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

(10) **DE 695 32 910 T2** 2004.09.02

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 720 138 B1**
(21) Deutsches Aktenzeichen: **695 32 910.3**
(96) Europäisches Aktenzeichen: **95 308 991.9**
(96) Europäischer Anmeldetag: **11.12.1995**
(97) Erstveröffentlichung durch das EPA: **03.07.1996**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **21.04.2004**
(47) Veröffentlichungstag im Patentblatt: **02.09.2004**

(51) Int Cl.[7]: **G09G 1/16**
G09G 5/36

(30) Unionspriorität:
364911 **27.12.1994** **US**

(73) Patentinhaber:
National Semiconductor Corp., Santa Clara, Calif., US

(74) Vertreter:
Dr. Weber, Dipl.-Phys. Seiffert, Dr. Lieke, 65183 Wiesbaden

(84) Benannte Vertragsstaaten:
CH, DE, ES, FR, GB, IE, IT, LI, NL

(72) Erfinder:
Perego, Richard E., Boulder, Colorado 80301, IS

(54) Bezeichnung: **Kompression der Videoauffrischungsdaten**

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

HINTERGRUND DER ERFINDUNG

1. Gebiet der Erfindung

[0001] Die vorliegende Erfindung bezieht sich allgemein auf Systeme und Verfahren einer Videoanzeige bzw. Videodarstellung und genauer gesagt auf Systeme und Verfahren zur Komprimierung von Pixeldaten in einem Computersystem.

2. Beschreibung des verwandten Standes der Technik

[0002] Ohne Beschränkung des Schutzumfangs der beanspruchten Erfindung wird diese Hintergrundinformation im Kontext eines speziellen Problems bereitgestellt, auf welches die Erfindung anwendbar ist.
[0003] Während der letzten Dekade ist die Qualität der Computergrafikanzeigen ständig verbessert worden mit Verbesserungen in der Pixelauflösung, der Farbtiefe und der Bilderneuerungsrate der Anzeigeeinrichtung – typischerweise eine Kathodenstrahlröhre (CRT) oder eine Flüssigkristallanzeige (LCD). Es ist üblich bei graphischen Darstellungen in Computern, daß man eine Bildauflösung von bis zu 1280 × 1024 Pixeln und bis zu 16,7 Millionen gleichzeitigen Farben verwendet. Eine Anzeige von Bildern mit derart hoher Auflösung und derart umfangreichem Farbinhalt, insbesondere bei hohen Erneuerungsraten, stellt große Anforderungen an das Speicherteilsystem, das den Einzelbildpuffer speichert bzw. darstellt. Typischerweise geht man Kompromisse ein, um geeignete Anzeigegeschwindigkeiten und Auflösungen zu erhalten, welche das Speicherteilsystem bereitstellen kann, während man noch immer genug Bandbreite hat, um Speicherzugriffe auszuführen, die durch die Grafikkarte bzw. Grafikmaschine oder die zentrale Verarbeitungseinheit (CPU) des Hosts erforderlich sind. Wenn die Datenrate der Anzeige zu groß ist, so wird das System durch andauernde Ablesungen von Pixeldaten aus dem Speicher gelähmt, da dies keine Zeit für andere Aufgaben läßt, um auf den Speicher zuzugreifen.
[0004] Zur Veranschaulichung dieses Aspekts hätte ein Computersystem, welches ein preiswertes Grafikteilsystem verwendet, beispielsweise ein Speicherarray mit 32 Bit breiten DRAMs, welche einen "schnellen Seitenzugriff" von 45 ns haben, eine theoretische verfügbare Spitzenbandbreite von 89 MByte/Sekunde. Realistisch muß dieser Wert jedoch unter anderem wegen fehlerhafter Seitenzugriffe korrigiert werden, was eine verfügbare Bandbreite von etwa 77 MByte/Sekunde ergibt. Mit einer Einzelbildauflösung von 1024 × 768 Pixeln und 8 Farbintensitätsbits pro Pixel sowie einer Erneuerungsrate von 75 Hz beträgt die erforderliche Anzeigebandbreite 59 MByte/Sekunde (1024 × 768 × 1 Byte × 75) = 77% der insgesamt verfügbaren Speicherbandbreite. Wenn die Auflösung der Farbintensität auf 16 Bits pro Pixel gesteigert wird, würde sich die Anforderung an die Anzeigebandbreite auf 118 MByte/Sekunde verdoppeln – 29 MByte/Sekunde mehr als die verfügbare maximale Bandbreite.
[0005] Ein Ansatz hinsichtlich des Umgangs mit diesen Einschränkungen besteht einfach darin, die Bandbreite des Speicherteilsystems zu erhöhen, indem spezielle Speicherports verwendet werden oder indem die Breite der DRAM-Schnittstelle vergrößert wird. Dementsprechend haben sich verschiedene Typen von speziellen integrierten Grafikspeicherschaltkreisen verbreitet, wie z. B. Dual-Port-VRAM oder der Windows™-RAM. Diese Speichertypen werden jedoch bei weitem nicht in so großen Mengen produziert wie die allseits verfügbaren DRAMs, die für den Hauptspeicher verwendet werden, und sie erfordern deshalb einen Preisaufschlag.
[0006] Bezüglich des weiteren Hintergrundes ist der Energieverbrauch ein anderes Hauptproblem bei der Auslegung von Teilsystemen einer Grafikanzeige, insbesondere bei tragbaren Computern aufgrund ihrer begrenzten Batteriestandzeit. Es ist bekannt, daß der Stromverbrauch sich proportional mit der verbrauchten Speicherbandbreite erhöht und daß dementsprechend Anzeigebetriebsarten mit hoher Auflösung und hohem Farbgehalt traditionell für Anwendungen bei tragbaren Computern nicht gut geeignet sind.
[0007] Die US-A-4,816,815 beschreibt eine Speichersteuerung eines Anzeigesystems mit zwei VRAMs zum Speichern von Anzeigedaten, einer Anzeigeeinheit (einer CRD) und einer CPU. Die CPU schreibt in dem Fall, daß ein Bild auf der CRD dargestellt wird, ein Einzelbild bzw. einen Frame aus Bilddaten in den ersten VRAM. Diese Daten werden auf der CRD dargestellt und gleichzeitig in dem zweiten VRAM gespeichert. In dem Fall, daß aktualisierte Anzeigedaten an die CRD geliefert werden, werden nicht-aktualisierte Daten angezeigt, indem sie aus dem zweiten VRAM ausgelesen werden.
[0008] Die EP-A-0 583 102 beschreibt eine Speichersteuerungsvorrichtung, bei welcher Frames (Einzelbilder) binärer Daten in einer Komprimierungseinheit komprimiert werden, bevor sie in einem Framespeicher gespeichert werden. Eine Erweiterungseinheit erweitert digitale Signale, die durch den Framespeicher ausgegeben werden.
[0009] Aus dem Vorstehenden erkennt man, daß ein Bedarf an einem System und einem Verfahren für eine Hochleistungsgrafikanzeige ohne erhöhten Energieverbrauch besteht.

Zusammenfassung der Erfindung

[0010] Um die oben beschriebenen Einschränkungen des Standes der Technik zu überwinden und um andere Einschränkungen zu überwinden, die beim Lesen und Verstehen der vorliegenden Beschreibung deutlich werden, wird ein Grafikanzeigesystem mit niedrigem Energieverbrauch und reduzierter Bandbreite sowie ein Verfahren für das Erzeugen von Pixeldaten unter Verwendung von Puffern für vollständige und komprimierte Frames (= Einzelbilder einer Bildschirmdarstellung) offenbart.

[0011] Gemäß einem ersten Aspekt der Erfindung wird ein Verfahren zum Erneuern einer Videoanzeige bereitgestellt, welches das Aktualisieren aus einer Quelle von Pixeldaten aufweist, das Anzeigen von Datenelementen, die in einem Vollframepufter (Puffer für vollständige, nicht komprimierte Frames bzw. Einzelbilder, im Folgenden „Vollbildpufter" genannt) gespeichert sind, das Ausgeben von Anzeigedatenelementen aus dem Vollbildpuffer für die Anzeigeerneuerung und das Speichern in einer Version eines Anzeigedatenelementes, welches aus dem Vollbildpuffer ausgegeben wird, in einem Komprimiertframepuffer (Puffer für komprimierte Frames bzw. Einzelbilder, im Folgenden „Komprimiertbildpuffer" genannt), und in Erneuerungszyklen, wenn ein Anzeigedatenelement in dem Vollbildpuffer noch nicht aktualisiert worden ist, Ausgeben eines Anzeigedatenelementes aus dem Komprimiertbildpuffer für die Anzeigeaktualisierung, wobei das Verfahren weiterhin aufweist: Setzen eines Verunreinigungsbits, welches zu einem Anzeigedatenelement gehört, jedesmal dann, wenn das Datenelement in dem Vollbildpuffer aktualisiert wird, Ausgeben eines Anzeigedatenelementes aus dem Vollbildpuffer für die Anzeigeaktualisierung, wenn das Verunreinigungsbit, welches zu diesem Anzeigedatenelement gehört, gesetzt ist, und Löschen des Verunreinigungsbits jedesmal dann, wenn das Anzeigedatenelement aus dem Vollbilddatenpuffer ausgegeben wird, und für ausgegebene Anzeigeelemente, die komprimierbar sind, Komprimieren und Speichern des komprimierten Anzeigedatenelementes in dem Komprimiertbildpuffer und Setzen eines Gültigkeitsbits, welches zu diesem Anzeigedatenelement gehört, und Ausgeben und Dekomprimieren eines Anzeigedatenelementes aus dem Komprimiertbildpuffer für die Anzeigeerneuerung, wenn und solange sein zugehöriges Verunreinigungsbit gelöscht ist und sein zugehöriges Gültigkeitsbit gesetzt ist.

[0012] Gemäß einem zweiten Aspekt der Erfindung wird ein System zum Erzeugen von Pixeldaten für das Aktualisieren einer Videoanzeige bereitgestellt, welches aufweist: eine Quelle von Pixeldaten, die mit einem Vollbildpuffer verbunden ist, um Anzeigedatenelemente für die Videoanzeige zu speichern, wobei jedes Anzeigedatenelement Pixeldaten aufweist, die von der Quelle empfangen wurden, ein Puffer für komprimierte Frames, der mit dem Puffer für volle Frames verbunden ist, der eine Version jedes Anzeigedatenelementes, welches von dem Puffer für volle Frames für eine Anzeigeerneuerung ausgegeben wird, empfängt und speichert, und eine Anzeigesteuerung, die mit dem Vollbildpuffer und dem Komprimiertbildpuffer verbunden ist, um Pixeldaten für ein Anzeigedatenelement für eine Anzeigeerneuerung aus dem Vollbildpuffer auszugeben, wenn dieses Anzeigedatenelement in dem Vollbildpuffer aktualisiert worden ist, oder aus dem Komprimiertbildpuffer auszugeben, wenn dieses Anzeigedatenelement noch nicht in dem Vollbildpuffer aktualisiert worden ist, wobei die Anzeigesteuerung derart betreibbar ist, daß sie ein Verunreinigungsbit setzt, das zu einem Anzeigedatenelement gehört, und zwar jedesmal dann, wenn dieses Datenelement in dem Vollbildpuffer aktualisiert wird, um ein Anzeigedatenelement aus dem Vollbildpuffer für eine Anzeigeerneuerung auszugeben, wenn das zu dem Anzeigedatenelement gehörige Verunreinigungsbit gesetzt ist, und um das Verunreinigungsbit jedesmal dann zu löschen, wenn das Anzeigedatenelement aus dem Vollbilddatenpuffer ausgegeben worden ist, ein Datenkomprimierer mit dem Vollbildpuffer und dem Komprimiertbildpuffer verbunden ist, wobei der Komprimierer derart betreibbar ist, daß er Anzeigeelemente, welche aus dem Vollbildpuffer ausgegeben werden und welche komprimierbar sind, komprimiert, um jedes komprimierte Anzeigedatenelement in dem Komprimiertbildpuffer zu speichern und um ein Gültigkeitsbit zu setzen, welches zu diesem Anzeigedatenelement gehört, und einen Datendekomprimierer, der mit dem Komprimiertbildpuffer verbunden und unter der Steuerung der Anzeigesteuerung betreibbar ist, um ein Anzeigedatenelement aus dem Komprimiertbildpuffer für eine Anzeigeerneuerung auszugeben und zu dekomprimieren, wenn und solange sein zugehöriges Verunreinigungsbit gelöscht ist und sein zugehöriges Gültigkeitsbit gesetzt ist.

[0013] In einer Ausführungsform werden Pixeldaten, wenn sie von dem Vollbildpuffer zu einer Anzeigeeinrichtung gesendet werden, gleichzeitig komprimiert und in dem Komprimiertbildpuffer aufgenommen, so daß nachfolgende unveränderte Einzelbilder direkt aus dem Komprimiertbildpuffer regeneriert werden. Eine Kohärenz zwischen den Voll- und Komprimiertbildpuffern wird aufrechterhalten mit einem Verunreinigungs-/Gültigkeits-Tag-RAM, so daß, wenn der Pixeldatenstrom übertragen und komprimiert wird, die komprimierten Daten für nachfolgende Frameaktualisierungen aus dem Komprimiertbildpuffer gültig gemacht werden.

[0014] Wenn der Pixeldatenstrom komprimiert, in dem Komprimiertbildpuffer gespeichert und gültig gemacht worden ist, so werden bei nachfolgenden Frames die Pixeldaten direkt aus dem Komprimiertbildpuffer wiedergewonnen und dekomprimiert, während sie zu der Anzeigeeinrichtung gesendet werden. Die Pixeldaten werden nach Bedarf kontinuierlich wiedergewonnen, um die Anzeige aus dem Komprimiertbildpuffer zu erneuern, bis die komprimierten Datenelemente durch künftige Schreibvorgänge in den Framepuffer ungültig gemacht

werden. Wenn neue Pixeldaten durch eine Grafikmaschine oder die Host-CPU in den Vollbildpuffer gegeben werden, werden die Verunreinigungs-Tags für entsprechende Datenelemente erneut gesetzt, so daß während der nächsten qualifizierten Frameabtastung die Pixeldaten von dem Vollbildpuffer anstatt von dem Komprimiertbildpufter wiedergewonnen werden.

[0015] Ein Merkmal der vorliegenden Erfindung besteht darin, Verunreinigungs- und Gültigkeitsbits zu trennen, um jedes Datenelement (wenn auch vorzugsweise nicht ausschließlich eine Rasterzeile) in einem Frame gültig zu machen, und in einem programmierbaren Steuermechanismus für die Frame- bzw. Einzelbildrate, um die Verunreinigungsbits zu setzen. Die Verunreinigungsbit werden in Reaktion darauf gesetzt, daß Pixeldaten in den Vollbildpuffer gegeben werden. Die Gültigkeitsbits werden gesetzt in Reaktion darauf, daß der Datenkomprimierer ein komprimiertes Datenelement in dem Komprimiertbildpuffer aktualisiert. Der programmierbare Steuermechanismus der Framerate stellt eine programmierbare Abtastrate bereit, um die Verunreinigungsbits auszubilden, so daß Aktualisierungen des Vollbildpuffers für eine vorbestimmte Zeitdauer ignoriert werden und weitere Frameanzeigen aus dem Komprimiertbildpuffer erfolgen, was demnach die Speicherbandbreite und den Energieverbrauch absenkt.

[0016] Ein weiteres Merkmal der vorliegenden Erfindung besteht in der Fähigkeit, einheitlichen Speicher in einem praktischen Grafiksystem zu verwenden, was eine einfache Möglichkeit für ein Aufrüsten entweder des Grafik- oder Hauptspeichers unter Hinzufügung weiterer DRAMs bereitstellt.

[0017] Diese und weitere Aufgaben, Merkmale und Vorteile der Neuheit, welche die Erfindung kennzeichnen, werden insbesondere in den anhängenden Patentansprüchen dargelegt und bilden einen Teil dieser Erfindung. Für ein besseres Verständnis der Erfindung, ihrer Vorteile und ihrer Aufgaben, die man durch ihre Verwendung erhält, wird Bezug auf die Figuren genommen, die einen weiteren Teil dieser Erfindung bilden, sowie auf die zugehörige Beschreibung, wo spezielle Beispiele von Systemen und Verfahren, die entsprechend der vorliegenden Erfindung ausgeführt werden, veranschaulicht und beschrieben sind.

Kurze Beschreibung der Figuren

[0018] **Fig.** 1 ist ein Blockdiagramm, welches ein Komprimierungssystem für die Videoerneuerung zeigt, welches gemäß den Prinzipien der vorliegenden Erfindung arbeitet, und

[0019] **Fig.** 2 ist ein Blockdiagramm, welches die Befehls- und Farbdatenpfade für das beispielhafte System nach **Fig.** 1 zeigt.

Beschreibung der bevorzugten Ausführungsform

[0020] In der folgenden Beschreibung der bevorzugten Ausführungsform wird Bezug genommen auf die beigefügten Zeichnungen, die einen Teil derselben bilden, und in welchen durch Veranschaulichung eine spezielle Ausführungsform dargestellt ist, durch welche die Erfindung ausgeführt werden kann. Es versteht sich, daß andere Ausführungsformen verwendet werden können und daß strukturelle Veränderungen vorgenommen werden können, ohne vom Schutzumfang dar beanspruchten Erfindung abzuweichen. Die genaue Beschreibung ist folgendermaßen aufgebaut:

1. Beispielhaftes Komprimierungssystem für die Erneuerung
2. Pfade der Komprimierungsbefehle und Farbdaten
3. Pfade der Dekomprimierungsbefehle und Farbdaten
4. Schlußfolgerungen

[0021] Diese Rahmenstruktur und die entsprechenden Überschriften werden in dieser genauen Beschreibung nur zur Erleichterung der Bezugnahme verwendet. Genaue Beschreibungen konventioneller oder bekannter Aspekte von Mikroprozessoren und graphischen Anzeigesystemen sind fortgelassen worden, um die Beschreibung der Erfindung nicht mit überflüssigen Einzelheiten zu verschleiern. Insbesondere ist eine gewisse Terminologie, die sich auf Standards der Cornputervideoanzeige und auf die Betriebsarten bezieht, für Praktiker auf dem Gebiet der Auslegung von Grafikanzeigen bekannt.

1. Beispielhaftes Erneuerungskomprimierungssystem

[0022] Gemäß **Fig.** 1 zeigt ein Blockdiagramm ein Komprimierungssystem für die Videoerneuerung, welches gemäß den Prinzipien der vorliegenden Erfindung ausgeführt ist. Eine Grafikmaschine **10** oder eine CPU (nicht dargestellt) aktualisiert ein Datenelement (vorzugsweise, wenn auch nicht ausschließlich, eine Rasterzeile), in einem Vollbildpuffer **12**, indem sie Pixeldaten hineinschreibt (eingibt) und in einem Verunreinigungs-/Gültigkeits-RAM **14** ein entsprechendes Verunreinigungsbit ("dirty bit") setzt, um anzuzeigen, daß die Rasterzeile aktualisiert worden ist. In der bevorzugten Ausführungsform hat der Vollbildpuffer **12** eine variable Größe, ist jedoch vorzugsweise groß genug, um eine Einzelbildauflösung von 1280 × 1024 Pixeln oder mehr aufzunehmen.

Mit Hilfe der vorliegenden Offenbarung werden Fachleute auf dem Gebiet erkennen, daß auch andere Bildauflösungen, Bildpuffergrößen, Anzahlen von in dem Bildpuffer gespeicherten Frames und Datenelementgrößen möglich sind, ohne vom Schutzumfang der beanspruchten Erfindung abzuweichen.

[0023] Der Verunreinigungs-/Gültigkeits-RAM **14** hält ein dirty bit und ein Gültigkeitsbit für jedes Datenelement (Rasterzeile), welches in dem Vollbildpuffer **12** gespeichert ist – was in der bevorzugten Ausführungsform 2048 (1024 × 2) Bits entspricht. Wenn das dirty bit in dem Verunreinigungs- /Gültigkeits-RAM **14** anzeigt, daß eine Rasterzeile aktualisiert worden ist (dirty bit gesetzt), so aktualisiert der Vollbildpuffer **12** in Reaktion auf die Anzeigesteuerschaltung **22** die Anzeigeeinrichtung (mit Ausnahme des nachstehend noch genauer Beschriebenen), indem er einen Strom von Pixeldaten, welche jeder Rasterzeile entsprechen, in einen Eingang eines Multiplexers **16** mit zwei Eingängen überträgt und das Gültigkeitsbit für dieses Datenelement löscht. Der Ausgang des Multiplexers **16** wird der Formatierungsstufe (nicht dargestellt) des Pixelausgangs zugeführt, wo ein Farbskalennachschlagen durchgeführt wird, falls erforderlich, irgendwelche Auflagen bzw. Abdeckungen eingefügt werden und eine Flachbild- (LCD-) Schnittstelle oder ein Digital-Analog-Wandler (DAC) (nicht dargestellt) einer Videofarbskala angesteuert wird, was wiederum eine CRT ansteuert (ebenfalls nicht dargestellt).

[0024] Der Strom von Pixeldaten aus dem Vollbildpuffer **12** wird auch mit einem Datenkomprimierer **18** verbunden, welcher in der bevorzugten Ausführungsform gleichzeitig die Pixeldaten komprimiert und in einem Komprimiertbildpuffer **20** speichert, wenn sie von dem Multiplexer **16** empfangen werden. Nachdem ein vollständiges Datenelement (Rasterzeile) komprimiert und in dem Komprimiertbildpuffer **20** gespeichert worden ist, erzeugt (setzt) der Datenkomprimierer **18** das entsprechende Gültigkeits-Tag in dem Verunreinigungs-/Gültigkeits-RAM **14**. Bei nachfolgenden Frameerneuerungen durch die Steuerschaltung **22** der Anzeige werden komprimierte Datenelemente, deren Gültigkeitsbits gesetzt und deren dirty bits nicht gesetzt oder ausgebildet sind, von dem Komprimiertbildpuffer **20** durch einen Datendekomprimierer **24** an den Multiplexer **16** zugeführt. Der Datendekomprimierer **24** dekomprimiert die Daten und führt sie durch den Multiplexer **16** für die Ausgabe zum Erneuern der Anzeigeeinrichtung zu. Der Vollbildpuffer **12**, der Komprimiertbildpuffer **20** und der Verunreinigungs-/Gültigkeits-RAM **14** können räumlich in demselben DRAM-Feld wie der Hauptspeicher angeordnet sein. Vorzugsweise liegt jedoch der Verunreinigungs-/Gültigkeits-RAM **14** in einem Zwischenregister-RAM, der von dem Hauptspeicher getrennt ist, da ein schnelles Erzeugen durch die Grafikmaschine **10** oder die CPU schnell große Blöcke von Daten "verunreinigen" kann.

[0025] Die dirty bits in dem Verunreinigungs-/Gültigkeits-RAM 14 müssen nicht mit der Bilderneuerungsrate von dem Datensteuerungsschaltkreis **22** abgetastet werden. Stattdessen kann eine langsamere Rate, die durch die Anzeigesteuerungsschaltung **22** gesetzt wird, Veränderungen in dem Zustand des dirty bits ausbilden, so daß Aktualisierungen, die in dem Vollbildpuffer **12** vorgenommen werden, für N Frames ignoriert werden. Da als flüssige Bewegung generell 30 Frames bzw. Einzelbilder pro Sekunde angesehen wird, gibt es kein Erfordernis, den angezeigten Frame irgendwie schneller zu aktualisieren und das Display kann von dem Komprimiertbildpuffer regeneriert werden. Darüber hinaus kann in dem Fall, wenn die Anzeigeeinrichtung eine noch langsamere Reaktionszeit hat, wie z. B. eine passive Flachbildschirm- (LCD-) Anzeige, die Frameaktualisierungsrate sogar noch geringer sein.

[0026] Beispielsweise ist es nicht erforderlich, die Anzeige irgendwie schneller als mit fünf Frames pro Sekunde zu aktualisieren bei einem Anzeigebildschirm, der eine Reaktionszeit von 200 ms hat. Wenn die Anzeigesteuerungsschaltung **22** eine Erneuerungsrate von 60 Hz liefert, kann die Ausbildungsfrequenz zwölfmal geringer sein, so daß die dirty bits nur bei jedem zwölften Frame ausgebildet werden (5 Hz), um eine Bildaktualisierungsrate sicherzustellen, die gleich fünf Einzelbildern pro Sekunde ist. Daher wird unter der Annahme, daß das gesamte Bild komprimierbar ist, mit einem Ausbildungsverhältnis von 12 : 1 die Anzeige in etwa während 92% der Zeit aus dem Komprimiertbildpuffer **20** aktualisiert, unabhängig davon, wie schnell neue Pixeldaten durch die Grafikmaschine **10** für den Vollbildpuffer **12** erzeugt werden.

2. Datenpfade der Farbkomprimierung und -steuerung

[0027] Es wird nun auf **Fig.** 2 Bezug genommen, welche die bevorzugten Farb- und Befehlsdatenpfade für ein System zeigt, welches gemäß den Prinzipien der vorliegenden Erfindung ausgebildet ist. Mit Hilfe der vorliegenden Offenbarung erkennen Fachleute weitere Formen und Anzahlen von Stufen für die Farb- und Befehlsdatenpfade, ohne vom Schutzumfang der beanspruchten Erfindung abzuweichen. Ein Anzeige-FIFO **30** ist über eine Speichersteuerung **31** mit einem DRAM-Array **11** verbunden, welches den Vollbildpuffer **12**, den Komprimiertbildpuffer **20** und optional auch den Hauptspeicher **21** enthält. Die Decodiersteuerungsschaltung **32** ist mit einem ersten Eingang mit dem Verunreinigungs-/Gültigkeits-RAM **14** und mit einem ersten Ausgang zum Steuern des Anzeige-FIFOs **30** verbunden, um Pixeldaten aus dem Vollbildpuffer **12** zu laden, wenn dirty bits ausgebildet und gesetzt oder Gültigkeitsbits nicht gesetzt sind. Alternativ lädt der Anzeige-FIFO **30** Pixeldaten aus dem Komprimiertbildpuffer **20**, wenn das Gültigkeitsbit gesetzt und das dirty bit nicht ausgebildet bzw. nicht aktiviert oder nicht gesetzt ist.

[0028] Die Decodiersteuerungsschaltung **32** ist mit einem zweiten Eingang mit dem Ausgang des Anzei-

ge-FIFOs **30** verbunden, um ein Steuerungswort zu erfassen und zu decodieren, welches in dem Komprimiertbildpuffer **20** (der im einzelnen nachstehend noch beschrieben wird) gespeichert ist, und ist mit einem zweiten Ausgang mit der Farbentpackungsschaltung **38**, der Befehlsentpackungsschaltung **40** und dem Multiplexer **16** verbunden. Der Multiplexer **16** leitet Pixeldaten von der Farbentpackungsschaltung **38**, wenn die Pixeldaten aus dem Vollbildpuffer **12** und aus dem Farbcache **32** stammen (oder aus der Farbentpackungsschaltung **38** im Falle des Ladens einer neuen Farbanweisung LNC), wenn die Pixeldaten aus dem Komprimiertbildpuffer **20** stammen. Der Ausgang des Multiplexers **16** ist mit der Formatierungsstufe (nicht dargestellt) des Pixelausgangs und mit einem Eingang auf der Farbverpackungsschaltung **58** verbunden. Farbdaten aus dem Multiplexer **16** werden durch die Farbverpackungsschaltung **58** verschachtelt in 32-Bit-Grenzen verpackt.

[0029] Die Befehlsverpackungsschaltung **16** empfängt und verschachtelt "Treffer-Operandencodes" ("hit opcodes") variabler Länge aus der Pipeline **50** der Trefferoperandencodes, dem RLE-Detektor **54** und dem RL8-Detektor **56** (die nachstehend alle noch genauer beschrieben werden) in 32-Bit-Grenzen bzw. -Paketen. Die Ausgangsgrößen der Farbverpackungsschaltung **58** und der Befehlsverpackungsschaltung **60** werden mit Eingängen auf dem Multiplexer **62** verbunden. Die Zeilenpuffersteuerschaltung **34** steuert den Multiplexer **62**, damit er einen komprimierten Zeilenpuffer **36** an seinen jeweils gegenüberliegenden Enden mit komprimierten Farb- und Befehlsdaten füllt, wobei er in Richtung der Mitte des Zeilenpuffers **36** fortschreitet. Wenn der Zeilenpuffer **36** zu dem Zeitpunkt, wenn das Ende der Rasterzeile erreicht ist, nicht überläuft, schreibt die Steuerschaltung **34** des Zeilenpuffers den Inhalt des komprimierten Zeilenpuffers **36** in den Komprimiertbildpuffer **20** und verschachtelt dabei die Farb- und Befehlsdaten innerhalb von 64-Bit-Grenzen ineinander. Ein Steuerwort für jedes Datenelement (Rasterzeile) wird durch die Steuerschaltung **34** des Zeilenpuffers berechnet und an den Anfang jedes Eintrags des komprimierten Zeilenpuffers **36** angehängt, um die Menge und Länge der Befehls- und Farbdaten festzulegen. Hinter dem Steuerungswort enthält jeder Eintrag in dem Komprimiertbildpuffer **20** Befehls- und Farbdaten abwechselnd innerhalb der 64-Bit-Grenzen, bis einer der Datenströme am Ende ist.

[0030] Auch wenn die Zeitbeziehung zwischen den Befehls- und Farbdaten aufgrund der ungleichmäßigen Pipelineanordnung zwischen den Befehls- und Farbdatenpfaden verloren geht, liefert die Verschachtelung der Farb- und Befehlsdaten in dem Komprimiertbildpuffer **20** die Daten in der näherungsweise erforderlichen Reihenfolge, wenn die Rasterzeile aus dem Komprimiertbildpuffer **20** bei künftigen Erneuerungen in ein Anzeige-FIFO **30** geladen wird. Wenn die Rasterzeile aus dem Komprimiertzeilenpuffer **36** erfolgreich in den Komprimiertbildpuffer **20** geschrieben worden ist, macht die Steuerungsschaltung **34** des Zeilenpuffers das entsprechende Gültigkeitsbit in dem Verunreinigungs-/Gültigkeits-RAM **14** für diese Rasterzeile gültig.

[0031] Ein Farbcache **42**, der vorzugsweise einen vollassoziativen Primärcache mit drei Eingängen, einen sekundären "Opfer"-Cache mit einem einzelnen Eingang und eine Mehrzahl von Komparatoren aufweist, empfängt Farbdaten aus dem Ausgang des Farbentpackungsschaltkreises **38**. Es versteht sich jedoch, daß mit Hilfe der vorliegenden Offenbarung Fachleute auf diesem Gebiet andere Cachekonfigurationen, Zuordnungen und Größen als realisierbar erkennen können, ohne vom Schutzumfang der Erfindung abzuweichen. Die Cachesteuerungsschaltung **44**, die mit dem Farbcache **42** verbunden ist, verfolgt und ersetzt die Einträge der letzten Verwendung (last recently used – LRU) in dem Primärcache, wenn eine neue Farbe der Farbentpackungsschaltung **38** gesendet bzw. übermittelt wird. Wenn die neue Farbe in den Sekundärcache gelangt, wird der Eintrag in dem Sekundärcache durch den LRU-Eintrag in dem Primärcache ausgetauscht. Wenn in dem Primärcache eine neue Farbe aktualisiert wird, wird die zuvor in dieser Position befindliche Farbe in den Sekundärcache verschoben.

[0032] Der Farbcache **42** signalisiert einen Treffer bzw. ein Eintreffen an die Cachestererungsschaltung **44**, wann immer Farbdaten aus der Farbentpackungsschaltung **38** mit Farbdaten in dem Farbcache **42** übereinstimmen. In Reaktion darauf codiert die Cachesteuerungsschaltung **44** und sendet einen "Treffer-Operationscode", welcher die Cacheposition des Treffers für den Multiplexer **48** identifiziert. Der Ausgang des Multiplexers **48** wird über die Pipeline **50** der Treffer-Operationscodes ausgesendet.

[0033] Zusätzlich zu den für die Treffer in dem Farbcache **42** verwendeten Befehlscodes werden die Laufdauer codierende (RLE) Befehlscodes verwendet, um eine Serie konstanter Farben zu komprimieren, die größer als **4** ist. Separate Befehlscodebefehle werden für kurze Serien (5–19) verwendet, und lange Serien (20–255) für eine Maximierung der Komprimierung. Konstante Farbfolgen, die weniger als **5** betragen, werden unter Verwendung eines Befehls für einen Cachewiederholungsbefehlscode codiert. Um Rasterzeilen, welche einen schwankenden Hintergrund enthalten, in effizienter Weise zu handhaben, wird ein Befehlscode der Wiederholung der letzten "N" (beispielsweise N = 8) (RL8) verwendet. Wenn eine Rasterzeile an die Formatierungsstufe des Pixelausgangs gesendet wird, werden die nächsten acht Pixel mit den vorangehenden acht Pixeln in derselben Reihenfolge übereinstimmen und vorausgesetzt, daß die Gruppe nicht vollständig aus derselben Farbe besteht, wird die Gruppe von acht Pixeln mit dem RL8-Befehlscode codiert.

[0034] Um eine Codierung wiederholter Befehlscodes zu vermeiden, wird eine "Pipeline der Befehlscodetreffer" **50** bereitgestellt, welche eine Mehrzahl von Stufen für das Anordnen von Befehlscodetreffern in einer Pipeline von dem Multiplexer **48** hat, so daß der RLE-Detektor **54** und der RL8-Detektor **56** RLE- bzw. RL8-Ketten

bestimmen können. Da ein Strom von Pixeldaten als eine Serie von Befehlscodetreffern in dem Farbcache **42** als ein RLE-Befehlscode oder möglicherweise als ein RL8-Opcode codiert werden kann, liefert die Pipeline **50** für Befehlscodetreffer ein Mittel für die Detektoren **54** und **56**, um besonders effizient mehrfache benachbarte Befehlscodetreffer zu vergleichen, zu zählen und in höchst effizienter Weise zu codieren. Die Anzahl der Stufen in den Pipelines **50** für Treffer beträgt vorzugsweise **8**. Fachleute auf diesem Gebiet erkennen jedoch, daß die Pipeline **50** kontrahiert oder erweitert werden kann, um an andere Befehlscodeketten angepaßt zu sein. Die Pipeline **50** für Befehlscodetreffer, der RLE-Detektor **54** und der RL8-Detektor **56** treiben die Befehlsverpackungsschaltung **60**, welche die entsprechenden Codes für die jeweilige Cacheposition oder Befehlscodeketten verpackt, wie oben beschrieben.

[0035] Wenn die aktuelle Pixelfarbe aus der Farbentpackungsschaltung **38** nicht mit einer der Farben in dem Farbcache **42** übereinstimmt, so codiert die Cachesteuerungsschaltung **44** einen Befehlscode für einen Befehl lade neue Farbe (load new colour – LNC) in den Pixeldatenstrom. Der LNC-Befehlscode erfordert 4 Bits zusätzlich zu den Pixeldatenbits, um den Farbwert selbst zu beschreiben und führt demnach zu einer Datenerweiterung anstatt zu einer Komprimierung. Die Datenerweiterung ist jedoch nicht signifikant, da der größte Teil des Bildschirms Wiederholungen enthält und sehr selten erfordert, daß eine neue Farbe geladen wird.

[0036] Die codierten Befehlscodes für die beispielhafte Ausführungsform sind nachstehend in Tabelle 1 zusammengefaßt.

Tabelle 1

BEFEHLS-CODE	NAME	BESCHREIBUNG	LÄNGE (Bits)	CODIERT (Bits/Pixel)	KOMPRIMIERUNGS-VERHÄLTNIS (8 Bits/Pixel)	KOMPRIMIERUNGS-VERHÄLTNIS (16 Bits/Pixel)
00	RC0	Wiederhole Cache 0	2	2	4:1	8:1
01	RC1	Wiederhole Cache 1	2	2	4:1	8:1
10	RC2	Wiederhole Cache 2	2	2	4:1	8:1
1100	RC3	Wiederhole Cache 3	4	4	2:1	4:1
1101	RL8	Wiederhole Letzte 8	4	0,5	16:1	32:1
1110	RLE4	RLE - 4-Bit count	8	0,42-1,6	5:1 bis 9:1	10:1 bis 38:1
1110 1111	RLE8	RLE - 8-Bit count	16	0,063-0,8	10:1 bis 128:1	20:1 bis 256:1
1111	LNC	Lade neue Farbe	12/20	12/20	0,67:1	0,8:1

3. Dekomprimierung der Farb- und Befehlsdatenpfade

[0037] Weiterhin erfaßt gemäß **Fig.** 2, wenn in dem Verunreinigungs-/Gültigkeits-RAM **14** für die ausgewählte Rasterzeile das Gültigkeitsbit gesetzt und das dirty bit nicht freigegeben oder nicht gesetzt ist, beim Erneuern die Decodiersteuerschaltung **32** das Steuerwort in dem Komprimiertbildpuffer **20** und decodiert es. Das Steuerwort kennzeichnet die Länge der Befehls- und Datenströme und instruiert dementsprechend die Decodiersteuerschaltung **32**, die Farbentpackungsschaltung **38** und die Befehlsentpackungsschaltung **40** so zu steuern, daß die Befehls- und Farbdaten aus dem Anzeige-FIFO **30** entpackt werden. Die Farbdaten aus der Farbentpackungsschaltung **38** werden in dem Farbcache **42** gespeichert, während die Befehlsdaten durch die Cachesteuerschaltung **44** decodiert werden. In Reaktion auf die Befehlsdaten wählt die Cachesteuerschaltung **44** einen von drei Eingängen zu dem Multiplexer **48** aus. Ein erster Eingang ist mit der Cachesteuerschaltung **44** verbunden, welche einen einzelnen Befehlscode ausgibt, der eine einzige Cacheposition oder einen LNC-Befehlscode ausgibt, um eine neue Farbe zu laden. Die zweiten und dritten Eingänge des Multiplexers **48** sind mit der Pipeline **50** für Befehlscodetreffer verbunden, welche sich wiederholende laufabschnittcodierte

(run-length encoded – RLE) und zur Wiederholung der letzten 8 (repeat last eight – RL8) vorgesehene Befehlscodes zurückkoppelt. Die erste Stufe der Pipeline **50** für Befehlscodetreffer (die dem Ausgang des Multiplexers **48**, verzögert um einen Taktzyklus, entspricht) wird zu der Cachesteuerschaltung **44** zurückgekoppelt. In Reaktion auf den Befehlscode, der durch die erste Stufe in der Pipeline **50** für Befehlscodetreffer erzeugt wird, weist die Cachesteuerschaltung **44** den Farbcache **42** an, Farbdaten zu senden oder neue Farbdaten aus der Farbentpackungsschaltung **38** in den Multiplexer **16** zu laden.

4. Zusammenfassung

[0038] Auch wenn die genaue Beschreibung der Erfindung auf bestimmte beispielhafte Ausführungsformen gerichtet worden ist, liegen verschiedene Modifikationen dieser Ausführungsformen ebenso wie alternative Ausführungsformen für Fachleute auf der Hand. Beispielsweise sind spezielle Registerstrukturen, Zuordnungen, Bitzuweisungen, Cacheassoziationen und Maße und andere Implementierungsdetails allein für den Zweck dargelegt worden, daß eine genaue Beschreibung der Erfindung bereitgestellt wird. Die Erfindung hat jedoch allgemeine Anwendungsmöglichkeiten auf irgendeine Computersystemarchitektur. Verschiedene Modifikationen, die auf Kompromissen zwischen Hardware- und Softwarelogik beruhen, liegen für Fachleute auf der Hand. Die Erfindung umfaßt jegliche Modifikationen oder alternative Ausführungsformen, die in den Schutzumfang der Ansprüche fallen.

Patentansprüche

1. Verfahren zum Erneuern einer Videoanzeige, welches das Aktualisieren einer Quelle (**10**) von Pixeldaten, das Anzeigen von Datenelementen, die in einem Puffer (**12**) für ein volles Einzelbild (Vollbildpuffer – full Frame buffer) gespeichert sind, Ausgeben von Anzeigedatenelementen aus dem Vollbildpuffer für die Anzeigeerneuerung und Speichern einer Version eines von dem Vollbildpuffer ausgegebenen Anzeigedatenelements in einem Puffer für ein komprimiertes Bild (Komprimiertbildpuffer), und bei Erneuerungszyklen, wenn ein Anzeigedatenelemente in dem Vollbildpuffer noch nicht aktualisiert worden ist, das Ausgeben eines Anzeigedatenelements aus dem Komprimiertbildpuffer für die Anzeigeerneuerung aufweist, wobei das Verfahren weiterhin aufweist:
Setzen eines Verunreinigungsbits (dirty bit), welches einem Anzeigedatenelement zugeordnet ist, jedesmal dann, wenn das Datenelement in dem Vollbildpuffer aktualisiert wird,
Ausgeben eines Anzeigedatenelements aus dem Vollbildpuffer für die Anzeigeerneuerung, wenn das Verunreinigungsbit, welches zu diesem Anzeigedatenelement gehört, gesetzt ist, und Löschen des Verunreinigungsbits jedesmal dann, wenn das Anzeigedatenelement aus dem Vollbildpuffer ausgegeben wird, und für ausgegebene Anzeigeelemente, die komprimierbar sind, Komprimieren und Speichern des komprimierten Anzeigedatenelements in dem Komprimiertbildpuffer (**20**) und Setzen eines Gültigkeitsbits, welches zu diesem Anzeigedatenelement gehört, und
Ausgeben und Dekomprimieren eines Anzeigedatenelements aus dem Komprimiertbildpuffer für die Anzeigeerneuerung, wenn und solange sein zugehöriges Verunreinigungsbit gelöscht bzw. nicht gesetzt ist und sein zugehöriges Gültigkeitsbit gesetzt ist.

2. Verfahren nach Anspruch 1, welches weiterhin das Löschen des Verunreinigungsbits und des zu einem Anzeigedatenelement gehörigen Gültigkeitsbits aufweist, wenn dieses Anzeigeelement aus dem Vollbildpuffer ausgegeben wird für eine Anzeigeerneuerung.

3. Verfahren nach Anspruch 1 oder 2, wobei das Verfahren weiterhin das Ausgeben eines Anzeigedatenelements für die Anzeigeerneuerung entweder aus dem Vollbildpuffer oder aus dem Komprimiertbildpufter aufweist, je nachdem, ob das zu dem Anzeigedatenelement gehörige Gültigkeitsbit gesetzt ist oder nicht.

4. Verfahren nach Anspruch 2, welches weiterhin die Modifikation aufweist, daß bei jedem N-ten Erneuerungszyklus, wobei N > 1 ist, ein Anzeigedatenelement aus dem Vollbildpufter ausgegeben wird für die Anzeigeerneuerung, falls das Verunreinigungsbit, welches zu diesem Anzeigedatenelement gehört, gesetzt ist, oder von dem Komprimiertbildpuffer ausgegeben und für die Anzeigeerneuerung dekomprimiert worden ist, falls sein zugehöriges Gültigkeitsbit gesetzt ist und sein zugehöriges Verunreinigungsbit frei ist, und in anderen Erneuerungszyklen der Zustand des. zugehörigen Verunreinigungsbits ignoriert und das Anzeigedatenelement aus dem Komprimiertbildpuffer ausgegeben und für eine Anzeigeerneuerung dekomprimiert wird.

5. Verfahren nach Anspruch 4, welches weiterhin das Abtasten der Verunreinigungsbits aufweist, die entsprechenden Anzeigedatenelementen zugeordnet sind, und zwar mit einer Geschwindigkeit, die geringer ist als die Anzeigeerneuerungsgeschwindigkeit.

6. System für das Erzeugen von Pixeldaten zum Erneuern einer Videoanzeige, welches aufweist:
eine Quelle (**10**) von Pixeldaten, die mit einem Vollbildpuffer (**12**) verbunden sind, um Anzeigedatenelemente für die Videoanzeige zu speichern, wobei jedes Anzeigedatenelement Pixeldaten aufweist, die von der Quelle empfangen wurden,
einen Komprimiertbildpuffer (**20**), der mit dem Vollbildpuffer so verbunden ist, daß er eine Version jedes Anzeigedatenelements, welches von dem Vollbildpuffer für eine Anzeigeerneuerung ausgegeben wird, empfängt und speichert und
eine Anzeigesteuerung (**22**), welche mit dem Vollbildpuffer und mit dem Komprimiertbildpuffer verbunden ist, um Pixeldaten für ein Anzeigedatenelement für die Anzeigeerneuerung aus dem Vollbildpuffer auszugeben, wenn das Anzeigedatenelement in dem Vollbildpuffer aktualisiert worden ist oder aus dem Komprimiertbildpuffer, wenn das Anzeigedatenelement in dem Vollbildpuffer nicht aktualisiert worden ist,
wobei die Anzeigesteuerung so betreibbar ist, daß sie ein Verunreinigungsbit setzt, welches zu einem Anzeigedatenelement gehört, und zwar jedesmal dann, wenn das Datenelement: in dem Vollbildpuffer aktualisiert wird, um ein Anzeigedatenelement aus dem Vollbildpuffer für die Anzeigeerneuerung auszugeben, wenn das Verunreinigungsbit, welches zu dem Anzeigedatenelement gehört, gesetzt ist, und um das Verunreinigungsbit jedesmal dann zu löschen, wenn das Anzeigedatenelement aus dem Vollbilddatenpuffer ausgegeben wird,
einen Datenkomprimierer (**18**), der mit dem Vollbildpuffer und dem Komprimiertbildpuffer verbunden ist, wobei der Komprimierer so betreibbar ist, daß er Anzeigedatenelemente, welche von dem Vollbildpuffer ausgegeben werden, und welche komprimierbar sind, komprimiert, um jedes komprimierte Anzeigedatenelement in dem Komprimiertbildpuffer zu speichern, und um ein Gültigkeitsbit zu setzen, welches diesem Anzeigedatenelement zugeordnet ist, und
einen Datenentkomprimierer (**24**), der mit dem Komprimiertbildpuffer verbunden ist und unter der Steuerung der Anzeigelogik betreibbar ist, um ein Anzeigedatenelement aus dem Komprimiertbildpuffer auszugeben und zu dekomprimieren für eine Anzeigeerneuerung, wenn und solange sein zugehöriges Verunreinigungsbit (dirty bit) gelöscht und sein zugehöriges Gültigkeitsbit gesetzt ist.

7. System nach Anspruch 6, wobei die Anzeigesteuerung in der Weise betreibbar ist, daß sie das Verunreinigungsbit und das Gültigkeitsbit, welche zu einem Anzeigedatenelement gehören, löscht, wenn dieses Anzeigeelement aus dem Vollbildpuffer für eine Anzeigeerneuerung ausgegeben wird.

8. System nach Anspruch 6 oder 7, wobei die Anzeigesteuerung weiterhin so betreibbar ist, daß sie ein Anzeigedatenelement für die Anzeigeerneuerung entweder von dem Vollbildpuffer oder dem Komprimiertbildpuffer ausgibt, und zwar je nachdem, ob das zu dem Datenelemente gehörige Gültigkeitsbit gesetzt wurde oder nicht.

9. System nach Anspruch 7, wobei die Anzeigesteuerung so betreibbar ist, daß sie auf den Zustand des Verunreinigungsbits eines Anzeigedatenelements nur in jedem N-ten (N > 1) Erneuerungszyklus reagiert, indem in jedem N-ten Erneuerungszyklus ein Anzeigedatenelement aus dem Vollbildpuffer für die Anzeigeerneuerung ausgegeben wird, wenn das Verunreinigungsbit, welches zu diesem Anzeigedatenelement gehört, gesetzt ist, oder aus dem Komprimiertbildpuffer ausgegeben und für die Anzeigeerneuerung dekomprimiert wird, wenn sein zugehöriges Gültigkeitsbit gesetzt ist und sein zugehöriges Verunreinigungsbit gelöscht ist, und daß während anderer Erneuerungszyklen der Zustand des zugehörigen Verunreinigungsbits ignoriert wird und die Anzeigeerneuerungssteuerung so betreibbar ist, daß sie ein Anzeigedatenelement aus dem Komprimiertbildpufter ausgibt für eine Dekomprimierung und eine Anzeigeerneuerung unabhängig von dem Zustand des Verunreinigungsbits, welches zu diesem Anzeigedatenelement gehört.

10. System nach Anspruch 9, welches weiterhin eine komprimierbare Steuereinrichtung für die Bildrate aufweist, die mit der Anzeigesteuerung für das Abtasten des Zustands der Verunreinigungsbits verbunden ist, welche Anzeigedatenelementen zugeordnet sind, und zwar bei jedem N-ten Erneuerungszyklus.

11. Videoerneuerungskomprimierung nach einem der Ansprüche 6 bis 10, wobei der Vollbildpufferausgang außerdem mit einem ersten Eingang eines Multiplexers (**16**) verbunden ist und der Datenkomprimiererausgang mit einem zweiten Eingang des Multiplexers verbunden ist, und daß die Anzeigesteuerung so betreibbar ist, daß sie mit den Multiplexerausgängen ein Anzeigedatenelement aus dem Vollbildpuffer oder ein dekomprimiertes Anzeigedatenelement aus dem Komprimiertbildpuffer verbindet.

12. System nach einem der Ansprüche 6 bis 11, wobei der Vollbildpuffer und der Komprimiertbildpuffer in einem einheitlichen Speicher (**11**), beispielsweise in dem Systemhauptspeicher, residieren.

13. System nach einem der Ansprüche 6 bis 12, wobei die Pixeldatenquelle eine Grafikmaschine oder ein

zentraler Prozessor ist.

14. System nach Anspruch 13, wobei die Verunreinigungsbits und die Gültigkeitsbits in einem Speicher (**14**) getrennt von dem Hauptspeicher gespeichert sind.

Es folgen 2 Blatt Zeichnungen

Anhängende Zeichnungen

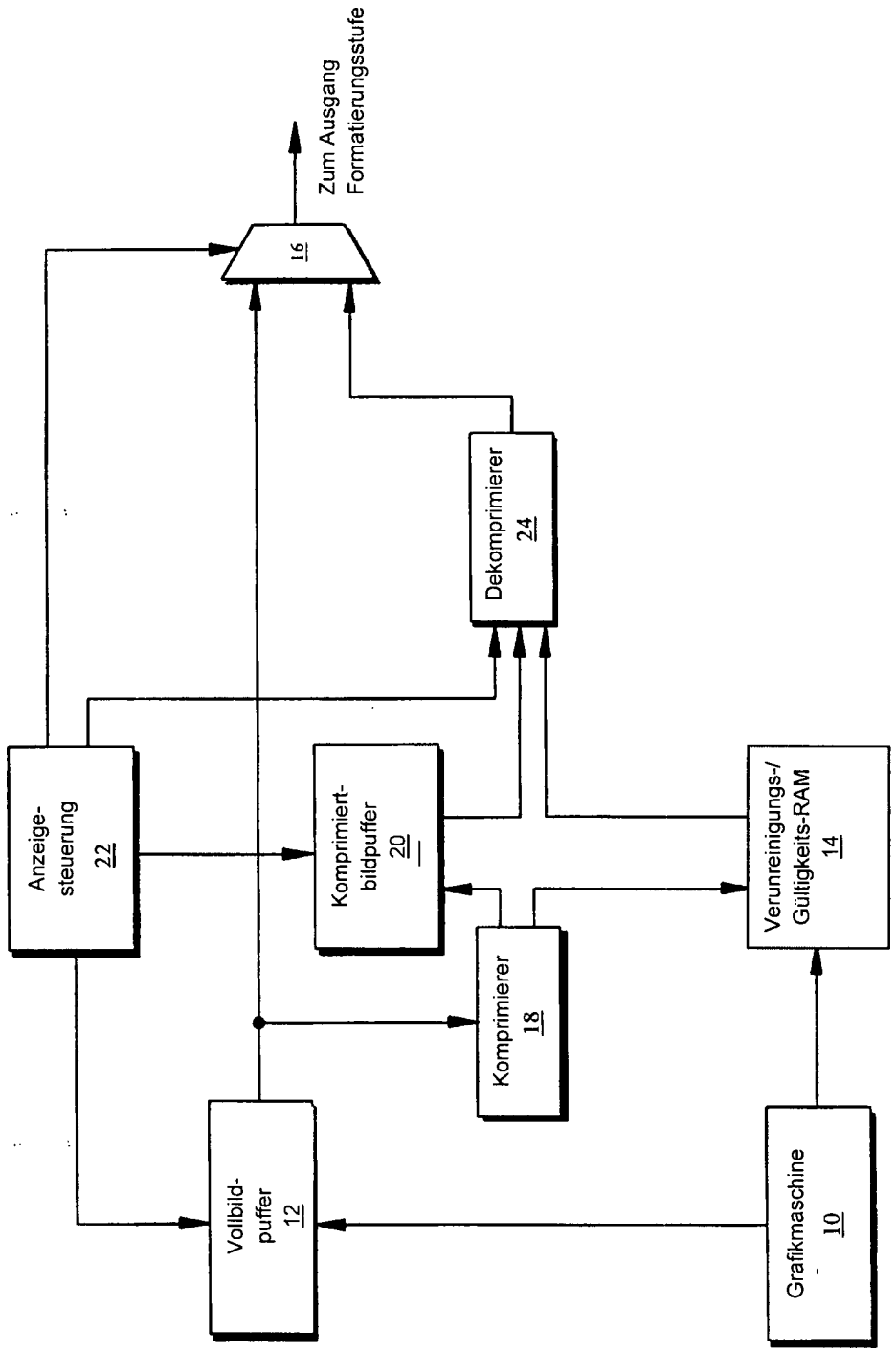


FIG. 1

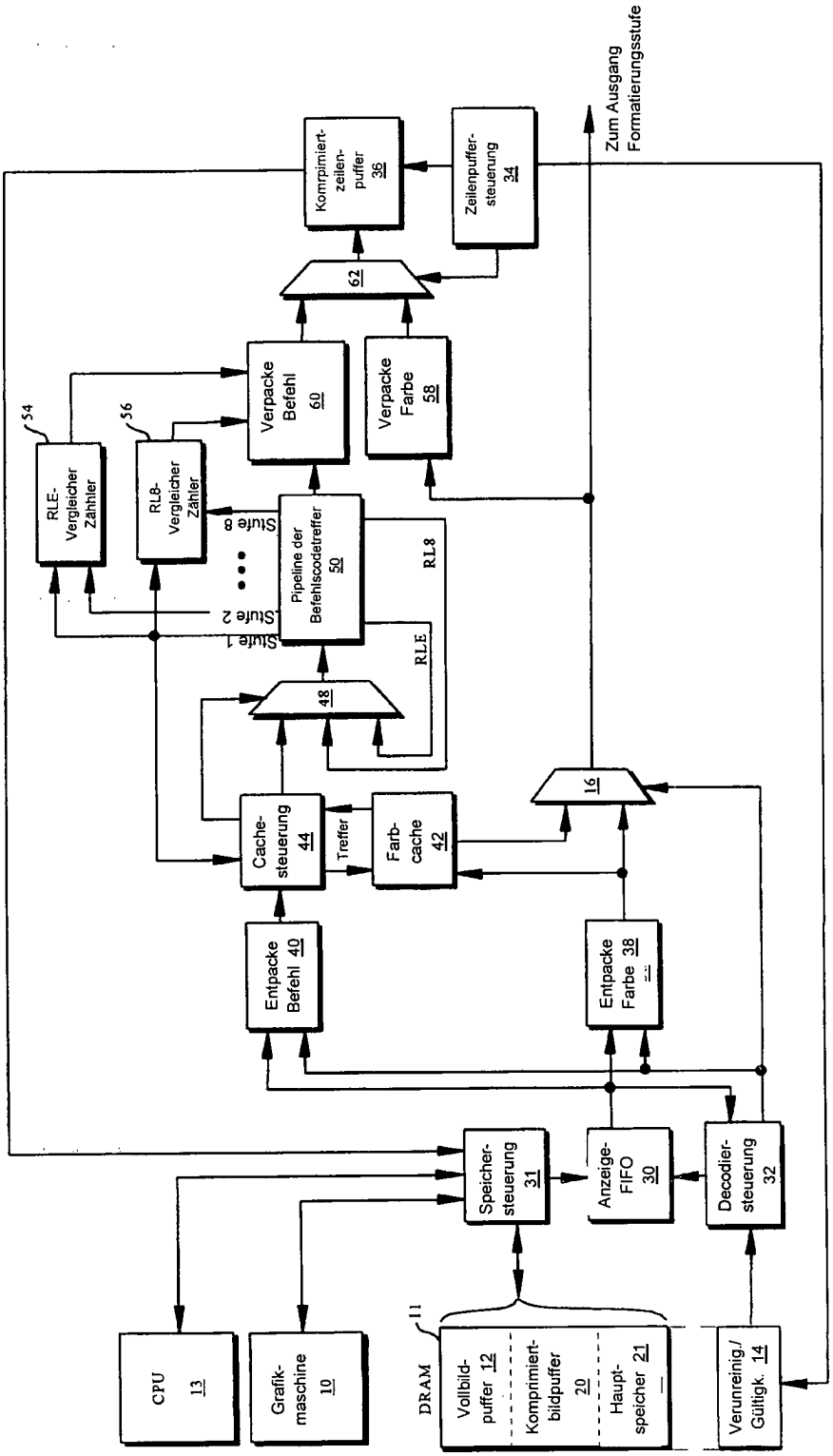

Zum Ausgang Formatierungsstufe
Komprimiert-zeilen-puffer 36
Zeilenpuffer-steuerung 34
62
Verpacke Befehl 60
Verpacke Farbe 58
54
56
RLE-Vergleicher Zähler
RL8-Vergleicher Zähler
Stufe 8
Stufe 2
Stufe 1
Pipeline der Befehlscodetreffer 50
RL8
RLE
48
16
Cache-steuerung 44
Treffer
Farb-cache 42
Entpacke Befehl 40
Entpacke Farbe 38
Speicher-steuerung 31
Anzeige-FIFO 30
Decodier-steuerung 32
CPU 13
Grafik-maschine 10
11
DRAM
Vollbild-puffer 12
Komprimiert-bildpuffer 20
Haupt-speicher 21
Verunreinig./ Gültigk. 14

FIG. 2