(19) Deutsches Patent- und Markenamt

(10) **DE 10 2015 114 233 B4** 2024.10.24

(12) # Patentschrift

(21) Aktenzeichen: **10 2015 114 233.3**
(22) Anmeldetag: **27.08.2015**
(43) Offenlegungstag: **02.03.2017**
(45) Veröffentlichungstag der Patenterteilung: **24.10.2024**

(51) Int Cl.: ***G06F 21/71*** (2013.01)
G06K 19/073 (2006.01)
G06F 21/64 (2013.01)
G06F 21/78 (2013.01)
G06F 11/16 (2006.01)

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

(73) Patentinhaber:
Infineon Technologies AG, 85579 Neubiberg, DE

(74) Vertreter:
Viering, Jentschura & Partner mbB Patent- und Rechtsanwälte, 01099 Dresden, DE

(72) Erfinder:
Driessen, Benedikt, 80636 München, DE;
Sonnekalb, Steffen, 82024 Taufkirchen, DE

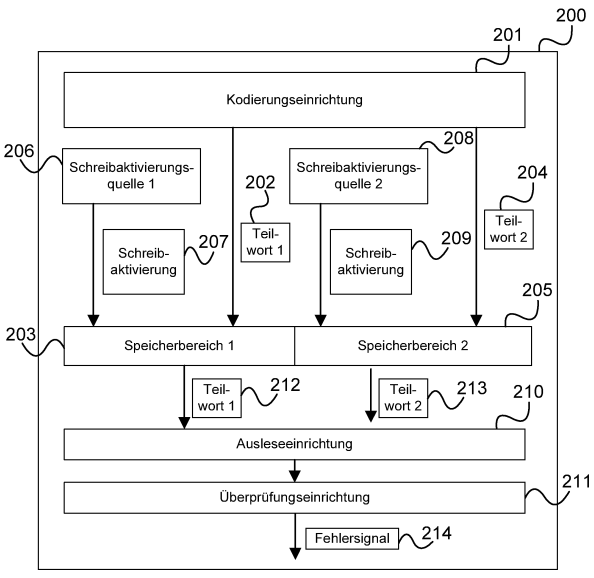
(56) Ermittelter Stand der Technik:

DE	**103 60 998**	**B4**
DE	**10 2013 205 542**	**A1**

FIENUP, M.: Computer Organization (810:041), Fall 2008, Lecture Notes, Lecture 13, Register File Implementation. University of Northern Iowa, 2008. URL: http://www.cs.uni.edu/~fienup/cs041f08/lectures/lec13_reg_file.pdf [abgerufen am 11.03.2024]Aufgerufen über: http://www.cs.uni.edu/~fienup/cs041f08/

(54) Bezeichnung: **Datenverarbeitungsvorrichtungen und Verfahren zum Speichern eines Datenvektors**

(57) Zusammenfassung: Gemäß einem Ausführungsbeispiel wird eine Datenverarbeitungsvorrichtung bereitgestellt, die aufweist: eine Kodierungseinrichtung, die eingerichtet ist, einen Datenvektor zu kodieren, ein erstes Teilwort des Codeworts einem ersten Speicherbereich zuzuführen und ein zweites Teilwort des Codeworts einem zweiten Speicherbereich zuzuführen, eine erste und eine zweite Schreibaktivierungssignalquelle, die dem ersten bzw. dem zweiten Speicherbereich ein erstes bzw. ein zweites Schreibaktivierungssignal zuführt, eine Ausleseeinrichtung, die das erste Teilwort und das zweite Teilwort aus den Speicherbereichen ausliest und eine Überprüfungseinrichtung, die überprüft, ob das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort ein gültiges Codewort des Codes bilden und ein Fehlersignal auszugeben, falls nicht.



Beschreibung

[0001] Ausführungsbeispiele betreffen allgemein Datenverarbeitungsvorrichtungen und Verfahren zum Speichern eines Datenvektors.

[0002] Sicherheitsrelevante Datenverarbeitungssysteme wie ein Mikroprozessor oder ein Mikrocontroller, die beispielsweise auf einer Chipkarte vorgesehen sind, sollen typischerweise gegen Angriffe geschützt werden. Insbesondere sollen beispielsweise Manipulationen von erzeugten Daten und von Befehlen (und damit einer Programmausführung) durch die Injektion von Fehlern durch einen Angreifer vermieden werden. Um zu hohe zusätzliche Kosten hinsichtlich Chipflächenbedarf und Energieverbrauch zu vermeiden, sind effiziente Herangehensweisen wünschenswert, die einen Schutz gegen solche Angriffe ermöglichen.

[0003] DE 10 2013 205 542 A1 offenbart eine Vorrichtung zur Verarbeitung von Daten, wobei die Vorrichtung eine Eingangsschnittstelle zum Empfang von Eingangsdaten aufweist und eine Verarbeitungseinheit zur Verarbeitung von Daten, dadurch gekennzeichnet, dass eine Kodiereinheit vorgesehen ist, die dazu ausgebildet ist, an der Eingangsschnittstelle als Eingangsdaten erhaltene Datenworte zu kodieren, um kodierte Datenworte zu erhalten, wobei für kodierte Datenworte die kodierten Datenworte und/oder ihre Verarbeitung durch die Vorrichtung charakterisierende Messwerte in Abhängigkeit mindestens einer physikalischen Größe der Vorrichtung ermittelbar sind, wobei die Kodiereinheit dazu ausgebildet ist, die Datenworte so zu kodieren, dass ein vorgebbarer Anteil aller Messwerte, vorzugsweise mindestens etwa 50% aller Messwerte, einen Unterschied zu einem Vorgabewert aufweist, der kleiner oder gleich einem vorgebbaren Schwellwert ist, und dass die Verarbeitungseinheit dazu ausgebildet ist, die kodierten Datenworte zu verarbeiten.

[0004] DE 103 60 998 B4 offenbart einen Chip, der eine Verarbeitungseinrichtung zur Durchführung einer vorbestimmten Operation, eine Einrichtung zum Bereitstellen des Takts, mit dem die Verarbeitungseinrichtung getaktet wird, einen Zähler zum, basierend auf dem Takt, De- oder Inkrementieren eines Zählerwerts, eine Einrichtung zum, abhängig vom Zählerwert, Signalisieren, dass die vorbestimmte Operation zu unterbinden ist, und eine Einrichtung zum nichtflüchtigen Speichern des Zählerwerts aufweist.

[0005] Eine Datenverarbeitungsvorrichtung gemäß Anspruch 1, eine Chipkarte gemäß Anspruch 16 und ein Verfahren zum Speichern eines Datenvektors gemäß Anspruch 17 werden bereitgestellt. Weitere Ausführungsbeispiele sind in den abhängigen Ansprüchen beschrieben.

[0006] Gemäß einer Ausführungsform wird eine Datenverarbeitungsvorrichtung bereitgestellt aufweisend eine Kodierungseinrichtung, die eingerichtet ist, einen Datenvektor zu einem Codewort eines Codes zu kodieren, ein erstes Teilwort des Codeworts einem ersten Speicherbereich zuzuführen und ein zweites Teilwort des Codeworts einem zweiten Speicherbereich zuzuführen, eine erste Schreibaktivierungssignalquelle, die eingerichtet ist, zum Speichern des ersten Teilworts in dem ersten Speicherbereich dem ersten Speicherbereich ein erstes Schreibaktivierungssignal zuzuführen, eine zweite Schreibaktivierungssignalquelle, die eingerichtet ist, zum Speichern des zweiten Teilworts in dem zweiten Speicherbereich dem zweiten Speicherbereich ein zweites Schreibaktivierungssignal zuzuführen, eine Ausleseeinrichtung, die eingerichtet ist, das erste Teilwort aus dem ersten Speicherbereich auszulesen und des zweite Teilwort aus dem zweiten Speicherbereich auszulesen und eine Überprüfungseinrichtung, die eingerichtet ist, zu überprüfen, ob das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort ein gültiges Codewort des Codes bilden und ein Fehlersignal auszugeben, falls das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort kein gültiges Codewort des Codes bilden.

[0007] Die Figuren geben nicht die tatsächlichen Größenverhältnisse wieder sondern sollen dazu dienen, die Prinzipien der verschiedenen Ausführungsbeispiele zu illustrieren. Im Folgenden werden verschiedene Ausführungsbeispiele mit Bezug auf die folgenden Figuren beschrieben.

Fig. 1 zeigt eine Chipkarte.

Fig. 2 zeigt eine Datenverarbeitungsvorrichtung gemäß einer Ausführungsform.

Fig. 3 zeigt ein Ablaufdiagramm, das ein Verfahren zum Speichern eines Datenvektors veranschaulicht.

Fig. 4 zeigt Speicheranordnung gemäß einer Ausführungsform mit einer Aufteilung eines Registers in Registerbereiche, die von unterschiedlichen CPUs angesteuert werden.

Fig. 5 zeigt eine Speicheranordnung gemäß einer Ausführungsform mit einer Aufteilung eines Arbeitsspeichers in Speicherbereiche, die von unterschiedlichen CPUs angesteuert werden.

Fig. 6 zeigt eine Speicheranordnung gemäß einer weiteren Ausführungsform mit einer Aufteilung eines Registers in Registerbereiche, die von unterschiedlichen CPUs angesteuert werden.

Fig. 7 zeigt eine Speicheranordnung gemäß einer weiteren Ausführungsform mit einer Aufteilung eines Registers in Registerbereiche, die

von unterschiedlichen CPUs angesteuert werden.

[0008] Die folgende detaillierte Beschreibung bezieht sich auf die beiliegenden Figuren, die Details und Ausführungsbeispiele zeigen. Diese Ausführungsbeispiele sind so detailliert beschrieben, dass der Fachmann die Erfindung ausführen kann. Andere Ausführungsformen sind auch möglich und die Ausführungsbeispiele können in struktureller, logischer und elektrischer Hinsicht geändert werden, ohne vom Gegenstand der Erfindung abzuweichen. Die verschiedenen Ausführungsbeispiele schließen sich nicht notwendig gegenseitig aus sondern es können verschiedene Ausführungsformen miteinander kombiniert werden, so dass neue Ausführungsformen entstehen. Im Rahmen dieser Beschreibung werden die Begriffe „verbunden“, „angeschlossen“ sowie „gekoppelt“ verwendet zum Beschreiben sowohl einer direkten als auch einer indirekten Verbindung, eines direkten oder indirekten Anschlusses sowie einer direkten oder indirekten Kopplung.

[0009] **Fig.** 1 zeigt eine Chipkarte 100.

[0010] Die Chipkarte 100 weist einen Träger 101 auf, auf dem ein Chipkartenmodul 102 angeordnet ist. Das Chipkartenmodul 102 ist beispielsweise ein Chip, der verschiedene Datenverarbeitungskomponenten, wie beispielsweise einen Arbeitsspeicher 103 (z.B. ein RAM (Random Access Memory), einen Prozessor 104 (auch bezeichnet als CPU für Central Processing Unit) und ein oder mehrere Register 105 aufweist. Der Prozessor 104 ist über Steuerleitungen 106 mit den Registern 105 und dem Arbeitsspeicher 103 verbunden. Der Prozessor 104 kann den Registern 105 und dem Arbeitsspeicher 103 über die Steuerleitungen Steuersignale wie beispielsweise Schreibaktivierungssignale zuführen, die bewirken, dass ein Register 105 oder der Arbeitsspeicher 103 an dem Register 105 bzw. an dem Arbeitsspeicher 103 anliegende, beispielsweise von dem Register 105 über Datenleitungen 107 zugeführte Daten, übernimmt.

[0011] Der Prozessor 103 und andere, über Datenleitungen 108 (wie beispielsweise über einen Datenbus) angeschlossene weitere Komponenten 109, wie beispielsweise eine Eingabe/Ausgabe-Schnittstelle, können Daten aus den Registern und dem Arbeitsspeicher lesen.

[0012] Das Chipkartenmodul 102 führt beispielsweise Operationen durch, die sicher sein sollen, d.h. die Operationen und/oder deren Ergebnisse, wie beispielsweise eine Schlüsselberechnungen, sollen vor Dritten geheim gehalten werden und sollen vor Manipulationen geschützt werden. In diesem Sinne soll das Chipkartenmodul 102 sicher gegen Angriffe sein.

[0013] Mikrokontroller wie das Chipkartenmodul 102 können mittels Redundanz gegen die gezielte Injektion von Fehlern geschützt werden. Dabei wird beispielsweise die CPU 104 mehrfach ausgeführt (d.h. es sind mehrere CPUs 104 vorgesehen), um damit Rechenfehler detektieren zu können. Um den Transport von Daten zu schützen, können diese mit einem Code versehen werden, der Manipulationen detektieren kann.

[0014] In dem Fall, dass Daten, beispielsweise bei der Ausführung eines Computerprogramms durch den Prozessor 104, über mehrere Takte hinweg transportiert werden (beispielsweise zwischen dem Prozessor 104 und einer weiteren Komponente 109) und für den Transport in Registern 105 oder dem Arbeitsspeicher 103 zwischengespeichert werden, könnte ein Angreifer versuchen, dass, wenn ein neues Datum transportiert werden soll, stattdessen ein altes Datum geliefert wird. Angriffspunkt hierfür wären die Steuerleitungen 106 von einem in dem Pfad, den die Daten durchlaufen (z.B. zwischen Prozessor 104 und der weiteren Komponente 109) liegenden Register 105 oder Arbeitsspeicher 103. Wenn, beispielsweise nach der Unterbrechung des Takts, der einem Register 105 zugeführt wird, das Register 105 keine neuen Werte von der Datenquelle (z.B. dem Prozessor 104) übernimmt, gibt es nur seinen bereits vorliegenden Wert an die Datensenke (z.B. eine weitere Komponente 109) weiter. Ist der alte Wert in sich konsistent (d.h. korrekt kodiert), so kann der Angreifer die Ausführung des Computerprogramms manipulieren werden, ohne dass es vom Computerprogramm bemerkt wird.

[0015] Ein Ziel für einen solchen Angriff wären z.B. Schleifenzähler, die ein Angreifer für eine gewisse Dauer immer wieder auf einen bestimmten Wert zurücksetzen könnte, wodurch die Programmausführung stark beeinflusst werden kann.

[0016] Gemäß einer Ausführungsform wird zur Abwehr solcher Angriffe die Ansteuerung (oder auch die Prüfung) der Steuerlogik eines Speichers (z.B. eines Registers 105 oder des Arbeitsspeichers 103) auf mehrere Prozessoren 104 aufgeteilt. Dies wird im Folgenden genauer erläutert.

[0017] **Fig.** 2 zeigt eine Datenverarbeitungsvorrichtung 200 gemäß einer Ausführungsform.

[0018] Die Datenverarbeitungsvorrichtung 200 weist eine Kodierungseinrichtung 201 auf, die eingerichtet ist, einen Datenvektor (in anderen Worten ein Datenwort) zu einem Codewort eines Codes zu kodieren, ein erstes Teilwort 202 des Codeworts einem ersten Speicherbereich 203 zuzuführen und ein zweites Teilwort 204 des Codeworts einem zweiten Speicherbereich 205 zuzuführen.

[0019] Die Datenverarbeitungsvorrichtung 200 weist ferner eine erste Schreibaktivierungssignalquelle 206, die eingerichtet ist, zum Speichern des ersten Teilworts 202 in dem ersten Speicherbereich 203 dem ersten Speicherbereich 203 ein erstes Schreibaktivierungssignal 207 zuzuführen, sowie eine zweite Schreibaktivierungssignalquelle 208 auf, die eingerichtet ist, zum Speichern des zweiten Teilworts 204 in dem zweiten Speicherbereich 205 dem zweiten Speicherbereich 205 ein zweites Schreibaktivierungssignal 209 zuzuführen.

[0020] Die Datenverarbeitungsvorrichtung 200 weist ferner eine Ausleseeinrichtung 210 auf, die eingerichtet ist, das erste Teilwort aus dem ersten Speicherbereich 203 auszulesen und des zweite Teilwort aus dem zweiten Speicherbereich 205 auszulesen und weist eine Überprüfungseinrichtung 211 auf, die eingerichtet ist, zu überprüfen, ob das erste ausgelesene Teilwort 212 und das zweite ausgelesene Teilwort 213 ein gültiges Codewort des Codes bilden (ggf. zusammen mit ein oder mehreren weiteren Teilworten, die in ein oder mehreren weiteren Speicherbereichen gespeichert wurden) und ein Fehlersignal 213 auszugeben, falls das erste ausgelesene Teilwort 211 und das zweite ausgelesene Teilwort 212 kein gültiges Codewort des Codes bilden.

[0021] In anderen Worten wird ein zu speichernder Datenvektor kodiert und das durch das Kodieren erzeugte Codewort in zwei (oder mehr) Teile aufgeteilt, die in unterschiedlichen Speicherbereichen, die von unterschiedlichen Einheiten (z.B. CPUs) angesteuert werden, gespeichert. Beim Auslesen werden die Speicherbereiche ausgelesen, die ausgelesenen Teilworte gemäß der Aufteilung des erzeugten Codeworts zusammengefügt (z.B. aneinandergehängt) und das entstandene (möglicherweise gegenüber dem gespeicherten Codewort geänderte) Datenwort wird überprüft, um festzustellen, ob es gültiges Codewort ist.

[0022] Für ein zu speicherndes Datenwort wird beispielsweise mittels eines linearen Codes (z.B. eines Simplex-Codes, eines Hamming-Codes oder eines Reed-Muller-Codes) ein Codewort erzeugt, wobei eine gewisse Redundanz eingeführt wird. Beispielsweise wird für ein Datenwort mit einer gewissen Länge ein Redundanzteil erzeugt, dessen Länge von der gewünschten Detektionswahrscheinlichkeit von Angriffen gewählt wird. Das Codewort (z.B. bestehend aus zu speicherndem Datenwort und Redundanzteil) wird in zwei Teilworte aufgeteilt und jedes Teilwort wird in einem Speicherbereich gespeichert. Die Überprüfungseinrichtung kann beispielsweise für das aus dem ersten ausgelesene Teilwort und aus dem zweiten ausgelesene Teilwort (und ggf. weiteren Teilworten) gebildete Datenwort mittels Multiplikation mit einer Kontrollmatrix gemäß dem Code überprüfen, ob es ein Codewort gemäß dem Code ist. Ist es kein Codewort, wird ein Alarmsignal ausgegeben, das beispielsweise bewirkt, dass die Datenverarbeitungsvorrichtung zurückgesetzt wird oder Komponenten der Datenverarbeitungsvorrichtung deaktiviert werden.

[0023] Ein Codewort kann auch in mehr als zwei Teilworte aufgeteilt werden. Entsprechend können mehr als zwei Speicherbereiche mit jeweils einer Schreibaktivierungssignalquelle vorgesehen sein.

[0024] In anderen Worten werden Angriffe, die auf der Unterbrechung Steuerleitungen (d.h. Schreibaktivierungsleitungen) basieren, beispielsweise auf der Unterbrechung eines Taktsignals oder der Unterdrückung eines Write-Enable-Signals, durch eine geeignete Redundanz der Schreibaktivierungssignale (z.B. Taktsignal oder Write-Enable-Signal) verhindert. In dem Fall, dass CPUs die Quellen der Schreibaktivierungssignale sind, ergibt sich die Stufe der Redundanz aus der Zahl der vorhandenen CPUs. So könnten beispielsweise so viele Registerbereiche oder Speicherbereiche vorgesehen werden, wie CPUs vorhanden sind. Gemäß einer Ausführungsform wird ein Register oder Arbeitsspeicher in mehrere, unabhängig steuerbare und adressierbare Bereiche aufgeteilt. Um eine Schreiboperation vollständig zu unterdrücken, muss ein Angreifer alle parallelen Schreibaktivierungssignale angreifen. Kann ein Angreifer nur eine Teilmenge der Signale unterdrücken, so kann er damit nur einen Fehler in den betroffenen Bereichen erwirken und erzeugt damit einen Fehler im gespeicherten Datenwort, der der XOR-Differenz zwischen dem neuen (zu speichernden) Codewort und dem alten (zuvor gespeicherten) Codewort in den betroffenen Bereichen entspricht. Abhängig von dem verwendeten Code (z.B. abhängig von seiner Hamming-Distanz) und der Anzahl und Verteilung der Fehler wird ein derart gestörtes Datenwort bei einer Überprüfung mit einer gewissen Wahrscheinlichkeit als inkorrekt detektiert.

[0025] Einem Registerbereich können auch mehrere CPUs zugeordnet werden. Beispielsweise wird ein Register (oder ein Arbeitsspeicher) in mehrere, unabhängig adressierbare Bereiche aufgeteilt und jedem Registerbereich werden zwei CPUs zugeordnet. Damit entspricht die Zahl der Registerbereiche der Hälfte der Anzahl von CPUs und die CPUs werden zu Paaren zusammengefasst und bei jedem Paar steuert eine CPU einen zugehörigen Registerbereich (mittels eines Schreibaktivierungssignals) an und die andere CPU des Paars prüft, ob dem Registerbereich das Steuersignal so zugeführt wurde, wie sie es auch selbst generiert. Zu diesem Zweck werden die Steuersignale zum Register hin- und dann wieder zurückgeführt.

[0026] Hierbei kann das Routing der zu einem Registerbereich hinführenden Steuerleitung und der

zugehörigen Rückleitung berücksichtigt werden. Liegen Steuerleitung und Rückleitung direkt nebeneinander, so kann ein Angreifer z.B. mittels eines Lasers beide Leitungen gleichzeitig attackieren. Gemäß einer Ausführungsform werden aus diesem Grund die Steuersignale komplementär übertragen, d.h. vor der Rückleitung vom Registerbereich zur zweiten CPU des CPU-Paares wird das Steuersignal invertiert. Unter der Annahme, dass ein Laser die Transition von 0 auf 1 unter anderen Umständen erwirkt als eine Transition von 1 auf 0, kann der Angreifer so nicht direkt beide Leitungen gleichzeitig beeinflussen.

[0027] Somit muss ein Angreifer zum Unterdrücken einer Schreiboperation auf einen Registerbereich zwei Einzelangriffe durchführen, damit dieser durch das Vergleichen der Steuersignale (durch die zweite CPU des CPU-Paares) nicht auffällt. Selbst wenn dem Angreifer dies gelingt, so kann durch das Schützen des zu speichernden Datenworts durch Kodieren wie mit Bezug auf **Fig.** 2 beschrieben eine erfolgreiche Attacke auf einen Speicherbereich detektiert werden. Manipuliert beispielsweise ein Angreifer die Steuersignale von nur einem CPU-Paar, d.h. für einen Registerbereich, und unterdrückt so das Speichern eines Teilworts in dem Registerbereich, so werden die ein oder mehreren andere Teilworte des (kodierten) Datenworts weiterhin geschrieben. Mittels eines fehlerdetektierender Code kann erkannt werden, dass das gespeicherte Wort in sich inkonsistent ist (d.h. es liegt eine Mischung aus einem altem und einem neuem Codewort vor). Somit müsste ein Angreifer so viele Einzelleitungen angreifen, wie CPUs vorhanden sind.

[0028] Die Komponenten der Datenverarbeitungsvorrichtung (z.B. Kodierungseinrichtung, Schreibaktivierungssignalquellen, Ausleseeinrichtung, Überprüfungseinrichtung etc.) können durch ein oder mehrere Schaltungen realisiert sein. In einer Ausführungsform ist eine Schaltung als jegliche Einheit zu verstehen, die eine Logik implementiert, und die sowohl Hardware, Software, Firmware oder eine Kombination daraus sein kann. Somit kann eine Schaltung in einer Ausführungsform ein hart-verdrahteter Logik-Schaltkreis oder ein programmierbarer Logik-Schaltkreis sein, wie beispielsweise ein programmierbarer Prozessor, z.B. ein Mikroprozessor. Unter einer Schaltung kann auch ein Prozessor zu verstehen sein, der Software ausführt.

[0029] Die Datenverarbeitungsvorrichtung führt beispielsweise ein Verfahren zum Speichern eines Datenvektors auf, wie es in **Fig.** 3 veranschaulicht ist.

[0030] **Fig.** 3 zeigt ein Ablaufdiagramm 300.

[0031] In 301 wird ein Datenvektor zu einem Codewort eines Codes kodiert.

[0032] In 302 wird ein erstes Teilwort des Codeworts einem ersten Speicherbereich zugeführt.

[0033] In 303 wird ein zweites Teilwort des Codeworts einem zweiten Speicherbereich zugeführt.

[0034] In 304 wird zum Speichern des ersten Teilworts in dem ersten Speicherbereich ein erstes Schreibaktivierungssignal dem ersten Speicherbereich zugeführt.

[0035] In 305 wird zum Speichern des zweiten Teilworts in dem zweiten Speicherbereich ein zweites Schreibaktivierungssignal dem zweiten Speicherbereich zugeführt.

[0036] In 306 wird das erste Teilwort aus dem ersten Speicherbereich ausgelesen und das zweite Teilwort aus dem zweiten Speicherbereich ausgelesen.

[0037] In 307 wird überprüft, ob das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort ein gültiges Codewort des Codes bilden.

[0038] In 308 wird ein Fehlersignal ausgegeben, falls das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort kein gültiges Codewort des Codes bilden.

[0039] Im Folgenden werden verschiedene Ausführungsformen angegeben.

[0040] Ausführungsform 1 ist eine Datenverarbeitungsvorrichtung, wie sie in **Fig.** 2 dargestellt ist.

[0041] Ausführungsform 2 ist eine Datenverarbeitungsvorrichtung gemäß Ausführungsform 1, aufweisend einen Speicher, der den ersten Speicherbereich und den zweiten Speicherbereich aufweist, wobei der erste Speicherbereich eingerichtet ist, als Reaktion auf das erste Schreibaktivierungssignal ein dem ersten Speicherbereich zugeführtes Datenwort zu speichern und wobei der zweite Speicherbereich eingerichtet ist, als Reaktion auf das zweite Schreibaktivierungssignal ein dem ersten Speicherbereich zugeführtes Datenwort zu speichern.

[0042] Ausführungsform 3 ist eine Datenverarbeitungsvorrichtung gemäß Ausführungsform 2, wobei der Speicher ein Register oder ein Arbeitsspeicher ist.

[0043] Ausführungsform 4 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 3, wobei das erste Schreibaktivierungssignal und das zweite Schreibaktivierungssignal ein Taktsignal oder ein Write-Enable-Signal sind.

[0044] Ausführungsform 5 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsfor-

men 1 bis 4, wobei das erste Schreibaktivierungssignal und das zweite Schreibaktivierungssignal dasselbe Signal ist.

[0045] Ausführungsform 6 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 4, wobei das erste Schreibaktivierungssignal und das zweite Schreibaktivierungssignal unterschiedliche Signale sind.

[0046] Ausführungsform 7 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 6, wobei der erste Speicherbereich mehrere erste Speicherelemente aufweist, wobei jedes erste Speicherelement eingerichtet ist, ein dem Speicherelement zugeführtes Datenbit in Reaktion auf das erste Schreibaktivierungssignal zu speichern, und wobei der zweite Speicherbereich mehrere zweite Speicherelemente aufweist, wobei jedes zweite Speicherelement eingerichtet ist, ein dem Speicherelement zugeführtes Datenbit in Reaktion auf das zweite Schreibaktivierungssignal zu speichern.

[0047] Ausführungsform 8 ist eine Datenverarbeitungsvorrichtung gemäß Ausführungsform 7, wobei die erste Schreibaktivierungssignalquelle eingerichtet ist, das erste Schreibaktivierungssignal den ersten Speicherelementen zuzuführen und wobei die zweite Schreibaktivierungssignalquelle eingerichtet ist, das zweite Schreibaktivierungssignal den zweiten Speicherelementen zuzuführen.

[0048] Ausführungsform 9 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 8, wobei die Kodierungseinrichtung eingerichtet ist, das durch Kodieren des Datenvektors erzeugte Codewort in mindestens das erste Teilwort und das zweite Teilwort aufzuteilen.

[0049] Ausführungsform 10 ist eine Datenverarbeitungsvorrichtung gemäß Ausführungsform 9, wobei die Überprüfungseinrichtung eingerichtet ist, das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort gemäß der Aufteilung des erzeugten Codeworts zu einem Datenwort zusammenzufügen und zu überprüfen, ob das Datenwort ein gültiges Codewort des Codes ist.

[0050] Ausführungsform 11 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 10, wobei die erste Schreibaktivierungssignalquelle eine erste CPU ist und die zweite Schreibaktivierungssignalquelle eine zweite CPU ist.

[0051] Ausführungsform 12 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 11, aufweisend eine erste Schreibaktivierungssignalkontrolleinrichtung, die eingerichtet ist, zu überprüfen, ob die erste Schreibaktivierungssignalquelle dem ersten Speicherbereich das erste Schreibaktivierungssignal zuführt und ein Fehlersignal auszugeben, falls die erste Schreibaktivierungssignalquelle dem ersten Speicherbereich das erste Schreibaktivierungssignal nicht zuführt.

[0052] Ausführungsform 13 ist eine Datenverarbeitungsvorrichtung gemäß Ausführungsform 12, wobei die erste Schreibaktivierungssignalkontrolleinrichtung eingerichtet ist, das erste Schreibaktivierungssignal selbst zu erzeugen und zu überprüfen, ob dem ersten Speicherbereich das erste Schreibaktivierungssignal von der ersten Schreibaktivierungssignalquelle zugeführt wird.

[0053] Ausführungsform 14 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 13, aufweisend eine zweite Schreibaktivierungssignalkontrolleinrichtung, die eingerichtet ist, zu überprüfen, ob die zweite Schreibaktivierungssignalquelle dem ersten Speicherbereich das zweite Schreibaktivierungssignal zuführt und ein Fehlersignal auszugeben, falls die zweite Schreibaktivierungssignalquelle dem zweite Speicherbereich das zweite Schreibaktivierungssignal nicht zuführt.

[0054] Ausführungsform 15 ist eine Datenverarbeitungsvorrichtung gemäß Ausführungsform 14, wobei die zweite Schreibaktivierungssignalkontrolleinrichtung eingerichtet ist, das zweite Schreibaktivierungssignal selbst zu erzeugen und zu überprüfen, ob dem zweiten Speicherbereich das zweite Schreibaktivierungssignal von der zweiten Schreibaktivierungssignalquelle zugeführt wird.

[0055] Ausführungsform 16 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 15, wobei der Code ein linearer Code ist.

[0056] Ausführungsform 17 ist eine Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis 16, wobei die Kodierungseinrichtung eingerichtet ist, den Datenvektor in drei oder mehr Teilworte aufzuteilen und jedes Teilwort des Codeworts einem jeweiligen Speicherbereich von drei oder mehr Speicherbereichen zuzuführen, die Datenverarbeitungsvorrichtung drei oder mehr Schreibaktivierungssignalquellen aufweist, wobei jede Schreibaktivierungssignalquelle eingerichtet ist, zum Speichern des jeweiligen Teilworts in einem jeweiligen Speicherbereich der Speicherbereiche dem Speicherbereich ein jeweiliges Schreibaktivierungssignal zuzuführen, die Ausleseeinrichtung eingerichtet ist, die Teilworte aus den Speicherbereichen auszulesen und die Überprüfungseinrichtung eingerichtet ist, zu überprüfen, ob die ausgelesenen Teilworte ein gültiges Codewort des Codes bilden und ein Fehlersignal auszugeben, falls die ausgelesenen Teilworte kein gültiges Codewort des Codes bilden.

[0057] Ausführungsform 18 ist eine Chipkarte mit einer Datenverarbeitungsvorrichtung gemäß einer der Ausführungsformen 1 bis Ausführungsform 17.

[0058] Ausführungsform 19 ist ein Verfahren zum Speichern eines Datenvektors, wie es in **Fig.** 3 dargestellt ist.

[0059] Es sollte beachtet werden, dass jede beschriebene Ausführungsform mit jeder anderen beschriebenen Ausführungsform kombiniert werden kann.

[0060] Im Folgenden werden Ausführungsformen in größerem Detail beschrieben.

[0061] **Fig.** 4 zeigt Speicheranordnung 400 gemäß einer Ausführungsform mit einer Aufteilung eines Registers in Registerbereiche, die von unterschiedlichen CPUs angesteuert werden.

[0062] Die Speicheranordnung 400 weist vier CPUs 401 auf, die beispielsweise dem Prozessor 104 entsprechen (d.h. es sind vier CPUs anstatt eines Prozessors vorgesehen) und weist ein 8-bit Register 402 auf, das beispielsweise dem Register 105 entspricht.

[0063] Jede CPU 401 erzeugt das gleiche Steuersignal ctrl, z.B. ein Set-Signal oder ein Taktsignal für acht Flip-Flops des Register 402 und steuert ein Viertel des Registers 402 an, z.B. zwei Flip-Flops des Registers 402. In anderen Worten wird das Register 402 in mehrere Speicherbereiche unterteilt, entsprechend der Speicherbereiche 203 und 205.

[0064] Ist ein Datenwort zu speichern, so wird dieses Datenwort (z.B. von einer der CPUs 401) zu einem 8-Bit-Codewort mit der binären Darstellung d0 d1 d2 d3 d4 d5 d6 d7 kodiert und jedem Flip-Flop des Registers 402 wird ein Bit zugeführt. Beim Auslesen des Registers 201 wird überprüft (z.B. von einer Überprüfungseinrichtung entsprechend der Überprüfungseinrichtung 211, die auch durch eine CPU 401 realisiert sein kann), ob die ausgelesenen Bits ein Codewort bilden. Ist dies nicht der Fall, wird das so interpretiert, dass das Steuersignal ctrl von mindestens einer CPU 401 dem entsprechenden Viertel (d.h. Registerbereich) des Registers 402 nicht korrekt zugeführt wurde und es wird ein Fehlersignal ausgegeben.

[0065] **Fig.** 5 zeigt eine Speicheranordnung 500 gemäß einer Ausführungsform mit einer Aufteilung eines Arbeitsspeichers in Speicherbereiche, die von unterschiedlichen CPUs angesteuert werden.

[0066] Ähnlich wie bei der Speicheranordnung 400 weist die Speicheranordnung 500 vier CPUs 501 auf, die beispielsweise dem Prozessor 104 entsprechen und weist einen Arbeitsspeicher (RAM) 502 auf der beispielsweise dem Arbeitsspeicher 103 entspricht und das parallele Speichern von 8 Bits ermöglicht.

[0067] Der Arbeitsspeicher 502 ist in vier Speicherbereiche 503 aufgeteilt (analog zu den Speicherbereichen 203 und 205) und jede CPU 501 erzeugt das gleiche Steuersignal ctrl und führt es einem der Speicherbereiche 503 über einen jeweiligen Steuerpfad 504 (d.h. eine jeweilige Steuerleitung) zu.

[0068] Ist ein Datenwort zu speichern, so wird dieses Datenwort (z.B. von einer der CPUs 501) zu einem 8-Bit-Codewort mit der binären Darstellung d0 d1 d2 d3 d4 d5 d6 d7 kodiert und jeweils zwei Datenbit werden über jeweilige Dateneingangspfade 505 (d.h. Datenleitungen) einem Speicherbereich 503 zugeführt, wobei jeder Dateneingangspfad 505 mittels eines jeweiligen UND-Gatters mit dem Steuersignal „gegatet“ (d.h. UND-verknüpft) wird. Das Steuersignal ist in diesem Beispiel entsprechend ein Write-Enable-Signal.

[0069] Im Folgenden werden mit Bezug auf **Fig.** 6 und 7 Erweiterungen erläutert, die für die Anwendung auf ein Register dargestellt sind, analog aber auch auf einen Arbeitsspeicher angewendet werden können (analog der Anwendung der Vorgehensweise aus **Fig.** 4 auf einen Arbeitsspeicher, wie in **Fig.** 5 gezeigt).

[0070] **Fig.** 6 zeigt eine Speicheranordnung 600 gemäß einer weiteren Ausführungsform mit einer Aufteilung eines Registers in Registerbereiche, die von unterschiedlichen CPUs angesteuert werden.

[0071] Die Speicheranordnung 600 weist vier CPUs 601, 602 auf, die beispielsweise dem Prozessor 104 entsprechen und weist ein 8-bit Register 603 auf, das beispielsweise dem Register 105 entspricht und in diesem Beispiel in zwei Registerbereiche 604 geteilt ist.

[0072] Jeweils zwei CPUs 601, 602 sind zu einem Paar (CPU0 und CPU1 sowie CPU 2 und CPU3) zusammengefasst. Jedes CPU-Paar ist einem Registerbereich 604 zugeordnet und jeweils die erste CPU 601 jedes CPU-Paars erzeugt das Steuersignal ctrl und führt es dem zugeordneten Registerpaar zu.

[0073] Das Steuersignal wird mittels eines jeweiligen ersten Inverters 605 invertiert und einem Vergleicher 606 zugeführt, dem von der zweiten CPU 602 über einen jeweiligen zweiten Inverter 607 ebenfalls das Steuersignal zugeführt wird. Stimmen die beiden invertierten Versionen des Steuersignals nicht überein, so gibt der Vergleicher 606 ein Fehlersignal oder Alarmsignal aus.

[0074] In anderen Worten wird mittels der zweiten CPU 602 das zurückgeführte Steuersignal und das von der zweiten CPU 602 selbst generierte Steuersignal auf Konsistenz geprüft und bei einer Abweichung ein Alarmsignal erzeugt.

[0075] Ein Angreifer muss somit mindestens zwei Leitungspaare (jeweils hinführende und rückführende Steuersignalleitung) angreifen. Das Invertieren durch die Inverter 605, 606 erhöht den Schutz gegen Laserangriffe: Liegen irgendwo zwischen CPUs 601, 602 und Register 603 alle Steuersignalleitungen direkt nebeneinander (bedingt durch automatisches Routing), so kann ein einzelner Laserspot alle Steuersignalleitungen gleichzeitig beeinflussen. Durch das Invertieren muss ein Angreifer aber nun unterschiedliche Transitionen für die hinführende Leitung und die rückführende Leitung (z.B. 0->1 für CPU0 und CPU2, 1->0 für CPU1 und CPU3) bewirken, was nicht mit einem einfachen Angriff möglich ist.

[0076] Analog zu dem mit Bezug auf **Fig.** 4 beschriebenen Beispiel wird ein Codewort in dem Register 604 gespeichert und beim Auslesen des Datenworts aus dem Register geprüft, ob es ein gültiges Codewort ist.

[0077] **Fig.** 7 zeigt eine Speicheranordnung 700 gemäß einer weiteren Ausführungsform mit einer Aufteilung eines Registers in Registerbereiche, die von unterschiedlichen CPUs angesteuert werden.

[0078] Die Speicheranordnung 700 weist vier CPUs 701, 702 auf, die beispielsweise dem Prozessor 104 entsprechen und weist ein 64-bit Register 703 auf, das beispielsweise dem Register 105 entspricht und in diesem Beispiel in zwei 32-Bit-Registerbereiche 704 geteilt ist.

[0079] Zum Speichern eines Datenworts wird das Datenwort in ein Codewort mit einem 32-bit Datenanteil (d0..d31), der gleich dem zu speichernden Datenwort ist, und einen 32-Bit Redundanzanteil (r0..r31) kodiert. Der verwendete Code ist somit ein [64,32,D] Code (wobei D die Hamming-Distanz des Codes ist).

[0080] Analog zu dem Beispiel von **Fig.** 6 wird jeder Registerbereich 704 von einem CPU-Paar angesteuert. In diesem Beispiel ist jedoch kein Rückkanal für das Steuersignal zur zweiten CPU 702 des Paars vorgesehen und auch auf das Invertieren wird verzichtet.

[0081] Stattdessen führen alle CPUs 701, 702 das gleiche Steuersignal dem Register 703 zu und für jeden Registerbereich 704 ist ein Vergleicher 705 vorgesehen, der den Zustand der von der ersten CPU 701 kommenden Steuersignalleitung mit dem Zustand der von der zweiten CPU 702 kommenden Steuersignalleitung vergleicht und bei Abweichung ein Alarmsignal ausgibt.

[0082] Bei Aufteilung des Registers 703 in mehr als zwei Registerbereiche kann gewährleistet werden, dass Fehlermuster, die durch den Ausfall eines CPU-Paares (d.h. nicht korrektes Schreiben für einen Registerbereich) im Codewort entstehen, immer erkannt werden. Dazu können Permutationen 706 auf die zu speichernden Bits des Codeworts angewandt werden. Die Aufgabe der Permutationen 706 ist eine Sortierung der Bits eines Codeworts so vorzunehmen, dass Fehler in einem der Registerbereiche kein korrektes Codewort erzeugen können. Die Permutationen 706 werden beim Schreiben des Registers 703 durchlaufen, beim Auslesen werden sie entsprechend invertiert. Die Permutationen 706 werden abhängig von der konkreten Matrix gewählt, die zur Prüfung des ausgelesenen Datenworts verwendet wird. Wie in den obigen Beispielen wird für ein ausgelesenes Datenwort überprüft, ob es ein gültiges Codewort ist und, falls dies nicht so ist, ein Alarmsignal ausgegeben.

[0083] Obwohl die Erfindung vor allem unter Bezugnahme auf bestimmte Ausführungsformen gezeigt und beschrieben wurde, sollte es von denjenigen, die mit dem Fachgebiet vertraut sind, verstanden werden, dass zahlreiche Änderungen bezüglich Ausgestaltung und Details daran vorgenommen werden können, ohne vom Wesen und Bereich der Erfindung, wie er durch die nachfolgenden Ansprüche definiert wird, abzuweichen. Der Bereich der Erfindung wird daher durch die angefügten Ansprüche bestimmt, und es ist beabsichtigt, dass sämtliche Änderungen, welche unter den Wortsinn oder den Äquivalenzbereich der Ansprüche fallen, umfasst werden.

Patentansprüche

1. Datenverarbeitungsvorrichtung (200) aufweisend:
eine Kodierungseinrichtung (201), die eingerichtet ist, einen Datenvektor zu einem Codewort eines Codes zu kodieren, das durch Kodieren des Datenvektors erzeugte Codewort in mindestens ein erstes Teilwort (202) und ein zweites Teilwort (204) aufzuteilen, das erste Teilwort (202) des Codeworts einem ersten Speicherbereich (203) zuzuführen und das zweite Teilwort (204) des Codeworts einem zweiten Speicherbereich (205) zuzuführen;
eine erste Schreibaktivierungssignalquelle (206), die eingerichtet ist, zum Speichern des ersten Teilworts (202) in dem ersten Speicherbereich (203) dem ersten Speicherbereich (203) ein erstes Schreibaktivierungssignal (207) zuzuführen;
eine zweite Schreibaktivierungssignalquelle (208), die von der ersten Schreibaktivierungssignalquelle (206) verschieden ist und eingerichtet ist, zum Spei-

chern des zweiten Teilworts (204) in dem zweiten Speicherbereich (205) dem zweiten Speicherbereich (205) ein zweites Schreibaktivierungssignal (209) zuzuführen;
eine Ausleseeinrichtung, die eingerichtet ist, das erste Teilwort (202) aus dem ersten Speicherbereich (203) auszulesen und das zweite Teilwort (204) aus dem zweiten Speicherbereich (205) auszulesen; und
eine Überprüfungseinrichtung, die eingerichtet ist, das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort gemäß der Aufteilung des erzeugten Codeworts zu einem Datenwort zusammenzufügen, zu überprüfen, ob das Datenwort ein gültiges Codewort des Codes ist, und ein Fehlersignal auszugeben, falls das Datenwort kein gültiges Codewort des Codes bildet.

2. Datenverarbeitungsvorrichtung (200) gemäß Anspruch 1, aufweisend einen Speicher, der den ersten Speicherbereich (203) und den zweiten Speicherbereich (205) aufweist, wobei der erste Speicherbereich (203) eingerichtet ist, als Reaktion auf das erste Schreibaktivierungssignal (207) ein dem ersten Speicherbereich (203) zugeführtes Datenwort zu speichern und wobei der zweite Speicherbereich (205) eingerichtet ist, als Reaktion auf das zweite Schreibaktivierungssignal (209) ein dem zweiten Speicherbereich (205) zugeführtes Datenwort zu speichern.

3. Datenverarbeitungsvorrichtung (200) gemäß Anspruch 2, wobei der Speicher ein Register oder ein Arbeitsspeicher ist.

4. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 3, wobei das erste Schreibaktivierungssignal (207) und das zweite Schreibaktivierungssignal (209) ein Taktsignal oder ein Write-Enable-Signal sind.

5. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 4, wobei das erste Schreibaktivierungssignal (207) und das zweite Schreibaktivierungssignal (209) dasselbe Signal ist.

6. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 4, wobei das erste Schreibaktivierungssignal (207) und das zweite Schreibaktivierungssignal (209) unterschiedliche Signale sind.

7. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 6, wobei der erste Speicherbereich (203) mehrere erste Speicherelemente aufweist, wobei jedes erste Speicherelement eingerichtet ist, ein dem Speicherelement zugeführtes Datenbit in Reaktion auf das erste Schreibaktivierungssignal (207) zu speichern, und wobei der zweite Speicherbereich (205) mehrere zweite Speicherelemente aufweist, wobei jedes zweite Speicherelement eingerichtet ist, ein dem Speicherelement zugeführtes Datenbit in Reaktion auf das zweite Schreibaktivierungssignal (209) zu speichern.

8. Datenverarbeitungsvorrichtung (200) gemäß Anspruch 7, wobei die erste Schreibaktivierungssignalquelle eingerichtet ist, das erste Schreibaktivierungssignal (207) den ersten Speicherelementen zuzuführen und wobei die zweite Schreibaktivierungssignalquelle eingerichtet ist, das zweite Schreibaktivierungssignal (209) den zweiten Speicherelementen zuzuführen.

9. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 8, wobei die erste Schreibaktivierungssignalquelle (206) eine erste CPU ist und die zweite Schreibaktivierungssignalquelle (208) eine zweite CPU ist.

10. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 9, aufweisend eine erste Schreibaktivierungssignalkontrolleinrichtung, die eingerichtet ist, zu überprüfen, ob die erste Schreibaktivierungssignalquelle (206) dem ersten Speicherbereich (203) das erste Schreibaktivierungssignal (207) zuführt und ein Fehlersignal auszugeben, falls die erste Schreibaktivierungssignalquelle (206) dem ersten Speicherbereich (203) das erste Schreibaktivierungssignal (207) nicht zuführt.

11. Datenverarbeitungsvorrichtung (200) gemäß Anspruch10, wobei die erste Schreibaktivierungssignalkontrolleinrichtung eingerichtet ist, das erste Schreibaktivierungssignal (207) selbst zu erzeugen und zu überprüfen, ob dem ersten Speicherbereich (203) das erste Schreibaktivierungssignal (207) von der ersten Schreibaktivierungssignalquelle (206) zugeführt wird.

12. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 11, aufweisend eine zweite Schreibaktivierungssignalkontrolleinrichtung, die eingerichtet ist, zu überprüfen, ob die zweite Schreibaktivierungssignalquelle (208) dem zweiten Speicherbereich (205) das zweite Schreibaktivierungssignal (209) zuführt und ein Fehlersignal auszugeben, falls die zweite Schreibaktivierungssignalquelle (208) dem zweiten Speicherbereich (205) das zweite Schreibaktivierungssignal (209) nicht zuführt.

13. Datenverarbeitungsvorrichtung (200) gemäß Anspruch 12, wobei die zweite Schreibaktivierungssignalkontrolleinrichtung eingerichtet ist, das zweite Schreibaktivierungssignal (209) selbst zu erzeugen und zu überprüfen, ob dem zweiten Speicherbereich (205) das zweite Schreibaktivierungssignal (209) von der zweiten Schreibaktivierungssignalquelle (208) zugeführt wird.

14. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 13, wobei der Code ein linearer Code ist.

15. Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 14, wobei die Kodierungseinrichtung (201) eingerichtet ist, den Datenvektor in drei oder mehr Teilworte aufzuteilen und jedes Teilwort des Codeworts einem jeweiligen Speicherbereich von drei oder mehr Speicherbereichen zuzuführen, die Datenverarbeitungsvorrichtung drei oder mehr Schreibaktivierungssignalquellen aufweist, wobei jede Schreibaktivierungssignalquelle eingerichtet ist, zum Speichern des jeweiligen Teilworts in einem jeweiligen Speicherbereich der Speicherbereiche dem Speicherbereich ein jeweiliges Schreibaktivierungssignal zuzuführen, die Ausleseeinrichtung eingerichtet ist, die Teilworte aus den Speicherbereichen auszulesen und die Überprüfungseinrichtung eingerichtet ist, zu überprüfen, ob die ausgelesenen Teilworte ein gültiges Codewort des Codes bilden und ein Fehlersignal auszugeben, falls die ausgelesenen Teilworte kein gültiges Codewort des Codes bilden.

16. Chipkarte mit einer Datenverarbeitungsvorrichtung (200) gemäß einem der Ansprüche 1 bis 15.

17. Verfahren zum Speichern eines Datenvektors aufweisend:
Kodieren eines Datenvektors zu einem Codewort eines Codes (301);
Aufteilen des erzeugten Codeworts in mindestens ein erstes Teilwort und ein zweites Teilwort;
Zuführen eines ersten Teilworts des Codeworts zu einem ersten Speicherbereich (302);
Zuführen eines zweiten Teilworts des Codeworts zu einem zweiten Speicherbereich (303);
Zuführen eines ersten Schreibaktivierungssignals einer ersten Schreibaktivierungssignalquelle zu dem ersten Speicherbereich zum Speichern des ersten Teilworts in dem ersten Speicherbereich (304);
Zuführen eines zweiten Schreibaktivierungssignals einer zweiten Schreibaktivierungssignalquelle, die von der ersten Schreibaktivierungssignalquelle verschieden ist, zu dem zweiten Speicherbereich zum Speichern des zweiten Teilworts in dem zweiten Speicherbereich (305); Auslesen des ersten Teilworts aus dem ersten Speicherbereich und des zweiten Teilworts aus dem zweiten Speicherbereich (306);
Zusammenfügen des ersten ausgelesenen Teilworts und des zweiten ausgelesenen Teilworts gemäß der Aufteilung des erzeugten Codeworts zu einem Datenwort;
Überprüfen, ob das Datenwort ein gültiges Codewort des Codes bildet; und
Ausgeben eines Fehlersignals, falls das Datenwort kein gültiges Codewort des Codes bildet.

Es folgen 7 Seiten Zeichnungen

Anhängende Zeichnungen

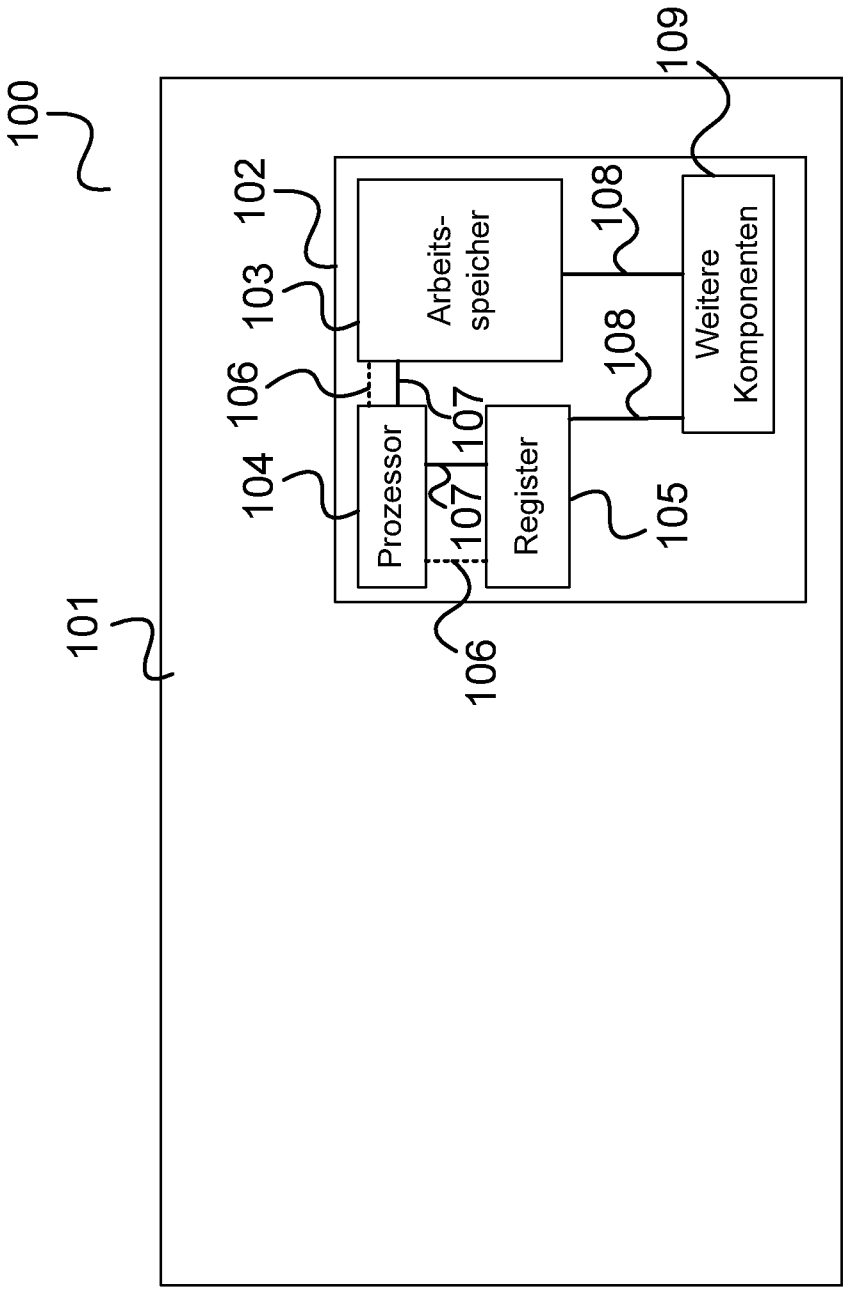


FIG 1

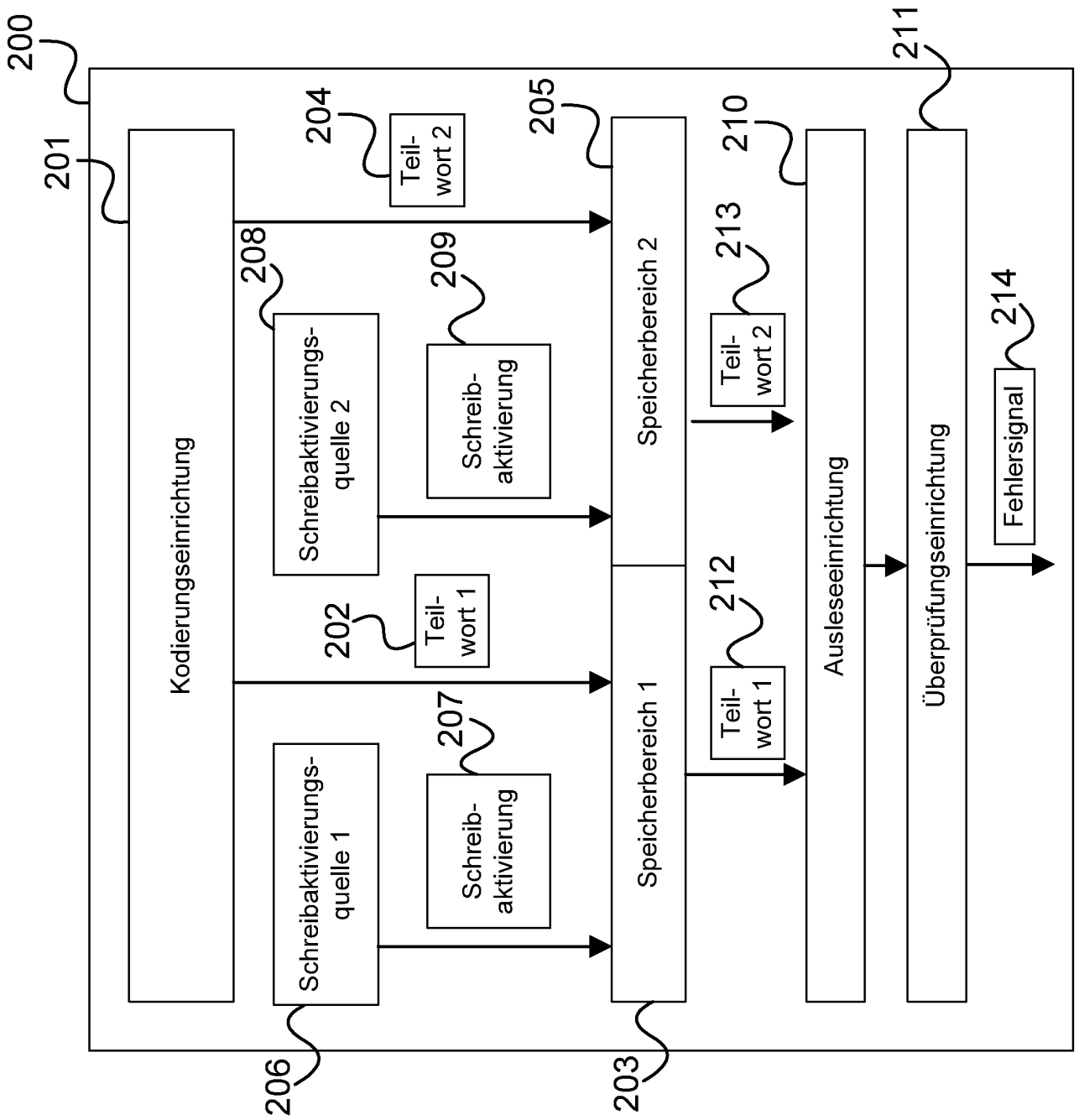
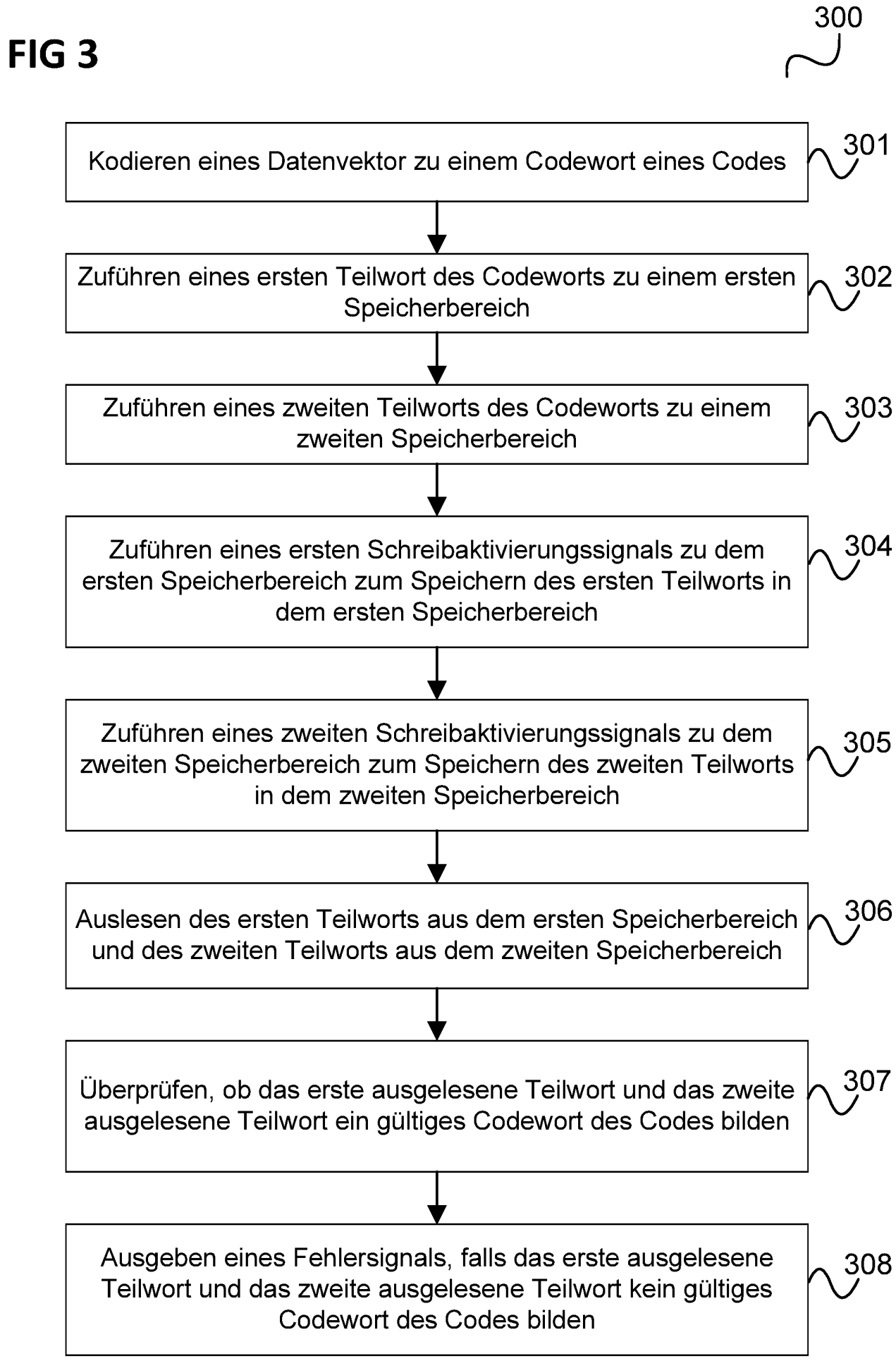

200
201
Kodierungseinrichtung
204
Teil-
wort 2
205
Speicherbereich 2
210
211
208
Schreibaktivierungs-
quelle 2
209
Schreib-
aktivierung
213
Teil-
wort 2
214
Fehlersignal
202
Teil-
wort 1
212
Teil-
wort 1
Ausleseeinrichtung
Überprüfungseinrichtung
207
Schreib-
aktivierung
Speicherbereich 1
Schreibaktivierungs-
quelle 1
206
203

FIG 2

FIG 3


300
Kodieren eines Datenvektor zu einem Codewort eines Codes
301
Zuführen eines ersten Teilwort des Codeworts zu einem ersten Speicherbereich
302
Zuführen eines zweiten Teilworts des Codeworts zu einem zweiten Speicherbereich
303
Zuführen eines ersten Schreibaktivierungssignals zu dem ersten Speicherbereich zum Speichern des ersten Teilworts in dem ersten Speicherbereich
304
Zuführen eines zweiten Schreibaktivierungssignals zu dem zweiten Speicherbereich zum Speichern des zweiten Teilworts in dem zweiten Speicherbereich
305
Auslesen des ersten Teilworts aus dem ersten Speicherbereich und des zweiten Teilworts aus dem zweiten Speicherbereich
306
Überprüfen, ob das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort ein gültiges Codewort des Codes bilden
307
Ausgeben eines Fehlersignals, falls das erste ausgelesene Teilwort und das zweite ausgelesene Teilwort kein gültiges Codewort des Codes bilden
308

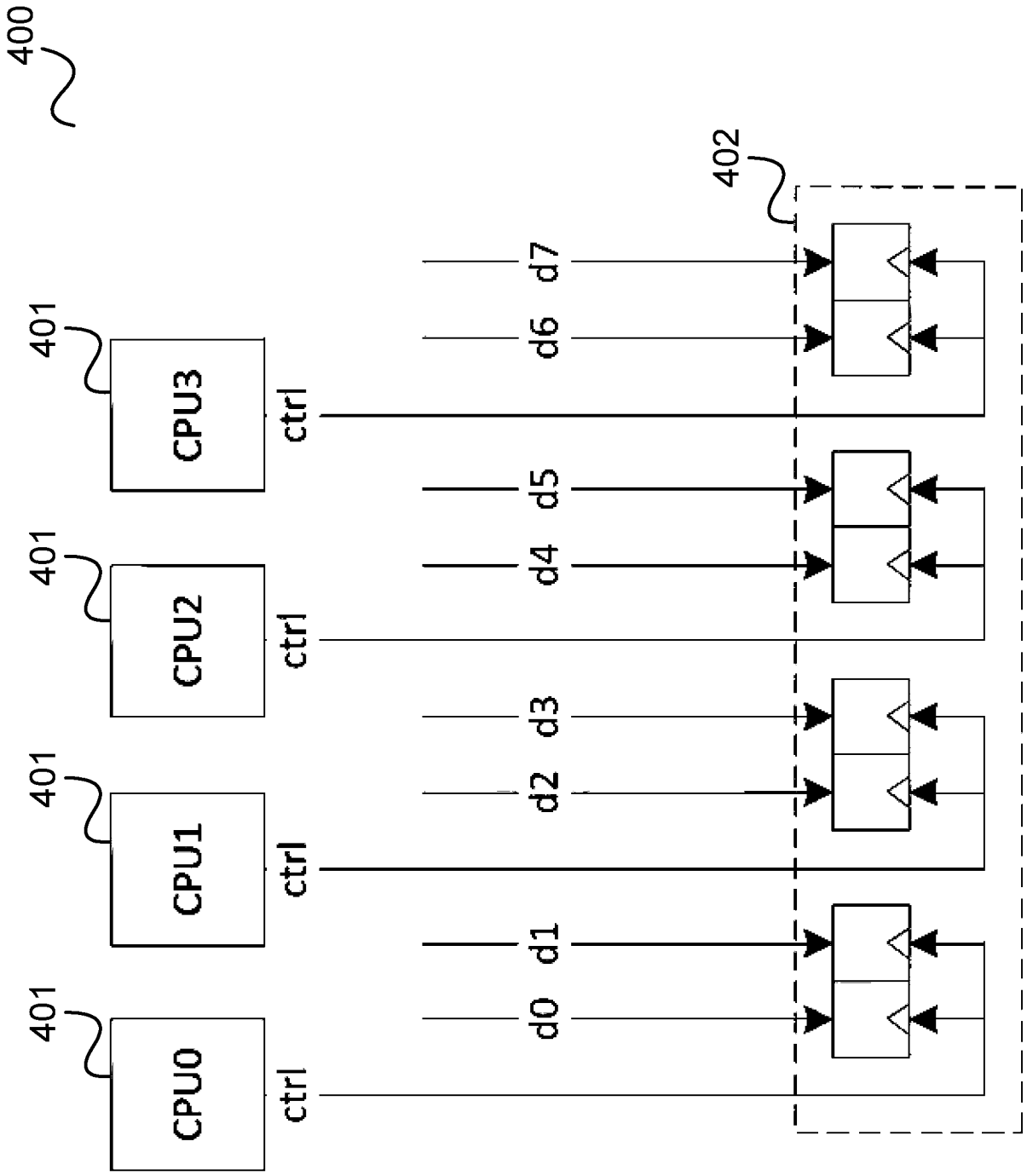

400
402
401
CPU3
401
CPU2
401
CPU1
401
CPU0
ctrl
ctrl
ctrl
ctrl
d7
d6
d5
d4
d3
d2
d1
d0

FIG 4

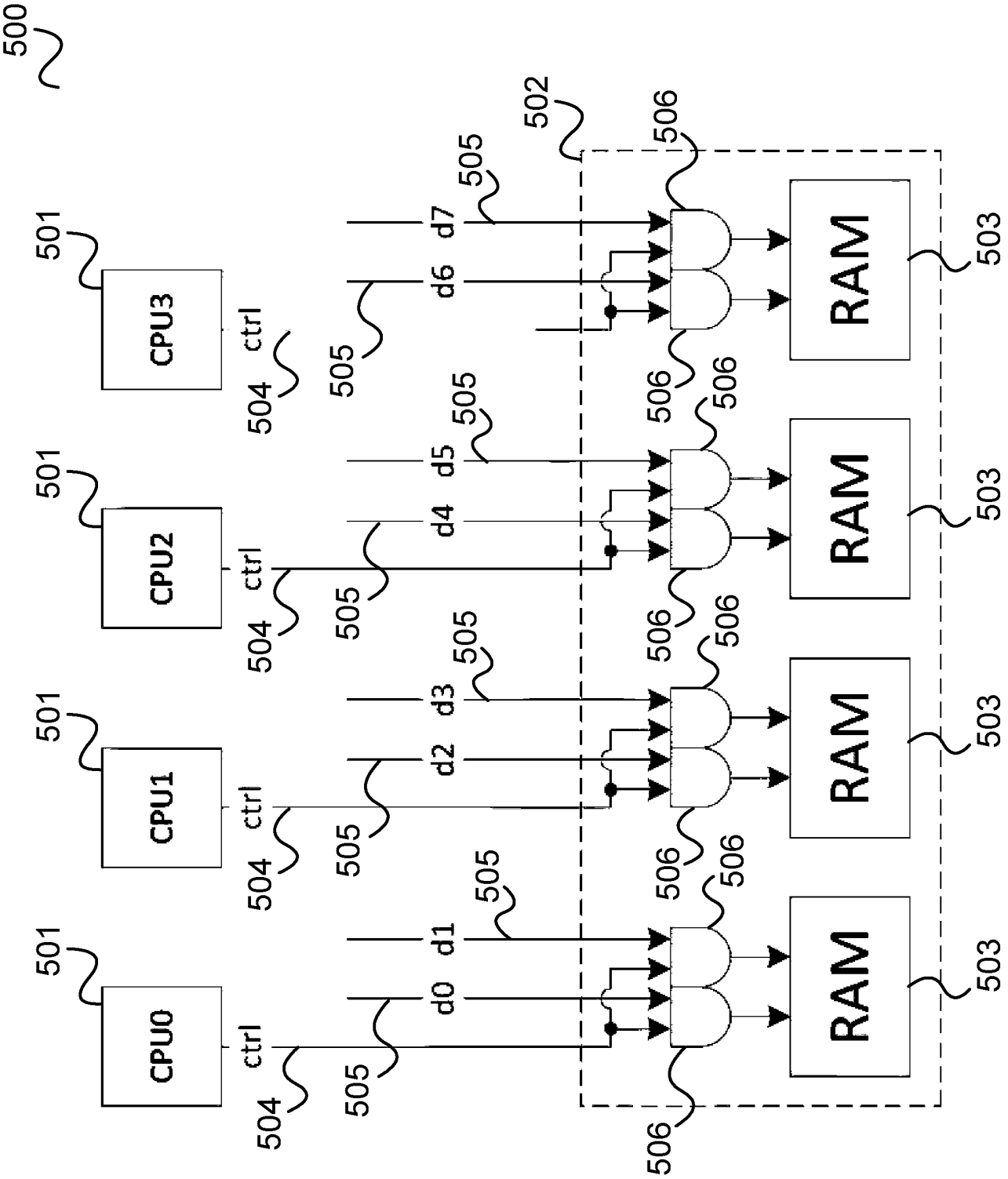

500
502
501
CPU3
CPU2
CPU1
CPU0
ctrl
504
505
d7
d6
d5
d4
d3
d2
d1
d0
506
RAM
503

FIG 5

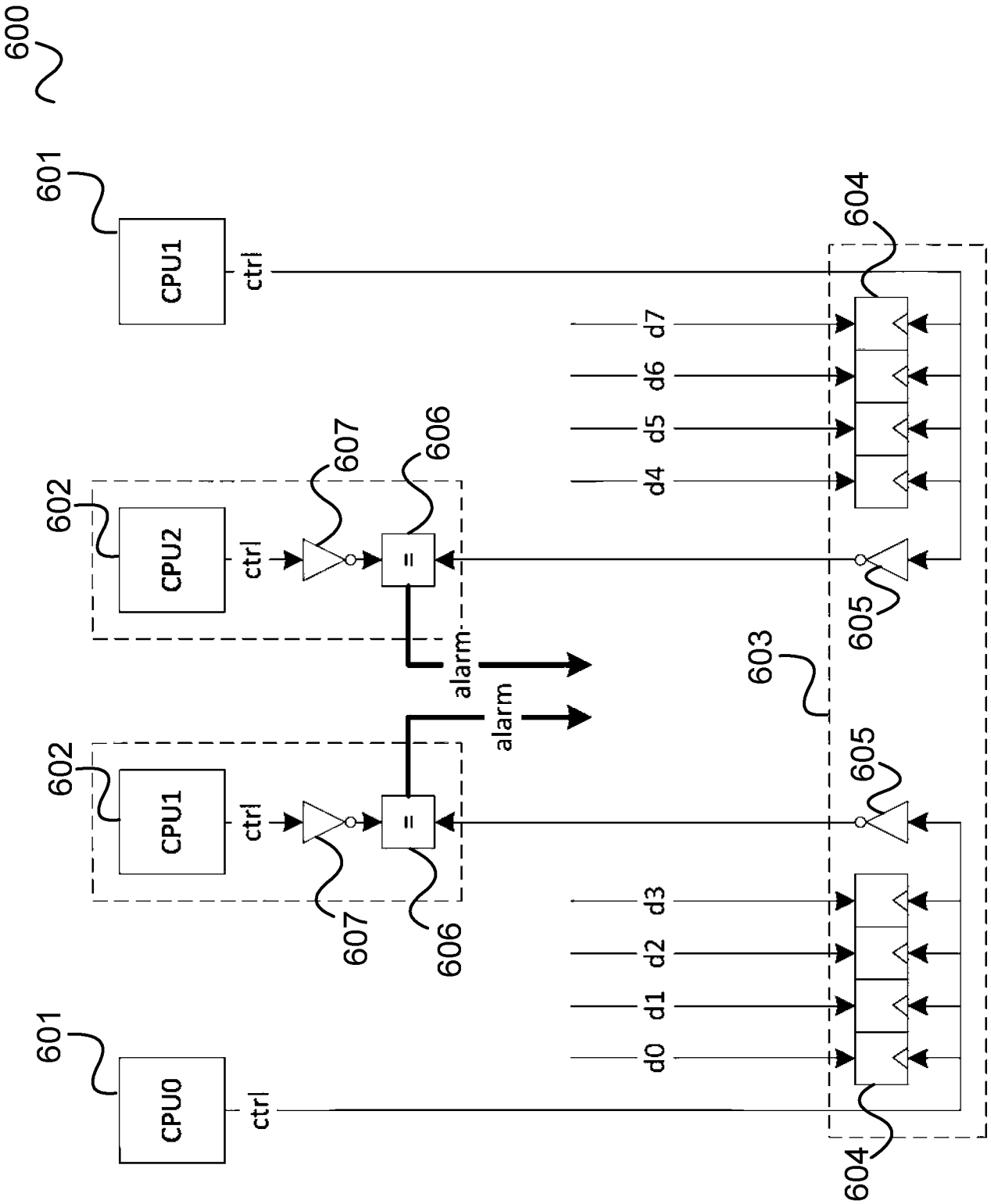

600
601
CPU1
ctrl
604
d7
d6
d5
d4
602
CPU2
ctrl
607
606
=
605
alarm
603
alarm
602
CPU1
ctrl
=
605
607
606
d3
d2
d1
d0
601
CPU0
ctrl
604

FIG 6

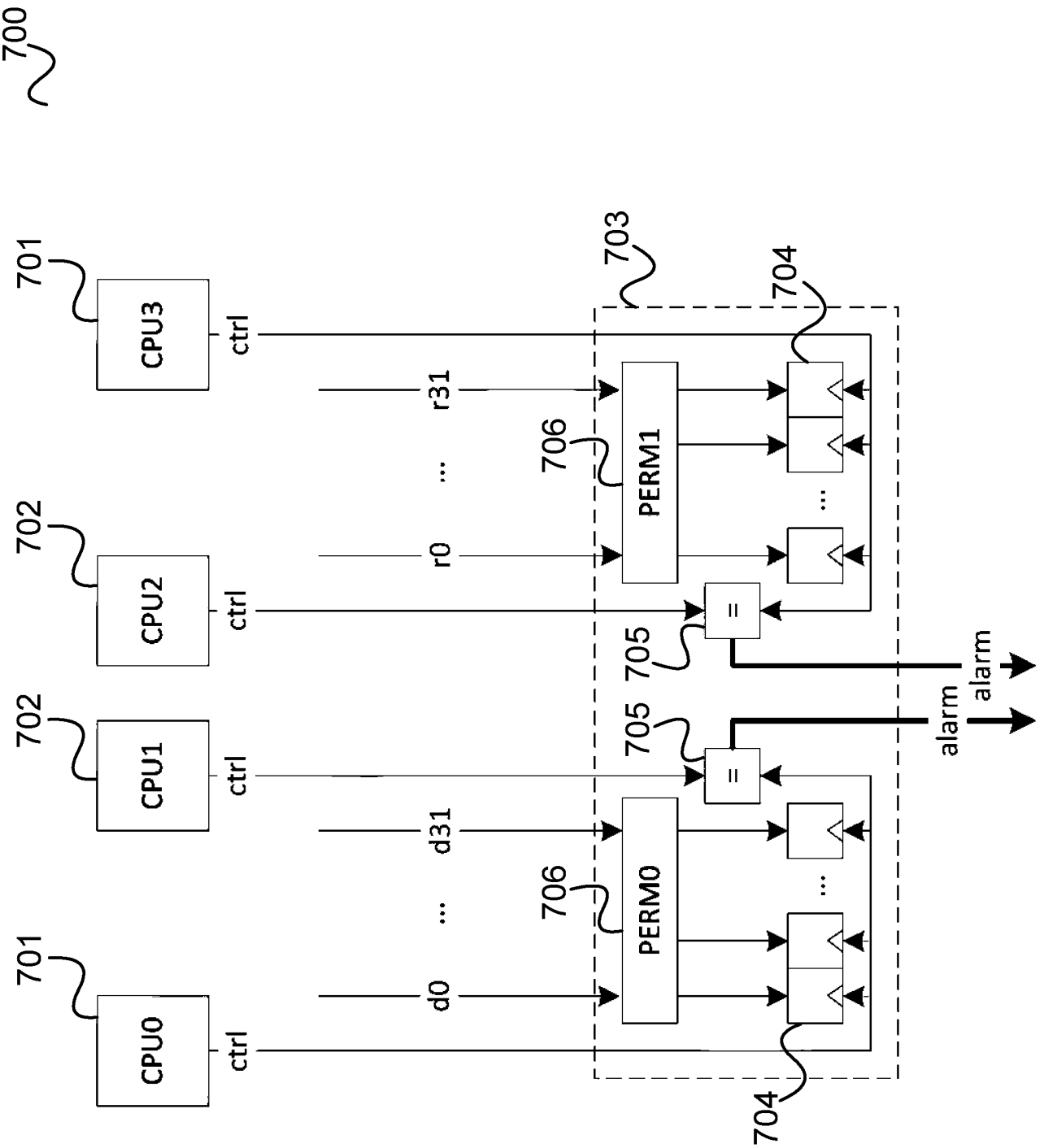


FIG 7

