



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) DE 600 20 046 T3 2010.01.14

(12) Übersetzung der geänderten europäischen Patentschrift

(97) EP 1 092 193 B2

(21) Deutsches Aktenzeichen: 600 20 046.9

(86) PCT-Aktenzeichen: PCT/US00/07087

(96) Europäisches Aktenzeichen: 00 916 466.6

(87) PCT-Veröffentlichungs-Nr.: WO 2000/060476

(86) PCT-Anmeldetag: 20.03.2000

(87) Veröffentlichungstag

der PCT-Anmeldung: 12.10.2000

(97) Erstveröffentlichung durch das EPA: 18.04.2001

(97) Veröffentlichungstag
der Patenterteilung beim EPA: 11.05.2005

(97) Veröffentlichungstag
des geänderten Patents beim EPA: 16.09.2009

(47) Veröffentlichungstag im Patentblatt: 14.01.2010

(51) Int Cl.⁸: G06F 13/36 (2006.01)
G06F 13/38 (2006.01)

Patentschrift wurde im Einspruchsverfahren geändert

(30) Unionspriorität:

285706 05.04.1999 US

(73) Patentinhaber:

SanDisk IL Ltd., Kfar Saba, IL

(74) Vertreter:

Patent- und Rechtsanwälte Kraus & Weisert,
80539 München

(84) Benannte Vertragsstaaten:

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

(72) Erfinder:

BAN, Amir, 47205 Ramat Hasharon, IL; MORAN,
Dov, 44406 Kfar Saba, IL; OGDAN, Oron, 93106
Jerusalem, IL

(54) Bezeichnung: Architektur einer USB-basierten PC-Flashspeicherkarte

Beschreibung

[0001] Die vorliegende Erfindung betrifft Halbleiter-Speichervorrichtungen und insbesondere löscharbe und programmierbare nichtflüchtige Speichermodule, die unter Verwendung eines USB-PC-Buses mit einer Hostrechner-Plattform verbunden sind.

[0002] Löscharbe und programmierbare nichtflüchtige Speichermodule, im Folgenden als Flash-Speicher oder Flash-Vorrichtungen bezeichnet, sind im Stand der Technik zur Speicherung von Information bekannt. Flash-Vorrichtungen umfassen elektrisch löscharbe und programmierbare Nur-Lese-Speicher (EEPROMS – electrically erasable programmable read-only memories) aus Transistoren des Flash-Typs mit schwabendem Gate (floating gate) und sind nichtflüchtige Speicher, die in der Funktionalität und Leistung ähnlich zu EPROM-Speichern sind, mit einer zusätzlichen Funktionalität, die einer systeminternen programmierbaren Operation ermöglicht, Seiten des Speichers zu löschen. Ein Beispiel einer Implementierung einer derartigen Flash-Vorrichtung wird in dem U.S.-Patent Nr. 5,799,168 gezeigt, das durch Bezugnahme aufgenommen ist, als ob es hier vollständig offenbart wird.

[0003] Flash-Vorrichtungen haben den Vorteil, dass sie im Vergleich zu herkömmlichen magnetischen Speicherplatten relativ kostengünstig sind und relativ wenig Leistung erfordern. In einer Flash-Vorrichtung ist es jedoch nicht möglich, einen vorher beschriebenen Bereich des Speichers ohne eine vorhergehende Seitenlöschung des Bereichs neu zu schreiben. Diese Einschränkung von Flash-Vorrichtungen verursacht, dass sie nicht kompatibel sind zu typischen existierenden Betriebssystemprogrammen, da Daten nicht in einen Bereich des Speichers in der Flash-Vorrichtung geschrieben werden können, in den vorher Daten geschrieben wurden, außer der Bereich wird zuerst gelöscht. Von einem Software-Verwaltungssystem, wie das in dem am 5. März 1993 angemeldeten U.S.-Patent Nr. 5,404,485 offenbart, ist erforderlich, dass es diese Funktionen der Flash-Speichervorrichtung verwaltet.

[0004] Im Moment haben diese Flash-Speichervorrichtungen eine zweite Einschränkung, die darin liegt, dass sie entweder statisch an der Hostrechner-Plattform angefügt werden müssen oder dynamisch angefügt und entfernt werden müssen unter Verwendung der PCMCIA[Personal Computer Memory Card International Association]-Schnittstelle. Beide Implementierungen haben Nachteile, einschließlich einer schwierigen Verwendung und hoher Kosten.

[0005] Eine nützlichere Implementierung würde dem USB-Standard folgen, wie in der USB-Spezifikation Version 1.1 beschrieben. Der USB-Standard bietet einen kleineren Form-Faktor und eine einfache

Verwendung für den Endbenutzer, während die Kosten der Implementierung gesenkt werden. Dieser Standard wird als ein Industrieweiher Standard spezifiziert, der von Firmen, wie Compaq Computer Corporation, Microsoft, IBM und Intel, unterstützt wird, um als eine Erweiterung der PC-Architektur mit einem Fokus auf CTI (computer telephony integration), dem Konsumenten und Produktivitätsanwendungen zu dienen.

[0006] Die Kriterien, die angelegt wurden, um die Architektur für den USB-Standard zu definieren, umfassen die Einfachheit einer peripheren PC(Personal Computer)-Erweiterung, geringe Kosten, Unterstützung von Übertragungsraten bis zu 12 Mb/Sekunde und volle Unterstützung für Daten, Sprache, Audio und komprimiertes Video in Echtzeit. Dieser Standard bietet auch eine Protokoll-Flexibilität für isochrone Datenübertragungen und asynchrones Messaging im Mischmodus, eine Integration in eine Gebräuchsvorrichtungstechnologie und eine Bereitstellung einer Standardschnittstelle für eine schnelle Integration in ein beliebiges Hostrechner-Produkt. Zusätzlich stellt der USB-Standard ein einzelnes Modell zum Verkabeln und Befestigen von Verbindern dar, dass alle Details der elektrischen Funktionen, einschließlich Bus-Abschlüsse, von dem Endbenutzer isoliert sind. Durch den Standard sind die peripheren Vorrichtungen selbstidentifizierend und unterstützen eine automatische Abbildung von Funktionen zu einem Treiber. Ferner ermöglicht der Standard, dass alle peripheren Vorrichtungen dynamisch hinzufügbar und neu konfigurierbar sind.

[0007] Ein gemäß dem USB-Standard konstruiertes System wird durch drei getrennte definierte Bereiche beschrieben: USB-Verbindung, USB-Vorrichtungen und die USB-Hostrechner-Plattform. Die USB-Verbindung ist die Art, wie die USB-Vorrichtungen mit der Hostrechner-Plattform verbunden sind und mit ihr kommunizieren. Die zugehörigen Funktionen und Komponenten umfassen die Bus-Topologie, die das Verbindungsmodell zwischen USB-Vorrichtungen und der Hostrechner-Plattform ist.

[0008] Die physikalische USB-Verbindung weist eine gestaffelte sternförmige Topologie auf. Ein Hub (Verteiler) befindet sich im Zentrum jedes Sterns. Jedes Leistungssegment ist eine Punkt-zu-Punkt-Verbindung zwischen der Hostrechner-Plattform und einem Hub oder einer Funktion oder ein mit einem anderen Hub oder Funktion verbundener Hub.

[0009] Hinsichtlich eines Stacks der Fähigkeiten umfassen die USB-Aufgaben (tasks), die an jeder Schicht in dem System ausgeführt werden, ein Datenflussmodell und einen Verarbeitungsplan (schedule). Ein Datenflussmodell ist die Art, wie Daten sich in dem System über den USB zwischen Datenerzeugern und Datenkonsumenten bewegen. Ein Verarbei-

tungsplan legt einen Zugriff zu der Verbindung fest, die geteilt wird. Ein derartiger Verarbeitungsplan ermöglicht, dass isochrone Datenübertragungen unterstützt werden und entfernt einen Arbitrations-Overhead.

[0010] Der USB selbst ist ein Bus mit zyklischer Abfrage (polled). Die Hostrechner-Steuereinrichtung auf der Hostrechner-Plattform initiiert alle Datenübertragungen. Alle Bus-Transaktionen umfassen die Übertragung von bis zu drei Paketen. Jede Transaktion beginnt, wenn die Hostrechner-Steuereinrichtung auf einer durch den Verarbeitungsplan festgelegten Basis ein USB-Paket sendet, das den Typ und die Richtung der Transaktion, die USB-Vorrichtungsadresse und die Endpunktnummer beschreibt. Dieses Paket wird als das „Tokenpaket“ bezeichnet. Die USB-Vorrichtung, an die das Paket adressiert ist, wählt sich selbst durch Decodieren der entsprechenden Adressenfelder aus. In einer Transaktion werden Daten entweder von der Hostrechner-Plattform an eine Vorrichtung oder von einer Vorrichtung an die Hostrechner-Plattform übertragen. Die Richtung einer Datenübertragung wird in dem Tokenpaket spezifiziert. Die Quelle der Transaktion sendet dann ein Datenpaket oder zeigt an, dass die Quelle keine Daten zur Übertragung aufweist. Das Ziel antwortet im Allgemeinen mit einem Handshake-Paket, das angeigt, ob die Übertragung erfolgreich war.

[0011] Das USB-Datenübertragungsmodell zwischen einer Quelle und einem Ziel auf der Hostrechner-Plattform und einem Endpunkt auf einer Vorrichtung wird als „Pipe“ (Röhre) bezeichnet. Es gibt zwei Typen von Pipes: Strom und Meldung. Stromdaten haben keine USB-definierte Struktur, Meldungsdaten hingegen schon. Zusätzlich haben Pipes Zuordnungen von Datenbandbreite, Übertragungsdiensttyp und Endpunkt-Charakteristiken, wie Richtung und Puffergrößen. Die meisten Pipes entstehen, wenn eine USB-Vorrichtung konfiguriert wird. Eine Meldungs-Pipe, die Standardsteuerungs-Pipe, existiert immer, sobald eine Vorrichtung betrieben wird, um einen Zugriff auf die Konfiguration, einen Status und eine Steuerungsinformation für die Vorrichtung bereitzustellen.

[0012] Der Transaktions-Verarbeitungsplan für den USB-Standard ermöglicht eine Flusssteuerung für einige Strom-Pipes. Auf dem Level der Hardware verhindert dies Situationen, in denen Puffer eine Untersorgung (underrun) oder Überlastung (overrun) erfahren, durch Verwendung eines NAK-Handshakes, um die Datenrate zu drosseln. Mit dem NAK-Handshake wird eine Transaktion neu versucht, wenn Bus-Zeit verfügbar ist. Der Flusssteuerungsmechanismus ermöglicht die Konstruktion von flexiblen Verarbeitungsplänen, die eine gleichzeitige Versorgung einer heterogenen Mischung von Strom-Pipes ermöglicht. Somit können mehrere Strom-Pipes

zu unterschiedlichen Intervallen mit Paketen unterschiedlicher Größen versorgt werden.

[0013] Der USB-Standard weist, wie beschrieben, drei Haupttypen von Pakten auf, einschließlich Tokenpakte, Datenpakte und Handshake-Pakete. Ein Beispiel jedes Typs von Paket wird in den [Fig. 1](#) bis [Fig. 3](#) für den Stand der Technik gezeigt. Die [Fig. 4](#) für den Stand der Technik zeigt eine beispielhafte abstrakte USB-Vorrichtung.

[0014] Ein Tokenpaket **10**, wie in der [Fig. 1](#) gemäß dem Stand der Technik gezeigt, weist ein PID(Paketidentifizierungs)-Feld **12** auf, das einen von drei Pakettypen spezifiziert: HINEIN (IN), HINAUS (OUT) oder AUFBAU (SETUP). Wenn das PID-Feld **12** den HINEIN-Pakettyp spezifiziert, wird die Datentransaktion von einer Funktion an die Hostrechner-Plattform definiert. Wenn das PID-Feld **12** den HINAUS- oder AUFBAU-Pakettyp spezifiziert, wird die Datentransaktion von der Hostrechner-Plattform an eine Funktion definiert.

[0015] Ein Adressenfeld **14** (ADDR) spezifiziert die Adresse, während ein Endpunktfeld **16** (ENDP) den Endpunkt für das Tokenpaket **10** spezifiziert. Für HINAUS- und AUFBAU-Transaktionen, in denen das PID-Feld **12** spezifiziert, dass das Tokenpaket ein HINAUS-Pakettyp oder ein AUFBAU-Pakettyp ist, identifizieren das Adressenfeld **14** und das Endpunktfeld **16** eindeutig den Endpunkt zum Empfang des nachfolgenden in [Fig. 2](#) gezeigten Datenpakets, das auf das Tokenpaket **10** folgt. Für HINEIN-Transaktionen, in denen das PID-Feld **12** spezifiziert, dass das Tokenpaket **10** ein HINEIN-Pakettyp ist, identifizieren das Adressenfeld **14** und das Endpunktfeld **16** eindeutig, welche Endpunkte ein Datenpaket übertragen. Ein CRC5-Feld **18** umfasst die Prüfsumme zum Feststellen, ob das Tokenpaket **10** ohne Korruption empfangen wurde. Nur die Hostrechner-Plattform kann Tokenpakte **10** ausgeben, da Tokenpakte **10** eine Steuerung der Übertragung der nachfolgenden Datenpakete liefern.

[0016] Wie in [Fig. 2](#) gemäß dem Stand der Technik gezeigt wird, weist ein USB-Datenpaket **20** gemäß dem Stand der Technik ebenfalls ein PID(Paketidentifizierungs)-Feld **22** zum Identifizieren des Typs des Datenpaketes auf. Das Datenpaket **20** weist ebenso ein Datenfeld **24**, das optional Daten umfasst, und ein CRC-Feld **26** auf, das die Prüfsumme wie oben beschrieben umfasst.

[0017] Die [Fig. 3](#) gemäß dem Stand der Technik zeigt ein USB-Handshake-Paket **28** gemäß dem Stand der Technik, das nur ein PID(Paketidentifizierungs)-Feld **30** aufweist. Die Handshake-Pakete **28** werden verwendet, um den Status einer Datentransaktion zu melden und können Werte zurücksenden, die einen erfolgreichen Empfang von Daten, eine Be-

fehlannahme oder -rückweisung, eine Flusssteuerung und Stopp-Bedingungen anzeigen. Nur Transaktionstypen, die eine Flusssteuerung unterstützen, können Handshake-Pakete **28** zurücksenden. Handshake-Pakete **28** werden immer in der Handshake-Phase einer Transaktion zurückgesendet und können statt Datenpaketen **20** in der Datenphase einer Transaktion zurückgesendet werden.

[0018] Diese drei unterschiedlichen Typen von Paketen werden während verschiedener Phasen der Transaktion, die eine USB-Vorrichtung umfasst, ausgetauscht. Eine schematische Blockdarstellung der funktionalen Blöcke in einer typischen USB-Vorrichtung **32** wird in [Fig. 4](#) für eine abstrakte USB-Vorrichtung gemäß dem Stand der Technik gezeigt. Die USB-Vorrichtung **32** umfasst typischerweise eine elektrische USB-Schnittstelle **34**, die ein Kabel und einen Verbinder aufweist und die eine physikalische Schnittstelle zum Empfang und zur Übertragung von elektrischen Signalen ist, die kompatibel zu der USB-Spezifikation, wie oben beschrieben, sind. Die Signale werden dann an eine logische Schnittstelle **36** weitergeleitet, die einen oder mehrere Puffer, den Vorrichtungsadresse-Decodierer zum Decodieren der Adresse der Quellenvorrichtung für die Signale und einen SYNC-Feld-Synchronisierer zum Synchronisieren der Signale umfasst. Eine Information und Strukturen, die für eine Verwaltung der abstrakten USB-Vorrichtung **32** als eine USB-Vorrichtung erforderlich sind, sind in einer USB-Klassensteuerungs- und Auflistungs-Maschine (engine) **38** gespeichert. Eine Funktions- und Vorrichtungs-Maschine (engine) **40**, auch als die „Anwendung“ bezeichnet, steuert und verwaltet die spezifischen Funktionen und Eigenschaften der abstrakten USB-Vorrichtung **32**. Zusätzlich konsumiert und erzeugt die Funktions- und Vorrichtungs-Maschine **40** die meisten der Daten über den USB-Bus.

[0019] Die USB-Spezifikation definiert jedoch nicht das Verhältnis zwischen unterschiedlichen Entitäten in der abstrakten USB-Vorrichtung **32**. Vielmehr beschreibt die USB-Spezifikation nur die Anforderungen für die Pakete und für die elektrische und physikalische Verbindung zwischen der abstrakten USB-Vorrichtung **32** und dem Bus. Folglich sind die Verbindungen und Verhältnisse, die in der [Fig. 4](#) gemäß dem Stand der Technik gezeigt werden, nur ein Beispiel einer Implementierung, welche die Anforderungen der USB-Spezifikation erfüllt. Somit muss jede spezifische Vorrichtung zur Erfüllung der USB-Spezifikation eine spezifisch definierte und beschriebene Architektur besitzen.

[0020] Leider existiert keine derartige Architektur für eine Flash-Speichervorrichtung mit einem oder mehreren Flash-Speichermodul(en), die der Flash-Speichervorrichtung eine Verbindung mit einem Bus ermöglichen würden, der gemäß einer USB-Spezifika-

tion definiert ist, und um somit ein Teil eines USB-Systems auf einer Hostrechner-Plattform zu werden. Zum Beispiel zeigt das U.S.-Patent Nr. 5,799,168 keine derartige Implementierung für die Flash-Vorrichtung oder schlägt diese vor. Wie oben erwähnt, wäre eine derartige Architektur insbesondere nützlich für eine Anzahl von Gründen, einschließlich geringe Kosten, einfache Verwendung und Transparenz für den Endbenutzer.

[0021] WO9901820A lehrt eine periphere Vorrichtung zur Verwendung mit einem Hostrechner. Anstatt Vorrichtungscharakteristiken der peripheren Vorrichtung in dem Speicher der peripheren Vorrichtung zu speichern, wie in dem in WO9901820A beschriebenen Stand der Technik, speichert das System von WO9901820A Vorrichtungscharakteristiken in dem Hostrechner und lädt Vorrichtungscharakteristiken auf die periphere Vorrichtung herunter, wenn die periphere Vorrichtung mit dem Hostrechner verbunden ist. Es wird auch erwähnt, dass der Speicher ein Flash-Speicher sein kann.

[0022] WO9908196A zeigt eine generische periphere USB-Vorrichtung, wie eine Massenspeichervorrichtung, in der nur ein oder zwei FIFO(s) bis zu dem nach dem USB-Standard möglichen Maximum von 16 Endpunkten versorgt/versorgen.

[0023] Somit besteht eine Notwendigkeit für eine Architektur zum Definieren und Beschreiben einer Flash-Speichervorrichtung, und es wäre nützlich, sie zu haben, die kompatibel zu einem USB-System ist und die der USB-Spezifikation derart folgt, dass die Flash-Speichervorrichtung sich auf einem USB-definierten Bus befindet und über den Bus mit der Hostrechner-Plattform kommuniziert.

[0024] Demgemäß besteht die vorliegende Erfindung aus einer USB-Flash-Speichervorrichtung zum Verbinden mit einem USB-definierten Bus, wobei die USB-Flash-Speichervorrichtung aufweist:

- (a) zumindest ein Flash-Speichermodul;
- (b) einen Verbinder, der ein USB-Verbinder ist, ausgebildet zur Verbindung mit einem USB-definierten Bus und zum Senden und Empfangen USB-definierter Pakete an den und von dem USB-definierten Bus; und
- (c) eine USB-Steuereinrichtung, die über den USB-Verbinder an einen Hostrechner anschließbar ist und die ein Lesen oder ein Schreiben des zumindest einen Flash-Speichermoduls gemäß den USB-definierten Paketen ausführt;

dadurch gekennzeichnet, dass die Steuereinrichtung einen Befehlsinterpretator aufweist, der ausgebildet ist, Lese- und Schreibbefehle, die als aus empfangenen USB-definierten Paketen gewonnenem Betriebscode über den USB-Verbinder empfangen werden, als Lese- und Schreib-Aktionen für das zumindest eine

Flash-Speichermodul zu extrahieren und interpretieren.

[0025] Weitere Ausführungsbeispiele der Erfindung werden in den angehängten abhängigen Ansprüchen spezifiziert.

[0026] Die vorliegende Erfindung betrifft eine Flash-Speichervorrichtung, die ein oder mehrere Flash-Modul(e) aufweist, wobei der Flash-Speicher dem Adressenbereich eines ASIC oder einer Steuereinrichtung zugeordnet ist, die eine USB-definierte elektrische Schnittstelle und eine USB-definierte logische Schnittstelle aufweist. Diese Steuereinrichtung/ASIC (im Folgenden als „Steuereinrichtung“ bezeichnet) unterstützt die USB-Funktionalität gemäß dem USB-Standard, wodurch sie eine Auflistung auf dem USB-Bus sowie einen Datenempfang und eine Datenübertragung über USB-Pipes an die USB-Endpunkte und von den USB-Endpunkten unterstützt. Diese Steuereinrichtung unterstützt auch die Funktionalität und Steuerung der Flash-Speichervorrichtung sowie die Verarbeitung von Befehls- und Datenpaketen von der Hostrechner-Steuereinrichtung. Die Hostrechner-Steuereinrichtung verwendet eines von mehreren möglichen Protokollen, entweder Standard oder proprietär, um den nächsten auszuführenden Befehl an die USB-Flash-Steuereinrichtung zu signalisieren. Somit wirkt die gesamte Vorrichtung als eine dynamisch anfügbare/abnehmbare nichtflüchtige Speichervorrichtung für die Hostrechner-Plattform.

[0027] Im Folgenden umfasst, ist aber nicht darauf begrenzt, der Begriff „Computer“ Personalcomputer (PC) mit einem Betriebssystem wie DOS, WindowsTM, OS/2TM oder Linux; MacintoshTM Computer; Computer mit JavaTM-OS als Betriebssystem; und graphische Workstations wie die Computer von Sun MicrosystemsTM und Silicon GraphicsTM und andere Computer mit einer Version des UNIX-Betriebssystems, wie AIXTM oder SOLARISTM von Sun MicrosystemsTM; oder jedes andere bekannte und verfügbare Betriebssystem, einschließlich Betriebssysteme wie Windows CETM für integrierte Systeme, einschließlich zelluläre Telefone, Handheld-Rechenvorrichtungen und Palmtop-Rechenvorrichtungen, und jede andere Rechenvorrichtung, die mit einem Netzwerk verbunden werden kann. Im Folgenden umfasst, ist aber nicht darauf begrenzt, der Begriff „WindowsTM“ Windows 95TM, Windows 3.xTM, wobei „x“ eine Ganzzahl wie „1“ ist, Windows NTTM, Windows 98TM, Windows CETM und jede aktualisierte Version dieser Betriebssysteme von Microsoft Inc. (Seattle, Washington, USA).

[0028] Die Erfindung wird nun beschrieben auf beispielhafte Weise unter Bezugnahme auf die beigefügten Zeichnungen, wobei

[0029] [Fig. 1](#) eine schematische Blockdarstellung

einer USB-Tokenpaketstruktur gemäß dem Stand der Technik ist;

[0030] [Fig. 2](#) eine schematische Blockdarstellung einer USB-Datenpaketstruktur gemäß dem Stand der Technik ist;

[0031] [Fig. 3](#) eine schematische Blockdarstellung einer USB-Handshake-Datenpaketstruktur gemäß dem Stand der Technik ist;

[0032] [Fig. 4](#) eine schematische Blockdarstellung einer beispielhaften USB-Vorrichtung gemäß dem Stand der Technik ist;

[0033] [Fig. 5](#) eine schematische Blockdarstellung eines Systems mit der Funktionalität einer USB-Vorrichtung gemäß der vorliegenden Erfindung ist;

[0034] [Fig. 6](#) eine schematische Blockdarstellung der USB-Flash-Platte ist;

[0035] [Fig. 7](#) eine schematische Blockdarstellung eines Flash-Identifizierungsanforderungspakets ist;

[0036] [Fig. 8](#) eine schematische Blockdarstellung eines Flash-Identifizierungsstatuspaket ist;

[0037] [Fig. 9](#) eine schematische Blockdarstellung eines Flash-Schreibanforderungspakets ist;

[0038] [Fig. 10](#) eine schematische Blockdarstellung eines Flash-Schreibstatuspaket ist;

[0039] [Fig. 11](#) eine schematische Blockdarstellung eines Flash-Leseanforderungspakets ist;

[0040] [Fig. 12](#) eine schematische Blockdarstellung eines Flash-Lesestatuspaket ist;

[0041] [Fig. 13](#) eine schematische Blockdarstellung eines Flash-Löschanforderungspakets ist; und

[0042] [Fig. 14](#) eine schematische Blockdarstellung eines Flash-Löschstatuspaket ist.

[0043] Die vorliegende Erfindung betrifft eine Flash-Speichervorrichtung, die ein oder mehrere Flash-Modul(e) aufweist, wobei der Flash-Speicher dem Adressenbereich eines ASIC oder einer Steuereinrichtung zugeordnet ist, die eine USB-definierte elektrische Schnittstelle und eine USB-definierte logische Schnittstelle aufweist. Diese Steuereinrichtung/ASIC (im Folgenden als „Steuereinrichtung“ bezeichnet) unterstützt die USB-Funktionalität gemäß dem USB-Standard, wodurch sie eine Auflistung auf dem USB-Bus sowie einen Datenempfang und eine Datenübertragung über USB-Pipes an die USB-Endpunkte und von den USB-Endpunkten unterstützt. Diese Steuereinrichtung unterstützt auch die Funkti-

onalität und Steuerung der Flash-Speichervorrichtung sowie die Verarbeitung von Befehls- und Datenpaketen von der Hostrechner-Steuereinrichtung. Die Hostrechner-Steuereinrichtung verwendet eines von mehreren möglichen Protokollen, entweder Standard oder proprietär, um den nächsten auszuführenden Befehl an die USB-Flash-Steuereinrichtung zu signalisieren. Somit wirkt die gesamte Vorrichtung als eine dynamisch anfügbare/abnehmbare nichtflüchtige Speichervorrichtung für die Hostrechner-Plattform.

[0044] Die Prinzipien und der Betrieb einer USB-Flash-Vorrichtung und eines USB-Flash-Systems gemäß der vorliegenden Erfindung ist einfacher zu verstehen unter Bezugnahme auf die Zeichnungen und die begleitende Beschreibung, wobei offensichtlich ist, dass diese Zeichnungen nur zum Zweck einer Illustration dienen und nicht einschränkend wirken sollen.

[0045] Unter Bezugnahme nun auf die Zeichnungen ist **Fig. 5** eine schematische Blockdarstellung der Hauptkomponenten einer Flash-Speichervorrichtung und eines Flash-Speichersystems gemäß der vorliegenden Erfindung. Ein Flash-Speichersystem **42** umfasst eine Hostrechner-Plattform **44**, wie gezeigt. Die Hostrechner-Plattform **44** betreibt eine USB-Flash-Vorrichtung **46** als einen nichtflüchtigen Speicherplatz.

[0046] Die Hostrechner-Plattform **44** ist mit der USB-Flash-Vorrichtung **46** gemäß der vorliegenden Erfindung über ein USB-Kabel **48** verbunden. Die Hostrechner-Plattform **44** ist mit dem USB-Kabel **48** über einen USB-Hostrechner-Verbinder **50** verbunden, während die USB-Flash-Vorrichtung **46** mit dem USB-Kabel **48** über einen USB-Flash-Vorrichtungs-Verbinder **52** verbunden ist. Die Hostrechner-Plattform **44** weist eine USB-Hostrechner-Steuereinrichtung **54** zum Steuern und Verwalten aller USB-Transfers auf dem USB-Bus auf.

[0047] Die USB-Flash-Vorrichtung **46** weist eine USB-Flash-Vorrichtungs-Steuereinrichtung **56** auf zum Steuern der anderen Komponenten der USB-Flash-Vorrichtung **46** und zum Vorsehen einer Schnittstelle für die USB-Flash-Vorrichtung **46** mit dem USB-Bus, dem USB-Flash-Vorrichtungs-Verbinder **52** und zumindest einem Flash-Speichermodul **58**. Das Flash-Speichermodul **58** ist vorzugsweise ein Array von Flash-Speichermodulen **58**, in denen die Daten gespeichert sind.

[0048] Wenn die USB-Flash-Vorrichtung **46** mit der Hostrechner-Plattform **44** verbunden wird, findet ein standardmäßiger USB-Auflistungsvorgang statt. Bei diesem Vorgang konfiguriert die Hostrechner-Plattform **44** die USB-Flash-Vorrichtung **46** und den Kommunikationsmodus mit der USB-Flash-Vorrichtung **46**. Obwohl es viele unterschiedliche Verfahren zur

Konfiguration der USB-Flash-Vorrichtung **46** gibt, wird, nur zum Zweck der Klarheit und ohne einschränkend sein zu wollen, die vorliegende Erfindung im Folgenden detaillierter erläutert hinsichtlich eines Verfahrens, in dem die Hostrechner-Plattform **44** Befehle und Anforderungen an die USB-Flash-Vorrichtung **46** durch einen Endpunkt ausgibt. Die Hostrechner-Plattform **44** fragt die USB-Flash-Vorrichtung **46** durch den anderen Endpunkt hinsichtlich Statusänderungen ab und empfängt verwandte Pakete, wenn derartige Pakete zum Empfang bereitstehen.

[0049] Die Hostrechner-Plattform **44** fordert Dienste von der USB-Flash-Vorrichtung **46** durch Senden von Anforderungspaketen an die USB-Hostrechner-Steuereinrichtung **54** an. Die USB-Hostrechner-Steuereinrichtung **54** überträgt Pakete auf dem USB-Kabel **48**. Diese Anforderungen werden von der USB-Flash-Vorrichtungs-Steuereinrichtung **56** empfangen, wenn die USB-Flash-Vorrichtung **46** die Vorrichtung am Endpunkt der Anforderung ist. Die USB-Flash-Vorrichtungs-Steuereinrichtung **56** führt dann verschiedene Operationen aus, wie Lesen von Daten aus dem/den Flash-Speichermodul(en) **58**, Schreiben von Daten in das/die Flash-Speichermodul(e) **58** oder Löschen von Daten aus dem/den Flash-Speichermodul(en) **58**, oder Unterstützen einer grundlegenden USB-Funktionalität, wie Auflistung und Konfiguration der Vorrichtung. Die USB-Flash-Vorrichtungs-Steuereinrichtung **56** steuert das/die Flash-Speichermodul(e) **58** durch die Verwendung einer Steuerungsleitung **60**, um das Flash-Speichermodul **58**/die Flash-Speichermodule **58** ein- bzw. auszuschalten, und ebenso durch verschiedene andere Signale, wie zum Beispiel Chip-Freigabe-, Lese- und Schreibsignale. Das/die Flash-Speichermodul(e) **58** ist/sind mit der USB-Flash-Vorrichtungs-Steuereinrichtung **56** auch über einen Adressen/Daten-Bus **62** verbunden. Der Adressen/Daten-Bus **62** überträgt Befehle zum Durchführen von Lese-, Schreib- oder Löschbefehlen an das/die Flash-Speichermodul(e) **58** sowie die Adressen und Daten für diese Befehle, wie von dem Hersteller des Flash-Speichermoduls **58**/der Flash-Speichermodule **58** definiert.

[0050] Damit die USB-Flash-Vorrichtung **46** die Hostrechner-Plattform **44** über das Ergebnis und den Status unterschiedlicher Operationen benachrichtigt, die von der Hostrechner-Plattform **44** angefordert wurden, überträgt die USB-Flash-Vorrichtung **46** Statuspakete unter Verwendung des „Statusendpunkts“. Gemäß diesem Vorgehen prüft (pollt) die Hostrechner-Plattform **44** hinsichtlich Statuspaketen und die USB-Flash-Vorrichtung **46** sendet entweder ein leerer Paket zurück, wenn keine Pakete für neue Statusmeldungen vorhanden sind, oder sendet alternativ das Statuspaket selbst zurück.

[0051] Eine detailliertere Struktur der funktionalen

Komponenten der USB-Flash-Vorrichtung **46** wird in [Fig. 6](#) gezeigt. Die USB-Flash-Vorrichtung **46** umfasst die für den USB-Standard definierte physikalische und elektrische Schnittstelle, die hier als der USB-Flash-Vorrichtungs-Verbinder **52** und eine Verbinder-Schnittstelle **64** gezeigt wird. Der USB-Flash-Vorrichtungs-Verbinder **52** empfängt die elektrischen Signale von dem USB-Kabel **48**, das elektrische Signale von der Hostrechner-Steuereinrichtung (nicht gezeigt) überträgt. Diese Signale werden dann durch die Verbinder-Schnittstelle **64** geleitet. Jede Millisekunde wird ein USB-Rahmen auf dem USB-Bus derart übertragen, dass Pakete an die USB-Flash-Vorrichtung **46** gesendet werden können.

[0052] Die Verbinder-Schnittstelle **64** empfängt dann diese Pakete durch eine erste Schnittstellenkomponente, wobei es sich um eine kombinierte physikalische und logische Schnittstelle **66** handelt. Eine funktionale Schnittstelle **68** wird speziell zum Empfang von Tokenpaketen gestaltet, wie in der USB-Spezifikation definiert und oben unter Bezugnahme auf [Fig. 1](#) beschrieben. Diese Tokenpakete betreffen nur bestimmte funktionale Aspekte der USB-Flash-Vorrichtung **46**, die für den USB-Standard erforderlich sind, und haben keinen Bezug zu einer bestimmten Anwendung der USB-Flash-Vorrichtung **46** als eine Flash-Platte gemäß der vorliegenden Erfindung. Diese Tokenpakete und ihre jeweiligen zurückgesendeten Datenpakete ermöglichen der USB-Hostrechner-Steuereinrichtung **54** (nicht gezeigt) und der Hostrechner-Plattform **44** (nicht gezeigt), die USB-Flash-Vorrichtung **46** zu identifizieren und Ressourcen für die USB-Flash-Vorrichtung **46** auf dem USB-Bus zuzuweisen. Somit unterstützt die funktionale Schnittstelle **68** nur eine USB-Funktionalität, die zur Identifizierung und Registrierung der USB-Flash-Vorrichtung **46** auf dem USB-Bus erforderlich ist.

[0053] Die USB-Flash-Vorrichtung **46** weist auch einen Anwendungspaket-Extraktor **70** auf, der die Anwendungsdaten und Befehle aus den USB-Anwendungspaketen derart extrahiert, dass der Anwendungspaket-Extraktor **70** nur anwendungsbezogene Pakete unterstützt. Dann werden alle Anforderungen an die USB-Flash-Vorrichtung **46** von der Hostrechner-Plattform **44** (nicht gezeigt) in der Form von Lese-, Schreib-, Identifizierungs- und Löschbefehlen von einem Anwendungsbefehlsinterpretator **72** interpretiert. Für alle Befehle, die Daten oder eine Adresse umfassen, wie Lese-, Schreib- und Löschbefehle, übersetzt ein Adressen-Auflösungsmodul **74** die Adresse von dem logischen Adressraum in den physikalischen Adressraum. Die Hostrechner-Plattform **44** (nicht gezeigt) bezieht sich auf einen linearen Adressraum von logischen Adressen, während die USB-Flash-Vorrichtung **46** zumindest ein und vorzugsweise eine Vielzahl von Flash-Modul(en) **58** umfasst, von denen jedes einen physikalischen Adress-

raum aufweist. Somit muss eine Übersetzung zwischen dem logischen Adressraum der Hostrechner-Plattform **44** (nicht gezeigt) und dem/den physikalischen Adressraum oder -räumen der USB-Flash-Vorrichtung **46** durchgeführt werden. Es gibt viele Wege, eine derartige Übersetzung zu implementieren, die für die vorliegende Erfindung geeignet sind. Ein Beispiel einer geeigneten Implementierung eines Adressübersetzungsverfahrens wird unter Bezugnahme auf das U.S.-Patent Nr. 5,404,485 beschrieben, das ein Verfahren zum Verwalten eines Flash-Speichers als eine Flash-Platte zeigt und das zum Einsatz mit der vorliegenden Erfindung geeignet ist.

[0054] Ein Daten-Handhabungsmittel **76** handhabt Daten-bezogene Aspekte von empfangenen Befehlen und überträgt die Daten durch die funktionale Schnittstelle **68** an das/die Flash-Modul(e) **58** und von dem/den Flash-Modul(en) **58**. Optional und vorzugsweise führt das Daten-Handhabungsmittel **76** Verfahren zur Fehlerkorrektur und -erfassung durch. Der Anwendungsbefehlsinterpretator **72**, das Daten-Handhabungsmittel **76** und das Adressen-Auflösungsmodul **74** arbeiten alle mit einem zugrunde liegenden Speichertechnologietreiber (MTD – memory technology driver) **78**, um ein bestimmtes Flash-Modul **58** und die gewünschte Adresse auf diesem Flash-Modul **58** zu schreiben, zu lesen oder zu löschen.

[0055] Die Hostrechner-Plattform **44** prüft hinsichtlich Statusänderungen in der USB-Flash-Vorrichtung **46** und liest Statuspakete von der USB-Flash-Vorrichtung **46**, wenn ein neues Statuspaket verfügbar ist. Unter Verwendung dieser Statuspakete kann die USB-Flash-Vorrichtung **46** an die Hostrechner-Plattform **44** die Ergebnisse unterschiedlicher Befehle übertragen, die von der Hostrechner-Plattform **44** in ihren Anforderungen (nicht gezeigt) ausgegeben wurden. Zum Beispiel umfasst das Lesebefehlsstatuspaket eines der verfügbaren Statuswörter, wie „Erfolg“, „Fehler“ oder „ungültige Adresse“, wodurch es der Hostrechner-Plattform **44** möglich ist, das Ergebnis des Lesebefehls (nicht gezeigt) zu bestimmen. Ähnlich umfasst das Löschstatuspaket ein Statuswort, das die Vollendung des Löschvorgangs anzeigt. Ein Schreibstatuspaket wird von der USB-Flash-Vorrichtung **46** verwendet, um die Hostrechner-Plattform **44** über das Ergebnis des Schreibbefehls zu benachrichtigen, zum Beispiel ob der Befehl erfolgreich oder fehlerhaft war und ob die USB-Flash-Vorrichtung **46** für zusätzliche Schreibanforderungen von der Hostrechner-Plattform **44** bereit ist.

[0056] Ein Speichertechnologietreiber (MTD) **78** umfasst typischerweise Routinen zum Lesen, Schreiben und Löschen der Flash-Speichervorrichtung, die von der Steuereinrichtung gesteuert wird, die den MTD **78** betreibt. Zusätzlich umfasst der MTD **78** eine

Identifizierungsroutine zum Erkennen des geeigneten Typs einer Flash-Speichervorrichtung, für die der MTD **78** entwickelt wurde, so dass die Steuereinrichtung bestimmen kann, welcher MTD bei einer Interaktion mit einem bestimmten Flash-Speichervorrichtungs-Array aktiviert werden soll. Zusätzlich sollte es einer Identifizierungsroutine möglich sein, die Größe des Arrays von Flash-Speichervorrichtungen, einschließlich der Anzahl von Flash-Speichervorrichtungen in dem Array, und verschiedene Merkmale der Flash-Array-Geometrie zu erfassen, wie Verschachtelung (interleaving) und Busbreite. Die Information ermöglicht später der Hostrechner-Plattform **44**, den Adressraum und die Größe des Speichermediums zu bestimmen. Das U.S.-Patent Nr. 5,799,168 offenbart ein Beispiel eines derartigen MTDs für eine Flash-Vorrichtung.

[0057] Unter Verwendung des oben beschriebenen Protokolls und der oben beschriebenen Architektur kann die Hostrechner-Plattform **44** optional jede Anwendung implementieren, die mit einer üblichen speicheradressierten (memory mapped) oder Ein-Ausgabe-zugeordneten (I/O mapped) Flash-Speichervorrichtung implementierbar ist. Zum Beispiel kann die Hostrechner-Plattform **44** jeder Anwendung eine standardmäßige Blockvorrichtungsschnittstelle zur Verfügung stellen, wie ein magnetisches Speichermedium „Festplattenlaufwerk“, wie in dem oben beschriebenen U.S.-Patent Nr. 5,404,485 offenbart wird.

[0058] Als ein Beispiel eines bevorzugten Ausführungsbeispiels der vorliegenden Erfindung wird der Betrieb eines Hostrechnersystems, das mit einer USB-Flash-Vorrichtung gemäß der vorliegenden Erfindung verbunden ist, hinsichtlich den Vorgängen einer Identifizierung, Programmierung, eines Lesens und Löschens der Flash-Vorrichtung beschrieben. Nur zum Zweck der Darstellung und ohne einschränkend wirken zu wollen, weist die beispielhafte USB-Flash-Vorrichtung ein Array von zwei Flash-Speichermodulen auf, von denen jedes eine Größe von 64 Mbit aufweist. Die Adressenübersetzungstabelle befindet sich in der Flash-Vorrichtung, so dass die Hostrechner-Plattform mit logischen Adressen arbeitet. Alle Befehle und Rückkehr-Codes zwischen der Flash-Vorrichtung und der Hostrechner-Plattform werden mit USB-Datenpaketen befördert und werden durch USB-Datenpipes übertragen. Die exakte Struktur der Pakete, Pipes und Timings werden in der USB-Spezifikation beschrieben.

[0059] Der Betrieb der beispielhaften Vorrichtung und des Systems gemäß der vorliegenden Erfindung ist wie folgt. Wenn die USB-Flash-Vorrichtung das erste Mal mit der Hostrechner-Plattform verbunden wird, weist die USB-Hostrechner-Steuereinrichtung der USB-Flash-Vorrichtung eine Adresse auf dem USB-Bus zu und weist auch Ressourcen zu, wie in

der USB-Spezifikation beschrieben wird. Tatsächlich fordert die USB-Flash-Vorrichtung die Hostrechner-Plattform auf, diese Ressourcen zu zuweisen und muss die Hostrechner-Plattform informieren, wie viele der Ressourcen erforderlich sind. Somit kann die USB-Flash-Platte optional langsamere Vorrichtungsgeschwindigkeiten unterstützen, wenn die USB-Hostrechner-Plattform Ressourcen bereits anderen Vorrichtungen zugewiesen hat.

[0060] Die USB-Steuereinrichtung verhandelt auch mit den Flash-Modulen und bestimmt die Größe und den Herstellungstyp dieser Module. Die Steuereinrichtung errichtet dann eine Identifizierungsstruktur, welche die Information aufweist, sowie die Übersetzungstabelle und einen logischen Adressraum.

[0061] Nachdem die USB-Hostrechner-Steuereinrichtung die USB-Flash-Vorrichtung identifiziert, lädt die Hostrechner-Plattform oft einen USB-Client-Treiber hoch. Der Treiber gibt einen Identifizierungsanforderungsbefehl an die USB-Hostrechner-Steuereinrichtung aus und veranlasst die Steuereinrichtung, ein in [Fig. 7](#) gezeigtes Identifizierungsdatenpaket **80** zu übertragen. Das Identifizierungspaket **80** umfasst ein PID-Feld **22** und ein Prüfsummenfeld **26**, wie oben für die [Fig. 2](#) gemäß dem Stand der Technik beschrieben wurde. Das Identifizierungspaket **80** umfasst auch einen „Identifizierungs-“Betriebscode in einem Betriebscodefeld **82**. Der Paket-Extraktor der USB-Flash-Vorrichtung empfängt das Identifizierungsdatenpaket **80** und überträgt den Betriebscode des Identifizierungsbefehls an den Anwendungsfehlsinterpretierer.

[0062] Als Antwort auf den Identifizierungsbefehl sendet die Flash-Vorrichtung dann ein Identifizierungsdatenpaket **84**, das in [Fig. 8](#) gezeigt wird. Zusätzlich zu den in der [Fig. 7](#) gezeigten Feldern, umfasst das Identifizierungsdatenpaket **84** auch eine Information über die Größe der Flash-Vorrichtung in einem Flash-Vorrichtungsgrößenfeld **86** sowie Information über die Größe der Minimallöscheinheit zum Löschen des Flash-Speichers in einem Löscheinheitsgrößenfeld **88**.

[0063] Alle der in diesem Beispiel beschriebenen Pakete sind nur Datenpakete, die auf dem USB-Bus gesendet werden. Bevor jedes Paket gesendet wird, wird ein USB-Tokenpaket übertragen, das die USB-Steuereinrichtung über die Identität des Vorrichtungsendpunkts anweist, an den das Datenpaket übertragen werden soll. Nach einem erfolgreichen Empfang des Pakets gibt die USB-Steuereinrichtung ein USB-ACK-Paket aus, wie in der USB-Spezifikation beschrieben wird.

[0064] Wenn die Vorrichtungstreiber in der Hostrechner-Plattform dieses Statuspaket erhalten, können die Treiber beginnen, Lese- und Schreibbefehle

an die USB-Flash-Vorrichtung mit den Anwendungsbefehlen auszugeben. Wenn eine Schreibanforderung gesendet wird, werden ein USB-Datenpaket mit dem Betriebscode für den Schreibbefehl und der die Daten enthaltende Puffer an die USB-Flash-Vorrichtung übertragen. Ein Schreibdatenpaket **90** wird in [Fig. 9](#) gezeigt, das wiederum die oben in [Fig. 8](#) gezeigten Felder umfasst, außer dass das Schreibdatenpaket **90** ebenso ein Schreibfeld **92** mit dem Schreibbetriebscode; ein ADDR-Feld **94** mit der zu schreibenden logischen Adresse; ein LEN-Feld **96** mit der zu schreibenden Länge; und ein DATEN-Feld **98** umfasst, das die zu schreibenden tatsächlichen Daten umfasst. Der Paket-Extraktor extrahiert den Betriebscode aus dem Schreibdatenpaket **90** und überträgt diesen Code an den Anwendungsbefehlinterpretierer. Die logische Adresse wird an das Adressen-Auflösungsmodul übertragen, das die logische Adresse in eine physikalische Adresse auf einem der Flash-Module übersetzt. Das Daten-Handhabungsmittel berechnet optional einen Fehlerkorrektur- und Fehlererkennungsmechanismus, wenn er von der USB-Flash-Vorrichtung eingesetzt wird. Wenn alle der Flash-Speichermodule bereit/fertig sind, wird ein Schreibbefehl an das Flash-Modul oder die Flash-Module gesendet, das/die die physikalische Adresse umfasst/umfassen, die sich optional über mehr als ein Flash-Modul für den MTD-Block erstrecken kann. Der MTD-Block gibt dann einen Schreibbefehl auf dem Daten/Adress-Bus aus, der die Flash-Module mit der USB-Vorrichtungs-Steuereinrichtung verbindet. Wenn die Operation vollständig ist und ein Statuspaket zu dem MTD zurückgesendet ist, wird das Ergebnis der Operation an die Hostrechner-Steuereinrichtung übertragen und an den Vorrichtungstreiber in der Hostrechner-Plattform geleitet.

[0065] Wenn die Flash-Steuereinrichtung den Schreibvorgang beendet, signalisiert die Steuereinrichtung an die Hostrechner-Plattform, dass sich der Status der USB-Flash-Speichervorrichtung geändert hat, durch Senden eines „Schreibstatus“-Pakets **100**, wie in [Fig. 10](#) gezeigt wird. Anstatt des Datenfelds **98** umfasst das Schreibstatuspaket **100** ein Statusfeld **102**. Die Hostrechner-Plattform liest die Statuspakete von der Flash-Speichervorrichtung und aus dem Schreibstatuspaket **100** ruft die Hostrechner-Plattform eine Information über den Vollendungszustand des Schreibbefehls durch das Lesestatusfeld **102** ab. In diesem Beispiel wiederholt die Flash-Speichervorrichtung das ADDR-Feld **94** und das LEN-Feld **96**, damit die Hostrechner-Plattform eine Referenz für den spezifischen Befehl hat, der das Statuspaket **100** betrifft.

[0066] Wie in [Fig. 11](#) gezeigt, umfasst ein „Leseanforderungs“-Paket **104** den Betriebscode für den „Lese“-Befehl in einem Lesefeld **106** und die logische Adresse der gewünschten Stelle, von der die

Flash-Steuereinrichtung lesen soll, in einem ADDR-Feld **108**. Bei Erhalt dieses Befehls gibt die Flash-Steuereinrichtung einen Lesebefehl an den MTD-Block aus, nachdem das Adressen-Auflösungsmodul die in dem ADDR-Feld **108** enthaltene Adresse in eine spezifische physikalische Adresse in einer der Flash-Komponenten übersetzt hat.

[0067] Wenn die Flash-Steuereinrichtung die Daten von der Flash-Vorrichtung erhält, entweder nachdem der Lesebefehl ausgegeben wurde oder wenn ein Fehler aufgetreten ist, sendet die Flash-Steuereinrichtung ein Signal an die Hostrechner-Plattform, um anzulegen, dass ein neues Statuspaket zu lesen ist. Die Hostrechner-Plattform gibt eine Leseanforderung aus und empfängt ein „Lesestatus“-Paket **110**, wie in [Fig. 12](#) gezeigt wird. Das Lesestatuspaket **110** umfasst die Adresse der Lesedaten in dem ADDR-Feld **108** sowie die Länge der Lesedaten in einem LEN-Feld **112** und die Daten selbst in einem Datenfeld **114**. Das Lesestatuspaket **110** weist auch das Statuswort, gemäß dem der Betrieb/die Operation abgeschlossen ist, in einem Statusfeld **116** auf. Der Lesebetrieb kann mit vielen unterschiedlichen Stausituationen abgeschlossen werden, wie Erfolg, Ausfall, Fehler erfasst, ungültige Adresse, ungültige Länge usw.

[0068] Wenn die Hostrechner-Plattform eine Löscheinheit in der Flash-Vorrichtung löschen muss, gibt die Hostrechner-Plattform ein „Löschanforderungs“-Paket **118** aus, wie in [Fig. 13](#) gezeigt. Dieses Paket umfasst den „Löschen“-Betriebscode in einem Löschfeld **120** und die logische Adresse der Löscheinheit in einem ADDR-Feld **122**. Bei Erhalt einer derartigen Anforderung übersetzt die Flash-Steuereinrichtung die logische Adresse in eine physikalische Löscheinheitsadresse in einem der physikalischen Adressräume der Flash-Module und gibt einen Löschbefehl an den MTD-Block aus.

[0069] Der Löschvorgang nimmt im Allgemeinen mehr Zeit in Anspruch als ein Lese- oder Schreibvorgang. Wenn dieser Löschvorgang beendet ist, benachrichtigt die Steuereinrichtung die Hostrechner-Plattform, dass ein neues Statuspaket bereit zur Übertragung ist. Die Steuereinrichtung überträgt dann ein „Löschstatus“-Paket **124**, wie in [Fig. 14](#) gezeigt. Das Löschstatuspaket **124** umfasst die Adresse der gelöschten Einheit in dem ADDR-Feld **122**, dadurch liefert es der Hostrechner-Plattform eine Referenz für die Löschanforderungen. Der Status, gemäß dem der Betrieb abgeschlossen ist, ist in einem Statusfeld **126** vorgesehen.

Patentansprüche

1. USB-Flash-Speichervorrichtung (**46**) zum Verbinden mit einem USB-definierten Bus (**48**), wobei die USB-Flash-Speichervorrichtung (**46**) umfasst:

- (a) zumindest ein Flash-Speichermodul (58);
- (b) einen einzigen Verbinder (52), der ein USB-Verbinder ist, ausgebildet zur Verbindung mit einem USB-definierten Bus und zum Senden und Empfangen USB-definierter Pakete an den und von dem USB-definierten Bus (48); und
- (c) eine USB-Steuereinrichtung (56), die sich an einen Hostrechner (44) über den USB-Verbinder (52) anschließen lässt und die zumindest ein Lesen oder ein Schreiben des zumindest einen Flash-Speichermoduls (58) gemäß den USB-definierten Paketen ausführt,

wobei die Vorrichtung als eine ganzheitliche Einheit mit dem einzigen Verbinder vorgesehen ist und die Steuereinrichtung (56) einen Befehlsinterpret (72), der zum Interpretieren von Lese- und Schreib-Befehlen, die als aus empfangenen USB-definierten Datenpaketen (90, 104) gewonnener Betriebscode über den USB-Verbinder (52) empfangen werden, als Lese- und Schreib-Aktionen für das zumindest eine Flash-Speichermodul (58) ausgebildet ist, und ein Adressen-Auflösungsmodul (74), das zum Umsetzen einer logischen Adresse (94) von den USB-definierten Datenpaketen (90, 104) in eine physikalische Adresse in einem oder mehreren des zumindest einen Flash-Speichermoduls (58) ausgebildet ist, umfasst.

2. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 1, wobei das Adressen-Auflösungsmodul (74) zum Empfangen der USB-definierten Datenpakete (90, 104) und zum Auflösen der in den USB-definierten Datenpaketen (90, 104) enthaltenen Adressen (94, 108) ausgebildet ist, wobei die Adressen (94, 108) derart an die USB-Steuereinrichtung (56) gesendet werden, dass die Lese- und Schreib-Befehle gemäß den Adressen (94, 108) durchgeführt werden.

3. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 2, wobei, wenn der Befehl ein Schreib-Befehl zum Schreiben von Daten (98) in das zumindest eine Flash-Speichermodul (58) und die Adresse (94) eine logische Adresse zum Schreiben der Daten (98) ist, das Adressen-Auflösungsmodul (74) die logische Adresse (94) in eine physikalische Adresse des zumindest einen Flash-Speichermoduls (58) auflöst.

4. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 2, wobei, wenn der Befehl ein Lese-Befehl zum Lesen von Daten (114) aus dem zumindest einen Flash-Speichermodul (58) und die Adresse (108) eine logische Adresse zum Lesen der Daten (114) ist, das Adressen-Auflösungsmodul (74) die logische Adresse (108) in eine physikalische Adresse des zumindest einen Flash-Speichermoduls (58) auflöst.

5. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 2, die ferner aufweist:
ein Daten-Handhabungsmittel (76) zum Durchführen

einer Fehlererkennungs- und Fehlerkorekturoutine für das zumindest eine Flash-Speichermodul (58).

6. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 5, die ferner aufweist:
ein Zustands-Handhabungsmittel (76) zum Empfangen der USB-definierten Datenpakete (90, 104) und zum Senden von Zustandspaketen (100, 110), die einen Zustand des zumindest einen Flash-Speichermoduls (58) gemäß den USB-definierten Datenpaketen (90, 104) betreffen.

7. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 6, die ferner aufweist:
einen Speichertechnologietreiber (MTD) (78) zum Empfangen eines Schreib-Befehls und einer physikalischen Adresse des zumindest einen Flash-Speichermoduls (58) und zum Durchführen des Schreib-Befehls auf die physikalische Adresse.

8. USB-Flash-Speichervorrichtung (46) gemäß einem der Ansprüche 1 bis 7, wobei die Vorrichtung (46) als eine dynamisch anfügbare/abnehmbare nichtflüchtige Speichervorrichtung für den Hostrechner (44) dient.

9. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei die USB-Steuereinrichtung (56) als eine einzelne integrierte Schaltung implementiert ist.

10. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei die USB-Steuereinrichtung (56) Schreib-Befehle interpretiert und von zu schreibenden Daten (98) aus den USB-definierten Datenpaketen (90) extrahiert.

11. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei der USB-definierte Bus mit dem Hostrechner (44) verbunden ist und wobei der Hostrechner (44) Befehle unter Verwendung eines proprietären Protokolls an die USB-Steuereinrichtung (56) liefert.

12. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei der USB-definierte Bus mit dem Hostrechner (44) verbunden ist und wobei der Hostrechner (44) Befehle unter Verwendung eines Standard-Protokolls an die USB-Steuereinrichtung (56) liefert.

13. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, mit einem Adressen/Daten-Bus (62), der die USB-Steuereinrichtung (56) mit dem zumindest einen Flash-Speichermodul (58) verbindet.

14. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei die USB-Steuereinrichtung (56) mit dem zumindest ei-

nen Flash-Speichermodul (58) verhandelt, um zu mindest ein Merkmal der Geometrie des Flashmoduls zu bestimmen.

15. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 14, wobei das zumindest eine Merkmal eine Größe umfasst.

16. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 14 oder 15, wobei das zumindest eine Merkmal einen Herstellungstyp umfasst.

17. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 16, wobei die USB-Steuereinrichtung (56) den bestimmten Typ verwendet, um einen Speicher-technologietreiber (MTD) (78) zur Verwendung für den Flash-Speicher (58) zu bestimmen.

18. USB-Flash-Speichervorrichtung (46) gemäß einem der Ansprüche 14 bis 17, wobei die Steuereinrichtung (56) den Hostrechner (44) benachrichtigt, dass sie nach der Verhandlung bereit ist.

19. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 14, wobei das zumindest eine Merkmal eine Größe und einen Herstellungstyp umfasst.

20. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 19, wobei die bestimmte Größe und der bestimmte Typ verwendet werden, um eine Übersetzungstabelle und einen Adressraum zu erzeugen.

21. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 20, wobei die USB-Steuereinrichtung (56) die Übersetzungstabelle und den Adressraum erzeugt.

22. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei die Steuereinrichtung (56) eine Vielzahl von Chip-Freigabe-Signalleitungen (60) zur Verbindung mit einer Vielzahl von Flash-Speichermodulen (58) umfasst.

23. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei der Verbinder (52) mit der Steuereinrichtung (56) über eine kombinierte physikalische/logische Schnittstelle (66) verbunden ist.

24. USB-Flash-Speichervorrichtung (46) gemäß Anspruch 23, wobei die kombinierte physikalische/logische Schnittstelle (66) ein Teil der Steuereinrichtung (56) ist.

25. USB-Flash-Speichervorrichtung (46) gemäß einem der vorhergehenden Ansprüche, wobei das zumindest ein Flash-Speichermodul (58) eine Vielzahl von Flash-Speichermodulen (58) aufweist.

26. USB-Flash-Speichervorrichtung gemäß ei-

nem der vorhergehenden Ansprüche, wobei die USB-Steuereinrichtung (56) ferner umfasst:

(i) eine funktionelle Schnittstelle, die ausgebildet ist, die USB-definierten Pakete derart zu empfangen, dass, wenn eines der USB-definierten Pakete ein USB-Token-Paket ist, die funktionelle Schnittstelle auf das Token-Paket wirkt; und

(ii) einen Paket-Extraktor, der nach der funktionellen Schnittstelle in Reihe geschaltet ist und ausgebildet ist, die USB-definierten Pakete zu empfangen, wobei der Paket-Extraktor zumindest die Befehle aus den USB-definierten Paketen extrahiert; wobei die Vorrichtung dynamisch anfügbar/abnehmbar ist.

27. USB-Flash-Speichervorrichtung (46) gemäß einem der Ansprüche 1–26, die ferner umfasst:

(d) Mittel (66, 68) zum Bereitstellen einer USB-Schnittstellenfunktionalität für den Verbinder (52);

und wobei die Steuereinrichtung (56) ferner umfasst:

(i) Mittel (70) zum Extrahieren von Flash-Speicherbefehlen aus den über den Verbinder (52) empfangenen USB-definierten Datenpaketen (90, 104); und
(ii) Mittel (72, 74, 78) zum Steuern der Flash-Speichermittel als Reaktion auf die Befehle.

28. USB-Flash-Speichervorrichtung (46) gemäß einem der Ansprüche 1–26, die ferner umfasst:

(d) Mittel (66, 68) zum Bereitstellen einer USB-Schnittstellenfunktionalität für den Verbinder (52);

und wobei die Steuereinrichtung (56) ferner aufweist:

(i) einen Befehls-Extraktor (70), der betriebsfähig ist, Flash-Speicherbefehle aus den über den Verbinder (52) empfangenen USB-definierten Datenpaketen (90, 104) zu extrahieren, und
(ii) einen Speicher-technologietreiber (78), der betriebsfähig ist, das zumindest eine Flash-Speichermodul (58) als Reaktion auf die Befehle zu steuern.

Es folgen 7 Blatt Zeichnungen

Anhängende Zeichnungen

Fig. 1 - USB-Tokenpaketstruktur gemäß dem Stand der Technik

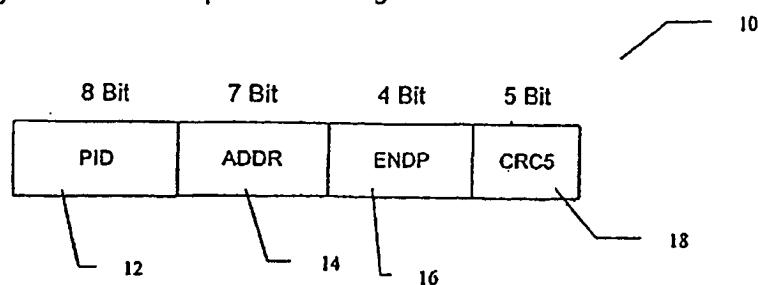


Fig. 2 - USB-Datenpaketstruktur gemäß dem Stand der Technik

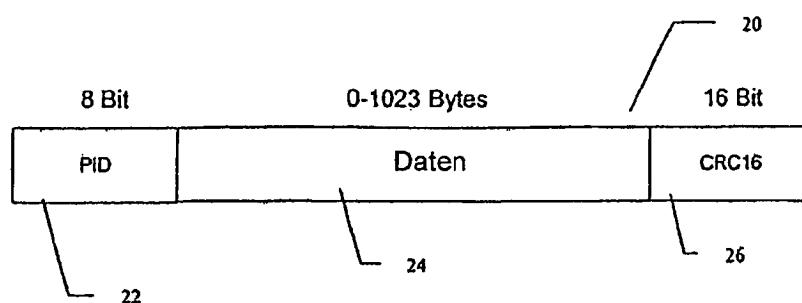


Fig. 3 - USB-Handshake-Paket gemäß dem Stand der Technik

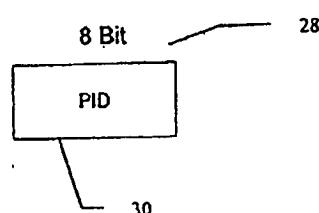


Fig. 4 – Funktionale Blöcke einer USB-Vorrichtung gemäß dem Stand der Technik

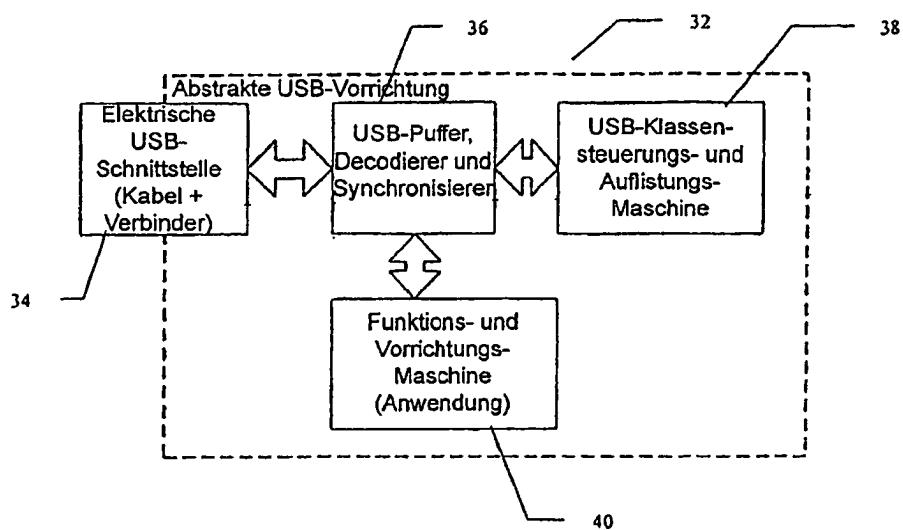


Fig. 5 – Computer-Hostrechnersystem mit einer USB-Flash-Speichervorrichtung

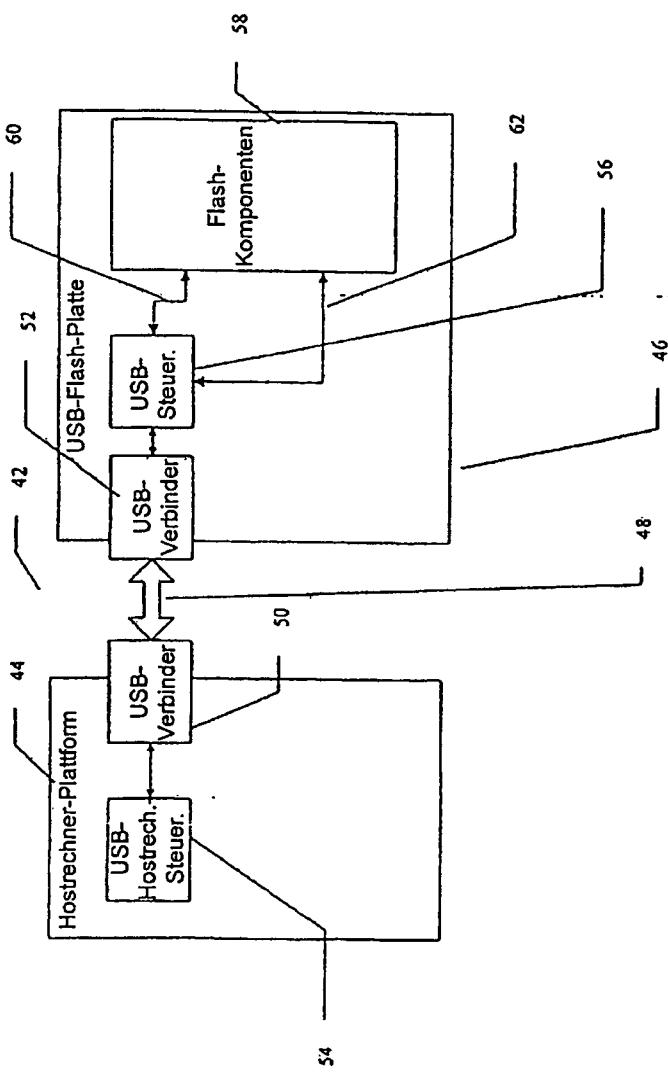


Fig. 6 – Detaillierte funktionale Blöcke einer USB-Flash-Speichervorrichtung

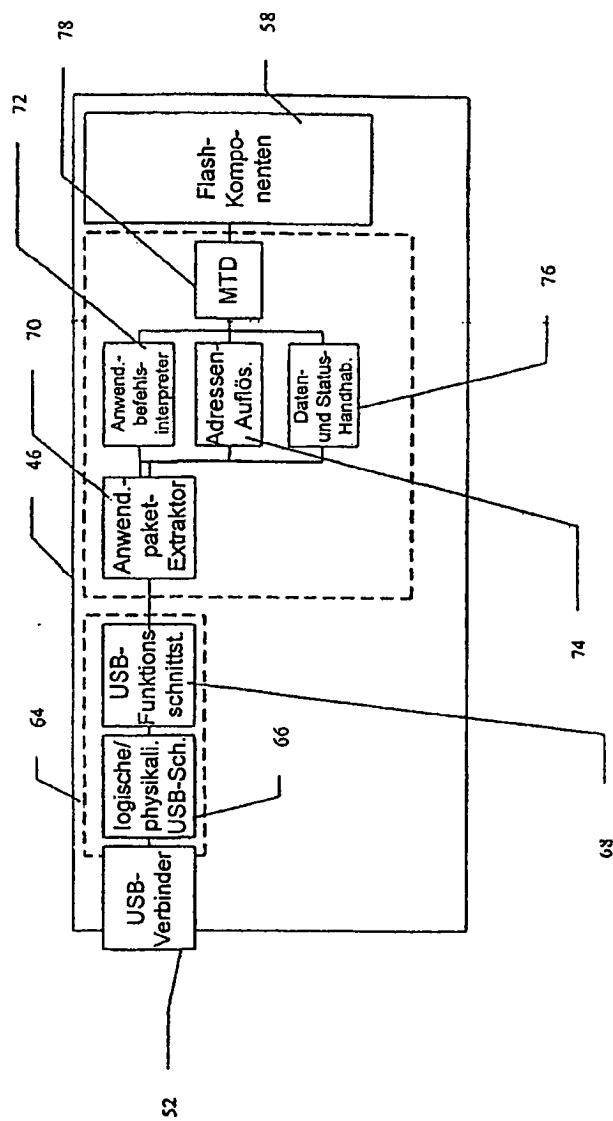


Fig. 7 – Flash-Identifizierungsanforderungspaket

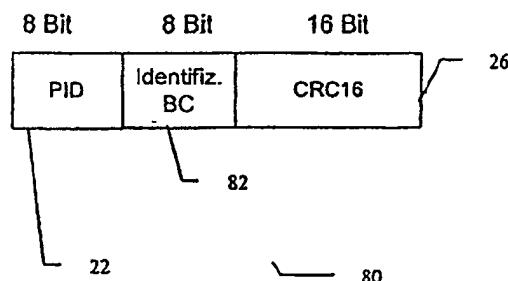


Fig. 8 – Flash-Identifizierungsantwortpaket

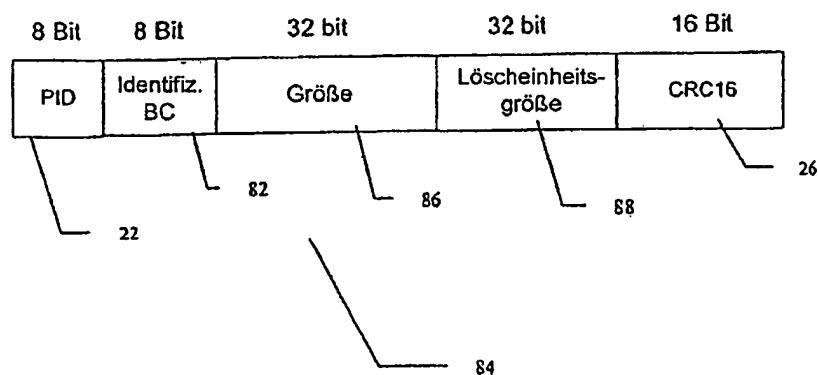


Fig. 9 - Schreibanforderungspaket

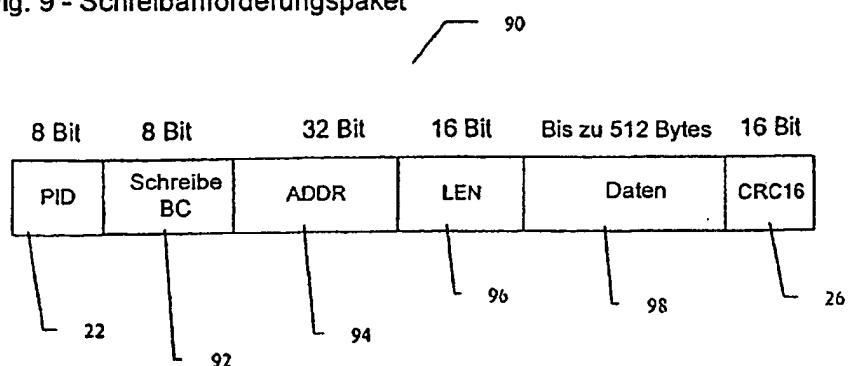


Fig. 10 - Schreibstatuspaket

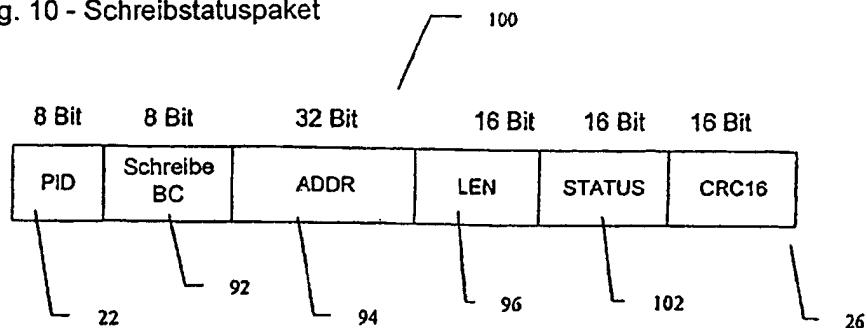


Fig. 11 – Leseanforderungspaket

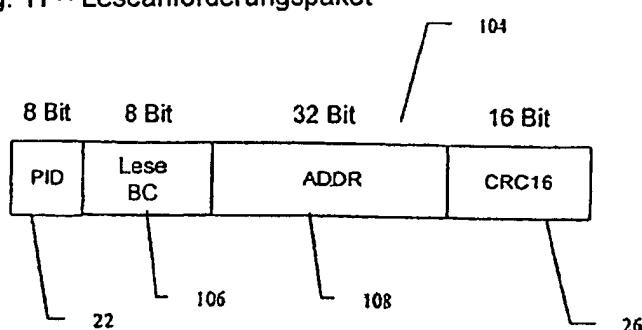


Fig. 12 – Lesestatuspaket

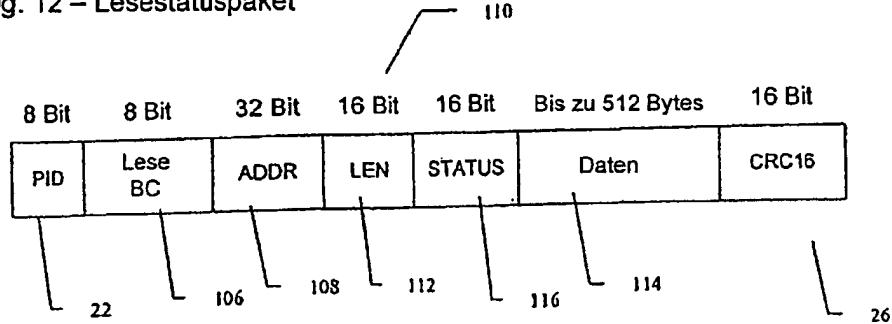


Fig. 13 – Löschanforderungspaket

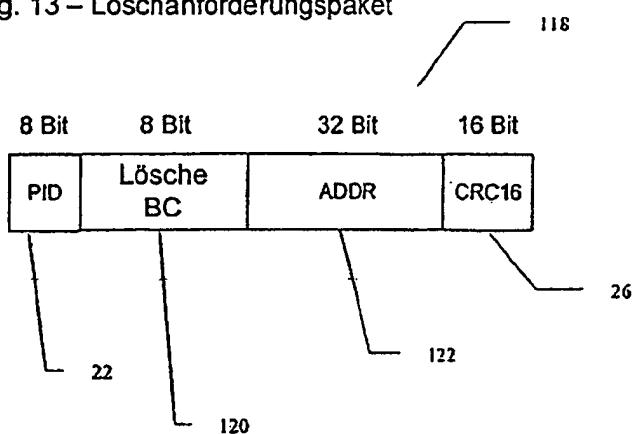


Fig. 14 – Löschstatuspaket

