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

(10) **DE 698 37 377 T2** 2007.11.29

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 927 938 B1**
(21) Deutsches Aktenzeichen: **698 37 377.4**
(86) PCT-Aktenzeichen: **PCT/JP98/01857**
(96) Europäisches Aktenzeichen: **98 917 635.9**
(87) PCT-Veröffentlichungs-Nr.: **WO 1998/048357**
(86) PCT-Anmeldetag: **22.04.1998**
(87) Veröffentlichungstag der PCT-Anmeldung: **29.10.1998**
(97) Erstveröffentlichung durch das EPA: **07.07.1999**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **21.03.2007**
(47) Veröffentlichungstag im Patentblatt: **29.11.2007**

(51) Int Cl.[8]: ***G06F 13/28*** (2006.01)
G06F 13/36 (2006.01)

(30) Unionspriorität:
10497997 22.04.1997 JP

(84) Benannte Vertragsstaaten:
AT, DE, ES, FR, GB, IT, NL

(73) Patentinhaber:
Sony Computer Entertainment Inc., Tokio/Tokyo, JP

(74) Vertreter:
Mitscherlich & Partner, Patent- und Rechtsanwälte, 80331 München

(72) Erfinder:
YAMAMOTO, Yasuyuki Sony Computer Entertainment, Tokyo 107-0052, JP

(54) Bezeichnung: **DATENÜBERTRAGUNGSSYSTEM UND VERFAHREN**

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

[0001] Diese Erfindung betrifft ein Datenübertragungsverfahren und eine Datenübertragungsvorrichtung. Die Erfindung kann auf das Übertragen von Daten zwischen Vorrichtungen bzw. Speichern, die mit zwei verschiedenen Bussen verbunden sind, und insbesondere auf ein Datenübertragungsverfahren und eine Datenübertragungsvorrichtung zum Ausführen von Diagnosen, ob die Datenübertragung normal funktioniert oder nicht, angewendet werden.

[0002] Bis heute ist ein System bekannt, bei dem verschiedene Busse, wie z. B. ein Hauptbus und ein Nebenbus über einen Bus-Repeater, wie z. B. ein Gateway usw., verbunden werden, um eine DMA-Übertragung von Daten zwischen diesen Bussen durch einen DMA-Controller (DMA: Direct Memory Access, Direkter Speicherzugriff), der in dem Hauptbus vorgesehen ist, durchzuführen.

[0003] Zum Beispiel sind bei dem Aufbau, der in Fig. 1 gezeigt ist, ein Hauptbus **101** und ein Nebenbus **102** miteinander mit einem Bus-Repeater **103**, wie z. B. einem Bus-Gateway usw., verbunden. Eine Vorrichtung **104**, wie z. B. eine CPU oder verschiedene Schnittstellen usw., und ein DMA-Controller **105** sind mit dem Hauptbus **101** verbunden, und eine Vorrichtung **106** und ein Speicher **107**, wie z. B. ein ROM usw., sind mit dem Nebenbus **102** verbunden.

[0004] In dem Beispiel der Fig. 1 steuert der DMA-Controller **105** auf dem Hauptbus **101** auch den Nebenbus **102** durch den Bus-Repeater **103**, um somit eine DMA-Übertragung z. B. zwischen der Vorrichtung **104** und der Vorrichtung **106** zu realisieren. Wie oben angegeben, kann eine effiziente Datenübertragung ohne eine nutzlose Warte-(Standby-)Zeit durchgeführt werden, wenn die Zugriffszeiten der entsprechenden Busse sogar zwischen verschiedenen Bussen **101**, **102** in derselben Größenordnung liegen.

[0005] Weiterhin gibt es in dem Fall, bei dem in einem System verschiedene Busse vorhanden sind, viele Situationen, in denen die Busbreiten und/oder die Datenzugriffsgeschwindigkeiten verschieden sind. Zum Beispiel weist in dem Beispiel der Fig. 1 der Hauptbus **101** eine Busbreite von 32 Bits und eine hohe Datenzugriffsgeschwindigkeit auf, und der Nebenbus **102** weist eine Busbreite von 16 Bits und eine niedrige Datenzugriffsgeschwindigkeit auf.

[0006] In dem Fall, bei dem die DMA-Übertragung zwischen den Bussen, deren Busbreiten und/oder Datenzugriffsgeschwindigkeiten verschieden sind, ausgeführt wird, besteht der Nachteil, dass eine nutzlose Warte-(Standby-)Zeit auf dem Hochgeschwindigkeitsbus, z. B. dem Hauptbus **101** der Fig. 1, besteht.

[0007] Angesichts des oben Beschriebenen ist es denkbar, zwei verschiedene Busse durch einen Pufferspeicher miteinander zu verbinden, um eine DMA-Übertragung durch diesen Pufferspeicher durchzuführen. Wenn versucht wird, eine Diagnose durchzuführen, ob eine DMA-Übertragung normal funktioniert, ist es in diesem Fall notwendig, den CPUs der entsprechenden Busse zu erlauben, in einem Debugging-Modus betrieben zu werden. Dies ist störend.

[0008] Weiterhin gibt es bei dem Fall, bei dem eine Datenübertragung nicht normal ausgeführt worden ist, viele Situationen, in denen die Ursachenforschung sehr schwierig wird, da angenommen wird, dass ein Defekt in den CPUs von beiden Bussen oder einem Diagnoseprogramm vorliegt.

[0009] Insbesondere in dem Fall, bei dem der CPU oder DMA-Controller usw. in einem LSI vorgesehen ist, benötigt es eine lange Entwicklungszeit, und ein Zeitplan für die Diagnosen usw. wird zu einem großen Problem.

[0010] Bevor der LSI entworfen wird oder als Prototyp in der Praxis hergestellt wird, wird eine Softwaresimulation auch der Peripheriegeräte ausgeführt, um die Funktionen soweit möglich zu bestätigen. Weil jedoch die Simulationszeit sehr hoch ist, kann eine ausreichende Überprüfung nicht ausgeführt werden, und es gibt tatsächliche Umstände, bei denen es erforderlich ist, die Probleme genauer in kurzer Zeit zu überprüfen; nachdem der Prototyp des LSI hergestellt worden ist. Somit gibt es viele Situationen, bei denen eine Schwierigkeit des Debuggens, wie oben beschrieben, zu einer Behinderung bei der Entwicklung des Produkts führt.

[0011] In dem Fall, bei dem die Datenübertragung zwischen Vorrichtungen ausgeführt wird, gibt es weiterhin Anordnungen von Daten, die für entsprechende Vorrichtungen geeignet sind. Um dies zu berücksichtigen, ist es notwendig, zusätzliche Daten abzurunden oder weitere Daten in den Abschnitt zwischen Datenzügen, die übertragen worden sind, einzufügen.

[0012] Wenn die CPU versucht, eine Operation wie oben beschrieben mit Bezug auf einen Datenzug, der in dem Speicher entwickelt ist, durchzuführen, liest diese einen solchen Datenzug einmal in das Register der CPU aus, um danach diese ein zweites Mal schreiben zu müssen. Aus diesem Grunde ist die Effizienz sehr schlecht (gering). Dies verringert die benötigte Zeit, wenn die CPU versucht, weitere Vorgänge durchzuführen, und ist daher nicht bevorzugt.

[0013] Hier ist es denkbar, dass der DMA-Controller bei der Datenübertragung jedes Mal die Übertragungsquelladresse oder Übertragungszieladresse ändert. In diesem Fall werden die Adressen der Übertragungsquelle und die Liste der Übertragungsquantität vorbereitet, und der DMA-Controller führt die DMA-Übertragung gemäß dieser Liste jedes Mal aus. Jedoch muss die CPU eine Übertragungsspezifikationstabelle erstellen. Als Ergebnis besteht die Schwierigkeit, dass jedes Mal ein Overhead zum Überprüfen der Übertragungsspezifikation stattfindet usw.

[0014] In dem Fall, bei dem verschiedene Busse nebeneinander in einem System, wie oben beschrieben, vorhanden sind, gibt es weiterhin viele Situationen, bei denen die Busbreiten verschieden sind. Zum Beispiel weist in dem Beispiel der Fig. 1 der Hauptbus **101** eine Busbreite von 32 Bits und eine hohe Datenzugriffsgeschwindigkeit auf, und der Nebenbus **102** weist eine Busbreite von 16 Bits und eine niedrige Datenzugriffsgeschwindigkeit auf. Auch sind in dem Fall, bei dem die DMA-Übertragung zwischen Bussen durchgeführt wird, bei denen die Busbreiten, wie oben angegeben, verschieden sind, Situationen, bei denen zusätzliche Daten abgerundet werden oder weitere Daten in den Abschnitt zwischen Datenzügen, die übertragen worden sind, eingefügt werden. Auch in diesem Fall ist es wünschenswert, dass eine Änderung der Datenstruktur oder eine Begrenzung der Adresse in einfacher Weise ausgeführt werden kann.

[0015] Die Offenlegungsschrift der japanischen Patentanmeldung Nummer JP-A-5 173 933 offenbart ein DMA-Übertragungssystem mit einem ersten und einem zweiten Bus, DMA-Controllern, die mit den entsprechenden Bussen verbunden sind, und einen Pufferspeicher, der zwischen den Bussen angeschlossen ist. Das Speichervolumen in dem Pufferspeicher wird gemäß der Datenübertragungsgeschwindigkeit erhöht oder erniedrigt, um eine schnelle Datenübertragung durchzuführen, ohne die jeweils andere Partei für eine lange Zeit warten zu lassen.

[0016] Die vorliegende Erfindung stellt ein Datenübertragungsverfahren nach Anspruch 1 und eine Datenübertragungsvorrichtung nach Anspruch 4 zur Verfügung.

[0017] Eine bevorzugte Ausführungsform dieser Erfindung, die nachfolgend beschrieben ist, stellt ein Datenübertragungsverfahren und eine Datenübertragungsvorrichtung zur Verfügung, die in der Lage sind, eine DMA-Übertragungsfunktion zwischen zwei verschiedenen Bussen in einfacher Weise und in einer kurzen Zeit auszuführen.

[0018] Es ist ebenfalls nachfolgend ein Datenübertragungsverfahren und eine Datenübertragungsvorrichtung beschrieben, die in der Lage sind, bei einer Datenübertragung zwischen zwei verschiedenen Bussen die Größe des Übertragungsdatenblockes in einfacher Weise, ohne eine Belastung der CPU zu verursachen, zu ändern, um so eine Verbesserung der Arbeitseffizienz zu realisieren.

[0019] Die Erfindung wird nachfolgend anhand darstellender Beispiele mit Bezug auf die beigefügten Zeichnungen beschrieben, in denen:

[0020] Fig. 1 ein Blockdiagramm zeigt, das ein herkömmliches Beispiel des Systems mit zwei Bussen darstellt;

[0021] Fig. 2 ein Blockdiagramm zeigt, das einen Abriss des Aufbaus einer Ausführungsform der Erfindung darstellt;

[0022] Fig. 3 ein Blockdiagramm zeigt, das ein Beispiel des internen Aufbaus des Bus-Repeaters, der in der Ausführungsform dieser Erfindung verwendet wird, darstellt;

[0023] Fig. 4 ein Flussdiagramm zeigt, um ein Beispiel der Arbeitsweise der Ausführungsform dieser Erfindung zu erläutern;

[0024] Fig. 5 ein Flussdiagramm zeigt, um ein weiteres Beispiel der Arbeitsweise der Ausführungsform dieser Erfindung zu erläutern;

[0025] Fig. 6 eine Ansicht zeigt, die ein praktischeres Beispiel eines Datenzuges vor der Übertragung darstellt;

[0026] Fig. 7 eine Ansicht zeigt, die ein praktischeres Beispiel eines Datenzuges, der durch Weglassen eines Abschnittes bei der Übertragung erhalten wird, darstellt;

[0027] Fig. 8 eine Ansicht zeigt, die ein praktischeres Beispiel eines Datenzuges darstellt, der durch Ersetzen von weiteren Daten bei der Übertragung erhalten wird;

[0028] Fig. 9 eine Ansicht zeigt, die ein Beispiel eines zu übertragenden Datenbereiches in einer schraffierten Darstellung darstellt,

[0029] Fig. 10 eine Ansicht darstellt, die eine Datenanordnung in dem Speicher der Fig. 9 zeigt,

[0030] Fig. 11 eine Ansicht zeigt, die ein Beispiel darstellt, bei dem Kopfabschnitte jeweils den mehreren Datenzügen hinzugefügt werden;

[0031] Fig. 12 eine Ansicht zeigt, die ein praktischeres Beispiel des Kopfabschnittes, der in Fig. 11 hinzugefügt wird, darstellt;

[0032] Fig. 13 ein Blockdiagramm zeigt, das ein Beispiel des Systems darstellt, auf das die Ausführungsform dieser Erfindung angewendet wird.

[0033] Die Ausführungsformen gemäß dieser Erfindung werden nun mit Bezug auf die beigefügten Zeichnungen beschrieben.

[0034] Fig. 2 zeigt ein Blockdiagramm, das einen Aufbau eines Systems darstellt, auf das ein Datenübertragungsverfahren, das der Ausführugsform gemäß dieser Erfindung entspricht, angewendet wird.

[0035] In dieser Fig. 2 sind sowohl ein erster Bus **11** und ein zweiter Bus **12** mit einem Bus-Repeater **13**, der mit Hilfe eines Pufferspeichers, wie z. B. einem FIFO usw., gebildet ist, verbunden, wodurch es ermöglicht wird, eine gegenseitige Übertragung von Daten zwischen den Bussen **11** und **12** durch diesen Bus-Repeater **13** durchzuführen. Eine CPU **21**, ein DMA-Controller **22** (DMA: Direct Memory Access – direkter Speicherzugriff), eine Vorrichtung **23** und einen Speicher **24** usw. sind mit dem ersten Bus **11** verbunden, und eine CPU **26**, ein DMA-Controller **27**, eine Vorrichtung **28** und ein Speicher **29** usw. sind mit dem zweiten Bus **12** verbunden.

[0036] Die Vorrichtung **23** kann eine DMA-Anfrage mit Bezug auf den DMA-Controller **22** ausgeben, und die Vorrichtung **28** kann eine DMA-Anfrage mit Bezug auf den DMA-Controller **27** ausgeben. Als diese Vorrichtungen **23**, **28** können genannt werden z. B. Codierer/Decodierer für Bild oder Sprache, eine Grafikeinheit für die Grafikverarbeitung, Bildbearbeitung oder Sprachverarbeitungs-IC usw., Festplatteneinheiten durch entsprechende dafür vorgesehene Schnittstellen, magnetooptische Platteneinheiten, Floppy-Disc-Einheiten und/oder Peripheriegeräte, wie z. B. eine CD-ROM-Einheit usw.. Der Bus-Repeater **13** kann entsprechend DMA-Anfragen (DREQ) mit Bezug auf die DMA-Controller **22**, **27** ausgeben. Durch diese DMA-Anfragen ist es möglich, jeden der mehreren DMA-Kanäle festzulegen. Zusätzlich kann der Bus-Repeater **13** z. B. eine Anfrage für ein Verwendungsrecht des Busses **12** (BREQ: Busanfrage) mit Bezug auf die CPU **26** auf den Bus **12** ausgeben und eine Antwort einer Busverwendungserlaubnis (BACK: Busbestätigung) von der CPU **26** empfangen (akzeptieren).

[0037] Ein Beispiel des Ausbaus des Bus-Repeaters **13**, der in einem solchen System der Fig. 2 verwendet wird, ist in Fig. 3 gezeigt. In dieser Fig. 3 ist der erste Bus **11** der Fig. 2 als in den Datenbus **11a** und den Adress-/Steuerbus **11b** unterteilt und der zweite Bus **12** als in einen Datenbus **12a** und Adress-/Steuerbus **12b** unterteilt angegeben. In dem Bus-Repeater **13** ist ein interner Bus **31** vorgesehen, der mit dem Datenbus **11a** des ersten Busses **11** verbunden ist, und ein interner Bus **32**, der mit dem Datenbus **12a** des zweiten Busses **12** verbunden ist, vorgesehen. Ein FIFO-(First in First out)Speicher **33** ist mit diesen internen Bussen **31**, **32** verbunden. Weiterhin kann eine Puffersteuereinheit **34** mit diesen internen Bussen **31**, **32** verbunden sein. Die Puffersteuereinheit **34** ist auch mit dem Adress-/Steuerbus **11b** des ersten Busses **11** und dem Adress-/Steuerbus **12b** des zweiten Busses **12** verbunden. Zusätzlich sind Steuersignalleitungen zum Durchführen der DMA-Anfrage (DREQ) oder Kanalbestimmung usw. zwischen der Puffersteuereinheit **34** und den DMA-Controllern **22**, **27** der Fig. 2 mit der Puffersteuereinheit **34** verbunden.

[0038] Ein Maskierungsmarker **35** entspricht einem Marker zum Unterdrücken des DMA seitens des Busses **11**. Wenn dieser Marker „Ein" oder „1" gesteuert durch die CPU **21** der Fig. 2 entspricht, wird die DMA-Anfrage (DREQ) von der Puffersteuereinheit **34** an den DMA-Controller **22** der Fig. 2 nicht ausgegeben. Der Maskierungsmarker **36** entspricht einem Marker zum Unterdrücken des DMA seitens des Busses **12**. Wenn dieser Marker „Ein" oder „1" gesteuert durch die CPU **26** der Fig. 2 entspricht, wird die DMA-Anfrage (DREQ von der Puffersteuereinheit **34** an den DMA-Controller **27** der Fig. 2 nicht ausgegeben.

[0039] DREQ-Bits **37** und **38** entsprechen Markern, die jeweils Ein/Aus (oder „1"/"0") der DMA-Anfrage mit Bezug auf die entsprechenden Busse **11** und **12** angeben, und diese Bits können von beiden CPUs **21**, **22** ausgelesen werden. Diese Marker der DREQ-Bits **37** und **38** sind nicht maskiert, sogar wenn die oben beschriebenen Maskierungsmarker **35**, **36** „Ein" entsprechen und die Zustände der DMA-Anfragen, die nicht als DMA-Anfragen (DREQ bei den entsprechenden DMA-Controllern **22**, **27** in Erscheinung treten, durch die entsprechende CPU **21**, **26** ausgelesen werden können.

[0040] Ein Bus-Gateway **39** dient dazu, die internen Busse **31** und **32** zu verbinden, so dass die CPU **21** seitens des Busses **11** der Fig. 2 auf den Bus **12** zugreift. Diese Maskierungsmarker **35**, **36**, die DREQ-Bits **37**, **38** und das Bus-Gateway **39** sind mit der Puffersteuereinheit **34** verbunden.

[0041] In dem in Fig. 3 gezeigten Bus-Repeater **13** dient der FIFO-Speicher **33** als der Puffer und ist ausgebildet, so dass eine Eingabe/Ausgabe von Daten hinsichtlich der Busse **11**, **12**, auf die durch die Puffersteuereinheit **34** zugegriffen wird, d. h. die internen Busse **31**, **32**, die damit verbunden sind, gesteuert wird. Die Puffersteuereinheit **34** steuert die Buszugriffsoperation des FIFO-Speichers **33** und gibt eine DMA-Anfrage (DREQ hinsichtlich der DMA-Controller **22**, **27** der entsprechenden Busse **11**, **12** aus, um ihre Antwort (DARK: DMA Acknowledge, DMA-Bestätigung) zu empfangen (zu akzeptieren). Diese DMA-Anfrage kann einen von mehreren DMA-Kanälen festlegen, um diese auszugeben. Die DMA-Kanalauswahlinformation von den DMA-Controllern **22**, **27** wird auch an diese Puffersteuereinheit **34** gesendet.

[0042] Wenn z. B. weiterhin die CPU **21** einen Zugriff auf die Adresse des Busses **12** zur Verfügung stellt, gibt die Puffersteuereinheit **34** des Bus-Repeaters **13** eine Anfrage des Benutzungsrechts auf den Bus **12** (BREQ: Busanfrage) bezüglich der CPU **26** des Busses **12** aus. Die CPU **26** sendet in Folge dieser Busanfrage an die Puffersteuereinheit **34** des Bus-Repeaters **13** eine Antwort, um eine Benutzung des Busses **12** (BACK: Busbestätigung) zuzulassen. Danach kann die CPU **21** auf den Maskierungsmarker **36** des internen Busses **32** des Bus-Repeaters **13**, den FIFO **33** und den DMA-Controller **27**, die Vorrichtung **28** und den Speicher **29** auf dem Bus **12**, usw. zugreifen.

[0043] In dem Fall, bei dem die DMA-Übertragung durch den Bus-Repeater **13** zwischen dem ersten Bus **11** und dem zweiten Bus **12** durchgeführt wird, ist es weiterhin notwendig, dass die DMA-Einstellungen (z. B. Datengröße usw.) der DMA-Controller **22** und **27** einander ohne Inkonsistenz entsprechen. Die CPU **21** führt eine DMA-Einstellung auf dem Bus bezüglich des DMA-Controllers **22** aus, und die CPU **26** führt eine DMA-Einstellung auf dem Bus hinsichtlich des DMA-Controllers **27** aus.

[0044] In dem Fall, bei dem z. B. eine Datenübertragung durch einen DMA von dem Speicher **24** des ersten Busses **11** zu dem Speicher **28** des zweiten Busses **12** durchgeführt wird, ist es notwendig, dass eine Einstellung so vorgenommen wird, dass DMA-Kanäle auf gleiche Datengrößen (Datenmengen) bezüglich des DMA von dem Speicher **24** an den Bus-Repeater **13** in dem DMA-Controller **22** des ersten Busses **11** und bezüglich des DMA von dem Bus-Repeater **13** an den Speicher **29** in dem DMA-Controller **27** des zweiten Busses **12** eingestellt sind. Der Verfahrensablauf, nachdem diese Einstellungen vorgenommen wurden, ist wie in Fig. 4 gezeigt.

[0045] In dieser Fig. 4 wird in einem ersten Schritt S61 eine DMA-Anfrage (DREQ von dem Bus-Repeater **13** an den DMA-Controller **22** ausgeführt. In einem nachfolgenden Schritt S62 stellt der DMA-Controller **22** eine Anfrage nach einem Benutzungsrecht des Busses **11** (BREQ bezüglich der CPU **21**, um ein Busbenutzungsrecht zu erhalten, um eine DMA-Übertragung von dem Speicher **24** an den Bus-Repeater **13** durchzuführen. Bei dem nachfolgenden Schritt S63 führ der Bus-Repeater **13** eine DMA-Anfrage (DREQ hinsichtlich des DMA-Controllers **27** aus. Bei dem nachfolgenden Schritt S64 führt der DMA-Controller **27** eine Anfrage für ein Benutzungsrecht des Busses **12** (BREQ bezüglich der CPU **26** aus, um ein Busbenutzungsrecht zu erhalten, um eine DMA-Übertragung von dem Bus-Repeater **13** an den Speicher **29** auszuführen.

[0046] In dem Fall, bei dem eine DMA-Übertragung von Daten von der Vorrichtung **28** auf den zweiten Bus **12** an die Vorrichtung **23** auf den ersten Bus **11** durchgeführt wird, ist es weiterhin notwendig, dass eine Ein-

stellung vorgenommen wird, so dass DMA-Kanäle bereitgestellt werden, die auf gleiche Datengrößen hinbezüglich des DMA von der Vorrichtung **28** zu dem Bus-Repeater **13** in dem DMA-Controller **27** und bezüglich des DMA von dem Bus-Repeater **13** an die Vorrichtung **23** in dem DMA-Controller **22** eingestellt sind. Der Verfahrensablauf, nachdem diese Einstellungen vorgenommen worden sind, ist wie in Fig. 5 gezeigt.

[0047] In dem ersten Schritt S71 der Fig. 5 führt die Vorrichtung **28** auf dem zweiten Bus **12** eine DMA-Anfrage (DREQ) hinsichtlich des DMA-Controllers **27** durch. Bei dem nachfolgenden Schritt S72 führt der Bus-Repeater **13** eine DMA-Anfrage (DREQ) hinsichtlich des DMA-Controllers **27** durch. In Schritt S73 antwortet der DMA-Controller **27** auf den Empfang der entsprechenden DMA-Anfragen von der Vorrichtung **28** und dem Bus-Repeater **13**, um eine Anfrage nach einem Verwendungsrecht des Busses **12** (BREQ) hinsichtlich der CPU **26** durchzuführen, um ein Busbenutzungsrecht zu erhalten, um eine DMA-Übertragung von der Vorrichtung **28** an den Bus-Repeater **13** durchzuführen. Gleichzeitig sendet die CPU **26** ähnlich zu der normalen DMA-Übertragung eine Antwort zurück an den DMA-Controller **27**, wenn dieser den Bus in Folge der Busanfrage (BREQ) freigibt, und der DMA-Controller **27** sendet eine DMA-Bestätigung (DARK) zurück an den Bus-Repeater **13** usw.. Bei dem nachfolgenden Schritt S74 führt der Bus-Repeater **13** eine DMA-Anfrage (DREQ) bezüglich des DMA-Controllers **22** an dem ersten Bus **11** aus. In Schritt S75 führt die Vorrichtung **23** eine DMA-Anfrage (DREQ) bezüglich des DMA-Controllers **22** aus. Bei dem nachfolgenden Schritt S76 antwortet der DMA-Controller **22** auf den Empfang der entsprechenden DMA-Anfragen von der Vorrichtung **23**, und der Bus-Repeater **13** führt eine Anfrage nach einem Benutzungsrecht des Busses **11** (BREQ) bezüglich der CPU **21** aus, um ein Busbenutzungsrecht zu erhalten, um eine DMA-Übertragung von dem Bus-Repeater **13** an die Vorrichtung **23** durchzuführen.

[0048] Es sollte angemerkt werden, dass, da die Kapazität des Speichers des Bus-Repeaters **13**, wie z. B. der FIFO usw., endlich ist, in dem Fall, bei dem Daten mit einer Größe über dieser Kapazität, übertragen werden, es ausreichend ist, eine Einstellung für eine Teilübertragung bezüglich der DMA-Controller **22**, **27** vorzunehmen, um die Verarbeitung der Schritte S61 bis S64 oder der Schritte S71 bis S76 zu wiederholen. Eine Übertragung (Block) in dieser Teilübertragung ist durch die Speicherkapazität des Bus-Repeaters **13** bestimmt.

[0049] Demgemäß wird die DMA-Übertragung zwischen zwei Bussen **11**, **12** durch den Pufferspeicher des Bus-Repeaters **13** durchgeführt, wodurch es ermöglicht wird, einen DMA zwischen verschiedenen Bussen durchzuführen, ohne dass sich eine Standby-(Warte-) Zeit daraus ergibt. Weiterhin wird zugelassen, dass mehrere DMA-Kanäle gleichzeitig betrieben werden, um dadurch die Verarbeitung der CPU zu vereinfachen, um eine einfache Programmierung zu ermöglichen und einen geringeren Overhead zu erreichen. Weiterhin können Puffer der Repeater zwischen den Bussen effizient genutzt werden. Zusätzlich kann ein Programm eines Multischlittens (multi-sled) in einfacher Weise geschrieben werden.

[0050] Wenn z. B. die CPU **21** auf den Bus **12** zugreift, stellt der Bus-Repeater **13** eine Anfrage nach einem Benutzungsrecht des Busses **12** (BREQ an die CPU **26**, um deren Antwort (BACK) zum Zugreifen auf den DMA-Controller **27**, auf die Vorrichtung **28** und auf den Speicher **29** usw. an dem Bus **12** zu empfangen (zu akzeptieren).

[0051] Gleichzeitig wird der FIFO **33** der Fig. 3 so betrieben, dass, wenn dieser die DMA-Anfragen (DREQ) an die entsprechenden DMA-Controller **22**, **27** durch die Maskierungsmarker **35**, **36** unterdrückt werden, I/O-Zugriffsoperationen an den entsprechenden Bussen **11**, **12** vorgenommen werden können. In diesem Fall muss ein solcher I/O-Zugriff äquivalent ohne eine Inkonsistenz mit der Verarbeitung des DMA durchgeführt werden. Demgemäß kann ein Zugriff nicht willkürlich vorgenommen werden, sondern der I/O-Zugriff wird unter der gleichen Bedingung als Zugriff des DMA zugelassen. Ein Beispiel der Bedingung dieses I/O-Zugriffs ist in der nachfolgenden Tabelle 1 gezeigt.

Tabelle 1

Zustand des Signals & Markers						21 Zugriff		22 Zugriff	
37	38	35	36	zurück	Übertragungs-richtung	Lesen	Schreiben	Lesen	Schreiben
0	0	*	*	*	*	X	x	x	x
1	0	0	*	*	*	X	x	x	x
1	0	1	*	*	12 → 11	O	x	x	x
1	0	1	*	*	11 → 12	X	o	x	x
0	1	*	0	*	*	X	x	x	x
0	1	*	1	0	12 → 11	X	x	o	x
0	1	*	1	0	11 → 12	X	x	x	o
0	1	*	1	1	12 → 11	O	x	x	x
0	1	*	1	1	11 → 12	X	o	x	x
0	1	*	1	1	11 → 12	X	o	x	x

„*" der Spalte des „Zustand des Signals und Marken" dieser Tabelle 1 stellt einen willkürlichen (nicht relevant)-Zustand dar.

[0052] In diesem Fall sind, da die Bedingung für die Diagnose der DMA-Funktion gegeben ist, nachfolgend beschriebene Fälle denkbar.

[0053] Bedingung #1: Datenübertragung wird mit Hilfe der DMA-Controller **22**, **27** entlang der Busse **11**, **12** durchgeführt.

[0054] Bedingung #2: CPU **21** führt eine Übertragung/Empfang von Daten durch einen I/O-Zugriff nur seitens des Busses **11** durch.

[0055] Bedingung #3: CPU **26** führt eine Übertragung/Empfang von Daten durch einen I/O-Zugriff nur seitens des Busses **12** durch.

[0056] Bdingung #4: Die CPUs **21**, **26** führen eine Übertragung/Empfang von Daten durch einen I/O-Zugriff sowohl seitens des Busses **11** als auch seitens des Busses **12** durch.

[0057] Bedingung #5: In dem Zustand, bei dem eine Vorrichtung an dem Bus **12** nicht vorhanden ist oder eine Vorrichtung an dem Bus **12** nicht verwendet wird, wird eine Datenübertragung durch den DMA-Controller **22** auf dem Bus **11** durchgeführt, und die CPU **21** führt eine Datenübertragung durch einen I/O-Zugriff auf dem Bus **12** durch.

[0058] Bedingung #6: In dem Zustand, bei dem eine Vorrichtung an dem Bus **12** nicht vorhanden ist oder eine Vorrichtung an dem Bus **12** nicht verwendet wird, führt die CPU **21** eine Datenübertragung durch einen I/O-Zugriff sowohl auf dem Bus **11** als auch auf dem Bus **12** durch.

[0059] Es sollte angemerkt werden, dass angenommen wird, dass eine Diagnose der DMA-Funktion in umgekehrter Reihenfolge von der Bedingung #6 ausgehend als Reihenfolge des tatsächlichen Debuggens durchgeführt wird.

[0060] Durch Durchführen einer Datenübertragung durch das nachfolgend beschriebenen Verfahren gemäß den oben beschriebenen entsprechenden Bedingungen #1 bis #6, um zu überprüfen, unter welcher Bedingung ein Problem auftritt, ist es möglich, nicht ordnungsgemäße Funktionseinheiten zu bestimmen.

Bedingung # 1

Maskierungsmarker **35**: Aus, Maskierungsmarker **36**: Aus.
Übertragungsrichtung: *
CPU **21**: Instruktion der Übertragung an DMAC **22**
CPU **26**: Instruktion der Übertragung an DMAC **27**

Bedingung #2

Maskierungsmarker **35**: Ein, Maskierungsmarker **36**: Aus
Übertragungsrichtung: Bus **11** → Bus **12**
CPU **26**: Instruktion der Übertragung an DMAC **27**
CPU **21**: Warten auf „Ein" des DREQ-Bits **37**, um Daten in den Bus-Repeater **13** zu schreiben.
Übertragungsrichtung: Bus **12** → Bus **11**
CPU **26**: Instruktion der Übertragung an DMAC **27**
CPU **21**: Warten auf „Ein" des DREQ-Bits **37**, um Daten von dem Bus-Repeater **13** auszulesen.

Bedingung #3

Maskierungsmarker **35**: Aus, Maskierungsmarker **36**: Ein
Übertragungsrichtung: Bus **11** → Bus **12**
CPU **21**: Instruktion der Übertragung an DMAC **22**
CPU **26**: Warten auf „Ein" des DREQ-Bits **38**, um Daten von dem Bus-Repeater **13** auszulesen
Übertragungsrichtung: Bus **12** → Bus **11**
CPU **21**: Instruktion der Übertragung an DMAC **22**
CPU **26**: Warten auf „Ein" des DREQ-Bits **38**, um Daten in den Bus-Repeater **13** zu schreiben.

Bedingung #4

Maskierungsmarker **35**: Ein, Maskierungsmarker **36**: Ein
Übertragungsrichtung: Bus **11** → Bus **12**
CPU **21**: Warten auf „Ein" des DREQ-Bits **37**, um Daten in den Bus-Repeater **13** zu schreiben
CPU **26**: Warten auf „Ein" des DREQ-Bits **38**, um Daten aus dem Bus-Repeater **13** auszulesen.
Übertragungsrichtung: Bus **12** → Bus **11**
CPU **26**: Warten auf „Ein" des DREQ-Bits **38**, um Daten in den Bus-Repeater **13** zu schreiben.
CPU **21**: Warten auf „Ein" des DREQ-Bits **37**, um Daten aus dem Bus-Repeater **13** auszulesen.

Bedingung#5

Maskierungsmarker **35**: Aus, Maskierungsmarker **36**: Ein
Übertragungsrichtung: Bus **11** → Bus **12**
CPU **21**: Instruktion des Übertragens an DMAC **22**
: Warten auf „Ein" des DREQ-Bits **38**, um Daten aus dem Bus-Repeater **13** auszulesen.
Übertragungsrichtung: Bus **12** → Bus **11**
CPU **21**: Befehl des Übertragens des DMAC **22**
: Warten auf „Ein" des DREQ-Bits **38**, um Daten in den Bus-Repeater **13** zu schreiben.

Bedingung #6

Maskierungsmarker **35**: Ein, Maskierungsmarker **36**: Ein
Übertragungsrichtung: Bus **11** → Bus **12**
CPU **21**: Warten auf „Ein" des DREQ-Bits **37**, um Daten in den Bus-Repeater **13** zu schreiben.
: Warten auf Ein des DREQ-Bits **38**, um Daten aus dem Bus-Repeater **13** auszulesen.
Übertragungsrichtung: Bus **12** → Bus **11**
CPU **21**: Warten auf „Ein" des DREQ-Bits **38**, um Daten in den Bus-Repeater **13** zu schreiben
: Warten auf Ein des DERQ-Bits **37**, um Daten aus dem Bus-Repeater **13** auszulesen.

[0061] Indem bestätigt wird, bei welcher Bedingung diese sechs Arten von Bedingungen des außergewöhnlichen Zustandes vorliegen, wird eine Diagnose des fehlerhaften Abschnittes oder Problempunktes durchgeführt. In diesem Fall können Daten, die auf den Bus **12** übertragen werden, durch ein Testprogramm der CPU

21 bestätigt werden, indem der CPU **21** erlaubt wird, direkt auf den Bus **12** zuzugreifen.

[0062] Demgemäß ist es gemäß einer solchen Ausführungsform der Erfindung möglich, die DMA-Übertragungsfunktion zwischen verschiedenen Bussen **11**, **12** in einfacher Weise zu debuggen, um so den Ort des Problems aufzudecken. Weiterhin kann nur durch die CPU **21** auf dem Bus **11**, der als Hauptbus dient, eine Bestätigung der Funktion durchgeführt werden. Weiterhin kann durch das direkte Bereitstellen eines Zugriffs auf den weiteren Bus **12** auch Debugging Exceptions (Debugging-Ausnahmebehandlungen) für DMA-Funktionen durchgeführt werden.

[0063] In Zusammenfassung der oben angeführten Beschreibung gemäß der Ausführungsform dieser Erfindung ist eine Bus-Repeater-Einrichtung mit einem Pufferspeicher zwischen dem ersten Bus und dem zweiten Bus vorgesehen, und eine erste Steuereinrichtung für einen direkten Speicherzugriff (DMA) ist mit dem ersten Bus und eine zweite Steuereinrichtung für einen direkten Speicherzugriff ist mit dem zweiten Bus verbunden, um durch den Pufferspeicher eine Übertragung im direkten Speicherzugriff zwischen einem Speicher oder einer Vorrichtung, die mit dem ersten Bus verbunden sind, und einem Speicher oder einer Vorrichtung, die mit dem zweiten Bus verbunden sind, durch diese erste und zweite Steuereinrichtung für den direkten Speicherzugriff durchzuführen, und um durch die Datenverarbeitungseinrichtung (CPU) Anfragen nach einem direkten Speicherzugriff an die erste und die zweite Steuereinrichtung für einen direkten Speicherzugriff von der Bus-Repeater-Einrichtung zu maskieren, um mit Hilfe der Datenverarbeitungseinrichtung direkt auf den Pufferspeicher in der Bus-Repeater-Einrichtung zuzugreifen, wodurch die Übertragung im direkten Speicherzugriff zwischen den entsprechenden Speichern oder Vorrichtungen auf verschiedenen Bussen durch diesen Pufferspeicher durchgeführt wird. Gleichzeitig maskieren die Datenverarbeitungseinrichtungen (CPUs) an den entsprechenden Bussen Anfragen nach einem direkten Speicherzugriff an den entsprechenden Bussen, um auf den Pufferspeicher direkt zuzugreifen, wodurch es ermöglicht wird, die Funktion des direkten Speicherzugriffs zu überprüfen. Weiterhin maskiert die Datenverarbeitungseinrichtung auf dem ersten Bus Anfragen nach direktem Speicherzugriff auf dem zweiten Bus, um auf dem Pufferspeicher seitens des zweiten Busses zuzugreifen, wodurch es ermöglicht wird, die Funktion des direkten Speicherzugriffs des zweiten Busses zu überprüfen.

[0064] Demgemäß ist es möglich, die Funktion der Übertragung im direkten Speicherzugriff zwischen verschiedenen Bussen in einfacher Weise zu debuggen, um den Ort des Problems aufzudecken. Weiterhin kann nur durch die Datenverarbeitungseinrichtung auf dem Hauptbus die Bestätigung der Funktion durchgeführt werden. Weiterhin können durch direktes Bereitstellen eines Zugriffs an dem weiteren Bus Debugging Exceptions (Ausnahmebehandlungen) für die Funktion des direkten Speicherzugriffs ebenfalls durchgeführt werden.

[0065] Nachfolgend wird das Beispiel beschrieben, bei dem die Blockgröße der Übertragungsdaten des Systems zum Ausführen der DMA-Übertragung durch den Puffer zwischen verschiedenen Bussen in einfacher Weise geändert werden kann.

[0066] In diesem Beispiel fügt der Bus-Repeater **13** Dummy-Daten beim Übertragen hinzu oder lässt einen Abschnitt von Daten weg, um dadurch die Blockgröße zu ändern.

[0067] In dem Fall, bei dem ein kontinuierlicher Datenzug von der Übertragungsquelle vollständig zu dem Übertragungsziel geschrieben werden soll, und da sie sich in einer DMA-Übertragung befinden, die oben mit Bezug auf die zuvor beschriebenen **Fig. 2** bis **Fig. 5** erläutert worden ist, werden nämlich Einstellungen vorgenommen, so dass eine Übertragungseinheit (Block) der Größe des FIFOs **33** in dem Bus-Repeater **13** gemeinsam mit den DMA-Controllern **22**, **27** entspricht, so dass eine effizienteste Übertragung durchgeführt werden kann.

[0068] In dem Fall, bei dem die Datenmenge, die dem Puffer (FIFO **33**) zugeführt wird, und die daraus ausgegebene Datenmenge voneinander verschieden sind, weist der Bus-Repeater **13** eine Funktion aus, um eine wie nachfolgend beschriebene Arbeitsweise durchzuführen.

[0069] Erstens gibt nämlich in dem Fall, bei dem, sogar wenn zugeführte Daten aus dem Puffer (FIFO **33**) verloren gehen, weiterhin eine Ausgabeanfrage ausgegeben wird, der Bus-Repeater **13** Dummy-Daten aus.

[0070] Zweitens werden in dem Fall, bei dem zugeführte Daten im dem Puffer (FIFO **33**) zum Zeitpunkt, wenn der DMA der Ausgangsseite vollständig durchgeführt worden ist, verblieben sind, die verbliebenen Daten verworfen.

[0071] Durch die Funktion eines solchen Bus-Repeaters **13** ist es möglich, die Konfigurationen des zu über-

tragenden Datenzuges zu variieren. Als Wechsel der Konfiguration dieses Datenzuges wird beschrieben, dass Dummy-Daten bei der DMA-Übertragung hinzugefügt werden, um dadurch die Größe des Datenblockes zu vergrößern, und ein Abschnitt von Daten wird bei der Datenübertragung weggelassen, um dadurch die Größe des Datenblocks zu verringern.

[0072] Ein praktischeres Beispiel in dem Fall, bei dem die Kapazität des Puffers (FIFO **33**) des Bus-Repeaters **13** 64 Bytes entspricht, wird nachfolgend beschrieben.

[0073] Es wird z. B. angenommen, dass die Datenzüge S_{1A}, S_{1B}, S_{1C}, S_{2A},..., wie in Fig. 6 gezeigt, in dem Speicher **29** der Fig. 2 vorliegen. A, B, C der Indizes dieser Datenzüge S_{1A}, S_{1B}, S_{1C}, S_{2A},... stellen Daten dar, die voneinander in ihrer Art verschieden sind, und der Fall, bei dem nur Daten der Arten von B, C dieser drei Arten von Daten an die Vorrichtung **23** der Fig. 2 übertragen werden, wird angenommen. Dabei ist es notwendig, Datenzüge, wie in Fig. 7 gezeigt, in die Vorrichtung **23** zu schreiben. In diesem Fall zeigen die Fig. 1, Fig. 2, Fig. 3 ... der Indizes der Datenzüge S_{1A}, S_{1B}, S_{1C}, S_{2A} ... Nummern der Datenblöcke, die jeweils einer Übertragungseinheit entsprechen.

[0074] In dem DMA-Controller **27** wird eine Einstellung einer solchen DMA-Übertragung vorgenommen, um drei Blöcke von dem Datenzug S_{1B} mit allen Arten (A, B, C) der Daten S_{1A}, S_{1B}, S_{1C}, S_{2A},..., d. h. 32 Bytes als ein Block zu übertragen. Im Gegensatz dazu wird in dem DMA-Controller **22** eine Einstellung so vorgenommen, um drei Blöcke mit nur den zwei Arten von B, C, d. h. 24 Bytes als ein Block zu übertragen. Somit werden 8 Bytes (Datenzüge S_{2A}, S_{3A}) die auf die Datenzüge (S_{1C}, S_{2C}) der Art C folgen, verworfen. Aus diesem Grund können folglich Datenzüge, die in Fig. 7 gezeigt sind, in die Vorrichtung **23** geschrieben werden.

[0075] Dann, wird der Fall angenommen, bei dem versucht wird, Datenzüge des in Fig. 7 angegebenen Formats von der Vorrichtung **23** der Fig. 2 auszugeben, um solche Datenzüge in z. B. den Speicher **29** zu schreiben, und den Datenzug der Art D nach den Datenzügen der Art C, die in Fig. 8 gezeigt sind, einzufügen.

[0076] In diesem Fall wird in dem DMA-Controller **27** eine Einstellung so vorgenommen, um drei Blöcke zu übertragen, wobei 36 Bytes einem Block entsprechen. Gleichzeitig werden als Ergebnis der DMA-Übertragung Dummy-Daten von 12 Bytes in den Abschnitt nach dem Datenzug von 24 Bytes der Arten B, C geschrieben. Aus diesem Grund kann die CPU **26** der Fig. 2 direkt einen Datenzug der Art D in diesen Dummy-Daten-Bereich schreiben. Die CPU **26** kann nämlich Rechenkapazität zum Verschieben von Daten in den Speicher **29** einsparen.

[0077] Es sollte angemerkt werden, dass es in diesen Fällen, dass z. B. die Vorrichtung **28** der Fig. 2 Datenzüge, wie sie in Fig. 8 gezeigt sind, ausgibt, um nur Datenzüge der Arten von B, C in die Vorrichtung **23** zu schreiben, ausreichend ist, eine Einstellung in dem DMA-Controller **27** vorzunehmen, um drei Blöcke mit 36 Bytes von Beginn des Datenzugs an als ein Block zu übertragen, und das es in dem DMA-Controller **27** ausreichend ist, drei Blöcke mit 24 Bytes als ein Block zu übertragen.

[0078] Durch Vorsehen, dass der Bus-Repeater **13**, der in die Übertragung eingreift, um zusätzlich die oben beschriebene Funktion vorzunehmen, ist es möglich, eine Konvertierung einer einfachen Datenkonfiguration bei der Übertragung des Datenzuges durchzuführen, wodurch es ermöglicht wird, die Leistungsfähigkeit des Systems zu verbessern.

[0079] Nachfolgend wird in Verbindung mit den beigefügten Zeichnungen ein praktischeres Beispiel erläutert, bei dem die oben beschrieben zusätzlichen Daten abgerundet werden, oder weitere Daten in den Abschnitt zwischen den Datenzügen eingefügt werden.

[0080] Fig. 9 zeigt einen Textur-Bild-Bereich für ein Texturmapping bei der Bildverarbeitung, wie z. B. bei Computergrafiken usw., und es wird angenommen, dass ein Abschnitt innerhalb des breiten Texturbereichs, z. B. Bereich T_1, T_2, T_3 und T_4, die durch schräge Linienabschnitte in der Figur angegeben sind, übertragen werden. Dieses Texturbild wird, wie in Fig. 10 gezeigt ist, z. B. in dem Speicher vorgesehen, und es ist erforderlich, dass, wenn die Daten der Teilbereiche T_1, T_2, T_3 und T_4 in diesen Speicher übertragen werden, die zusätzlichen Daten herausgerundet werden.

[0081] Als Beispiel des Hinzufügens von Daten wird der Fall beschrieben, bei dem jeweils den mehreren Datenzügen, die in Fig. 11 gezeigt sind, Kopfabschnitte hinzugefügt werden, z. B. Polygondaten. Mit Bezug auf die Polygondaten ändert sich die Größe des Datenzuges nämlich abhängig von der Anzahl von Eckpunkten oder des Vorhandenseins/Nicht-Vorhandenseins einer Schattierung oder einer Textur usw. Weiterhin gibt es

zum Zwecke der Unterscheidung von den Texturdaten oder Unterscheidung der Übertragungsziele Instanzen, bei denen ein Kopfabschnitt (GPUIF-TAG) angehängt wird. Daten, die durch eine Schnittstelle GPUIF einer GPU (Grafikverarbeitungseinheit) übertragen werden, entsprechen solchen Daten, dass eine Gruppe von Daten, die als Primitiv bezeichnet wird, die durch den einleitenden Kopfabschnitt (GPUIFtag) gebildet wird, und nachfolgende Daten als die zugrunde legende Einheit vorgesehen werden, und mehrere Primitive, die gemeinsam verarbeitet werden, werden als ein gemeinsames GPU-Paket vorgesehen. Fig. 12 zeigt ein Beispiel des Aufbaus des Kopfabschnittes (GPUIFtag) und ist aus einem Registerbeschreiber REGS, einer Registerbeschreibernummer NREG und einer Datenform FLG usw. in der Reihenfolge ausgehend von dem MSB zusammengesetzt. In dem Fall, bei dem der Kopfabschnitt (GPUIFtag) den oben beschriebenen Polygondaten hinzugefügt wird, ist es notwendig, weitere Daten in den Abschnitt zwischen den Datenzügen einzufügen.

[0082] In dem Fall, bei dem die Konvertierung der einfachen oben beschriebenen Datenkonfiguration erforderlich ist, wird die oben beschriebene Ausführungsform der Erfindung verwendet, wodurch es möglich wird, Berechnungen mit schlechter Effizienz wegzulassen, bei denen die CPU ein Sequenzieren des Datenzugs durchführt, der in den Speicher per DMA übertragen wird, oder des Datenzugs, der in den Speicher per DMA übertragen werden soll, durchführt. Somit wird die Leistungsfähigkeit des Systems verbessert. Weiterhin ist es möglich, eine DMA-Übertragung zwischen Vorrichtungen mit verschiedenen Datenformaten durchzuführen. Weiterhin kann eine Berechnung für das Vorbereiten der Übertragung einer Quellenadresse einer speziellen Spezifikation oder Tabelle der Übertragungsmenge oder eine Bezugnahme auf die Tabelle weggelassen werden.

[0083] Wie durch die obige Beschreibung offensichtlich ist, ist gemäß dem Beispiel, das gemeinsam mit den Fig. 6 bis Fig. 11 der Ausführungsform der Erfindung beschrieben worden ist, die Bus-Repeater-Einrichtung mit einem Pufferspeicher zwischen dem ersten Bus und dem zweiten Bus vorgesehen, um durch den Pufferspeicher innerhalb der Bus-Repeater-Einrichtung eine Datenübertragung zwischen dem ersten Bus und dem zweiten Bus durchzuführen. Die Bus-Repeater-Einrichtung fügt damit Daten bei der Datenübertragung hinzu, um dadurch die Größe des Übertragungsdatenblocks zu vergrößern oder lässt einen Abschnitt von Daten bei der Datenübertragung weg, um dadurch die Größe des Datenübertragungsblocks zu verringern, wodurch es möglich wird, eine Änderung eines einfachen Datenaufbaus bei der Übertragung durchzuführen. Somit kann die Leistungsfähigkeit des Systems verbessert werden.

[0084] Weiterhin wird eine erste Steuereinrichtung für einen direkten Speicherzugriff mit dem ersten Bus verbunden, um eine Steuerung der Datenübertragung zwischen dem ersten Bus und dem Pufferspeicher in der Bus-Repeater-Einrichtung durchzuführen, und eine zweite Steuereinrichtung für einen direkten Speicherzugriff mit dem zweiten Bus verbunden, um eine Steuerung der Datenübertragung zwischen dem zweiten Bus und dem Pufferspeicher innerhalb der Bus-Repeater-Einrichtung durchzuführen. Die Bus-Repeater-Einrichtung wird mit einer Funktion versehen, um Dummy-Daten auszugeben, wenn eine Ausgabeanfrage ausgegeben wird, sogar wenn Daten in dem Pufferspeicher bei der Datenübertragung verloren gehen, und mit einer Funktion zum Verwerfen von verbleibenden Daten, wenn Daten in dem Pufferspeicher zu dem Zeitpunkt verbleiben, zu dem die Datenübertragung der Ausgangsseite bei der Datenübertragung vollständig durchgeführt worden ist. Somit kann eine wenig effiziente Berechnung vermieden werden, so dass die CPU usw. ein Sequenzieren von Daten in dem Speicher durchführt, der an einer Datenübertragung beteiligt war oder der an einer Datenübertragung beteiligt werden soll. Weiterhin kann auf eine Berechnung zum Vorbereiten der oder für die Bezugnahme auf die Übertragung der Quelladresse für die Datenübertragung oder eine Spezifikationstabelle für die spezielle Übertragung der Übertragungsmenge usw. verzichtet werden. Zusätzlich ist es möglich, eine Datenübertragung zwischen den Vorrichtungen mit verschiedenen Datenformaten durchzuführen.

[0085] Fig. 13 zeigt ein Beispiel des Systems, auf das die Ausführungsform dieser Erfindung, die oben beschrieben wurde, angewendet wird. In diesem System sind ein Hauptbus **111** zum Durchführen einer Hochgeschwindigkeitsbildverarbeitung und ein Nebenbus **112**, die mit Peripheriegeräten, wie z. B. ein CD-ROM-Laufwerk usw. mit niedriger Geschwindigkeit verbunden sind, über einen Bus-Repeater **113** mit einem Pufferspeicher, wie z. B. einem FIFO usw. verbunden.

[0086] In Fig. 13 sind nämlich eine Haupt-CPU **121**, ein DMA-Controller **122**, eine Grafikmaschine **123** für eine Hochgeschwindigkeitsbildverarbeitung und ein Hauptspeicher **124** mit dem Hauptbus **111** für hohe Übertragungsgeschwindigkeit verbunden, und eine Neben-CPU **126**, ein DMA-Controller **127**, ein Datenaufzeichnungsmedium **128** wie z. B. eine CD-ROM usw., und ein Nebenspeicher **129** mit dem Nebenbus **122** mit relativ niedriger Übertragungsgeschwindigkeit verbunden. Dieser Hauptbus **111** und Nebenbus **112** sind über den Bus-Repeater **113** mit dem Pufferspeicher wie z. B. dem FIFO usw., der oben beschrieben ist, verbunden, und dieser Bus-Repeater **113** kann mehrere Arten von DMA-Anfragen entsprechend den mehreren DMA-Kanälen,

z. B. drei Arten von DMA-Anfragen, ausgeben. Da es ausreichend ist, dass ein praktischerer Aufbau und Arbeitsweise dieses Bus-Repeaters **113** ähnlich des Bus-Repeaters **113** der Ausführungsform, die mit Bezug auf die Fig. 2 bis Fig. 5 beschrieben worden ist, durchgeführt wird, wird dessen Erläuterung weggelassen.

[0087] In dem Fall, bei dem die DMA-Übertragung zwischen dem Bus mit hoher Übertragungsgeschwindigkeit und dem Bus mit niedriger Übertragungsgeschwindigkeit in oben beschriebener Weise durchgeführt wird, kann die Datenübertragung durchgeführt werden, ohne zuzulassen, dass eine nutzlose Standby-Zeit auf dem Bus mit hoher Übertragungsgeschwindigkeit vorkommt, und die Arbeitsweise der CPU kann vereinfacht werden. Zusätzlich ist es möglich, das Debuggen der Funktion der DMA-Übertragung zwischen verschiedenen Bussen einfacher durchzuführen.

[0088] Es sollte angemerkt werden, dass diese Erfindung nicht auf die Ausführungsformen beschränkt ist. Während diese Erfindung in Verbindung mit dem Beispiel, bei dem eine bidirektionale DMA-Übertragung zwischen dem ersten Bus und dem zweiten Bus durchgeführt wird, beschrieben worden ist, kann diese Erfindung auch für einen Fall angewendet werden, bei dem nur eine DMA-Übertragung von dem ersten Bus zu dem zweiten Bus oder nur eine DMA-Übertragung von dem zweiten Bus zu dem ersten Bus durchgeführt werden. Zusätzlich ist es selbstverständlich, dass die Anzahl der DMA-Kanäle und/oder Schaltungen, die mit den entsprechenden Bussen usw. verbunden sind, nicht auf diejenigen der Ausführungsformen beschränkt sind.

Patentansprüche

1. Datenübertragungsverfahren mit folgenden Schritten:
Verbinden eines ersten Busses (**11**) und eines zweiten Busses (**12**), die voneinander verschieden sind, über eine Bus-Repeater-Einrichtung (**13**), die einen Pufferspeicher (**33**) aufweist;
Ausführen einer Datenübertragungssteuerung zwischen dem ersten Bus (**11**) und dem Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) durch eine erste Steuereinrichtung (**22**) für einen direkten Speicherzugriff, die mit dem ersten Bus (**11**) verbunden ist;
Ausführen einer Datenübertragungssteuerung zwischen dem zweiten Bus (**12**) und dem Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) durch eine zweite Steuereinrichtung (**27**) für einen direkten Speicherzugriff, die mit dem zweiten Bus (**12**) verbunden ist;
gekennzeichnet durch:
selektives Maskieren einer Anfrage für einen direkten Speicherzugriff auf die erste oder zweite Steuereinrichtung (**22**/**27**) für den direkten Speicherzugriff von der Bus-Repeater-Einrichtung (**13**) durch die Datenverarbeitungseinrichtung (**21**/**26**), um direkt während der Ausführung einer Diagnoseprozedur zum Diagnostizieren, ob die Steuereinrichtung für den direkten Speicherzugriff (**22**/**27**) korrekt funktioniert, auf den Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) zuzugreifen.

2. Datenübertragungsverfahren nach Anspruch 1, wobei:
die Datenverarbeitungseinrichtung (**21**/**26**) einen ersten Datenprozessor (**21**), der an dem ersten Bus (**11**) vorgesehen ist, und einen zweiten Datenprozessor umfasst, der an dem zweiten Bus (**12**) vorgesehen ist;
wobei der erste Datenprozessor eine Anfrage für einen direkten Speicherzugriff auf dem ersten Bus (**11**) während der Diagnoseprozedur selektiv maskiert;
wobei der zweite Datenprozessor (**26**) eine Anfrage für einen direkten Speicherzugriff auf dem zweiten Bus (**12**) während der Diagnoseprozedur selektiv maskiert; und
wobei der erste und der zweite Datenprozessor (**21**/**26**) auf den Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) mit derselben Übertragungsbedingung wie die erste und zweite Steuereinrichtung (**22**/**27**) für den direkten Speicherzugriff auf den entsprechenden Bus direkt zugreift.

3. Datenübertragungsverfahren nach Anspruch 1, wobei
die Datenverarbeitungseinrichtung (**21**/**26**) einen Datenprozessor (**21**) umfasst, der an dem ersten Bus (**11**) vorgesehen ist;
wobei der Datenprozessor (**21**) eine Anfrage für einen direkten Speicherzugriff auf dem zweiten Bus (**12**) während der Diagnoseprozedur selektiv maskiert; und
wobei der Datenprozessor (**21**) auf den Pufferspeicher (**33**) innerhalb der Bus-Repeater-Einrichtung (**13**) seitens des zweiten Busses direkt zugreift.

4. Datenübertragungsvorrichtung umfassend:
einen ersten Bus (**11**) und einen zweiten Bus (**12**);
eine Bus-Repeater-Einrichtung (**13**) mit einem Pufferspeicher (**33**), die sowohl mit dem ersten als auch dem zweiten Bus (**11**, **12**) verbunden ist;

eine Steuereinrichtung (**22**) für einen direkten Speicherzugriff, die mit dem ersten Bus (**11**) verbunden ist; und einen Datenprozessor (**21**), der mit dem ersten Bus (**11**) verbunden ist;
wobei die Bus-Repeater-Einrichtung (**13**) eine Funktion aufweist, um eine Anfrage für einen direkten Speicherzugriff an die Steuereinrichtung (**22**) für den direkten Speicherzugriff abzugeben, um direkt auf den Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) zuzugreifen, und wobei der Datenprozessor (**21**) dadurch gekennzeichnet ist, dass er eine Funktion aufweist, um diese Anfrage für den direkten Speicherzugriff selektiv während der Ausführung einer Diagnoseprozedur zum Diagnostizieren, ob die Steuereinrichtung (**22**) für den direkten Speicherzugriff korrekt arbeitet, zu maskieren.

5. Datenübertragungsvorrichtung nach Anspruch 4, wobei:
eine zweite Steuereinrichtung (**27**) für den direkten Speicherzugriff und ein zweiter Datenprozessor (**26**) mit dem zweiten Bus (**12**) verbunden sind;
und die Steuereinrichtung (**22**) für den direkten Speicherzugriff und die zweite Steuereinrichtung (**27**) für den direkten Speicherzugriff eine Lese-/Schreiboperation von Daten mit Bezug auf den Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) ausführen, um so eine Datenübertragung zwischen dem ersten und dem zweiten Bus (**11**, **12**) auszuführen; und
die Bus-Repeater-Einrichtung (**13**) eine Funktion aufweist, um eine Anfrage für einen direkten Speicherzugriff an die zweite Steuereinrichtung für den direkten Speicherzugriff (**27**) abzugeben, um direkt auf den Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) zuzugreifen, und wobei der zweite Datenprozessor (**26**) eine Funktion aufweist, um die Anfrage für den direkten Speicherzugriff während der Diagnoseprozedur selektiv zu maskieren.

6. Datenübertragungsvorrichtung nach Anspruch 5, wobei der Datenprozessor (**21**) oder der zweite Datenprozessor (**26**) während der Diagnoseprozedur selektiv eine Anfrage für einen direkten Speicherzugriff auf den zweiten Bus (**12**) bzw. auf den ersten Bus (**11**) in der Bus-Repeater-Einrichtung (**13**) maskieren, um auf den Pufferspeicher (**33**) in der Bus-Repeater-Einrichtung (**13**) von dem zweiten Bus (**12**) bzw. von dem ersten Bus (**11**) zuzugreifen.

Es folgen 10 Blatt Zeichnungen

Anhängende Zeichnungen

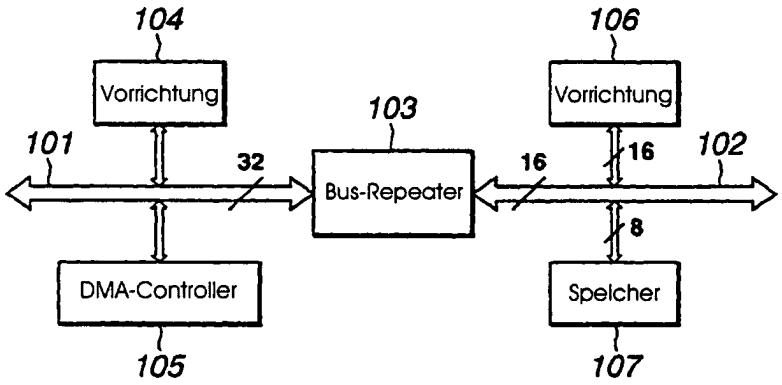


FIG.1

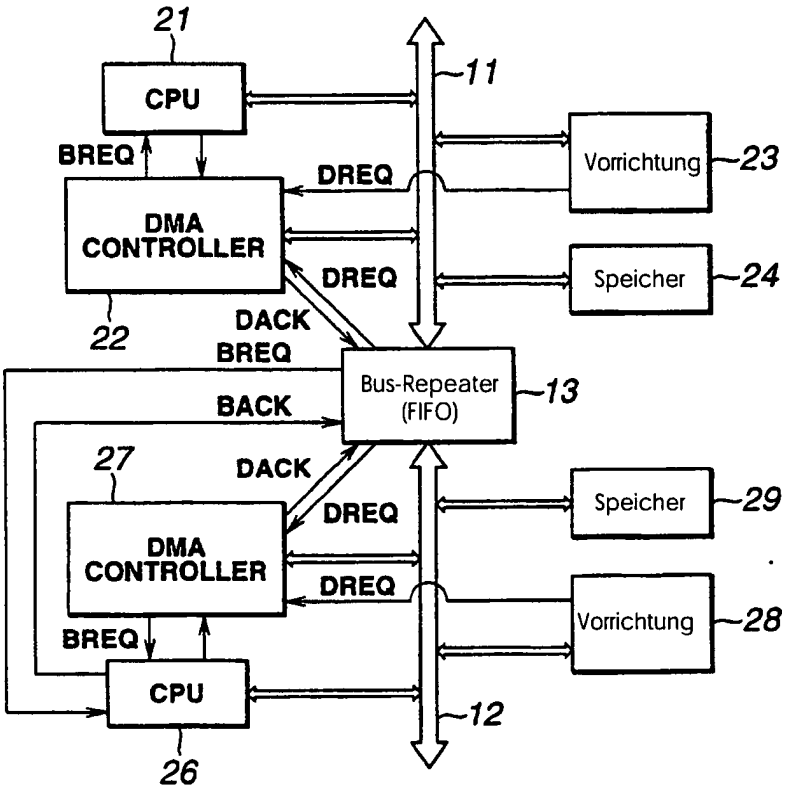

21
CPU
11
BREQ
DREQ
Vorrichtung
23
DMA CONTROLLER
DREQ
Speicher
24
22
DACK
BREQ
Bus-Repeater (FIFO)
13
BACK
27
DACK
DREQ
Speicher
29
DMA CONTROLLER
DREQ
Vorrichtung
28
BREQ
CPU
12
26

FIG.2

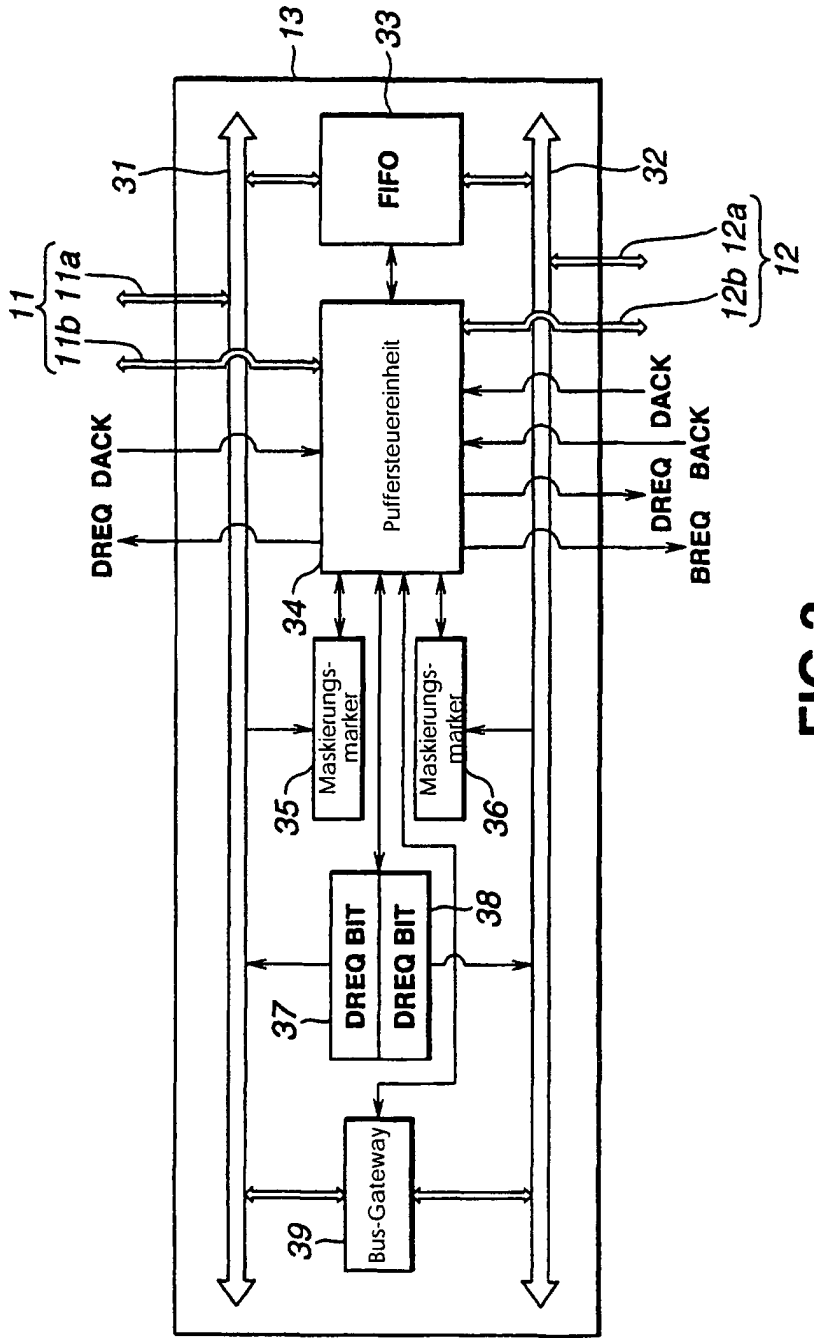

13
33
FIFO
31
32
11
11b 11a
12b 12a
12
Pufferspeichereinheit
DREQ DACK
DACK
DREQ BACK
BREQ
34
Maskierungs-marker
35
Maskierungs-marker
36
DREQ BIT
DREQ BIT
38
37
Bus-Gateway
39

FIG.3

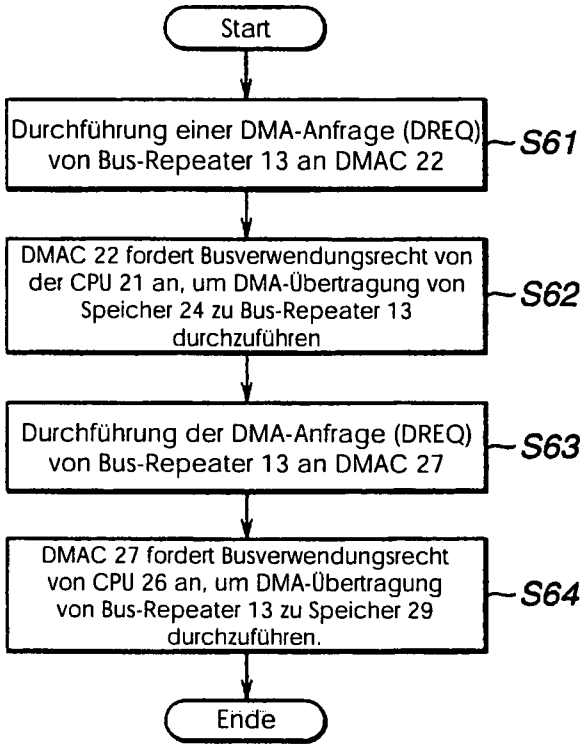

Start
Durchführung einer DMA-Anfrage (DREQ) von Bus-Repeater 13 an DMAC 22
S61
DMAC 22 fordert Busverwendungsrecht von der CPU 21 an, um DMA-Übertragung von Speicher 24 zu Bus-Repeater 13 durchzuführen
S62
Durchführung der DMA-Anfrage (DREQ) von Bus-Repeater 13 an DMAC 27
S63
DMAC 27 fordert Busverwendungsrecht von CPU 26 an, um DMA-Übertragung von Bus-Repeater 13 zu Speicher 29 durchzuführen.
S64
Ende

FIG.4

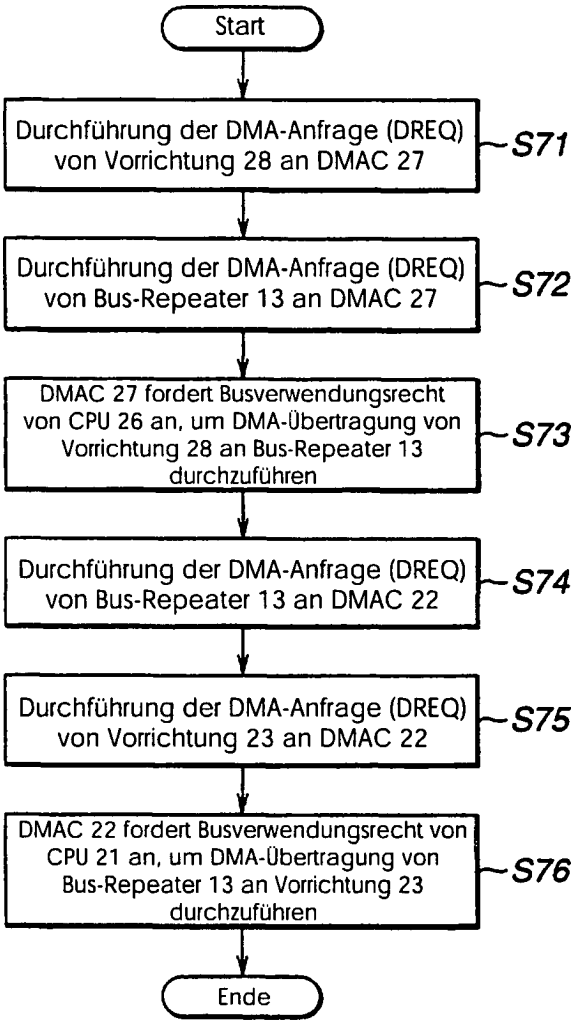

Start
Durchführung der DMA-Anfrage (DREQ) von Vorrichtung 28 an DMAC 27
S71
Durchführung der DMA-Anfrage (DREQ) von Bus-Repeater 13 an DMAC 27
S72
DMAC 27 fordert Busverwendungsrecht von CPU 26 an, um DMA-Übertragung von Vorrichtung 28 an Bus-Repeater 13 durchzuführen
S73
Durchführung der DMA-Anfrage (DREQ) von Bus-Repeater 13 an DMAC 22
S74
Durchführung der DMA-Anfrage (DREQ) von Vorrichtung 23 an DMAC 22
S75
DMAC 22 fordert Busverwendungsrecht von CPU 21 an, um DMA-Übertragung von Bus-Repeater 13 an Vorrichtung 23 durchzuführen
S76
Ende

FIG.5

Datenzug	S_{1A} (8BYTE)
Datenzug	S_{1B} (8BYTE)
Datenzug	S_{1C} (16BYTE)
Datenzug	S_{2A} (8BYTE)
Datenzug	S_{2B} (8BYTE)
Datenzug	S_{2C} (16BYTE)
Datenzug	S_{3A} (8BYTE)
Datenzug	S_{3B} (8BYTE)
Datenzug	S_{3C} (16BYTE)

FIG.6

Datenzug	S_{1B} (8BYTE)
Datenzug	S_{1C} (16BYTE)
Datenzug	S_{2B} (8BYTE)
Datenzug	S_{2C} (16BYTE)
Datenzug	S_{3B} (8BYTE)
Datenzug	S_{3C} (16BYTE)

FIG.7

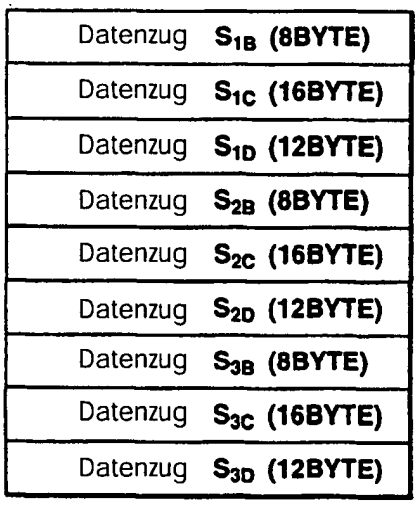

Datenzug S_{1B} (8BYTE)
Datenzug S_{1C} (16BYTE)
Datenzug S_{1D} (12BYTE)
Datenzug S_{2B} (8BYTE)
Datenzug S_{2C} (16BYTE)
Datenzug S_{2D} (12BYTE)
Datenzug S_{3B} (8BYTE)
Datenzug S_{3C} (16BYTE)
Datenzug S_{3D} (12BYTE)

FIG.8

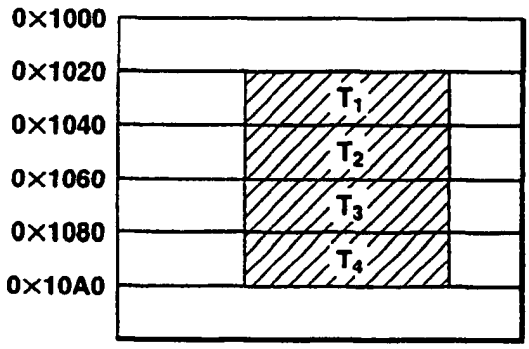

0×1000
0×1020
0×1040
0×1060
0×1080
0×10A0
T_1
T_2
T_3
T_4

FIG.9

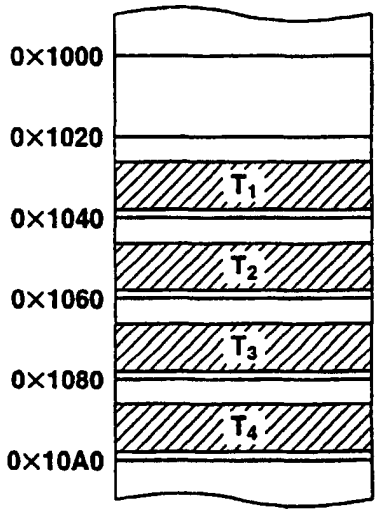

0×1000
0×1020
T1
0×1040
T2
0×1060
T3
0×1080
T4
0×10A0

FIG.10

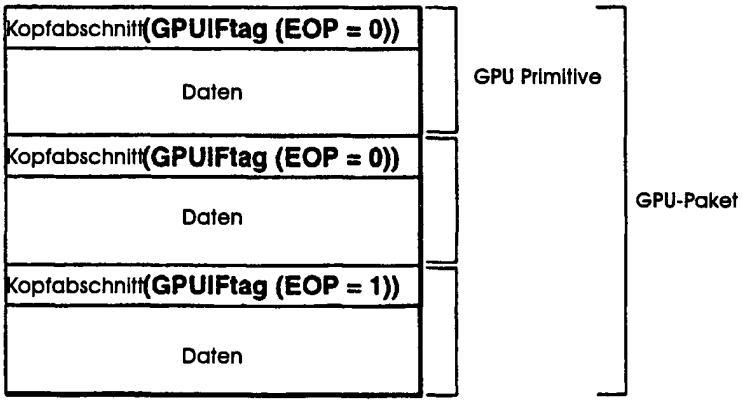

Kopfabschnitt(GPUIFtag (EOP = 0))
Daten
Kopfabschnitt(GPUIFtag (EOP = 0))
Daten
Kopfabschnitt(GPUIFtag (EOP = 1))
Daten
GPU Primitive
GPU-Paket

FIG.11

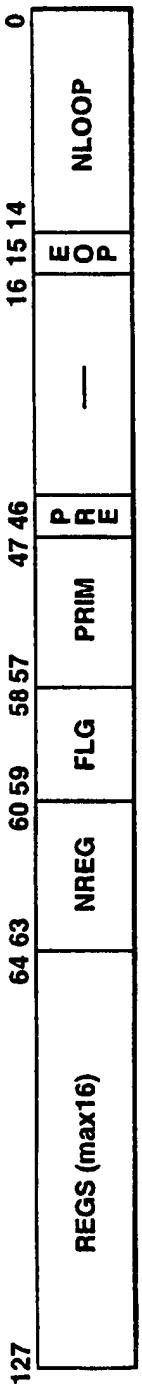

0
NLOOP
16 15 14
EOP
—
47 46
PRE
PRIM
58 57
FLG
60 59
NREG
64 63
REGS (max16)
127

FIG.12

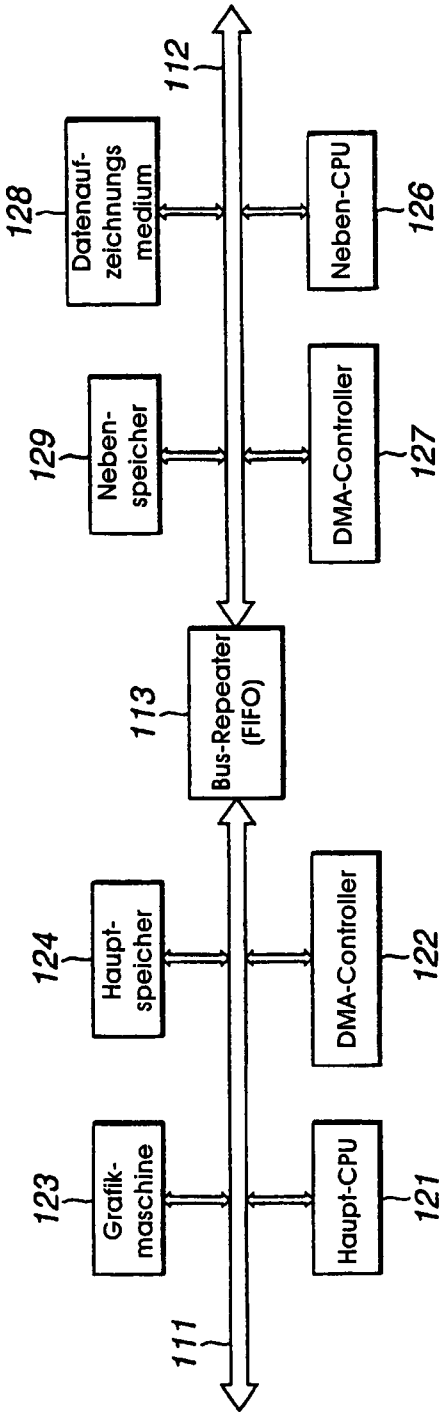

112
128
Datenauf-
zeichnungs
medium
Neben-CPU
126
129
Neben-
speicher
DMA-Controller
127
113
Bus-Repeater
(FIFO)
124
Haupt-
speicher
DMA-Controller
122
123
Grafik-
maschine
Haupt-CPU
121
111

FIG.13