



(12) **Offenlegungsschrift**

(21) Aktenzeichen: **10 2017 121 465.8**  
 (22) Anmeldetag: **15.09.2017**  
 (43) Offenlegungstag: **29.03.2018**

(51) Int Cl.: **G06F 13/20 (2006.01)**  
**G06F 13/38 (2006.01)**  
**G06F 13/42 (2006.01)**

(30) Unionspriorität:  
**15/277,844**      **27.09.2016**      **US**

(74) Vertreter:  
**Schweiger & Partners, 80687 München, DE**

(71) Anmelder:  
**Lenovo Enterprise Solutions (Singapore) Pte.  
 Ltd., Singapur, SG**

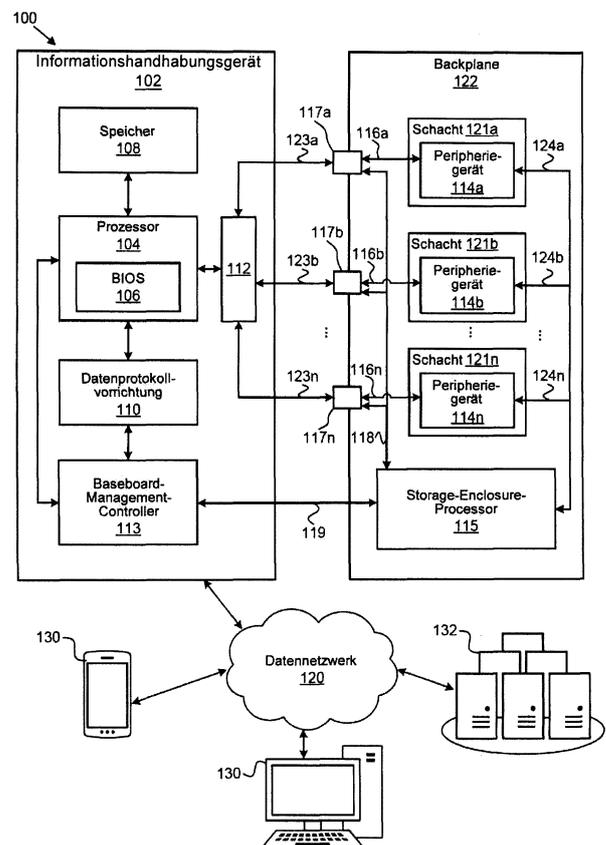
(72) Erfinder:  
**Remis, Luke, Raleigh, N.C., US; Velez, Wilson,  
 Raleigh, N.C., US; Andersen, Mark E., Cary, N.C.,  
 US**

Prüfungsantrag gemäß § 44 PatG ist gestellt.

**Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen.**

(54) Bezeichnung: **DATENPROTOKOLL ZUM VERWALTEN VON PERIPHERIEGERÄTEN**

(57) Zusammenfassung: Es werden eine Vorrichtung, ein System, ein Verfahren und ein Programmprodukt zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls offenbart. Ein Verbindungsmodul bestimmt ein oder mehrere Kommunikationsbuskabel, die ein oder mehrere Peripheriegeräte kommunikationstechnisch mit einem Informationshandhabungsgerät koppeln. Ein Datenmodul liest über jedes des einen oder der mehreren Kommunikationsbuskabel ein Datenpaket, das mit einem Peripheriegerät assoziiert ist. Jedes Datenpaket beinhaltet eine Kennung für eine Position, an der das Peripheriegerät installiert ist. Ein Topologiemodul bestimmt eine Kabelverbindungskonfiguration für das eine oder die mehreren Kommunikationsbuskabel und das eine oder die mehreren Peripheriegeräte, basierend auf der Installationspositionskenung, die aus jedem der Datenpakete empfangen wurde, die mit dem einen oder dem mehreren Peripheriegeräten assoziiert sind.



**Beschreibung**

## GEBIET DER ERFINDUNG

**[0001]** Der hier offenbarte Erfindungsgegenstand betrifft Peripheriecomputergeräte und insbesondere ein Datenprotokoll zum Verwalten von Peripheriegeräten.

## STAND DER TECHNIK

**[0002]** Computergeräte können Peripheriegeräte beinhalten, die mit Hilfe eines oder mehrerer Kommunikationsbusse mit dem Computergerät gekoppelt sind. Diese Kommunikationsbusse können Kommunikationsbuskabel sein, die Peripheriegeräte kommunikationstechnisch mit einem Informationshandhabungsgerät koppeln. Herkömmliche Kommunikationsbusarchitekturen und -protokolle sind möglicherweise nicht in der Lage, mehrere Peripheriegeräte zu verwalten, die mit Hilfe von Kommunikationsbuskabeln mit einem Informationshandhabungsgerät gekoppelt sind.

## KURZDARSTELLUNG

**[0003]** Es wird eine Vorrichtung zum Verwalten von Peripheriegeräten offenbart. Außerdem führen ein Verfahren und ein Computerprogrammprodukt die Funktionen der Vorrichtung aus. Die Vorrichtung beinhaltet ein Verbindungsmodul, das ein oder mehrere Kommunikationsbuskabel bestimmt, die ein oder mehrere Peripheriegeräte kommunikationstechnisch mit einem Informationshandhabungsgerät koppeln. Die Vorrichtung beinhaltet außerdem ein Datenmodul, das über jedes des einen oder der mehreren Kommunikationsbuskabel ein Datenpaket liest, das mit einem Peripheriegerät assoziiert ist. Jedes Datenpaket beinhaltet eine Kennung für eine Position, an der das Peripheriegerät installiert ist.

**[0004]** Die Vorrichtung beinhaltet ferner ein Topologiemodul, das basierend auf der Installationspositionskennung, die aus jedem der Datenpakete empfangen wurde, die mit dem einen oder den mehreren Peripheriegeräten assoziiert sind, eine Kabelverbindungskonfiguration für das eine oder die mehreren Kommunikationsbuskabel und das eine oder die mehreren Peripheriegeräte bestimmt. Bei einer Ausführungsform stellt das Topologiemodul die Kabelverbindungskonfiguration des Kommunikationsbuskabels und der Peripheriegeräte einem Basic Input/Output System (BIOS) bereit.

**[0005]** Bei einer weiteren Ausführungsform beinhaltet die Vorrichtung ein Benachrichtigungsmodul, das in Reaktion darauf, dass die bestimmte Kabelverbindungskonfiguration nicht mit einer vordefinierten Kabelverbindungskonfiguration übereinstimmt, eine Benachrichtigung sendet. Bei einigen Ausführungsformen ist/sind das ein oder die mehreren Peripheriegeräte jeweils in einem Laufwerksschacht einer Backplane installiert und beinhaltet die Kennung eine Schachtkennung und eine assoziierte Slotkennung des Laufwerksschachts. Bei verschiedenen Ausführungsformen beinhaltet das Datenpaket mehrere Felder, welche die Kennung definieren. Die mehreren Felder können ein Feld für die Schachtkennung und ein Feld für die Slotkennung beinhalten.

**[0006]** Bei einer Ausführungsform beinhaltet das Datenpaket ferner ein oder mehrere Felder, die Daten beinhalten, welche eine oder mehrere Eigenschaften eines Storage Enclosure Processor (SEP) einer Backplane beschreiben. Bei verschiedenen Ausführungsformen beinhaltet die Vorrichtung ein Abstimmungsmodul, das basierend auf der einen oder den mehreren Eigenschaften des SEP eine oder mehrere Leistungseigenschaften eines Field Programmable Gate Array (FPGA) dynamisch modifiziert. Das FPGA kann mit Hilfe eines Kommunikationsbuskabels kommunikationstechnisch mit dem SEP und einem oder mehreren Peripheriegeräten der Backplane gekoppelt sein.

**[0007]** Bei einigen Ausführungsformen beinhalten die eine oder die mehreren Eigenschaften für den SEP einen Bitschalter, der anzeigt, ob der SEP Datenparität unterstützt, eine maximale Taktgeschwindigkeit des SEPs, einen Zeitsperreparameter für das Polling des SEP und/oder eine maximale Anzahl an Peripheriegeräten, die von dem SEP unterstützt werden. Bei bestimmten Ausführungsformen beinhaltet die Vorrichtung ein Bestätigungsmodul, das mit Hilfe einer Prüfsumme, die in einem Prüfsummenfeld des Datenpakets enthalten sein kann, eine Integrität von Daten prüft, die in einem oder mehreren Feldern des Datenpakets enthalten sind.

**[0008]** Bei einigen Ausführungsformen beinhaltet die Vorrichtung ein Übertragungsmodul, das einen Befehl für einen Virtual Pin Port (VPP) von einem VPP-aktivierten Prozessor empfängt. Der VPP-Befehl kann für ein Peripheriegerät auf einer Backplane vorgesehen sein, welches durch die Installationspositionskennung identifiziert wird. Das Übertragungsmodul schreibt in einer weiteren Ausführungsform den VPP-Befehl in ein

entsprechendes Feld in dem Datenpaket und sendet das Datenpaket, das den VPP-Befehl beinhaltet, an einen Storage Enclosure Processor (SEP) auf der Backplane, mit Hilfe eines Kommunikationsbuskabels, das mit dem SEP gekoppelt und mit dem vorgesehenen Peripheriegerät assoziiert ist.

**[0009]** Bei einigen Ausführungsformen schreibt das Übertragungsmodul einen Paritätswert in ein Paritätsfeld des Datenpakets, das vom SEP verwendet werden kann, um die Integrität von Daten zu bestätigen, die in das Datenpaket geschrieben sind. Bei einer weiteren Ausführungsform beinhaltet der VPP-Befehl einen oder mehrere Peripheriegerät-Aktivierungsbefehle, einen Peripheriegerät-Deaktivierungsbefehl und/oder einen Peripheriegerät-Tätigkeitsanzeichen-Umschaltbefehl. Bei einigen Ausführungsformen beinhaltet das Kommunikationsbuskabel ein Peripheral Component Interconnection express („PCIe“) – kabel. Bei verschiedenen Ausführungsformen beinhaltet das Peripheriegerät ein Non Volatile Memory express (NVMe)-Speichergerät.

**[0010]** Ein Verfahren beinhaltet das Bestimmen eines oder mehrerer Kommunikationsbuskabel, die ein oder mehrere Peripheriegeräte kommunikationstechnisch mit einem Informationshandhabungsgerät koppeln. Das Verfahren beinhaltet ferner das Lesen eines Datenpakets, das mit einem Peripheriegerät assoziiert ist, über das eine oder die mehreren Kommunikationsbuskabel. Jedes Datenpaket beinhaltet eine Kennung für eine Position, an der das Peripheriegerät installiert ist. Das Verfahren beinhaltet außerdem das Bestimmen einer Kabelverbindungskonfiguration des einen oder der mehreren Kommunikationsbuskabel und des einen oder der mehreren Peripheriegeräte, basierend auf der Installationspositionskennung, die aus jedem der Datenpakete empfangen wurde, die mit dem einen oder den mehreren Peripheriegeräten assoziiert sind.

**[0011]** Das Verfahren beinhaltet bei einer Ausführungsform das Bereitstellen der Kabelverbindungskonfiguration des Kommunikationsbuskabels und des Peripheriegeräts für ein Basic Input/Output System („BIOS“) und das Senden einer Benachrichtigung in Reaktion darauf, dass die bestimmte Kabelverbindungskonfiguration nicht mit einer vordefinierten Kabelverbindungskonfiguration übereinstimmt.

**[0012]** Bei einer Ausführungsform ist/sind das eine oder die mehreren Peripheriegeräte jeweils in einen Laufwerksschacht einer Backplane installiert, beinhaltet die Kennung eine Schachtkennung und eine assoziierte Slotkennung des Laufwerksschachts und beinhaltet das Datenpaket mehrere Felder, welche die Kennung definieren. Die mehreren Felder können ein Feld für die Schachtkennung und ein Feld für die Slotkennung beinhalten.

**[0013]** Bei bestimmten Ausführungsformen beinhaltet das Verfahren das dynamische Modifizieren einer oder mehrerer Leistungseigenschaften eines Field Programmable Gate Array („FPGA“), basierend auf einer oder mehreren Eigenschaften eines Storage Enclosure Processors („SEP“) einer Backplane. Das FPGA kann mit Hilfe eines Kommunikationsbuskabels kommunikationstechnisch mit dem SEP und einem oder mehreren Peripheriegeräten der Backplane gekoppelt sein. Das Datenpaket kann ferner ein oder mehrere Felder beinhalten, die Daten beinhalten, welche die eine oder die mehreren Eigenschaften des SEPs beschreiben.

**[0014]** Bei einigen Ausführungsformen beinhaltet das Verfahren ferner das Empfangen eines Befehls für einen Virtual Pin Port („VPP“) von einem VPP-aktivierten Prozessor. Der VPP-Befehl kann für ein Peripheriegerät auf einer Backplane vorgesehen sein, welches durch die Installationspositionskennung identifiziert wird. Das Verfahren kann außerdem das Schreiben des VPP-Befehls in ein entsprechendes Feld im Datenpaket und das Senden des Datenpakets, das den VPP-Befehl beinhaltet, an einen Storage Enclosure Processor („SEP“) auf der Backplane mit Hilfe eines Kommunikationsbuskabels, das mit dem SEP gekoppelt und mit dem vorgesehenen Peripheriegerät assoziiert ist, beinhalten.

**[0015]** Ein Programmprodukt beinhaltet ein computerlesbares Speichermedium, das Code speichert, der von einem Prozessor ausgeführt werden kann. Der ausführbare Code beinhaltet Code zum Ausführen eines Bestimmens eines oder mehrerer Kommunikationsbuskabel, die ein oder mehrere Peripheriegeräte kommunikationstechnisch mit einer Informationshandhabungsvorrichtung koppeln. Der ausführbare Code beinhaltet Code zum Ausführen eines Lesens eines Datenpakets, das mit einem Peripheriegerät assoziiert ist, über jedes des einen oder der mehreren Kommunikationsbuskabel. Jedes Datenpaket beinhaltet eine Kennung für eine Position, an der das Peripheriegerät installiert ist. Der ausführbare Code beinhaltet Code zum Ausführen eines Bestimmens einer Kabelverbindungskonfiguration für das eine oder die mehreren Kommunikationsbuskabel und das eine oder die mehreren Peripheriegeräte, basierend auf der Installationspositionskennung, die aus jedem der Datenpakete empfangen wurde, die mit dem einen oder den mehreren Peripheriegeräten assoziiert sind.

## KURZE BESCHREIBUNG DER ZEICHNUNGEN

**[0016]** Zu den vorstehend kurz beschriebenen Ausführungsformen wird in Bezug auf spezielle Ausführungsformen, die in den beigefügten Zeichnungen dargestellt sind, eine genauere Beschreibung gegeben. In dem Verständnis, dass diese Zeichnungen nur einige Ausführungsformen darstellen und somit nicht als den Geltungsbereich einschränkend zu betrachten sind, werden die Ausführungsformen anhand der dazugehörigen Zeichnungen mit weiterer Spezifität und weiteren Details beschrieben und erläutert. Es zeigen:

**[0017]** Fig. 1A ein schematisches Blockdiagramm, das eine Ausführungsform eines Systems zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls veranschaulicht;

**[0018]** Fig. 1B ein schematisches Blockdiagramm, das eine Ausführungsform eines weiteren Systems zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls veranschaulicht;

**[0019]** Fig. 1C ein schematisches Blockdiagramm, das eine Ausführungsform eines weiteren Systems zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls veranschaulicht,

**[0020]** Fig. 2 ein schematisches Blockdiagramm, das eine Ausführungsform einer Vorrichtung zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls veranschaulicht;

**[0021]** Fig. 3 ein schematisches Blockdiagramm, das eine Ausführungsform einer weiteren Vorrichtung zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls veranschaulicht;

**[0022]** Fig. 4 ein schematisches Flussdiagramm, das eine Ausführungsform eines Verfahrens zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls veranschaulicht; und

**[0023]** Fig. 5 ein schematisches Flussdiagramm, das eine Ausführungsform eines weiteren Verfahrens zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls veranschaulicht.

## DETAILLIERTE BESCHREIBUNG

**[0024]** Wie Fachleute verstehen werden, können Aspekte der Ausführungsformen als ein System, Verfahren oder Programmprodukt umgesetzt werden. Dementsprechend können Ausführungsformen die Form einer komplett aus Hardware bestehenden Ausführungsform, einer komplett aus Software bestehenden Ausführungsform (einschließlich Firmware, residenter Software, Mikrocode usw.) oder einer Ausführungsform, die Soft- und Hardware-Aspekte kombiniert, auf die hierin alle allgemein als „Schaltung“, „Modul“ oder „System“ bezuggenommen wird, annehmen. Darüber hinaus können Ausführungsformen die Form eines Programmprodukts annehmen, das in einem oder mehreren computerlesbaren Speichergeräten umgesetzt ist, die maschinenlesbaren Code, computerlesbaren Code und/oder Programmcode, im Weiteren als Code bezeichnet, speichern. Die Speichergeräte können materiell, nicht-flüchtig und/oder nicht übertragend sein. Die Speichergeräte verkörpern möglicherweise keine Signale. Bei einer bestimmten Ausführungsform verwenden die Speichergeräte Signale lediglich für den Zugriff auf Code.

**[0025]** Viele der in dieser Beschreibung beschriebenen funktionalen Einheiten wurden als Module gekennzeichnet, um ihre Implementierungsunabhängigkeit deutlicher hervorzuheben. Zum Beispiel kann ein Modul als eine Hardware-Schaltung implementiert sein, die gebräuchliche VLSI-Schaltungen oder Gatearrays, standardmäßige Halbleiter wie beispielsweise Logik-Chips, Transistoren oder andere einzelne Komponenten, umfassen. Ein Modul kann auch in programmierbaren Hardwaregeräten implementiert sein, wie beispielsweise in Field Programmable Gatearrays, programmierbarer Arraylogik, programmierbaren Logikgeräten oder dergleichen.

**[0026]** Module können auch in Code und/oder Software zur Ausführung durch verschiedene Arten von Prozessoren implementiert sein. Ein identifiziertes Modul aus Code kann zum Beispiel einen oder mehrere physische oder logische Blöcke von ausführbaren Codes umfassen, die zum Beispiel als ein Objekt, ein Ablauf oder eine Funktion organisiert sein können. Dennoch müssen die ausführbaren Befehle eines identifizierten Moduls nicht physisch zusammen angeordnet sein, sondern können ganz verschiedene Befehle umfassen, die an verschiedenen Positionen gespeichert sind und die, wenn sie logisch zusammengefügt werden, das Modul ausmachen und den angegebenen Zweck des Moduls erfüllen.

**[0027]** Tatsächlich kann ein Modul aus Code eine einzelne Anweisung oder viele Anweisungen umfassen und kann sogar über mehrere verschiedene Code-Segmente, unter verschiedenen Programmen und über mehrere Speichergeräte verteilt sein. Ebenso können Betriebsdaten hierin in Modulen identifiziert und verteilt sein und können diese in jeder geeigneten Form umgesetzt und in jeder geeigneten Art Datenstruktur organisiert sein. Die Betriebsdaten können als ein einzelner Datensatz zusammengefasst sein oder können über verschiedene Positionen verteilt sein, einschließlich über verschiedene computerlesbare Speichergeräte. Wo ein Modul oder Teile eines Moduls in Software implementiert sind, sind die Software-Teile auf einem oder mehreren computerlesbaren Speichergeräten gespeichert.

**[0028]** Es kann jede Kombination aus einem oder mehreren computerlesbaren Medien verwendet werden. Das computerlesbare Medium kann ein computerlesbares Speichermedium sein. Das computerlesbare Speichermedium kann ein Speichergerät sein, das den Code speichert. Das Speichergerät kann zum Beispiel ein elektronisches, magnetisches, optisches, elektromagnetisches, Infrarot-, holografisches, mikromechanisches oder Halbleiter- System, -Vorrichtung, oder -Gerät, oder jede geeignete Kombination aus dem Vorstehenden sein, ohne darauf beschränkt zu sein.

**[0029]** Spezifischere Beispiele (eine nicht erschöpfende Liste) des Speichergeräts würden Folgendes beinhalten: eine elektrische Verbindung, die einen oder mehrere Drähte umfasst, eine tragbare Computerdiskette, eine Festplatte, einen Arbeitsspeicher (RAM), einen Nur-Lese-Speicher (ROM), einen löschbaren programmierbaren Nur-Lese-Speicher (EPROM oder Flash-Speicher), einen tragbare Compact Disk Nur-Lese-Speicher (CD-ROM), ein optisches Speichergerät, ein magnetisches Speichergerät, oder jede geeignete Kombination des Vorstehenden. Im Zusammenhang mit diesem Dokument kann ein computerlesbares Speichermedium jedes materielle Medium sein, das ein Programm zur Verwendung durch oder in Verbindung mit einem Anweisungsausführungssystem, einer derartigen Vorrichtung, oder einem derartigen Gerät enthalten oder speichern kann.

**[0030]** Code zum Ausführen von Operationen für Ausführungsformen kann in jeder Kombination aus einer oder mehreren Programmiersprachen geschrieben sein, einschließlich einer objektorientierten Programmiersprache wie beispielsweise Python, Ruby, Java, Smalltalk, C++ oder dergleichen, und konventioneller prozeduraler Programmiersprachen, wie beispielsweise der „C“-Programmiersprache oder dergleichen, und/oder Maschinensprachen, wie beispielsweise Assemblersprachen. Der Code kann gänzlich auf dem Computer des Benutzers ausgeführt werden, teilweise auf dem Computer des Benutzers, als eigenständiges Software-Paket, teilweise auf dem Computer des Benutzers und teilweise auf einem Remote-Computer oder gänzlich auf dem Remote-Computer oder Server. In letzterem Fall kann der Remote-Computer durch jede Art von Netzwerk mit dem Computer des Benutzers verbunden sein, einschließlich eines lokalen Netzwerks (LAN) oder eines Weitverkehrsnetzes (WAN,), oder kann die Verbindung mit einem externen Computer hergestellt werden (zum Beispiel über das Internet unter Verwenden eines Internet Service Providers).

**[0031]** Die Bezugnahme auf „eine Ausführungsform“ oder ein ähnlicher Wortlaut in der gesamten Beschreibung bedeutet, dass ein spezielles Merkmal, eine spezielle Struktur oder Eigenschaft, die im Zusammenhang mit der Ausführungsform beschrieben wird, in mindestens einer Ausführungsform enthalten ist. Folglich kann sich das Vorkommen der Formulierung „in einer Ausführungsform“ und ein ähnlicher Wortlaut in der gesamten Beschreibung auf die gleiche Ausführungsform beziehen, muss es aber nicht notwendigerweise, sondern bedeutet „eine oder mehrere, aber nicht alle Ausführungsformen“, sofern nichts anderes angegeben ist. Die Begriffe „beinhaltend“, „umfassend“, „aufweisend“ und Varianten davon bedeuten „beinhaltend, aber nicht darauf beschränkt“, sofern nichts anderes angegeben ist. Eine Aufzählung von Elementen impliziert nicht, dass sich einige oder alle der Elemente gegenseitig ausschließen, sofern nichts anderes angegeben ist. Die Begriffe „ein“ und „der, die oder das“ beziehen sich auch auf „ein oder mehrere“, sofern nichts anderes angegeben ist.

**[0032]** Darüber hinaus können die beschriebenen Merkmale, Strukturen oder Eigenschaften der Ausführungsformen in jeder geeigneten Weise kombiniert werden. In der folgenden Beschreibung sind zahlreiche spezifische Details bereitgestellt, wie etwa Beispiele für das Programmieren, für Software-Module, Benutzerauswahlen, Netzwerktransaktionen, Datenbankabfragen, Datenbankstrukturen, Hardware-Module, Hardware-Schaltungen, Hardware-Chips usw., um ein gründliches Verständnis der Ausführungsformen bereitzustellen. Fachleute werden jedoch erkennen, dass Ausführungsformen ohne ein oder mehrere der spezifischen Details oder mit anderen Verfahren, Komponenten, Materialien usw. umgesetzt werden können. In andern Fällen sind allgemein bekannte Strukturen, Materialien oder Operationen nicht im Detail gezeigt oder beschrieben, um eine Verschleierung von Aspekten einer Ausführungsform zu vermeiden.

**[0033]** Aspekte der Ausführungsformen sind im Weiteren mit Bezug auf schematische Ablaufdiagramme und/oder schematische Blockdiagramme von Verfahren, Vorrichtungen, Systemen und Programmprodukten gemäß den Ausführungsformen beschrieben. Es versteht sich, dass jeder Block des schematischen Ablaufdiagramms und/oder des schematischen Blockdiagramms und Kombinationen von Blöcken in den schematischen Ablaufdiagrammen und/oder schematischen Blockdiagrammen durch Code implementiert werden können. Dieser Code kann einem Prozessor eines Allzweckcomputers, eines Spezialcomputers oder einer anderen programmierbaren Datenverarbeitungsvorrichtung bereitgestellt werden, um eine Maschine zu erzeugen, so dass die Anweisungen, die über den Prozessor des Computers oder der anderen programmierbaren Datenverarbeitungsvorrichtung ausgeführt werden, Mittel zum Implementieren der Funktionen/Vorgänge erzeugen, die in dem Block/den Blöcken der schematischen Ablaufdiagramme und/oder schematischen Blockdiagramme spezifiziert sind.

**[0034]** Der Code kann außerdem auf einem Speichergerät gespeichert sein, das einen Computer, andere programmierbare Datenverarbeitungsvorrichtungen oder andere Geräte anleiten kann, in einer bestimmten Weise zu funktionieren, so dass die in der Speichervorrichtung gespeicherten Anweisungen einen Herstellungsgegenstand erzeugen, einschließlich Anweisungen, welche die Funktion/den Vorgang implementieren, die/der in dem Block/den Blöcken der schematischen Ablaufdiagramme und/oder der schematischen Blockdiagramme spezifiziert ist.

**[0035]** Der Code kann außerdem auf einen Computer, andere programmierbare Datenverarbeitungsvorrichtungen oder andere Geräte geladen werden, um zu bewirken, dass eine Reihe von operativen Schritten auf dem Computer, der anderen programmierbaren Datenverarbeitungsvorrichtung oder anderen Geräten ausgeführt wird, um einen computerimplementierten Prozess zu erzeugen, so dass der Code, der auf dem Computer oder anderen programmierbaren Datenverarbeitungsvorrichtungen ausgeführt wird, Prozesse zum Implementieren der Funktionen/Vorgänge bereitstellt, die in dem Block/den Blöcken der schematischen Ablaufdiagramme und/oder der schematischen Blockdiagramme spezifiziert sind.

**[0036]** Die schematischen Ablaufdiagramme und/oder schematischen Blockdiagramme in den Figuren veranschaulichen die Architektur, Funktionalität und Operation möglicher Implementierungen von Vorrichtungen, Systemen, Verfahren und Programmprodukten gemäß verschiedenen Ausführungsformen. In dieser Hinsicht kann jeder Block in den schematischen Ablaufdiagrammen und/oder schematischen Blockdiagrammen ein Modul, Segment oder einen Teil von Code darstellen, das/der eine oder mehrere ausführbare Anweisungen des Codes zum Implementieren der spezifizierten logischen Funktion(en) umfasst.

**[0037]** Es sei außerdem angemerkt, dass bei einigen alternativen Implementierungen die in den Blöcken angegebenen Funktionen nicht in der in den Figuren angegebenen Reihenfolgen auftreten müssen. Zum Beispiel können zwei Blöcke, die aufeinanderfolgend gezeigt sind, tatsächlich im Wesentlichen gleichzeitig ausgeführt werden oder können die Blöcke, abhängig von der damit einhergehenden Funktionalität, manchmal in umgekehrter Reihenfolge ausgeführt werden. Es können andere Schritte und Verfahren konzipiert werden, die in Funktion, Logik oder Wirkung äquivalent zu einem oder mehreren der in den Figuren dargestellten Blöcke oder Teilen davon sind.

**[0038]** Obwohl in den Ablauf- und/oder Blockdiagrammen verschiedene Pfeil- und Linienarten verwendet worden sind, sollen diese den Geltungsbereich der entsprechenden Ausführungsformen nicht einschränken. Tatsächlich können einige Pfeile oder andere Verbindungselemente nur dazu verwendet worden sein, um den logischen Ablauf der abgebildeten Ausführungsform anzuzeigen. Zum Beispiel kann ein Pfeil eine Wartezeit oder Überwachungsperiode unbestimmter Dauer zwischen aufgezählten Schritten der abgebildeten Ausführungsform anzeigen. Es sei außerdem angemerkt, dass jeder Block der Blockdiagramme und/oder Ablaufdiagramme und Kombinationen aus den Blöcken der Blockdiagramme und/oder Ablaufdiagramme durch Systeme auf Spezial-Hardware-Basis implementiert werden können, welche die spezifizierten Funktionen oder Vorgänge ausführen, oder durch Kombinationen von Spezial-Hardware und -Code.

**[0039]** Die Beschreibung von Elementen in jeder Figur kann sich auf Elemente vorstehender Figuren beziehen. In allen Figuren bezeichnen gleiche Bezugszeichen gleiche Elemente, einschließlich alternativer Ausführungsformen von gleichen Elementen.

**[0040]** Fig. 1A stellt eine Ausführungsform eines Systems **100** zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls dar. Bei einer Ausführungsform beinhaltet das System **100** eines Informationshandhabungsgerätes **102**. Das Informationshandhabungsgerät **102** kann einen Desktop-Computer, einen Laptop, ein Tablet, ein Smartphone, eine Set-Top-Box, eine Spielkonsole, ein Smart-TV, eine Smart-Watch, ein Fit-

nessarmband oder ein anderes am Körper tragbares Aktivitätsverfolgungsgerät, eine optisches Head-Mounted Display (z. B. ein Virtual Reality Headset, eine Datenbrille oder dergleichen), einen Personal Digital Assistant oder ein anderes Computergerät beinhalten, das einen Prozessor **104**, einen flüchtigen Speicher **108** und/oder ein nicht-flüchtiges Speichermedium beinhaltet, welches als mindestens eines der Peripheriegeräte **114a–n** umgesetzt sein kann.

**[0041]** Der Prozessor **104** kann eine oder mehrere zentrale Verarbeitungseinheiten (CPU), einen oder mehrere Prozessorkerne, ein Field Programmable Gate Array (FPGA) oder eine andere programmierbare Logik, eine anwendungsspezifische integrierte Schaltung (ASIC), einen Controller, einen Mikrocontroller und/oder ein anderes Gerät mit integrierter Halbleiterschaltung beinhalten. Der Prozessor **104** kann Firmware beinhalten, die verwendet wird, um während des Bootvorgangs eine Hardware-Initialisierung auszuführen, wie beispielsweise von einem Basic Input/Output System (BIOS) **106**. Das BIOS **106** kann bei einer Ausführungsform ein Exemplar der Unified Extensible Firmware Interface (UEFI) – Firmware beinhalten.

**[0042]** Das System **100** beinhaltet bei bestimmten Ausführungsformen ein oder mehrere Peripheriegeräte **114a–n** (zusammen **114**). Die Peripheriegeräte **114** können, wie hierin verwendet, interne und/oder externe Geräte beinhalten, die eine Ein- und/oder Ausgabe für das Informationshandhabungsgerät **102** bereitstellen. Zum Beispiel können die Peripheriegeräte **114** als Speichergeräte (z. B. nicht-flüchtige Speichergeräte (NVM), NVMe-express (NVMe)-Geräte, Festspeichergeräte (SSSD) und/oder dergleichen), Flash-Speichergeräte, Netzwerkkarten, Grafikkarten und/oder dergleichen umgesetzt sein.

**[0043]** Bei einer Ausführungsform sind die Peripheriegeräte **114** über eine Backplane **122** mit dem Informationshandhabungsgerät **102** verbunden. Wie hierin verwendet, kann die Backplane **122** als eine Leiterplatte (PCB) umgesetzt sein, die mehrere Slots **117a–n** (zusammen **117**), Anschlüsse und/oder dergleichen beinhaltet, um mehrere Peripheriegeräte **114** oder andere Leiterplatten mit der Backplane **122** zu verbinden. Die Backplane **122** kann zum Beispiel mehrere Slots **117**, Kabelverbindungen (z. B. Datenbus- **116** und/oder Management-Bus-Kabelverbindungen **118**) und/oder andere Arten von Verbindungselementen zum Verbinden der NVMe-Speichergeräte mit dem Informationshandhabungsgerät **102** beinhalten. Wie in **Fig. 1A** dargestellt, kann die Backplane **122** in einem separaten Gerät oder Gehäuse eingeschlossen sein, wie beispielsweise einem separaten Speichergehäuse, z. B. einem Speicher-Rack, das mit Hilfe eines oder mehrerer Kommunikationsbuskabel **123a–n** (zusammen **123**) mit dem Informationshandhabungsgerät **102** gekoppelt ist. Bei einigen Ausführungsformen ist die Backplane **122** in das Informationshandhabungsgerät **102** eingeschlossen.

**[0044]** Wie hierin verwendet, kann ein Slot **117** auf der Backplane **122** ein Zugangspunkt, Anschluss, Slot oder dergleichen zum Verbinden eines Kommunikationsbuskabels **123** von dem Informationshandhabungsgerät **102** sein. Der Slot **117** kann für verschiedene Arten von Kommunikationsbuskabeln **123** konfiguriert sein, wie beispielsweise PCIe-Kommunikationsbuskabel **123**. Mit Hilfe einer Slotkennung können die Slots **117** identifiziert werden, kann auf diese zugegriffen, verwiesen oder dergleichen werden. Die Slotkennung kann weltweit einzigartig für einen Slot **117**, einzigartig für einen Slot **117** auf der Basis einer Backplane **122** oder dergleichen sein. Die Slotkennung kann zumindest teilweise verwendet werden, um auf ein Peripheriegerät **114**, das kommunikationstechnisch mit dem Kommunikationsbuskabel **123** verbunden ist, das mit dem Slot **117** verbunden ist, zuzugreifen.

**[0045]** Bei einigen Ausführungsformen beinhaltet die Backplane **122** Laufwerksschächte **121a–n** (zusammen **121**), in denen Peripheriegeräte **114** installiert, befestigt, montiert und/oder dergleichen werden können. Zum Beispiel kann ein Laufwerksschacht **121** für die Montage eines NVMe- Speichergeräts an die Backplane **122** konfiguriert sein. Bei einigen Ausführungsformen ist jeder Schacht **121** mit einer Kennung identifiziert, die einzigartig für die Backplane **122**, global einzigartig und/oder dergleichen sein kann. Wenn die Schachtkennung für die Backplane **122** einzigartig ist, kann bei einer Ausführungsform für den Zugriff auf ein Peripheriegerät **114**, das in einem Schacht **121** installiert ist, auch eine Kennung der Backplane **122** notwendig sein, um sicherzustellen, dass auf das korrekte Peripheriegerät **114** auf der korrekten Backplane **122** zugegriffen wird. Bei einigen Ausführungsformen werden die Slotkennung und die Schachtkennung verwendet, um auf ein Peripheriegerät **114** zuzugreifen, welches in dem Schacht **121** installiert ist, der durch die Schachtkennung identifiziert ist, und kommunikationstechnisch mit dem Kommunikationsbuskabel **123** gekoppelt ist, das mit dem Slot **117** verbunden ist, der durch die Slotkennung identifiziert ist.

**[0046]** Bei bestimmten Ausführungsformen sind die Peripheriegeräte **114** mit Hilfe eines oder mehrerer Kommunikationsbuskabel **123** kommunikationstechnisch mit dem Prozessor **104** verbunden. Die Kommunikationsbuskabel **123** können ein Datenbuskabel beinhalten, das einem Datenbus **116a–n** (zusammen **116**) auf der Backplane **122** entspricht und das als ein Kabel, eine Datenleitung einer PCB der Backplane **122** und/oder

dergleichen umgesetzt sein kann. Das Datenbuskabel und der entsprechende Datenbus **116** auf der Backplane **122** können verwendet werden, um Ein-/Ausgabeoperationen (z. B. das Übermitteln von Daten) zwischen dem Prozessor **104** und den Peripheriegeräten **114** auszuführen. Bei bestimmten Ausführungsformen ist das Datenbuskabel und/oder der Datenbus **116** ein Peripheriekomponenten-Verbindungsbus (PCI-Bus), ein PCI Extended („PCI-X“- Bus, ein Accelerated Graphics Port („AGP“) Bus, ein PCI Express („PCIe“) Bus, ein Universal Serial Bus („USB“), ein Serial Advanced Technology Attachment („SATA“) Bus und/oder dergleichen.

**[0047]** Bei einigen Ausführungsformen ist jedes Peripheriegerät **114** mit Hilfe eines separaten Datenbuskabels und/oder Datenbusses **116** mit dem Prozessor **104** gekoppelt. Zum Beispiel kann zwischen den Peripheriegeräten **114** und den Datenbuskabeln und/oder den Datenbussen **116** ein 1:1-Verhältnis bestehen, so dass jedes Peripheriegerät **114** mit Hilfe eines separaten und unabhängigen Datenbuskabels und/oder Datenbusses **116** mit dem Informationshandhabungsgerät **102** gekoppelt ist.

**[0048]** Bei einer Ausführungsform beinhaltet das Kommunikationsbuskabel **123** ein Management-Bus-Kabel, das einem Management-Bus **118** auf der Backplane **122** entspricht und das als ein Kabel, als eine Datenleitung einer PCB der Backplane **122** und/oder dergleichen umgesetzt sein kann. Bei einer Ausführungsform ist ein Management-Bus **118** dafür konfiguriert, Verwaltungsnachrichten vom Prozessor **104** über ein Management-Bus-Kabel zu den Peripheriegeräten **114** zu übermitteln und umgekehrt. Ein Management-Bus **118** kann als ein Inter-Integrated-Circuit („I<sup>2</sup>C“) Kommunikationsbus umgesetzt sein. Verwaltungsnachrichten können Nachrichten beinhalten, die einen Zustand des Peripheriegeräts **114** steuern, wie beispielsweise Strom ein/aus, Aktivieren/Deaktivieren, Bereitschaft, Zurücksetzen und/oder dergleichen. Zum Beispiel kann ein Betriebssystem, das auf dem Informationshandhabungsgerät **102** läuft, über ein Management-Bus-Kabel und einen Management-Bus **118** eine Verwaltungsnachricht senden, ein Speichergerät, das über ein PCIe-Kabel mit dem Informationshandhabungsgerät **102** verbunden ist.

**[0049]** Bei einigen Ausführungsformen ist jedes Management-Bus-Kabel und/oder jeder Management-Bus **118** kommunikationstechnisch mit einem einzelnen entsprechenden Peripheriegerät **114** verbunden. Bei einer derartigen Ausführungsform kann zwischen Management-Bus-Kabeln und/oder Management-Bussen **118** und Peripheriegeräten **114** ein 1:1-Verhältnis bestehen, so dass jedes Peripheriegerät **114** mit Hilfe eines separaten und unabhängigen Management-Bus-Kabels mit dem Informationshandhabungsgerät **102** verbunden ist. Bezugnehmend auf **Fig. 1A** kann jede Management-Bus-Verbindung **124a–n** (zusammen **124**) für die Peripheriegeräte **114** mit einem separaten und eigenen Management-Bus **118** assoziiert sein, der einem separaten Management-Bus-Kabel entspricht. Zum Beispiel kann jedes Peripheriegerät **114** mit einem entsprechenden Management-Bus-Kabel gekoppelt sein, statt ein einziges Management-Bus-Kabel zu verwenden, um mehrere Peripheriegeräte **114** zu verwalten.

**[0050]** Bei einer weiteren Ausführungsform kann ein 1:n- oder ein 1:vielen-Verhältnis zwischen einem Management-Bus-Kabel, einem Management-Bus **118** und mehreren Peripheriegeräten **114** bestehen. Bei einer derartigen Ausführungsform kann ein Management-Bus **118** mehrere Management-Bus-Verbindungen **124** beinhalten, die jeweils mit einem Peripheriegerät **114** verbunden werden können und die mit Hilfe einer Kennung, einer einzigartigen Adresse (z. B. einer Adresse für die Management-Bus-Verbindung **124**, einer Adresse für ein Peripheriegerät **114** oder dergleichen) und/oder dergleichen adressierbar sind.

**[0051]** Bei einer Ausführungsform ist ein Management-Bus **118** mit einem Storage Enclosure Processor („SEP“) **115** der Backplane **122** verbunden. Wie hierin verwendet, kann ein SEP **115** ein Prozessor sein, der sich auf der Backplane **122** befindet, um Peripheriegeräte **114** auf der Backplane **122** zu verwalten. Zum Beispiel kann der SEP **115** die Management-Bus-Verbindungen **124** für den Management-Bus **118** verwalten, um mit Hilfe eines Management-Bus-Kabels und basierend auf einer einzigartigen Adresse für die Management-Bus-Verbindung **124** und/oder das Peripheriegerät **114** Verwaltungsnachrichten zwischen dem Informationshandhabungsgerät **102** und den Peripheriegeräten **114** auf der Backplane **122** zu übermitteln.

**[0052]** Zum Beispiel kann ein Betriebssystem für das Informationshandhabungsgerät **102** einen Zurücksetz-Befehl an ein Peripheriegerät **114** senden, indem es eine Kennung oder Adresse für das Peripheriegerät **114** und/oder die Management-Bus-Verbindung **124** des Management-Busses **118** bereitstellt. Bei einem derartigen Beispiel kann der SEP **115** die bereitgestellte Kennung oder Adresse verwenden, um den Zurücksetz-Befehl an das Peripheriegerät **114** zu senden, das mit der Kennung oder der Adresse assoziiert ist. Bei einem weiteren Beispiel kann der SEP **115** die bereitgestellte Kennung oder Adresse in eine andere Kennung oder Adresse übertragen, abbilden oder auf andere Art querverweisen, die verwendet wird, um mit dem Peripheriegerät **114** auf der Backplane **122** zu kommunizieren.

**[0053]** Wie hierin verwendet, ist eine Verwaltungsverbindung zwischen dem Prozessor **104**, einer Datenprotokollvorrichtung **110** oder dergleichen und einem Peripheriegerät **114** als eine Management-Bus-Verbindung **124** bekannt, ob nun mit jedem Peripheriegerät **114** ein einzelnes Management-Bus-Kabel und/oder ein einzelner Management-Bus **118** verbunden ist (z. B. eine Management-Bus-Verbindung **124** pro Management-Bus-Kabel/Management-Bus **118**) oder ob ein einzelnes Management-Bus-Kabel und/oder ein einzelner Management-Bus **118** mit mehreren Peripheriegeräten **114** verbunden ist (z. B. mehrere adressierbare Management-Bus-Verbindungen **124** pro Management-Bus-Kabel/Management-Bus **118**).

**[0054]** Der Baseboard-Management-Controller (BMC) **113** ist in einer Ausführungsform eine spezielle Hardware-Komponente des Informationshandhabungsgerätes **102**, die verwendet wird, um mit Hilfe eines System-Management-Busses **119** die Schnittstelle zwischen der Systemverwaltungssoftware, wie beispielsweise dem BIOS **106**, einem Betriebssystem oder anderer Verwaltungssoftware, und Plattform-Hardware, wie beispielsweise den Peripheriegeräten **114**, zu verwalten. Zum Beispiel kann der BMC **113** das Übermitteln von Verwaltungsnachrichten zwischen dem Prozessor, dem BIOS **106**, der Kabelverwaltungsvorrichtung **110**, einem Betriebssystem und/oder dergleichen und den Peripheriegeräten **114**, dem SEP **115** und/oder dergleichen verwalten.

**[0055]** Die Brücke oder der Schalter **112** verbindet bei einer Ausführungsform die Peripheriegeräte **114** über die Kommunikationsbuskabel **123** mit dem Prozessor **104** und/oder der Kabelverwaltungsvorrichtung **110**. Die Brücke **112** kann als ein Schalter dienen und mit Hilfe der Kommunikationsbuskabel **123** (z. B. ein Datenbuskabel, ein Management-Bus-Kabel oder ein kombiniertes Datenbus-/Management-Bus-Kabel) Daten und/oder Verwaltungsbefehle zwischen dem Prozessor **104** und den entsprechenden Peripheriegeräten **114** übertragen. Bei bestimmten Ausführungsformen können sich mehrere Brücken **112** in dem Informationshandhabungsgerät **102** befinden, wobei jede Brücke **112** mit einem oder mehreren Peripheriegeräten **114** einer oder mehrerer Backplanes **122** verbunden ist.

**[0056]** Die Datenprotokollvorrichtung **110** ist bei einer Ausführungsform dafür konfiguriert, eine oder mehrere Eigenschaften zu bestimmen, die mit einem Peripheriegerät **114**, einer Backplane **122**, einem SEP **115** und/oder dergleichen assoziiert ist, indem sie ein vordefiniertes Datenpaket liest, das ein oder mehrere Felder für die verschiedenen Eigenschaften beinhaltet. Die Datenprotokollvorrichtung **110** ist dafür konfiguriert, ein oder mehrere Kommunikationsbuskabel **123** zu bestimmen, die ein oder mehrere Peripheriegeräte **114** kommunikationstechnisch mit einem Informationshandhabungsgerät **102** koppeln, ein Datenpaket zu lesen, das über ein Kommunikationsbuskabel **123** mit einem Peripheriegerät **114** assoziiert ist, und basierend auf den Eigenschaften, die in dem Datenpaket enthalten sind, das über das Kommunikationsbuskabel **123** gelesen wird, eine Kabelverbindungsconfiguration für das eine oder die mehreren Kommunikationsbuskabel **123** und das eine oder die mehreren Peripheriegeräte **114** zu bestimmen. Bei einigen Ausführungsformen befindet sich mindestens ein Teil der Datenprotokollvorrichtung **110** auf dem Prozessor **104**, auf der Backplane **122**, auf dem BMC **113**, auf dem SEP **115** und/oder dergleichen. Die Datenprotokollvorrichtung **110** wird im Weiteren in Bezug auf **Fig. 2** und **Fig. 3** näher beschrieben.

**[0057]** Bei einer Ausführungsform ist das Informationshandhabungsgerät **102** mit einem Datennetzwerk **120** verbunden. Das Datennetzwerk **120** beinhaltet bei einer Ausführungsform ein digitales Kommunikationsnetzwerk, das digitale Kommunikationen übermittelt. Das Datennetzwerk **120** kann ein drahtloses Netzwerk beinhalten, wie beispielsweise ein Mobilfunknetz, ein lokales drahtloses Netzwerk, wie etwa ein Wi-Fi-Netzwerk, ein Bluetooth®-Netzwerk, ein Netzwerk für die Nahbereichskommunikation (NFC), ein Ad-hoc-Netzwerk und/oder dergleichen. Das Datennetzwerk **120** kann ein Weitverkehrsnetz (WAN), ein Speichernetzwerk (SAN), ein lokales Netzwerk (LAN), ein Glasfasernetz, das Internet oder ein anderes digitales Netzwerk sein. Das Datennetzwerk **120** kann zwei oder mehr Netzwerke beinhalten. Das Datennetzwerk **120** kann eine(n) oder mehrere Server, Router, Schaltungen und/oder andere Netzwerkausstattung beinhalten. Das Datennetzwerk **120** kann außerdem ein oder mehrere computerlesbare Speichermedien, wie beispielsweise ein Festplattenlaufwerk, ein optisches Laufwerk, einen nicht-flüchtigen Speicher, RAM oder dergleichen beinhalten.

**[0058]** Bei einer Ausführungsform ist das Informationshandhabungsgerät **102** über das Datennetzwerk **120** mit einem oder mehreren anderen Computergeräten **130/132** verbunden. Die anderen Computergeräte **130** können zum Beispiel Smartphones, Tablets, Laptops und/oder dergleichen beinhalten. Bei einer weiteren Ausführungsform können die anderen Computergeräte **132** Server, andere Geräte in einem Datenzentrum, das sich in einem lokalen oder externen Datennetzwerk **120** befindet, und/oder dergleichen beinhalten.

**[0059]** **Fig. 1B** stellt eine Ausführungsform eines „direkt angebrachten“ Systems **150** zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls dar. Bei einer Ausführungsform beinhaltet das System **150** Ele-

mente, die im Wesentlichen gleichen Elementen ähneln, die im Vorstehenden in Bezug auf **Fig. 1A** dargestellt und beschrieben wurden. Das System **150** von **Fig. 1B** beinhaltet bei einer Ausführungsform zwei separate Verarbeitungseinheiten **202a–b** (zusammen **202**), die jeweils kommunikationstechnisch mit Peripheriegeräten **114** auf entsprechenden Backplanes **122a–b** (zusammen **122**) gekoppelt sind. Jede Verarbeitungseinheit **202** kann einen oder mehrere Prozessoren **104**, Prozessorkerne und/oder dergleichen, Speicherelemente (z. B. Register) und/oder dergleichen beinhalten.

**[0060]** Jede Verarbeitungseinheit **202** kann mit Hilfe eines vorstehend beschriebenen Kommunikationsbuskabels **123** kommunikationstechnisch mit den Peripheriegeräten **114** auf jeder entsprechenden Backplane **122** und mit einem oder mehreren Datenbussen **216a–b** (zusammen **216**) auf dem Informationshandhabungsgerät **102**, z. B. auf dem Motherboard oder einer anderen PCB des Informationshandhabungsgerätes **102**, und mit einem oder mehreren Datenbussen **116** auf den Backplanes **122** gekoppelt sein. Eine Verarbeitungseinheit **202** kann mit Hilfe der Kommunikationsbusse **123** und/oder der Datenbusse **216**, **116** verschiedene Befehle, Daten, Anweisungen und/oder dergleichen an die Peripheriegeräte **114** senden. Zum Beispiel kann eine Verarbeitungseinheit **202** einen Leseanfragebefehl, einen Schreibanfragebefehl oder dergleichen an ein NVMe-Peripheriespeichergerät senden, das sich an der Slot-ID **10** und der Schacht-ID **62** auf der Backplane **122** befindet.

**[0061]** Bei einer weiteren Ausführungsform ist jede Verarbeitungseinheit **202** kommunikationstechnisch mit einem Field Programmable Gate Array (FPGA) **204** oder einem anderen programmierbaren Logikgerät, wie beispielsweise einer anwendungsspezifischen integrierten Schaltung (ASIC), oder dergleichen gekoppelt. Bei einer Ausführungsform ist eine Verarbeitungseinheit **202** mit Hilfe eines I<sup>2</sup>C-Kommunikationsbusses, eines SHP-Management-Busses und/oder einigen anderen seriellen Kommunikationsbussen kommunikationstechnisch mit dem FPGA **204** gekoppelt. Das FPGA **204** kann mit Hilfe eines oder mehrerer Kommunikationsbuskabel **123** und/oder eines oder mehrerer Management-Busse **218a–b** (zusammen **218**), die sich auf dem Informationshandhabungsgerät **102** befinden, und eines oder mehrerer Management-Busse **118**, die sich auf den Backplanes **122** befinden, kommunikationstechnisch mit einer Backplane **122** und insbesondere mit einem SEP **115** und/oder einem Peripheriegerät **114**, die sich auf der Backplane **122** befinden, gekoppelt sein.

**[0062]** Das FPGA **204** kann, wie hierin verwendet, Daten, Informationen, Befehle usw. zwischen den Verarbeitungseinheiten **202** und/oder der Datenprotokollvorrichtung **110** auf dem Informationshandhabungsgerät **102** und dem SEP **115** und/oder den Peripheriegeräten **114** auf der Backplane **122** kommunizieren, übermitteln, senden/empfangen oder dergleichen. Zum Beispiel kann das FPGA **204** mit Hilfe eines bestimmten Kommunikationsbuskabels **123** mit dem SEP **115** auf einer Backplane **122** kommunizieren, um ein Datenpaket zu lesen, das eine Slot-ID und eine Schacht-ID für das Peripheriegerät **114** beinhaltet, das mit dem Kommunikationsbuskabel **123** gekoppelt ist. Bei einer derartigen Ausführungsform kann das FPGA **204** als ein Bus-Master konfiguriert sein und kann jeder SEP **115** auf jeder Backplane **122** als Bus-Slave konfiguriert sein. Bei einer derartigen Master/Slave-Konfiguration leitet das FPGA **204**, das als der Bus-Master fungiert, Befehle, Datenpakete, Informationen oder dergleichen über die Kommunikationsbuskabel **123** an die SEPs **115**, die als Bus-Slaves dienen und für ein oder mehrere Peripheriegeräte **114** vorgesehen sind. Das FPGA **204** kann über verschiedene Kommunikationsbuskabel **123** auch Datenpakete von den SEPs **115** anfordern oder lesen. Bei einer derartigen Ausführungsform ist jedes Peripheriegerät **114** als ein Slave-Gerät konfiguriert.

**[0063]** Das FPGA **204** kann die Slot-ID- und Schacht-ID-Informationen, die mit jedem Kommunikationsbuskabel **123** und jedem mit dem Kommunikationsbuskabel **123** gekoppelten Peripheriegerät **114** assoziiert sind, nachverfolgen, speichern, erfassen und/oder dergleichen. Bei einer derartigen Ausführungsform kann das FPGA **204** die Slot-ID- und Schacht-ID-Informationen für jedes Peripheriegerät **114**, einschließlich der Kommunikationsbuskabel **123**, die mit jedem Peripheriegerät **114** gekoppelt sind, für ein Betriebssystem für das Informationshandhabungsgerät **102**, für das BIOS **106** und/oder dergleichen bereitstellen, dahin senden oder auf andere Weise verfügbar machen. Bei bestimmten Ausführungsformen beinhalten die Management-Busse **118**, **218**, die mit dem FPGA **204** und dem SEP **115** gekoppelt sind, SHP-Management-Busse.

**[0064]** Bei bestimmten Ausführungsformen beinhaltet das Informationshandhabungsgerät **102** oder das Motherboard des Informationshandhabungsgerätes **102** einen oder mehrere Slots **220a–n** (zusammen **220**), die dafür konfiguriert sind, ein Kommunikationsbuskabel **123** aufzunehmen, das mit einem entsprechenden Slot **117** auf der Backplane **122** gekoppelt ist. Der Slot **220** kann mit Hilfe einer Slot-ID, einer PCIe-Lane-Kennung und/oder dergleichen identifiziert sein.

**[0065]** Bei einer Ausführungsform ist jede Verarbeitungseinheit **202** kommunikationstechnisch mit einem Plattform-Controller-Hub (PCH) **206** gekoppelt, der dafür konfiguriert ist, Datenwege zu steuern, wie etwa die

Datenbusse **216**, und verschiedene Funktionen zu verwalten, deren Last von den Verarbeitungseinheiten **202** genommen wurde, wie beispielsweise Taktung. Bei einigen Ausführungsformen verwenden die Verarbeitungseinheiten **202** den PCH **206**, um verschiedene Daten und Verwaltungsbefehle über das FPGA **204** an die Peripheriegeräte **114** zu senden. Bei einer derartigen Ausführungsform kann der PCH **206** mit Hilfe eines Kommunikationsbusses **222**, der als ein Enhanced Serial Peripheral Interface („eSPI“) – Bus umgesetzt sein kann, mit dem FPGA **204** gekoppelt sein

**[0066]** Bei einigen Ausführungsformen ist auch die Datenprotokollvorrichtung **110** über einen Kommunikationsbus **212** mit dem FPGA **204** gekoppelt, um die verschiedenen Funktionen der Datenprotokollvorrichtung **110** auszuführen, wie im Weiteren näher beschrieben werden wird. Darüber hinaus kann außerdem der BMC **113** über einen Kommunikationsbus **214** mit dem FPGA **204** gekoppelt sein, um die Schnittstelle zwischen der Systemverwaltungssoftware, wie beispielsweise dem BIOS **106**, einem Betriebssystem oder einer anderen Verwaltungssoftware und den Peripheriegeräten **114**, dem SEP **115** und/oder dergleichen zu verwalten.

**[0067]** Fig. 1C stellt eine Ausführungsform eines „geschalteten“ Systems **175** zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls dar. Bei einer Ausführungsform beinhaltet das System **175** Elemente, die im Wesentlichen gleichen Elementen ähneln, die im Vorstehenden mit Bezug auf Fig. 1A und Fig. 1B dargestellt und beschrieben wurden. Bei einer Ausführungsform beinhaltet das System **175** mehrere Schaltadapter **302a–b** (zusammen **302**), die dem vorstehend mit Bezug auf Fig. 1A beschriebenen Schalter **112** ähneln.

**[0068]** Jeder Schaltadapter **302** kann dafür konfiguriert sein, jede Verarbeitungseinheit **202**, die Datenprotokollvorrichtung **110**, den BMC **113** oder dergleichen und jedes Peripheriegerät **114** einer Backplane **122** zu verwalten und zwischen diesen zu übertragen. Zum Beispiel kann eine Verarbeitungseinheit **202** Daten oder Verwaltungsbefehle für ein bestimmtes Peripheriegerät **114** über einen Kommunikationsbus **306a–b**, der mit einem Kommunikationsbusanschluss **304a–b** des Schaltadapters **302** gekoppelt ist, für einen gekoppelten Schaltadapter **302** bereitstellen. Der Schaltadapter **302** kann den Befehl an ein bestimmtes Peripheriegerät **114** weiterleiten oder übermitteln. Das Peripheriegerät **114** kann mit Hilfe einer einzigartigen Adresse, einer Slot-ID, einer Schacht-ID und/oder dergleichen identifiziert werden. Basierend auf der Kennung kann der Schaltadapter **302** bestimmen, welches Kommunikationsbuskabel **123** mit dem identifizierten Peripheriegerät **114** gekoppelt ist.

**[0069]** Fig. 2 stellt eine Ausführungsform einer Vorrichtung **200** zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls dar. Die Vorrichtung **200** beinhaltet eine Ausführungsform einer Datenprotokollvorrichtung **110**. Die Datenprotokollvorrichtung **110** beinhaltet ein Verbindungsmodul **230**, ein Datenmodul **235** und ein Topologiemodul **240**, die im Weiteren näher beschrieben werden.

**[0070]** Das Verbindungsmodul **230** bestimmt ein oder mehrere Kommunikationsbuskabel **123**, die ein oder mehrere Peripheriegeräte **114** kommunikationstechnisch mit einem Informationshandhabungsgerät **102** koppeln. Bei einer Ausführungsform kann das Verbindungsmodul **230** über jedes Kommunikationsbuskabel **123** einen Befehl senden, um zu bestimmen, welches Kommunikationsbuskabel **123** mit einem Peripheriegerät **114** gekoppelt ist. Bei einer weiteren Ausführungsform kann das BIOS **106** beim Systemstart eine Systemprüfung ausführen, um zu bestimmen, welche Peripheriegeräte **114** vorhanden und mit Hilfe eines Kommunikationsbuskabels **123** mit dem Informationshandhabungsgerät **102** gekoppelt sind. Das Verbindungsmodul **230** kann eine Tabelle für Peripheriegeräte **114** aufbauen, erzeugen, verwalten, führen und/oder dergleichen, die mit dem Informationshandhabungsgerät **102** und den assoziierten Kommunikationsbuskabeln **123** verbunden sind, welche mit den bestimmten Peripheriegeräten **114** gekoppelt sind.

**[0071]** Das Datenmodul **235** liest über jedes des einen oder der mehreren Kommunikationsbuskabel **123** ein Datenpaket, das mit einem Peripheriegerät **114** assoziiert ist. Das Datenpaket kann ein oder mehrere Datenfelder enthalten, die Daten, Informationen oder dergleichen für ein Peripheriegerät **114**, für eine Backplane **122**, für einen SEP **115** einer Backplane **122** und/oder dergleichen beinhalten. Die folgende Tabelle 1 stellt ein Beispiel für ein Datenpaket für ein Protokoll zum Verwalten von Peripheriegeräten **114** dar.

<u>NVMe-</u> <u>Regis-</u> <u>ter-</u> <u>Satz</u>	<u>Off-</u> <u>set</u>	<u>Bit</u>	<u>Definition</u>	<u>Details</u>	<u>I2C-</u> <u>Master</u> <u>(R/W)</u>
NVMe- Ge- räte- ID	0	7	I2C-Slave- Gerät nicht bereit	Auf ,1` gestellt, wenn I2C-Slave-Gerät kon- figuriert wird und nicht vollständig funktionsbereit ist, sonst auf ,0`. Master wird dieses Register bis ,0` pollen. Offset	Nur Le- sen

				1-9 ist bei Einstellung auf ,1` ungültig.	
		6	Fest codiert auf 4Eh	ASCII ,N` für NVMe. Achtung: Dieses Register ist ungültig, auch wenn „NVMe-Geräte-ID“-Bit 7 eine ,1` ist.	
		5			
		4			
		3			
		2			
		1			
		0			
Versionsregister	1	7	I2C-Slave-Gerät Version in BCD	Die Version sollte in binär codierten Dezimalstellen präsentiert sein. Zum Beispiel wird eine I2C-Slave-Geräte-Version zehn als 10h geschrieben.	Nur Lesen
		6			
		5			
		4			
		3			
		2			
		1			
		0			
Fähigkeitsregister 2	2	7	Reserviert	Reserviert als Fh	Nur Lesen
		6			
		5			
		4			
		3	I2C-Master-Lese-Zeitsperre	Definiert die Zeitspanne, die der NVMe-I2C-Master warten muss, bevor der Registersatz erneut gelesen wird. Codiert als 0h=0 s, 1h=50 ms, 2h=100 ms. Andere reserviert.	
		2			
		1			

			unterstützung	Datenparität im Register „Steuerungs-Control/Statusregister 0“ unterstützt und „Nutzlast-Prüfsumme“ gültig ist.	
		0	I2C Slave-Gerät-Busgeschwindigkeitsfähigkeit	Auf ,1` gestellt, wenn das I2C-Slave-Gerät 400 kHz unterstützt (bevorzugt), sonst ,0` für 100 kHz. Ein Wert von ,1` impliziert nicht, dass der Master bei 400 kHz laufen muss.	
Fähigkeitsregister 1	3	7	Reserviert	Reserviert als 1Fh	Nur Lesen
		6			
		5			
		4			
		3			
		2	Backplane-Art	Bezeichnet die Art der Backplane: 0h=nur NVMe, 1h = NVMe + SAS, 2h = nur SAS, andere reserviert	
		1			
		0			
Fähigkeitsregister 0	4	7	Gesamt-NVMe-Geräteunterstützung	Anzahl der NVMe Geräte, die vom I2C-Slave-Gerät unterstützt werden. Codiert als 1h=1 Laufwerk insgesamt, FFh=225 Laufwerke insgesamt. Dies wird	Nur Lesen
		6			
		5			
		4			
		3			
		2			
		1			
		0			

				basierend auf der Anzahl der I2C-Master berechnet, die das I2C-Slave-Gerät sieht.	
Steuerungs-/Statusregister 3	5	7	Slot-ID	Slot-ID des NVMe-Geräts. Da der NVMe tatsächlich ein PCIe-Slot ist, definiert dies die Slot-ID, die das Betriebssystem verwenden wird.	Nur Lesen
		6			
		5			
		4			
		3			
		2			
		1			
		0			
Steuerungs-/Statusregister 2	6	7	Schacht-ID	Schacht-ID des NVMe-Geräts. Es ist zu beachten, dass Schacht-ID die physische Position des Slots vorn im System ist und die gleiche wie die SAS-„Bay ID“ sein wird.	Nur Lesen
		6			
		5			
		4			
		3			
		2			
		1			
		0			
Steuerungs-/Statusregister 1	7	7	Reserviert	Reserviert als Fh	Nur Lesen
		6			
		5			
		4			
		3	IFDET_#	Zustand des IFDET-Signals. Für NVMe ist dies eine ,0`.	
		2	HP_PRESENT#	Zustand des Laufwerksanwesenheitssignals. Für NVMe ist dies eine ,1`.	

		1	Auto-Konfigurationsstatus	Auf ,1` gestellt, wenn BMC die Anschluss-(Port-)ID-Werte eingestellt hat, sonst ,0`.	
		0	HP_PWR_LED#	NVME-Power-LED (z. B. Grün) geringfügig tätig. Ein Lesen reflektiert den Zustand des Pins an dem I2C-Slave-Gerät. Standard ,1`.	
Steuerungs-/Statusregister 0	8	7	Datenparität	Wenn Parität nicht unterstützt wird (siehe „Fähigkeitsregister 2“ Bit 1), ist dieses Bit als ,1` reserviert. Zum Lesen des I2C-Masters reflektiert dieses Bit die SUMME der Bits 6 bis 0 des „Steuerungs-/Statusregisters 0“. Zum Schreiben des I2C-Masters deckt das durch den Master eingestellte Paritätsbit die Bits 6 bis 0 ab. Ist keine Paritätsübereinstimmung gegeben, werden die Daten aussortiert.	Lesen/ Schreiben
		6	Reserviert	Reserviert als 3h	Nur Lesen
		5			

		4	Geräte-Scratchpad	Scratch Pad des I2C-Masters. Standard ,0`.	
		3	HP_PERST#	NVMe-Geräte-Nullstellen aktiv ist niedrig. Ein Lesewert reflektiert den Wert der letzten I2C-Master-Schreibung auf diese Position. Standard ,1`.	
		2		Codiert: 0h=0 Hz (auf fest), 1h=1 Hz, 2h=2 Hz, 3h=4 Hz. I2C-Master muss HP_ATN_LED# zur Aktivierung der LED durchsetzen. Vorgaben auf ,0`.	Lesen/ Schreiben
		1	HP_ATN_LED Blink-Raten-Register		
		0	HP_ATN_LED#	NVMe-Achtung-LED (z. B. bernsteinfarben) aktiv niedrig. Ein Lesewert reflektiert den Wert der letzten I2C-Master - Schreibung auf diese Position. Standard ,1`.	
Nutzlast-Prüfsumme	9	7	Prüfsumme		Nur Lesen
		6	von Byte 0	Byte 0 + Byte 1 + ... + Byte 6 + Byte 7	
		5	bis Byte 7		
		4	(z. B. be-		
		3	inhaltet		

Tabelle 1.

	2	dies nicht	
	1	Byte 8)	
	0		

**[0072]** Wie in Tabelle 1 dargestellt, beinhaltet jedes Datenpaket ein oder mehrere Datenfelder, die Informationen beinhalten, welche eine Kennung für eine Position definieren, an der das Peripheriegerät **114** installiert ist. Zum Beispiel kann das Datenpaket ein Datenfeld für Daten beinhalten, die eine Schacht- ID identifizieren, welche einen Laufwerksschacht **121** identifiziert, wo ein Peripheriegerät **114** installiert ist, und ein Datenfeld für Daten, die einen Slot **117** identifizieren, der mit dem Kommunikationsbuskabel **123** verbunden ist, der kommunikationstechnisch mit dem Peripheriegerät **114** gekoppelt ist.

**[0073]** Bei einer weiteren Ausführungsform beinhaltet das Datenpaket ferner ein oder mehrere Felder, die Daten beinhalten, welche eine oder mehrere Eigenschaften eines SEP **115** einer Backplane **122** beschreiben. Bei einigen Ausführungsformen beinhaltet das Datenpaket einen Bitschalter, der anzeigt, ob der SEP **115** Datenparität unterstützt. Zum Beispiel kann das FPGA **204** Daten in ein oder mehrere beschreibbare Felder des Datenpakets schreiben, einschließlich eines Paritätsdatenfeldes, das Paritätsdaten für den SEP **115** beinhaltet, der verwendet werden soll, um die Integrität der geschriebenen Daten zu betätigen, wenn der SEP **115** Datenparität unterstützt, wie es durch den Paritätsbitschalter in dem Datenpaket angezeigt wird. Bei solch einem Beispiel kann das FPGA **204** zuerst ein Datenpaket für ein bestimmtes Peripheriegerät **114** von dem SEP **115** abrufen, das Paritätsdatenfeld lesen und ein Paritätsbit zusammen mit den in die anderen beschreibbaren Datenfelder geschriebenen Daten des Datenpakets einschließen, bevor das Datenpaket an den SEP **115** gesendet wird.

**[0074]** Bei einem weiteren Beispiel kann das Datenpaket ein Datenfeld beinhalten, das Daten beinhaltet, welche eine maximale, minimale, durchschnittliche, mediane oder dergleichen Taktgeschwindigkeit des SEPs **115** anzeigen. Bei einem weiteren Beispiel kann das Datenpaket ein Datenfeld beinhalten, das Daten beinhaltet, welche einen Zeitsperreparameter anzeigen, der eine Zeitspanne definiert, bevor das Datenmodul **235** in Reaktion auf ein Nicht-Empfangen einer Antwort von dem SEP **115** den SEP **115** erneut nach einem Datenpaket abfragt. Bei einem weiteren Beispiel kann das Datenpaket ein Datenfeld beinhalten, das eine maximale Anzahl an Peripheriegeräten **114** anzeigt, die von dem SEP **115**, der Backplane **122**, auf der der SEP **115** angeordnet ist, und/oder dergleichen unterstützt werden. Weitere im Datenpaket enthaltene Daten können eine Kennung für das Peripheriegerät **114**, die Backplane **122**, den SEP **115** und/oder dergleichen beinhalten, Versionsinformationen für das Peripheriegerät **114**, die Backplane **122**, den SEP **115** und/oder dergleichen, Zustandsinformationen für das Peripheriegerät **114** (z. B. aktiv, aktiviert, deaktiviert oder dergleichen), eine Prüfsumme für ein oder mehrere Datenfeldern, die verwendet werden kann, um die Integrität eines oder mehrerer Datenfelder zu bestätigen und/oder dergleichen. Bei bestimmten Ausführungsformen, die im Weiteren näher beschrieben werden, stimmt das FPGA **204** ein oder mehrere Leistungsmerkmale basierend auf den Eigenschaften des SEP **115**, der Peripheriegeräte **114**, der Backplane **122** und/oder dergleichen, wie sie aus den Daten bestimmt werden, die in dem empfangenen Datenpaket enthalten sind, ab.

**[0075]** Das Topologiemodul **240** bestimmt eine Kabelverbindungskonfiguration für das eine oder die mehreren Kommunikationsbuskabel **123** und das eine oder die mehreren Peripheriegeräte **114** basierend auf der Installationspositionskennung, die aus jedem der mit dem einen oder den mehreren Peripheriegeräten **114** assoziierten Datenpakete empfangen wurden. Das Topologiemodul **240** kann zum Beispiel eine Tabelle von Pfaden von Kommunikationsbuskabeln **123** zu den entsprechenden Installationspositionskennungen erzeugen, wie z. B. den Slot-IDs und den Schacht-IDs der Peripheriegeräte **114**, mit denen die Kommunikationsbuskabel **123** jeweils kommunikationstechnisch gekoppelt sind.

**[0076]** Bei einer Ausführungsform ist die Tabelle von Pfaden als eine Advanced-Configuration-and-Power-Interface (ACPI) Tabelle umgesetzt. Wie hierin verwendet, ist eine ACPI eine Art von Schnittstelle, die ein Betriebssystem in die Lage versetzt, Hardware-Komponenten direkt zu konfigurieren und zu verwalten. Bei einer derartigen Ausführungsform stellt das Topologiemodul **240** die Pfadetabelle bereit, welche die Kabelverbindungskonfiguration der Kommunikationsbuskabel **123** und der Peripheriegeräte **114** für das BIOS **106**, das Betriebssystem des Informationshandhabungsgerätes **102** und/oder dergleichen beschreibt. Bei einer weiteren Ausführungsform können das BIOS **106** und/oder das Betriebssystem die ACPI-Tabelle verwenden, um

einen PCIe-Anschluss an dem Informationshandhabungsgerät **102** (z. B. am Motherboard) und/oder an der Backplane **122** mit der Kennung, z. B. der Slot-ID für einen Slot **117**, der im Datenpaket empfangen wurde, zu programmieren, diese zuzuweisen oder auf andere Weise zuzuordnen.

**[0077]** Auf diese Weise stellt die Datenprotokollvorrichtung **110** ein Datenpaket oder ein Protokoll für das Senden und Empfangen von Daten bereit, die verschiedene Zustände, Eigenschaften, Status und/oder dergleichen für Peripheriegeräte **114**, SEPs **115**, Backplanes **122** und/oder dergleichen in einem System **100/150/175** beschreiben, das Kommunikationsbuskabel **123** verwendet, um verschiedene Peripheriegeräte **114** mit einem Informationshandhabungsgerät **102** zu verbinden, im Gegensatz zu direkten physischen Verbindungen, z. B. Kommunikationsbusleitungen auf einer Leiterplatte zwischen einem Prozessor **104** und einem PCIe-Slot, in den ein Peripheriegerät **114** eingesetzt ist. Dementsprechend können der Prozessor **104**, das FPGA **204**, der BMC **113**, das Betriebssystem und/oder andere Komponenten des Informationshandhabungsgerätes **102** dynamisch auf die Eigenschaften der Peripheriegeräte **114**, der Backplane **122**, des SEP **115** und/oder dergleichen abgestimmt werden. Darüber hinaus kann die Datenprotokollvorrichtung **110** Informationen, Benachrichtigungen, Warnungen, Nachrichten und/oder dergleichen für das BIOS **106**, das Betriebssystem, einen Endbenutzer und/oder dergleichen bereitstellen, um einen Zustand des Systems **100/150/175** anzuzeigen, wie beispielsweise ob ein Sachverhalt, Fehler, Problem oder dergleichen mit der Kommunikationsbuskabelkonfiguration zwischen dem Informationshandhabungsgerät **102** und einem oder mehreren Peripheriegeräten **114** der Backplanes **122** vorliegt.

**[0078]** Fig. 3 stellt eine Ausführungsform einer Vorrichtung **300** zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls dar. Die Vorrichtung **300** beinhaltet eine Ausführungsform einer Datenprotokollvorrichtung **110**. Die Datenprotokollvorrichtung **110** beinhaltet ein Verbindungsmodul **230**, ein Datenmodul **235** und ein Topologiemodul **240**, die im Wesentlichen dem Verbindungsmodul **230**, dem Datenmodul **235** und dem Topologiemodul **240** ähneln, die vorstehend mit Bezug auf Fig. 2 beschrieben wurden. Bei einer Ausführungsform beinhaltet die Datenprotokollvorrichtung **110** ein oder mehrere Benachrichtigungsmodule **310**, ein Abstimmungsmodul **315**, ein Bestätigungsmodul **320** und ein Übertragungsmodul **325**, die im Weiteren näher beschrieben werden.

**[0079]** Das Benachrichtigungsmodul **310** sendet bei einer Ausführungsform eine Benachrichtigung in Reaktion darauf, dass die bestimmte Kabelverbindungskonfiguration nicht mit der vordefinierten Kabelverbindungskonfiguration übereinstimmt. Bei bestimmten Ausführungsformen kann ein spezifisches System **100/150/175** eine empfohlene Kabelverbindungskonfiguration zwischen dem Informationshandhabungsgerät **102** und einem oder mehreren Peripheriegeräten **114** der Backplanes **122** aufweisen, die eine erforderliche oder empfohlene Kabelverbindungstopologie definiert, z. B. welche Kommunikationsbuskabel **123** mit welchen Installationspositionen, beispielsweise welchen Slots **117** und Schächte **121** für Peripheriegeräte **114** auf einer Backplane **122**, verbunden werden sollten.

**[0080]** Das Benachrichtigungsmodul **310** sendet basierend auf einer Bestimmung, dass die bestimmte Kommunikationsbuskabelkonfiguration nicht mit der vordefinierten Kommunikationsbuskabelkonfiguration übereinstimmt, eine Benachrichtigung, um das BIOS **106**, das Betriebssystem, das FPGA **204**, einen Endbenutzer und/oder dergleichen zu warnen oder darüber zu informieren, dass die bestimmte Kommunikationsbuskabelkonfiguration falsch oder auf andere Weise als in empfohlener Weise konfiguriert ist.

**[0081]** Bei einer Ausführungsform bestimmt das Topologiemodul **240**, ob die bestimmte Kommunikationsbuskabelkonfiguration mit der vordefinierten Kommunikationsbuskabelkonfiguration übereinstimmt, indem es Daten in der Pfadetabelle von Installationspositionskennungen zu den Kommunikationsbuskabeln, die mit der Kennungsinstallationsposition verbunden sind, mit einer vordefinierten Pfadetabelle von im Wesentlichen ähnlichen Installationspositionskennungen zu Kommunikationsbuskabelpfaden vergleicht. Bei bestimmten Ausführungsformen sendet das Benachrichtigungsmodul **310** basierend auf in dem Datenpaket empfangenen Informationen verschiedene Benachrichtigungen, die Informationen beinhalten, welche die Zustände der Peripheriegeräte **114**, die Anzahl der Peripheriegeräte **114**, die vorhanden und/oder aktiv/deaktiviert und/oder dergleichen sind, betreffen.

**[0082]** Bei einer Ausführungsform modifiziert das Abstimmungsmodul **315** dynamisch eine oder mehrere Leistungseigenschaften des FPGA **204** basierend auf der einen oder den mehreren Eigenschaften des SEP **115**. Zum Beispiel kann das Abstimmungsmodul **315** basierend auf den Daten im Zeitsperrenfeld des Datenpakets verändern, wie oft das FPGA den SEP **115** nach einem Datenpaket für ein Peripheriegerät **114** abfragt. Bei einem weiteren Beispiel kann das Abstimmungsmodul **315** die Rate abstimmen, mit der Befehle an den SEP **115** vom FPGA **204** oder dem Prozessor **104** über das FPGA **204** gesendet werden, basierend auf der

Taktgeschwindigkeit des SEP **115**, ob der SEP **115** gegenwärtig tätig ist oder nicht, den Fähigkeiten der Backplane **122** (z. B. der Art der Backplane **122**, den Kommunikationsbusgeschwindigkeiten auf der Backplane **122**, oder dergleichen) und/oder dergleichen.

**[0083]** Bei einer Ausführungsform prüft oder bestätigt das Bestätigungsmodul **320** mit Hilfe einer Prüfsumme eine Integrität von Daten, die in einem oder mehreren Feldern des Datenpakets enthalten sind. Die Prüfsumme ist bei einigen Ausführungsformen in einem Prüfsummenfeld des Datenpakets enthalten. Zum Beispiel kann der SEP **115** jedes Datenfeld des Datenpakets mit Daten bestücken, mit Hilfe der Daten in dem Datenpaket eine Prüfsumme erzeugen und die Prüfsumme im Prüfsummenfeld des Datenpakets speichern. Demzufolge verwendet das Bestätigungsmodul **320**, wenn das Datenmodul **235** das Datenpaket über ein Kommunikationsbuskabel **123** empfängt, die Prüfsumme, um zu bestätigen, dass die Daten (z. B. die Schacht-ID und die Slot-ID, die Fähigkeitsdaten des SEP **115** und/oder der Backplane **122** und/oder dergleichen, die im Datenpaket gespeichert nicht) nicht verfälscht sind, keine Fehler beinhalten und/oder dergleichen.

**[0084]** Bei einer Ausführungsform empfängt das Übertragungsmodul **325** einen Befehl für einen virtuellen Pin-Anschluss („VPP“) von einem VPP-aktivierten Prozessor **104** oder einem Prozessor **104**, der dafür konfiguriert ist, über eine VPP-Seitenbandverbindung Befehle zu senden und zu empfangen. Wie hierin verwendet, kann ein VPP eine auf einem seriellen Bus basierende Verbindung für Peripheriegeräte **114** beinhalten, wie beispielsweise PCIe-NVMe-Geräte, die bei laufendem Betrieb ansteckbar sind. Die VPPs können Teil eines Eingangs/Ausgangs-Hubs des Prozessors **104** sein, z. B. ein VPP-aktiver Chipsatz von Intel®, und kann dafür konfiguriert sein, die Signale des bei laufendem Betrieb angestecktem Seitenband-PCIe seriell zu verschieben.

**[0085]** Die VPP-Architektur ist für eine Schnittstellenverbindung mit Peripheriegeräten **114** gestaltet, die sich in mechanisch befestigten Slots befinden, wie beispielsweise physischen PCIe-Slots auf einem Motherboard, und erstreckt sich unter Umständen nicht auf eine Schnittstellenverbindung mit Peripheriegeräten **114**, die mit Hilfe von Kommunikationsbuskabeln **123** mit dem Prozessor **104** verbunden sind. Zum Beispiel kann die VPP-Architektur eine Unterstützung bereitstellen, um ein Peripheriegerät **114** zu aktivieren, um zu bestimmen, ob an einem bestimmten physischen Slot ein Laufwerk vorhanden ist und/oder die Aktivitätsanzeige-LED am Peripheriegerät **114** zu steuern. Bei Systemen **100/150/175**, die Kommunikationsbuskabel **123** verwenden, um einen Prozessor **104** mit einem oder mehreren Peripheriegeräten **114** zu koppeln, besitzt die VPP-Architektur jedoch nicht die Fähigkeiten, Peripheriegeräte **114** vollständig zu verwalten, die auf einer Backplane **122** installiert und mit Hilfe eines oder mehrerer Kommunikationsbuskabel **123** mit einem Prozessor **104** verbunden sind.

**[0086]** Wenn zum Beispiel Kommunikationsbuskabel **123** vertauscht und mit der falschen Installationsposition eines Peripheriegeräts **114** verbunden sind, z. B. dem falschen Slot **117** und/oder Schacht **121**, würde die VPP-Architektur die fehlerhafte Konfiguration der Kommunikationsbuskabel **123** nicht kennen, was bewirken kann, dass das Betriebssystem und/oder das BIOS **106** nicht mit der gegenwärtigen Hardware-Topologie synchron ist. Darüber hinaus ermöglicht die VPP-Architektur wegen Einschränkungen bei der elektrischen Weiterleitung (Routing), z. B. auf dem Motherboard, möglicherweise keine direkten Verbindungen mit mehreren Backplanes **122**. Des Weiteren stellt die VPP-Architektur möglicherweise keine Datenintegritätsüberprüfung, wie beispielsweise Prüfsummen und Datenparitätswerte, bereit.

**[0087]** Somit stellt das hierin beschriebene Protokoll eine erweiterte und beschreibendere Architektur bereit, welche die Fähigkeiten zum Verwalten mehrerer Peripheriegeräte **114** bereitstellt, die sich auf mehreren Backplanes **122** befinden und mit Hilfe eines oder mehrerer Kommunikationsbuskabel **123** mit dem Prozessor **104** verbunden sind. Bei einer Ausführungsform empfängt das Übertragungsmodul **325** einen VPP-Befehl von dem, Prozessor **104** und überträgt, verwandelt, codiert oder dergleichen den empfangenen VPP-Befehl in ein entsprechendes Datenfeld in dem Datenpaket.

**[0088]** Der VPP-Befehl kann bei einigen Ausführungsformen für ein Peripheriegerät **114** auf einer Backplane **122** vorgesehen sein, welches durch die Installationspositionskennung identifiziert ist, die das Datenmodul **235** empfängt. Bei einer weiteren Ausführungsform schreibt das Übertragungsmodul den VPP-Befehl in ein entsprechendes Datenfeld in dem Datenpaket. Bei einigen Ausführungsformen beinhaltet der VPP-Befehl einen Aktivierungsbefehl für ein Peripheriegerät **114**, einen Deaktivierungsbefehl für ein Peripheriegerät **114** und/oder einen Befehl, einen Aktivitätsindikator an dem Peripheriegerät **114**, wie beispielsweise eine LED, umzuschalten. Somit kann das Übertragungsmodul **325** einen VPP-Befehl empfangen, die Aktivitätsindikator-LED an einem spezifischen Peripheriegerät **114** umzuschalten, bestimmen, welches Datenpaket des Datenfelds dem empfangenen VPP-Befehl entspricht, und einen Wert, der dem Wert des VPP-Befehls entspricht, in das entsprechende Datenfeld in dem Datenpaket zu schreiben.

**[0089]** Bei verschiedenen Ausführungsformen sendet das Übertragungsmodul **325** das Datenpaket, das den VPP-Befehl beinhaltet, mit Hilfe eines Kommunikationsbuskabels **123**, das mit dem SEP **115** gekoppelt und mit dem vorgesehenen Peripheriegerät **114** assoziiert ist, an den SEP **115** auf der Backplane **122**. Zum Beispiel kann das Übertragungsmodul **325** einen VPP-Befehl zum Deaktivieren eines Peripheriegeräts **114** empfangen. Das Übertragungsmodul **325** kann eine Slot-ID und eine Schacht-ID für das vorgesehene Peripheriegerät **114** und das Kommunikationsbuskabel **123** bestimmen, das mit der bestimmten Slot-ID und Schacht-ID gekoppelt ist, z. B. mit Hilfe der Pfadetabelle, die durch das Topologiemodul **240** bereitgestellt wird. Das Übertragungsmodul **325** kann den VPP-Befehl dann auf einen entsprechenden Wert in dem Datenfeld des Datenpakets codieren, das dem Deaktivierungsbefehl entspricht, und das Datenpaket über das identifizierte Kommunikationsbuskabel **123** an den SEP **115** senden.

**[0090]** Bei einer Ausführungsform schreibt das Übertragungsmodul **325** einen Paritätswert in ein Paritätsfeld des Datenpakets, bevor es das Datenpaket zum vorgesehenen Peripheriegerät **114** sendet. Der Paritätswert kann von dem SEP **115** verwendet werden, um die Integrität der in das Datenpaket geschriebenen Daten zu bestätigen, wie beispielsweise die Integrität des VPP-Befehls, der in das Datenpaket codiert ist. Der Paritätswert kann, wie vorstehend in Tabelle 1 beschrieben, mit Hilfe einer Kombination aus einem oder mehreren Bits von verschiedenen Datenfeldern erzeugt werden. Bei einer Ausführungsform erzeugt das Übertragungsmodul **325** einen Paritätswert und schreibt diesen in das Paritätsfeld, wenn der SEP **115** Datenparität unterstützt, wie in verschiedenen Datenfeldern des Datenpakets angezeigt.

**[0091]** Fig. 4 stellt eine Ausführungsform eines Verfahrens **400** zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls dar. Das Verfahren **400** beginnt und bestimmt **402** ein oder mehrere Kommunikationsbuskabel **123**, die ein oder mehrere Peripheriegeräte **114** mit einem Informationshandhabungsgerät **102** koppeln. Das Verfahren **400** liest **404** über jedes der einen oder der mehreren Kommunikationsbuskabel **123** ein Datenpaket, das mit einem Peripheriegerät **114** assoziiert ist. Jedes Datenpaket beinhaltet eine Kennung für die Position, an der das Peripheriegerät **114** installiert ist. Das Verfahren **400** bestimmt **406** eine Kabelverbindungsconfiguration für das eine oder die mehreren Kommunikationsbuskabel **123** und das eine oder die mehreren Peripheriegeräte **114**, basierend auf der Installationspositionskennung, die aus jedem der dem einen oder den mehreren Peripheriegeräten **114** assoziierten Datenpakete empfangen wurde, und das Verfahren **400** endet. Bei einigen Ausführungsformen führen das Verbindungsmodul **230**, das Datenmodul **235** und/oder das Topologiemodul **240** die verschiedenen Schritte des Verfahrens **400** aus.

**[0092]** Fig. 5 stellt eine Ausführungsform eines Verfahrens **500** zum Verwalten von Peripheriegeräten mit Hilfe eines Datenprotokolls dar. Das Verfahren **500** beginnt und bestimmt **502** ein oder mehrere Kommunikationsbuskabel **123**, die ein oder mehrere Peripheriegeräte **114** kommunikationstechnisch mit einem Informationshandhabungsgerät **102** koppeln. Das Verfahren **500** liest **504** über jedes der einen oder der mehreren Kommunikationsbuskabel **123** ein Datenpaket, das mit einem Peripheriegerät **114** assoziiert ist. Bei einer Ausführungsform bestätigt **506** das Verfahren **500** mit Hilfe einer Prüfsumme, die in einem Prüfsummenfeld des Datenpakets enthalten ist, eine Integrität von Daten, die in einem oder mehreren Feldern des Datenpakets enthalten sind.

**[0093]** Wenn bei einigen Ausführungsformen das Verfahren **500** basierend auf der Prüfsumme, die anzeigen kann, dass in den Daten ein Fehler vorliegt, dass die Daten verfälscht sind oder dergleichen, bestimmt **508**, dass die Integrität des Datenpakets nicht gegeben ist, endet das Verfahren **500**. Wenn das Verfahren **500** andererseits bei bestimmten Ausführungsformen bestimmt **508**, dass die Integrität der Daten gegeben ist, bestimmt **510** das Verfahren **500** basierend auf der Installationspositionskennung, die aus jedem der mit dem einen oder den mehreren Peripheriegeräten **114** assoziierten Datenpakete empfangen wurde, eine Kabelverbindungsconfiguration für das eine oder die mehreren Kommunikationsbuskabel **123** und das eine oder die mehreren Peripheriegeräte **114**.

**[0094]** Bei einigen Ausführungsformen stellt **512** das Verfahren **500** die Kabelverbindungsconfiguration für die Kommunikationsbuskabel **123** und die Peripheriegeräte **114** für das BIOS oder das Betriebssystem des Informationshandhabungsgeräts **102** bereit. Bei einer Ausführungsform bestimmt **514** das Verfahren **500**, ob die bestimmte Kabelverbindungsconfiguration mit einer vordefinierten Kabelverbindungsconfiguration übereinstimmt. Falls nicht, sendet **516** das Verfahren **500** bei einer Ausführungsform eine Benachrichtigung, um anzuzeigen, dass die bestimmte Verbindungsconfiguration der Kommunikationsbuskabel **123** nicht mit der vordefinierten oder empfohlenen Kabelverbindungsconfiguration übereinstimmt.

**[0095]** Bei einer weiteren Ausführungsform stimmt **518** das Verfahren **500** basierend auf einer oder mehreren Eigenschaften des SEP **115**, die in dem Datenpaket enthalten sind, welches das Verfahren **500** empfangen hat,

eine oder mehrere Leistungseigenschaften des FPGA **204** dynamisch ab. Wie vorstehend beschrieben, wird das FPGA **204** bei einer Ausführungsform mit Hilfe von Kommunikationsbuskabeln **123** in einer Master/Slave-Konfiguration, in der das FPGA **204** der Bus-Master ist und Kommunikationen mit dem SEP **115** als Bus-Slave und den Peripheriegeräten **114** als Slave-Geräten führt oder leitet, kommunikationstechnisch mit dem SEP **115** und einem oder mehreren Peripheriegeräten **114** der Backplane **122** gekoppelt und das Verfahren **500** endet. Bei einigen Ausführungsformen führen das Verbindungsmodul **230**, das Datenmodul **235** und/oder das Topologiemodul **240**, das Benachrichtigungsmodul **310**, das Abstimmungsmodul **315**, das Bestätigungsmodul **320** und/oder das Übertragungsmodul **325** die verschiedenen Schritte des Verfahrens **500** aus.

**[0096]** Ausführungsformen können in anderen spezifischen Formen praktisch angewendet werden. Die beschriebenen Ausführungsformen sind in jeglicher Hinsicht lediglich als veranschaulichend und nicht als beschränkend zu betrachten. Der Geltungsbereich der Erfindung wird somit durch die angefügten Ansprüche angegeben anstatt durch die vorstehende Beschreibung. Alle Veränderungen, die dem Sinn und dem Bedeutungsbereich von Äquivalenz zu den Ansprüchen entsprechen, sollen in deren Geltungsbereich eingeschlossen sein.

### Patentansprüche

1. Vorrichtung, umfassend:
  - ein Verbindungsmodul, das ein oder mehrere Kommunikationsbuskabel bestimmt, die ein oder mehrere Peripheriegeräte kommunikationstechnisch mit einem Informationshandhabungsgerät koppeln,
  - ein Datenmodul, das über jedes des einen oder der mehreren Kommunikationsbuskabel ein Datenpaket liest, das mit einem Peripheriegerät assoziiert ist, wobei jedes Datenpaket eine Kennung für eine Position umfasst, an der das Peripheriegerät installiert ist, und
  - ein Topologiemodul, das basierend auf der Installationspositionskennung, die aus jedem der mit dem einen oder den mehreren Peripheriegeräten assoziierten Datenpakete empfangen wurde, eine Kabelverbindungskonfiguration für das eine oder die mehreren Kommunikationsbuskabel und das eine oder die mehreren Peripheriegeräte bestimmt.
2. Vorrichtung nach Anspruch 1, wobei das Topologiemodul die Kabelverbindungskonfiguration der Kommunikationsbuskabel und der Peripheriegeräte einem Basic Input/Output System (BIOS) bereitstellt.
3. Vorrichtung nach Anspruch 2, ferner ein Benachrichtigungsmodul umfassend, das in Reaktion darauf, dass die bestimmte Kabelverbindungskonfiguration nicht mit einer vordefinierten Kabelverbindungskonfiguration übereinstimmt, eine Benachrichtigung sendet.
4. Vorrichtung nach Anspruch 1, wobei jedes des einen oder der mehreren Peripheriegeräte in einem Laufwerksschacht einer Backplane installiert ist und die Kennung eine Schachtkennung und eine assoziierte Slotkennung des Laufwerksschachts umfasst.
5. Vorrichtung nach Anspruch 4, wobei das Datenpaket mehrere Felder umfasst, welche die Kennung definieren, wobei die mehreren Felder ein Feld für die Schachtkennung und ein Feld für die Slotkennung umfassen.
6. Vorrichtung nach Anspruch 1, wobei das Datenpaket ferner ein oder mehrere Felder umfasst, die Daten beinhalten, welche eine oder mehrere Eigenschaften eines Storage Enclosure Processor (SEP) einer Backplane beschreiben.
7. Vorrichtung nach Anspruch 6, ferner ein Abstimmungsmodul umfassend, das basierend auf der einen oder den mehreren Eigenschaften des SEPs ein oder mehrere Leistungseigenschaften eines Field Programmable Gate Array (FPGA) dynamisch modifiziert, wobei das FPGA mit Hilfe eines Kommunikationsbuskabels kommunikationstechnisch mit dem SEP und einem oder mehreren Peripheriegeräten gekoppelt ist.
8. Vorrichtung nach Anspruch 6, wobei die eine oder die mehreren Eigenschaften des SEP Folgendes umfassen:
  - einen Bitschalter, der anzeigt, ob der SEP eine Datenparität unterstützt,
  - eine maximale Taktgeschwindigkeit für den SEP, einen Zeitsperreparameter für das Polling des SEP und
  - eine maximale Anzahl an Peripheriegeräten, die von dem SEP unterstützt werden.

9. Vorrichtung nach Anspruch 1, ferner ein Bestätigungsmodul umfassend, das mit Hilfe einer Prüfsumme eine Integrität von Daten prüft, die in einem oder mehreren Feldern des Datenpakets enthalten sind, wobei die Prüfsumme in einem Prüfsummenfeld des Datenpakets enthalten ist.

10. Vorrichtung nach Anspruch 1, ferner ein Übertragungsmodul umfassend, das Folgendes ausführt:

- Empfangen eines Befehls für einen virtuellen Port-Anschluss (VPP) von einem VPP-aktivierten Prozessor, wobei der VPP-Befehl für ein Peripheriegerät auf einer Backplane vorgesehen ist, wobei das Peripheriegerät durch die Installationspositionskenung identifiziert wird,
- Schreiben des VPP-Befehls in ein entsprechendes Feld in dem Datenpaket und
- Senden des Datenpakets, das den VPP-Befehl beinhaltet, an einen Storage Enclosure Processor (SEP) auf der Backplane mit Hilfe eines Kommunikationsbuskabels, das mit dem SEP gekoppelt und mit dem vorgesehenen Peripheriegerät assoziiert ist.

11. Vorrichtung nach Anspruch 10, wobei das Übertragungsmodul einen Paritätswert in ein Paritätsfeld des Datenpakets schreibt, wobei der Paritätswert von dem SEP dazu verwendet wird, um eine Integrität von Daten zu bestätigen, die in das Datenpaket geschrieben sind.

12. Vorrichtung nach Anspruch 10, wobei der VPP-Befehl einen oder mehrere der Folgenden umfasst:

- einen Peripheriegerät-Aktivierungsbefehl,
- einen Peripheriegerät-Deaktivierungsbefehl und
- einen Peripheriegerät-Aktivitätsindikator-Umschaltbefehl.

13. Vorrichtung nach Anspruch 1, wobei das Kommunikationsbuskabel ein Peripheral-Component-Interconnection-Express (PCIe) Kabel umfasst.

14. Vorrichtung nach Anspruch 1, wobei das Peripheriegerät ein Non-Volatile-Memory-Express (NVMe) Speichergerät umfasst.

15. Verfahren, umfassend:

- Bestimmen eines oder mehrerer Kommunikationsbuskabel, die ein oder mehrere Peripheriegeräte kommunikationstechnisch mit einem Informationshandhabungsgerät koppeln,
- Lesen eines Datenpakets über jedes des einen oder der mehreren Kommunikationsbuskabel, das mit einem Peripheriegerät assoziiert ist, wobei jedes Datenpaket eine Kennung für eine Position umfasst, an der das Peripheriegerät installiert ist, und
- Bestimmen einer Kabelverbindungsconfiguration für das eine oder die mehreren Kommunikationsbuskabel und das eine oder die mehreren Peripheriegeräte, basierend auf der Installationspositionskenung, die aus jedem der Datenpakete empfangen wurde, das mit dem einen oder den mehreren Peripheriegeräten assoziiert ist.

16. Verfahren nach Anspruch 15, ferner Folgendes umfassend:

- Bereitstellen einer Kabelverbindungsconfiguration der Kommunikationsbuskabel und der Peripheriegeräte für ein Basic Input/Output System (BIOS) und
- Senden einer Benachrichtigung in Reaktion darauf, dass die bestimmte Kabelverbindungsconfiguration nicht mit einer vordefinierten Kabelverbindungsconfiguration übereinstimmt.

17. Verfahren nach Anspruch 15, wobei:

- jedes des einen oder der mehreren Peripheriegeräte in einem Laufwerksschacht einer Backplane installiert ist, die Kennung eine Schachtkennung und eine assoziierte Slotkennung des Laufwerksschachts umfasst und
- das Datenpaket mehrere Felder umfasst, welche die Kennung identifizieren, wobei die mehreren Felder ein Feld für die Schachtkennung und ein Feld für die Slotkennung umfassen.

18. Verfahren nach Anspruch 15, ferner das dynamische Modifizieren einer oder mehrerer Leistungseigenschaften eines Field Programmable Gate Array (FPGA), basierend auf einer oder mehreren Eigenschaften eines Storage Enclosure Processor (SEP) einer Backplane umfassend, wobei das FPGA mit Hilfe eines Kommunikationsbuskabels kommunikationstechnisch mit dem SEP und einem oder mehreren Peripheriegeräten der Backplane gekoppelt ist, wobei das Datenpaket ferner ein oder mehrere Felder umfasst, die Daten beinhalten, welche die eine oder die mehreren Eigenschaften des SEP beschreiben.

19. Verfahren nach Anspruch 15, ferner Folgendes umfassend:

- Empfangen eines Befehls für einen virtuellen Port-Anschluss (VPP) von einem VPP-aktivierten Prozessor, der für ein Peripheriegerät auf einer Backplane vorgesehen ist, wobei das Peripheriegerät durch die Installationspositionskennung identifiziert ist,
- Schreiben des VPP-Befehls in ein entsprechendes Feld in dem Datenpaket und
- Senden des Datenpakets, das den VPP-Befehl umfasst, an einen Storage Enclosure Processor (SEP) auf der Backplane mit Hilfe eines Kommunikationsbuskabels, das mit dem SEP gekoppelt und mit dem vorgesehenen Peripheriegerät assoziiert ist.

20. Programmprodukt, umfassend ein computerlesbares Speichermedium, das Code speichert, der von einem Prozessor ausgeführt werden kann, wobei der ausführbare Code Code zum Ausführen des Folgenden umfasst:

- Bestimmen eines oder mehrerer Kommunikationsbuskabel, die ein oder mehrere Peripheriegeräte kommunikationstechnisch mit einem Informationshandhabungsgerät koppeln,
- Lesen über jedes des einen oder der mehreren Kommunikationsbuskabel eines Datenpakets, das mit einem Peripheriegerät assoziiert ist, wobei jedes Datenpaket eine Kennung für eine Position umfasst, an der das Peripheriegerät installiert ist, und
- Bestimmen einer Kabelverbindungskonfiguration für das eine oder die mehreren Kommunikationsbuskabel und das eine oder die mehreren Peripheriegeräte, basierend auf der Installationspositionskennung, die aus jedem der Datenpakete empfangen wurde, die mit dem einen oder den mehreren Peripheriegeräten assoziiert sind.

Es folgen 7 Seiten Zeichnungen

Anhängende Zeichnungen

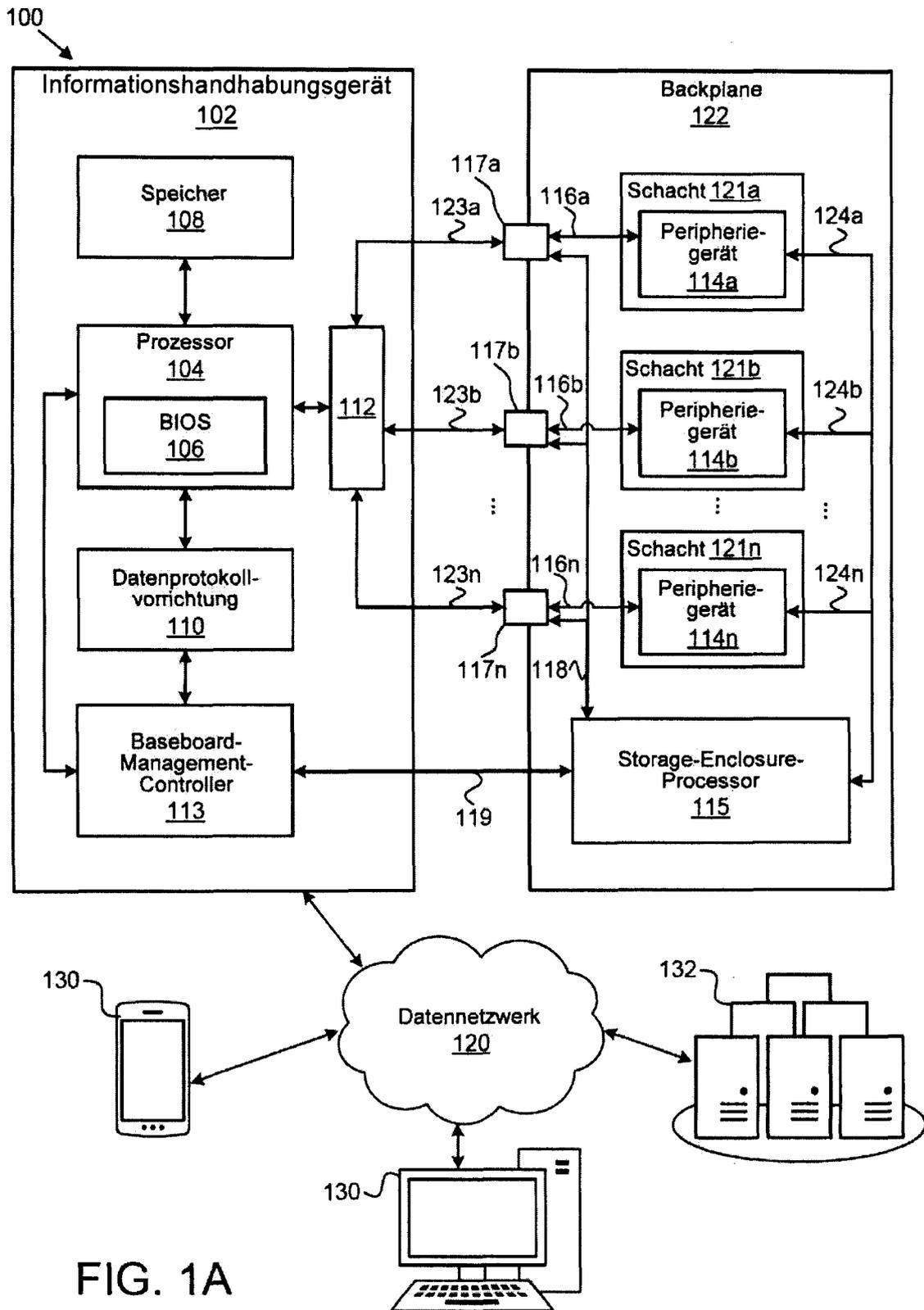


FIG. 1A

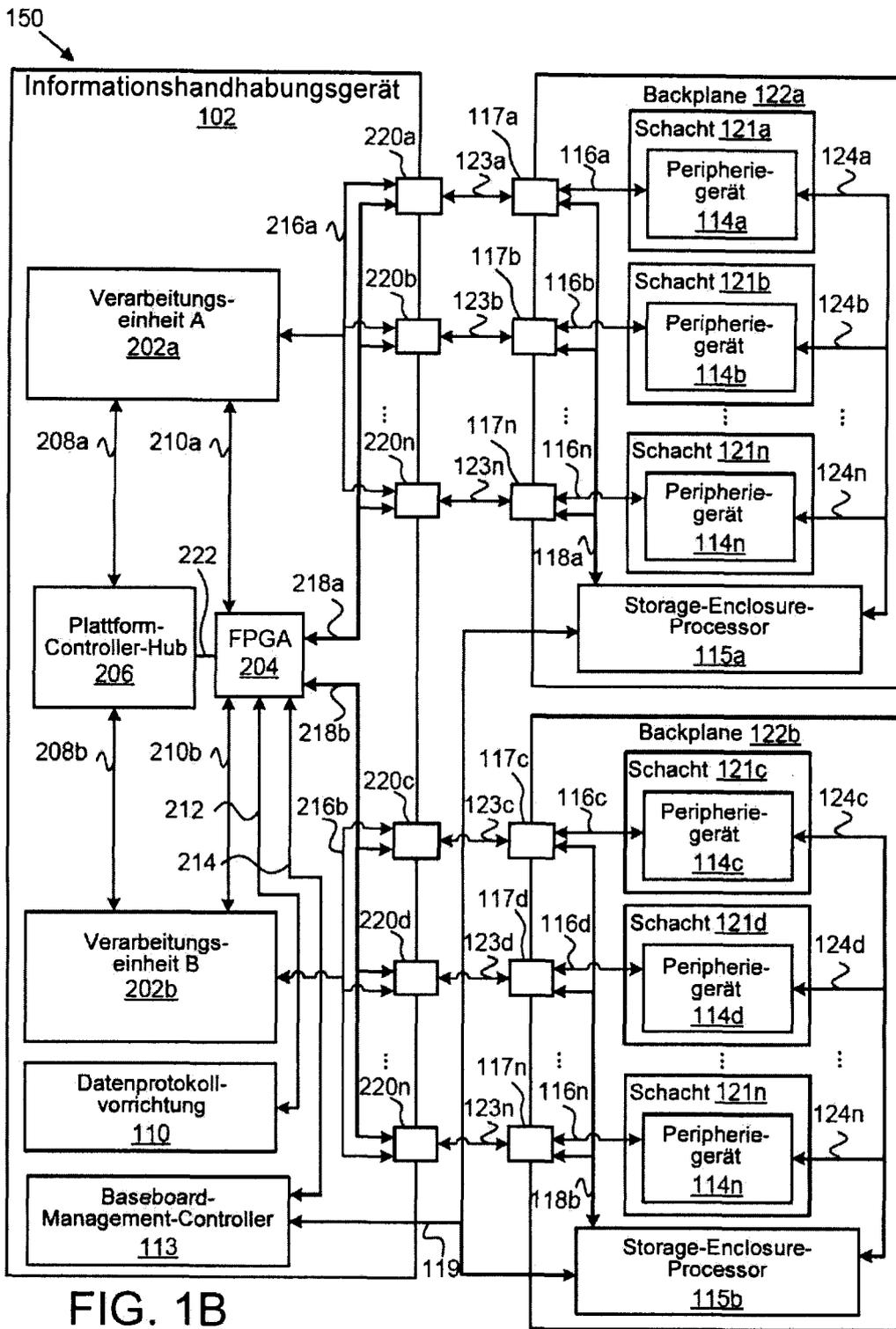


FIG. 1B

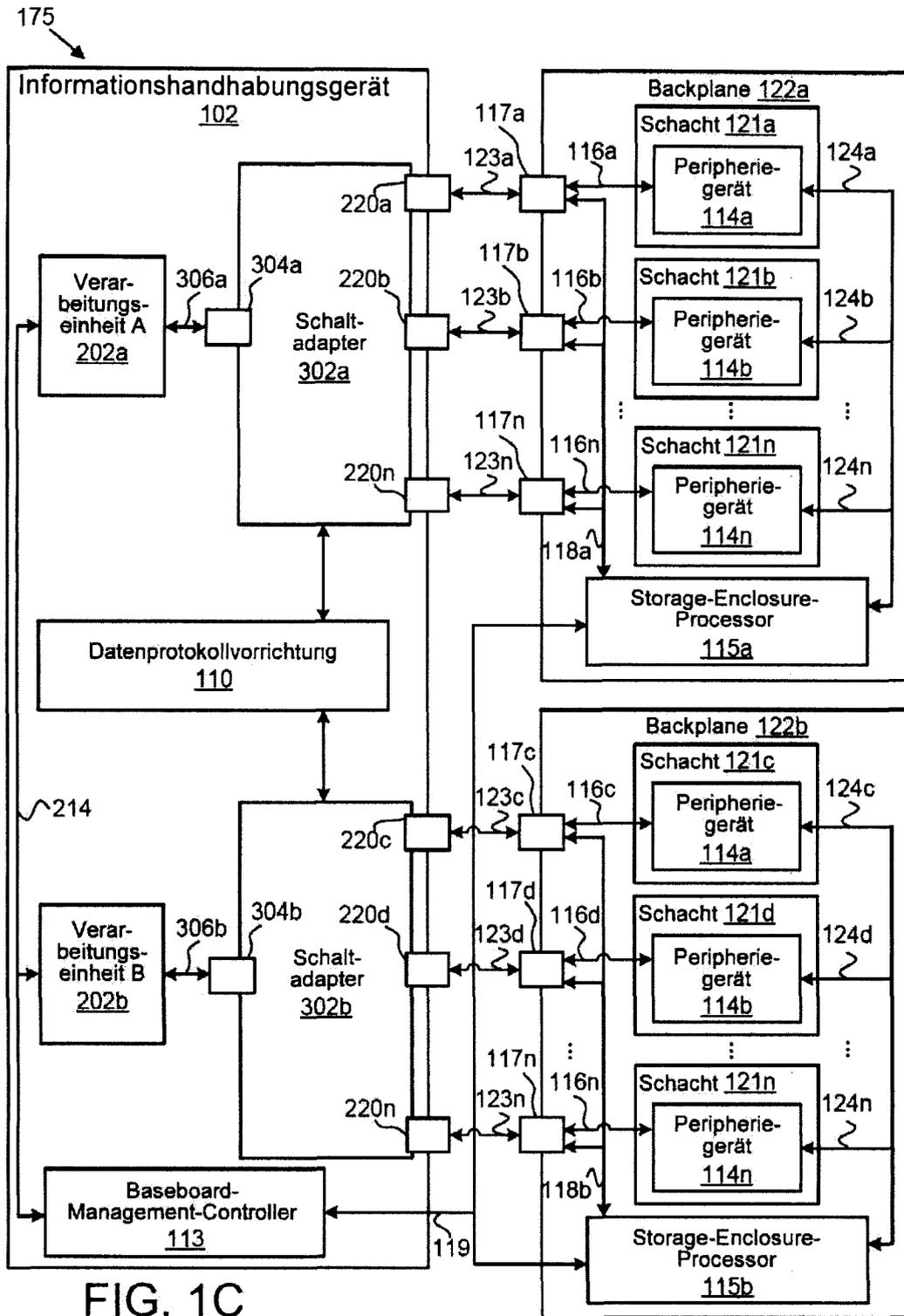


FIG. 1C

200 →

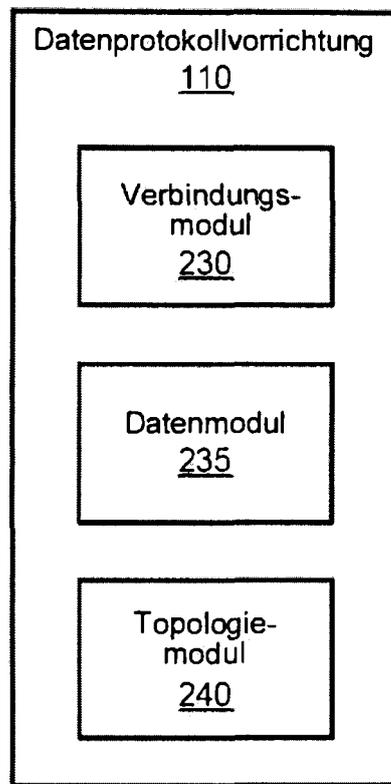


FIG. 2

300 →

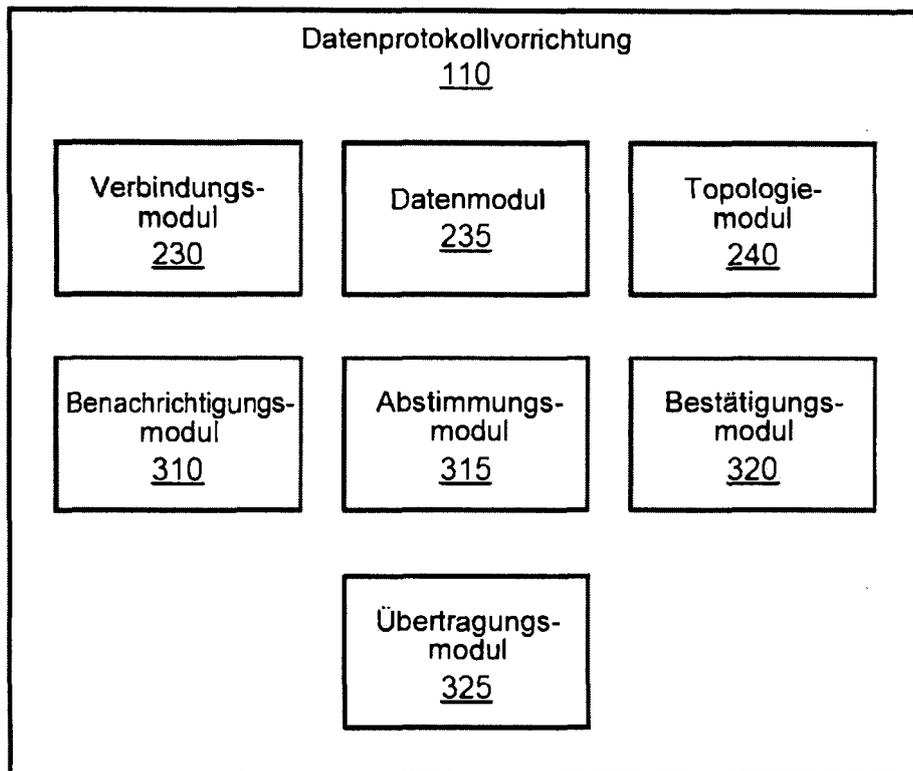


FIG. 3

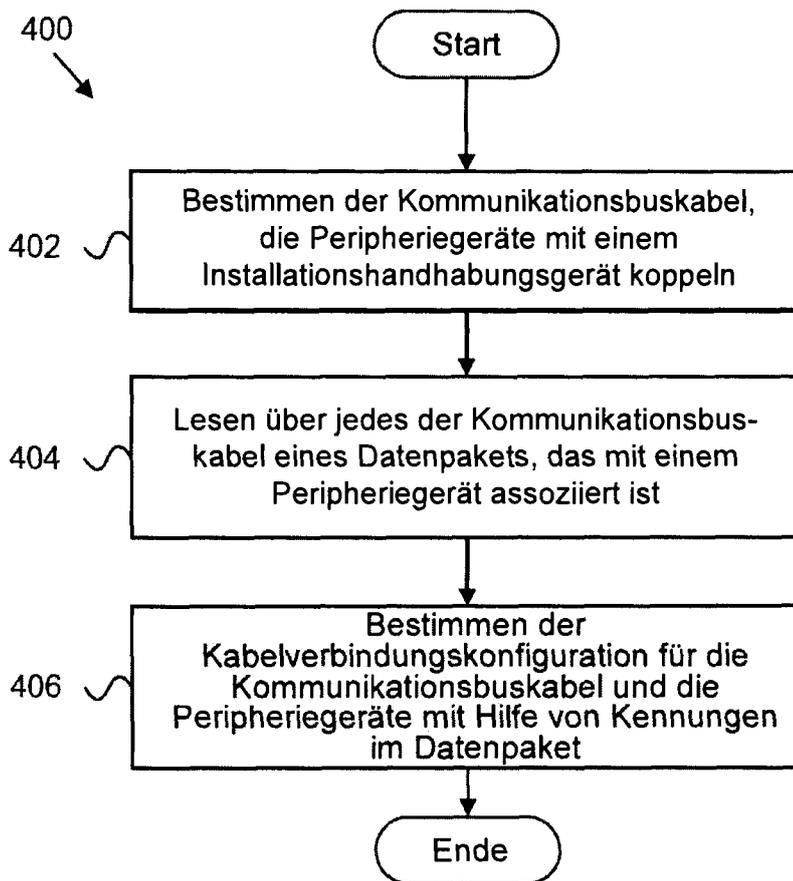


FIG. 4

