(19) **Bundesrepublik Deutschland**
Deutsches Patent- und Markenamt

(10) **DE 102 96 979 B4** 2010.03.11

(12) **Patentschrift**

(21) Deutsches Aktenzeichen: **102 96 979.5**
(86) PCT-Aktenzeichen: **PCT/US02/10808**
(87) PCT-Veröffentlichungs-Nr.: **WO 2003/001386**
(86) PCT-Anmeldetag: **05.04.2002**
(87) PCT-Veröffentlichungstag: **03.01.2003**
(43) Veröffentlichungstag der PCT Anmeldung in deutscher Übersetzung: **09.12.2004**
(45) Veröffentlichungstag der Patenterteilung: **11.03.2010**

(51) Int Cl.[8]: ***G06F 13/00*** (2006.01)
G06F 13/16 (2006.01)

Innerhalb von drei Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten(§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 1 Patentkostengesetz).

(30) Unionspriorität:
09/888,105 22.06.2001 US

(73) Patentinhaber:
Intel Corporation, Santa Clara, Calif., US

(74) Vertreter:
ZENZ Patent- und Rechtsanwälte, 45128 Essen

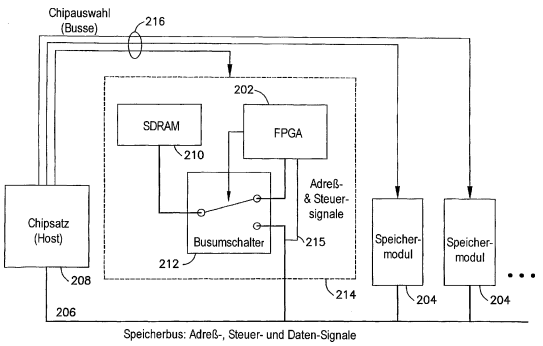
(72) Erfinder:
Ruehle, Michael D., Santa Clara, Calif., US

(56) Für die Beurteilung der Patentfähigkeit in Betracht gezogene Druckschriften:

DE	**195 30 154**	**A1**
US	**59 99 654**	**A**
US	**58 92 826**	**A**
US	**58 78 240**	**A**
US	**56 87 346**	**A**
US	**55 98 575**	**A**
US	**52 37 616**	**A**

(54) Bezeichnung: **Computersystem mit einer aktiven Speicherbusperipherieeinrichtung und Verfahren zur Steuerung einer aktiven Speicherbusperipherieeinrichtung**

(57) Hauptanspruch: Computersystem mit
einem einen Prozessor umfassenden Host (208), der über einen Speicherbus (206) mit Speichermodulen (204) gekoppelt ist, und
einer mit dem Speicherbus (206) gekoppelten aktiven Speicherbusperipherieeinrichtung (214),
wobei die Speicherbusperipherieeinrichtung (214) ein Speicherbauelement (210), eine Logikschaltung (202) und einen von der Logikschaltung (202) gesteuerten Busumschalter (212), der einen Bus des Speicherbauelements (210) entweder mit dem Speicherbus (206) oder mit der Logikschaltung (202) verbindet, aufweist,
wobei die Logikschaltung (202) – unabhängig von der Stellung des Busumschalters (212) – über einen Signalabgriff (215) mit Adressleitungen des Speicherbusses (206) gekoppelt ist, so dass die Logikschaltung (202) auf dem Speicherbus (206) angesteuerte Adresswerte überwachen kann,
wobei die Logikschaltung (202) bei Erfassen einer vorgegebenen Sequenz von Adresswerten auf dem Speicherbus (206), die jeweils eines der neben der Speicherbusperipherieeinrichtung (214) mit dem Speicherbus (206) gekoppelten Speichermodule (204) an dem Speicherbus (206) adressieren, den Busumschalter (212) so steuert, dass dieser den...



Beschreibung

[0001] Die Erfindung bezieht sich auf ein Computersystem mit einem einen Prozessor umfassenden Host, der über einen Speicherbus mit Speichermodulen gekoppelt ist, und einer mit dem Speicherbus gekoppelten aktiven Speicherbusperipherieeinrichtung. Ferner bezieht sich die Erfindung auf ein Verfahren zur Steuerung einer aktiven Speicherbusperipherieeinrichtung in einem solchen Computersystem.

[0002] Bei einem fortgesetzten Streben nach erhöhter Computergeschwindigkeit und Effizienz benutzen Entwickler manchmal Einrichtungen für spezielle Zwecke, um Aktivitäten zu behandeln, für welche die Einrichtungen (oder Geräte) speziell entwickelt sein können. Beispielsweise werden Videokarten (Graphikbeschleuniger) oftmals benutzt, um die Fähigkeit eines Computersystems zu verbessern, Videobilder anzuzeigen, ohne die Gesamtcomputerleistungsfähigkeit zu verringern. Sie entlasten eine zentrale Verarbeitungseinheit (CPU) eines Computers derart, daß sie andere Befehle ausführen kann, während die Videokarte Graphikberechnungen handhabt.

[0003] Ein anderes Beispiel betrifft spezielle Einrichtungen (Bauelemente oder Geräte) zum Verschlüsseln und Entschlüsseln. In dem Maße, wie immer mehr Informationen über das Internet übermittelt werden, wurden Sicherheitsaspekte zunehmend wichtiger. Verschlüsselungstechniken werden im Stand der Technik verwendet, um das nicht autorisierte Abfangen von über das Internet übertragenen Daten zu verhindern. Ein Beispiel eines üblichen Protokolls zur Datenverschlüsselung ist Security Sockets Layer (SSL) (SSL 2.0, überarbeitet am 09. Februar 1995). Wenn eine SSL-Sitzung initiiert wird, leitet der Server seinen „öffentlichen" Schlüssel an den Browser des Benutzers weiter, welchen der Browser verwendet, um einen zufällig erzeugten „geheimen" Schlüssel an den Server zurückzusenden, so daß ein Austausch eines sicheren Schlüssels für diese Sitzung vorgenommen wird. Das von Netscape™ entwickelte SSL wurde mit weiteren Protokollen und Authentisierungsverfahren durch die Internet Engineering Task Force (IETF) zu einem neuen Protokoll verschmolzen, das als Transport Layer Security (TLS) (TLS 1.0, revidiert 1999) bekannt ist.

[0004] Verschlüsselungs/Entschlüsselungs-Protokolle, wie sie bei SSL verwendet werden, sind sehr berechnungsintensiv. Der Prozeß des Codierens und Decodierens von Informationen kann einen großen Teil der wertvollen Verarbeitungsressourcen einer zentralen Verarbeitungseinheit (CPU) rauben. Neben der Verschlüsselungs/Entschlüsselung und der Videoverarbeitung ziehen weitere Aktivitäten, die berechnungsintensive und sich wiederholende Prozesse einschließen, einen Vorteil aus einer Verarbeitung in speziellen Peripherieeinrichtungen.

[0005] Beim Bereitstellen einer Einrichtung (Bauelement oder Gerät) für einen speziellen Zweck an einem Speicherbus (einer Speicherbusperipherieeinrichtung), wie beispielsweise einer zur Verschlüsselung/Entschlüsselung, muß die Einrichtung aktiv sein und darüber hinaus in der Lage sein, Kommandos oder Befehle aus der CPU zu empfangen. Es ist folglich ein System erwünscht, das eine CPU von einem Teil der Verantwortung für berechnungsintensive Aktivitäten entbindet, indem eine spezielle, aktive Speicherbusperipherieeinrichtung zur Verfügung gestellt wird. Ferner ist ein verbessertes Kommunikationssystem zwischen der CPU und der speziellen, aktiven Speicherbusperipherieeinrichtung erwünscht.

[0006] Aus der Druckschrift DE 195 30 154 A1 ist ein Computersystem bekannt, bei dem ein Host über einen Systembus mit einer intelligenten Schaltung und weiteren Komponenten gekoppelt ist. Die intelligente Schaltung umfasst einen Speicher, der über eine Schnittstellenschaltung wahlweise an einen internen Daten- und Adreßbus der intelligenten Schaltung oder an den Systembus ankoppelbar ist. Normalerweise ist der Speicher der intelligenten Schaltung mit dem internen Daten- und Adreßbus verbunden, um über diesen mit einer Recheneinrichtung zu kommunizieren. An den internen Adreß- und Datenbus sind ferner eine Reihe von Ansteuerschaltungen angekoppelt, die beispielsweise Schrittmotoren ansteuern. Wenn aber der Host den Speicher der intelligenten Schaltung adressiert, um Daten in diesen zu schreiben oder Daten aus diesen zu lesen, erkennt dies ein Adreßdekodierer der intelligenten Schaltung. Der Adreßdekodierer gibt über eine Steuerleitung ein entsprechendes Signal an die Schnittstellenschaltung aus, so daß diese den Speicher der intelligenten Schaltung mit dem Systembus verbindet.

[0007] Aus dem US-Patent 5,237,616 A ist eine Vorrichtung und ein Verfahren bekannt, die es einem Computer ermöglichen, in einem sicheren Modus zu arbeiten, welcher kryptographische Funktionen zur Verfügung stellt. In dieser Druckschrift ist unter anderem beschrieben, daß ein Mikroprozessor über einen Daten- und Adreßbus mit einem innerhalb einer sicheren Umgrenzung angeordneten privilegierten Speicher gekoppelt ist. In dieser sicheren Umgrenzung befindet sich darüber hinaus ein ASIC, über den sowohl der Daten- und Adreßbus des Mikroprozessors als auch der privilegierte Speicher mit einer Schnittstelle zu einem nicht privilegierten Speicher und I/O-Schnittstellen gekoppelt sind. Die Druckschrift beschreibt unter anderem, daß der Betriebsmodus des Computers in einen privilegierten oder nicht privilegierten Modus geändert wird, indem ein unbedingter indirekter Inter-Segment-Sprung auf bestimmte Adressen ausgeführt wird. Dies bewirkt sequentielle Speicherzugriffe auf zwei bestimmte aufeinanderfolgende Adressen. Somit bewirken die Zugriffe auf zwei aufeinanderfolgende Adressen die Mo-

dusumschaltung.

[0008] Ausgehend von dem oben erörterten Stand der Technik ist es eine Aufgabe der Erfindung, ein verbessertes Kommunikationssystem zwischen einer CPU (einem Host) und einer aktiven Speicherbusperipherieeinrichtung zu schaffen, bei welchem die Speicherbusperipherieeinrichtung in herkömmliche Steckplätze am Speicherbus eingesteckt werden kann und auf einen Speicher der Speicherbusperipherieeinrichtung sowohl durch den Host als auch durch eine interne Verarbeitungsschaltung der Speicherbusperipherieeinrichtung zugegriffen werden kann.

[0009] Diese Aufgabe wird erfindungsgemäß durch ein Computersystem mit den Merkmalen des Anspruchs 1 bzw. ein Verfahren mit den Merkmalen des Anspruchs 5 gelöst.

[0010] Vorteilhafte und/oder bevorzugte Weiterbildungen sind in den Unteransprüchen gekennzeichnet.

[0011] Im Folgenden wird die Erfindung anhand von in den Zeichnungen dargestellten bevorzugten Ausführungsbeispielen näher beschrieben.

[0012] Fig. 1 stellt eine Veranschaulichung eines typischen Speicherbusses im Stand der Technik zur Verfügung.

[0013] Fig. 2 veranschaulicht die Betriebsweise einer aktiven Speicherbusperipherieeinrichtung gemäß den Prinzipien der vorliegenden Erfindung.

[0014] Fig. 3a–Fig. 3c geben ein Ablaufdiagramm an, das für den Prozeß der Busumschaltung für eine dynamische Busperipherieeinrichtung gemäß den Prinzipien der vorliegenden Erfindung repräsentativ ist.

[0015] Fig. 4 stellt eine Veranschaulichung von Beispieladreßorten zur Verfügung, die bei einem sequentiellen Adreßruf benutzt werden, der zum Auslösen eines „Erlange Bus" unter den Prinzipien der vorliegenden Erfindung verwendet wird.

[0016] Fig. 1 veranschaulicht einen typischen Speicherbus im Stand der Technik. Ein Mikroprozessor-Chipsatz **102** (der Host) benutzt einen oder mehrere Speichermodule **104**, beispielsweise Dual In-line Memory Modules (DIMM). Der Host **102** kommuniziert üblicherweise mit den Speichermodulen über einen gemeinsamen Speicherbus. Mit anderen Worten, jedes Speichermodul sieht sämtliche Adreß-, Steuer- und Datensignale, die auf dem Speicherbus **106** übermittelt werden. Der Host ist über die Benutzung einer Reihe von „Chipauswahl"-Leitungen (Bussen) **108** in der Lage zu definieren, welches Speichermodul für den Empfang einer Nachricht vorgesehen ist. Gemäß Fig. 1 sind eine Reihe von Chipauswahl-„Bussen" **108** zur Verfügung gestellt. Bei einem DIMM beispielsweise würde jeder Chipauswahlbus **108** eine Chipauswahl an die Vorderseite des Moduls und eine an die Rückseite des Moduls zur Verfügung stellen. Jede Chipauswahlleitung **108** ist einem speziellen Speichermodul **104** zugeordnet. Die angelegte oder aktivierte Chipauswahlleitung **108** gibt an, welches Speichermodul die gegenwärtig auf dem Speicherbus **106** übermittelten Daten empfangen soll.

[0017] Fig. 2 veranschaulicht die Betriebsweise einer aktiven Speicherbusperipherieeinrichtung gemäß den Prinzipien der vorliegenden Erfindung. Bei einem Ausführungsbeispiel der vorliegenden Erfindung wird ein feld-programmierbares Gate-Array **202** (FPGA) zum Beschleunigen verschiedener berechnungsintensiver Aufgaben benutzt (wie beispielsweise der Verschlüsselung und Entschlüsselung). Das FPGA **202** ist für eine optimale Ausführung der seinem Zweck (Verschlüsselung/Entschlüsselung, etc.) zugeordneten sich wiederholenden Berechnungen durch parallele Verarbeitungseinheiten, etc. konfiguriert. Bei einem Ausführungsbeispiel ist das FPGA **202** in einem DIMM-Steckplatz an einem PC-100-(registrierte DIMM-Design-Spezifikation (Revision 1.2)) oder einem PC-133-Speicherbus **206** (registrierte DIMM-Design-Spezifikation (Revision 1.1)) angeordnet. Bei einem Ausführungsbeispiel ist ein moduleigener (on-board) SDRAM (synchroner dynamischer Speicher mit wahlfreiem Zugriff) **210** zwischen dem Host-Computer **208**, welcher ihn als normalen Speicher wahrnimmt (lediglich ein weiteres Speichermodul), und dem FPGA **202** geteilt, indem die Adreß/Daten/Steuer-Verbindungen zu dem moduleigenen SDRAM **210** zwischen dem Host **208** und dem FPGA **202** umgeschaltet werden **212**. Bei einem Ausführungsbeispiel hat zu jedem beliebigen Zeitpunkt entweder der Host **208** oder das FPGA **202** Zugriff auf den moduleigenen SDRAM **210**. Das Umschalten mit Hilfe des Busschalters **212** dieser moduleigenen SDRAM **210** wird von der Host-Maschine **208** angefordert, aber direkt von dem FPGA **202** gesteuert. Bei einem Ausführungsbeispiel muß der Host **208** in der Lage sein, dem FPGA **202** zwei Kommandos zu senden: „Schalte den SDRAM-Bus zu dem Host" und „Schalte den SDRAM-Bus zu dem FPGA". Verwendet man die Sichtweise oder Perspektive des Hosts, können diese Kommandos „Erlange Bus" beziehungsweise „Übergebe Bus" genannt werden.

[0018] Ein Signalabgriff **215** wird benutzt, um das FPGA **202** mit den Adreß- und Steuersignalen sowie der Chipauswahl der Einrichtung **214** auf dem Speicherbus **206** des Hosts zu verbinden, unabhängig davon, mit welcher Einrichtung der moduleigene SDRAM-Bus-Schalter **212** verbunden ist, so daß das FPGA die von dem Host **208** angesteuerten Werte

überwachen kann. Bei einem Ausführungsbeispiel weist das FPGA **202** infolge von Größeneinschränkungen nicht genug Pins auf, um die Datenleitungen zu überwachen. Somit werden die Datensignale nicht überwacht.

[0019] Eine Möglichkeit des Sendens des „Erlange Bus"-Kommandos könnte darin bestehen, daß der Host **208** aus einer von zwei entsprechenden Auslöseradressen in dem Speicher des moduleigenen SDRAM **210** liest oder in diese schreibt. Durch Überwachen der Adreß- und Steuersignale könnte das FPGA **202** erfassen, wenn auf die Auslöseradresse für das „Erlange Bus"-Kommando zugegriffen wird, und den Bus dementsprechend umschalten. Jedoch könnte dies bei Systemen, die einen Fehlerkorrekturcode(ECC)-Speicher benutzen, möglicherweise ein Problem verursachen. Wenn der Host **208** ein „Erlange Bus"-Kommando ausgibt, ist er nicht mit dem Speicher des moduleigenen SDRAM **210** verbunden. Sobald der Chipsatz **208** versucht, aus dem moduleigenen SDRAM **210** zu lesen, liest er ungültige Daten oder „Müll" – je nachdem, welche Werte gerade auf den Daten- und Paritätsleitungen des Speicherbusses **206** im Ergebnis zuvor angesteuerte Werte liegen (Kapazität und Ladungsabbau durch Leckströme) – und dies könnte einen ECC-Fehler mit möglichen Abbruchkonsequenzen erzeugen. Das System könnte entscheiden, daß der Speicher (die Einrichtung **214**) defekt ist und die Kommunikation mit ihm vollständig abschalten. Bei einigen Systemen könnte sogar eine Schreiboperation, die von der zentralen Verarbeitungseinheit (CPU) angefordert wird, ein Lesen durch den Chipsatz **208** erzeugen, indem beispielsweise der Chipsatz **208** von verschiedenen Speicherplätzen liest, einige der Daten in der angeforderten Weise modifiziert und dann sämtliche Daten zurückschreibt. Der ECC könnte folglich einen falschen Fehler erfassen, und es könnten sich Probleme ergeben.

[0020] Wegen dieser möglichen Probleme ist es erforderlich, den Busschalter **212** über ein alternatives Mittel auszulösen. Der Host **208** könnte, statt in den Speicher des moduleigenen SDRAM **210** zu schreiben, um „Erlange Bus" auszulösen, in Speicherbereiche auf einem anderen DIMM **204** an dem Speicherbus **206** des Systems schreiben, und das FPGA **202** könnte dies erfassen, indem die Adreßsignale des Speicherbusses **206** überwacht werden, welche sich der Chipsatz **208**, die Einrichtung **214** (SDRAM **210**, Busschaltung **212** und FPGA **202**) und die anderen DIMMs (Speichermodule) **204** teilen. Da die Chipsauswahlsignale **216** nicht von den verschiedenen DIMMs **214**, **204** (allgemein) gemeinsam benutzt werden, kann aber die Einrichtung **214** nicht feststellen, auf welches andere Speichermodul **204** (oder welche Seite dieses Moduls) zugegriffen wird. Da außerdem die genaue Verwendung der Speicherbusadreßleitungen zum Auswählen von Zeilen, Bänken und Spalten von Speichermodul **204** zu Speichermodul **204** variiert, kann es sein, daß die Einrichtung **214** nicht in der Lage ist, genau festzustellen, bei welchem Offset in ein Speichermodul **204** (vom Beginn der reservierten 2KB, was unten erläutert wird) zugegriffen wird. Man könnte auf die Verwendung der acht am geringsten bewerteten Busadreßleitungen als die acht am geringsten bewerteten Spaltenadreßbits vertrauen. Bei einem Ausführungsbeispiel mit 64-Bit-Datenworten kann die Einrichtung **214** feststellen, auf welche physikalische Adresse modulo 2KB zugegriffen wird. Sie kann beispielsweise feststellen, daß ein Zugriff auf eine physikalische Adresse 2048·N + 1224 Bytes bei irgendeinem nicht bekannten Wert N auftrat. Die Informationen der Einrichtung **214** können in dem Offset von 1224 Bytes oder 153 64-Bit-Speicherplätzen bestehen. Dies stellt nur 8 Bits Informationen zur Verfügung. Sofern das FPGA **202** eine „Erlange Bus·-Anforderung jedesmal dann ausführen würde, wenn ein bestimmtes Offset in die 2 KB (dem reservierten Bereich des Speichers) zu sehen ist, so würde es diese zu häufigen nicht-beabsichtigten Zeitpunkten ausführen, die nicht ausschließlich durch absichtliche „Erlange Bus"-Kommandos ausgelöst würden, sondern darüber hinaus auch durch dazu in keinerlei Beziehung stehende Speicherzugriffe durch das Betriebssystem oder Software-Anwendungen. Um derartige versehentliche „Erlange Bus"-Umschaltungen zu minimieren, wird bei der Erfindung die Informationsmenge in dem Kommando erhöht, indem nicht nur eine einzige Adresse sondern eine Sequenz von Adressen geschrieben wird. Bei einem Ausführungsbeispiel kann es unwahrscheinlich gemacht werden, daß der Chipsatz **208** zufällig Speicherzugriffe ausführt, die mit der Sequenz übereinstimmen, indem die Sequenz sorgfältig ausgewählt und ausreichend lang gemacht wird.

[0021] Bei einem Ausführungsbeispiel ist es nicht erforderlich, eine Sequenz von Adreßrufen auch für das „Übergebe Bus"-Kommando zu verwenden. Da der Host **208** zum Zeitpunkt eines „Übergebe Bus"-Kommandos mit dem SDRAM **210** der Einrichtung verbunden ist, gibt es kein Problem beim Schreiben einer einzelnen Auslöseradresse in dem SDRAM **210** der Einrichtung. Nach einem solchen Kommando schaltet das FPGA **202** den Bus zu sich selbst um.

[0022] Fig. 3a–Fig. 3c stellen ein Ablaufdiagramm zur Verfügung, das für den Prozeß der Busumschaltung für eine dynamische Busperipherieeinrichtung gemäß den Prinzipien der vorliegenden Erfindung repräsentativ ist. Bei einem Ausführungsbeispiel der vorliegenden Erfindung findet sich der Busschalter an der Standardposition, welche eine Kommunikation zwischen dem moduleigenen SDRAM und dem FPGA zur Verfügung stellt, **302**. Bei einem Ausführungsbeispiel würde der Host dann, wenn der Host auf den Speicher der Einrichtung zuzugreifen wünscht **304** (für Verschlüsselung/Entschlüsselung,

etc.), das System in einer Schleife blockieren („spin-lock"; eine unendliche Schleife bewirken), so viele Interrupts wie möglich sperren und einen möglichst exklusiven Zugriff auf den Speicher und eine möglichst nicht-unterbrechbare Ausführungspriorität einrichten **306**. Bei einem Ausführungsbeispiel schreibt der Host so schnell wie möglich zu einer vorgegebenen Sequenz von Adressen in die reservierten 2 KB **308**. Da die Adressen, die die Einrichtung sieht, auf 64-Bit-Datenworten basieren, ist jede Adresse in der Sequenz durch ein anderes Vielfaches von 8 Bytes versetzt (Offset). Bei einem Ausführungsbeispiel ist eine gültige Sequenz von acht Offsets folgende: 1208, 646, 1736, 1056, 408, 1840, 1256 und 704 Bytes. Damit bei einem Ausführungsbeispiel das FPGA die „Erlange Bus"-Kommandosequenz erfaßt, werden die acht am geringsten bewerteten (am wenigsten signifikanten) Adreßleitungen aus dem Speicherbus des Systems bei jeder geeigneten Taktflanke überwacht. Bei einem Ausführungsbeispiel werden diese acht Bits mit dem Kommandosequenzwerten verglichen, die durch Teilen der Byte-Offsets, die von dem Host verwendet werden, durch acht bestimmt worden sind. Für die oben angegebene Sequenz sind diese Werte 151, 58, 217, 132, 51, 230, 157 und 88. Bei einem Ausführungsbeispiel wird der Abschnitt der Kommandosequenz, die zuvor gesehen wurde, überwacht, und der Schalter wird auf den Host umgeschaltet, wenn die vollständige Sequenz wahrgenommen wurde.

[0023] Bei einem Ausführungsbeispiel wird dann die Schleifen-Verriegelung (Spin-Lock) beseitigt und werden die Interrupts wieder freigegeben **310**. Bei einem Ausführungsbeispiel wartet das System eine gewisse Zeitdauer, die es dem FPGA ermöglicht, die Kommandosequenz zu erfassen (**312**) und den SDRAM-Bus auf den Host umzuschalten **314**, **316**. Bei einem Ausführungsbeispiel beträgt diese Zeitdauer etwa 5 Mikrosekunden.

[0024] Bei einem Ausführungsbeispiel wird der moduleigene SDRAM als nächstes von dem Host mit Daten geladen, die verschlüsselt/entschlüsselt werden sollen (oder die irgendwelchen anderen Zwecken dienen) **318**. Der Host führt dann eine vorgegebene Sequenz von Adreßaufrufen aus, um ein „Übergebe Bus" auszulösen **320**. Die Daten werden dann an das FPGA weitergeleitet, so daß die Berechnungsaktivität (wie beispielsweise Verschlüsselung/Entschlüsselung) ausgeführt werden kann **322**. Bei einem Ausführungsbeispiel werden nach der Aktivität die verschlüsselten/entschlüsselten Daten an den SDRAM zurückgegeben, um dort gehalten zu werden **324**. Der Host löst dann ein „Erlange Bus" mit Hilfe der richtigen sequentiellen Adreßaufrufe aus **326** (so wie zuvor bei **306** bis **316** ausgeführt). Bei einem Ausführungsbeispiel nimmt das FPGA diese sequentiellen Adreßaufrufe wahr und schaltet den Bus zu dem Host um (**328**). Bei einem Ausführungsbeispiel liest der Host, nachdem er auf den Abschluß des Umschaltens gewartet hat (**330**, **332**), die geänderten (verschlüsselten/entschlüsselten, etc.) Daten aus dem SDRAM und benutzt sie **334**.

[0025] Fig. 4 veranschaulicht Beispieladreßorte, die bei einem sequentiellen Adreßruf benutzt werden, der zum Auslösen eines „Erlange Bus" gemäß den Prinzipien der vorliegenden Erfindung verwendet wird. Bei einem Ausführungsbeispiel initiiert der Host **402** ein „Erlange Bus"-Kommando, indem er in spezielle zuvor definierte Speicheradreßorte in einem reservierten Bereich eines Speichers außerhalb der Einrichtung in einer zuvor definierten Sequenz schreibt (oder daraus liest).

[0026] Um bei einem Ausführungsbeispiel das System während des Kernel- und Treiber-Ladens zu initiieren, werden in Software wenigstens 2 KB des Speichers reserviert (auf einigen DIMM(s) **410**, **411**, **412** außerhalb der Einrichtung **406**) an einem physikalischen Ort an einer 2 KB-Grenze. Bei einem Ausführungsbeispiel werden die höchsten 1 MB unter dem Offset der Einrichtung reserviert. Bei einem Ausführungsbeispiel wird als nächstes der reservierte Bereich des Speichers als „nicht cachespeichbar" gesetzt, so daß Schreiboperationen in diesen Bereich sofort ausgeführt werden.

[0027] Da die Einrichtung **406** die Chipauswahl **408** nicht sieht, weiß sie bei einem Ausführungsbeispiel nicht, auf welchen DIMM **410**, **411**, **412** die gegebene Adresse des Hosts Bezug nimmt. Folglich ist bei einem Ausführungsbeispiel die unterscheidende Charakteristik zwischen Adreßaufrufen die Tiefe in den reservierten Bereich unabhängig davon, für welchen DIMM **410**, **411**, **412** der Aufruf vorgesehen ist. Wie zuvor ausgeführt, ist es egal, ob die Sequenz der Adreßaufrufe sich an nur einen DIMM **410**, **411**, **412** richtet oder ob sie sich an mehrere DIMMs **410**, **411**, **412** richtet.

[0028] Bei einer hypothetischen Sequenz von Adreßaufrufen wird bei einem Ausführungsbeispiel ein erster Speicheraufruf **413** an eine spezielle Adresse im dritten DIMM **412** gerichtet. Bei einem Ausführungsbeispiel wird dann ein zweiter Speicheraufruf **414** an eine spezielle Speichereinrichtung in dem zweiten DIMM **411** gerichtet und dann ein dritter Speicheraufruf **415** an einen speziellen Speicherplatz in dem ersten DIMM **410**. Schließlich wird bei einem Ausführungsbeispiel der vierte Speicheraufruf **416** an einen speziellen Speicherplatz in dem dritten DIMM **412** ausgeführt. Beim Wahrnehmen der vollständigen Sequenz führt die Einrichtung **406** das Umschalten aus.

[0029] Wie zuvor ausgeführt, können bei einem Ausführungsbeispiel sämtliche Adreßaufrufe für diese Sequenz an denselben DIMM **410**, **411**, **412** ge-

richtet werden, ohne das Ergebnis zu beeinflussen. Die einzige Differenz würde darin bestehen, welche Chipauswahl **408** freigegeben wird. Da die Einrichtung **406** die Chipauswahl **408** nicht sieht, ergäbe sich keine Änderung für das Ergebnis. Dieselbe Sequenz von Adreßaufrufen würde das „Erlange Bus" bewirken.

Patentansprüche

1. Computersystem mit
einem einen Prozessor umfassenden Host (**208**), der über einen Speicherbus (**206**) mit Speichermodulen (**204**) gekoppelt ist, und
einer mit dem Speicherbus (**206**) gekoppelten aktiven Speicherbusperipherieeinrichtung (**214**),
wobei die Speicherbusperipherieeinrichtung (**214**) ein Speicherbauelement (**210**), eine Logikschaltung (**202**) und einen von der Logikschaltung (**202**) gesteuerten Busumschalter (**212**), der einen Bus des Speicherbauelements (**210**) entweder mit dem Speicherbus (**206**) oder mit der Logikschaltung (**202**) verbindet, aufweist,
wobei die Logikschaltung (**202**) – unabhängig von der Stellung des Busumschalters (**212**) – über einen Signalabgriff (**215**) mit Adressleitungen des Speicherbusses (**206**) gekoppelt ist, so dass die Logikschaltung (**202**) auf dem Speicherbus (**206**) angesteuerte Adresswerte überwachen kann,
wobei die Logikschaltung (**202**) bei Erfassen einer vorgegebenen Sequenz von Adresswerten auf dem Speicherbus (**206**), die jeweils eines der neben der Speicherbusperipherieeinrichtung (**214**) mit dem Speicherbus (**206**) gekoppelten Speichermodule (**204**) an dem Speicherbus (**206**) adressieren, den Busumschalter (**212**) so steuert, dass dieser den Bus des Speicherbauelements (**210**) mit dem Speicherbus (**206**) verbindet.

2. Computersystem nach Anspruch 1, dadurch gekennzeichnet, dass die Logikschaltung (**202**) beim Erfassen wenigstens eines Adresswerts auf dem Speicherbus (**206**), der einen vorgegebenen Speicherplatz in dem Speicherbauelement (**210**) adressiert, den Busumschalter (**212**) so steuert, dass dieser den Bus des Speicherbauelements (**210**) mit der Logikschaltung (**202**) verbindet.

3. Computersystem nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass die Logikschaltung (**202**) ein feldprogrammierbares Gate-Array (FPGA) ist, jedes der Speichermodule (**204**) ein Dual-In-Line-Memory-Module (DIMM) ist und das Speicherbauelement (**210**) ein synchroner dynamischer Speicher mit wahlfreiem Zugriff (SDRAM) ist.

4. Computersystem nach einem der Ansprüche 1–3, dadurch gekennzeichnet, dass die Logikschaltung (**202**) eine Logik zum Verschlüsseln oder Entschlüsseln von aus dem Speicherbauelement gelesenen Daten und zum Einschreiben verschlüsselter bzw. entschlüsselter Daten in das Speicherbauelement aufweist.

5. Verfahren zur Steuerung einer aktiven Speicherbusperipherieeinrichtung (**214**) in einem Computersystem, das einen Host (**208**), einen mit dem Host (**208**) gekoppelten Speicherbus (**206**), mehrere mit dem Speicherbus (**206**) gekoppelte Speichermodule (**204**) und die aktive Speicherbusperipherieeinrichtung (**214**) umfasst, wobei die Speicherbusperipherieeinrichtung (**214**) ein Speicherbauelement (**210**), eine Logikschaltung (**202**), einen von der Logikschaltung (**202**) gesteuerten Busumschalter (**212**), der einen Bus des Speicherbauelements (**210**) entweder mit dem Speicherbus (**206**) oder mit der Logikschaltung (**202**) koppelt, und eine die Logikschaltung (**202**) mit dem Speicherbus (**206**) koppelnde Abgriffleitung (**215**) enthält, wobei:
Sequenzen von Adresswerten von dem Host über den Speicherbus an die Speichermodule übertragen werden, wobei die Adresswerte jeweils eines der Speichermodule adressieren,
die Sequenzen von Adresswerten über die Abgriffleitung von der Logikschaltung der Speicherbusperipherieeinrichtung erfasst werden, und
dann, wenn von der Logikschaltung eine vorgegebene Sequenz von Adresswerten auf der Abgriffleitung festgestellt wird (**328**), der Busumschalter so umgeschaltet wird (**330**, **332**), dass der Bus des Speicherbauelements mit dem Speicherbus gekoppelt wird.

6. Verfahren nach Anspruch 5, dadurch gekennzeichnet, dass dann, wenn die Logikschaltung wenigstens einen Adresswert auf dem Speicherbus erfasst (**320**), der einen vorgegebenen Speicherplatz in dem Speicherbauelement der Speicherbusperipherieeinrichtung adressiert, den Busumschalter so umgeschaltet wird, dass der Bus des Speicherbauelements mit der Logikschaltung gekoppelt wird.

7. Verfahren nach Anspruch 6, dadurch gekennzeichnet, dass dann, wenn der Bus des Speicherbauelements mit dem Speicherbus gekoppelt worden ist, Daten von dem Host aus dem Speicherbauelement gelesen oder in das Speicherbauelement geschrieben werden (**334**).

8. Verfahren nach Anspruch 6 oder 7, dadurch gekennzeichnet, dass dann, wenn der Bus des Speicherbauelements mit der Logikschaltung gekoppelt worden ist, Daten von der Logikschaltung aus dem Speicherbauelement gelesen (**322**) und verschlüsselt oder entschlüsselt werden und die verschlüsselten bzw. entschlüsselten Daten wieder von der Logikschaltung in das Speicherbauelement geschrieben werden (**324**).

Es folgen 6 Blatt Zeichnungen

Anhängende Zeichnungen

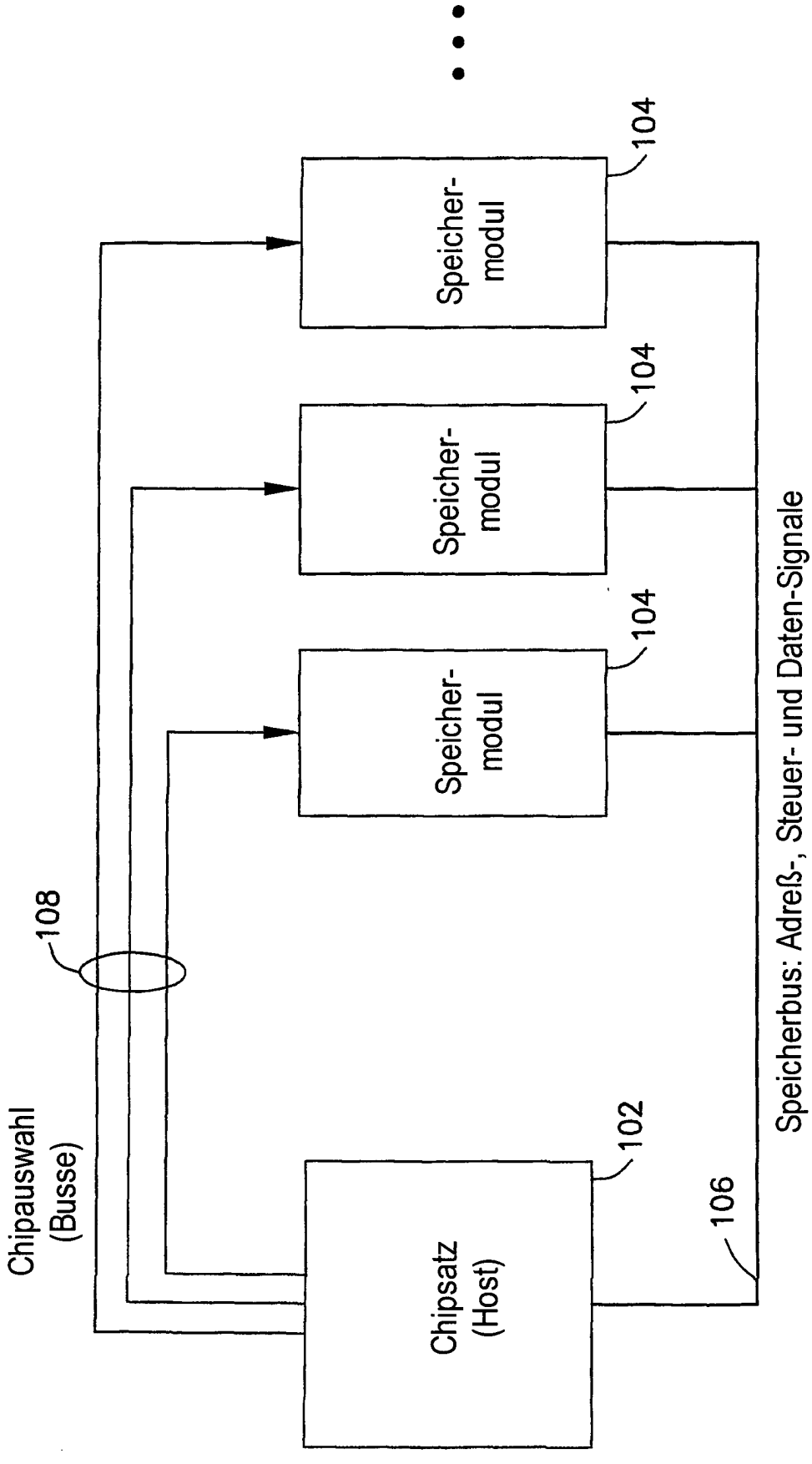


Fig. 1

Stand der Technik

Fig. 2

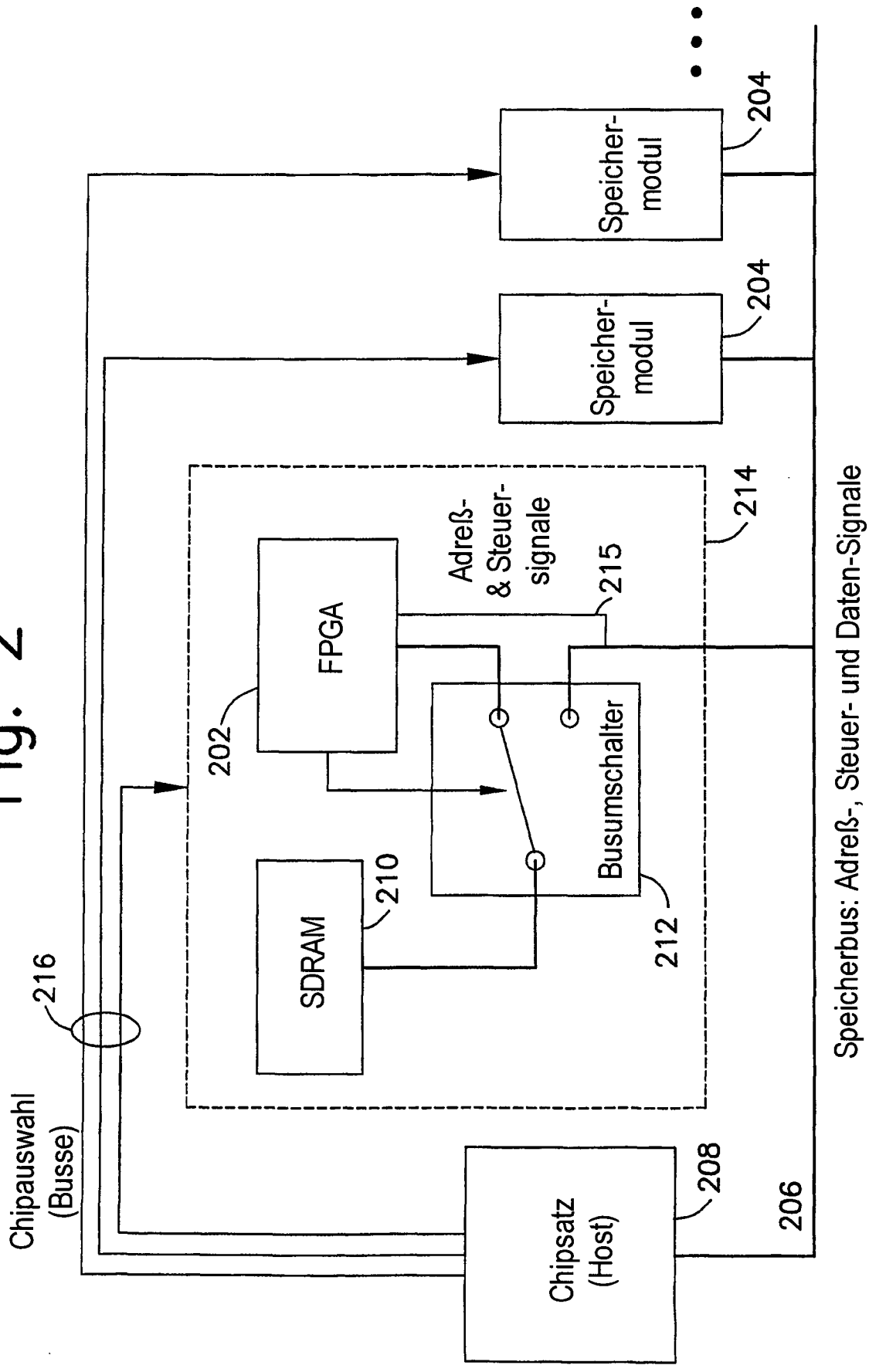

Chipauswahl (Busse)
216
Chipsatz (Host)
208
206
Speicherbus: Adreß-, Steuer- und Daten-Signale
Speicher-modul
204
Speicher-modul
204
214
Adreß- & Steuer-signale
215
FPGA
202
Busumschalter
212
SDRAM
210

Fig. 3a

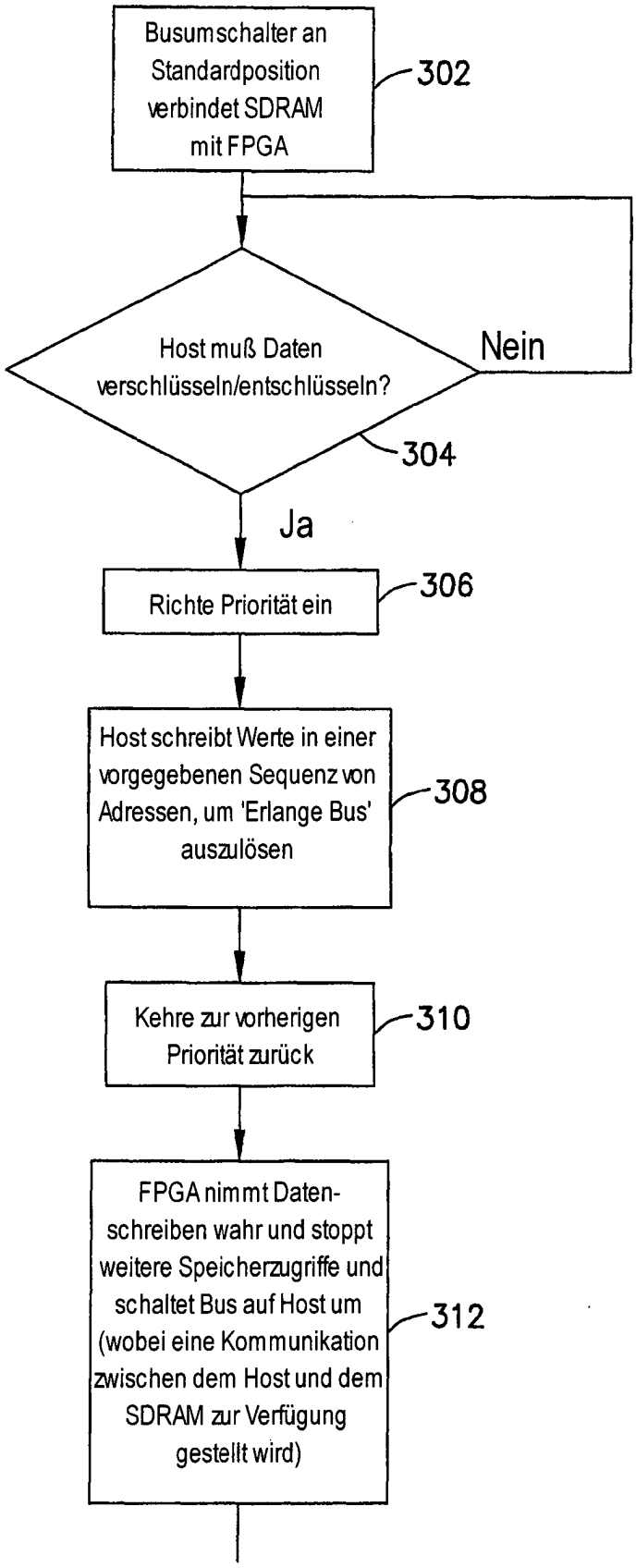
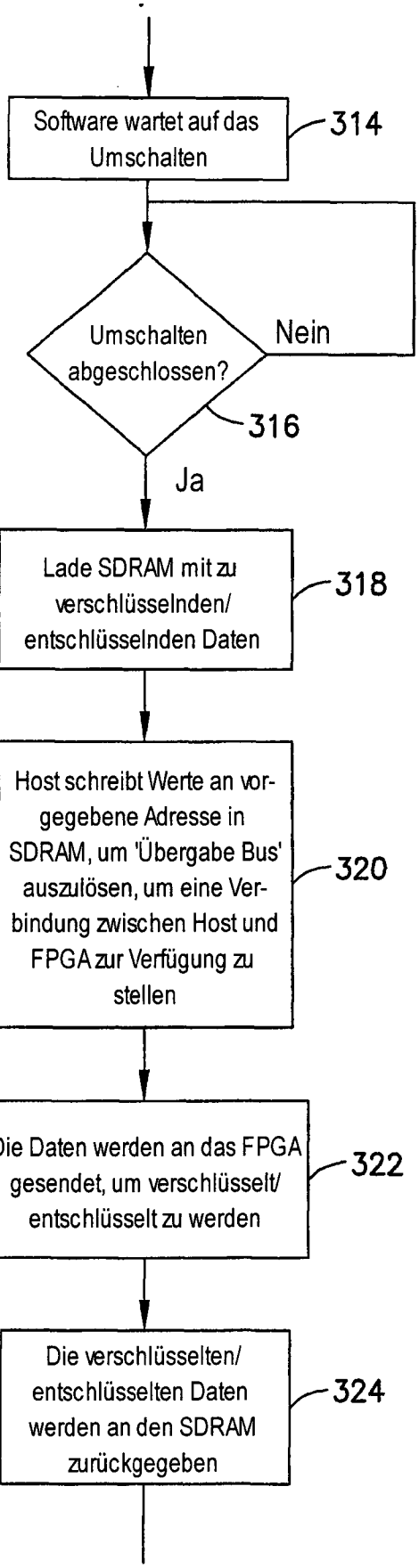

Busumschalter an Standardposition verbindet SDRAM mit FPGA
302
Host muß Daten verschlüsseln/entschlüsseln?
Nein
304
Ja
Richte Priorität ein
306
Host schreibt Werte in einer vorgegebenen Sequenz von Adressen, um 'Erlange Bus' auszulösen
308
Kehre zur vorherigen Priorität zurück
310
FPGA nimmt Daten-schreiben wahr und stoppt weitere Speicherzugriffe und schaltet Bus auf Host um (wobei eine Kommunikation zwischen dem Host und dem SDRAM zur Verfügung gestellt wird)
312

Fig. 3b


Software wartet auf das Umschalten
314
Umschalten abgeschlossen?
Nein
316
Ja
Lade SDRAM mit zu verschlüsselnden/ entschlüsselnden Daten
318
Host schreibt Werte an vorgegebene Adresse in SDRAM, um 'Übergabe Bus' auszulösen, um eine Verbindung zwischen Host und FPGA zur Verfügung zu stellen
320
Die Daten werden an das FPGA gesendet, um verschlüsselt/ entschlüsselt zu werden
322
Die verschlüsselten/ entschlüsselten Daten werden an den SDRAM zurückgegeben
324

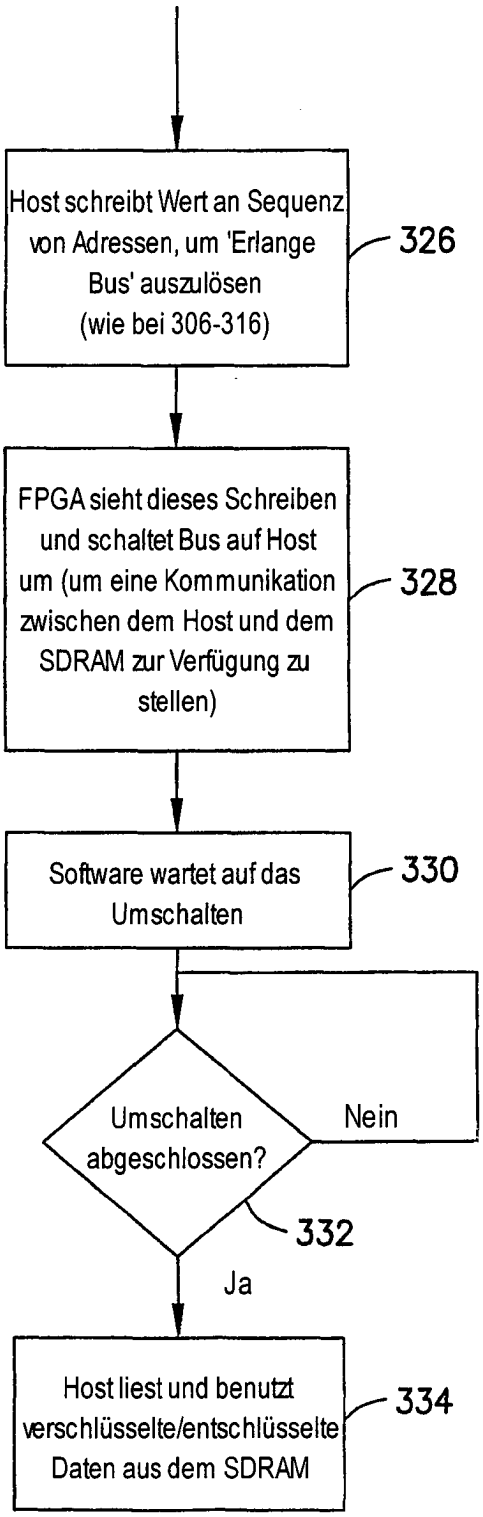

Host schreibt Wert an Sequenz von Adressen, um 'Erlange Bus' auszulösen (wie bei 306-316)
326
FPGA sieht dieses Schreiben und schaltet Bus auf Host um (um eine Kommunikation zwischen dem Host und dem SDRAM zur Verfügung zu stellen)
328
Software wartet auf das Umschalten
330
Umschalten abgeschlossen?
Nein
332
Ja
Host liest und benutzt verschlüsselte/entschlüsselte Daten aus dem SDRAM
334

Fig. 3c

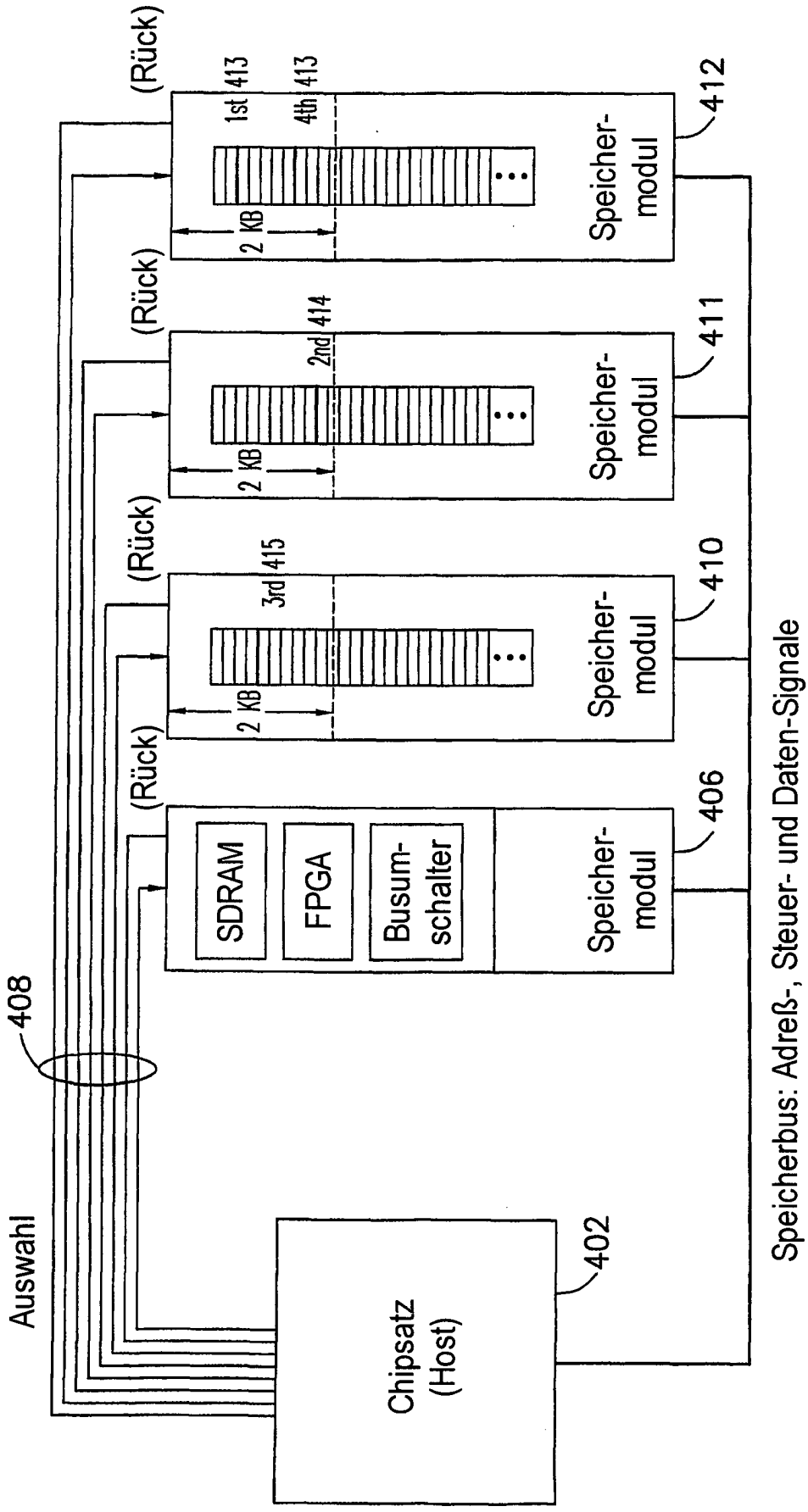

Auswahl
408
Chipsatz (Host)
402
Speicherbus: Adreß-, Steuer- und Daten-Signale
(Rück)
SDRAM
FPGA
Busum-schalter
Speicher-modul
406
3rd 415
2 KB
410
2nd 414
411
1st 413
4th 413
412

Fig. 4