



(19)  
Bundesrepublik Deutschland  
Deutsches Patent- und Markenamt

(10) **DE 600 31 712 T2 2007.09.06**

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 1 232 612 B1**

(21) Deutsches Aktenzeichen: **600 31 712.9**

(86) PCT-Aktenzeichen: **PCT/US00/30144**

(96) Europäisches Aktenzeichen: **00 980 263.8**

(87) PCT-Veröffentlichungs-Nr.: **WO 2001/037495**

(86) PCT-Anmeldetag: **17.11.2000**

(87) Veröffentlichungstag  
der PCT-Anmeldung: **25.05.2001**

(97) Erstveröffentlichung durch das EPA: **21.08.2002**

(97) Veröffentlichungstag  
der Patenterteilung beim EPA: **02.11.2006**

(47) Veröffentlichungstag im Patentblatt: **06.09.2007**

(51) Int Cl.<sup>8</sup>: **H04L 12/56 (2006.01)**

**H04L 29/12 (2006.01)**

**H04L 12/44 (2006.01)**

(30) Unionspriorität:

**166225 P 18.11.1999 US**

(73) Patentinhaber:

**Broadcom Corp., Irvine, Calif., US**

(74) Vertreter:

**Bosch, Graf von Stosch, Jehle  
Patentanwalts-gesellschaft mbH, 80639 München**

(84) Benannte Vertragsstaaten:

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

(72) Erfinder:

**MALALUR, Govind, Fremont, CA 94536, US**

(54) Bezeichnung: **TABELLEN-NACHSCHLAGE-MECHANISMUS ZUR ADRESSAUFLÖSUNG IN EINER PAKET-NETZWERKVERMITTLUNG**

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

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

**Beschreibung**

## VERWANDTE ANMELDUNGEN:

**[0001]** Die vorliegende Anmeldung beansprucht die Priorität der vorläufigen US-Patentanmeldung (U.S. Provisional Patent Application) Nr. 60/166,225 vom 18. November 1999, veröffentlicht als Internationale Anmeldung WO 01/37495 am 25. Mai 2001.

## HINTERGRUND DER ERFINDUNG

## GEBIET DER ERFINDUNG:

**[0002]** Die vorliegende Erfindung ist auf ein Verfahren und eine Vorrichtung für einen Tabellennachschlageindex gerichtet, der schnellen und effizienten Zugriff auf eine Tabelle, z.B. eine Adresstabelle, bereitstellt. Insbesondere dient der Tabellennachschlageindex für die Übertragung von Datenpaketen in einer Netzwerkvermittlung.

## BESCHREIBUNG DES STANDS DER TECHNIK:

**[0003]** In der Welt der Datenübertragung ist es notwendig, Daten so schnell wie möglich zu übertragen, um nahezu unverzögerten Zugriff auf Informationen zu haben.

**[0004]** Tabellennachschlagen wird häufig verwendet, um Informationen zu indizieren, damit schnell auf häufig benötigte Daten zugegriffen werden kann. Bei der Netzwerkvermittlung ist es beispielsweise zwingend, dass, wenn Daten in einem Port empfangen werden, der geeignete Port für die Ausgabe identifiziert wird, um Daten zu übertragen. Daher wird Tabellennachschlagen verwendet, um Informationen bezüglich ankommender Daten zu speichern, die die ankommenden Daten auf einen geeigneten Port für die Ausgabe leiten.

**[0005]** Auf dem Gebiet des Tabellennachschlagens werden mehrere Verfahren verwendet, um auf Informationen zuzugreifen. Ein Verfahren ist ein linearer Index.

**[0006]** **Fig. 1B** zeigt eine 64K Tabelle 100, die linear indiziert wird. Der Index **105** für eine 64K Tabelle muss 16 Bit lang sein ( $2^{16} = 64K$ ), um auf Einträge in der Tabelle linear zuzugreifen. Da der Index linear auf Einträge in der Tabelle zugreift, liegt eine Eins-zu-Eins-Entsprechung zwischen jedem Index und jedem Tabelleneintrag vor (d.h. jeder Index hat einen entsprechenden Tabelleneintrag).

**[0007]** **Fig. 1C** ist eine Darstellung eines linearen Indexes in einer 64K Tabelle. Die Indices I(1), I(2), I(3) ... werden jeweils linear in die 64K Tabelle indiziert, so dass eine Eins-zu-Eins-Entsprechung zwischen jedem Index und jedem Tabelleneintrag entsteht.

Wenn die Indices I(1), I(2), I(3) ... jeweils 16 Bit lang sind, kann in 16 Taktzyklen unter Verwendung standardmäßigen binären Durchsuchens ein Eintrag gefunden werden. Wenn jedoch durch Einfügen eines Eintrags oder Löschen eines Eintrags eine neue Adresse gelernt werden muss, dauert es lange, die Adresse aus der Tabelle zu löschen oder in sie einzufügen, weil der Index nach jedem Einfügen und jedem Löschen sortiert werden muss. Im ungünstigsten Fall muss jeder der 16 Bit Adress-Indices bewegt und sortiert werden, was zeitaufwendig ist. Dieses Sortieren verschlechtert die Leistung der Vermittlung in hohem Maße. Wenn die neue Adresse in die Tabelle eingefügt wird, müssen Adressen auf und/oder ab bewegt werden, um Platz für die neue Adresse zu schaffen. Daher verschlechtert sich die Leistung der Vermittlung mit dem Anwachsen der Tabellengröße.

**[0008]** Andere Verfahren verwenden Zeiger, Hash-Funktionen oder Baum-Funktionen, um das Problem zu beheben. Die meisten dieser Verfahren haben, wenn sie implementiert sind, eine „schlechte“ Leistung im ungünstigsten Fall – insbesondere für Einfüg- und Löschoperationen. Wie oben angemerkt, verschlechtert sich die Leistung mit dem Anwachsen der Tabellengröße. Einige der Hauptgründe für die schlechten Ergebnisse sind, dass die Tabellengröße die Leistung unmittelbar beeinflusst und die Verfahren spezielle Speicher- und Datenstrukturen benötigen.

**[0009]** Die US-Patentschrift US-A-5 852 607 betrifft eine Netzwerkvermittlung eines Datenkommunikationsnetzes, die Informationen, z.B. Datenpakete, zwischen Einheiten des Netzes überträgt. Eine Netzeinheit kann aus einer beliebigen Vorrichtung bestehen, die Datenpakete über derartige Medien überträgt oder empfängt. Das Netz ist physikalisch durch lokale Netze oder „LANs“ segmentiert, die die verschiedenen Netzwerkeinheiten mit der Vermittlung verbinden. Bei Netzen, die VLANs aufweisen, umfassen die kompletten Quell- und Zielinformationen, die zu übertragen sind, die Quelladresse, die Y-Bit-Zieladresse und ebenso einen V-Bit VLAN Kennzeichner. Der VLAN Kennzeichner kann in dem Paket enthalten sein oder durch den Port bestimmt werden, durch den das Paket in die Vermittlung eintritt. Der VLAN Kennzeichner ist eine wesentliche Information für das korrekte Leiten eines Pakets durch die Vermittlung an die beabsichtigte(n) Einheit(en).

**[0010]** Die europäische Patentanmeldung EP-A-0 594 196 ist gerichtet auf ein Verfahren zum Durchführen von Adressnachschlagen für eine N-Bit Eingangsadresse in einem Paketdatenkommunikationssystem unter Verwendung einer Hash-Funktion zum Erzeugen einer N-Bit Eingangs-Hash-Adresse, Indizieren in eine Hash-Tabelle, um einen Hash-Bucket unter Verwendung eines Indexteils der Hash-Adresse auszuwählen, wobei der Hash-Bucket einen Zei-

ger und einen Größenwert aufweist, Indizieren in eine Translationstabelle, um einen Binärbaum von Aufzeichnungen an einer von dem Zeiger identifizierten Stelle auszuwählen, wobei der ausgewählte Binärbaum eine Größe aufweist, die von dem Größenwert spezifiziert ist, Durchsuchen der Aufzeichnung des ausgewählten Binärbaums, um eine gespeicherte Hash-Adresse aus jeder Aufzeichnung zu erhalten, einschließlich Vergleichen der gespeicherten Hash-Adressen aus jeder Aufzeichnung mit einem verbleibenden Teil der Eingangs-Hash-Adresse, und wenn eine Übereinstimmung gefunden wird, Abrufen einer gespeicherten neuen Adresse aus den Aufzeichnungen.

**[0011]** Die US-Patentschrift US-A-5 860 136 beschreibt ein System zum Routen einer Nachricht zwischen einer Quelle und einem Ziel, das ein Nachrichtenformat verwendet, das von der Stelle des Nachrichtenziels strukturunabhängig ist. Das System gemäß diesem Patent umfasst eine erste Signaltransceivervorrichtung mit nur einem ersten festgelegten eindeutigen Identifizierungscode wo immer die Transceivervorrichtung sich befinden mag, eine zweite Signaltransceivervorrichtung zur Kommunikation mit der ersten Transceivervorrichtung, die einen zweiten festgelegten eindeutigen Identifizierungscode aufweist, und Routingknoten zum Koppeln eines von der ersten Transceivervorrichtung übertragenen Signals mit der zweiten Transceivervorrichtung an einer unbekanntem physikalischen Stelle.

#### ZUSAMMENFASSUNG DER ERFINDUNG:

**[0012]** Die vorliegende Erfindung ist, wie in Verfahrensanspruch 1 und Vorrichtungsansprüchen 7 und 13 beansprucht, so ausgelegt, dass sie die Mängel des Standes der Technik behebt und die Geschwindigkeit der Datenübertragung erhöht.

**[0013]** Bei einer Ausführungsform ist die vorliegende Erfindung ein Verfahren zum Durchführen von Tabellennachschriften. Das Verfahren umfasst die Schritte des Empfangens von Daten durch eine Eingangsquelle; Parsen der Daten in einen Indexabschnitt und einen entsprechenden Bucketabschnitt; Indizieren des Indexabschnitts auf den entsprechenden Bucketabschnitt; und Zugreifen auf Tabelleninformationen, die in einer Nachschlagetabelle gespeichert sind, unter Verwendung des Bucketabschnitts.

**[0014]** Bei einer anderen Ausführungsform ist die vorliegende Erfindung eine Tabellen-Nachschlage-Indiziervorrichtung. Die Tabellen-Nachschlage-Indiziervorrichtung weist einen Empfänger auf, der ankommende Daten empfängt; einen Datenparser, der die Daten in einen Indexabschnitt und einen entsprechenden Bucketabschnitt parst; einen Indizierer der den Indexabschnitt auf den Bucketabschnitt indiziert; und eine Nachschlagevorrichtung, die auf eine Nach-

schlagetabelle unter Verwendung des entsprechenden Bucketabschnitts zugreift.

**[0015]** Bei einer weiteren Ausführungsform ist die vorliegende Erfindung eine Netzwerkvermittlung mit mehreren Ports, die verwendet werden, um Daten zu empfangen und zu exportieren. Alle der mehreren Ports sind durch ein Kommunikationsmedium miteinander verbunden. Mehrere Address Resolution Logic (ARL) Vorrichtungen sind mit einem der mehreren Ports verbunden, so dass jeder der mehreren Ports eine entsprechende ARL Vorrichtung hat. Jede der mehreren ARL Vorrichtungen besteht aus einem Parser, der Daten in einen Indexabschnitt und einen entsprechenden Bucketabschnitt parst; einem Indizierer, der den Indexabschnitt auf einen entsprechenden Bucketabschnitt indiziert; und einer Nachschlagevorrichtung, die auf Tabelleneinträge in einer Nachschlagetabelle unter Verwendung des Bucketabschnitts zugreift.

#### KURZE BESCHREIBUNG DER ZEICHNUNGEN:

**[0016]** Die Aufgaben und Merkmale der Erfindung werden unter Bezug auf die nachfolgende Beschreibung und die beigefügten Zeichnungen leichter verstanden. Es zeigt:

**[0017]** [Fig. 1A](#) eine Darstellung einer Netzwerkvermittlung mit Nachschlagetabellen (L Tabellen) und Address Resolution Logic (ARL).

**[0018]** [Fig. 1B](#) eine Darstellung eines 16-Bit Indexes und einer 64K Tabelle nach dem Stand der Technik.

**[0019]** [Fig. 1C](#) eine Darstellung eines linearen Indexes in einer 64K Tabelle nach dem Stand der Technik.

**[0020]** [Fig. 2A](#) eine Darstellung eines 48-Bit Schlüssels, der in einen Indexabschnitt und einen Bucketabschnitt geparkt wird, um einen bestimmten Eintrag in einer 64K Tabelle auszuwählen.

**[0021]** [Fig. 2B](#) eine Darstellung des Indizierens einer 64K Tabelle unter Verwendung eines Indexabschnitts und eines Bucketabschnitts für mehrere Indizes.

**[0022]** [Fig. 3](#) eine Darstellung eines 48-Bit Schlüssels, der eine Bucketgröße von 32 aufweist und in eine 64K Tabelle indiziert wird.

**[0023]** [Fig. 4A](#) eine Darstellung der Verfahrensschritte einer Ausführungsform der vorliegenden Erfindung.

**[0024]** [Fig. 4B](#) eine Darstellung der Verfahrensschritte einer weiteren Ausführungsform der vorliegenden Erfindung.

[0025] [Fig. 5](#) eine Darstellung einer Netzwerkvermittlung, die eine Adresse empfängt und einen Indexabschnitt und einen Bucketabschnitt verwendet, um einen bestimmten Eintrag in einer Tabelle zu identifizieren, um den geeigneten Ausgabeport für Datenübertragung zu bestimmen.

[0026] [Fig. 6](#) eine Darstellung einer Netzwerkvermittlung gemäß der vorliegenden Erfindung.

#### DETAILLIERTE BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNGSFORMEN:

[0027] Die vorliegende Erfindung ist gerichtet auf ein Verfahren und eine Vorrichtung, die Tabellen nachschlagen durchführen, wobei ankommende Daten in einen Indexabschnitt und einen Bucketabschnitt geparkt und der Indexabschnitt und der Bucketabschnitt zum Nachschlagen verwendet werden.

[0028] [Fig. 1A](#) zeigt eine Netzwerkvermittlung **50** mit 5 Ports **52**, **54**, **56**, **58** und **60**. Jeder der 5 Ports weist entsprechende Schalttabellen **62**, **64**, **66**, **68** und **70** auf. Jede der Schalttabellen **62**, **64**, **66**, **68** und **70** enthält Address Resolution Logic (ARL) und Nachschlagetabellen (L Tabellen). Alle der 5 Ports sind untereinander durch eine Kommunikationsleitung **72** verbunden.

[0029] Wenn Daten von einem der Ports **52**, **54**, **56**, **58** oder **60** empfangen werden, werden die Daten von der entsprechenden ARL indiziert. Gleichzeitig werden die Daten an alle Ports gesendet, um zu ermitteln, zu welchem Port die Daten zur Ausgabe gesendet werden sollen. Wenn der Ausgabeport ermittelt ist, speichert die ARL die Ausgabeportdaten unter Verwendung des Indexes in der L Tabelle. Auf diese Weise wird der Ausgabeport für eine bestimmte Datenart „gelernt“.

[0030] Sobald der Ausgabeport für eine bestimmte Datenart „gelernt“ ist, indiziert die ARL die Daten, wenn ein Port Daten empfängt, und „schlägt“ den geeigneten Ausgabeport in der Nachschlagetabelle, L Tabelle, „nach“. Die Daten werden dann an den geeigneten Port zur Ausgabe gesendet, und der Schritt des Sendens der Daten an jeden Port entfällt, wodurch der Vermittlungsvorgang beschleunigt wird.

[0031] Es sei beispielsweise angenommen, dass Anfangsdaten von dem Port **52** empfangen werden und durch den Port **60** auszugeben sind. Zunächst werden die Daten an alle Ports gesendet um den geeigneten Ausgabeport zu ermitteln. Wenn der geeignete Ausgabeport ermittelt ist, werden die Anfangsdaten an die Schalttabelle **62** gesendet, damit sie von der ARL indiziert werden, und die Ausgabeportdaten werden unter Verwendung des von der ARL erzeugten Indexes in der Nachschlagetabelle, L Tabelle, gespeichert. Im vorliegenden Fall enthält die L Tabelle

Ausgabeportdaten, die alle Daten mit einem bestimmten Index so leiten, dass sie an den Port **60** gesendet werden.

[0032] Wenn der Port **52** Folgedaten empfängt, indiziert die Schalttabelle **62** die Daten und „schlägt“ den Index in der Nachschlagetabelle, L Tabelle, nach. Wenn die Folgedaten denselben Index haben wie die Anfangsdaten und er in der Nachschlagetabelle gefunden wird, zeigt die Nachschlagetabelle, dass die Folgedaten zur Ausgabe an den Port **60** gesendet werden sollten, und die Folgedaten werden automatisch an den Port **60** zur Ausgabe gesendet.

[0033] Wenn die Folgedaten indiziert sind und der Index nicht in der Nachschlagetabelle gefunden wird, werden die Folgedaten an alle Ports gesendet, um den geeigneten Ausgabeport zu ermitteln. Die Folgedaten werden dann indiziert und die Ausgabeportinformationen werden durch die Schalttabelle **62** in einer Nachschlagetabelle gespeichert. Die Folgedaten werden nun „gelernt“, und wenn das nächste Mal ankommende Daten mit demselben Index ankommen wie die zur Folgedaten, werden die ankommenden Daten unverzüglich an den geeigneten Port zur Ausgabe gesendet, da die Daten bzw. der Index nun „gelernt“ sind.

[0034] [Fig. 2A](#) zeigt eine Datenstruktur **200**, die aus 48 Bits aufgebaut ist, die als Daten zum Empfang durch die Ports **52**, **54**, **56**, **58** und **60** verwendet werden. Diese 48-Bit Datenstruktur **200**, die gewöhnlich als Schlüssel bezeichnet wird, ist typischerweise eine Adressart, die in mehrere Gruppen geparkt ist. Im vorliegenden Fall ist die in [Fig. 2A](#) gezeigte Datenstruktur in 3 Segmente geparkt, das Segment M, das Indexsegment I und das Bucketindex N. Das Bucketindex N wird als Bucket bezeichnet, das Indexsegment I als Index, und das Segment M sind die verbleibenden Bit des Schlüssels.

[0035] Für eine Schlüsselgröße von 48 und eine Tabellengröße von 64K werden 16 Bits zum Indizieren verwendet. Der 48-Bit Schlüssel ist in 3 Segmente geparkt, das Segment M, das Indexsegment I und das Bucketindex N.

[0036] [Fig. 2B](#) ist eine Darstellung eines Indexsegments I(1), das linear auf ein Bucketindex N(1) indiziert wird, eines Indexsegments I(2), das linear auf ein Bucketindex N(2) indiziert wird, und eines Indexsegments I(3), das linear auf ein Bucketindex N(3) indiziert wird ... Jedes Indexsegment I wählt ein Bucketindex N aus, und die Kombination des Indexsegments I und des Bucketindex N wählt einen Eintrag in der Tabelle aus. Jedes Bucketindex N kann so programmiert werden, dass es eine bestimmte Größe hat. Da die Größe des Indexsegments I von der Größe des Bucketindex N abhängt, ändert sich die Größe des Indexsegments I dementsprechend.

[0037] [Fig. 3](#) zeigt einen 48-Bit Schlüssel mit einer Bucketgröße von 32 (d.h. das Bucketindex N ist 5 Bit lang und  $2^5 = 32$ ) für eine 64K Tabelle. Da 16 Bits ( $64K = 2^{16}$ ) benötigt werden, um die 64K Tabelle zu indizieren, und die Bucketgröße 32 (5 Bit lang) ist, ist das Indexsegment I 11 Bit lang (16 Bit – 5 Bit = 11 Bit) und das Segment M ist 32 Bit lang (48 Bit – 16 Bit = 32 Bit).

[0038] Das Indexsegment I wird Index genannt und wird linear auf einen Bucket indiziert, der als Bucketindex N bezeichnet wird. Die Kombination des Indexes und des Buckets wählt direkt einen Eintrag in der Tabelle aus. Daher dauert es, sobald der Index ausgewählt ist, maximal fünf Taktzyklen, um den Bucket zu durchsuchen, und nur 5 Bits müssen für eine Einfüg- oder Löschoption binär sortiert werden. Somit hängt eine Einfüg- oder Löschoption von der Bucketgröße ab, die so programmiert werden kann, dass sie für eine bestimmte Anforderung passend ist.

[0039] Es wird angemerkt, dass es in einigen Fällen wichtig ist, für das Indexsegment I Bits aus einem Schlüssel auszuwählen, die für alle ankommenden Schlüssel wiederkehrende Indexsegmente I aufweisen. Dies ermöglicht Bucketgrößen von 2 oder größer, was für Hochgeschwindigkeitsvermittlung erwünschter ist.

[0040] Bei einigen Anwendungen verwenden Chiphersteller eine Herstelleridentifikationsnummer als Teil des Schlüssels. Es kann daher nützlich sein, Bits in dem Schlüssel, die für die Herstelleridentifikationsnummer verwendet werden, als Teil des Indexsegments I zu verwenden. Ein weiteres nützlich Verfahren zum Indizieren des Indexsegments I ist, einen XOR (exklusives Oder) Index zu verwenden, wobei mit den verbleibenden Bits des Schlüssels eine XOR Operation für den Index durchgeführt wird.

[0041] [Fig. 4A](#) ist eine Darstellung der Schritte, die nötig sind, um eine Nachschlagetabelle gemäß der vorliegenden Erfindung zu indizieren. In Schritt 410 werden Daten als Eingabe empfangen. Die Daten werden dann in Schritt 420 in einen Indexabschnitt und einen entsprechenden Bucketabschnitt geparkt. In Schritt 430 wird der Indexabschnitt auf den Bucketabschnitt indiziert. In Schritt 440 wird unter Verwendung des Bucketabschnitts auf die in einer Nachschlagetabelle gespeicherten Tabelleninformationen zugegriffen.

[0042] [Fig. 4B](#) zeigt Schritte, die denen von [Fig. 4A](#) sehr ähnlich sind. In Schritt 450 werden Daten in einem Port empfangen. Die Daten werden dann in Schritt 460 in einen Indexabschnitt und einen entsprechenden Bucketabschnitt geparkt. In Schritt 470 wird der Indexabschnitt auf den Bucketabschnitt indiziert. Der Hauptunterschied zwischen den in [Fig. 4A](#) und den in [Fig. 4B](#) gezeigten Schritten ist der folgen-

de abschließende Schritt. In Schritt 480 wird unter Verwendung des Bucketabschnitts auf die in einer Nachschlagetabelle gespeicherten Tabelleninformationen zugegriffen. Wenn jedoch in der Nachschlagetabelle keine Tabelleninformationen für den Bucketabschnitt gespeichert sind, speichert die vorliegende Erfindung Tabelleninformationen entsprechend dem Bucketabschnitt in der Nachschlagetabelle, wie in Schritt 490 gezeigt ist.

[0043] Bei einer Ausführungsform der Erfindung wird eine MAC Adresse in einem Port einer Netzwerkvermittlung mit mehreren Ports empfangen. Die MAC Adresse wird in einen Indexabschnitt und einen Bucketabschnitt geparkt. Der Indexabschnitt wird auf einen Bucketabschnitt indiziert, und der Bucketabschnitt wird verwendet, um auf einen Tabelleneintrag in einer Nachschlagetabelle zuzugreifen, um zu ermitteln, an welchen Port die MAC Adresse und die assoziierten Daten zur Ausgabe gesendet werden sollen.

[0044] Wenn es für den entsprechenden Bucketabschnitt einen Eintrag gibt, werden die MAC Adresse und die assoziierten Daten an den geeigneten Port zur Ausgabe gesendet. Falls es keinen Tabelleneintrag gibt, muss der Ausgabeport „gelernt“ werden. Dies geschieht durch Senden der MAC Adresse an alle Ports in der Vermittlung. Der geeignete Ausgabeport sendet eine Nachricht zurück, dass er der geeignete Ausgabeport ist, und diese Information wird in der Nachschlagetabelle mit dem entsprechenden Bucket gespeichert. Daher können alle folgenden MAC Adressen, die denselben Index und Bucket haben, auf den Tabelleneintrag in der Nachschlagetabelle zugreifen, um den geeigneten Ausgabeport zu ermitteln, ohne die MAC Adresse an alle Ports in der Vermittlung zu senden.

[0045] [Fig. 5](#) ist eine Darstellung einer Vermittlung 500. Um Daten sehr schnell und effizient zu übertragen, lernt die Vermittlung 500, an welchen Ausgabeport jede ankommende MAC Adresse 505 zu übertragen ist, und speichert diese Information in der Tabelle 510. Wenn die Vermittlung 500 eine neue MAC Adresse 505 empfängt, wird die MAC Adresse in der Tabelle 510 nachgeschlagen und an den geeigneten Ziel-Ausgabeport 540 oder 545 vermittelt.

[0046] Eine Adresse 505 wird durch einen Eingabeport 515 von der Vermittlung 500 empfangen. Der Eingabeport 515 überträgt die Adresse 505 an den Parser 520. Der Parser 520 segmentiert die Adresse 505 in drei Teile, das Segment 525, das mit M bezeichnet ist, das Segment 530, das mit I bezeichnet ist, und das Segment 535, das mit N bezeichnet ist. Das Segment I wird linear auf einen Bucket 535 indiziert, und der Bucket 535 indiziert direkt einen Eintrag in der Tabelle 510. Tatsächlich schlägt der Bucket 535 in der Tabelle 510 nach, um zu ermitteln, welcher

der Ausgabeports **540** oder **545** zu verwenden ist, um die ankommenden Daten zu übertragen. Der Index **I** wählt nur einen Bucket aus.

[0047] **Fig. 6** ist eine Darstellung einer Netzwerkvermittlung **600** mit einem Port **605**, einem Port **610** und einem Port **615**. Alle dieser Ports sind durch eine Kommunikationsleitung **620** untereinander verbunden.

[0048] Der Port **605** weist eine entsprechende Address Resolution Logic (ARL) Vorrichtung **625** und eine entsprechende Nachschlagetabelle **630** auf. Die ARL **625** besteht aus einem Parser **635**, einem Indizierer **640** und einer Nachschlagevorrichtung **645**.

[0049] Der Port **610** weist eine entsprechende Address Resolution Logic (ARL) Vorrichtung **650** und eine entsprechende Nachschlagetabelle **655** auf. Die ARL **650** besteht aus einem Parser **660**, einem Indizierer **665** und einer Nachschlagevorrichtung **670**.

[0050] Der Port **615** weist eine entsprechende Address Resolution Logic (ARL) Vorrichtung **675** und eine entsprechende Nachschlagetabelle **680** auf. Die ARL **675** besteht aus einem Parser **685**, einem Indizierer **690** und einer Nachschlagevorrichtung **695**.

[0051] Wenn der Port **605** eine an den Port **610** zu sendende MAC Adresse empfängt, ist die Operation der Vermittlung wie folgt. Eine MAC Adresse wird in dem Port **605** der Netzwerkvermittlung **600** mit mehreren Ports **605**, **610** und **615** empfangen. Die MAC Adresse wird zur Verarbeitung an die ARL **625** gesendet. Der Parser **635** parst die MAC Adresse in einen Indexabschnitt und einen Bucketabschnitt. Der Indizierer **640** indiziert den Indexabschnitt auf einen Bucket in der Nachschlagevorrichtung **645**. Der Bucketabschnitt in der Nachschlagevorrichtung **645** wird verwendet, um auf einen Tabelleneintrag in der Nachschlagetabelle **630** zuzugreifen, um zu ermitteln, an welchen Port die MAC Adresse und die assoziierten Daten zur Ausgabe gesendet werden sollen. Wenn es für den entsprechenden Bucketabschnitt einen Eintrag gibt, werden die MAC Adresse und die assoziierten Daten an den geeigneten Ausgabeport gesendet. In diesem Fall leitet der Nachschlagetabellen-Eintrag die MAC Adresse und die assoziierten Daten an den Ausgabeport **610**. Wenn jedoch kein Tabelleneintrag vorhanden ist, muss der Ausgabeport „gelernt“ werden. Dies geschieht durch Senden der MAC Adresse an alle Ports, die Ports **605**, **610** und **615**, in der Vermittlung **600**. Der geeignete Ausgabeport sendet eine Nachricht zurück, dass er der geeignete Ausgabeport ist, und diese Information wird in der Nachschlagetabelle **630** mit dem entsprechenden Bucket gespeichert. In diesem Fall sendet der Port **610** dem Port **605** ein Signal, das angibt, dass der Port **610** der geeignete Ausgabeport ist, und diese Information wird in der Nachschlagetabelle **630**

gespeichert. Daher können folgende MAC Adressen, die denselben Index und Bucket haben, auf den Tabelleneintrag in der Nachschlagetabelle zugreifen, um den geeigneten Ausgabeport zu ermitteln, ohne die MAC Adresse an alle Ports in der Vermittlung zu senden.

[0052] Ein Vorteil der vorliegenden Erfindung ist, dass Einfüg- und Lösch-Operationen nicht länger von der Tabellengröße abhängig sind. Wie oben erläutert, ist Adressnachschiagen im Stand der Technik von der Tabellengröße abhängig. Wenn daher die Tabellengröße 64K beträgt, ist der Index 16 Bit lang. Somit umfasst eine Einfüg- oder Lösch-Operation das Sortieren und Bewegen von 16-Bit Indices.

[0053] Die vorliegende Erfindung hat jedoch für eine 64K Tabelle mit einer Bucketgröße von 32 (5 Bit) nur einen 11-Bit Index (16 Bit – 5 Bit = 11 Bit). Daher sind, sobald ein Bucket von einem Index ausgewählt ist, Einfüg- oder Lösch-Operationen von der Bucketgröße abhängig und umfassen, in diesem Fall, das Sortieren und Bewegen von 5-Bit Bucket-Indices, was viel effizienter ist, als das Sortieren und Bewegen von 16-Bit Indices.

[0054] Obwohl die Erfindung auf der Grundlage der oben erläuterten Ausführungsformen beschrieben worden ist, ist es für den Fachmann offensichtlich, dass bestimmte Modifikationen, Variationen und alternative Konstruktionen vorgenommen werden können, ohne vom Umfang der Erfindung abzuweichen. Anstelle der Verwendung eines linearen Indexes ist beispielsweise unter gewissen Umständen ein XOR Index angebracht. Ebenso können die Bucket- und Indexgrößen so programmiert werden, dass sie sich verschiedenen Umständen anpassen und auf bestimmte Anforderungen zugeschnitten sind.

## Patentansprüche

1. Verfahren zum Durchführen von Tabellennachschiagen in einer Netzwerkvorrichtung (**50**), mit den folgenden Schritten:  
Empfangen (**410**, **450**) eines Datenpakets (**505**) durch einen Eingangsport (**52**, **515**, **605**) der Netzwerkvorrichtung (**50**);  
Parsen (**420**, **460**) des Datenpakets (**505**) in einen Indexabschnitt (**I**, **535**) und einen entsprechenden Bucketabschnitt (**N**, **535**);  
Indizieren (**430**, **470**) des Indexabschnitts (**I**, **535**) auf den entsprechenden Bucketabschnitt (**N**, **535**);  
Zugreifen auf Adresstabelleninformationen, die in einer Adressnachschiagetabelle (**510**, **630**, **655**) gespeichert sind, unter Verwendung des Bucketabschnitts (**N**, **535**);  
**dadurch gekennzeichnet**, dass das Verfahren des Weiteren den folgenden Schritt umfasst:  
Speichern (**440**, **480**) von Informationen über das Datenpaket (**505**) in der Adressnachschiagetabelle

(**510, 630, 655**) als

Adresstabelleninformationen, wenn keine Adresstabelleninformationen verfügbar sind, wobei der Bucketabschnitt (N, **535**) verwendet wird, um auf Adresstabelleninformationen zuzugreifen.

2. Verfahren nach Anspruch 1, wobei der Schritt des Indizierens (**430, 470**) des Indexabschnitts (I, **535**) auf den Bucketabschnitt (N, **535**) der Schritt des linearen Indizierens des Indexabschnitts (I, **535**) auf den Bucketabschnitt (N, **535**) ist.

3. Verfahren nach Anspruch 1, wobei der Schritt des Indizierens (**430, 470**) des Indexabschnitts (I, **535**) auf den Bucketabschnitt (N, **535**) ist der Schritt des XOR-Indizierens des Indexabschnitts (I, **535**) auf den Bucketabschnitt (N, **535**) ist.

4. Verfahren nach Anspruch 1, das des Weiteren den Schritt des Sortierens des Bucketabschnitts (N, **535**) umfasst.

5. Verfahren nach Anspruch 1, das des Weiteren den Schritt des binären Sortierens des Bucketabschnitts (N, **535**) umfasst.

6. Verfahren nach Anspruch 1, wobei der Schritt des Parsens (**420, 460**) des Datenpakets (**505**) in einen Indexabschnitt (I, **535**) und einen entsprechenden Bucketabschnitt (N, **535**) des Weiteren den Schritt des Parsens des Indexabschnitts (I, **535**) so umfasst, dass der Indexabschnitt (I, **535**) wiederkehrt, wenn andere Daten in den Indexabschnitt (I, **535**) und den entsprechenden Bucketabschnitt (N, **535**) geparkt werden.

7. Adresstabellen-Nachschlage-Indiziervorrichtung (**645**) mit:

einem Empfangsabschnitt eines Ports (**52, 515, 605**) einer Netzwerkvorrichtung (**50**), der so ausgelegt ist, dass er ein ankommendes Datenpaket (**505**) empfängt;

einem Datenparser (**520, 635, 660, 665, 685**), der so ausgelegt ist, dass er das Datenpaket (**505**) in einen Indexabschnitt (I, **535**) und einen entsprechenden Bucketabschnitt (N, **535**) parst;

einen Indizierer (**640, 665, 690**), der so ausgelegt ist, dass er den Indexabschnitt (I, **535**) auf den Bucketabschnitt (N, **535**) indiziert;

einer Adressnachschlagevorrichtung (**645**), die so ausgelegt ist, dass sie auf eine Adressnachschlagetabelle (**510, 630, 655**) unter Verwendung des Bucketabschnitts (N, **535**) zugreift; gekennzeichnet durch einen Speichermechanismus, der so ausgelegt ist, dass er Informationen über das Datenpaket (**505**) in der Adressnachschlagetabelle (**510, 630, 655**) als Adresstabelleninformationen speichert, wenn keine Adresstabelleninformationen verfügbar sind, wobei der Bucketabschnitt (N, **535**) verwendet wird, um auf Adresstabelleninformationen zuzugreifen.

8. Vorrichtung nach Anspruch 8, wobei der Indizierer (**640, 665, 690**) so ausgelegt ist, dass er den Indexabschnitt (I, **535**) linear auf den Bucketabschnitt (N, **535**) indiziert.

9. Vorrichtung nach Anspruch 8, wobei der Indizierer (**640, 665, 690**) so ausgelegt ist, dass er den Indexabschnitt (I, **535**) auf den Bucketabschnitt (N, **535**) XOR-indiziert.

10. Vorrichtung nach Anspruch 8, die des Weiteren einen Sortierer aufweist, der so ausgelegt ist, dass er den Bucketabschnitt (N, **535**) sortiert.

11. Vorrichtung nach Anspruch 10, wobei der Sortierer so ausgelegt ist, dass er den Bucketabschnitt (N, **535**) binär sortiert.

12. Vorrichtung nach Anspruch 7, wobei der Datenparser (**520, 635, 660, 665, 685**) so ausgelegt ist, dass er die Indexabschnitte (I, **535**) so in Gruppen parst, dass jeder Indexabschnitt (I, **535**) in einer Gruppe derselbe ist wie die anderen Indexabschnitte (I, **535**) in dieser Gruppe.

13. Netzwerkvermittlung (**500, 600**) mit: mehreren Ports (**52, 54, 56, 58, 60, 540, 545, 605, 610, 615**), die verwendet werden, um Daten (**505**) zu empfangen und zu exportieren, wobei alle der mehreren Ports durch ein Kommunikationsmedium miteinander verbunden sind; mehreren Address Resolution Logic (ARL) Vorrichtungen (**625, 650, 675**), wobei jede der mehreren ARL Vorrichtungen (**625, 650, 675**) mit einem der mehreren Ports verbunden ist, wobei jeder der mehreren Ports eine entsprechende ARL Vorrichtung (**625, 650, 675**) hat, wobei jede der mehreren ARL Vorrichtungen (**625, 650, 675**) aufweist: einen Parser (**520, 635, 660, 665, 685**), der so ausgelegt ist, dass er Daten in einen Indexabschnitt (I, **535**) und einen entsprechenden Bucketabschnitt (N, **535**) parst; einen Indizierer (**640, 665, 690**), der so ausgelegt ist, dass er den Indexabschnitt (I, **535**) auf einen entsprechenden Bucketabschnitt (N, **535**) indiziert; eine Nachschlagevorrichtung (**645, 670, 695**), die so ausgelegt ist, dass sie auf Tabelleneinträge in einer Nachschlagetabelle (**510, 630, 655**) unter Verwendung des Bucketabschnitts (N, **535**) zugreift; gekennzeichnet durch einen Speichermechanismus, der so ausgelegt ist, dass er Informationen über das Datenpaket (**505**) in der Adressnachschlagetabelle (**510, 630, 655**) als Adresstabelleninformationen speichert, wenn keine Adresstabelleninformationen verfügbar sind, wobei der Bucketabschnitt (N, **535**) verwendet wird, um auf Adresstabelleninformationen zuzugreifen.

Es folgen 8 Blatt Zeichnungen

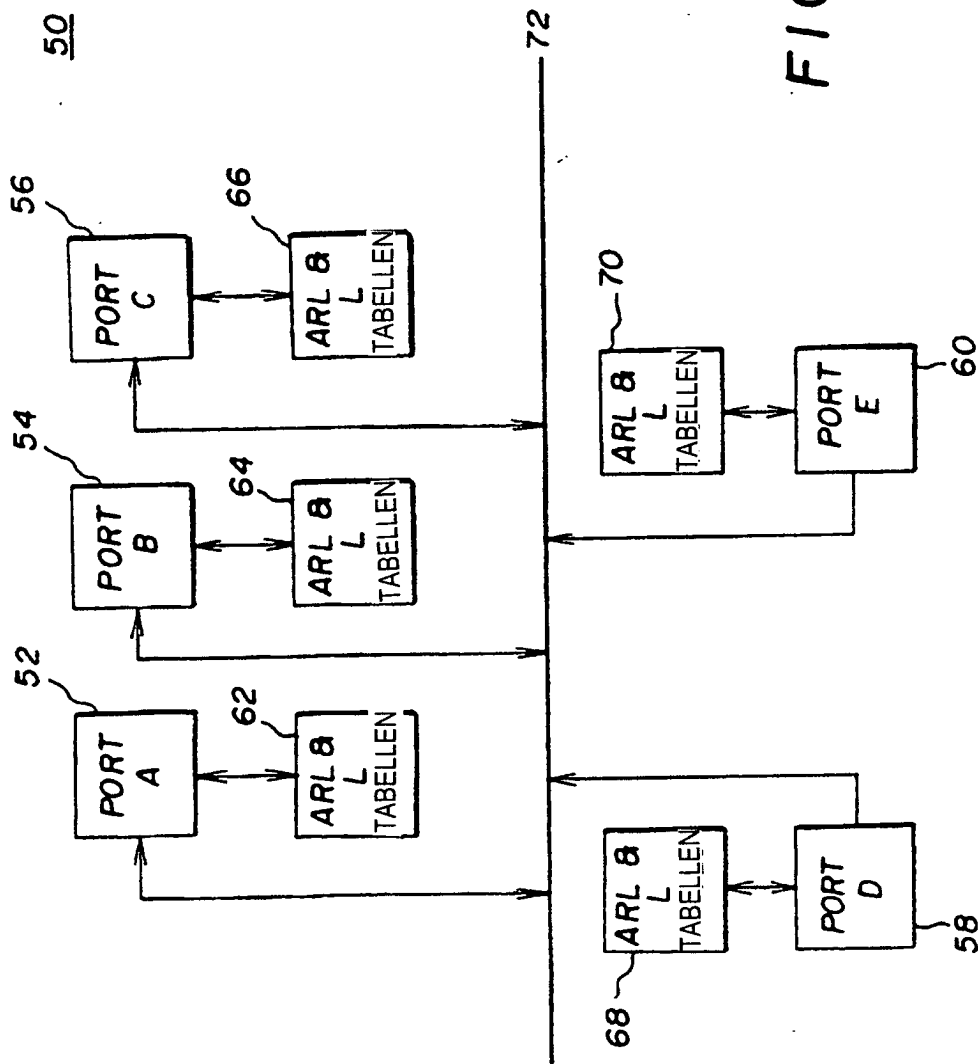
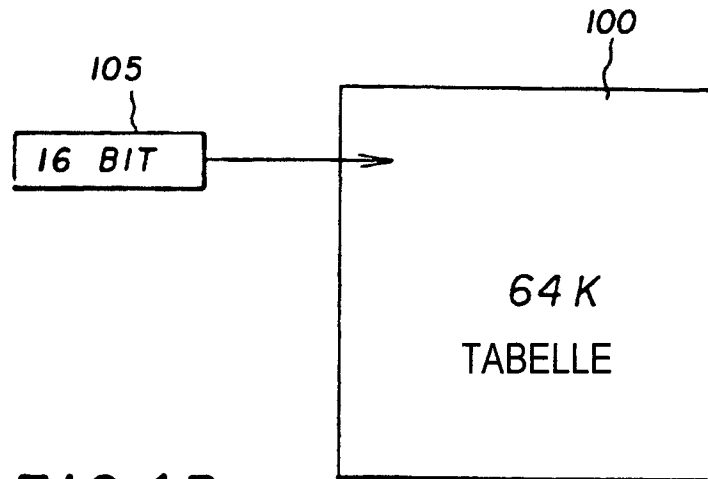


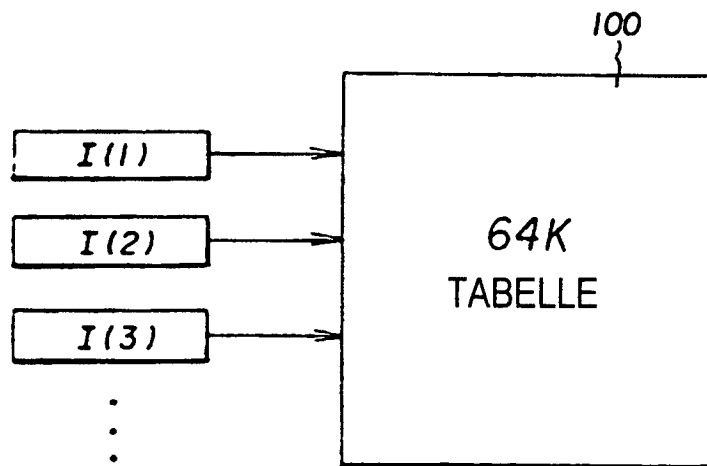
FIG. 1A





**FIG. 1B**

STAND DER TECHNIK



**FIG. 1C**

STAND DER TECHNIK

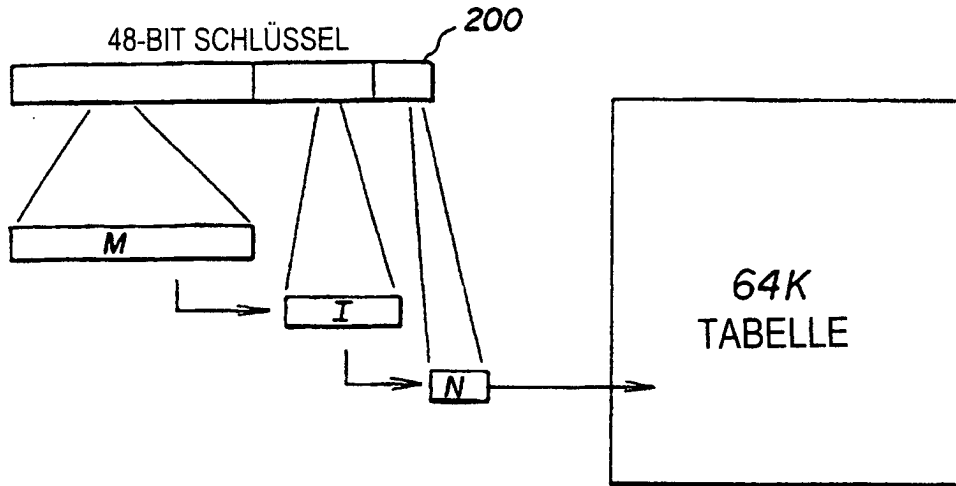


FIG. 2A

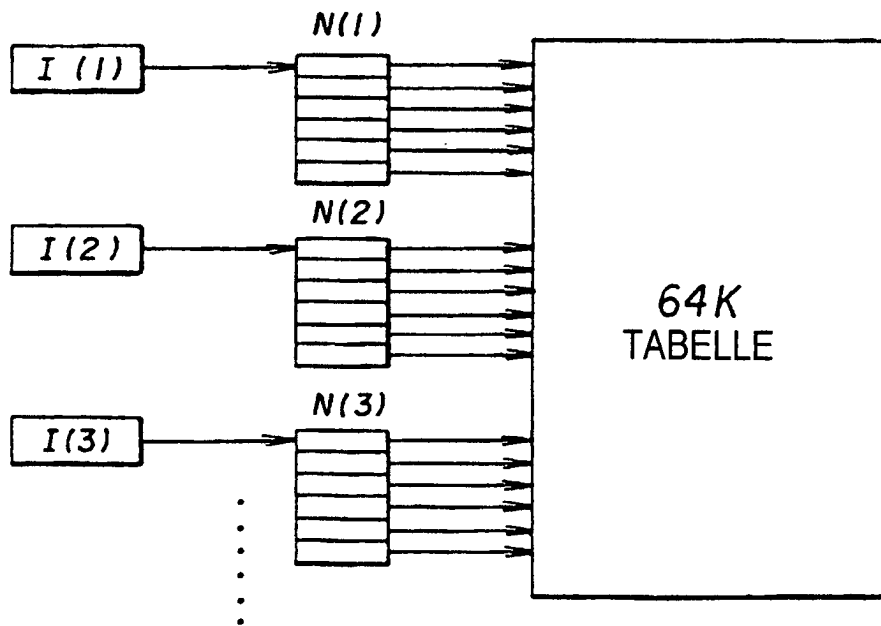


FIG. 2B

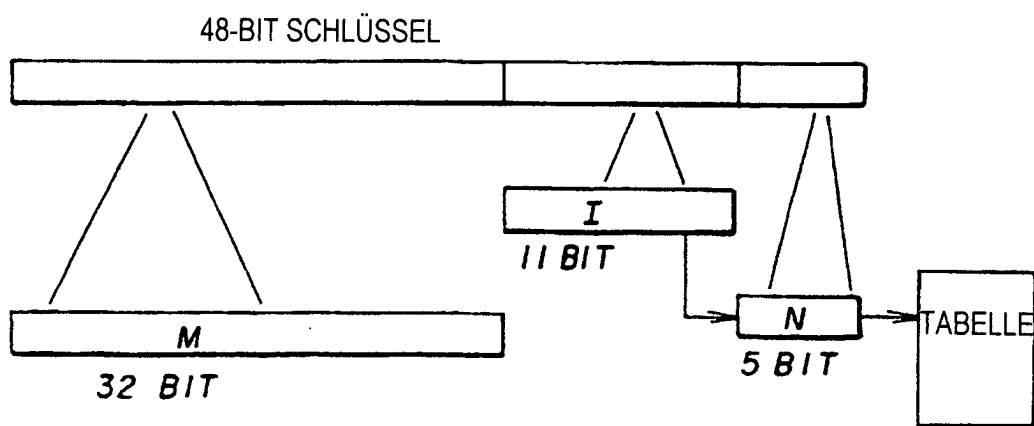


FIG. 3

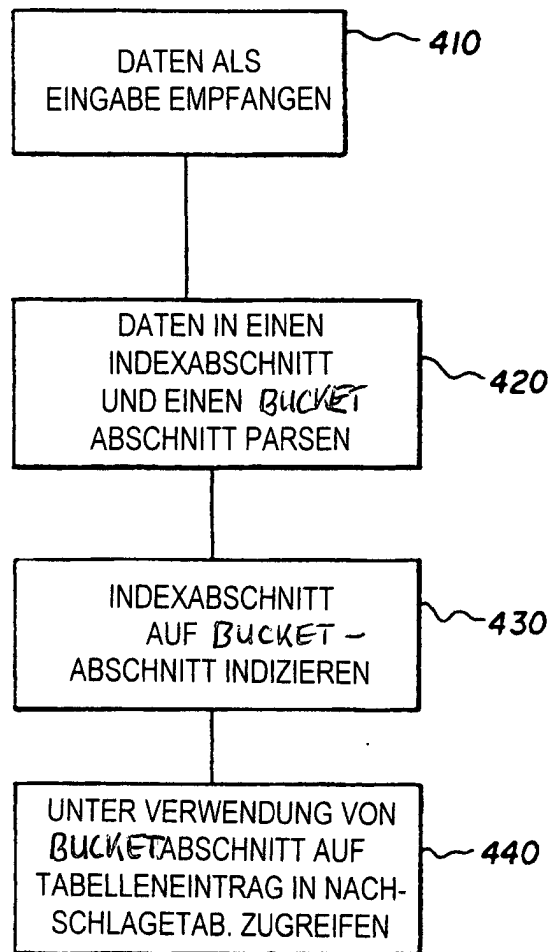


FIG. 4A

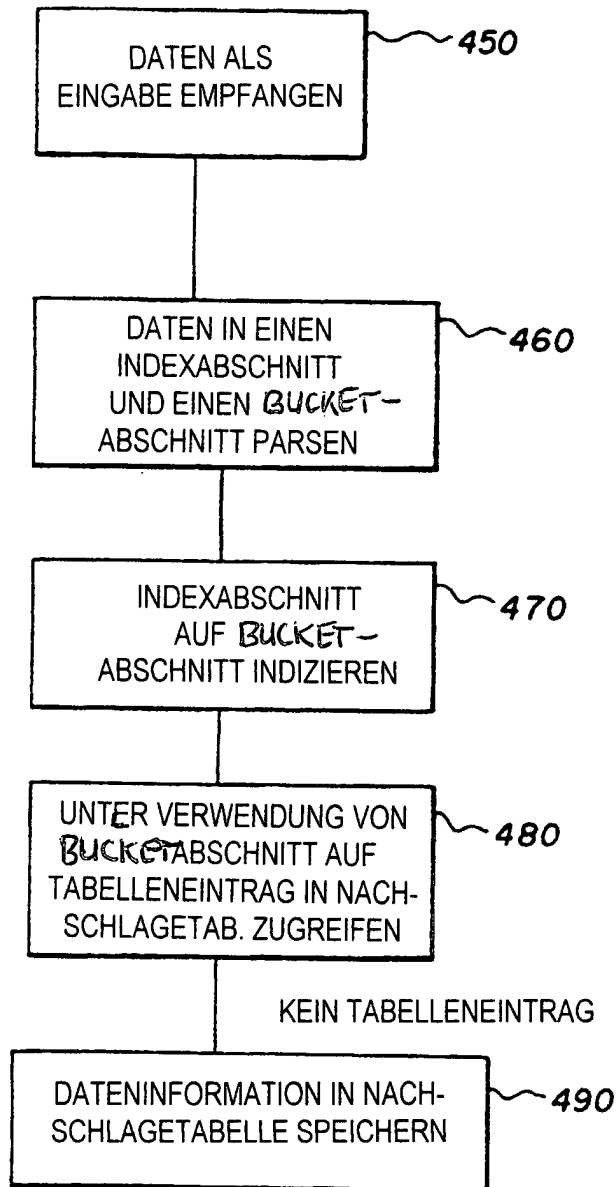
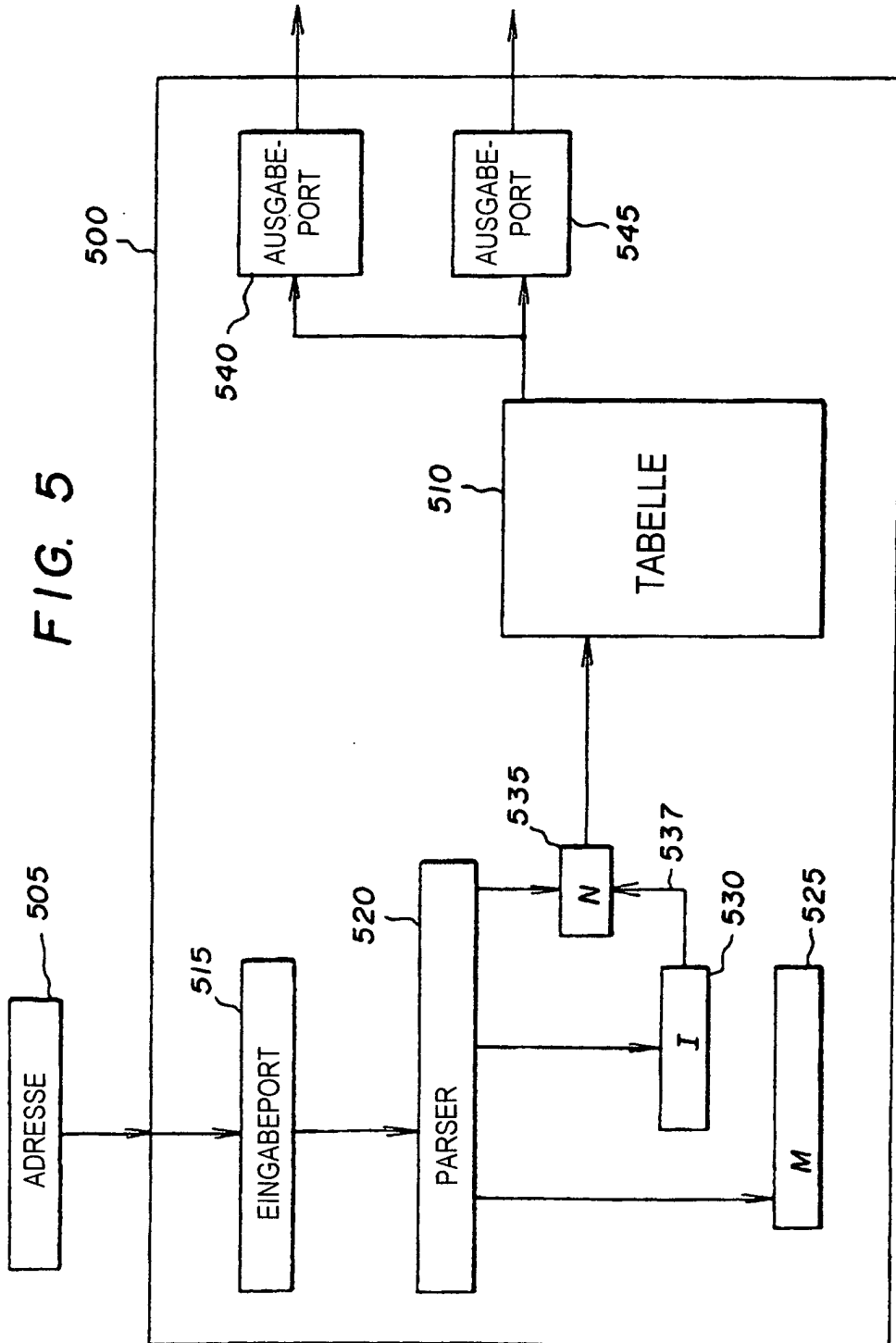


FIG. 4B

FIG. 5



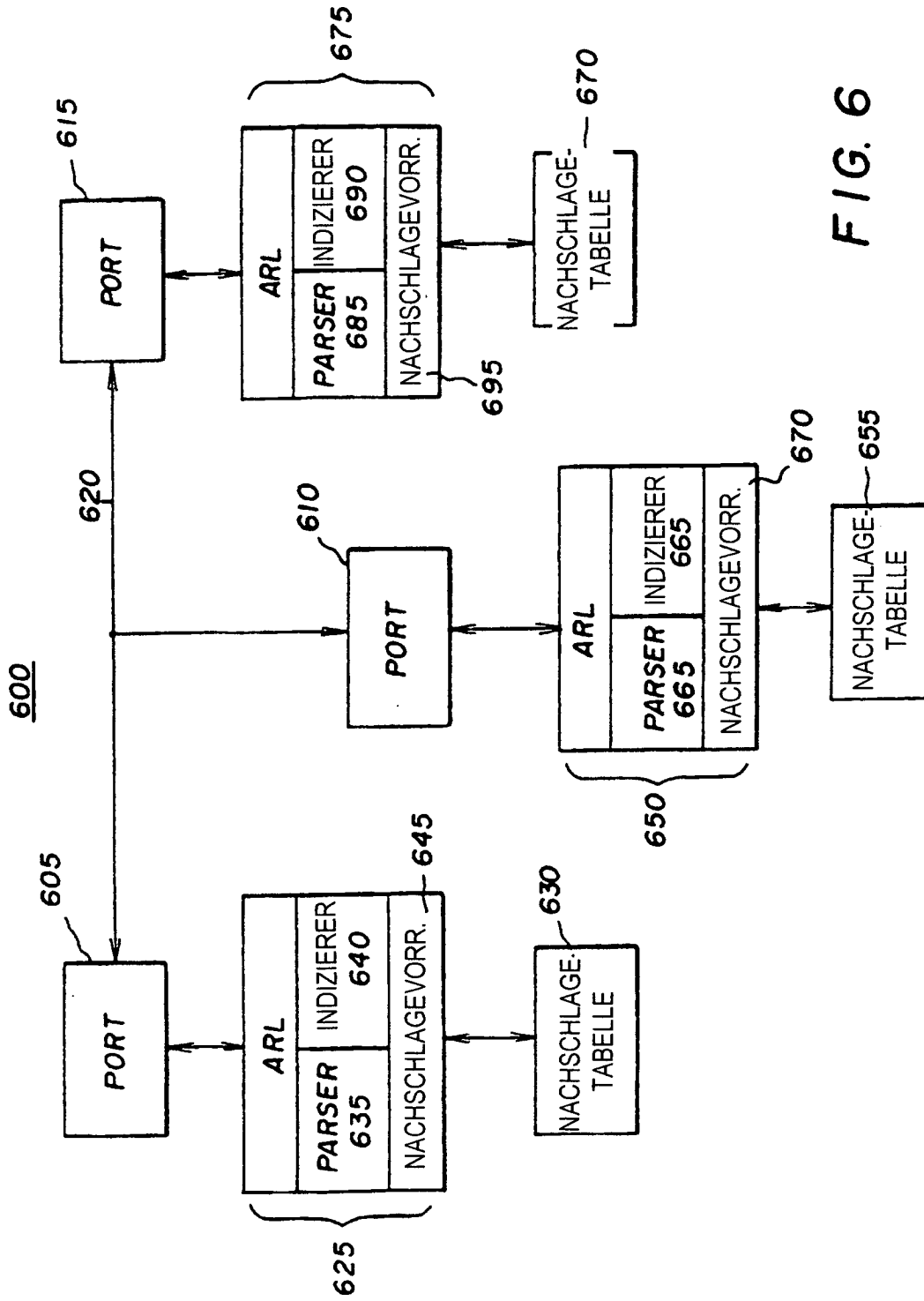


FIG. 6