



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 601 30 711 T2** 2008.07.03

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 1 176 768 B1**

(21) Deutsches Aktenzeichen: **601 30 711.9**

(96) Europäisches Aktenzeichen: **01 306 304.5**

(96) Europäischer Anmeldetag: **23.07.2001**

(97) Erstveröffentlichung durch das EPA: **30.01.2002**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **03.10.2007**

(47) Veröffentlichungstag im Patentblatt: **03.07.2008**

(51) Int Cl.⁸: **H04Q 11/04** (2006.01)
G06F 12/08 (2006.01)

(30) Unionspriorität:

627800 28.07.2000 US

(73) Patentinhaber:

Ericsson AB, Stockholm, SE

(74) Vertreter:

**Benedum, U., Dipl.-Chem.Univ.Dr.rer.nat.,
Pat.-Anw., 80333 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:

**Zhou, Fan, Petaluma, CA 94954, US; Bianchini,
Ronald P Jr., Pittsburgh, PA 15238, US**

(54) Bezeichnung: **Verfahren und Vorrichtung zum Speichern von Paketen mit einem Grenzeanzeiger**

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

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

Beschreibung**GEBIET DER ERFINDUNG**

[0001] Die Erfindung betrifft die Speicherung von Paketen entlang von Cache-Zeilen eines Speichers mit Paketgrenzenindikatoren. Genauer gesagt, betrifft die Erfindung die Speicherung einer Mehrzahl von Paketen entlang Cache-Zeilen eines Speichers, wobei nur ein Paketgrenzenindikator mit der Mehrzahl an Paketen verbunden ist, so dass weniger von dem Speicher zur Markierungsfunktion verwendet wird.

HINTERGRUND DER ERFINDUNG

[0002] Bei dem breiten TDM-Schema, das im Speichercontroller eingesetzt wird, werden mehrere Pakete in einem Datenwort abgespeichert. Zwar ist die Paketlänge in den Paketdaten enthalten, aber der Speichercontroller hat nicht genügend Information zum Ableiten von Paketgrenzen. Die Paketdaten werden in Streifen auf mehrere Speichercontroller aufgeteilt. So hat kein einzelner Speichercontroller genügend Information zum Bestimmen der tatsächlichen Länge. Die Speicherung zusätzlicher Längen- oder Paketgrenzeninformation für jedes Paket wäre recht kostspielig. Der Speichercontroller kann die Prioritäten beim Warteschlangenabbau nur an bekannten Paketgrenzen vermitteln und muss daher eine gewisse Längen- oder Paketgrenzeninformation speichern. Ein Beispiel für eine Datenspeichervorrichtung wird von US 5893121 bereitgestellt.

ZUSAMMENFASSUNG DER ERFINDUNG

[0003] Unter einem Aspekt der Erfindung wird eine Vorrichtung bereitgestellt zum Speichern von Paketen, umfassend: einen Speicher zum Halten von Paketen; einen Mechanismus zum Speichern von mindestens zwei Paketen in dem Speicher, wobei nur ein Paketgrenzenindikator mit den mindestens zwei Paketen einhergeht, der Speichermechanismus einen Speichercontroller enthält, der derart ausgelegt ist, dass er einen Paketgrenzenindikator in dem Speicher platziert, nachdem eine festgelegte Anzahl bits in dem Speicher gespeichert worden ist, wobei der Speicher durch Cache-Zeilen definiert wird und die Pakete entlang der Cache-Zeilen in dem Speicher gespeichert werden, wobei der Speichercontroller derart ausgelegt ist, dass er bits von Daten von Paketen in einer Cache-Zeile in dem Speicher speichert und ein Identifizierer in der Cache-Zeile von Daten anzeigt, wie viele bits in der Cache-Zeile gültig sind.

[0004] Unter einem anderen Aspekt der Erfindung wird ein Verfahren bereitgestellt zum Speichern von Paketen, gekennzeichnet durch die Schritte: Empfangen von Paketen in einem Speicher und Speichern mindestens zweier Pakete mit einem Speichercontroller in dem Speicher, wobei mit den mindestens zwei Paketen nur ein Paketgrenzenindikator einhergeht; wobei der Speicherschnitt die Schritte enthält: Platzieren eines Paketgrenzenindikators mit dem Speichercontroller in dem Speicher, nachdem eine festgelegte Anzahl bits in dem Speicher gespeichert worden ist; Speichern der Pakete entlang Cache-Zeilen in dem Speicher und Speichern von bits von Daten von Paketen in einer Cache-Zeile in dem Speicher, wobei ein Identifizierer in der Cache-Zeile von Daten angibt, wie viele bits in der Cache-Zeile gültig sind.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0005] In den beigefügten Zeichnungen sind die bevorzugte Ausführungsform der Erfindung und bevorzugte Verfahren zur Ausführung der Erfindung dargestellt. Es zeigt/zeigen:

[0006] [Fig. 1](#) eine schematische Darstellung des Paket-Striping in dem erfindungsgemäßen Switch;

[0007] [Fig. 2](#) eine schematische Darstellung von einer OC-48-Schnittstellenkarte;

[0008] [Fig. 3](#) eine schematische Darstellung von einem verketteten Netzwerk-Blade;

[0009] die [Fig. 4a](#) und [Fig. 4b](#) schematische Darstellungen zur Konnektivität der Fabric-ASICs;

[0010] [Fig. 5](#) eine schematische Darstellung der Sync-Impulsverteilung;

[0011] [Fig. 6](#) eine schematische Darstellung der Beziehung zwischen Übertragungs- und Empfangssequenz-zählern für den Separator bzw. den Unstriper;

[0012] [Fig. 7](#) eine schematische Darstellung einer erfindungsgemäßen Vorrichtung;

[0013] [Fig. 8](#) eine schematische Darstellung eines Speichers.

EINGEHENDE BESCHREIBUNG

[0014] Siehe die Zeichnungen, in denen gleiche Bezugszahlen in allen Ansichten ähnliche oder identische Teile bezeichnen, und insbesondere die [Fig. 7](#) und [Fig. 8](#): Dargestellt ist eine Vorrichtung **10** zum Speichern von Paketen **12**. Die Vorrichtung **10** umfasst einen Speicher **14** zum Halten von Paketen **12**. Die Vorrichtung **10** umfasst einen Mechanismus zum Speichern eines Pakets **12** und mindestens zwei Pakete **12** in dem Speicher **14**, mit denen nur ein Paketgrenzenindikator **18** einhergeht.

[0015] Vorzugsweise enthält der Speichermechanismus **16** einen Speichercontroller **20**. Der Speichercontroller **20** platziert vorzugsweise einen Paketgrenzenindikator **18** in dem Speicher **14**, nachdem eine festgelegte Anzahl an bits in dem Speicher **14** gespeichert worden ist. Vorzugsweise setzt der Speichercontroller **20** einen Paketgrenzenindikator **18** ein, nachdem ein Wert an Paketen **12** von 15 Cache-Zeilen im Speicher **14** gespeichert wurde. Der Speichercontroller **20** vermittelt vorzugsweise, welche Pakete **12** von dem Speicher **14** übertragen werden sollen, auf Basis von Paketgrenzenindikatoren **18** in Bezug auf die Priorität der Pakete **12**.

[0016] Vorzugsweise ist der Speicher **14** durch Cache-Zeilen **22** definiert, und die Pakete werden entlang von Cache-Zeilen **22** in dem Speicher **14** gespeichert. Der Speichercontroller **20** speichert vorzugsweise bits von Daten von Paketen **12** in einer Cache-Zeile in dem Speicher **14** ab. Es gibt einen Identifizierer **24** in der Cache-Zeile von Daten, der angibt, wie viele bits in der Cache-Zeile gültig sind. Vorzugsweise ist jede Cache-Zeile 200 bits lang. Der Identifizierer **24** ist vorzugsweise 2 bits von den 200 bits jeder Cache-Zeile.

[0017] Die Erfindung betrifft ein Verfahren zum Speichern von Paketen **12**. Das Verfahren umfasst die Schritte Empfangen von Paketen **12** in einem Speicher **14**. Dann erfolgt ein Schritt Speichern eines Pakets und vorzugsweise mindestens zweier Pakete **12** mit einem Speichercontroller **20** in dem Speicher **14**, mit denen nur ein Paketgrenzenindikator **18** einhergeht.

[0018] Vorzugsweise enthält der Speicherschritt den Schritt Platzieren eines Paketgrenzenindikators **18** mit dem Speichercontroller **20** in dem Speicher **14**, nachdem eine festgelegte Anzahl bits von Paketen **12** in dem Speicher **14** gespeichert worden ist. Der Speicherschritt umfasst vorzugsweise den Schritt Speichern der Pakete **12** entlang Cache-Zeilen **22** in dem Speicher **14**.

[0019] Der Platzierungsschritt beinhaltet vorzugsweise den Schritt Speichern von bits von Daten von Paketen **12** in einer Cache-Zeile in dem Speicher **14**, wobei ein Identifizierer **24** in der Cache-Zeile von Daten angibt, wie viele bits in der Cache-Zeile gültig sind. Nach dem Speicherschritt erfolgt vorzugsweise der Schritt Vermitteln von Paketen **12** aus dem Speicher **14** auf Basis von Paketgrenzenindikatoren **18** in Bezug auf die Priorität der Pakete **12**.

[0020] Beim Betrieb der bevorzugten Ausführungsform wird die Vorrichtung **10** dazu verwendet, die Mindestzahl an Paketgrenzen beizubehalten. Im engsten Sinne lässt sie sich als verlustreicher Kompressionsalgorithmus verstehen, der zum Komprimieren von Paketgrenzeninformation verwendet wird. Jede Cache-Zeile von Daten wird zusammen mit einem Zwei-bit-Feld gespeichert, das die komprimierte Paketgrenzeninformation enthält. Eine völlig unkomprimierte Version der Daten wäre ein einziges Kontrollbit für jedes Daten-bit, das anzeigt, dass das entsprechende Daten-bit das letzte bit des Pakets war. Verglichen mit dieser Informationsform hat die Kompressionsform nur 1/100 der Größe. Die Vorrichtung **10** verringert den Speicherplatz durch Beseitigung unnötiger Information.

[0021] Jede Cache-Zeile von Daten geht mit einem Zwei-bit-Feld einher, das angibt, wie viele bits gültig sind. Die codierten Werte sind:

- alle 200 bits sind gültig, enden aber nicht an einer Paketgrenze
- alle 200 bits sind gültig und enden an einer Paketgrenze
- mindestens 191 bits sind gültig, und das erste ungültige bit ist die erste 0, die bei der Suche von 200 abwärts gefunden wird, und
- weniger als 191 bits sind gültig, wobei die bits 200-193 eine Zählung der Anzahl gültiger bits enthalten.

[0022] Oder gleichwertig:

- keine bekannten Paketgrenzen
- Paketgrenze bei 200 bits
- Paketgrenze neben der signifikantesten 0 und

– Paketgrenze bei dem bit, das durch die Daten-bits 200-193 angegeben wird

[0023] Indem man nur 2 bits pro 200 bit Eingabe hinzufügt, kann man soviel Paketgrenzeninformation wie nötig durch diese besondere Anwendung beibehalten.

[0024] Gewöhnlich sind alle 200-bit-Caches vollständig gefüllt. Dies ist effizient, würde aber den Verlust aller bekannten Paketgrenzen bewirken. Damit sichergestellt wird, dass bekannte Paketgrenzen regelmäßig vorkommen, erzwingt die Logik, welche die Caches füllt, einen der bekannten Paketgrenzenindikatoren **18** nach 15 "keine bekannte Paketgrenze"-Indikatoren. Der Speicher **14** wird auf diese Weise entlang der Cache-Zeilen gefüllt.

[0025] Wenn die Daten im Speicher **14** aus der Warteschlange abgebaut werden, werden alle Daten entlang der Cache-Zeilen **22** gesendet, bis ein Paketgrenzenindikator **18** erreicht wird. Am Paketgrenzenindikator **18** ergibt sich die Gelegenheit für Prioritäten, die mit den Daten einhergehen, die vermittelt werden sollen. Wenn gewünscht, blockieren daher Daten niedriger Priorität nicht Daten mit hoher Priorität.

[0026] Das übliche Queueing und Dequeueing der Pakete **12** selbst ist im Stand der Technik bekannt. Die Standard-Speicherung von Paketen **12** im Speicher **14** wird von der Beschreibung hier dahingehend modifiziert, dass die Ineffizienz, weil Paketgrenzenindikatoren **18** an jedem Paket gespeichert werden müssen, gegenüber der Vermittlungsfrequenz zu Prioritätszwecken minimiert wird. Siehe eine vollständigere Beschreibung eines Gesamtverfahrens und -systems im Hinblick auf die Übertragung von Paketen **12**, dass die hier beschriebene Vorrichtung und das hier beschriebene Verfahren vorteilhaft genutzt werden können, in der U.S.-Patentanmeldung 09/293 563 mit dem Titel "Very Wide Memory TDM Switching System".

[0027] Der Switch verwendet RAID-Techniken zur Erhöhung der Gesamt-Switch-Bandbreite und gleichzeitig zur Minimierung der Bandbreite einer einzelnen Fabrik. Bei der Switch-Architektur werden alle Daten gleichmäßig über alle Fabriken verteilt. Somit fügt der Switch Bandbreite hinzu, indem er Fabriken hinzufügt. Die Fabrik muss ihre Bandbreitenkapazität nicht erhöhen, weil der Switch die Bandbreitenkapazität steigert.

[0028] Jede Fabrik liefert 40G Vermittlungsbandbreite. Das System unterstützt 1, 2, 3, 4, 6 oder 12 Fabriken, wobei die redundante/Ersatz-Fabrik ausgenommen ist. Anders gesagt, kann der Switch ein 40G-, 80G-, 120G-, 160G-, 240G- oder 480G-Switch sein, je nachdem, wie viele Fabriken installiert sind.

[0029] Eine Portcard liefert 100G Port-Bandbreite. Auf jeweils vier Portcards muss 1 Fabrik vorhanden sein. Die Switch-Architektur unterstützt keine beliebigen Installationen von Portcards und Fabriken.

[0030] Die Fabrik-ASICs unterstützen Zellen und Pakete. Im Ganzen überträgt der Switch die Daten im Format des Senders ("receiver make right"), wobei der Abflusspfad auf ATM-Blades Frames zu Zellen segmentieren muss und der Abflusspfad auf Frame-Blades das Zusammensetzen von Zellen zu Paketen durchführen muss.

[0031] Zurzeit werden acht Switch-ASICs im Switch verwendet:

- Striper – Der Striper liegt auf dem Portcard- und SCP-IM. Er formatiert die Daten in einen 12-bit-Datenstrom, hängt ein Prüfwort an, spaltet den Datenstrom auf die N Nicht-Ersatz-Fabriken in dem System auf, erzeugt einen Paritätsstreifen derselben Breite wie die Streifen, die zu der anderen Fabrik gehen und sendet die N + 1 Datenströme an die Backplane.
- Unstriper – Der Unstriper ist der andere Portcard-ASIC in der Switch-Architektur. Er empfängt Datenstreifen von allen Fabriken im System. Dann rekonstruiert er den ursprünglichen Datenstrom unter Verwendung von Prüfwort und Paritätsstreifen, so dass Fehler ermittelt und korrigiert werden.
- Aggregator – Der Aggregator übernimmt die Datenströme und Sendewörter von den Stripern und multiplexiert sie zu einem einzigen Eingabestrom an den Speichercontroller.
- Speichercontroller – Der Speichercontroller implementiert die Queueing- und Dequeueing-Mechanismen des Switch. Dazu gehört die eigene "Wide-Memory"-Schnittstelle, wodurch das gleichzeitige En-/De-Queueing mehrerer Zeilen pro Taktzyklus erzielt wird. Die Dequeueing-Seite des Speichercontrollers läuft bei 80 Gbps verglichen mit 40 Gbps zur Bewältigung des Großteils des Queueing, und Formen von Verbindungen erfolgt auf der Portcard.
- Separator – Der Separator implementiert den umgekehrten Arbeitsschritt zum Aggregator. Der Datenstrom vom Speichercontroller wird in mehrere Datenströme demultiplexiert und an einen geeigneten Unstriper-ASIC weitergeleitet. In der Schnittstelle zum Unstriper ist ein Warteschlangen- und Flusskontroll-Handshaking enthalten.

[0032] Die Verbindung zwischen den Fabriken kann man auf 3 verschiedene Arten betrachten: physikalisch, logisch und "aktiv". Physikalisch sind sämtliche Verbindungen zwischen den Portcards und den Fabriken serielle Differential-Pair-Verbindungen mit Gigabit-Geschwindigkeit. Dabei handelt es sich strikt um eine Implementierungsangelegenheit zur Verringerung der Anzahl an Signalen, die über die Backplane laufen. Die "aktive" Perspektive betrachtet die Konfiguration eines einzelnen Switch. Man kann sie als Momentaufnahme davon ansehen, wie Daten in einem gegebenen Moment verarbeitet werden. Die Schnittstelle zwischen dem Fabrik-ASIC auf den Portcards und den Fabriken ist effektiv 12 bits breit. Diese 12 bits sind je nach der Art der Konfiguration der Fabrik-ASICs über 1, 2, 3, 4, 6 oder 12 Fabriken gleichmäßig verteilt ("striped"). Die "aktive" Perspektive betrifft die Anzahl an bits, die von jeder Fabrik bei der gegenwärtigen Konfiguration verarbeitet wird, die genau 12, geteilt durch die Anzahl an Fabriken, ist.

[0033] Die logische Perspektive kann als Vereinigung oder maximale Funktion aller möglichen aktiven Konfigurationen betrachtet werden. Der Fabrik-Slot 1 kann je nach der Konfiguration 12, 6, 4, 3, 2 oder 1 bits der Daten von einem einzigen Stripper verarbeiten und wird daher mit einem 12-bit-Bus gezeichnet. Dagegen kann der Fabrikslot 3 nur zur Verarbeitung von 4, 3, 2 oder 1 bits von einem einzigen Stripper verwendet werden und ist deshalb mit einem 4-bit-Bus gezeichnet.

[0034] Im Gegensatz zu früheren Switches hat der Switch nicht wirklich ein Konzept eines mittels Software steuerbaren Fabrik-Redundanzmodus. Die Fabrik-ASICs implementieren $N + 1$ -Redundanz ohne jegliche Intervention, solange die Ersatz-Fabrik installiert ist.

[0035] Hinsichtlich dessen, was sie bereitstellt, bedeutet $N + 1$ -Redundanz, dass die Hardware ein einzelnes Versagen ohne Datenverlust automatisch ermittelt und korrigiert.

[0036] Die Redundanz arbeitet recht einfach. Damit sie noch leichter verständlich wird, wird ein Spezialfall eines 120G-Switch verwendet mit 3 Fabriken (A, B und C) und einem Ersatz (S). Der Stripper nimmt den 12-bit-Bus und erzeugt zunächst ein Prüfwort, das an die Dateneinheit (Zelle oder Frame) angehängt wird. Dateneinheit und Prüfwort werden dann jeweils für die Fabriken A, B und C in einen Datenstreifen von 4 bit pro Taktzyklus aufgeteilt ($A_3A_2A_1A_0$, $B_3B_2B_1B_0$ und $C_3C_2C_1C_0$). Diese Streifen werden dann zum Erzeugen des Streifens für die Ersatzfabrik $S_3S_2S_1S_0$ verwendet, wobei $S_n = A_n \text{ XOR } B_n \text{ XOR } C_n$, und diese 4 Streifen werden an die entsprechenden Fabriken gesendet. Auf der anderen Seite der Fabriken empfängt der Unstriper 4 4-bit-Streifen von A, B, C und S. Alle möglichen Kombinationen von 3 Fabriken (ABC, ABS, ASC und SBC) werden dann zur Rekonstruktion eines probeweisen 12-bit-Datenstroms verwendet. Dann wird für jeden der 4 Probeströme ein Prüfwort berechnet und dieses dann mit dem Prüfwort am Ende der Dateneinheit verglichen. Wenn kein Fehler bei der Übertragung aufgetreten ist, dann stimmen bei allen 4 Strömen die Prüfwörter überein und der ABC-Strom wird an die Unstriper-Ausgabe weitergeleitet. Ist ein (einziger) Fehler aufgetreten, besteht nur eine Prüfwort-Übereinstimmung. Dann wird der Strom mit der Übereinstimmung off-chip weitergeleitet, und der Unstriper identifiziert den fehlerhaften Fabrikstreifen.

[0037] Für verschiedene Switch-Konfigurationen, d. h. 1, 2, 4, 6 oder 12 Fabriken, ist der Algorithmus gleich, aber die Streifenbreite ändert sich.

[0038] Versagen zwei Fabriken, sind alle durch den Switch laufenden Daten fast sicher korumpiert.

[0039] Die Fabrik-Slots sind nummeriert und müssen in aufsteigender Reihenfolge ausgefüllt werden. Auch die Ersatzfabrik ist ein spezifischer Slot, so dass Ausfüllen der Fabrik-Slots 1, 2, 3 und 4 anders ist als Ausfüllen der Fabrik-Slots 1, 2, 3 und Ersatz. Ersterer ist ein 160G-Switch ohne Redundanz und letzterer 120G mit Redundanz.

[0040] Zunächst werden die ASICs konstruiert und die Backplane angeschlossen, so dass die Verwendung eines bestimmten Portcard-Slot erfordert, dass mindestens eine bestimmte Mindestanzahl von Fabriken installiert wird, wobei der Ersatz ausgenommen ist. Diese Beziehung ist in Tabelle 0 dargestellt.

[0041] Außerdem ist die APS-Redundanz innerhalb des Switch auf spezifisch gepaarte Portcards beschränkt. Die Portcards 1 und 2 sind gepaart, 3 und 4 sind gepaart usw. bis zu den Portcards 47 und 48. D. h., ist APS-Redundanz erforderlich, müssen die gepaarten Slots zusammen ausgefüllt werden.

[0042] Als einfaches Beispiel soll eine Konfiguration mit 2 Portcards und nur 1 Fabrik dienen. Möchte der Nutzer keine APS-Redundanz verwenden, dann können die 2 Portcards in beliebigen zwei der Portcard-Slots 1 bis 4 installiert werden. Ist APS-Redundanz gewünscht, dann müssen die zwei Portcards in den Slots 1 und 2

oder den Slots 3 und 4 installiert werden.

Portcard-Slot	Mindestanzahl an Fabriken
1–4	1
5–8	2
9–12	3
13–16	4
17–24	6
25–48	12

Tabelle 0: Fabrik-Anforderungen für die Portcard-Slot-Nutzung

[0043] Zu Erhöhung der Kapazität wird (werden) die neue(n) Fabrik(en) hinzugefügt. Man wartet, bis der Switch die Veränderung erkennt, und rekonfiguriert das System, so dass Streifen über die neue Anzahl an Fabriken gebildet werden. Die neuen Portcards werden installiert.

[0044] Man beachte, dass es technisch nicht notwendig ist, dass man über alle 4 Portcards pro Fabrik verfügt. Der Switch arbeitet ordnungsgemäß mit 3 installierten Fabriken und einer einzigen Portcard im Slot 12. Dies ist zwar nicht kostengünstig, funktioniert aber.

[0045] Zum Entfernen von Kapazität wird das Verfahren zum Erhöhen der Kapazität umgekehrt.

[0046] D. h. ist der Switch überbelegt, installiert man 8 Portcards und nur eine Fabrik.

[0047] Dies sollte nur infolge einer falschen Erweiterung des Switch oder durch irgendein Systemversagen auftreten. Je nachdem, wie diese Situation entsteht, tritt eine von zwei Möglichkeiten auf. Ist der Switch als 40G-Switch konfiguriert und werden die Portcards vor der Fabrik hinzugefügt, sind die 5. bis 8. Portcard tot. Ist der Switch als nicht-redundanter 80G-Switch konfiguriert und versagt die zweite Fabrik oder wird entfernt, dann werden alle Daten durch den Switch korumpiert (unter der Annahme, dass die Ersatzfabrik nicht installiert ist). Nur der Vollständigkeit halber sei erwähnt, dass bei Installation von 8 Portcards in einem redundanten 80G-Switch bei Versagen oder Entfernen der zweiten Fabrik der Switch weiterhin normal arbeitet, wobei der Ersatz die ausgefallene/entfernte Fabrik abdeckt.

[0048] [Fig. 1](#) zeigt die Paketerlegung in Streifen im Switch.

[0049] Der Chipset unterstützt ATM- und POS-Portcards in OC48- und OC192c-Konfigurationen. OC48-Portcards bilden Schnittstellen zu den Switching-Fabriken mit vier getrennten OC48-Strömen. OC192-Portcards kombinieren die 4 Kanäle logisch zu einem 10G-Strom. Die Zuflussseite einer Portcard führt keine Verkehrsumwandlungen durch für Verkehr, der zwischen ATM-Zellen und Paketen wechselt. Empfangener Verkehr in gleich welcher Form wird an die Switch-Fabriken gesendet. Die Switch-Fabriken mischen Pakete und Zellen und bauen dann ein Gemisch von Paketen und Zellen aus der Warteschlange auf die Abflussseite einer Portcard ab.

[0050] Die Abflussseite einer Portcard ist für die Umwandlung des Verkehrs in das geeignete Format für den Ausgabe-Port verantwortlich. Diese Konvention wird im Kontext des Switch als Senden von Daten im Format des Senders ("receiver makes right") bezeichnet. Ein Zellen-Blade ist für die Segmentierung von Paketen sowie für das Zusammensetzen von Zellen zu Paketen verantwortlich. Damit die Fabrikbeschleunigung unterstützt wird, unterstützt die Abflussseite der Portcard eine Verbindungsbandbreite, die dem doppelten der Eingangsseite der Portcard entspricht.

[0051] [Fig. 2](#) zeigt ein Blockdiagramm einer ATM-Portcard auf Poseidon-Basis. Jeder 2,5G-Kanal besteht aus 4 ASICs: Ein Eingangs-TM- und ein Stripper-ASIC auf der Eingangsseite und ein Unstriper-ASIC und ein Ausgangs-TM-ASIC auf der Ausgangsseite.

[0052] Auf der Eingangsseite sind OC-48c- oder 4 OC-12c-Schnittstellen aggregiert. Jeder Vortex sendet einen 2,5G-Zellenstrom in einen zugewiesenen Stripper-ASIC (mit dem BIB-Bus, wie weiter unten beschrieben). Der Stripper wandelt das zuführte Sendewort in zwei Teile um. Ein Teil des Sendeworts wird zur Fabrik geleitet,

so dass der (die) Ausgabeport(s) für die Zelle festgelegt wird (werden). Das gesamte Sendewort wird zudem auf dem Datenteil des Busses als Sendewort für die Verwendung durch den Ausgangs-Speichercontroller geschickt. Das erste Sendewort wird als "Fabrik-Sendewort" bezeichnet und das Sendewort für den Ausgangs-Speichercontroller als "Abflusssendewort".

[0053] Auf der Ausgangsseite übernimmt der Unstriper-ASIC in jedem Kanal Verkehr von jeder Portcard, überprüft die Daten auf Fehler und korrigiert sie und sendet dann korrekte Pakete auf seinem Ausgabebus nach außen. Der Unstriper verwendet Daten der Ersatz-Fabrik und die vom Stripper eingesetzte Prüfsumme zum Finden und Korrigieren von Datenkorruption.

[0054] [Fig. 2](#) zeigt eine OC48-Portcard.

[0055] Die OC192-Portcard unterstützt einen einzelnen 10G-Strom zur Fabrik und zwischen einem 10G und einem 20G-Abflusstrom. Diese Festplatte verwendet zudem 4 Stripper und 4 Unstriper, aber die 4 Chips arbeiten parallel auf einem breiteren Datenbus. Die an jede Fabrik gesendeten Daten sind für OC48- und OC192-Portcards identisch, so dass Daten zwischen dem Port-Typen strömen können, ohne dass spezielle Umwandlungsfunktionen nötig sind.

[0056] [Fig. 3](#) zeigt ein verkettetes 10G-Netzwerk-Blade.

[0057] Jede 40G-Switch-Fabrik stellt bis zu 40 Gbps Zellen/Frames in die Warteschlange und baut sie mit 80 Gbps ab. Diese 2fache Beschleunigung verringert die Menge an Verkehr, der an der Fabrik abgepuffert wird, und bewirkt, dass der Ausgangs-ASIC Verkehrsspitzen deutlich oberhalb der Line Rate bewältigt. Eine Switch-Fabrik besteht aus drei Arten von ASICs: Aggregatoren, Speichercontroller und Separatoren. Neun Aggregator-ASICs empfangen 40 Gbps an Verkehr von bis zu 48 Netzwerk-Blades und dem Kontroll-Port. Die Aggregator-ASICs vereinigen das Fabriksendewort und den Nutzdatenanteil zu einem einzigen Datenstrom und TDM zwischen seinen Quellen und platzieren die erhaltenen Daten auf einem breiten Ausgabebus. Mit einem zusätzlichen Kontrollbus (Destid) wird kontrolliert, wie die Speichercontroller die Daten in die Warteschlange stellen. Der Datenstrom von jedem Aggregator-ASIC wird dann nach bit in 12 Speichercontroller aufgeteilt.

[0058] Der Speichercontroller empfängt bis zu 16 Zellen/Frames in jedem Taktzyklus. Jeder der 12 ASICs speichert 1/12 der aggregierten Datenströme. Dann speichert er die eingehenden Daten auf Basis der Kontrollinformation, die auf dem Destid-Bus empfangen wird. Das Speichern dieser Daten ist leichter in dem Speichercontroller, der Paketgrenzen vergleichsweise wenig registriert (Cache-Zeilen-Konzept). Alle 12 ASICs bauen die gespeicherten Zellen gleichzeitig mit einer aggregierten Geschwindigkeit von 80 Gbps aus der Warteschlange ab.

[0059] Neun Separator-ASICs führen die umgekehrte Funktion zu den Aggregator-ASICs aus. Jeder Separator empfängt Daten von allen 12 Speichercontrollern und decodiert die Sendewörter, die durch den Aggregator in die Datenströme eingebettet wurden, so dass Paketgrenzen gefunden werden. Jeder Separator-ASIC sendet dann die Daten an bis zu 24 verschiedene Unstriper je nach dem genauen Bestimmungsort, der vom Speichercontroller angegeben wurde, während die Daten zum Separator geschickt wurden.

[0060] Das Dequeueing-Verfahren wird durch den Rückstau angetrieben. Wird Rückstau auf den Unstriper ausgeübt, wird der Rückstau zum Separator zurückgemeldet. Der Separator und die Speichercontroller haben ebenfalls einen Rückstaumechanismus, der regelt, wann ein Speichercontroller Verkehr zu einem Ausgabe-Port aus der Warteschlange abbauen kann.

[0061] Zur effizienten Unterstützung von OC48 und OC192 im Chipset werden die 4 OC48-Ports von einer Portcard immer zum gleichen Aggregator und vom gleichen Separator gesendet (die Port-Verbindungen für den Aggregator und den Separator sind immer symmetrisch).

[0062] Die [Fig. 4a](#) und [Fig. 4b](#) zeigen die Konnektivität der Fabrik-ASICs.

[0063] Die externen Schnittstellen der Switches sind der Eingabebus (BIB) zwischen dem Stripper-ASIC und dem Zufluss-Blade-ASIC, wie Vortex, und der Ausgabebus (BOB) zwischen dem Unstriper-ASIC und dem Abfluss-Blade-ASIC, wie Trident.

[0064] Der Stripper-ASIC akzeptiert Daten vom Zufluss-Port über den Eingabebus (BIB) (auch als

DIN_ST_bl_ch_-Bus bezeichnet).

[0065] Dieser Bus kann entweder als 4 getrennte 32-bit-Eingabebusse ($4 \times \text{OC48c}$) oder ein einziger breiter 128-bit-Datenbus mit einem gemeinsamen Satz von Kontrollleitungen zu allen Stripern arbeiten. Dieser Bus unterstützt je nach der Softwarekonfiguration des Strip-Chips entweder Zellen oder Pakete.

[0066] Der Unstriper-ASIC sendet Daten an den Abfluss-Port über den Ausgabebus (BOB) (auch als DOUT_UN_bl_ch_-Bus bezeichnet), der ein 64-(oder 256-)bit-Datenbus ist, der Zelle oder Paket unterstützen kann. Er besteht aus folgenden Signalen:

Dieser Bus kann entweder als 4 getrennte 32-bit-Ausgabebusse ($4 \times \text{OC48c}$) oder ein einziger breiter 128-bit-Datenbus mit einem gemeinsamen Satz von Kontrollleitungen zu allen Unstripers arbeiten. Dieser Bus unterstützt je nach der Softwarekonfiguration des Unstriper-Chips entweder Zellen oder Pakete.

[0067] Der Synchronisator hat zwei Hauptaufgaben. Die erste Aufgabe ist die Aufrechterhaltung der logischen Zelle-/Paket- oder Datengramm-Ordnung über alle Fabriken. An der Fabrikzuflussschnittstelle müssen Datengramme, die von den Kanälen einer Portcard an mehr als einer Fabrik eintreffen, in der gleichen Reihenfolge über alle Fabriken verarbeitet werden. Die zweite Aufgabe des Synchronisators ist, dass er den Abflusskanal einer Portcard alle Segmente oder Streifen eines Datengramms wieder zusammensetzen lässt, die zusammen gehören, obwohl die Datengrammsegmente von mehr als einer Fabrik gesendet werden und an den Abflusseingängen des Blade zu unterschiedlichen Zeitpunkten eintreffen können. Dieser Mechanismus muss in einem System aufrechterhalten werden, das unterschiedliche Nettoverzögerungen und variable Ausmaße an Taktverschiebung zwischen Blades und Fabriken aufweist.

[0068] Der Switch verwendet ein System synchronisierter Fenster, in denen Startinformation über das System übertragen wird. Jeder Transmitter und Empfänger kann auf relative Taktzahlen von der letzten Resynchronisierungsanzeige blicken, so dass Daten von mehreren Quellen synchronisiert werden. Der Empfänger verzögert den Empfang von Daten, die der erste Taktzyklus von Daten in einem Synchronisierungszeitraum sind, bis zu einer programmierbaren Verzögerung, nachdem er die globale Synchronisierungsanzeige erhalten hat. An diesem Punkt werden die Daten so betrachtet, als seien sie alle gleichzeitig empfangen worden, und es gilt eine feste Reihenfolge. Obwohl die Verzögerungen für Paket 0 und Zelle 0 dazu führten, dass sie an den Empfängern aufgrund von Verzögerungen durch die Box in unterschiedlichen Reihenfolgen gesehen wurden, ist die erhaltene Reihenfolge beider Ströme bei der Empfangszeit = 1 gleich, Paket 0, Zelle 0 auf Basis des physikalischen Busses, auf dem sie empfangen wurden.

[0069] Mehrere Zellen oder Pakete können bei einem Zählerticken gesendet werden. Alle Bestimmungsorte ordnen alle Zellen von der ersten Schnittstelle, bevor sie auf die zweite Schnittstelle bewegt werden usw. Die Zellsynchronisierungstechnik wird an allen Zellenschnittstellen verwendet. Unterschiedliche Auflösungen sind an einigen Schnittstellen erforderlich.

[0070] Der Synchronisator besteht grundsätzlich aus zwei Hauptblöcken, dem Transmitter und dem Empfänger. Der Transmitterblock liegt in den Strip- und Separator-ASICs und der Empfängerblock in den Aggregator- und Unstriper-ASICs. Der Empfänger im Aggregator handhabt bis zu 24 ($6 \text{ Portcards} \times 4 \text{ Kanäle}$) Eingabe-Lanes. Der Empfänger im Unstriper handhabt bis zu 13 ($12 \text{ Fabriken} \times 1 \text{ Ausgleichsfabrik}$) Eingabe-Lanes.

[0071] Wird ein Synchronisierungsimpuls empfangen, berechnet der Transmitter zunächst die Anzahl an Taktzyklen, der er schneller ist (als N Takte bezeichnet).

[0072] Der Übertragungssynchronisator unterbricht den Ausgabestrom und überträgt N K-Zeichen, die anzeigen, dass er sperrt. Am Ende der Sperrphase überträgt der Transmitter ein K-Zeichen, das anzeigt, dass die gültigen Daten beim nächsten Taktzyklus beginnen. Diese Anzeige, dass der nächste Zyklus gültig ist, wird von den Empfängern zur Synchronisierung des Verkehrs von allen Quellen verwendet.

[0073] Am nächsten Ende der Übertragung setzt der Transmitter mindestens ein Idle-Symbol an der Schnittstelle ein. Mithilfe dieser Idle-Symbole können die 10-bit-Decodierer sich wieder richtig mit dem seriellen 10-bit-Code-Fenster synchronisieren, wenn sie aus der Synchronisierung herausfallen.

[0074] Der Empfangssynchronisator empfängt den globalen Synchronisierungsimpuls und verzögert den Synchronisierungsimpuls um eine programmierte Zahl (die je nach dem maximalen Ausmaß an Transportverzögerung programmiert ist, die eine physikalische Box haben kann). Nach Verzögern des Synchronisierungsimpulses betrachtet der Empfänger dann den Taktzyklus unmittelbar nach dem Synchronisierungszeichen als

annehmbar für den Empfang. Dann werden Daten in jedem Taktzyklus empfangen, bis das nächste Synchronisierungszeichen auf dem Eingabestrom entdeckt wird. Diese Daten werden als nicht annehmbar für den Empfang angesehen, bis der verzögerte globale Synchronisierungsimpuls entdeckt wird.

[0075] Weil sich Transmitter und Empfänger auf physikalisch verschiedenen Platten befinden und von verschiedenen Oszillatoren getaktet werden, bestehen zwischen ihnen Taktgeschwindigkeitsunterschiede. Zur Beschränkung der Anzahl an Taktzyklen zwischen verschiedenen Transmittern und Empfängern wird ein globaler Synchronisierungsimpuls auf Systemebene zur Resynchronisierung aller Sequenzzähler verwendet. Jeder Chip ist so programmiert, dass er sicherstellt, dass unter allen gültigen Taktversätzen jeder Transmitter und Empfänger denkt, dass er mindestens einen Taktzyklus schneller ist. Jeder Chip wartet dann auf die passende Anzahl an Taktzyklen, bis sie sich in ihrem derzeitigen Sync-Impuls-Fenster befinden. Dies gewährleistet, dass alle Quellen N^* gültige Sync-Impuls-Fenster-Taktzyklen zwischen den Synchronisierungsimpulsen vergehen lassen.

[0076] Zum Beispiel kann das Synchronisierungsimpulsfenster auf 100 Takte programmiert werden. Die Synchronisierungsimpulse können mit einer nominellen Rate von einem Synchronisierungsimpuls alle 10000 Takte gesendet werden. Im schlimmsten Fall einer Verschiebung der Synchronisierungsimpulstransmittertake und der Synchronisierungsimpulsempfängertakte kann es tatsächlich 9995 bis 10005 Takte am Empfänger pro 10000 Takte auf dem Synchronisierungsimpulstransmitter geben. In diesem Fall wird ein Synchronisierungsimpulstransmitter derart programmiert, dass er alle 10006 Taktzyklen Synchronisierungsimpulse aussendet. Die 10006 Takte gewährleisten, dass sich alle Empfänger in ihrem nächsten Fenster befinden müssen. Ein Empfänger mit einem schnellen Takt kann tatsächlich 10012 Takte gesehen haben, wenn der Synchronisierungsimpulstransmitter einen langsamen Takt hat. Weil der Synchronisierungsimpuls 12 Taktzyklen in das nächste Synchronisierungsimpulsfenster hinein empfangen wurde, verzögert der Chip 12 Taktzyklen lang. Ein anderer Empfänger kann 10006 Takte gesehen haben und für 6 Taktzyklen am Ende des Synchronisierungsimpulsfensters sperren. In beiden Fällen hat jede Quelle 10100 Taktzyklen durchlaufen.

[0077] Ist eine Portcard oder Fabrik nicht vorhanden oder wurde gerade eingesetzt und wird von einer angenommen, dass sie die Eingaben von einem Empfangssynchronisator antreibt, wird das Schreiben von Daten an den bestimmten Eingabe-FIFO verhindert, weil der Eingabetakt nicht vorhanden oder instabil und der Status der Datenzeilen unbekannt ist. Wenn die Portcard oder Fabrik eingesetzt wird, muss Software hinzukommen und die Eingabe in die byte-Lane freigeben, so dass Daten von dieser Quelle freigegeben sind. Schreiben an den Eingabe-FIFO wird freigegeben. Es wird angenommen, dass das Freigabesignal bestätigt wird, nachdem die Daten, das Sendewort und der Takt von der Portcard oder Fabrik stabil sind.

[0078] Auf Systemebene gibt es einen primären und einen sekundären Synchronisierungsimpulstransmitter, die sich auf zwei getrennten Fabriken befinden. Es gibt auch einen Synchronisierungsimpulsempfänger auf jeder Fabrik und jedem Blade. Dies ist aus [Fig. 5](#) ersichtlich. Ein primärer Synchronisierungsimpulstransmitter ist ein freier Synchronisierungsimpulsgenerator, und ein sekundärer Synchronisierungsimpulstransmitter synchronisiert seine Synchronisierungsimpulse mit dem primären. Die Synchronisierungsimpulsempfänger empfangen primäre und sekundäre Synchronisierungsimpulse und wählen auf Basis eines Fehlerprüfalgorithmus den korrekte Synchronisierungsimpuls zum Weiterleiten an die ASICs, die sich auf der Festplatte befinden. Der Synchronisierungsimpulsempfänger stellt sicher, dass ein Synchronisierungsimpuls nur dann zum Rest der Festplatte weitergeleitet wird, wenn der Synchronisierungsimpuls von den Synchronisierungsimpulstransmittern in seine eigene Sequenz-"0"-Zählung fällt. Zum Beispiel liegen der Synchronisierungsimpulsempfänger und ein Unstriper-ASIC auf dem gleichen Blade. Der Synchronisierungsimpulsempfänger und der Empfangssynchronisator im Unstriper werden von dem gleichen Kristall-Oszillator getaktet, so dass zwischen den Takten, die zum inkrementellen Erhöhen der internen Sequenzzähler verwendet werden, keine Taktverschiebung vorhanden sein sollte. Der Empfangssynchronisator verlangt, dass der Synchronisierungsimpuls, den er empfängt, immer im "0"-Zählungsfenster liegt.

[0079] Stellt der Synchronisierungsimpulsempfänger fest, dass der primäre Synchronisierungsimpulstransmitter nicht mehr synchronisiert ist, schaltet er um zum sekundären Synchronisierungsimpulstransmitter. Der sekundäre Synchronisierungsimpulstransmitter stellt ebenfalls fest, dass der primäre Synchronisierungsimpulstransmitter nicht mehr synchronisiert ist, und beginnt mit der Erzeugung seines eigenen Synchronisierungsimpulses unabhängig von dem primären Synchronisierungsimpulstransmitter. Dies ist die Arbeitsweise des sekundären Synchronisierungsimpulstransmitters. Stellt der Synchronisierungsimpulsempfänger fest, dass der primäre Synchronisierungsimpulstransmitter wieder synchronisiert ist, schaltet er zur primären Seite. Der sekundäre Synchronisierungsimpulstransmitter stellt ebenfalls fest, dass der primäre Synchronisierungsimpulstransmitter wieder synchronisiert ist, und schaltet wieder in den sekundären Modus. Im sekundären Modus syn-

chronisiert er seinen eigenen Synchronisierungsimpuls mit dem primären Synchronisierungsimpuls. Der Synchronisierungsimpulsempfänger hat weniger Toleranz in seinem Synchronisierungsimpuls-Filtermechanismus als der sekundäre Synchronisierungsimpulstransmitter. Der Synchronisierungsimpulsempfänger schaltet schneller um als der sekundäre Synchronisierungsimpulstransmitter. Dies erfolgt, damit gewährleistet ist, dass alle Empfängersynchronisatoren auf die Verwendung der sekundären Synchronisierungsimpulstransmitterquelle umgeschaltet haben, bevor der sekundäre Synchronisierungsimpulstransmitter zum primären Modus umschaltet.

[0080] [Fig. 5](#) zeigt die Synchronisierungsimpulsverteilung.

[0081] Zum Sperren der Backplane-Übertragung von einer Fabrik anhand der Anzahl an Taktzyklen, die in der Synchronisierungsberechnung angegeben wird, muss die gesamte Fabrik für so viele Taktzyklen effektiv sperren, dass gewährleistet ist, dass die gleichen Queueing- und Dequeueing-Entscheidungen synchron bleiben. Dies benötigt Unterstützung in jeder der Fabrik-ASICs. Sperren beendet alle Funktionen, einschließlich Funktionen, wie Warteschlagenresynchronisierung.

[0082] Das Synchronisierungssignal von dem Synchronisierungsimpulsempfänger wird an alle ASICs weitergeleitet. Jeder Fabrik-ASIC enthält einen Zähler in der Kerntaktdomäne, der Taktzyklen zwischen globalen Synchronisierungsimpulsen zählt. Nach Empfang des Synchronisierungsimpulses berechnet jeder ASIC die Anzahl an Taktzyklen, die er schneller ist. Weil das globale Sync nicht mit seinem eigenen Takt übertragen wird, ist der berechnete Sperrzykluswert möglicherweise nicht für alle ASICs auf derselben Fabrik gleich. Dieser Unterschied wird berücksichtigt, indem alle Schnittstellen-FIFOs in einer Tiefe gehalten werden, in der sie den maximalen Versatz an Sperrtakten tolerieren können.

[0083] Sperrzyklen auf allen Chips werden immer an dem gleichen logischen Punkt in Bezug zum Beginn der letzten Sequenz "brauchbarer"(Nicht-Sperr-)Zyklen eingesetzt. D. h. jeder Chip führt immer die gleiche Anzahl an "brauchbaren" Zyklen zwischen Sperrereignissen aus, auch wenn die Anzahl an Sperrzyklen variiert.

[0084] Sperren kann auf verschiedenen Chips zu unterschiedlichen Zeiten erfolgen. Alle Fabrik-Eingabe-FIFOs werden zu Beginn derart eingestellt, dass Sperren zunächst auf jeder Seite des FIFO erfolgen kann, ohne dass der FIFO trocken- oder überläuft. An jeder Chip-Chip-Schnittstelle gibt es einen Synchronisierung-FIFO zum Ausgleichen von Sperrzyklen (sowie Festplatten-Trace-Längen und Taktversätzen). Die Transmittersignale sperren, während er gesperrt ist. Der Empfänger führt während angegebener Zyklen keinen Push-Schritt aus und keinen Pop-Schritt, während er selbst gesperrt ist. Die FIFO-Tiefe variiert je nachdem, welcher Chip zuerst sperrt. Die Variation ist jedoch durch die maximale Anzahl an Sperrzyklen beschränkt. Die Anzahl an Sperrzyklen, die ein bestimmter Chip während eines globalen Synchronisierungszeitraums sieht, kann variieren. Sie haben aber alle die gleiche Anzahl an brauchbaren Zyklen. Die Gesamtzahl an Sperrzyklen, die jeder Chip auf einer bestimmten Fabrik sieht, ist innerhalb von Toleranzgrenzen gleiche.

[0085] Die Aggregator-Kerntaktdomäne stoppt während der Sperrdauer vollständig – alle Flops und Speicher halten ihren Status. Die Eingabe-FIFOs können sich aufbauen. Sperrbuszyklen werden in die Ausgabenschlangen eingesetzt. Wann genau die Kernabspernung ausgeführt wird, ist dadurch vorgegeben, wann das DOUT_AG-Busprotokoll das Einfügen von Sperrzyklen zulässt. DOUT_AG-Sperrzyklen werden auf dem DestID-bus angegeben.

[0086] Der Speichercontroller muss alle Flops für eine angemessene Anzahl an Zyklen sperren. Um den Einfluss auf den Siliziumbereich im Speichercontroller zu verringern, wird eine als Fortpflanzungsspernung (propagated lockdown) bezeichnete Technik verwendet.

[0087] Die Chip-Chip-Synchronisierung in der Fabrik erfolgt bei jedem Synchronisierungsimpuls. Zwar müssen einige ASICs über eine gewisse Synchronisierungsfehler-Ermittlungsfähigkeit verfügen, aber es ist die Aufgabe des Unstripers, Fabriksynchronisierungsfehler aufzudecken und die störende Fabrik zu beseitigen. Die Chip-Chip-Synchronisierung ist eine Kaskadenfunktion, die erfolgt, bevor ein Paketstrom auf der Fabrik zugelassen wird. Die Synchronisierung fließt vom Aggregator zum Speichercontroller, zum Separator und zurück zum Speichercontroller. Nach Zurücksetzen des Systems wartet der Aggregator auf das erste globale Synchronisierungssignal. Nach Empfang überträgt der Aggregator einen lokalen Synchronisierungsbefehl (Wert 0×2) auf dem DestID-Bus an jeden Speichercontroller.

[0088] Die Striping-Funktion weist bits von eingehenden Datenströmen einzelnen Fabriken zu. Zwei Punkte wurden beim Ableiten der Striping-Zuweisung optimiert:

1. Die Backplane-Effizienz sollte für OC48 und OC192 optimiert werden.
2. Die Backplane-Verschaltung sollte für den OC192-Betrieb nicht signifikant verändert sein.

[0089] Diese wurden gegen zusätzliche Multiplexierungsschenkel für die Stripper- und Unstriper-ASICs ausbalanciert. Ungeachtet der Optimierung muss der Switch im Speichercontroller für OC48 und OC192 das gleiche Datenformat haben.

[0090] Die Backplane-Effizienz erfordert, dass bei der Bildung der Backplane-Busse minimales Padding hinzugefügt wird. Für den 12-bit-Backplane-Bus für OC48 und den 48-bit-Backplane-Bus für OC192 erfordert eine optimale Zuweisung, dass die Anzahl unbenutzter bits für eine Übertragung gleich $(\text{Anzahl bytes} \cdot 8) / \text{Busbreite}$ ist, wobei "/" eine ganzzahlige Division ist. Bei OC48 kann der Bus 0, 4 oder 8 unbenutzte bits haben. Bei OC192 kann der Bus 0, 8, 16, 24, 32 oder 40 8 unbenutzte bits haben.

[0091] Das bedeutet, dass kein bit zwischen 12-bit-Grenzen verschoben werden kann oder dass das OC48-Padding für bestimmte Pakettlängen nicht optimal ist.

[0092] Bei OC192 bedeutet maximale Bandbreitennutzung, dass jeder Stripper die gleiche Anzahl an bits empfangen muss (was bit-Verschränkung in die Stripper hinein impliziert). Kombiniert mit derselben Backplane-Verschaltung impliziert dies, dass bei OC192 jeder Streifen genau die richtige Anzahl an bits haben muss, die von jedem Stripper kommen, der $\frac{1}{4}$ der bits aufweist.

[0093] Für das Zuweisen von Daten-bits an Fabriken wird ein 48-bit-Frame verwendet. Im Inneren des Strippers ist ein FIFO, der 32 bits breit bei 80–100 MHz geschrieben und 24 bits breit bei 125 MHz gelesen wird. Drei 32-bit-wörter ergeben vier 24-bit-Wörter. Jedes Paar von 24-bit-Wörtern wird als 48-bit-Frame behandelt. Die Zuweisungen zwischen bits und Fabriken hängen von der Anzahl der Fabriken ab.

Tabelle 11: bit-Striping-Funktion

		Fab 0	Fab 1	Fab 2	Fab 3	Fab 4	Fab 5	Fab 6	Fab 7	Fab 8	Fab 9	Fab 10	Fab 11
	0:11	0:11											
1 fab	12:2 3	12:2 3											
	24:3 5	24:3 5											
	36:4 7	36:4 7											
	0:11	0, 2, 5, 7, 8, 10	1,3,4 , 6,9,1 1										
2 fab	12:2 3	13, 15, 16, 18, 21	12, 14, 17, 19 20, 22										
	24:3 5	+24 bis 0:11	+24 bis 0:11										
	36:4 7	+24 bis 12:2 3	+24 bis 12:2 3										
	0:11	0, 3, 5, 10	2, 4, 7, 9	1, 6, 8, 11									

3 fab	12:2 3	15, 17, 22, 13	14, 16, 19, 21	13, 18, 20, 23									
	24:3 5	+24 bis 0:11	+24 to 0:11	24 bis 0:11									
	36:4 7	+24 bis 12:2 3	+24 to 12:2 3	+24 bis 12:2 3									
	0:11	0, 5, 10	3, 4, 9	2, 7, 8	1, 6, 11								
4 fab	12:2 3	15, 16, 21	14, 19, 20	13, 18, 23	12, 17, 22								
	24:3 5	26, 31, 32	25, 30, 35	24, 29, 34	27, 28, 33								
	36:4 7	37, 42, 47	36, 41, 46	39, 40, 43	38, 43, 44								
	0, 11	0, 11	1, 4	5, 8	2, 9	3, 6	7, 10						
6 fab	12:2 3	14, 21	15, 18	19, 22	12, 23	13, 16	17, 20						
	24:3 5	+24 bis 0:11											
		+ 24 bis 12:2 3											
	0:11	0	4	8	1	5	9	2	6	10	3	7	11
12 fab	12:2 3	15	19	23	12	16	20	13	17	21	14	18	22
	24:3 5	26	30	34	27	31	35	24	28	32	25	29	33
	36:4 7	37	41	45	38	42	46	39	43	47	37	40	44

[0094] Die folgende Tabelle gibt die byte-Lanes an, die im Aggregator zuerst gelesen und zu den ersten im Separator geschrieben werden. Die vier Kanäle sind als A, B, C, D bezeichnet. Die verschiedenen Fabriken haben eine unterschiedliche Lese-/Schreib-Reihenfolge der Kanäle, so dass alle Busse vollständig genutzt werden können.

Eine-Fabrik-40G

[0095] Die folgende Tabelle zeigt die Schnittstellenlesereihenfolge für den Aggregator.

Fabrik	1.	2.	3.	4.
0	A	B	C	D
Par	A	B	C	D

Zwei-Fabriken-80G

Fabrik	1.	2.	3.	4.
0	A	C	B	D
1	B	D	A	C
Par	A	C	B	D

120G

Fabrik	1.	2.	3.	4.
0	A	D	B	C
1	C	A	D	B
2	B	C	A	D
Par	A	D	B	C

Drei-Fabriken-160G

Fabrik	1.	2.	3.	4.
0	A	B	C	D
1	D	A	B	C
2	C	D	A	B
3	B	C	D	A
Par	A	B	C	D

Sechs-Fabriken-240G

Fabrik	1.	2.	3.	4.
0	A	D	C	B
1	B	A	D	C
2	B	A	D	C
3	C	B	A	D
4	D	C	B	A
5	D	C	B	A
Par	A	C	D	B

Zwölf-Fabriken-480G

Fabrik	1.	2.	3.	4.
0, 1, 2	A	D	C	B
3, 4, 5	B	A	D	C
6, 7, 8	C	B	A	D
9, 10, 1	D	C	B	A
Par	A	B	C	D

[0096] Schnittstellen zu dem Gigabit-Transceivern verwenden den Transceiver-Bus als geteilten Bus mit zwei getrennten Sendewort- und Datenbussen. Der Sendewortbus hat eine feste Größe (2 bit für OC48-Zufluss, 4 bit für OC48-Abfluss, 8 bit für OC192- Zufluss und 16 bit für OC192- Abfluss), der Datenbus hat eine variable

Größe. Die Übertragungsreihenfolge hat immer Sendewort-bits an festen Stellen. Jede Striping-Konfiguration hat einen Transceiver, der zur Kommunikation mit einem Bestimmungsort in allen gültigen Konfigurationen verwendet wird. Dieser Transceiver wird zum Senden beider Sendewortbusse und zum Starten der Datenübertragung verwendet.

[0097] Die Backplane-Schnittstelle wird physikalisch unter Verwendung von Schnittstellen zu den Backplane-Transceivern implementiert. Der Bus für Zu- und Abfluss wird so betrachtet, als bestehe er aus zwei Hälften, jeweils mit Sendewortdaten. Die zwei Bushälften können Information über getrennte Pakete haben, wenn die erste Bushälfte ein Paket beendet.

[0098] Zum Beispiel hat eine zu den Fabriken gehende OC48-Schnittstelle lokal gesprochen 24 Daten-bits und 2 Sendewort-bits. Dieser Bus wird derart verwendet, als sei er $2 \times (12\text{-bit-Datenbus} + 1\text{ bit Sendewortbus})$. Die zwei Bushälften werden als A und B bezeichnet. Bus A sind die ersten Daten, darauf folgt Bus B. Ein Paket kann auf Bus A oder B beginnen und auf Bus A oder B enden.

[0099] Bei Abbilden von Daten-bits und Sendewort-bits auf Transceiver-bits werden die Bus-bits verschränkt. Dies gewährleistet, dass alle Transceiver den gleichen gültig/ungültig-Status haben, auch wenn das Striping-Ausmaß sich verändert. Sendewörter sollte so interpretiert werden, dass Bus A vor Bus B erscheint.

[0100] Das BusA-/BusB-Konzept entspricht nahezu Schnittstellen zwischen den Chips.

[0101] Alle Backplane-Busse unterstützen Datenfragmentierung. Das verwendete Protokoll markiert die letzte Übertragung (über das Endsegment-bit im Sendewort). Alle Übertragungen, die nicht Endsegment sind, müssen die gesamte Busbreite nutzen, sogar wenn keine gerade Anzahl an bytes vorliegt. Jedes gegebene Paket muss für alle Übertragungen dieses Pakets an dieselbe Anzahl von Fabriken in Streifen zerlegt werden. Wird das Striping-Ausmaß während der Übertragung eines Pakets im Striper aktualisiert, wird nur das Striping bei Beginn des nächsten Pakets aktualisiert.

[0102] Jeder Transmitter auf den ASICs hat die folgenden I/O für jeden Kanal:

8-bit-Datenbus, 1-bit-Takt, 1-bit-Kontrolle

Auf der Empfangsseite empfängt der ASIC pro Kanal
einen Empfangstakt, 8-bit-Datenbus, 3-bit-Statusbus

[0103] Der Switch optimiert die Transceiver, indem ein Transmitter auf zwischen 1 und 3 Backplane-Paare und jeder Empfänger mit zwischen 1 und 3 Backplane-Paaren abgebildet wird. Dies lässt nur genügend Transmitter, dass der benötigte Verkehr in einer Konfiguration unterstützt wird, die auf der Festplatte aufgefüllt werden soll, wobei ein vollständiger Satz von Backplane-Netzen aufrechterhalten wird. Die Motivation für diese Optimierung war die Verringerung der Anzahl der benötigten Transceiver.

[0104] Nachdem die Optimierung erfolgt war, war immer noch erforderlich, dass jederzeit zwei verschiedene Striping-Ausmaße in den Gigabit-Transceivern unterstützt werden. Dadurch kann der Verkehr von Striping-Daten an eine Fabrik und Striper-Striping-Daten an zwei Fabriken gleichzeitig in die Warteschlange gestellt werden.

[0105] Je nach der Buskonfiguration müssen gegebenenfalls mehrere Kanäle verkettet werden, so dass eine Pipe mit größerer Bandbreite gebildet wird (jederzeit mehr als ein Transceiver in einer logischen Verbindung steht). Zwar können Quad-Gbit-Transceiver 4 Kanäle zusammenbinden, aber diese Funktion wird nicht verwendet. Stattdessen ist der Empfangs-ASIC für die Synchronisierung zwischen den Kanälen von einer Quelle verantwortlich. Dies erfolgt in dem gleichen Kontext wie der generische Synchronisierungsalgorithmus.

[0106] Die 8b/10b-Codierung/Decodierung in den Gigabit-Transceivern ermöglicht das Senden einer Reihe von Kontrollereignissen über den Kanal. Die Bezeichnung für diese Kontrollereignisse sind K-Zeichen. Sie werden auf Basis des codierten 10-bit-Wertes nummeriert. Mehrere dieser K-Zeichen werden im Chipset verwendet. Die verwendeten K-Zeichen und ihre Funktionen sind in der folgenden Tabelle angegeben.

Tabelle 12: K-Zeichen-Verwendung

K-Zeichen Funktion	Bemerkungen
28.0 Sync-Anzeige	nach Sperrzyklen übertragen, als oberstes Synchronisierungsereignis an den Empfängern behandelt
28.1 Sperren	während Sperrzyklen auf der Backplane übertragen
28.2 Paket-Abbruch	wird übertragen, um anzuzeigen, dass die Card das derzeitige Paket nicht beenden kann. Derzeitige Verwendung ist auf eine Portcard beschränkt, die gezogen wird, während sie Verkehr überträgt
28.3 Resynchronisierungsfenster	vom Stripper zu Beginn eines Synchronisierungsfensters übertragen, wenn eine Resynchronisierung im derzeitigen Synchronisierungsfenster enthalten ist
28.4 BP einstellen	Vom Stripper übertragen, wenn der Bus derzeit nicht arbeitet und der Wert des bp-bit eingestellt werden muss
28.5 Idle	zeigt den Idle-Zustand an
28.6 BP löschen	vom Stripper übertragen, wenn der Bus derzeit nicht arbeitet und das bp-bit gelöscht werden muss

[0107] Der Switch hat eine variable Anzahl an Daten-bits, die an jeden Backplane-Kanal je nach der Striping-Konfiguration für ein Paket aufrechterhalten werden. Innerhalb eines Transceiver-Satzes werden Daten in folgender Reihenfolge aufgefüllt:

F[Fabrik]_[OC192-Port-Anzahl][OC48-Port-Bezeichnung (a, b, c, d)] [Transceiver-Anzahl]

[0108] Der Chipset implementiert bestimmte Funktionen, die hier beschrieben sind. Die meisten hier erwähnten Funktionen werden in mehreren ASICs aufrechterhalten, so dass ihre Dokumentierung auf ASICs-für-ASIC-Basis kein klares Verständnis des gesamten Umfangs der erforderlichen Funktionen liefert.

[0109] Der Switch-Chipset ist derart gebaut, dass er mit Paketen arbeitet, die bis zu 64K + 6 bytes lang sind. Auf der Zuflussseite des Switch gibt es Busse, die sich mehrere Ports teilen. Für die meisten Pakete werden sie ohne Unterbrechung vom Beginn des Pakets bis zum Ende des Pakets übertragen. Dieser Ansatz kann jedoch zu großen Verzögerungsschwankungen für verzögerungsempfindlichen Verkehr führen. Damit verzögerungsempfindlicher Verkehr und langer Verkehr auf der gleichen Switch-Fabrik koexistieren können, wird das Konzept langer Pakete eingeführt. Grundsätzlich können mit langen Paketen Datenblöcke an eine Stelle zur Einreihung in eine Warteschlange gesendet werden, an der Stelle zur Einreihung in eine Warteschlange auf einer Quellenbasis aufgestaut und dann alle zusammen in die Warteschlange eingefügt werden, wenn das Ende des langen Pakets übertragen wird. Die Definition eines langen Pakets basiert auf der Anzahl an bits auf jeder Fabrik.

[0110] Läuft der Switch in einer Umgebung, in der Ethernet-MPU über das gesamte Netzwerk aufrechterhalten wird, werden lange Pakete nicht in einem Switch mit mehr als 40G Größe gesehen.

[0111] Eine breite Cache-Zeilen-shared-memory-Technik wird zum Speichern von Zellen/Paketen in den Port/Prioritätswarteschlangen verwendet. Der shared memory speichert Zellen/Pakete stetig, so dass es praktisch keine Fragmentierung und Bandbreitenabfall im shared memory gibt.

[0112] Im shared memory gibt es mehrere Warteschlangen. Sie basieren auf dem Bestimmungsort und der Priorität. Alle Zellen/Pakete mit derselben Ausgabepriorität und Blade-/Kanal-ID werden in derselben Warteschlange gespeichert. Zellen werden aus Warteschlangen immer vom oberen Ende der Liste her abgebaut und in den Schwanz der Warteschlange eingereiht. Jede Zelle/jedes Paket besteht aus einem Teil des Abflusssendeworts, einer Paketlänge und Paketdaten variabler Länge. Zellen und Pakete werden stetig gespeichert, d. h. der Speichercontroller selbst erkennt nicht die Grenzen von Zellen/Paketen bei den Punkt-zu-Punkt-(Unicast-)Verbindungen. Die Paketlänge wird für MC-Pakete gespeichert.

[0113] Der Gruppenaufruf-(Multicast-)Port-Maskenspeicher 64K × 16-bit wird zum Speichern der Bestimmungs-Port-Maske für die Multicast-Verbindungen verwendet, und zwar ein Eintrag (oder mehrere Einträge)

pro Multicast-VC. Die Port-Masken der Kopf-Multicast-Verbindungen, die von den Multicast-DestID-FIFOs angegeben werden, werden intern zur Ablaufreferenz gespeichert. Der Port-Maskenspeicher wird zurückgezogen, wenn die Port-Maske einer Kopf-Verbindung bereinigt ist und eine neue Kopf-Verbindung bereitgestellt wird.

[0114] APS steht für Automatic Protection Switching, d. h. einen SONET-Redundanzstandard. Damit das APS-Merkmal im Switch unterstützt wird, senden zwei Ausgabe-Ports auf zwei verschiedenen Portcards im Großen und Ganzen denselben Verkehr. Die Speichercontroller erhalten einen Satz von Warteschlangen für einen APS-Port aufrecht und senden duplizierte Daten an beide Ausgabe-Ports.

[0115] Damit Datenduplizierung im Speichercontroller-ASIC unterstützt wird, hat jede der mehreren Unicast-Warteschlangen ein programmierbares APS-bit. Wird das APS-bit an eine Warteschlange gesendet, wird ein Paket an beide Ausgabe-Ports abgebaut. Wird das APS-bit für einen Port auf Null gestellt, arbeitet die Unicast-Warteschlange im normalen Modus. Ist ein Port als APS-Slave konfiguriert, liest er aus den Warteschlangen des APS-Master-Ports. Bei OC48-Ports befindet sich der APS-Port immer auf dem gleichen OC48-Port auf der benachbarten Portcard.

[0116] Die shared-memory-Warteschlangen in den Speichercontrollern können zwischen den Fabriken durch Taktverschiebungen oder eine neu eingesetzte Fabrik aus der Synchronisierung geraten (d. h. gleiche Warteschlangen haben zwischen verschiedenen Speichercontroller-ASICs unterschiedliche Tiefen). Es ist wichtig, dass die Fabrik-Warteschlangen von beliebigen zufälligen Zuständen in den gültigen und synchronisierten Zustand gebracht werden. Es ist ebenfalls wünschenswert, dass bei einem beliebigen Wiederherstellungssystem keine Zellen fallengelassen werden.

[0117] Eine Resynchronisierungszelle wird an alle Fabriken (neue und bestehende) übertragen, damit sie in den Resynchronisierungszustand eintreten. Die Fabriken versuchen, jeglichen vor der Resynchronisierungszelle empfangenen Verkehr abzubauen, bevor die Warteschlangenresynchronisierung endet. Jedoch wird kein nach der Resynchronisierungszelle empfangener Verkehr abgebaut, bis die Warteschlangenresynchronisierung endet. Eine Warteschlangenresynchronisierung endet bei einem der folgenden Ereignisse:

1. Ein Zeitgeber läuft aus.
2. Die Menge an neuem Verkehr (nach der Resynchronisierungszelle empfangenem Verkehr) übersteigt eine Schwelle.

[0118] Am Ende der Warteschlangenresynchronisierung spülen alle Speichercontroller jeglichen übrigen alten Verkehr (vor der Warteschlangenresynchronisierungszelle empfangenen Verkehr) heraus. Der Freimachschritt ist genügend schnell, dass gewährleistet wird, dass alle Speichercontroller den gesamten Speicher füllen können, gleich wann sie in den Resynchronisierungsstatus eingetreten sind.

[0119] Die Warteschlangenresynchronisierung betrifft alle 3 Fabrik-ASICs. Die Aggregatoren müssen sicherstellen, dass die FIFOs nach einer Warteschlangenresynchronisierungszelle identisch abfließen lassen. Die Speichercontroller implementieren das Einreihen in die Warteschlange und das Fallenlassen. Die Separatoren müssen Speichercontroller handhaben, die Verkehr fallenlassen und Längen-Parsing-Status-Maschinen zurücksetzen, wenn dies eintritt. Einzelheiten zur Unterstützung von Warteschlangenresynchronisierung in einzelnen ASICs siehe bei den Chip-ADSS.

[0120] Auf der Seite des Warteschlangenabbaus haben Multicast-Verbindungen unabhängige 32 Token pro Port, die jeweils bis zu 50 bit Daten oder ein vollständiges Paket wert sind. Die Kopf-Verbindung und ihre Port-Maske einer Warteschlange höherer Priorität werden aus dem Verbindungs-FIFO und dem Port-Maskenspeicher in jedem Zyklus ausgelesen. Ein vollständiges Paket wird aus der Multicast-Cache-Zeile auf Basis des Längenfeldes der Kopf-Verbindung isoliert. Das Kopf-Paket wird an alle seine Bestimmungs-Ports gesendet. Die 8 Warteschlangenabbauer übertragen das Paket an die Separatoren, wenn nicht-Null Multicast-Tokens für die Ports verfügbar sind. Die nächste Kopf-Verbindung wird nur verarbeitet, wenn das derzeitige Kopf-Paket an alle seine Ports versendet wird.

[0121] Die Warteschlangenstruktur kann während der Übertragung durch die Fabrik-Resynchronisierungszelle verändert werden, wobei die Zahl der Priorität pro Port-Feld dazu verwendet wird anzuzeigen, wie viele Prioritätswarteschlangen jeder Port hat. Der Stripper-ASIC liegt auf dem Netzwerk-Blade.

[0122] Die folgenden Wörter haben vernünftigerweise spezifische Bedeutungen im Vokabular des Switch. Viele sind an anderer Stelle erwähnt, aber hier wird versucht, sie an einer Stelle mit Definitionen zusammen-

zubringen.

Tabelle 23:

Wort	Bedeutung
APS	Automatic Protection Switching. Ein Sonet/Sdh-Standard zur Implementierung von Redundanz auf physikalischen Verbindungen. APS wird auch zur Wiederherstellung nach ermittelten Portcard-Fehlern verwendet.
Backplane-Synch	generischer Begriff, der entweder den allgemeinen Prozess beschreibt, den Switch-Platten zum Ausgleich von variablen Transportverzögerungen zwischen Festplatten und Taktverschiebung verwenden, oder die Logik, welche die TX/RX-Funktion implementiert, die erforderlich ist, damit die Switch-ASICs variable Transportverzögerungen und Taktverschiebung berücksichtigen.
BIB	Switch-Eingabebus. Der zum Leiten von Daten an den (die) Stripper verwendete Bus (siehe auch BOB).
Blade	Ein anderer Begriff für Portcard. Bezugnahmen auf Blades sollten aus diesem Dokument entfernt worden sein, können aber noch bestehen.
BOB	Switch-Ausgabebus. Der Ausgabebus vom Stripper, der mit dem Abfluss-Speichercontroller verbindet (siehe auch BIB).
Abflusssendewort (egress routeword)	Das Sendewort, das dem Chip nach dem Unstriper zugeführt wird. Aus der Perspektive eines internen Chipset wird das Abflusssendewort als Daten behandelt (siehe auch Fabriksendewort).
Fabriksendewort (fabric routeword)	Von der Fabrik zum Festlegen der Ausgabewarteschlange verwendetes Sendewort. Dieses Sendewort gelangt nicht nach außerhalb des Unstripers. Ein signifikanter Teil dieses Sendeworts entfällt in den Fabriken.
Einfrieren (freeze)	Die Logik behält ihre Werte während Sperrzyklen.
Sperren (lock-down)	Zeitraum, in dem die Fabrik effektiv alle Arbeit stoppt, so dass Taktverschiebung kompensiert wird. Bestimmt die Backplane-Synchronisierungslogik, dass eine Fabrik 8 Taktzyklen zu schnell ist, wird die Fabrik für 8 Taktzyklen gesperrt.
Warteschlangen-Resynch	Eine Warteschlangenresynchronisierung ist eine Reihe von Schritten, die ausgeführt wird, um zu gewährleisten, dass der Logikzustand aller Fabrikwarteschlangen für alle Ports zu einem logischen Zeitpunkt identisch ist. Warteschlangenresynchronisierung ist in keiner Weise auf Backplane-Resynchronisierung (einschließlich Sperren) festgelegt, ausgenommen dass Sperren bei einer Warteschlangenresynchronisierung vorkommen kann.
SIB	Striped-Eingabebus. Größtenteils obsoletter Begriff, der den Ausgabebus vom Stripper und den Eingabebus an den Aggregator beschreibt.
SOB	Striped-Ausgabebus, d. h. der Ausgabebus von der Fabrik bzw. der Eingabebus des Aggregators (siehe auch SIB).
Sync	Hängt sehr vom Kontext ab. Verwandte Begriffe sind Warteschlangenresynchronisierung, Sperren, Einfrieren und Backplane-Resynchronisierung.
Wacking	Implizite bit-Steuerung, die auf der OC192-Zuflussstufe auftritt, weil die Daten zwischen den Strippern bit-verschränkt sind. Die bit-Steuerung wird von den Aggregatoren umgekehrt.

[0123] Die Beziehung zwischen den Übertragungs- und Empfangszählern ist aus [Fig. 6](#) ersichtlich.

[0124] Die Erfindung wurde zwar in den vorstehenden Ausführungsformen zu Veranschaulichungszwecken eingehend beschrieben, aber es sollte selbstverständlich sein, dass diese Einzelheiten nur diesem Zweck dienen und Varianten darin vom Fachmann vorgenommen werden können, ohne dass er vom Umfang der Erfindung abweicht, ausgenommen wie sie durch die folgenden Ansprüche beschrieben ist.

Patentansprüche

1. Vorrichtung (10) zum Speichern von Paketen, umfassend:

einen Speicher (14) zum Halten von Paketen und einen Mechanismus (16) zum Speichern von mindestens zwei Paketen in dem Speicher (14), wobei nur ein Paketgrenzenindikator (18) mit den mindestens zwei Paketen einhergeht, der Speichermechanismus (16) einen Speichercontroller (20) enthält, der derart ausgelegt ist, dass er einen Paketgrenzenindikator (18) in dem Speicher (14) platziert, nachdem eine festgelegte Anzahl bits in dem Speicher (14) gespeichert worden ist, wobei der Speicher (14) durch Cache-Zeilen (22) definiert wird und die Pakete entlang der Cache-Zeilen (22) in dem Speicher (14) gespeichert werden;
wobei der Speichercontroller (20) derart ausgelegt ist, dass er bits von Daten von Paketen in einer Cache-Zeile in dem Speicher (14) speichert und ein Identifizierer (24) in der Cache-Zeile von Daten anzeigt, wie viele bits in der Cache-Zeile gültig sind.

2. Vorrichtung (10) nach Anspruch 1, wobei jede Cache-Zeile 200 bits lang ist.

3. Vorrichtung (10) nach Anspruch 2, wobei der Identifizierer (24) sich 2 bits von den 200 bits der Cache-Zeile befindet.

4. Vorrichtung (10) nach Anspruch 3, wobei der Speichercontroller (20) derart ausgelegt ist, dass er einen Paketgrenzenindikator (18) nach einem Paket-Wert von 15 Cache-Zeilen (22), die im Speicher (14) gespeichert wurden, einsetzt.

5. Vorrichtung (10) nach Anspruch 4, wobei der Speichercontroller (20) derart ausgelegt ist, dass er auf Basis von Paketgrenzenindikatoren (18) in Bezug auf die Priorität der Pakete (12) vermittelt, welche Pakete (12) von dem Speicher (14) übertragen werden sollen.

6. Verfahren zum Speichern von Paketen (12), umfassend die Schritte:
Empfangen von Paketen (12) in einem Speicher (14) und
Speichern mindestens zweier Pakete mit einem Speichercontroller (20) in dem Speicher (14), wobei mit den mindestens zwei Paketen nur ein Paketgrenzenindikator (18) einhergeht;
wobei der Speichersschritt die Schritte enthält: Platzieren eines Paketgrenzenindikators (18) mit dem Speichercontroller (20) in dem Speicher (14), nachdem eine festgelegte Anzahl bits in dem Speicher (14) gespeichert worden ist;
Speichern der Pakete (12) entlang Cache-Zeilen (22) in dem Speicher (14) und
Speichern von bits von Daten von Paketen (12) in einer Cache-Zeile in dem Speicher (14), wobei ein Identifizierer (24) in der Cache-Zeile von Daten angibt, wie viele bits in der Cache-Zeile gültig sind.

7. Verfahren nach Anspruch 6, wobei nach dem Speichersschritt ein Schritt erfolgt Vermitteln von Paketen (12) aus dem Speicher (14) auf Basis von Paketgrenzenindikatoren (18) in Bezug auf die Priorität der Pakete (12).

Es folgen 8 Blatt Zeichnungen

Anhängende Zeichnungen

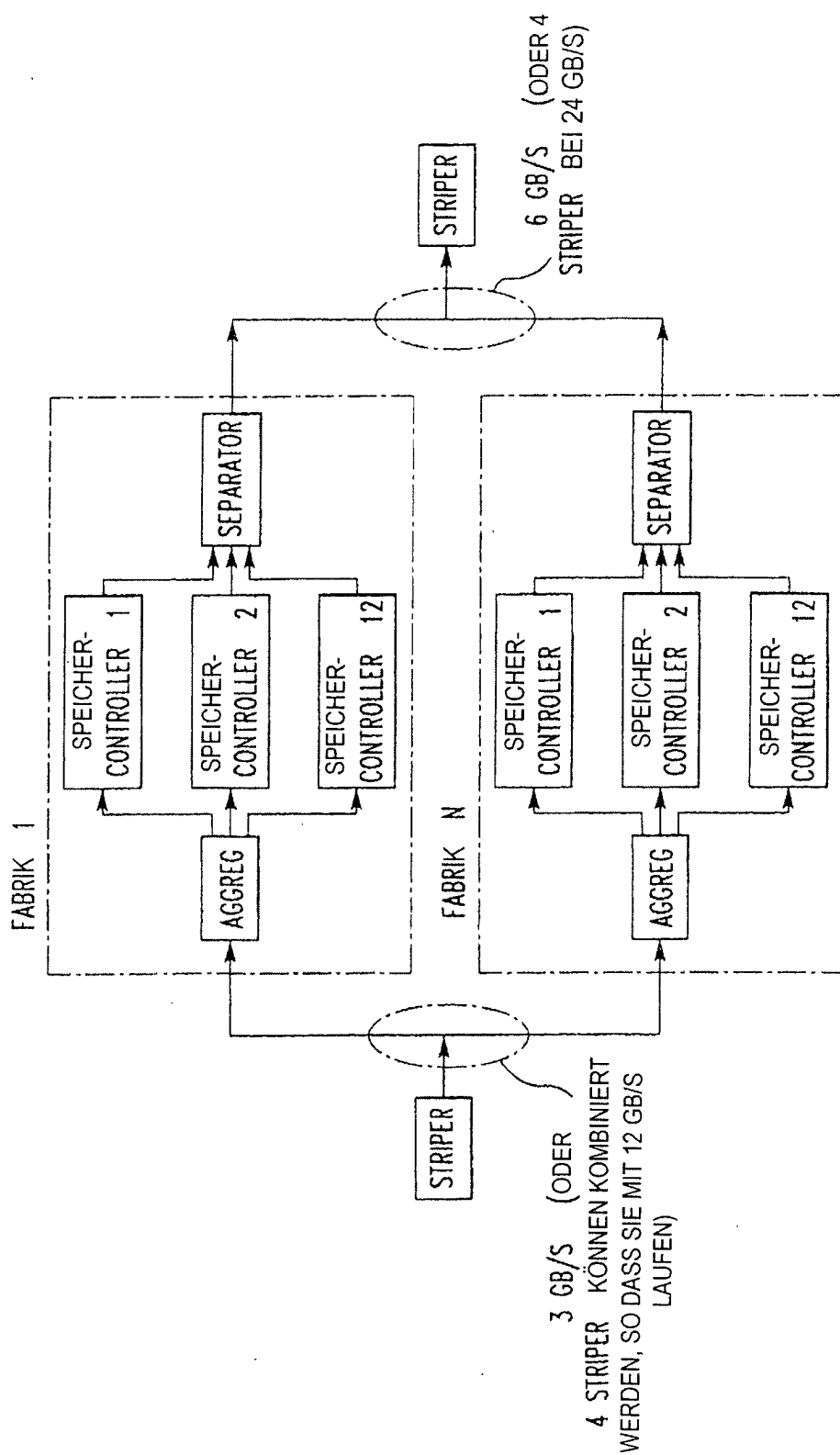


FIG.1

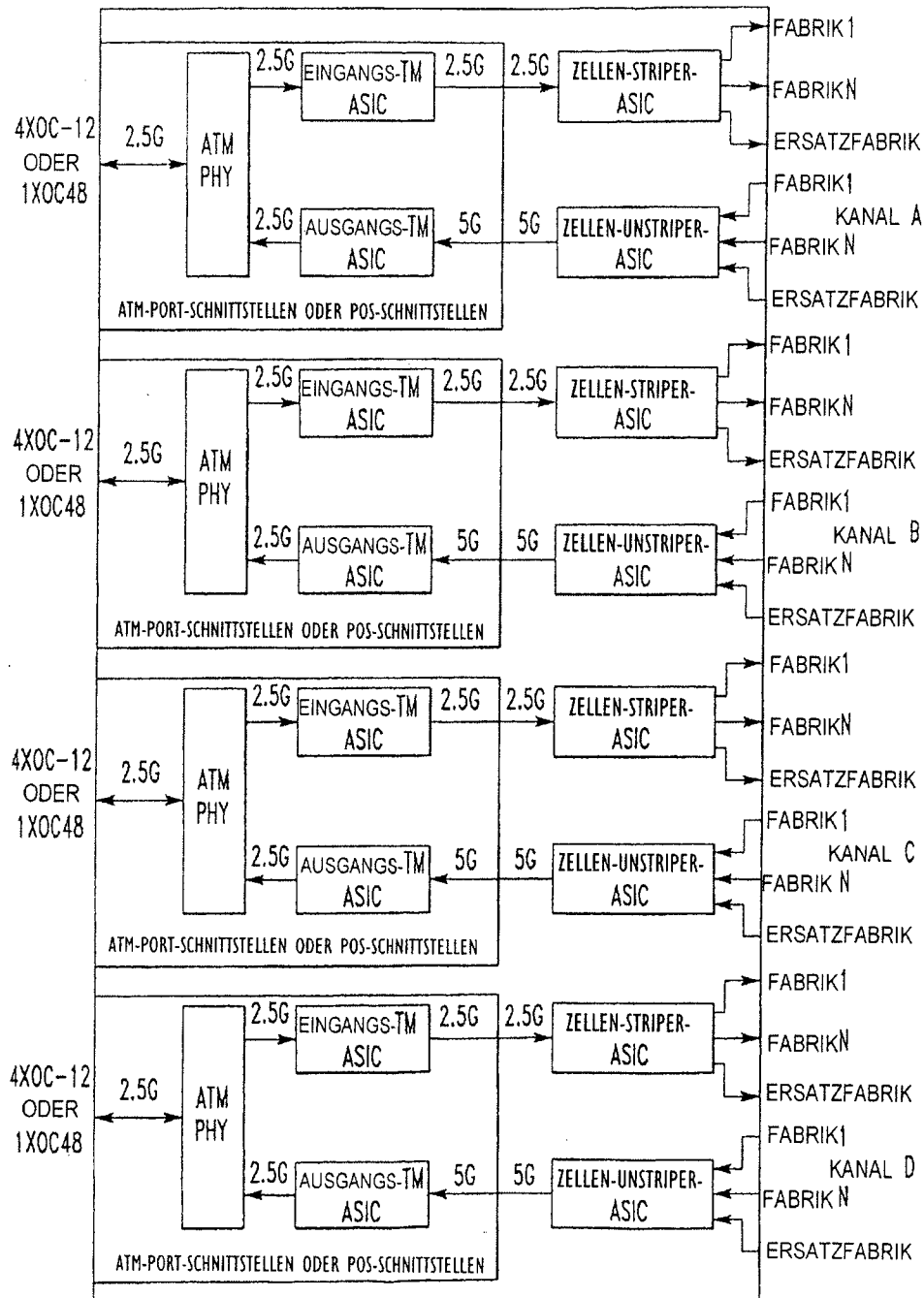


FIG. 2

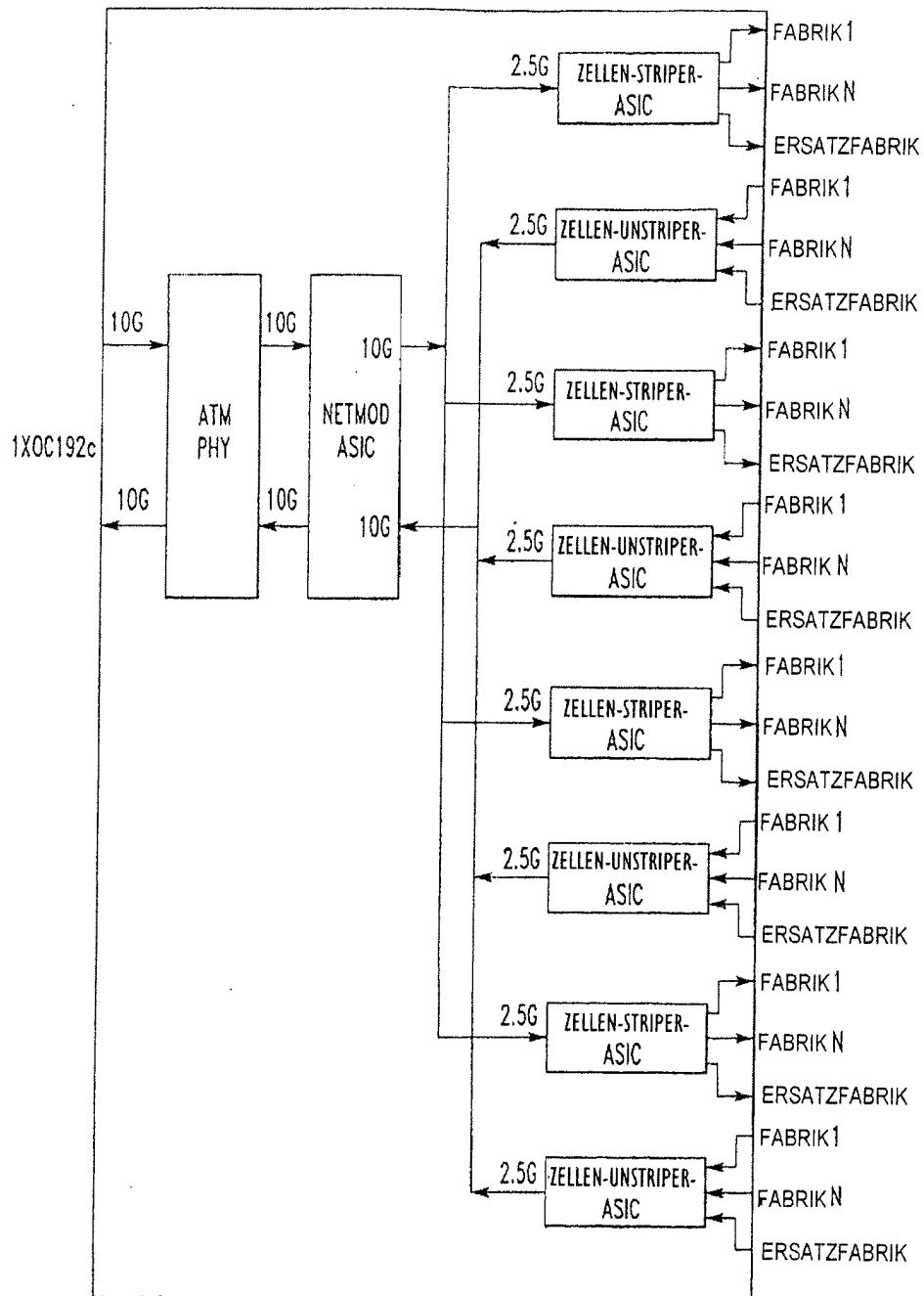
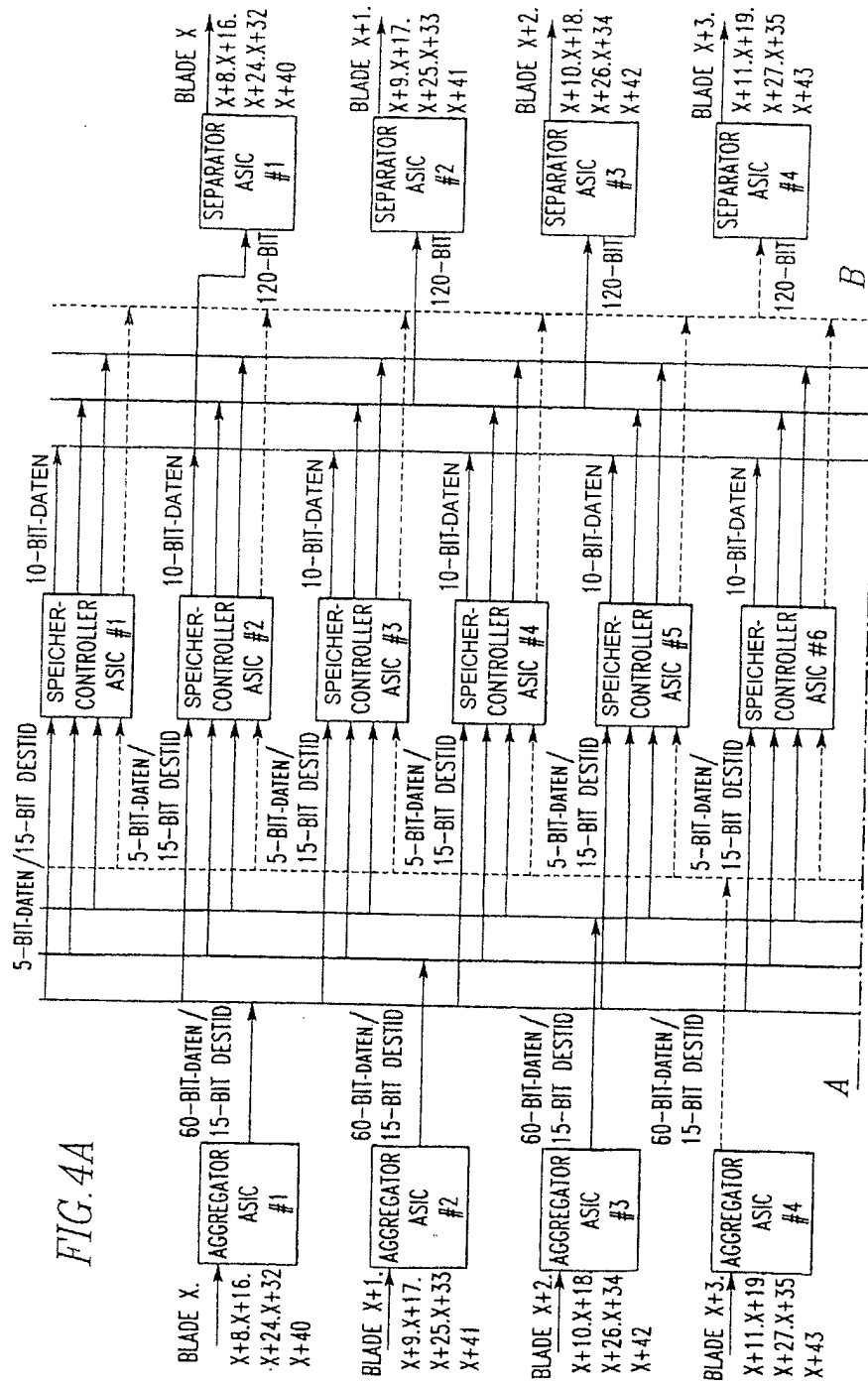
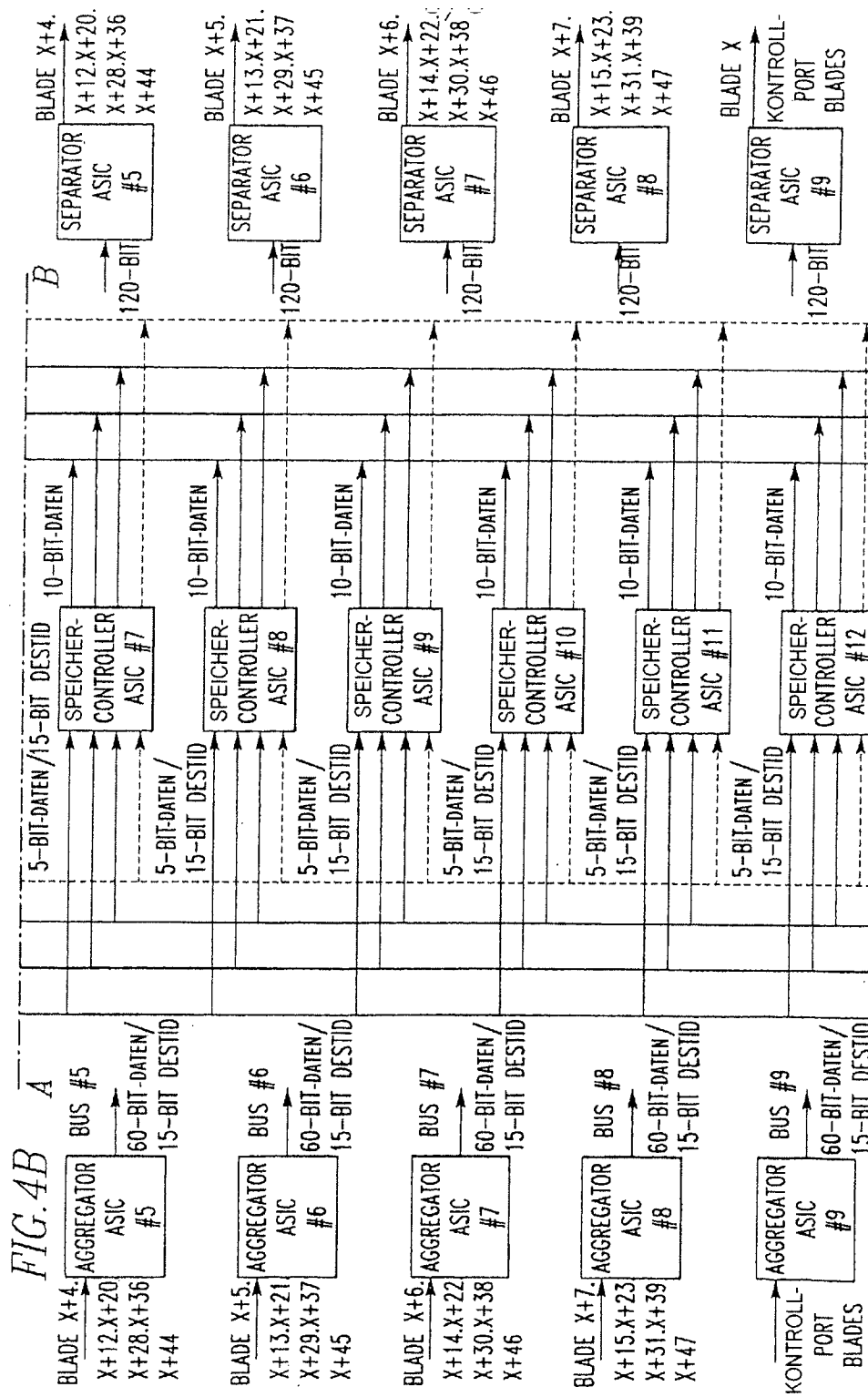


FIG. 3





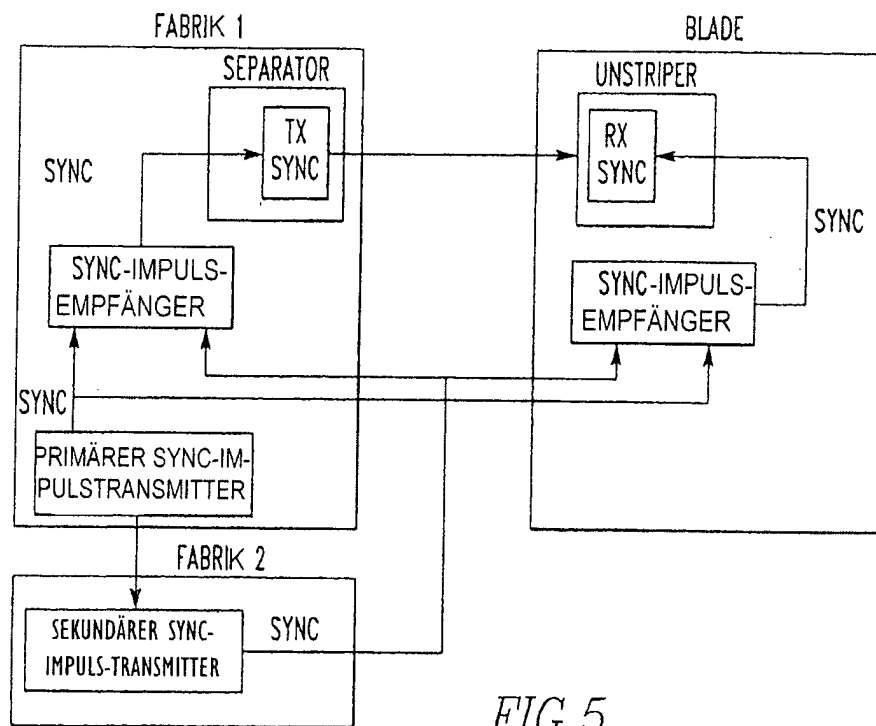


FIG. 5

