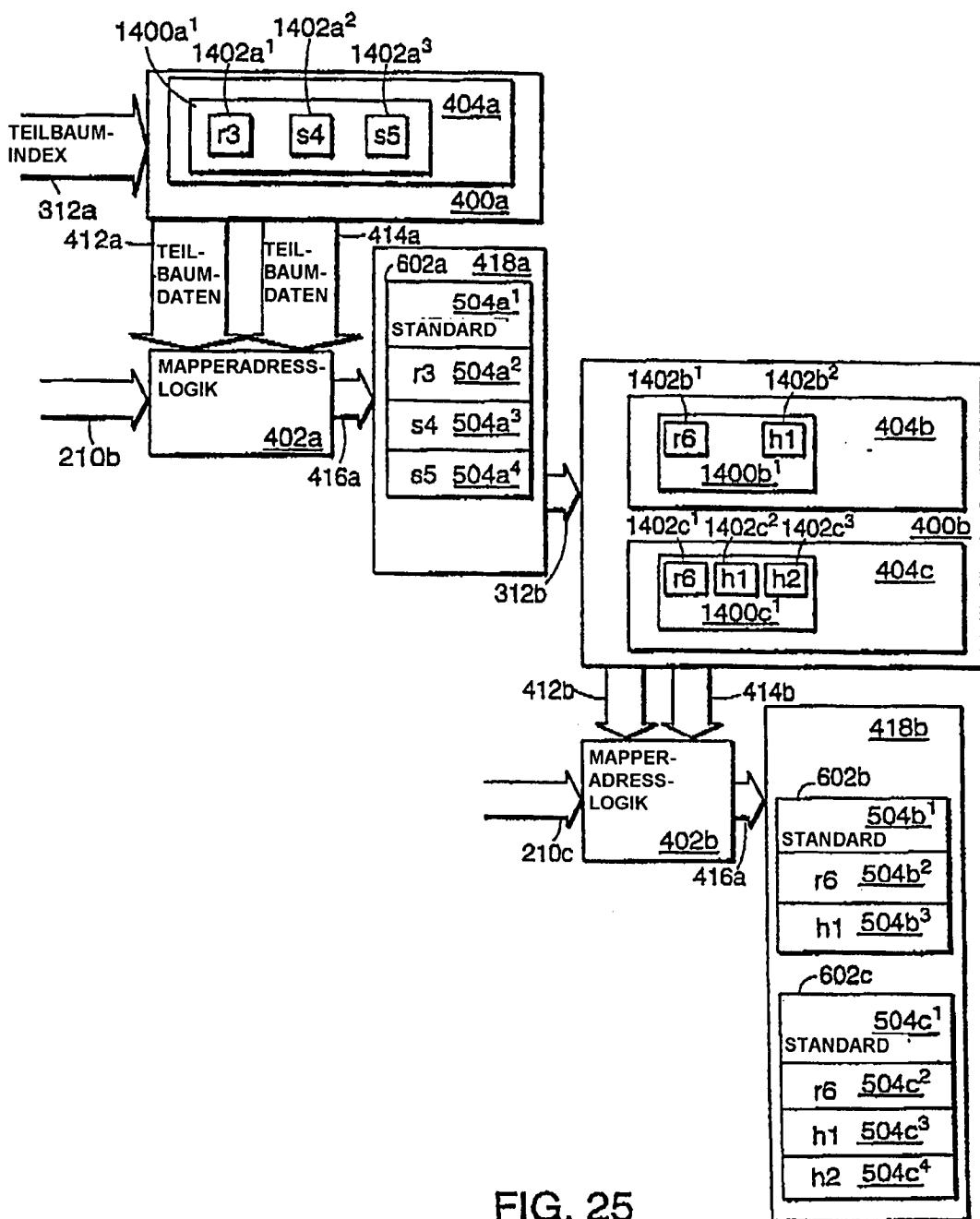


FIG. 25

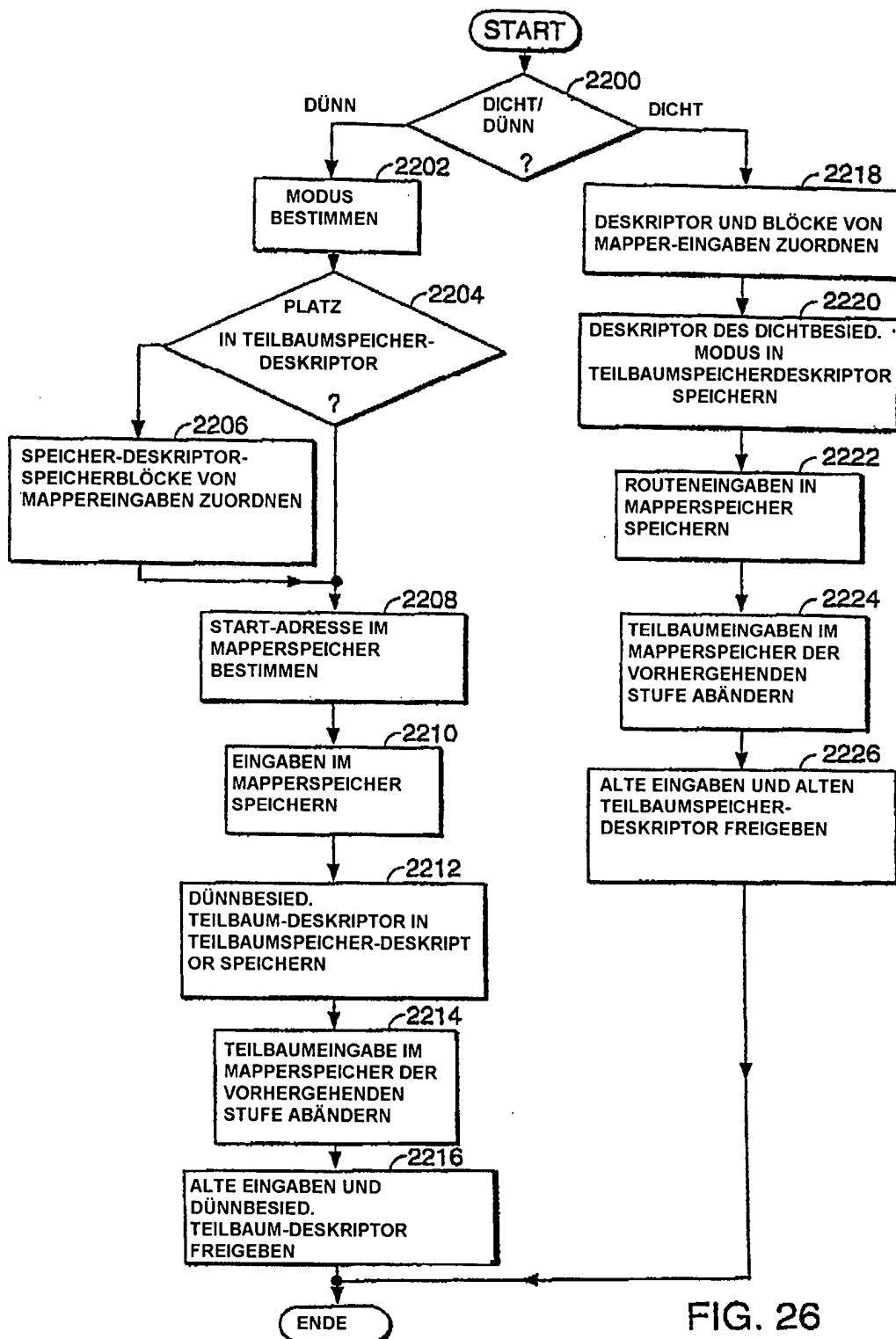


FIG. 26