(19) Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 10 2006 033 694 A1** 2007.03.01

(12) **Offenlegungsschrift**

(21) Aktenzeichen: **10 2006 033 694.1**
(22) Anmeldetag: **20.07.2006**
(43) Offenlegungstag: **01.03.2007**

(51) Int Cl.[8]: ***G06F 9/445*** (2006.01)

(30) Unionspriorität:
2005-210401 20.07.2005 JP

(71) Anmelder:
Denso Corp., Kariya, Aichi, JP

(74) Vertreter:
WINTER, BRANDL, FÜRNISS, HÜBNER, RÖSS, KAISER, POLTE, Partnerschaft, 85354 Freising

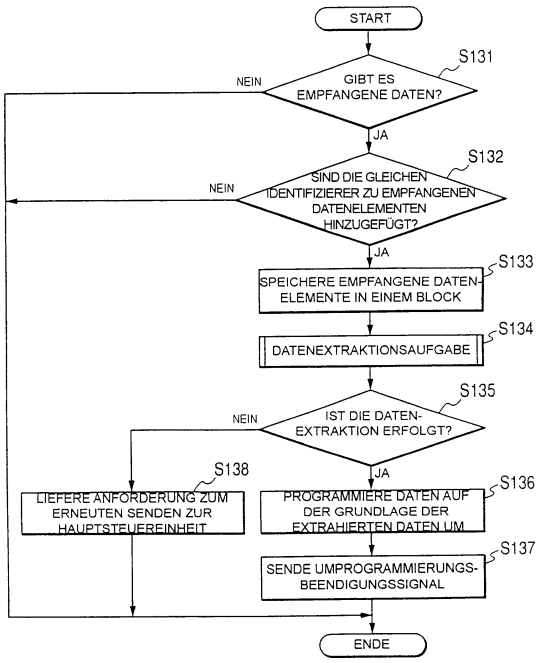
(72) Erfinder:
Nakagaki, Yoshio, Kariya, Aichi, JP

Prüfungsantrag gemäß § 44 PatG ist gestellt.

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: **Datenumprogrammierungsverfahren und -system**

(57) Zusammenfassung: In einem Datenumprogrammierungssystem empfängt eine Empfangseinheit eine Mehrzahl von Elementen von Daten zum Verwenden beim Umprogrammieren. Die Mehrzahl von Elementen von Daten wird zu der Empfangseinheit gesendet. Der Mehrzahl von Elementen von Daten wird jeweils der gleiche Identifizierer hinzugefügt. Eine Datenextraktionseinheit überprüft die empfangene Mehrzahl von Elementen von Daten und extrahiert mindestens ein Element der Daten als gültig gemachte Daten auf ein Bestimmen, dass das mindestens eine Element der Daten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element der Daten erfüllt, auf der Grundlage des überprüften Ergebnisses.



Beschreibung

[0001] Diese Anmeldung beruht auf der japanischen Patentanmeldung 2005-210401, eingereicht am 20. Juli 2005. Diese Anmeldung beansprucht den Zeitrang der Priorität der japanischen Patentanmeldung, so dass deren Beschreibungen hierin durch Verweis eingeschlossen sind.

GEBIET DER ERFINDUNG

[0002] Die vorliegende Erfindung betrifft Datenumprogrammierungsverfahren und -systeme, welche eine Funktion eines umschreibbaren Speicherns von Daten, wie zum Beispiel Steuerprogrammdaten und/oder Steuerdaten, die verwendet werden, um ein Ziel, wie zum Beispiel eine in einem Fahrzeug eingebaute Vorrichtung, zu steuern, aufweist. Genauer gesagt betrifft die vorliegende Erfindung Datenumprogrammierungsverfahren und -systeme, welche dazu ausgelegt sind, die gespeicherten Daten auf der Grundlage von Daten zum Verwenden beim Umprogrammieren umzuprogrammieren; diese Daten zum Verwenden beim Umprogrammieren sind zum Beispiel durch Funkwellen extern gesendet worden. Die Daten zum Verwenden beim Umschreiben werden als "Umprogrammierungsdaten" bezeichnet.

Stand der Technik

HINTERGRUND DER ERFINDUNG

[0003] Eine Mehrzahl von elektronischen Steuereinheiten (ECUs) ist normaler Weise in ein Fahrzeug eingebaut. Zum Beispiel beinhalten die ECUs eine Motorsteuereinheit zum Steuern einer Kraftstoffeinspritzung eines Motors, der in ein Fahrzeug eingebaut ist, und eine Bremssteuereinheit zum Steuern eines darin eingebauten Bremssystems.

[0004] Jede der ECUs ist mit einem nicht flüchtigen Speicher ausgestattet, in welchem ein Steuerprogramm und Steuerdaten, die verwendet werden, um mindestens eine Zielvorrichtung zu steuern, die in das Fahrzeug eingebaut ist, im Voraus gespeichert sind. Genauer gesagt ist jede ECU programmiert, um das Steuerprogramm auf der Grundlage der Steuerdaten und/oder Signale auszuführen, die von in das Fahrzeug eingebauten Sensoren gemessen und von diesen gesendet werden, um dadurch die entsprechende Zielvorrichtung in Übereinstimmung mit dem Steuerprogramm zu steuern; diese gemessenen Signale stellen den Betriebszustand des Fahrzeugs dar.

[0005] Beschreibungen eines derartigen Steuerprogramms und von derartigen Steuerdaten, die diesen zugehörig sind, die in dem nicht flüchtigen Speicher gespeichert sind, können es erfordern, mit dem Versionsupgrade des Steuerprogramms umprogrammiert zu werden.

[0006] In diesem Fall ist eine Leiterplatte, auf welcher eine ECU, deren Steuerprogramm und Steuerdaten, die diesem zugeordnet sind, ein Ziel zum Umprogrammieren sind, montiert ist, von dem Fahrzeug abgefragt, und danach werden die abgefragte ECU und ein zweckmäßiges externes Werkzeug über ein Kabel verbunden, um zwischen diesen kommunizieren zu können. Deshalb lassen Kommunikationen zwischen der ECU und dem externen Werkzeug zu, dass das Steuerprogramm und/oder die Steuerdaten umprogrammiert werden.

[0007] Jedoch kann es eine Menge Schwierigkeiten und Zeit erfordern, um das Steuerprogramm- und Datenumprogrammierungsverfahren zu verwenden, das vorhergehend dargelegt worden ist.

[0008] Um das Problem zu lösen, ist herkömmlicher Weise eine Steuereinheit in der japanischen umgeprüften Patentveröffentlichung Nr. H05-195859 offenbart.

[0009] Genauer gesagt speichert in der Veröffentlichung eine externe Verwaltungszentrale im Voraus VIN-(Fahrzeugidentifikationsnummer)-Codes von Fahrzeugen, die zu verwalten sind, und eine Versionsinformation, die eine bestimmte Version eines Steuerprogramms (Softwareelements) anzeigt, und verwendet wird, um diese Version von anderen Versionen zu unterscheiden. Wenn sie eine Anforderung zum Umprogrammieren mindestens eines Elements in dem Steuerprogramm und den Steuerdaten empfängt, kommuniziert eine Steuereinheit, die in ein Fahrzeug eingebaut ist, drahtlos mit der Verwaltungszentrale, um von dieser Umprogrammierungsdaten zu erzielen. Als Nächstes programmiert die Steuereinheit das mindestens eine Element, das in einem umschreibbaren Bereich des nicht flüchtigen Speichers gespeichert ist, auf der Grundlage der Umprogrammierungsdaten um. Ein Installieren der zuvor aufgebauten Steuereinheit in ein Fahrzeug lässt zu, dass ein Steuerprogramm und Steuerdaten, die darin gespeichert sind, sehr einfach aktualisiert werden.

[0010] Wenn die Steuereinheit dazu ausgelegt, dass ein Steuerprogramm und Steuerdaten, die darin gespeichert sind, unbedingt umprogrammierbar sind, kann ein nicht autorisiertes Umprogrammieren und/oder illegales Ändern durchgeführt werden, was zu einer Fahrzeugleistungsvermögenverschlechterung und/oder Umweltschädigung aufgrund einer Fahrzeugemissionsverschlechterung, wie zum Beispiel Emissionen von NOx, HC und CO, führt, die die gesetzlichen Emissionsstandards überschreiten.

[0011] Um ein derartiges nicht autorisiertes Umprogrammieren und/oder illegales Ändern zu vermeiden, ist die Steuereinheit, die in der Veröffentlichung offenbart ist, dazu ausgelegt, einen Identifizierer (Identifikationscode) als ein umzuprogrammierendes Ziel

mindestens eines der zu verwaltenden Fahrzeuge zu identifizieren; dieser Identifizierer ist zusammen mit den Umprogrammierungsdaten gesendet worden.

[0012] Weiterhin ist die Steuereinheit dazu ausgelegt, zu bestimmen, ob der empfangene Identifikationscode und ein Code, wie zum Beispiel ein VIN-Code, das Fahrzeug bestimmen und darin gespeichert sind. Wenn es bestimmt wird, dass der empfangene Identifikationscode und der Bestimmungscode miteinander übereinstimmen, ist die Steuereinheit dazu ausgelegt, Steuerprogramm- und Datenumprogrammierungsvorgänge auf der Grundlage der Umprogrammierungsdaten durchzuführen.

[0013] In der herkömmlichen Steuereinheit können die Steuerprogramm- und Datenumprogrammierungsvorgänge unter der Bedingung durchgeführt werden, dass der empfangene Identifikationscode und der Bestimmungscode miteinander übereinstimmen. Dies lässt zu, dass der Sicherheitspegel der Steuereinheit, der dem Steuerprogramm- und Datenumprogrammieren zugeordnet ist, stark aufrecht erhalten wird.

[0014] Auf Steuereinheiten dieses Typs, die dazu ausgelegt sind, drahtlos mit der Verwaltungszentrale zu kommunizieren, um die Umprogrammierungsdaten zu erzielen, wird häufiger von externen Vorrichtungen als vorhergehende Steuereinheiten zugegriffen, die dazu ausgelegt sind, über ein Kabel mit einem externen Werkzeug zu kommunizieren, um die Umprogrammierungsdaten von diesen zu erzielen. Die Hochfrequenz eines Zugriffs kann bewirken, dass ein Identifikationscode zusammen mit nicht autorisierten Daten und/oder illegal geänderten Daten mit einem Bestimmungscode des Fahrzeugs übereinstimmt. Die Übereinstimmung zwischen dem Identifikationscode zusammen mit den nicht autorisierten Daten und/oder illegal geänderten Daten und der Bestimmungscode des Fahrzeugs per Zufall oder Absicht kann bewirken, dass ein Steuerprogramm und Steuerdaten auf der Grundlage der nicht autorisierten Daten und/oder illegal geänderten Daten umprogrammiert werden.

[0015] Genauer gesagt kann die unbeabsichtigte oder beabsichtigte Übereinstimmung dieser Codes eine Fahrzeugsleistungsvermögenverschlechterung und/oder Umweltschädigung auf Grund einer Fahrzeugemissionsverschlechterung, wie zum Beispiel Emissionen von NOx, HC und CO, bewirken, die die gesetzlichen Emissionsstandards überschreiten.

Aufgabenstellung

KURZFASSUNG DER ERFINDUNG

[0016] Im Hinblick auf den Hintergrund ist eine Aufgabe von mindestens einem Aspekt der vorliegenden Erfindung, den Sicherheitspegel, der einem Umprogrammieren von Daten, wie zum Beispiel einem Steuerprogramm und/oder Steuerdaten, die in einem umschreibbaren Speicher, wie zum Beispiel einem nicht flüchtigen Speicher, gespeichert sind, aufrechtzuerhalten.

[0017] Gemäß einem ersten Aspekt der vorliegenden Erfindung wird ein Datenumprogrammierungssystem geschaffen, das mit einem umschreibbaren Speicher kommunizieren kann, in welchem im Voraus Daten gespeichert sind. Die gespeicherten Daten sind auf der Grundlage von gültig gemachten Daten umprogrammierbar. Das Datenumprogrammierungssystem beinhaltet eine Empfangseinheit, die dazu ausgelegt, eine Mehrzahl von Elementen von Daten zur Verwendung beim Umprogrammieren zu empfangen. Die Mehrzahl von Elementen von Daten wird zu der Empfangseinheit gesendet. Der Mehrzahl von Elementen von Daten wird jeweils der gleichen Identifizierer hinzugefügt. Das Datenumprogrammierungssystem beinhaltet ebenso eine Datenextraktionseinheit, welche dazu ausgelegt ist, die empfangene Mehrzahl von Elementen von Daten zu überprüfen und mindestens ein Element der Daten als die gültig gemachten Daten nach einem Bestimmen auf der Grundlage des überprüften Ergebnisses zu extrahieren, das das mindestens eine Element der Daten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element von Daten erfüllt.

[0018] Gemäß einem weiteren Aspekt der vorliegenden Erfindung wird ein Datenumprogrammierungs-Programmprodukt geschaffen, das in einen Speicher des Computers eingebettet ist, welcher mit einem umschreibbaren Speicher kommunizieren kann, in welchem Daten im Voraus gespeichert sind. Die gespeicherten Daten werden auf der Grundlage von gültig gemachten Daten umprogrammiert. Das Datenumprogrammierungs-Programmprodukt beinhaltet eine erste Einrichtung zum Anweisen eines Computers, eine Mehrzahl von Elementen von Daten zum Verwenden beim Umprogrammieren zu empfangen. Die Mehrzahl von Elementen von Daten wird zu dem Computer gesendet, wobei der Mehrzahl von Elementen von Daten jeweils der gleiche Identifizierer hinzugefügt ist. Das Datenumprogrammierungs-Programmprodukt beinhaltet eine zweite Einrichtung zum Anweisen eines Computers, die empfangene Mehrzahl von Elementen von Daten zu überprüfen, um mindestens ein Element der Daten als die gültigen Daten nach einem Bestimmen auf der Grundlage des überprüften Ergebnisses zu extrahieren, das das mindestens eine Element der Daten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element der Daten erfüllt.

[0019] Gemäß einem weiteren Aspekt der vorliegenden Erfindung wird ein Verfahren zum Umprogrammieren von Daten, die im Voraus in einem um-

schreibbaren Speicher gespeichert sind, auf der Grundlage von gültig gemachten Daten geschaffen. Das Verfahren beinhaltet ein Empfangen einer Mehrzahl von Elementen von Daten zum Verwenden beim Umprogrammieren. Die Mehrzahl von Elementen von Daten wird derart gesendet, dass der Mehrzahl von Elementen von Daten jeweils der gleiche Identifizierer hinzugefügt ist. Das Verfahren beinhaltet ein Überprüfen der empfangenen Mehrzahl von Elementen von Daten und ein Extrahieren von mindestens einem Element der Daten als die gültig gemachten Daten nach einem Bestimmen, das das mindestens eine Element der Daten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element erfüllt, auf der Grundlage des überprüften Ergebnisses.

Ausführungsbeispiel

KURZE BESCHREIBUNG DER ZEICHNUNG

[0020] Andere Aufgaben und Aspekte der Erfindung werden aus der folgenden Beschreibung von Ausführungsbeispielen unter Bezugnahme auf die beiliegende Zeichnung ersichtlich, in welcher:

[0021] Fig. 1 ein Blockschaltbild ist, das schematisch ein Beispiel des Aufbaus eines in ein Fahrzeug eingebauten Steuersystems gemäß einem ersten Ausführungsbeispiel der vorliegenden Erfindung darstellt;

[0022] Fig. 2 ein Blockschaltbild ist, das schematisch ein Beispiel des Systemsaufbau von jedem einer Motorsteuereinheit und einer Hauptsteuereinheit darstellt, die in Fig. 1 dargestellt sind;

[0023] Fig. 3A eine Ansicht ist, die schematisch ein Beispiel der Struktur von Daten darstellt, die zwischen einer Verwaltungszentrale und dem in einem Fahrzeug eingebauten Steuersystem kommunizierbar sind, das in Fig. 1 dargestellt ist;

[0024] Fig. 3B eine Ansicht ist, die schematisch ein Beispiel der Struktur eines Verwaltungszentralenspeichers, in welchem Umprogrammierungsfaktordaten und Spiegeldaten als Elemente von Umprogrammierungsdaten im Voraus gespeichert sind, und ein Beispiel darstellt, wie die Umprogrammierungsfaktordaten und die Spiegeldaten von der Verwaltungszentrale zu senden sind;

[0025] Fig. 4 ein Flussdiagramm ist, das schematisch eine Empfangsverarbeitung darstellt, die von einer CPU der Hauptsteuereinheit auszuführen ist;

[0026] Fig. 5 ein Flussdiagramm ist, das schematisch eine Sendeverarbeitung darstellt, die von der CPU der Hauptsteuereinheit auszuführen ist;

[0027] Fig. 6 ein Flussdiagramm ist, das schematisch eine Verarbeitung zum erneuten Senden darstellt, die von der CPU der Hauptsteuereinheit auszuführen ist;

[0028] **Fig.** 7 ein Flussdiagramm ist, das schematisch eine Umprogrammierungsverarbeitung darstellt, die von einer CPU der Motorsteuereinheit auszuführen ist, die in Fig. 1 dargestellt ist;

[0029] Fig. 8 eine Ansicht ist, die schematisch eine Extraktions- und Speicherverarbeitung darstellt, die von der CPU der Motorsteuereinheit auszuführen ist, die in Fig. 1 dargestellt ist;

[0030] Fig. 9 ein Flussdiagramm ist, das schematisch ein bestimmtes Beispiel einer Datenextraktionsaufgabe in einem Schritt S134 von **Fig.** 7 darstellt;

[0031] Fig. 10A eine Ansicht ist, die schematisch ein Beispiel der Struktur von Daten, die zwischen der Verwaltungszentrale und dem in ein Fahrzeug eingebauten Steuersystem kommunizierbar sind, gemäß einem zweiten Ausführungsbeispiel der vorliegenden Erfindung darstellt;

[0032] Fig. 10B eine Ansicht ist, die schematisch ein Beispiel der Struktur des Verwaltungszentralenspeichers, in welchem Elemente von Umprogrammierungsdaten im Voraus gespeichert sind, und ein Beispiel darstellt, wie die Umprogrammierungsdaten von der Verwaltungszentrale zu senden sind;

[0033] Fig. 11 eine Ansicht ist, die schematisch eine Extraktions- und Speicherverarbeitung, die von der CPU der Motorsteuereinheit auszuführen sind, gemäß dem zweiten Ausführungsbeispiel darstellt;

[0034] Fig. 12 schematisch ein bestimmtes Beispiel einer Datenextraktionsaufgabe in dem Schritt S134 von **Fig.** 7 gemäß dem zweiten Ausführungsbeispiel darstellt;

[0035] Fig. 13A eine Ansicht ist, die schematisch ein Beispiel der Struktur von Daten, die zwischen der Verwaltungszentrale und dem in eine Fahrzeug eingebauten Steuersystem kommunizierbar sind, gemäß dem zweiten Ausführungsbeispiel darstellt;

[0036] Fig. 13B eine Ansicht ist, die schematisch ein Beispiel der Struktur des Verwaltungszentralenspeichers, in welchem Elemente der Umprogrammierungsdaten im Voraus gespeichert sind, und ein Beispiel, wie die Umprogrammierungsdaten von der Verwaltungszentrale zu senden sind, gemäß dem zweiten Ausführungsbeispiel darstellt;

[0037] Fig. 14 eine Ansicht ist, die schematisch eine Extraktions- und Speicherverarbeitung, die von der CPU der Motorsteuereinheit auszuführen sind, ge-

mäß einem dritten Ausführungsbeispiel der vorliegenden Erfindung darstellt;

[0038] Fig. 15 ein Flussdiagramm ist, das schematisch ein bestimmtes Beispiel einer Datenextraktionsaufgabe im Schritt S134 von **Fig.** 7 gemäß dem dritten Ausführungsbeispiel darstellt;

[0039] Fig. 16A eine Ansicht ist, die schematisch einen Datenblock [5n-4] in den fünf Datenblöcken [5n-4] bis [5n], der als eine Referenz gegen die verbleibenden Datenblöcke darin beim Ausführen einer identischen Überprüfung verwendet wird, gemäß dem dritten Ausführungsbeispiel darstellt;

[0040] Fig. 16B eine Ansicht ist, die schematisch einen Datenblock [5n-3] in den fünf Datenblöcken [5n-4] bis [5n], der als eine Referenz gegen die verbleibenden Datenblöcke darin beim Ausführen der identischen Überprüfung verwendet wird, gemäß dem dritten Ausführungsbeispiel darstellt;

[0041] Fig. 16C eine Ansicht ist, die schematisch einen Datenblock [5n-2] in den fünf Datenblöcken [5n-4] bis [5n], der als eine Referenz gegen die verbleibenden Datenblöcke darin beim Ausführen der identischen Überprüfung verwendet wird, gemäß dem dritten Ausführungsbeispiel darstellt;

[0042] Fig. 16D eine Ansicht ist, die schematisch einen Datenblock [5n-1] in den fünf Datenblöcken [5n-4] bis [5n], der als eine Referenz gegen die verbleibenden Datenblöcke darin beim Ausführen der identischen Überprüfung verwendet wird, gemäß dem dritten Ausführungsbeispiel darstellt;

[0043] Fig. 17 ein Flussdiagramm ist, das schematisch ein Beispiel des Algorithmus eines Identitätsvergleichs in einem Schritt S3100 von Fig. 15 darstellt;

[0044] Fig. 18 ein Flussdiagramm ist, das schematisch ein Beispiel des Algorithmus einer Majoritätsregelüberprüfung in einem Schritt 3200 von Fig. 15 darstellt;:

[0045] Fig. 19 eine Ansicht ist, die schematisch ein anderes Ergebnis des Identitätsvergleichs in dem Schritt S3100 von Fig. 15 darstellt;

[0046] Fig. 20 eine Ansicht ist, die schematisch eine Extraktions- und Speicherverarbeitung, die von der CPU der Motorsteuereinheit auszuführen ist, gemäß einer Ausgestaltung des dritten Ausführungsbeispiels der vorliegenden Erfindung darstellt;

[0047] Fig. 21 ein Flussdiagramm ist, das schematisch ein bestimmtes Beispiel einer Datenextraktionsaufgabe in dem Schritt S134 von **Fig.** 7 gemäß der Ausgestaltung des dritten Ausführungsbeispiels darstellt;

[0048] Fig. 22 eine Ansicht ist, die schematisch eine Extraktions- und Speicherverarbeitung, die von der CPU der Motorsteuereinheit auszuführen ist, gemäß einem vierten Ausführungsbeispiel der vorliegenden Erfindung darstellt;

[0049] Fig. 23 ein Flussdiagramm ist, das schematisch ein bestimmtes Beispiel der Datenextraktionsaufgabe in dem Schritt S134 von **Fig.** 7 gemäß dem vierten Ausführungsbeispiel darstellt;

[0050] Fig. 24 ein Flussdiagramm ist, das schematisch ein Beispiel des Algorithmus einer Majoritätsbelegungsüberprüfung in einem Schritt S5100 von Fig. 23 darstellt;

[0051] Fig. 25 eine Ansicht ist, die schematisch eine Extraktions- und Speicherverarbeitung, die von der CPU der Motorsteuereinheit auszuführen ist, gemäß einer Ausgestaltung des vierten Ausführungsbeispiels der vorliegenden Erfindung darstellt;

[0052] Fig. 26 ein Flussdiagramm ist, das schematisch ein bestimmtes Beispiel einer Datenextraktionsaufgabe in dem Schritt S134 von **Fig.** 7 gemäß der Ausgestaltung des vierten Ausführungsbeispiel darstellt;

[0053] Fig. 27 ein Flussdiagramm ist, das schematisch eine Auswahlverarbeitung, die von der Motorsteuereinheit im Zusammenwirken mit der Verwaltungszentrale auszuführen ist, gemäß einem fünften Ausführungsbeispiel der vorliegenden Erfindung darstellt;

[0054] Fig. 28 eine Ansicht ist, die schematisch eine Tabelle, die die Beziehung zwischen ersten bis vierten Identifizierern und die ersten bis vierten Bestimmungseinheiten darstellt, gemäß dem fünften Ausführungsbeispiel darstellt;

[0055] Fig. 29 ein Flussdiagramm ist, das schematisch eine Datenextraktionsverarbeitung, die von der CPU der Hauptsteuereinheit auszuführen ist, gemäß einer Ausgestaltung von jedem Ausführungsbeispiel der vorliegenden Erfindung darstellt;

[0056] Fig. 30 ein Flussdiagramm ist, das schematisch eine Umprogrammierungsverarbeitung, die von der CPU der Motorsteuereinheit auszuführen ist, gemäß der Ausgestaltung von jedem Ausführungsbeispiel der vorliegenden Erfindung darstellt;

[0057] Fig. 31 eine Ansicht ist, die schematisch eine Extraktions- und Speicherverarbeitung, die von der CPU der Motorsteuereinheit auszuführen ist, gemäß einer Ausgestaltung von jedem Ausführungsbeispiel der vorliegenden Erfindung darstellt; und

[0058] Fig. 32 ein Flussdiagramm ist, das schema-

tisch eine Umprogrammierungsverarbeitung, die von der CPU der Motorsteuereinheit auszuführen ist, gemäß der Ausgestaltung von jedem Ausführungsbeispiel der vorliegenden Erfindung darstellt.

Erstes Ausführungsbeispiel

[0059] In einem ersten Ausführungsbeispiel der vorliegenden Erfindung wird ein Datenumprogrammierungssystem gemäß der vorliegenden Erfindung an einem in ein Fahrzeug eingebauten Steuersystem **100** angewendet.

[0060] Wie es in Fig. 1 dargestellt ist, beinhaltet das in das Fahrzeug eingebaute Steuersystem **100** hauptsächlich eine Hauptsteuereinheit **140**, die derart arbeitet, dass sie ein zentralisiertes Steuern über eine Mehrzahl von elektronischen Steuereinheiten (ECUs) durchführt und im Voraus eine Information, die jeder ECU zugeordnet ist, wie zum Beispiel eine Versionsinformation eines Steuerprogramms, die in jeder ECU gespeichert ist, speichert.

[0061] Die ECUs und die Hauptsteuereinheit **140** sind über einen Kommunikationsbus **101**, der ein LAN (local area network) eines Bustyps, wie zum Beispiel ein CAN (Controller Area Network), bildet, kommunizierbar verbunden. Das Kommunikationsnetz **101** lässt zu, dass die Hauptsteuereinheit **140** und die CPUs Daten miteinander kommunizieren.

[0062] Die Mehrzahl von ECUs beinhalten ECUs **110**, **120** und **130**.

[0063] Die ECU **110** dient als eine Motorsteuereinheit, die derart arbeitet, dass sie eine Kraftstoffeinspritzung eines Motors, der in ein Fahrzeug eingebaut ist, als ein Steuerziel steuert.

[0064] Die ECU **120** dient als eine Getriebesteuereinheit, die derart arbeitet, dass sie eine Änderung des Übersetzungsverhältnisses eines Getriebes (Steuerziels) des Fahrzeugs automatisch steuert.

[0065] Die ECU **130** dient als eine Bremssystem-Steuereinheit, die derart arbeitet, dass sie ein Bremssystem (Steuerziel) des Fahrzeugs steuert, was zu einem Stoppen des Fahrzeugs beiträgt.

[0066] Die ECUs **110**, **120** und **130** kommunizieren über den Kommunikationsbus **101** miteinander eine Information, die den Steuerzustand des entsprechenden Ziels und das Ergebnis eines Steuerns des entsprechenden Ziels anzeigt.

[0067] Jede der ECUs **110**, **120** und **130** weist einen nicht flüchtigen Speicher auf und speichert in ihrem nicht flüchtigen Speicher im Voraus ein Steuerprogramm und Steuerdaten, die erforderlich sind, damit die ECU das entsprechende Programm laufen lässt.

[0068] Genauer gesagt arbeitet jede der ECUs **110**, **120** und **130**, um das entsprechende Steuerprogramm, das in dem nicht flüchtigen Speicher gespeichert ist, auf der Grundlage der entsprechenden Steuerdaten und der Information auszuführen, die von den anderen ECUs kommuniziert wird, um dadurch im Zusammenwirken die Steuervorgänge der Ziele durchzuführen.

[0069] Zum Beispiel ist die Getriebesteuereinheit **120** kommunizierbar mit einem Geschwindigkeitssensor verbunden, der an einer Abtriebswelle des Getriebes oder einer Achse angebracht ist.

[0070] Wenn der Geschwindigkeitssensor die Geschwindigkeit des Fahrzeugs misst und zu der Getriebesteuereinheit **120** sendet, wobei die Information der Geschwindigkeit in der Form von zum Beispiel binären Daten ist, erfasst die Getriebesteuereinheit **120** die Geschwindigkeitsinformation (binären Daten) und erzeugt Daten, die die Fahrzeuggeschwindigkeit anzeigen, auf der Grundlage der Geschwindigkeitsinformation (binären Daten). Danach sendet die Getriebesteuereinheit **120** die erzeugten Daten, die zum Beispiel als serielle Daten formatiert sind, über den Kommunikationsbus **101**.

[0071] Wenn die seriellen Daten, die über den Kommunikationsbus **101** gesendet werden, von der Bremssystem-Steuereinheit **130** erfasst werden, verwendet die Bremssystem-Steuereinheit **130** die seriellen Daten, um das Bremssystem zu steuern.

[0072] Die Hauptsteuereinheit **140** kann Kommunikationen durch Funkwellen mit einer externen Verwaltungszentrale **200** bilden. Die externe Verwaltungszentrale **200** speichert im Voraus VIN-Codes von Fahrzeugen, die zu verwalten sind, und eine Versionsinformation einer bestimmten Version von jedem der Steuerprogramme, die in den ECUs **110**, **120** und **130** gespeichert sind.

[0073] Genauer gesagt arbeitet die Hauptsteuereinheit **140** derart, dass sie drahtlos mit der Verwaltungszentrale **200** kommuniziert, um auf der Grundlage der drahtlosen Kommunikationen Umprogrammierungsdaten zu erzielen, die erforderlich sind, damit die Einheit **140** Datenumprogrammierungsvorgänge durchführt. Genauer gesagt lassen die Umprogrammierungsdaten zu, dass die Einheit **140** die Umprogrammierungsvorgänge durchführt.

[0074] Die Hauptsteuereinheit **140** arbeitet ebenso derart, dass sie Datenelemente, die in den ECUs **110**, **120** und **130** gespeichert sind, auf der Grundlage der erzielten Umprogrammierungsdaten umprogrammiert.

[0075] Genauer gesagt kann es erforderlich sein, dass Beschreibungen der Datenelemente, die in den

ECUs **110**, **120** und **130** gespeichert sind, wie zum Beispiel die Steuerprogramme und die Steuerdaten, mit dem Versionsupgrade der Steuerprogramme und/oder einer Änderung von diesen umprogrammiert werden.

[0076] Bei diesem Erfordernis zum Umprogrammieren kommuniziert die Hauptsteuereinheit **140** drahtlos mit der Verwaltungszentrale **200**, um von der Zentrale **200** zu empfangen:
die Umprogrammierungsdaten, die erforderlich sind, um Datenumprogrammierungsvorgänge durchzuführen; und
einen Identifizierer (ID), der mindestens eines der Fahrzeuge, das zu verwalten ist, und mindestens eine der ECUs **110**, **120** und **130** als ein umzuprogrammierendes Ziel identifiziert.

[0077] Andererseits speichert die Hauptsteuereinheit **140** im Voraus einen Code, wie zum Beispiel einen VIN-Code, der das entsprechende Fahrzeug bestimmt. Daher bestimmt die Hauptsteuereinheit **140**, ob der empfangene Identifizierer und der bestimmte Code des entsprechenden Fahrzeugs zueinander identisch sind.

[0078] Wenn es bestimmt wird, dass der empfangene Identifizierer mit dem bestimmten Code des entsprechenden Fahrzeugs übereinstimmt, programmiert er die Hauptsteuereinheit **140** ein Steuerprogramm und Steuerdaten auf der Grundlage der Umprogrammierungsdaten im Zusammenwirken mit der mindestens einen der ECUs **110**, **120** und **130** um. Dieses Steuerprogramm und diese Steuerdaten werden im Voraus in mindestens einer der ECUs **110**, **120** und **130** gespeichert. Die mindestens eine der ECUs **110**, **120** und **130** entspricht dem umzuprogrammierenden Ziel.

[0079] Ein Einbau des in das Fahrzeug eingebauten Steuersystems **100** in das Fahrzeug lässt zu, dass die Steuerprogramme und Steuerdaten, die in den ECUs **110**, **120** und **130** gespeichert sind, sehr einfach aktualisiert werden.

[0080] Genauer gesagt ist das in das Fahrzeug eingebaute Steuersystem **100** dazu ausgelegt, drahtlos mit der Zentrale **200** zu kommunizieren, um die Umprogrammierungsdaten von dieser zu erzielen.

[0081] Wie es zuvor beschrieben worden ist, wird es in einem herkömmlichen Steuersystem, das den Aufbau aufweist, angenommen, dass als die Umprogrammierungsdaten Daten eines nicht autorisierten Umprogrammierens und/oder Daten eines illegalen Änderns zu dem herkömmlichen Steuersystem gesendet worden sind. Unter dieser Annahme ist es, wenn ein Identifizierer, der zu den Daten eines nicht autorisierten Umprogrammierens und/oder Daten eines illegalen Änderns hinzugefügt worden ist, mit dem bestimmten Code des entsprechenden Fahrzeugs übereinstimmt, wahrscheinlich, dass auf Grund einer Fahrzeugemissionsverschlechterung, wie zum Beispiel Emissionen von NOx, HC und CO, die die gesetzlichen Emissionsstandards überschreiten, eine Fahrzeugleistungsvermögenverschlechterung und/oder Umweltschädigung bewirkt wird.

[0082] Im Gegensatz dazu ist in dem ersten Ausführungsbeispiel das in das Fahrzeug eingebaute Steuersystem **100** dazu ausgelegt:
eine Mehrzahl von Elementen der Umprogrammierungsdaten zu empfangen, zu welchen der gleiche Identifizierer hinzugefügt worden ist, welcher von der Verwaltungszentrale **200** gesendet worden ist;
die empfangene Elemente der Umprogrammierungsdaten miteinander zu überprüfen;
als gültig gemachte Daten mindestens eines der Elemente der Umprogrammierungsdaten zu extrahieren, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass das mindestens eine der Elemente der Umprogrammierungsdaten eine vorbestimmte Regel gegen mindestens ein verbleibendes Element erfüllt; und
auf der Grundlage des extrahierten Elements der Umprogrammierungsdaten mindestens eines des Computerprogramms und der Steuerdaten umzuprogrammieren, die in mindestens einer der ECUs **110**, **120** und **130** gespeichert sind.

[0083] In dem vorhergehend erläuterten Aufbau wird es auch dann, wenn ein Identifizierer von Daten eines nicht aktualisierten Umprogrammierens und/oder einer illegalen Änderung mit einem Code übereinstimmt, der das entsprechende Fahrzeug bestimmt, möglich sein, zu verhindern, dass das Steuersystem **100** die Umprogrammierungsvorgänge durchführt, außer es erfüllt andere Erfordernisse, in welchen:

(1) eine Mehrzahl von Elementen von Daten, wobei zu jedem von diesen der gleiche Identifizierer hinzugefügt worden ist, empfangen wird; und
(2) die empfangenen Elemente von Daten ein Element von Daten beinhalten, das gegenüber den anderen von diesen eine vorbestimmte Regel erfüllt.

[0084] Genauer gesagt ist es in dem Aufbau des in das Fahrzeug eingebauten Steuersystems möglich:
Elemente der Umprogrammierungsdaten zu filtern, die von der Verwaltungszentrale **200** gesendet werden;
als validierte Daten mindestens eines der Elemente der Umprogrammierungsdaten zu extrahieren, wenn es bestimmt wird, dass mindestens eines der Elemente der Umprogrammierungsdaten die vorbestimmte Regel gegen mindestens ein verbleibendes Element erfüllt; und
das extrahierte Element der Umprogrammierungsdaten zu verwenden, um mindestens eines des Compu-

terprogramms und der Steuerdaten umzuprogrammieren, die in mindestens einen der ECUs **110**, **120** und **130** gespeichert sind.

[0085] Daher lässt der Aufbau des in das Fahrzeug eingebauten Steuersystems **100** zu, dass der Sicherheitspegel von diesem stärker aufrecht erhalten wird.

[0086] Fig. 2 stellt schematisch ein Beispiel des Systemsaufbaus von jeder der Motorsteuereinheit **100** und der Hauptsteuereinheit **140** in dem in ein Fahrzeug eingebauten Steuersystem **100** als ein Blockschaltbild dar. Der Systemaufbau und elektrische Betriebe von jeder der Steuereinheiten **110** und **140** werden hier im weiteren Verlauf unter Bezugnahme auf Fig. 2 beschrieben. Es ist anzumerken, dass hier in den Beschreibungen im weiteren Verlauf angenommen wird, dass Daten, die in der Motorsteuereinheit **110** gespeichert sind, umprogrammiert werden.

[0087] Wie es in Fig. 2 dargestellt ist, besteht die Hauptsteuereinheit **140** aus einer CPU **141**, einer Funkkommunikationseinheit **142**, einem Programmspeicher **143**, einem Datenspeicher **144** und einer Kommunikationseinheit **145**. Die CPU **141** ist elektrisch mit der Funkkommunikationseinheit **142**, dem Programmspeicher **143**, dem Datenspeicher **144** und der Kommunikationseinheit **145** verbunden. Die CPU **141** arbeitet, um verschiedene Typen von Elementen von Daten mit jedem der Elemente **142** bis **145** zu kommunizieren, und um Betriebe auf der Grundlage der Elemente von Daten durchzuführen.

[0088] Die Funkkommunikationseinheit **142** dient als eine Datenkommunikationsschnittstelle zwischen der CPU **141** und der Verwaltungszentrale **200**. Genauer gesagt besteht die Funkkommunikationseinheit **142** aus einer Antenne **142a**, einem Sender/Empfängermodul **142b**, einem Modulator und Demodulator (Modem) **142c** und einem Schieberegister **142d**. Die Funktionen der elektrischen Komponenten **142a** bis **142d** werden hier im weiteren Verlauf beschrieben.

[0089] Der Programmspeicher **143** besteht zum Beispiel aus einem Nur-Lese-Speicher (ROM), in welchem verschiedene Steuerprogramme und Steuerdaten, die nach einem Ausführen von diesen verwendet werden, im Voraus gespeichert sind. Genauer gesagt ist in dem ersten Ausführungsbeispiel ein Umprogrammierungssteuerprogramm, das bewirkt, dass die CPU **141** Elemente von Daten umprogrammiert, die in jeder der ECUs **110**, **120** und **130** gespeichert sind, ebenso im Voraus in dem Programmspeicher **143** gespeichert. Weiterhin ist ein VIN-Code, der das entsprechende Fahrzeug bestimmt, ebenso im Voraus in zum Beispiel dem ROM **143** gespeichert.

[0090] Der Datenspeicher **144** besteht zum Beispiel aus einem Direktzugriffsspeicher (RAM). Der Datenspeicher **144** dient als eine Speichereinheit zum Speichern von Elementen von Daten darin, die von der Verwaltungszentrale **200** und jeder der ECUs **110**, **120** und **130** gesendet werden.

[0091] Die Kommunikationseinheit **145** dient als eine Datenkommunikationsschnittstelle zwischen der CPU **141** und jeder der ECUs **110**, **120** und **130** über den Kommunikationsbus **101**.

[0092] In der Hauptsteuereinheit **140** kommuniziert, wenn Daten umprogrammiert werden, die in der Motorsteuereinheit **10** gespeichert sind, die CPU **141** der Hauptsteuereinheit **140** durch Funkwellen mit der Verwaltungszentrale **200**, um eine Mehrzahl von Elementen von Umprogrammierungsdaten zu erzielen. Die Mehrzahl von Elementen von Umprogrammierungsdaten ist erforderlich, damit die Einheit **140** die Datenumprogrammierungsvorgänge durchführt. Weiterhin weist die Mehrzahl von Elementen von Umprogrammierungsdaten den gleichen hinzugefügten Identifizierer auf.

[0093] In dem ersten Ausführungsbeispiel ist in der Verwaltungszentrale **200** jedes der Elemente von Umprogrammierungsdaten vorzugsweise zu seriellen Paketen gewandelt worden, und sind die seriellen Pakete auf eine vorbestimmte Trägerwelle moduliert worden, so dass die modulierte Trägerwelle von der Verwaltungszentrale **200** zu der Hauptsteuereinheit **140** gesendet worden ist.

[0094] In der Funkkommunikationseinheit **142** der Hauptsteuereinheit **140** wird die modulierte Trägerwelle, die von der Zentrale **200** gesendet worden ist, von dem Sende/Empfängermodul **142** über die Antenne **142a** empfangen. Die empfangene Trägerwelle wird zu dem Modem **142c** gesendet, so dass die seriellen Pakete demoduliert werden. Die seriellen Pakete werden zu dem Schieberegister **142d** gesendet, so dass sie in ein paralleles Format gewandelt werden, um dadurch zu der CPU **141** gesendet zu werden. Die CPU **141** empfängt die Pakete in einem parallelen Format und erzielt die Elemente von Umprogrammierungsdaten aus den empfangenen Paketen.

[0095] Nach einem Erzielen der Elemente von Umprogrammierungsdaten lässt die CPU **141** das Umprogrammierungssteuerprogramm laufen, um Betriebe zu starten, um die Datenelemente umzuprogrammieren, die in der Motorsteuereinheit **110** gespeichert sind.

[0096] Genauer gesagt bestimmt die CPU **141**, ob der Identifizierer von jedem der erzielten Elemente von Umprogrammierungsdaten mit dem VIN-Code übereinstimmt, der das entsprechende Fahrzeug bestimmt.

[0097] Wenn es bestimmt worden ist, dass der Identifizierer von jedem der erzielten Elemente von Umprogrammierungsdaten mit dem VIN-Code übereinstimmt, speichert die CPU **141** die erzielten Elemente von Umprogrammierungsdaten in dem Datenspeicher **144** und sendet diese über die Kommunikationseinheit **145** zu der Motorsteuereinheit **110**.

[0098] Die Motorsteuereinheit **110** besteht aus einer CPU **111**, einer Kommunikationseinheit **112**, einem Programmspeicher **113**, einem Datenspeicher **114** und einer Eingabe/Ausgabe-(I/O)-Schnittstelle **115**. Die CPU **111** ist elektrisch mit der Kommunikationseinheit **112**, dem Programmspeicher **113**, dem Datenspeicher **114** und der I/O-Schnittstelle **115** verbunden. Die CPU **111** arbeitet derart, dass sie verschiedene Typen von Elementen von Daten mit jedem der Elemente **112** bis **115** kommuniziert und Betriebe auf der Grundlage der Elemente von Daten durchführt.

[0099] Die Kommunikationseinheit **112** dient als eine Datenkommunikationsschnittstelle zwischen der CPU **111** und der Hauptsteuereinheit **140**.

[0100] Der Programmspeicher **113** besteht aus einem nicht flüchtigen Speicher, wie zum Beispiel einem Flash-Speicher oder einem EEPROM, welcher Daten, die darin gespeichert sind, hält, wenn die Energieversorgung ausgeschaltet wird. In einem ersten vorbestimmten Bereich des Programmspeichers **113** werden Datenelemente, die ein Motorsteuerprogramm und Motorsteuerdaten beinhalten, das diesen zugeordnet ist, im Voraus gespeichert. Die Datenelemente sind ein Ziel zum umprogrammieren.

[0101] Daher wird ein Umprogrammierungssteuerprogramm, das bewirkt, dass die CPU **111** die Datenelemente, die in dem Programmspeicher **113** gespeichert sind, umprogrammiert, ebenfalls im Voraus in zum Beispiel einem zweiten vorbestimmten Bereich des Programmspeichers **113** gespeichert, welcher von dem ersten vorbestimmten Bereich von diesem verschieden ist.

[0102] Der Datenspeicher **114** besteht aus zum Beispiel einem RAM. Der Datenspeicher **114** dient als eine Speichereinheit zum Speichern von Elementen von Daten, die von den ECUs **110**, **120** und **130** gesendet werden, und Daten darin, die von verschiedenen Typen von Sensoren gesendet werden, die über die I/O-Schnittstelle **115** kommunizierbar mit der Motorsteuereinheit **110** verbunden sind. Die verschiedenen Typen von Sensoren (nicht gezeigt) sind dazu ausgelegt, die Elemente einer Information zu messen, die dem Betriebszustand des Fahrzeugs zugehörig ist.

[0103] Die I/O-Schnittstelle **115** dient als eine Datenkommunikationsschnittstelle zwischen den verschiedenen Typen von Sensoren und der CPU **111** und zwischen verschiedenen Typen von Aktoren, die kommunizierbar mit der Motorsteuereinheit **110** und der CPU **111** verbunden sind. Die verschiedenen Typen von Aktoren (nicht gezeigt) sind dazu ausgelegt, auf der Grundlage von Ansteueranweisungen, die von der CPU **111** gesendet werden, verschiedene Typen von in ein Fahrzeug eingebauten Vorrichtungen anzusteuern, die Steuerziele der Motorsteuereinheit **110** sind.

[0104] Die Motorsteuereinheit **110** mit dem zuvor erwähnten Aufbau ist normalerweise aufgebaut, um:
eine Temperaturinformation zu erfassen, die die Temperatur von mindestens einer von Motorkomponenten, wie zum Beispiel eines Motorkühlmittels, und eine Geschwindigkeitsinformation anzeigt, die die Geschwindigkeit des Fahrzeugs anzeigt; und
das Motorsteuerprogramm, das in dem Programmspeicher **113** gespeichert ist, auf der Grundlage der erfassten Elemente einer Information auszuführen, um dadurch den Motor zu steuern.

[0105] Um die Datenelemente, die in dem Programmspeicher **113** gespeichert sind, umzuprogrammieren, muss die CPU **111** der Motorsteuereinheit **110** arbeiten, um:
die Mehrzahl von Elementen der Umprogrammierungsdaten zu erfassen, wobei zu jedem von diesen der gleiche Identifizierer hinzugefügt ist, welcher von der Hauptsteuereinheit **140** gesendet worden sind;
die empfangenen Elemente der Umprogrammierungsdaten in dem Datenspeicher **114** zu speichern;
die empfangenen Elemente der Umprogrammierungsdaten miteinander zu überprüfen;
auf der Grundlage eines überprüften Ergebnisses zu bestimmen, ob es mindestens ein Element der Umprogrammierungsdaten gibt, das eine vorbestimmte Regel gegenüber den anderen Elementen erfüllt;
als gültig gemachte Daten mindestens eines der Elemente der Umprogrammierungsdaten zu extrahieren, wenn es bestimmt wird, dass mindestens ein Element der Umprogrammierungsdaten die vorbestimmte Regel gegenüber mindestens einem verbleibenden Element erfüllt; und
auf der Grundlage des extrahierten mindestens einen Elements der Umprogrammierungsdaten mindestens eines des Computerprogramms und der Steuerdaten, die in dem Programmspeicher **113** gespeichert sind, als ein umzuprogrammierendes Ziel umzuprogrammieren.

[0106] Es ist anzumerken, dass der Systemaufbau der Motorsteuereinheit **110** im Wesentlichen identisch zu dem der anderen ECUs **120** und **130** ist, so dass die Beschreibungen von diesen weggelassen werden.

[0107] Als Nächstes wird hier im weiteren Verlauf ein bestimmtes Beispiel von Datenumprogrammierungsvorgängen unter einem Zusammenwirken der

Hauptsteuereinheit **114** und der Motorsteuereinheit **110** unter Bezugnahme auf die **Fig.** 3 bis Fig. 9 beschrieben.

[0108] Wie es zuvor beschrieben worden ist, wird, wenn die Datenumprogrammierungsvorgänge durchgeführt werden, die Mehrzahl von Elementen von Umprogrammierungsdaten, zu welchen der gleiche Identifizierer hinzugefügt worden ist, von der Verwaltungszentrale **200** zu dem in ein Fahrzeug eingebauten Steuersystem **100** gesendet. Genauer gesagt ist die Verwaltungszentrale **200** gemäß dem ersten Ausführungsbeispiel dazu ausgelegt, als die Mehrzahl von Elementen von Umprogrammierungsdaten und Umprogrammierungsfaktordaten zur tatsächlichen Verwendung beim Umprogrammieren in Spiegeldaten von diesen zu senden, die das Komplementär der Umprogrammierungsfaktordaten sind.

[0109] Die Sendevorgänge der Umprogrammierungsfaktordaten und der Spiegeldaten der Umprogrammierungsdaten von der Verwaltungszentrale **200** werden hier im weiteren Verlauf unter Bezugnahme auf die Fig. 3A und Fig. 3B beschrieben.

[0110] Fig. 3A stellt schematisch ein Beispiel des Aufbaus von Daten dar, die zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** kommunizierbar sind. Fig. 3B stellt schematisch ein Beispiel des Aufbaus des Verwaltungszentralenspeichers dar, in welchen die Umprogrammierungsfaktordaten und die Spiegeldaten als die Elemente der Umprogrammierungsdaten im Voraus gespeichert sind. Fig. 3B stellt ebenso schematisch ein Beispiel dar, wie die Umprogrammierungsfaktordaten und die Spiegeldaten von der Verwaltungszentrale **200** zu senden sind.

[0111] Wie es in Fig. 3A dargestellt ist, werden in dem ersten Ausführungsbeispiel alle der Umprogrammierungsfaktordaten und der Spiegeldaten zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** als ein Datenrahmen DF verarbeitet. Der Datenrahmen DF ist eine geschlossene Einheit von Daten, die zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** zu verarbeiten ist.

[0112] Genauer gesagt besteht der Datenrahmen DF aus einer Mehrzahl von Feldern und einem Überprüfungscode. Der Datenrahmen DF ist dazu ausgelegt, in einem Paket gesendet zu werden. Es ist anzumerken, dass, wenn alle der Umprogrammierungsfaktordaten und der Spiegeldaten als der Datenrahmen DF verarbeitet werden, der gleiche Identifizierer allen der Umprogrammierungsdaten und der Spiegeldaten als die Elemente von Umprogrammierungsdaten hinzugefügt wird.

[0113] Die Mehrzahl von Feldern beinhaltet:
einen Start eines Rahmens (SOF), der den Beginn des Datenrahmens DF markiert;
ein Identifiziererfeld, das den Identifizierer von entsprechenden Daten darstellt;
ein Datenfeld, das die Daten selbst enthält; und
ein Ende eines Rahmens (EOF), das das Ende des Datenrahmens DF markiert.

[0114] Wie es in Fig. 3B dargestellt ist, werden in der Verwaltungszentrale **200** alle der Umprogrammierungsfaktordaten und der Spiegeldaten im Voraus in eine Mehrzahl von Datenblöcken [1] bis [a] geteilt, wobei "a" eine vorbestimmte Ganzzahl darstellt. Die Datenblöcke [1] bis [a] der Umprogrammierungsfaktordaten werden im Voraus in einem ersten Speicherbereich **201a** der Zentrale **200** gespeichert. Ähnlich werden die Datenblöcke [1] bis [a] der Spiegeldaten im Voraus in einem zweiten Speicherbereich **201b** der Zentrale **200** gespeichert.

[0115] Zum Beispiel sind alle der Datenblöcke [1] bis [a] von allen der Umprogrammierungsfaktordaten der Spiegeldaten als beliebige Bits hexadezimal dargestellt. In Fig. 3B ist der Datenblock [1] der Umprogrammierungsfaktordaten durch 0× 0123456..., dargestellt und ist der entsprechende Datenblock [2] der Spiegeldaten durch 0 × FEDCBA9..., dargestellt, wobei dieser Wert 0 × FEDCBA9... das Komplementär des Werts ist.

[0116] Beim Senden der Umprogrammierungsfaktordaten und der Spiegeldaten liest die Verwaltungszentrale **200** aufeinanderfolgend die Umprogrammierungsfaktordaten in der Reihenfolge von dem Blocks [1] zu dem Block (a].

[0117] Parallel zu den Auslesevorgängen liest die Verwaltungszentrale **200** aufeinanderfolgend irgendeinen der Blöcke [1] bis [a] der Umprogrammierungsfaktordaten. Die aufeinanderfolgenden Auslesevorgänge der Verwaltungszentrale **200** werden derart ausgeführt, dass der ausgelesenen der Blöcke [1] bis [a] der Spiegeldaten das Komplementär der ausgelesenen der Blöcke [1] bis [a] der Umprogrammierungsfaktordaten sind.

[0118] Zum Beispiel wird nach einem Auslesen des Blocks [1] der Umprogrammierungsfaktordaten der Block [1] der Spiegeldaten aus den Blöcken [1] bis [a] der Spiegeldaten extrahiert. Ähnlich wird nach einem Auslesen des Blocks [2] der Umprogrammierungsfaktordaten der Block [2] der Spiegeldaten aus den Blöcken [1] bis [a] der Spiegeldaten extrahiert.

[0119] Dies lässt zu, dass Paare von Datenblöcken, die zueinander komplementär sind, aufeinanderfolgend aus den ersten und zweiten Speicherbereichen **201a** und **201b** als Paare P1, P2,..., Pa von Datenblöcken DB1, DB2, DB3, DB4,..., DB2a-1, DB2a ex-

trahiert werden. Die Verwaltungszentrale **200** weist derart kategorisierte Datenblöcke DB1, DB2, DB3, DB4,..., DB2a-1, DB2a auf, dass eine Mehrzahl der Datenrahmen DF jeweils die Datenblöcke DB1, DB2, DB3, DB4,..., DB2a-1, DB2a enthalten.

[0120] Als Nächstes sendet die Verwaltungszentrale **200** aufeinanderfolgend die Mehrzahl von Datenrahmen DF derart zu der in ein Fahrzeug eingebauten Steuereinheit **100**, dass die Datenblöcke DB1, DB3,..., DB2a-1 der Umprogrammierungsfaktordaten zu den komplementären Datenblöcken DB2, DB4, ..., DB2a der Spiegeldaten (in einem Paket oder Block) gepaart sind.

[0121] Wenn die Umprogrammierungsfaktordaten und die Spiegeldaten von der Verwaltungszentrale **200** auf die gleiche wie zuvor beschriebene Weise gesendet werden, führt die Hauptsteuereinheit **140** des in ein Fahrzeug eingebauten Steuersystems **100** eine Empfangsverarbeitung zum Empfangen der Umprogrammierungsfaktordaten und der Spiegeldaten durch und speichert diese in dem Datenspeicher **144**.

[0122] Fig. 4 ist ein Flussdiagramm, das schematisch die Empfangsverarbeitung darstellt, die von der CPU **141** auszuführen ist. Die Empfangsverarbeitung wird zum Beispiel jede vorbestimmte Periode wiederholt. Wie es in Fig. 4 dargestellt ist, wartet beim Ausführen der Empfangsverarbeitung die CPU **141** bis zu einem Empfangen eines Paars von Blöcken der Umprogrammierungsfaktordaten und der Spiegeldaten, wobei zu allen von diesen der gleiche Identifizierer in einem Schritt S101 hinzugefügt worden ist.

[0123] Wenn die gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten, die in einem Block gesendet werden, empfangen werden, ist die Bestimmung in dem Schritt S101 JA, so dass die CPU **141** zu einem Schritt S102 fortschreitet.

[0124] In dem Schritt S102 bestimmt die CPU **141**, ob die Identifizierer, die zu den gepaarten Blöcken der Umprogrammierungsfaktordaten und der Spiegeldaten hinzugefügt sind, mit dem bestimmten Code des entsprechendes Fahrzeugs übereinstimmen.

[0125] Wenn es bestimmt wird, dass die Identifizierer, die zu den gepaarten Blöcken der Umprogrammierungsfaktordaten und der Spiegeldaten hinzugefügt sind, mit dem bestimmten Code des entsprechenden Fahrzeugs übereinstimmen (die Bestimmung in dem Schritt S102 ist JA), schreitet die CPU **141** zu einem Schritt S103 fort. In dem Schritt S103 bestimmt die CPU **141** auf der Grundlage einer Empfangshistorie, die in dem Datenspeicher **144** gespeichert ist, ob die empfangenen gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten äquivalent zu gepaarten Blöcken sind, die bereits empfangen worden sind, um in dem Datenspeicher **144** gespeichert zu werden.

[0126] Wenn es bestimmt wird, dass die empfangenen gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten noch nicht empfangen worden sind (die Bestimmung in Schritt S103 in NEIN), speichert die CPU **141** in einem Block die empfangenen gepaarten Blöcke in dem Datenspeicher **144** als die Empfangshistorie in einem Schritt S104. Genauer gesagt speichert die CPU **141** in einem Block die empfangenen gepaarten Datenblöcke in dem Datenspeicher **144** derart, dass die gepaarten Datenblöcke einander zugeordnet sind.

[0127] Im Gegensatz dazu verlässt, wenn es bestimmt wird, dass die Identifizierer, die zu den gepaarten Blöcken der Umprogrammierungsfaktordaten und der Spiegeldaten hinzugefügt sind, nicht mit dem bestimmten Code des entsprechenden Fahrzeugs übereinstimmen (die Bestimmung in Schritt S102 in NEIN), die CPU **141** die Empfangsverarbeitung ohne ein Speichern der gepaarten Blöcke in dem Datenspeicher **144**.

[0128] Ähnlich verlässt, wenn es bestimmt wird, dass die empfangenen gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten äquivalent zu gepaarten Blöcken sind, die bereits empfangen worden sind, um in dem Datenspeicher **144** gespeichert zu werden (Schritt S103 → JA), die CPU die Empfangsverarbeitung ohne ein Speichern der gepaarten Blöcke in dem Datenspeicher **144**.

[0129] Genauer gesagt wartet die CPU **141** in diesem Datenempfangsverarbeitungs-Endefällen, bis ein Paar von Blöcken der Umprogrammierungsfaktordaten und der Spiegeldaten empfangen wird, zu welchen der gleiche Identifizierer hinzugefügt ist (siehe Schritt S101).

[0130] Wenn die gepaarten Datenblöcke der Umprogrammierungsfaktordaten und der Spiegeldaten in dem Datenspeicher **144** empfangen werden, führt die Hauptsteuereinheit **140** eine Sendeverarbeitung zum Senden der empfangenen gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten zu der Motorsteuereinheit **110** als ein umzuprogrammierendes Ziel durch.

[0131] Fig. 5 ist ein Flussdiagramm, das schematisch die Sendeverarbeitung zeigt, die von der CPU **141** auszuführen ist. Die Sendeverarbeitung wird zum Beispiel jede vorbestimmte Periode wiederholt, wenn die Umprogrammierungsfaktordaten und die Spiegeldaten in dem Datenspeicher **144** gespeichert worden sind.

[0132] Wie es in Fig. 5 dargestellt ist, überprüft beim Ausführen der Sendeverarbeitung die CPU **141**

einen Umprogrammierungsbeendigungsmerker, um zu bestimmen, ob der Umprogrammierungsbeendigungsmerker gesetzt ist (Setzzustand), in einem Schritt S111. Der Umprogrammierungsbeendigungsmerker ist dazu ausgelegt, nach einem Beenden des Umprogrammierens von Datenelementen, die in dem Programmspeicher **113** der Motorsteuereinheit **110** gespeichert sind, gesetzt zu werden. Der Umprogrammierungsbeendigungsmerker ist in dem Datenspeicher **144** gespeichert worden und ist durch die CPU **141** setzbar.

[0133] Wenn es bestimmt wird, dass kein Umprogrammierungsbeendigungsmerker in dem Schritt S111 gesetzt ist, bestimmt die CPU **141** auf der Grundlage des Überprüfungsergebnisses, dass das Umprogrammieren von Datenelementen, die in dem Programmspeicher **113** der Motorsteuereinheit **110** gespeichert sind, noch nicht beendet worden ist, wobei die CPU **141** aufeinanderfolgend Schritte S112 und S113 ausführt.

[0134] Genauer gesagt bestimmt in dem Schritt S112 die CPU **141**, ob das Fahrzeug in einem Zustand ist, der zum Umprogrammieren geeignet ist. Wenn es bestimmt wird, dass das Fahrzeug in einem Zustand ist, der zum Umprogrammieren geeignet ist (die Bestimmung in Schritt S112 in JA), sendet die CPU **141** die gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten, die in dem Datenspeicher **144** gespeichert sind, über den Kommunikationsbus **101** in dem Schritt S113 zu der Motorsteuereinheit **110**.

[0135] Es ist anzumerken, dass die gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten in einem Block (Paket) zu der Motorsteuereinheit gesendet wurden, so dass sei einander zugeordnet werden.

[0136] Andererseits verlässt die CPU **141**, wenn es bestimmt wird, dass der Umprogrammierungsbeendigungsmerker in dem Schritt S111 gesetzt ist (Setzzustand), auf der Grundlage des Überprüfungsergebnisses (die Bestimmung in dem Schritt S111 ist NEIN), die Sendeverarbeitung. In diesem Fall kann die CPU die Umprogrammierungsfaktordaten und die Spiegeldaten löschen, die in dem Datenspeicher **144** gespeichert sind.

[0137] Andererseits verlässt die CPU **141**, wenn es bestimmt wird, dass das Fahrzeug nicht mehr in einem Zustand ist, der zum Umprogrammieren geeignet ist (die Bestimmung in Schritt S112 ist NEIN), die Sendeverarbeitung. In diesem Fall wartet die CPU **141**, bis das Fahrzeug in einem Zustand ist, der zum Umprogrammieren geeignet ist.

[0138] Zum Beispiel besteht beim Ausführen der Sendeverarbeitung die Aufgabe in Schritt S112 darin, zu bestimmen, ob eine vorbestimmte Periode seit der Zeit verstrichen ist, zu welcher die Motorumdrehungen weniger als 50 U/min sind, welches im Wesentlichen äquivalent zu null ist. Genauer gesagt ist die CPU **111** der Motorsteuereinheit **110** normalerweise programmiert, um eine Nachverarbeitung, die eine Aufgabe beinhaltet, Daten, wie zum Beispiel Lernwerte, die bis zu dem nächsten Betreiben des Motors gehalten werden sollten, in einem vorbestimmten Sicherungsbereich des Datenspeichers **114** zu speichern, durchzuführen.

[0139] Daher wartet in der Aufgabe in Schritt S112 die CPU **111**, bis die vorbestimmte Periode seit der Zeit verstrichen ist, zu welcher die Motorumdrehungen weniger als 50 U/min sind, welches im Wesentlichen äquivalent zu null sind. Diese Warteaufgabe kann verhindern, dass Datenelemente, die in dem Programmspeicher **113** gespeichert sind, parallel zu der Nachverarbeitung umprogrammiert werden.

[0140] Wenn die gepaarten Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten durch die Sendeverarbeitung zu der Motorsteuereinheit **110** gesendet werden, überprüft die Motorsteuereinheit **110**, wie es vorhergehend erläutert worden ist, die empfangenen Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten miteinander.

[0141] Als Nächstes extrahiert die Motorsteuereinheit **110** als gültig gemachte Daten mindestens einen der Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten, wenn es auf der Grundlage des Überprüfungsergebnisses bestimmt wird, dass mindestens einer der Blöcke von diesen die vorbestimmte Regel gegenüber mindestens einem verbleibenden Block erfüllt.

[0142] Auf der Grundlage des extrahierten Blocks der Umprogrammierungsfaktordaten und der Spiegeldaten programmiert die Motorsteuereinheit **110** mindestens eines des Computerprogramms und der Steuerdaten, die darin gespeichert sind.

[0143] Es ist anzumerken, dass, wenn es bestimmt wird, dass keine Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten die vorbestimmte Regel erfüllen, die Motorsteuereinheit **110** bestimmt, dass die Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten, die per Funkkommunikationen von der Verwaltungszentrale **200** gesendet werden, keine Gültigkeit aufweisen.

[0144] Dies kann die Hauptsteuereinheit **140** mit einer Anforderung zum erneuten Senden der Umprogrammierungsfaktordaten und der Spiegeldaten versehen; diese Aufgabe, wenn keine Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten die vorbestimmte Regel erfüllen, werden hier im weiteren Verlauf im Detail beschrieben. Wen die Anfor-

derung zum erneuten Senden empfangen wird, führt die Hauptsteuereinheit **140** eine Sendeverarbeitung zum Anfordern des erneuten Sendens der Umprogrammierungsfaktordaten und der Spiegeldaten an der Verwaltungszentrale **200** durch; diese Sendeverarbeitung ist in Fig. 6 dargestellt.

[0145] Genauer gesagt fordert die CPU **141** der Hauptsteuereinheit **140** in einem Schritt S121, wenn die Anforderung zum erneuten Senden von der Motorsteuereinheit **110** empfangen wird, an der Verwaltungszentrale **200** ein erneutes Senden der Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten in einem Schritt S122 an. Es ist anzumerken, dass in einem Schritt S122, wenn die Blöcke der Umprogrammierungsfaktordaten und der Spiegeldaten in dem Datenspeicher **144** gespeichert worden sind, welche durch die vorhergehenden Funkkommunikationen erzielt worden sind, die CPU **111** vorzugsweise diese aus dem Datenspeicher **144** löschen kann.

[0146] **Fig.** 7 ist ein Flussdiagramm, das schematisch eine Umprogrammierungsverarbeitung darstellt, die von der CPU **111** der Motorsteuerreinheit **110** auszuführen ist; diese Umprogrammierungsverarbeitung besteht darin, Datenelemente umzuprogrammierend die umzuprogrammieren sind, welche im Voraus in dem Programmspeicher **111** gespeichert sind.

[0147] Wie es in **Fig.** 7 dargestellt ist, wartet beim Ausführen der Umprogrammierungsverarbeitung der CPU **111** bis zu einem Empfangen von Elementen von Daten durch den Kommunikationsbus **101** in einem Schritt S131.

[0148] Wenn sie Elemente von Daten über den Kommunikationsbus **101** empfängt, identifiziert die CPU **111** den Typ von jedem der empfangenen Elemente von Daten auf der Grundlage der Identifizierer, die zu den empfangenen Elementen von Daten hinzugefügt sind, in einem Schritt S132. Anders ausgedrückt bestimmt die CPU **111**, ob die Identifizierer, die zu den empfangenen Elementen von Daten hinzugefügt sind, miteinander übereinstimmen.

[0149] Wenn es identifiziert wird, dass die Identifizierer, die zu den empfangenen Elementen von Daten hinzugefügt sind, miteinander übereinstimmen, kann die CPU **111** identifizieren, dass die empfangenen Elemente von Daten den Elementen der Umprogrammierungsfaktordaten und der Spiegeldaten entsprechen (die Bestimmung in dem Schritt S132 ist JA). Dann schreitet die CPU **111** zu einem Schritt S133 fort und führt den Satz von Schritten S133 bis S138 zum Umprogrammieren der Datenelemente durch, die in dem Programmspeicher **113** gespeichert sind.

[0150] In einem Schritt S113 speichert die CPU **111** in dem Datenspeicher **114** die Elemente der Umprogrammierungsfaktordaten und der Spiegeldaten in einem Datenblock.

[0151] Genauer gesagt speichert die CPU **111**, wie es in Fig. 8 dargestellt ist, in einem ersten Speicherbereich **114a** des Datenspeichers **114** die Datenblöcke DB1, DB3,..., DB2a-1 der Umprogrammierungsfaktordaten und die komplementären Datenblöcke DB2, DB4,..., DB2a der Spiegeldaten derart, dass die Datenblöcke DB1, DB3,..., DB2a-1 der Umprogrammierungsfaktordaten mit den komplementären Datenblöcken DB2, DB4,..., DB2a der Spiegeldaten gepaart sind.

[0152] Nachfolgend überprüft die CPU **111** die Umprogrammierungsfaktordaten und die Spiegeldaten, die in dem ersten Speicherbereich **114a** des Datenspeichers **114** gespeichert sind. Als Nächstes extrahiert die CPU **111** als die gültig gemachten Daten mindestens ein Element der Umprogrammierungsfaktordaten, wenn es auf der Grundlage des Überprüfungsergebnisses bestimmt wird, dass das mindestens eine Element der Umprogrammierungsfaktordaten die vorbestimmte Regel gegenüber einem entsprechenden mindestens einem Element der Spiegeldaten erfüllt.

[0153] Genauer gesagt führt die CPU **111** in dem Schritt S134 eine Spiegelüberprüfung von Elementen der Umprogrammierungsfaktordaten und denjenigen der Spiegeldaten durch. In dem ersten Ausführungsbeispiel bestimmt die CPU **111**, wie es in Fig. 8 dargestellt ist, ob die Datenelemente (Datenblöcke) von jeden Paar komplementär zueinander sind.

[0154] Als ein Ergebnis der Spiegelüberprüfung extrahiert die CPU **111** mindestens einen Datenblock der Umprogrammierungsfaktordaten, wenn es bestimmt wird, dass der mindestens eine Datenblock der Umprogrammierungsfaktordaten komplementär zu dem entsprechenden mindestens einen Datenblock der Spiegeldaten ist. Danach speichert die CPU **111** den extrahierten mindestens einen Datenblock der Umprogrammierungsfaktordaten in dem zweiten Speicherbereich **114b** des Datenspeichers **114**.

[0155] Zum Beispiel kombiniert die CPU **111**, wenn es bestimmt wird, dass alle Datenblöcke der Umprogrammierungsfaktordaten komplementär zur entsprechenden allen Datenblöcken die Spiegeldaten sind, aufeinanderfolgend die extrahierten Datenblöcke [1] bis [a] (Datenblöcke DB1 bis DB2a-1), um dadurch die Umprogrammierungsfaktordaten zu dekodieren.

[0156] Als ein Ergebnis der Datenextraktionsaufgabe in Schritt S134 führt die CPU **111**, wenn es be-

stimmt wird, dass die Umprogrammierungsfaktordaten, die aus den Datenblöcken [1] bis [a] bestehen, dekodiert werden (die Bestimmung in Schritt S135 ist JA), den Satz von Schritten von S136 und S137 durch. Genauer gesagt programmiert die CPU **111** Datenelemente in den Datenspeicher, die umzuprogrammieren sind, auf der Grundlage der dekodierten Umprogrammierungsfaktordaten in Schritt S136 um.

[0157] Nach einem Überprüfen einer Beendigung des Umprogrammierens der Datenelemente sendet die CPU **111** ein Umprogrammierungsbeendigungssignal, das ein Beenden eines Umprogrammierens von Datenelementen anzeigt, die in der Motorsteuereinheit **110** gespeichert sind, zu der Hauptsteuereinheit **140**. Dies lässt zu, dass die Hauptsteuereinheit **140** den Umprogrammierungsbeendigungsmerker als Reaktion auf ein Empfangen des Umprogrammierungsbeendigungssignals setzt.

[0158] Andererseits werden, wenn es bestimmt wird, dass mindestens einer der Datenblöcke [1] bis [a] der Umprogrammierungsfaktordaten nicht dekodiert wird, anders ausgedrückt, die Spiegelüberprüfung an mindestens einem der Datenblöcke [1] bis [a] der Umprogrammierungsfaktordaten ausfällt (Schritt **135** → NEIN), die Umprogrammierungsfaktordaten zur tatsächlichen Verwendung beim Umprogrammieren nicht dekodiert.

[0159] In diesem Fall bestimmt die CPU **111**, wie es zuvor beschrieben worden ist, dass die Länge der Umprogrammierungsfaktordaten und der Spiegeldaten, die über Funkkommunikationen von der Verwaltungszentrale **200** gesendet worden sind, keine Gültigkeit aufweisen, was daher die Hauptsteuereinheit **140** mit einer Anforderung zum erneuten Senden der Umprogrammierungsfaktordaten und der Spiegeldaten in einem Schritt S138 vorsieht. Dies lässt zu, dass die Hauptsteuereinheit **140** die Sendeverarbeitung zum Anfordern eines erneuten Sendens der Umprogrammierungsfaktordaten und der Spiegeldaten zu der Verwendungszentrale **200** durchführt (siehe Fig. 6).

[0160] Fig. 9 stellt schematisch ein bestimmtes Beispiel der Datenextraktionsaufgabe in dem Schritt S134 dar.

[0161] Es ist anzumerken, dass die Umprogrammierungsfaktordaten und die Spiegeldaten, die bereits in Fig. 8 dargestellt sind, die in die Datenblöcke [1], [2],... [2a-1], [2a] und die Datenblöcke [1], [2],... [2a-], [2a] geteilt worden sind, und die Datenblöcke [1], [2],... [2a-1], [2a] als der Datenblock DB1, DB2,... DB2a-1, DB2a in dem ersten Speicherbereich **114a** des Datenspeichers **114** gespeichert worden sind.

[0162] In der Datenextraktionsaufgabe weist die CPU **111**, wie es in Fig. 9 dargestellt ist, wenn die ungeraden Blöcke [1], [3],... [2a-], der Blöcke [1], [2],... [2a-1], [2a] durch [2n-1 (n ist ein Ganzzahl)] ausgedrückt sind und die verbleibenden geraden Blöcke [2], [4],..., [2a] durch [2n] ausgedrückt sind, der Variablen n in einem Schritt 1001 "1" zu.

[0163] Als Nächstes wiederholt CPU **111** den folgenden Satz von Schritten S1002, S1003 und S1011, welche zulassen, dass jedes Paar der Datenblöcke [1] bis [2a] der Umprogrammierungsfaktordaten und der Spiegeldaten der Spiegelüberprüfung unterzogen wird, wie es vorhergehend dargelegt worden ist.

[0164] In Schritt S1002 bestimmt die CPU **111**, ob der Datenblock [2n] das Komplementäre des Datenblocks [2n-1] ist.

[0165] In Schritt S1003 bestimmt die CPU **111**, ob alle der Datenblöcke [1] bis [2a] der Spiegelüberprüfung unterzogen werden.

[0166] In Schritt S1011 addiert die CPU **111[**1] zu der Variablen (n).

[0167] Genauer gesagt, werden, wenn "1" der Variablen "n" in Schritt S1001 zugewiesen wird, in dem S1002 zuerst der Datenblock DB1 und DB2 der Spiegelüberprüfung unterzogen. Zum Beispiel wird in dem ersten Ausführungsbeispiel die Spiegelüberprüfung aus einer Bestimmung bezüglich dessen ausgeführt, ob die Summe des Werts 0 × 0123456... von DB1 und der Wert 0 × FEDCBA9... von DB2 mit einem Erwartungswert 0 × FFFFFFF... übereinstimmen, der normalerweise durch die Summe des Werts von DB1 und der von DB2 erzielt werden sollte.

[0168] Wenn es bestimmt wird, dass die Summe des Werts 0 × 0123456... von DB1 und des Werts 0 × FEDCBA9... von DB2 mit dem Erwartungswert übereinstimmen, wird die Spiegelüberprüfung in den Datenblöcken DB1 und DB2 gebildet, anders ausgedrückt wird die vorbestimmte Regel in den Datenblöcken DB1 und DB2 gebildet. Dann wird die Aufgabe, die zum Ausführen der Spiegelüberprüfung für den Datenblock DB3 und DB4 erforderlich ist, ausgeführt.

[0169] Genauer gesagt wird in dem Schritt S1011 "1" zu der Variablen (n) hinzugefügt und werden danach die Vorgänge in den Schritten S1002 und 1003 erneut ausgeführt. Dies lässt zu, dass in dem Schritt S1002 die Spiegelüberprüfung auf die gleiche Weise wie die Datenblöcke DB1 und DB2 an den Datenblöcken DB3 und DB4 angewendet wird.

[0170] Als ein Ergebnis eines wiederholten Ausführens des Satzes von Schritten S1002, S1003 und S1011 führt die CPU **111**, wenn es in Schritt S1003 bestimmt wird, dass alle geraden Blöcke [1] bis [2a] der Spiegelüberprüfung unterzogen werden, den Satz der Schritte S1004 und S1005 aus.

[0171] Genauer gesagt extrahiert die CPU **111** in Schritt S1004 als die gültigen Daten die Datenblöcke [1] bis [a] (Datenblöcke DB1, DB3,..., DB2a-1) aus den Datenblöcken DB1 bis DB2a der Umprogrammierungsfaktordaten und der Spiegeldaten, die in dem ersten Speicherbereich **114a** des Datenspeichers **114** gespeichert sind. Dann kombiniert die CPU **111** aufeinanderfolgend die extrahierten Datenblöcke [1] bis [a], um die Umprogrammierungsfaktordaten zu dekodieren (zu demodulieren) und speichert den kombinierten Satz der Datenblöcke [1] bis [a] in dem zweiten Speicherbereich **114b** des Datenspeichers **114**.

[0172] Wenn die Umprogrammierungsfaktordaten als ein Ergebnis der Datenextraktionsaufgabe dekodiert werden, die in **Fig.** 7 dargestellt ist, setzt die CPU **111** einen Spiegelmerke in zum Beispiel dem Datenspeicher **114**; dieser Spiegelmerker stellt dar, dass die Spiegelüberprüfung gebildet ist. Dies lässt zu, dass die CPU **111** auf der Grundlage des Zustands des Spiegelmerkers bestimmt, ob die Umprogrammierungsfaktordaten dekodiert werden.

[0173] Als ein Ergebnis eines wiederholten Ausführens des Satzes von Schritten S1002, S1003 und S1011 kann es bestimmt werden, dass die Spiegelüberprüfung nicht gebildet wird, anders ausgedrückt die vorbestimmte Regel nicht erfüllt ist. In diesem Fall bestimmt die CPU **111** wahrscheinlich, dass die Elemente von Daten, die von der Verwaltungszentrale **200** gesendet werden, nicht autorisierte Daten und/oder illegal geänderte Daten sind, um dadurch die Datenextraktionsaufgabe nach einem derartigen Bestimmen zu verlassen, dass die Spiegelüberprüfung ohne Setzen des Spiegelmerkers nicht gebildet wird.

[0174] Wie es zuvor beschrieben worden ist, kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß dem ersten Ausführungsbeispiel die ersten und zweiten Effekte erzielen, die hier im weiteren Verlauf beschrieben werden.

[0175] Genauer gesagt ist es als der erste Effekt beim Umprogrammieren von Datenelementen von Steuerprogrammen und/oder Steuerdaten, die in mindestens einer der ECUs **110**, **120** und **130** gespeichert sind, wenn die Elemente der Umprogrammierungsdaten zur Verwendung beim Umprogrammieren empfangen werden, wobei jedem von diesen der gleiche Identifizierer hinzugefügt ist, möglich:
die Empfängerelemente der Umprogrammierungsdaten gegeneinander zu überprüfen; und
als die gültig gemachten Daten mindestens eines der Elemente der Umprogrammierungsdaten zu extrahieren, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass mindestens eines der Elemente der Umprogrammierungsdaten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element erfüllt.

[0176] Dies lässt zu, dass der Sicherheitspegel, der dem Steuerprogramm- und Datenumprogrammieren zugehörig ist, stark aufrechterhalten wird.

[0177] Als der zweite Effekt ist es möglich, als die Elemente der Umprogrammierungsdaten die Umprogrammierungsfaktordaten mit einer Gültigkeit und die Spiegeldaten zu verwenden, die das Komplementär von diesen ohne Gültigkeit sind. Dies lässt zu, dass der Sicherheitspegel von mindestens einem Element selbst, das aus den Elementen der Umprogrammierungsdaten extrahiert ist, stärker aufrechterhalten wird.

Zweites Ausführungsbeispiel

[0178] Ein in das Fahrzeug eingebautes Steuersystem gemäß einem zweiten Ausführungsbeispiel der vorliegenden Erfindung wird hier im weiteren Verlauf beschrieben. Das in ein Fahrzeug eingebaute Steuersystem des zweiten Ausführungsbeispiels weist im Wesentlichen den gleichen Aufbau wie den des in das Fahrzeug eingebauten Steuersystems des ersten Ausführungsbeispiels auf. Aus diesem Grund sind ähnliche Bezugszeichen ähnlichen Teile den in das Innenfahrzeug eingebauten Steuersystemen gemäß den ersten und zweiten Ausführungsbeispielen zugewiesen, so dass Beschreibungen der Teile des in das Fahrzeug eingebauten Steuersystems des zweiten Ausführungsbeispiels weggelassen werden.

[0179] Genauer gesagt besteht das in das Fahrzeug eingebaute Steuersystem **100** des zweiten Ausführungsbeispiels hauptsächlich aus:
einer Mehrzahl von ECUs **110**, **120** und **130**, die verteilt angeordnet sind, und arbeiten, um im Zusammenwirken die entsprechenden Steuervorgänge der entsprechenden in ein Fahrzeug eingebauten Vorrichtungen (Ziele) durchzuführen; und
eine Hauptsteuereinheit **140**, die arbeitet, um ein zentralisiertes Steuern in einer Mehrzahl von ECUs **110**, **120** und **130** durchzuführen.

[0180] Ein Kommunikationsnetz **101** eines Bustyps lässt zu, dass die Hauptsteuereinheit **140** und ECUs **110**, **120** und **130** Daten miteinander kommunizieren. Der Systemaufbau von jeder der Hauptsteuereinheit **140** und jeder ECUs **110**, **120** und **130** des in das Fahrzeug eingebauten Steuersystems des zweiten Ausführungsbeispiels ist im Wesentlichen äquivalent zu dem, der in Fig. 2 dargestellt ist.

[0181] Im Wesentlichen gleich dem ersten Ausführungsbeispiel ist in dem zweiten Ausführungsbeispiel das in das Fahrzeug eingebaute Steuersystem **100** aufgebaut um:
eine Mehrzahl von Elementen der Umprogrammierungsdaten zu empfangen, zu welchem der gleiche

Identifizierer hinzugefügt ist, welcher von der Verwaltungszentrale **200** gesendet worden ist;
die empfangenen Elemente der Umprogrammierungsdaten gegeneinander zu überprüfen; und
mindestens eines der Elemente der Umprogrammierungsdaten als gültig gemachte Daten zu extrahieren, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass das mindestens eine der Elemente der Umprogrammierungsdaten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element erfüllt.

[0182] Es ist anzumerken, dass in dem zweiten Ausführungsbeispiel, wie es in den **Fig.** 10 und Fig. 11 verglichen mit den **Fig.** 3 und Fig. 8 dargestellt ist, ein Beispiel, wie Elemente der Umprogrammierungsdaten durch die Verwaltungszentrale zu senden sind, und dass, wie die Umprogrammierungsfaktordaten zu extrahieren sind, dadurch unterschiedlich zu denjenigen in dem ersten Ausführungsbeispiel sind.

[0183] Die Sendeorgänge der Umprogrammierungsdaten von der Verwaltungszentrale **200** werden im weiteren Verlauf unter Bezugnahme auf die Fig. 10A und Fig. 10B beschrieben.

[0184] Fig. 10A stellt schematisch ein Beispiel des Aufbaus von Daten dar, die zwischen der Verwaltungszentrale **200** und dem in das Fahrzeug eingebauten Steuersystem **100** kommunizierbar sind. Fig. 10B stellt schematisch ein Beispiel des Aufbaus eines Verwaltungszentralenspeichers dar, in welchem die Elemente der Umprogrammierungsdaten im Voraus gespeichert sind. Fig. 10B stellt ebenso schematisch ein Beispiel dar, wie die Umprogrammierungsdaten von der Verwaltungszentrale **200** zu senden sind.

[0185] Wie in Fig. 10A dargestellt ist, wird in dem zweiten Ausführungsbeispiel jedes der Elemente von Umprogrammierungsdaten zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** als ein Datenrahmen DF verarbeitet. Der Datenrahmen DF ist eine geschlossene Einheit von Daten, die zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** zu verarbeiten sind.

[0186] Genauer gesagt besteht der Datenrahmen DF aus einer Mehrzahl von Feldern und einem Überprüfungscode. Der Datenrahmen DF ist dazu ausgelegt, ein Paket zu senden. Es ist anzumerken, dass, wenn jedes der Elemente der Umprogrammierungsdaten aus dem Datenrahmen DF verarbeitet wird, der gleiche Identifizierer jedem der Elemente der Umprogrammierungsdaten hinzugefügt wird.

[0187] Der Datenaufbau des Datenrahmens DF ist im Wesentlichen äquivalent zu dem des Datenrahmens DF gemäß dem ersten Ausführungsbeispiel.

[0188] Wie es in Fig. 10B dargestellt ist, werden in der Verwaltungszentrale **200** an Stelle der Spiegeldaten, die komplementär zu den Umprogrammierungsfaktordaten sind, Pseudodaten als nicht gültig gemachte Daten verwendet, die unvereinbar zu den Umprogrammierungsfaktordaten (Gültigmachungsdaten) sind.

[0189] Alle der Umprogrammierungsfaktordaten und der Pseudodaten werden in eine Mehrzahl von Datenblöcken [1] bis [b] geteilt, wobei "b" eine vorbestimmte Ganzzahl darstellt. Der Datenblöcke [1] bis [b] der Umprogrammierungsfaktordaten werden im Voraus in einem ersten Speicherbereich **201c** der Zentrale **200** gespeichert. Ähnlich werden die Datenblöcke [1] bis [b] der Pseudodaten im Voraus in einem zweiten Speicherbereich **201d** der Zentrale **200** gespeichert.

[0190] Beim Senden der Umprogrammierungsfaktordaten und der Pseudodaten liest die Verwaltungszentrale **200** aufeinanderfolgend die Umprogrammierungsfaktordaten und die Pseudodaten in der Reihenfolge von dem Block [1] zu dem Block [b] aus.

[0191] Weiterhin kombiniert die Verwaltungszentrale **200** ein Paar der gleichen ausgelesenen Blöcke der Umprogrammierungsfaktordaten mit dem ausgelesenen Block der Pseudodaten, um dadurch den Satz von drei Elemente der Blockdaten als die Elemente der Umprogrammierungsdaten zu senden.

[0192] Zum Beispiel wird, wie es in Fig. 10B dargestellt ist, nach einem doppelten Auslesen von einem Block aufeinanderfolgend von dem Block [1] zu dem Block [b] der Umprogrammierungsfaktordaten ein Block von dem Block [1] zu dem Block [b] der Pseudodaten aufeinanderfolgend extrahiert. Das doppelte Auslesen von jedem Block der Umprogrammierungsfaktordaten wird mit dem entsprechenden ausgelesenen einem Block der Pseudodaten kombiniert.

[0193] Die lässt zu, dass der Satz eine Gruppe G1 zu einer Gruppe Gb erzeugt wird. Die Gruppe G1 besteht aus einem Paar DB1 und DB2 des Datenblocks [1] der Umprogrammierungsfaktordaten und des Datenblocks DB3 (Datenblock [1]) der Pseudodaten. Die Gruppe Gb besteht aus einem Paar DB3b-2 und DB3b-1 des Datenblocks [b] der Umprogrammierungsfaktordaten und dem Datenblock [b] (Datenblock [1]) der Pseudodaten.

[0194] Die Verwaltungszentrale **200** kategorisiert Datenblöcke DB1, DB2, DB3,..., DB3b-2, DB3b-1, DB3b derart, dass eine Mehrzahl der Datenrahmen DF die Datenblöcke DB1, DB2, DB3,..., DB3b-2, DB3b-1, DB3b enthält. Als Nächstes sendet die Verwaltungszentrale **200** aufeinanderfolgend die Mehr-

zahl der Datenrahmen DF zu der in ein Fahrzeug eingebauten Steuereinheit **100** in einem Paket bzw. in einem Block.

[0195] Wenn die Elemente der Umprogrammierungsfaktordaten und der Pseudodaten von der Verwaltungszentrale **200** auf die zuvor beschriebene Sendeweise gesendet werden, führt die Hauptsteuereinheit **140** durch:
eine Empfangsverarbeitung zum Empfangen der Umprogrammierungsfaktordaten und der Pseudodaten; und
ein Speichern von ihnen in dem Datenspeicher **144**, welches ähnlich zu dem ersten Ausführungsbeispiel ist (siehe Fig. 4).

[0196] Als Nächstes führt, wenn die Elemente der Umprogrammierungsfaktordaten und der Pseudodaten in dem Datenspeicher **144** empfangen werden, die Hauptsteuereinheit **140** eine Sendeverarbeitung durch. Die Sendeverarbeitung besteht darin, die empfangenen Elemente der Umprogrammierungsfaktordaten und der Pseudodaten zu der Motorsteuereinheit **110** als ein zu umprogrammierendes Ziel zu senden, welches ähnlich zu dem ersten Ausführungsbeispiel ist (siehe Fig. 5).

[0197] Nachfolgend führt die CPU **111**, wenn die Elemente der Umprogrammierungsfaktordaten und der Pseudodaten in der Motorsteuereinheit **110** empfangen werden, wie es in **Fig.** 7 dargestellt ist, eine Umprogrammierungsverarbeitung zum Umprogrammieren von umzuprogrammierenden Datenelementen durch, welche im Voraus in dem Programmspeicher **111** gespeichert sind.

[0198] Wie es zuvor beschrieben worden ist, ist ein bestimmtes Beispiel der Datenextraktionsaufgabe in dem Schritt S134 in **Fig.** 7 verschieden zwischen dem ersten Ausführungsbeispiel und dem zweiten Ausführungsbeispiel.

[0199] Wie es in Fig. 11 dargestellt ist, werden in der Motorsteuereinheit **110** die Elemente der Umprogrammierungsfaktordaten und der Spiegeldaten in den Blöcken [1] bis [3b] geteilt und sind die Datenblöcke [1] bis [3b] in einem ersten Speicherbereich **114c** des Datenspeichers **114** als die Datenblöcke DB1 bis DB3b gespeichert worden.

[0200] Genauer gesagt ist der Satz von Gruppen G1 bis Gb aufeinanderfolgend in dem ersten Speicherbereich **114c** des Datenspeichers **114** gespeichert worden. Die Gruppe G1 besteht aus dem Paar DB1 und DB2 des Datenblocks [1] der Umprogrammierungsfaktordaten und dem Datenblock DB3 (Datenblock [1]) der Pseudodaten. Weiterhin besteht die Gruppe Gb aus dem Paar DB3b-2 und DB3b-1 des Datenblocks [b] der Umprogrammierungsfaktordaten und dem Datenblock DB3b (Datenblock [b]) der Pseudodaten.

[0201] In der Datenextraktionsaufgabe führt die CPU **111** bezüglich dem ersten Speicherbereich **114c** eine identische Überprüfung von Elementen der Umprogrammierungsfaktordaten und denjenigen der Pseudodaten derart durch, dass mindestens zwei der drei Datenelemente von jeder Gruppe zueinander identisch sind. In dem zweiten Ausführungsbeispiel führt die CPU **111**, wie es in Fig. 11 dargestellt ist, die identischen Überprüfungsvorgänge der drei Datenelemente von jeder Gruppe durch.

[0202] Als ein Ergebnis der identischen Überprüfung extrahiert die CPU **111** mindestens einen Datenblock der Umprogrammierungsfaktordaten, wenn es bestimmt wird, dass der mindestens eine Datenblock der Umprogrammierungsfaktordaten identisch zu mindest einem Datenblock der anderen Datenelemente von jeder Gruppe ist. Dann speichert CPU **111** den mindestens einen Datenblock der Umprogrammierungsfaktordaten in einem zweiten Speicherbereich **114d** des Datenspeichers **114**.

[0203] Zum Beispiel kombiniert die CPU **111**, wenn es bestimmt wird, dass die identische Überprüfung an allen Datenblöcken der Umprogrammierungsfaktordaten und der Pseudodaten gebildet ist, aufeinanderfolgend die extrahierten Datenblöcke [1] bis [b] (Datenblöcke DB1 bis DB3b-2), um dadurch die Umprogrammierungsfaktordaten zu dekodieren.

[0204] Fig. 12 stellt schematisch ein bestimmtes Beispiel der Datenextraktionsaufgabe in Schritt S134 gemäß des zweiten Ausführungsbeispiels dar.

[0205] Es ist anzumerken, dass, wie es in Fig. 12 dargestellt ist, die Datenblöcke [1] bis [3b] (siehe Fig. 11) durch [3n-2 (n ist eine Ganzzahl)], [3n-1] oder [3n] für jeden Block ausgedrückt sind.

[0206] In der Datenextraktionsverarbeitung weist die CPU **111** der Variablen n in Schritt S2100 "1" zu.

[0207] Als Nächstes wiederholt die CPU **111** den folgenden Satz von Schritten S2200, S2300, S2400 und S2700, was zulässt, dass jede Gruppe, die aus den Datenblöcken [3n-2], [3n-1], und [3n] der Umprogrammierungsfaktordaten und der Pseudodaten besteht, der Identitätsüberprüfung unterzogen wird, wie es zuvor dargelegt worden ist.

[0208] In Schritt S2200 bestimmt die CPU **111**, ob die drei Elemente [3n-2],[3n-1] und [3n] jeder Gruppe zueinander identisch sind.

[0209] In Schritt S2300 extrahiert die CPU **111** mindestens ein Element aus den drei Elementen [3n-2], [3n-1] und [3n] jeder Gruppe auf der Grundlage des Überprüfens des Ergebnisses des Schritts S2200.

[0210] In Schritt S2400 bestimmt die CPU **111** ob alle der Datenblöcke [1] bis [3b] der Identitätsüberprüfung unterzogen worden sind.

[0211] In Schritt S2700 addiert die CPU **111** [1] zu der Variablen [n].

[0212] Genauer gesagt wird in Schritt S2100 [1] zu der Variablen [n] hinzugefügt und wird danach in Schritt S2200 die Identitätsüberprüfung für die Datenblöcke [1] bis [3], die den Datenblöcken DB1 bis DB3 in den Datenblöcken [1] bis [3b] entsprechen, durchgeführt.

[0213] Genauer gesagt bestimmt die CPU **111** in einem Schritt S2201 des Schritts S2200, ob der Datenblock DB1 identisch zu dem Datenblock DB2 ist. In dem Schritt S2201 führt die CPU **111**, wenn es bestimmt wird, dass der Datenblock DB1 nicht identisch zu dem Datenblock DB2 ist (die Bestimmung in Schritt S2201 ist NEIN), ob der Datenblock DB1 identisch zu dem Datenblock DB3 in Schritt S2202. Wenn es bestimmt wird, dass der Datenblock DB1 nicht identisch zu dem Datenblock DB3 ist (die Bestimmung in dem Schritt S2202 ist NEIN), bestimmt die CPU **111**, ob der Datenblock DB2 identisch zu dem Datenblock DB3 ist, in Schritt **2203**.

[0214] In irgendeinem der Schritte S2201 bis S2203 wird die Datenidentität gebildet, wobei die CPU **111** erkennt, das mindestens zwei der drei Datenblöcke von jeder Gruppe zueinander identisch sind, anders ausgedrückt, die Identitätsüberüberprüfung wird gebildet, wobei zu einem Schritt S2300 fortgeschritten wird. In Schritt S2300 erhält die CPU **111** ein Datenelement aus den mindestens zwei Datenblöcken von jeder Gruppe.

[0215] Genauer gesagt extrahiert die CPU **111** in Schritt S2301 des Schritts S2300 den Datenblock DB1 aus den drei Datenelementen DB1 bis DB3, wenn es bestimmt wird, dass der Datenblock DB1 identisch zu mindestens einem von anderen Datenblöcken DB2 und DB3 ist, in Schritt S2201 oder S2202.

[0216] Anderseits extrahiert die CPU **111** in Schritt S23020 des Schritts S2300 den Datenblock DB2 aus den drei Datenelementen DB1 bis DB3, wenn es in Schritt S2203 bestimmt wird, dass der Datenblock DB2 identisch zu dem Datenblock DB3 ist.

[0217] Nachdem die Identitätsüberprüfung für die Blöcke [1] bis [3] gebildet worden ist, führt die CPU **111** die Schritte S2400 und S2700 aus, die zum Ausführen der nächsten Gruppe der drei Blöcke [4] bis [6] erforderlich sind.

[0218] Nachdem die Aufgabe in Schritt S2700 beendet worden ist, führt die CPU **111** den Satz von Schritten S2200, S2300, 2400 und S2700 erneut aus. Da "1" zu der Variablen "n" in Schritt S2700 hinzugefügt wird, wird die Identitätsüberprüfung an den dritten Blöcken DB4 bis DB6 angewendet, die den Blöcken [4] bis [6] entsprechen.

[0219] Genauer gesagt wird bei jedem Hinzufügen der Variablen "n" zu "1" in Schritt S2700 der Satz von drei Blöcken, in welchen die Identitätsüberprüfung angewendet wird, aufeinanderfolgend in der Reihenfolge von den drei Blöcken [4] bis [6] zu den drei Blöcken [3b-2] bis [3b] in Schritt S2200 geändert.

[0220] Als ein Ergebnis von wiederholten Ausführungen des Satzes von Schritten S2200, S2300, S2400 und S2700 führt die CPU **111** in Schritt S2400, wenn es bestimmt wird, das alle der Datenblöcke [1] bis [3b] der Identitätsüberprüfung unterzogen werden, den Satz von Schritten S2500 und S2600 aus.

[0221] Genauer gesagt liest die CPU **111** in Schritt S2500, wie es in Fig. 11 dargestellt ist, die Datenblöcke DB1, DB4,..., DB3b-2, die extrahiert worden sind, aus den gültig gemachten Daten aus den Datenblöcken DB1 bis DB3b der Umprogrammierungsfaktordaten und der Pseudodaten aus. Als Nächstes kombiniert die CPU **111** aufeinanderfolgend die ausgelesenen Datenblöcke DB1, DB4,..., DB3b-2, um die Umprogrammierungsfaktordaten zu dekodieren, und speichert den kombinierten Satz der Datenblöcke DB1, DB4,..., DB3b-2 in dem zweiten Speicherbereich **144d** des Datenspeichers **114**.

[0222] Wenn die Umprogrammierungsfaktordaten als ein Ergebnis der Datenextraktionsaufgabe, die in Fig. 12 dargestellt ist, dekodiert werden, setzt die CPU **111** ein Identitätsmerker in zum Beispiel dem Datenspeicher **114** in Schritt S2600; dieser Identitätsmerker stellt dar, dass die Identitätsüberprüfung gebildet ist. Dies lässt zu, dass die CPU **111** auf der Grundlage des Zustands des Spiegelmerkers bestimmt, ob die Umprogrammierungsfaktordaten dekodiert sind, und danach verlässt die CPU **111** die Datenextraktionsverarbeitung.

[0223] Als ein Ergebnis eines wiederholten Ausführens des Satzes von Schritten S2200, S2300, S2400 und S2700 kann es bestimmt werden, dass in Schritt S2200 (Schritt S2203) alle der drei Blöcke von einer Gruppe nicht zueinander identisch sind, anders ausgedrückt, die Identitätsüberprüfung nicht gebildet ist (die vorbestimmte Regel ist nicht erfüllt). In diesem Fall bestimmt die CPU **111** wahrscheinlich, dass die Daten, die von der Verwaltungszentrale **200** gesendet werden, nicht autorisierte Daten und/oder illegal geänderte Daten sind, um dadurch die Datenextraktionsaufgabe nach einem derartigen Bestimmen zu verlassen, dass die Identitätsüberprüfung ohne ein Setzen des Identitätsmerkers nicht gebildet ist.

[0224] In Schritt S135 (siehe **Fig.** 7) wird deshalb die Bestimmung bezüglich dessen, ob die Umprogrammierungsfaktordaten dekodiert sind, abhängig von dem Zustand des Identitätsmerkers aufgeführt.

[0225] Wie es zuvor beschrieben worden ist, kann das in das Fahrzeug eingebaute Steuersystem **100** gemäß dem zweiten Ausführungsbeispiel im Wesentlichen die gleichen Effekte wie die ersten und zweiten Effekte gemäß dem ersten Ausführungsbeispiel erzielen.

Drittes Ausführungsbeispiel

[0226] Ein in ein Fahrzeug eingebautes Steuersystem gemäß einem dritten Ausführungsbeispiel der vorliegenden Erfindung wird hier im weiteren Verlauf beschrieben. Das in ein Fahrzeug eingebaute Steuersystem des dritten Ausführungsbeispiels weist im Wesentlichen den gleichen Aufbau wie den des in das Fahrzeug eingebauten Steuersystems des zweiten Ausführungsbeispiels auf. Aus diesem Grund sind ähnliche Bezugszeichen ähnlichen Teile dem in ein Fahrzeug eingebauten Steuersystemen gemäß den zweiten und dritten Ausführungsbeispielen zugewiesen, so dass Beschreibungen der Teile des in das Fahrzeug eingebauten Steuersystems des dritten Ausführungsbeispiels weggelassen werden.

[0227] Im Wesentlichen ähnlich dem zweiten Ausführungsbeispiel ist das in das Fahrzeug eingebaute Steuersystem **100** in dem dritten Ausführungsbeispiel ausgelegt, um:
eine Mehrzahl von Elementen der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) zu empfangen, zu welchen der gleiche Identifizierer hinzugefügt worden ist, welcher von der Verwaltungszentrale **200** gesendet worden ist;
die empfangenen Elemente der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) gegeneinander zu überprüfen; und
mindestens eines der Elemente der Umprogrammierungsdaten als gültig gemachte Daten zu extrahieren, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass mindestens einer der Elemente der Umprogrammierungsdaten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element erfüllt.

[0228] Es ist anzumerken, dass in dem dritten Ausführungsbeispiel, wie es in den **Fig.** 13 und Fig. 14 verglichen mit den **Fig.** 10 und Fig. 11 dargestellt ist, ein Beispiel, wie die Umprogrammierungsfaktordaten und die Pseudodaten von der Verwaltungszentrale zu senden sind, und das, wie die Umprogrammierungsfaktordaten von diesen zu extrahieren sind, von denjenigen in dem zweiten Ausführungsbeispiel verschieden sind.

[0229] Die Sendevorgänge der Umprogrammierungsfaktordaten und der Spiegeldaten der Umprogrammierungsdaten durch die Verwaltungszentrale **200** werden hier im weiteren Verlauf unter Bezugnahme auf die Fig. 13A und Fig. 13B beschrieben.

[0230] Fig. 13A stellt schematisch ein Beispiel des Aufbaus der Daten dar, die zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** kommunizierbar sind. Fig. 13B stellt schematisch ein Beispiel des Aufbaus des Verwaltungszentralenspeichers dar, in welchem die Elemente der Umprogrammierungsdaten im Voraus gespeichert sind. Fig. 13B stellt schematisch ebenso ein Beispiel dar, wie die Umprogrammierungsdaten von der Verwaltungszentrale **200** zu senden sind.

[0231] Wie es in Fig. 13A dargestellt ist, wird in dem dritten Ausführungsbeispiel jedes der Elemente von Umprogrammierungsdaten zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** als ein Datenrahmen DF verarbeitet. Der Datenrahmen DF ist eine geschlossene Einheit von Daten, die zwischen der Verwaltungszentrale **200** und dem in ein Fahrzeug eingebauten Steuersystem **100** zu verarbeiten sind.

[0232] Genauer gesagt besteht der Datenrahmen DF aus einer Mehrzahl von Feldern und einem Überprüfungscode. Der Datenrahmen DF ist dazu ausgelegt, in einem Paket zu senden. Es ist anzumerken, dass, wenn jedes der Elemente der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) als der Datenrahmen DF verarbeitet wird, der gleiche Identifizierer jedem der Elemente der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) hinzugefügt wird.

[0233] Der Datenaufbau des Datenrahmens DF ist im Wesentlichen identisch zu dem des Datenrahmens DF gemäß jedem der ersten und zweiten Ausführungsbeispiele.

[0234] Wie es in Fig. 13B dargestellt ist, werden in der Verwaltungszentrale **200** gemäß dem dritten Ausführungsbeispiel erste bis dritte Pseudodaten aus nicht gültig gemachten Daten verwendet, die unvereinbar mit den Umprogrammierungsfaktordaten (Gültigmachungsdaten) sind.

[0235] Alle der Umprogrammierungsfaktordaten und alle der ersten bis dritten Pseudodaten sind in eine Mehrzahl von Datenblöcken [1] bis [c] geteilt, wobei "c" eine vorbestimmte Ganzzahl darstellt. Die Datenblöcke [1] bis [c] der Umprogrammierungsfaktordaten werden im Voraus in einem ersten Speicherbereich **201e** der Zentrale **200** gespeichert. Ähnlich werden die Datenblöcke [1] bis [c] der ersten bis drit-

ten Pseudodaten im Voraus in zweiten bis vierten Speicherbereichen **201f** bis **201h** der Zentrale **200** gespeichert.

[0236] Beim Senden der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten liest die Verwaltungszentrale **200** aufeinanderfolgend die Umprogrammierungsfaktordaten und alle der ersten bis dritten Pseudodaten in der Reihenfolge von dem Block [1] zu dem Block [c] unter den folgenden ersten bis dritten Zuständen aus:
der erste Zustand ist, das fünf Datenblöcke in jedem der Datenblöcke [1] bis [c] kombiniert werden, um ausgelesen zu werden.

[0237] Der zweite Zustand ist, das jede kombinierte Gruppe von fünf Blöcken mindestens einen der ersten bis dritten Pseudodaten beinhalten.

[0238] Der dritte Zustand ist, dass die größte Anzahl von identischen Blöcken in einer Gruppe bezüglich allen der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten der Anzahl von identischen Blöcken für die Umprogrammierungsfaktordaten entspricht.

[0239] Zum Beispiel wird, wie es in Fig. 13B dargestellt ist, nach einem doppelten Auslesen von einem Block [1] aus den Umprogrammierungsfaktordaten ein Block [1] von einen der ersten bis dritten Pseudodaten extrahiert. Das doppelte Auslesen des Blocks [1] der Umprogrammierungsfaktordaten wird mit den entsprechenden ausgelesenen Blöcken [1] der ersten bis dritten Pseudodaten kombiniert, um eine Gruppe G1 auszubilden.

[0240] Da die Gruppe G1 aus den fünf Blöcken [1] besteht, ist der erste Zustand erfüllt, und da die Gruppe G1 aus den Blöcken [1] der ersten bis dritten Pseudodaten besteht, ist der zweite Zustand erfüllt. Weiterhin ist die größte Anzahl von identischen Blöcken wie in einer Gruppe (fünf kombinierte Blöcke) bezüglich alle drei Umprogrammierungsfaktordaten und den ersten bis dritten Pseudodaten "2" bezüglich den Umprogrammierungsfaktordaten "2".

[0241] Dies lässt zu, dass der Satz der Gruppe G1 zu einer Gruppe Gc erzeugt wird. Die Gruppe G1 besteht aus den fünf kombinierten Blöcken DB1 bis DB5 der Datenblöcke [1] der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten. Die Gruppe Gc besteht aus den fünf kombinierten Blöcken DB5c-4 bis DB5c der Datenblöcke [c] der Umprogrammierungsfaktordaten und der ersten Pseudodaten, die zu erzeugen sind.

[0242] Die Verwaltungszentrale **200** kategorisiert Datenblöcke DB1, DB2, DB3,..., DB5c, derart, dass eine Mehrzahl der Datenrahmen DF die Datenblöcke DB1, DB2, DB3,..., DB5c, enthält. Als Nächstes liest die Verwaltungszentrale **200** aufeinanderfolgend die Mehrzahl der Datenrahmen DF in einem Paket bzw. in einem Block zu der in ein Fahrzeug eingebauten Steuereinheit **100**.

[0243] Die Elemente der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten werden von der Verwaltungszentrale **200** auf die zuvor beschriebene Sendeweise gesendet. Die CPU **111** der Motorsteuereinheit **100** empfängt die Elemente der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten und speichert diese, wie es in Fig. 14 dargestellt ist, in einem Speicherbereich **114e** des Datenspeichers **114** als die Blockdaten [1] bis [5c], die den Datenblöcken DB1 bis DB5c entsprechen.

[0244] In der Datenextraktionsaufgabe bezüglich dem ersten Speicherbereich **114e** extrahiert die CPU **111** Umprogrammierungsfaktordaten in einem Schritt S134 mit einer Gültigkeit auf die folgenden Weise.

[0245] Genauer gesagt führt in dem dritten Ausführungsbeispiel die CPU **111** eine Majoritätsregelüberprüfung von Elementen der Umprogrammierungsfaktordaten und Pseudodaten der ersten und dritten Pseudodaten durch, um zu bestimmten, ob es eine größte Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierte Elemente) der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten jeder Gruppe gibt.

[0246] Als ein Ergebnis der Majoritätsregelüberprüfung wird, wenn es bestimmt wird, dass es die größte Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierte Elemente) bezüglich den Umprogrammierungsfaktordaten und den ersten bis dritten Pseudodaten gibt, die Majoritätsregelüberprüfung gebildet, anders ausgedrückt, die vorbestimmte Regel ist erfüllt. Dann extrahiert die CPU **111** als die gültig gemachten Daten mindestens einer der identischen Datenelemente (Blöcke) von jeder Gruppe aus den fünf Datenelementen von diesen. Danach kombiniert die CPU **111**, wie es in Fig. 14 dargestellt ist, aufeinanderfolgende extrahierte Datenblöcke DB1, DB7,..., DB5c-4, um dadurch diese in einem zweiten Speicherbereich **114f** des Datenspeichers **114** zu speichern. Dies lässt zu, dass die Umprogrammierungsfaktordaten in dem zweiten Speicherbereich **114f** des Datenspeicherspeichers **114** dekodiert werden.

[0247] Fig. 15 stellt schematisch ein bestimmtes Beispiel der Datenextraktionsaufgabe in Schritt **134** gemäß dem dritten Ausführungsbeispiel dar.

[0248] Es ist anzumerken, dass, wie es in Fig. 15 dargestellt ist, die Datenblöcke [1] bis [5c] (siehe Fig. 14) durch [5n-4 (n ist eine Ganzzahl)], [5n-3], [5n-2], [5n-1] oder [5n] für jeden Block ausgedrückt

sind.

[0249] Wie es in Fig. 15 dargestellt ist, weist die CPU **111** in der Datenextraktionsverarbeitung der Variablen n in einem Schritt S3001 und "1" zu.

[0250] Als Nächstes wiederholt die CPU **111** den folgenden Satz von Schritten S3100, S3200, S3300, S3400 und S3700, was zulässt, dass jede Gruppe, die aus den fünf Datenblöcken [5n-4] bis [5n] der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten für jede Gruppe besteht, der Majoritätsregelüberprüfung unterzogen wird, wie sie vorhergehend dargelegt worden ist:
In Schritt S3100 bestimmt die CPU **111**, ob die fünf Blöcke [5n-4] bis [5n] jeder Gruppe zueinander identisch sind.

[0251] In Schritt S3200 führt die CPU **111** eine Majoritätsregelüberprüfungsaufgabe durch um:
die fünf Blöcke [5n-4 bis [5n] jeder Gruppe der Majoritätsregelüberprüfung auf der Grundlage des Ergebnisses eines Identitätsvergleichs in dem Schritt S3100 zu unterziehen; und
einen Majoritätsregelmerker unter der Bedingung zu setzen, dass die Majoritätsregelüberprüfung gebildet wird.

[0252] In Schritt S3300 überprüft die CPU **111** den Zustand des Majoritätsregelmerkers.

[0253] In Schritt S3400 bestimmt die CPU **111**, ob alle der Datenblöcke [1] bis [5c] der Majoritätsregelüberprüfung unterzogen werden.

[0254] In Schritt S3700 fügt die CPU **111** [1] der Variablen [n] hinzu.

[0255] Genauer gesagt wird in Schritt S3100, wie es in Fig. 16A dargestellt ist, der Datenblock [5n-4] in den fünf Datenblöcken [5n-4] bis [5n] als eine Referenz gegenüber den verbleibenden Datenblöcken darin verwendet, und es wird bestimmt, ob jeder der verbleibenden Datenblöcke [5n-3] bis [5n] zu dem Referenzdatenblock [5n-4] identisch ist.

[0256] Nach dem Identitätsvergleich unter Verwendung des Datenblocks [5n-4] als eine Referenz wird, wie es in Fig. 16B dargestellt ist, der Datenblock [5n-3] als eine Referenz gegenüber den verbleibenden Datenblöcken darin verwendet, und es wird bestimmt, ob jeder der verbleibenden Datenblöcke [5n-4] und [5n-2] bis [5n] zu dem Referenzdatenblock [5n-3] identisch ist.

[0257] Nach dem Identitätsvergleich unter Verwendung des Datenblocks [5n-3] als eine Referenz wird, wie es in Fig. 16C dargestellt ist, der Datenblock [5n-2] als eine Referenz gegenüber den verbleibenden Datenblöcken darin verwendet, und es wird bestimmt, ob jeder der verbleibenden Datenblöcke [5n-4], [5n-3], [5n-1] und [5n] zu dem Referenzdatenblock [5n-2] identisch ist.

[0258] Nach dem Identitätsvergleich unter Verwendung des Datenblocks [5n-2] als eine Referenz wird, wie es in Fig. 16D dargestellt ist, der Datenblock [5n-1] als eine Referenz gegenüber den verbleibenden Datenblöcken darin verwendet, und es wird bestimmt, ob jeder der verbleibenden Datenblöcke [5n-4], [5n-3], [5n-2] und [5n] zu dem Referenzdatenblock [5n-1] identisch ist.

[0259] Wie es in den Fig. 16A bis Fig. 16D dargestellt ist, werden die vier Zahlen von "2", "1", "0" und "1" von identischen Blöcken in den fünf Datenblöcken [5n-4] bis [5n] bezüglich den Referenzen von Blöcken [5n-4], [5n-3], [5n-2] und [5n-1] erzielt. In diesem Fall würde es in Schritt S3200 bestimmt, dass die Zahl von identischen Blöcken von "2" in dem Fall eines Verwendens der Referenz des Blocks [5n-4] die größte Zahl ist. Dann wird der Datenblock [5n-4] als die gültig gemachten Daten aus den fünf Datenblöcken [5n-4] bis [5n] extrahiert und wird deshalb der Majoritätsregelmerker gesetzt; dieser Majoritätsregelmerker stellt dar, dass die Majoritätsregelüberprüfung gebildet ist.

[0260] In Schritt S3400 wird es unter der Bedingung, dass der Majoritätsregelmerker gesetzt ist (siehe Schritt S3300), bestimmt, ob alle Blöcke [1] bis [5c] der Majoritätsregelüberprüfung unterzogen werden. Wenn es bestimmt wird, dass alle Blöcke [1] bis [5c] nicht der Majoritätsregelüberprüfung unterzogen werden, wird "1" der Variablen "n" durch die CPU **111** in Schritt S3700 hinzugefügt und werden danach die Schritte S3100 und S3200 erneut ausgeführt, so dass die fünf Datenblöcke der nächsten Gruppe der Majoritätsregelüberprüfung unterzogen werden.

[0261] Als ein Ergebnis von wiederholten Ausführungen des Satzes von Schritten S3100, S3200, S3300, S3400 und S3700 führt die CPU **111** in einem Schritt S3400, wenn es bestimmt wird, das alle der Datenblöcke [1] bis [5c] bereits der Majoritätsregelüberprüfung unterzogen worden sind, den Satz von Schritten S3500 und S3600 aus.

[0262] Genauer gesagt liest die CPU **111** in einem Schritt S3500, wie es in Fig. 14 dargestellt ist, die Datenblöcke DB1, DB7,..., DB5c-4, die extrahiert worden sind, als die gültig gemachten Daten aus den Datenblöcken DB1 bis DB5c der Umprogrammierungsfaktordaten und der Pseudodaten für jede Gruppe aus. Als Nächstes kombiniert die CPU **111** aufeinanderfolgend der ausgelesenen Datenblöcke DB1, DB7,..., DB5c-4, um die Umprogrammierungsfaktordaten zu dekodieren und den kombinierten Satz der Datenblöcke DB1, DB7,..., DB5c-4 in dem zweiten Speicherbereich **114f** des Datenspeichers **114** zu

speichern.

[0263] Wenn die Umprogrammierungsfaktordaten als ein Ergebnis der Datenextraktionsaufgabe, wie in Fig. 15 dargestellt ist, dekodiert werden, wird der Majoritätsregelmerker in zum Beispiel dem Datenspeicher **114** gesetzt. Dies lässt zu, dass die CPU **111** auf der Grundlage des Zustands des Majoritätsregelmerkers bestimmt, ob die Umprogrammierungsfaktordaten dekodiert worden sind, und danach definiert die CPU **111** den Zustand des Majoritätsregelmerkers, worauf die Datenextraktionsverarbeitung verlassen wird.

[0264] Als ein Ergebnis eines wiederholten Ausführens des Satzes von Schritten S3100, S3200, S3300, S3400 und S3700 bestimmt die CPU **111** in dem Schritt S3300 wahrscheinlich, wenn kein Majoritätsregelmerker gesetzt werden kann, dass die Daten, die von der Verwaltungszentrale **200** gesendet werden, nicht autorisierte Daten und/oder illegal geänderte Daten sind. Daher verlässt die CPU **111** die Datenextraktionsaufgabe nach einem derartigen Bestimmen, dass der identische Vergleich nicht ohne ein Setzen des Majoritätsregelmerkers gebildet wird.

[0265] In Schritt S135 (siehe **Fig.** 7) wird deshalb die Bestimmung, ob die Umprogrammierungsfaktordaten dekodiert werden, abhängig von dem Zustand des Majoritätsregelmerkers ausgeführt.

[0266] Fig. 17 ist en Flussdiagramm, das schematisch ein Beispiel des Algorithmus des Identitätsvergleichs in Schritt S3100 in Fig. 15 darstellt, und Fig. 18 ist ein Flussdiagramm, das schematisch ein Beispiel des Algorithmus der Majoritätsregelüberprüfung in Schritt S3200 von Fig. 15 darstellt.

[0267] Der Identitätsvergleich in Schritt S3100 wird zuerst unter Bezugnahme auf Fig. 17 beschrieben.

[0268] Wie es zuvor dargelegt worden ist, werden in dem Identitätsvergleich, wie es in den Fig. 16A bis Fig. 16D dargestellt ist, die kombinierten fünf Datenblöcke [5n-4] bis [5n] von jeder Gruppe mit einem der Blöcke [5n-4] bis [5n] als eine zu ändernde Referenz verglichen. Dies bestimmt die vier Zahlen von identischen Blöcken in den fünf Datenblöcken [5n-4] bis [5n] bezüglich den Referenzen von [5n-4], [5n-3], [5n-2] bzw. [5n-1]. Die bestimmten vier Zahlen von identischen Blöcken in den fünf Datenblöcken [5n-4] bis [5n] werden als Vergleichsergebnisse in dem Identitätsvergleich in dem Datenspeicher **114** gespeichert. In dem Speicher **114** werden die bestimmten vier Zahlen von identischen Blöcken entsprechenden der Referenzen [5n-4] bis [5n-1] zugewiesen.

[0269] In dem dritten Ausführungsbeispiel werden, wie es in Fig. 17 dargestellt ist, Variablen "j", "k" und eine Mehrzahl von identischen Anzahlzählern auf die hier im weiteren Verlauf beschriebene Weise manipuliert, um den Identitätsvergleich zu realisieren. Es ist anzumerken, dass die inneren Zähler der CPU **111** als die Mehrzahl von identischen Anzahlzählern verwendet werden können, aber andere Zähler, als die Mehrzahl von identischen Anzahlzählern vorbereitet und verwendet werden.

[0270] Wie es in Fig. 17 dargestellt ist, löscht die CPU **111** in einem Identitätsvergleich den Zählwert von jedem der identischen Anzahlzähler. Als Nächstes setzt die CPU **111** die Variable "j" auf "0" und die Variable "k" auf "1" und danach wiederholt die CPU **111** den folgenden Satz von Schritten S3103 bis S3108. Dies lässt zu, dass die fünf Datenblöcke [5n-4] bis [5n] dem Identitätsvergleich auf der Grundlage des Datenblocks [5n-4] als eine Referenz gegenüber den verbleibenden Datenblöcken darin unterzogen werden, um dadurch zu bestimmen, ob jeder der verbleibenden Datenblöcke [5n-3] bis [5n] identisch zu dem Referenzdatenblock [5n-4] ist.

[0271] In Schritt S3103 berechnet die CPU **111** die Summe der Variablen "j" und "k", um den Summenwert mit der Kombinationsanzahl "5" der fünf verglichenen Blöcke zu vergleichen.

[0272] In Schritt S3104 vergleicht die CPU **111** den Datenblock [5n-4+j] mit dem Datenblock [5n-4+j+k] unter der Bedingung, dass der Summenwert kleiner als die Kombinationsanzahl "5" als ein Ergebnis des Vergleichs in Schritt S3103 ist.

[0273] In Schritt S3105 überprüft die CPU **111**, ob der Datenblock [5n-4+j] unter der Bedingung, dass der Datenblock [5n-4+j] als ein Ergebnis des Überprüfens in Schritt S3104 identisch zu dem Datenblock [5n-4+j+k] ist, bereits als Kandidatendaten gespeichert worden ist. Die Kandidatendaten sollten als ein Datenblock extrahiert werden, dessen Anzahl von Blöcken, die in den fünf Datenblöcken identisch zu diesen sind, die größte darin ist.

[0274] In Schritt S3106 speichert die CPU **111** den Datenblock [5n-4+j] als die Kandidatendaten [5n-4+j] unter der Bedingung, dass der Datenblock [5n-4+j] als ein Ergebnis des Überprüfens in Schritt S3105 noch nicht als die Kandidatendaten gespeichert worden ist.

[0275] In Schritt S3107 zählt die CPU **111** den Zählwert des identischen Anzahlzählers [5n-4+j], der die identische Anzahl (Vergleichsergebnis) des Datenblocks [5n-4+j] darstellt, unter der Bedingung, dass der Datenblock [5n-4+j] als ein Ergebnis des Vergleichs in Schritt S3104 identisch zu dem Datenblock [5n-4+j+k] ist.

[0276] In Schritt S3108 addiert die CPU **111** "1" zu der Variablen "k".

[0277] Zum Beispiel wird es in Schritten S3101 und S3102, wenn der Zählwert des identischen Zählers [5n-4+j] nicht gelöscht worden ist und die Variablen "j" und "k" auf "0" bzw. "1" gesetzt worden sind, in Schritt S3103 bestimmt, dass der Summenwert der Variablen "j" und "k" kleiner als der Referenzwert "5" ist.

[0278] Als Nächstes wird in einem Schritt S3104 der Datenblock [5n-4], der als eine Referenz verwendet wird, mit dem Datenblock [5n-3] verglichen, um zu bestimmen, ob der Datenblock [5n-4] identisch zu dem Datenblock [5n-3] ist.

[0279] Wenn es bestimmt wird, dass der Datenblock [5n-4] identisch zu dem Datenblock [5n-3] ist, wird in Schritten S3105 und S3106 der Datenblock [5n-4] als die Kandidatendaten [5n-4] gespeichert, die als Datenblock extrahiert werden sollten, dessen Anzahl von Blöcken, die in diesem mit einer Gruppe (fünf kombinierten Blöcken) identisch sind, die größte darin ist.

[0280] Als Nächstes wird als der Vorgang in Schritt S3107 der Zählwert des identischen Anzahlzählers [5n-4], der die identische Anzahl des Datenblocks [5n-4] darstellt, aufwärts gezählt (inkrementiert), so dass der Zählwert "1" des identischen Anzahlzählers [5n-4] erzielt wird. Deshalb wird in Schritt S3108 "1" zu der Variablen "k" addiert

[0281] In Schritt S3104 kann es bestimmt werden, dass der Datenblock [5n-4] nicht identisch zu dem Datenblock [5n-3] ist. In diesem Fall geht die CPU **111** zu dem Schritt S3108 ohne ein Ausführen des Speichervorgangs der Kandidatendaten [5n-4] in Schritt S3106 und des Zählermanipulationsvorgangs in Schritt S3107. Das heißt, die CPU **111** addiert "1" zu der Variablen "k" als Reaktion auf den Zeitpunkt, zu dem es bestimmt wird, dass der Datenblock [5n-4] nicht identisch zu dem Datenblock [5n-3] ist.

[0282] Nach einem Ausführen des Vorgangs in Schritt S3108, der zuvor dargelegt worden ist, wird der Satz von aufeinanderfolgenden Schritten S3103 bis S3108 erneut ausgeführt. Es ist anzumerken, dass bei diesem Ausführen der Vorgänge in den Schritten S3103 bis S3108, da "1" zu der Variablen "k" addiert wird, in Schritt S3104 der Datenblock [5n-4], der als eine Referenz verwendet wird, mit dem Datenblock [5n-2] verglichen wird, um zu bestimmen, ob der Datenblock [5n-4] identisch zu dem Datenblock [5n-2] ist.

[0283] Genauer gesagt wird zu jeder Zeit, zu der "1" zu der Variablen "k" addiert wird, das Ziel eines Identitätsvergleichs des Referenzdatenblocks [5n-4] aufeinanderfolgend in der Reihenfolge des Datenblocks [5n-3], des Datenblocks [5n-2], des Datenblocks [5n-1] und des Datenblocks [5n] in Schritt S3104 verschoben. Der Zählwert des identischen Anzahlzählers [5n-4], der durch die Vorgänge in Schritten S3103 bis S3106 zu manipulieren ist, zeigt letztlich die Anzahl von identischen Blöcken in einer Gruppe in dem Fall eines Verwendens des Datenblocks [5n-4] als die Referenz, wie zum Beispiel "2" in Fig. 16A als ein Beispiel in Schritt S3107.

[0284] Als ein Ergebnis des wiederholten Ausführens der Vorgänge in Schritten S3103 bis S3108 geht der Summenwert der Variablen "j" und "k" in Schritt S3103 zu "5", so dass es bestimmt wird, dass der Summenwert gleich oder größer als der Referenzwert "5" ist. Dies bewirkt, dass das Ausführen der CPU zu Schritt S3111 verschoben wird.

[0285] Genauer gesagt ist in diesem Fall die Variable "k" "5", so dass der Identitätsvergleich zischen den fünf kombinierten Datenblöcken [5n-4] bis [5n] unter Verwendung des Datenblocks [5n-4] als eine Referenz beendet worden ist. In Schritt 3111 wird deshalb, wie es in Fig. 16B dargestellt ist, als eine Aufgabe zum Verschieben des Datenblocks, der als eine Referenz verwendet wird, gegenüber den verbleibenden Datenblöcken von dem Datenblock [5n-4] zu dem Datenblock [5n-3] "1" zu der Variablen "j" hinzugefügt. Parallel zu dieser Additionsaufgabe wird die Variable "k" auf "1" gesetzt, anders ausgedrückt, wird gelöscht, um den Identitätsvergleich zwischen den fünf kombinierten Blöcken [5n-4] bis [5n] unter Verwendung des Datenblocks [5n-3] als eine Referenz auszuführen.

[0286] Als Nächstes werden die Vorgänge in Schritten S3103 bis S3108 wiederholt erneut unter der Bedingung ausgeführt, dass die Variable "j" kleiner als der Referenzwert "4" ist, der durch Subtrahieren von "1" von dem vorhergehenden Referenzwert "5" erzielt wird. Dies lässt den Identitätsvergleich zwischen den fünf kombinierten Datenblöcken [5n-4] bis [5n] unter Verwendung des Datenblocks [5n-3] als eine Referenz zu.

[0287] Es ist anzumerken, dass in dem dritten Ausführungsbeispiel, wie es in Fig. 16B dargestellt ist, der Identitätsvergleich zwischen dem Datenblock [5n-4] und dem Referenzdatenblock [5n-3] nicht ausgeführt wird. Genauer gesagt ist der Identitätsvergleich zwischen dem Datenblock [5n-4] und dem Referenzdatenblock [5n-3] durch den Identitätsvergleich unter Verwendung des Datenblocks [5n-4] als eine Referenz bereits ausgeführt worden. Aus diesem Grund lässt ein Weglassen des Identitätsvergleichs zwischen dem Datenblock [5n-4] und dem Referenzdatenblock [5n-3] zu, dass die Verarbeitungslast der Datenextraktionsverarbeitung in Schritt S135 verringert wird. Es ist anzumerken, dass das Weglassen ähnlich an dem Identitätsvergleich unter Verwendung der Datenblöcke [5n-2] und [5n-1] als Referenzen angewendet wird. Weiterhin ist es anzumerken, dass der Identitätsvergleich unter Verwendung des Daten-

blocks [5n] als eine Referenz nicht ausgeführt wird.

[0288] Nach dem aufeinanderfolgenden wiederholten Ausführen des Satzes von Vorgängen in Schritten S3103 bis S3108 aufeinanderfolgend unter Verwendung der Datenblöcke [5n-3], [5n-2] und [5n-1] als Referenzen durch die Vorgänge in Schritt S3111 und S3112 wird es letztlich in Schritt S3112 bestimmt, dass die Variable "j" nicht kleiner als "4" ist. Dann ist der Identitätsvergleich zwischen den fünf kombinierten Datenblöcken [5n-4] bis [5n] beendet, so dass der Identitätsvergleich in Schritt S3100 zu dem Zeitpunkt beendet wird, zu dem es bestimmt wird, dass die Variable "j" nicht kleiner als "4" ist.

[0289] Wie es vorhergehend dargelegt worden ist, werden jedoch nach Schritt S3100 in Schritt S3700, wenn [1] zu der Variablen "n" addiert wird, die nächsten fünf kombinierten Gruppen den Identitätsvergleich unterzogen, wie es in Fig. 17 dargestellt ist.

[0290] Als Nächstes wird die Majoritätsregelüberprüfung in Schritt S3200 unter Bezugnahme auf Fig. 18 beschrieben.

[0291] Wie es vorhergehend dargelegt worden ist, werden in dem Identitätsvergleich, wie es in den Fig. 16A bis Fig. 16D dargestellt ist, die Anzahlen von identischen Blöcken in einer Gruppe in dem Fall eines Verwendens der Datenblöcke [5n-4] bis [5n-1] als die Referenz jeweils durch die Zählwerte der identischen Anzahlzähler [5n-4] bis [5n-1] berechnet.

[0292] In der Majoritätsregelüberprüfung wird es, wie es zuvor dargelegt worden ist, bestimmt, ob einer der Datenblöcke, dessen Anzahl von Blöcken identisch zu diesen in einer Gruppe (fünf kombinierte Datenblöcke [5n-4] bis [5n]) ist, der größte darin ist. Weiterhin wird der Majoritätsregelmerker unter einer Bedingung gesetzt, dass es einen der Datenblöcke gibt, dessen Anzahl von Blöcken, die darin in einer Gruppe (fünf kombinierten Datenblöcke) identisch sind, die größte darin ist.

[0293] In dem dritten Ausführungsbeispiel werden, wie es in Fig. 18 dargestellt ist, eine Variable "j" und ein Maximalzähler, der die Anzahl von identischen Blöcken von einem der Datenblöcke anzeigt, deren Anzahl von Blöcken, die darin identisch sind, die größte in einer Gruppe (fünf Datenblöcke [5n-4] bis [5n]) ist, auf die hier im weiteren Verlauf beschriebene Weise manipuliert, um die Majoritätsregelüberprüfung zu realisieren. Es ist anzumerken, dass ein innerer Zähler der CPU **111** als der Maximalzähler verwendet werden kann, aber ein äußerer Zähler als der Max-Zähler vorbereitet und verwendet werden kann.

[0294] Wie es in Fig. 18 dargestellt ist, löscht die CPU **111** bei einer Majoritätsregelüberprüfung den Zählwert des Maximalzählers in Schritt S3201. Als Nächstes setzt die CPU **111** die Variable "j" auf "0" und setzt den Majoritätsregelmerker zurück. Danach wiederholt die CPU **111** den folgenden Satz von Schritten S3203 bis S3208, S3211 und S3212. Dies ermöglicht es, zu bestimmen, ob es einen in den Datenblöcken gibt, dessen Anzahl von Blöcken, die darin in einer Gruppe (fünf kombinierten Datenblöcken [5n-4] bis [5n]) identisch sind, die größte darin ist:
In Schritt S3203 vergleicht die CPU **111** die Variable "j" mit dem Wert "4", der durch Subtrahieren von "1" von dem Referenzwert "5" erzielt wird.

[0295] In Schritt S3204 vergleicht die CPU **111** den Zählwert des identischen Anzahlzählers [5n-4+j] mit dem Zählwert des Maximalzählers unter der Bedingung, dass die Variable "j" kleiner als der Wert "4" ist, als ein Ergebnis des Vergleichs in Schritt S3203. Der Zählwert des identischen Anzahlzählers [5n-4+j] stellt die Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierten Datenblöcken [5n-4] bis [5n]) bezüglich dem Referenzdatenblock [5n-4+j] dar.

[0296] In Schritt S3205 zählt die CPU **111** den Zählwert des Maximalzählers derart aufwärts, dass der Zählwert des Maximalzählers mit dem Zählwert des identischen Anzahlzählers [5n-4+j] unter der Bedingung übereinstimmt, dass der Zählwert des identischen Anzahlzählers [5n-4+j] größer als der Zählwert des Maximalzählers als ein Ergebnis des Vergleichs in Schritt S3204 ist.

[0297] In Schritt S3206 speichert die CPU **111** den Datenblock [5n-4+j] (die Kandidatendaten [5n-4+j]) in zum Beispiel dem Datenspeicher **114** als die gültig gemachten Daten, die den Datenblock darstellen, dessen identische Anzahl die größte in den fünf Datenblöcken ist, unter der Bedingung, dass der Zählwert des identischen Anzahlzählers [5n-4+j] größer als der Zählwert des Maximalzählers als ein Ergebnis des Vergleichs in Schritt S3204 ist.

[0298] In Schritt S3207 setzt die CPU **111** den Majoritätsregelmerker unter der Bedingung, dass der Zählwert des identischen Anzahlzählers [5n-4+j] größer als der Zählwert des Maximalzählers als ein Ergebnis des Vergleichs in Schritt S3204 ist.

[0299] In Schritt S3208 addiert die CPU **111** "1" zu der Variablen "j".

[0300] In Schritt S3211 vergleicht die CPU **111** den Zählwert des identischen Anzahlzählers [5n-4+j] mit dem Zählwert des Maximalzählers unter der Bedingung, dass die Variable "j" als Ergebnis des Vergleichs im Schritt S3203 nicht kleiner als der Wert "4" ist. Der Zählwert des identischen Anzahlzählers [5n-4+j] stellt die Anzahl von identischen Blöcken in einer Gruppe (den fünf kombinierten Datenblöcken [5n-4] bis [5n]) bezüglich dem Referenzdatenblock [5n-4+j] dar.

[0301] In Schritt S3212 setzt die CPU **111** den Majoritätsregelmerker unter der Bedingung zurück, dass der Zählwert des identischen Anzahlzählers [5n-4+j] identisch zu dem Zählwert des Maximalzählers ist.

[0302] Zum Beispiel wird es in Schritten S3201 und S3202, wenn der Zählwert des Maximalzählers gelöscht worden ist und die Variable "j" auf "0" gesetzt ist, bestimmt, dass die Variable "j" kleiner als der Wert "4" ist, der durch Subtrahieren von "1" von dem Referenzwert "5" in Schritt S3203 erzielt wird.

[0303] Als Nächstes wird in einem Schritt S3204 der Zählwert des identischen Anzahlzählers [5n-4], der die Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierten Datenblöcken [5n-4] bis [5n]) bezüglich den Kandidatendaten [5n-4] anzeigt, mit dem Zählwert des Maximalzählers verglichen.

[0304] Wenn es bestimmt wird, dass der Zählwert des identischen Anzahlzählers [5n-4], der den Kandidatendaten [5n-4] entspricht, größer als der Zählwert des Maximalzählers ist, wird der Datenblock [5n-4], der als die Kandidatendaten gespeichert ist, als der Datenblock bestimmt, dessen Anzahl von Blöcken, die darin einer Gruppe (Datenblöcke [5n-4] bis [5n]) identisch sind, der größte darin ist.

[0305] In Schritt S3205 wird deshalb der Zählwert des Maximalzählers derart aufwärts gezählt, dass der Zählwert des Maximalzählers mit dem Zählwert des identischen Anzahlzählers [5n-4] übereinstimmt. Nachfolgend werden in Schritten S3206 und S3207 die Kandidatendaten [5n-4] als die gültig gemachten Daten gespeichert, deren Anzahl von Blöcken, die darin in einer Gruppe (fünf kombinierten Datenblöcken) identisch sind, die größte darin ist, und wird der Majoritätsregelmerker gesetzt. Danach wird in Schritt S3208 "1" zu der Variablen "j" addiert.

[0306] In Schritt S3207 kann es bestimmt werden, dass der Zählwert des identischen Anzahlzählers [5n-4] gleich oder kleiner als der Zählwert des Maximalzählers ist. In diesem Fall hat der Maximalzähler bereits als ein Zähler gedient, der die größte Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierten Datenblöcken [5n-4] bis [5n]) darstellt.

[0307] Die CPU **111** addiert "1" zu der Variablen "j" in Schritt S3208 als Reaktion auf den Zeitpunkt, zu dem der Vorgang in Schritt S3211 oder S3212 ohne Ausführen des Satzes von Schritten S3205 bis 3207 ausgeführt wird.

[0308] Nach einem Ausführen des Vorgangs in Schritt S2308, wie es zuvor dargelegt worden ist, wird der Satz von aufeinanderfolgenden Schritten S3203 bis S3208, S3211 und S3212 erneut ausgeführt. Es ist anzumerken, dass bei diesem Ausführen der Vorgänge in Schritten S3203 bis S3208, S3211 und S3212, da "1" zu der Variablen "j" addiert wird, in Schritt S3208 bestimmt wird, ob der Zählwert des identischen Anzahlzählers [5n-3] die gültig gemachten Daten wird; dieser Zählwert des identischen Anzahlzählers [5n-3] stellt die Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierten Datenblöcken [5n-4] bis [5n]) bezüglich dem Referenzdatenblock [5n-3] dar.

[0309] Genauer gesagt wird zu jeder Zeit, zu der "1" zu dem Ziel eines Vergleichs des Zählwerts des Maximalzählers addiert wird, dieser aufeinanderfolgend in der Reihenfolge des Zählwerts identischer Anzahlzählers [5n-3], des Zählwerts des identischen Anzahlzählers [5n-2] und des Zählwerts des identischen Anzahlzählers [5n-1] verschoben. Diese Vorgänge lassen zu, dass die Kandidatendaten, die als die gültig gemachten Daten gespeichert worden sind, letztlich als Daten erzielt werden, die aus den fünf kombinierten Datenblöcken in Schritt S3207 extrahiert werden sollten.

[0310] Nach dem wiederholten Ausführen des Satzes von Vorgängern in Schritten S3203 bis 3208, S3211 und S3212 wird es letztlich in Schritt S3203 bestimmt, dass die Variable "j" nicht kleiner als "4" ist. Zu dieser Zeit ist die Bestimmung bezüglich dessen, ob es einen der Datenblöcke gibt, dessen Anzahl von Blöcken, die darin in einer Gruppe (fünf kombinierten Datenblöcken [5n-4] bis [5n]) identisch sind, geändert wurden. Daher wird der Majoritätsregelüberprüfungsvergleich zu dem Zeitpunkt beendet, zu dem es bestimmt wird, dass die Variable "j" nicht kleiner als "4" ist.

[0311] Wie es vorhergehend dargelegt worden ist, werden jedoch nach Schritt S3203 in Schritt S3700, wenn [1] zu der Variablen "n" addiert wird, die nächsten fünf kombinierten Gruppen der Majoritätsregelüberprüfung unterzogen, die in Fig. 18 dargestellt ist.

[0312] Es ist anzumerken, dass, wenn die fünf kombinierten Datenblöcke [5n-4] bis [5n] in einer Gruppe der Majoritätsregelüberprüfung unterzogen werden, die zuvor dargelegt worden ist, wie es in Fig. 19 als ein Beispiel dargestellt ist, in Schritt S3100 die Anzahl von identischen Blöcken in einer Gruppe bezüglich den Kandidatendaten [5n-1] und die von identischen Blöcken in der Gruppe bezüglich den Kandidatendaten [5n-4] zueinander gleich und die größte in der Gruppe sein können.

[0313] In einem derartigen Fall, in dem eine Mehrzahl von Blöcken, deren Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierten Datenblöcken) gleich zueinander und darin die größten sind, kann es schwierig sein, irgendeinen der Blöcke als die gültig gemachten Daten selektiv zu extrahieren.

[0314] In dem dritten Ausführungsbeispiel wird es

deshalb in Schritt S3204, wenn es bestimmt wird, dass der Zählwert des identischen Anzahlzählers [5n-2] gleich oder kleiner als der Zählwert des Maximalzählers ist, in dem nächsten Schritt S3211 bestimmt, ob der Zählwert des identischen Anzahlzählers [5n-2] gleich dem Zählwert des Maximalzählers ist.

[0315] Wenn es bestimmt wird, dass der Zählwert des identischen Anzahlzählers [5n-2] gleich dem Zählwert des Maximalzählers ist (die Bestimmung in Schritt S3211 ist JA) wird in dem nächsten Schritt S3212 die Majoritätsregelmerker zurückgesetzt, um die Datenextraktionsaufgabe in Schritt S134 zu erhalten. Der Rücksetzzustand des Majoritätsregelmerkers, der während der Majoritätsregelüberprüfung in Schritt S3200 gehalten wird, lässt zu, dass die Datenextraktionsaufgabe in Schritt S134 in Schritt S3300 zwangsweise beendet wird. Dies ermöglicht es, die Zuverlässigkeit eines Umprogrammierens für Daten zu verbessern.

[0316] Wie es zuvor beschrieben worden ist, kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß dem dritten Ausführungsbeispiel im Wesentlichen die gleichen Effekte wie die ersten und zweiten Effekte gemäß dem ersten Ausführungsbeispiel erzielen. Weiterhin kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß dem dritten Ausführungsbeispiel den hier im weiteren Verlauf beschriebenen dritten Effekt erzielen.

[0317] Genauer gesagt ist es als der dritte Effekt, wenn es bestimmt wird, dass es eine Mehrzahl von Blöcken gibt, deren Anzahl von identischen Blöcken in einer Gruppe (fünf kombinierten Datenblöcken) zueinander gleich und die größten darin sind, möglich, zwangsweise die Datenextraktionsaufgabe in Schritt S134 zu beenden. Dies lässt, dass der Sicherheitspegel, der dem Datenumprogrammieren zugehörig ist, stark aufrechterhalten wird.

[0318] In dem dritten Ausführungsbeispiel wird die Datenextraktionsaufgabe in Schritt S134 unter Verwendung der Majoritätsregelüberprüfung in Übereinstimmung mit dem Verfahren und Algorithmus ausgeführt, die in den Fig. 15 bis Fig. 18 dargestellt sind. Es ist anzumerken, dass als eine Ausgestaltung des dritten Ausführungsbeispiels jede Gruppe aus drei Datenblöcken besteht, so dass die Majoritätsregelüberprüfung an den drei Blöcken angewendet wird.

[0319] Bei dieser Ausgestaltung kann die Datenextraktionsaufgabe in Schritt S134 unter Verwendung der Majoritätsregelüberprüfung in Übereinstimmung mit dem Verfahren und Algorithmus, die in Fig. 21 dargestellt sind, an Stelle der Fig. 15 bis Fig. 18 ausgeführt werden.

[0320] Genauer gesagt ist bei der Ausgestaltung die Verwaltungszentrale **200** dazu ausgelegt, die Elemente der Umprogrammierungsdaten (die Umprogrammierungsfaktordaten und die Pseudodaten) zu dem in ein Fahrzeug eingebauten Steuersystem **100** in Übereinstimmung mit dem Verfahren und Algorithmus ähnlich dem zweiten Ausführungsbeispiel (siehe die Fig. 10A und Fig. 10B) zu senden.

[0321] Wie es in Fig. 20 dargestellt ist, teilt die Motorsteuereinheit **110** die Elemente der Umprogrammierungsfaktordaten und der Spiegeldaten in Blöcke [1] bis [3d] und speichert die Datenblöcke [1] bis [3d] in einem ersten Speicherbereich **114g** des Datenspeichers **114** als Datenblöcke DB1 bis DB3d (d ist eine Ganzzahl).

[0322] Genauer gesagt sind die Sätze von Gruppen G1 bis Gd aufeinanderfolgend in dem ersten Speicherbereich **114g** des Datenspeichers **114** gespeichert worden. Die Gruppe G1 besteht aus dem Paar DB1 und DB2 des Datenblocks [1] der Umprogrammierungsfaktordaten und dem Datenblock DB3 (Datenblock [1]) der Pseudodaten. Weiterhin besteht die Gruppe Gd aus dem Paar BD3d-2 und DB3d-1 des Datenblocks [d] der Umprogrammierungsfaktordaten und dem Datenblock DB3d (Datenblock [d]) der Pseudodaten.

[0323] Bei der Ausgestaltung führt die CPU **111** das Datenextrahieren in Schritt S134 bezüglich den Datenblöcken DB1 bis DB3d unter Verwendung der Majoritätsregelüberprüfung in Übereinstimmung mit dem Verfahren und Algorithmus aus, die in Fig. 21 dargestellt sind.

[0324] Es ist anzumerken, dass, wie es in Fig. 21 dargestellt ist, die Datenblöcke [1] bis [3d] (siehe Fig. 20) durch [3n-2 (n ist eine Ganzzahl)], [3n-1] oder [3n] von jedem Block ausgedrückt sind.

[0325] Wie es in Fig. 21 dargestellt ist, ordnet die CPU **111** in der Datenextraktionsverarbeitung der Variablen n in Schritt S4100 "1" zu.

[0326] Als Nächstes wiederholt die CPU **111** den folgenden Satz von Schritten S4200, S4300, S4400 und S4700, was zulässt, dass jede Gruppe, die aus den Datenblöcken [3n-2], [3n-1] und [3n] der Umprogrammierungsfaktordaten und der Pseudodaten besteht, der Majoritätsregelüberprüfung unterzogen wird, die zuvor dargelegt worden ist:
In Schritt S4200 bestimmt die CPU **111**, ob die drei Blöcke [3n-2] bis [3n] jeder Gruppe zueinander identisch sind.

[0327] In Schritt S4300 extrahiert die CPU **111** einen Datenblock, dessen Anzahl von Blöcken, die darin in den drei Datenblöcken identisch sind, darin der größte ist, auf der Grundlage des Ergebnisses eines Identitätsvergleichs in Schritt S4200.

[0328] In Schritt S4400 bestimmt die CPU **111**, ob alle der Datenblöcke [1] bis [3d] der Majoritätsregelüberprüfung unterzogen werden.

[0329] In Schritt S4700 addiert die CPU **111** [1] zu der Variablen [n].

[0330] Zum Beispiel wird in Schritt S4100 [1] zu der Variablen (n] addiert und wird danach in Schritt S4200 die Majoritätsregelüberprüfung für die Datenblöcke [1] bis [3], die den Datenblöcken DB1 bis DB3 in den Datenblöcken [1] bis [3d] entsprechen, durchgeführt.

[0331] Genauer gesagt bestimmt die CPU **111** in Schritt S4201 von Schritt S4200, ob der Datenblock DB1 identisch zu dem Datenblock DB2 ist. In Schritt S4201 bestimmt die CPU **111**, wenn es bestimmt wird, dass der Datenblock DB1 nicht identisch zu dem Datenblock DB2 ist (die Bestimmung in Schritt S4201 ist NEIN), ob der Datenblock DB1 identisch zu dem Datenblock DB3 ist, in Schritt S4202.

[0332] Wenn es bestimmt wird, dass der Datenblock DB1 nicht identisch zu dem Datenblock DB3 ist (die Bestimmung in Schritt S4202 ist NEIN), bestimmt die CPU **111**, ob der Datenblock DB2 identisch zu dem Datenblock DB3 ist, in Schritt S4203.

[0333] In der Ausgestaltung führt die CPU **111** die Majoritätsregelüberprüfung für die Datenblöcke [3n-2] bis [3n] jeder Gruppe durch. Aus diesem Grund kann die CPU **111** in Schritt S4200, wenn es bestimmt wird, dass einer der Datenblöcke identisch zu einem anderen der Datenblöcke ist, die gepaarten Datenblöcke (den einen und den anderen der Datenblöcke) als das Paar der Datenblöcke bestimmen, dessen Anzahl von Blöcken, die darin in den entsprechenden drei Datenblöcken identisch sind, die größte darin ist.

[0334] Die CPU **111** geht daher zu Schritt S3400, wenn es nach einem Erfassen der gepaarten identischen Datenblöcke in jeder Gruppe bestimmt wird, dass es einen Datenblock gibt, dessen Anzahl von identischen Blöcken in den entsprechenden drei Datenblöcken die größte darin ist.

[0335] In Schritt S4300 extrahiert die CPU **111** ein Datenelement aus den gepaarten identischen Datenelementen auf der Grundlage des Ergebnisses einer Majoritätsregelüberprüfung.

[0336] Genauer gesagt extrahiert die CPU **111** in Schritt S4301 von Schritt S4300 den Datenblock DB1 aus den drei Datenelementen DB1 bis DB3, wenn es bestimmt wird, dass der Datenblock DB1 identisch zu mindestens einem von anderen Datenblöcken DB2 und DB3 ist, in Schritt S4201 oder S4202.

[0337] Andererseits extrahiert die CPU **111** in Schritt S4302 von Schritt S4300 den Datenblock DB2 aus den drei Datenelementen DB1 bis DB3, wenn es bestimmt wird, dass der Datenblock DB2 identisch zu dem Datenblock DB3 ist, in Schritt S4203.

[0338] Nachdem die Majoritätsregelüberprüfung für die Blöcke [1] bis [3] gebildet worden ist, führt die CPU **111** die Schritte S4400 und S4700 aus, die erforderlich sind, um die nächste Gruppe der drei Blöcke [4] bis [6] auszuführen.

[0339] Nachdem die Aufgabe in Schritt S4700 beendet worden ist, führt die CPU **111** den Satz von Schritt S4200, S4300, S4400 und S4700 erneut aus. Da "1" zu der Variablen "n" in Schritt S2700 addiert wird, wird die Majoritätsregelüberprüfung an den drei Blöcken DB4 bis DB6 angewendet, die den Blöcken [4] bis [6] entsprechen.

[0340] Genauer gesagt wird bei jeder Addition der Variablen "n" zu "1" in Schritt S4700 der Satz von drei Blöcken, an welchen die Majoritätsregelüberprüfung angewendet wird, aufeinanderfolgend in der Reihenfolge von drei Blöcken [4] bis [6] zu den drei Blöcken [3d-2] bis [3d] in Schritt S4200 verschoben.

[0341] Als ein Ergebnis eines wiederholten Ausführens des Satzes von Schritt S4200, S4300, S4400 und S4700 führt die CPU **111** in Schritt S4400, wenn es bestimmt wird, dass alle der Datenblöcke [1] bis [3d] der Majoritätsregelüberprüfung unterzogen werden, den Satz von Schritt S4500 und S4600 aus.

[0342] Genauer gesagt liest die CPU **111** in Schritt S4500, wie es in Fig. 20 dargestellt ist, die Datenblöcke BD1, DB4,..., DB3d-2, die extrahiert worden sind, als die gültig gemachten Daten aus den Datenblöcken DB1 bis DB3d der Umprogrammierungsfaktordaten und der Pseudodaten aus. Als Nächstes kombiniert die CPU **111** aufeinanderfolgend die ausgelesenen Datenblöcke DB1, DB4,..., DB3d-2, um die Umprogrammierungsfaktordaten zu dekodieren, und speichert den kombinierten Satz der Datenblöcke DB1, DB4,..., DB3d-2 in einem zweiten Speicherbereich **144h** des Datenspeichers **114**.

[0343] Wenn die Umprogrammierungsfaktordaten als ein Ergebnis der Datenextraktionsaufgabe dekodiert werden, setzt die CPU **111** den Majoritätsregelmerker in zum Beispiel dem Datenspeicher **114** in Schritt S4600; dieser Majoritätsregelmerker stellt dar, dass die Majoritätsregelüberprüfung gebildet ist. Dies lässt zu, dass die CPU **111** auf der Grundlage des Zustands des Majoritätsregelmerkers bestimmt, ob die Umprogrammierungsfaktordaten dekodiert werden, und danach verlässt die CPU **111** die Datenextraktionsverarbeitung.

[0344] Als ein Ergebnis eines wiederholten Ausfüh-

rens des Satzes von Schritt S4200, S4300, S4400 und S4700 kann es bestimmt werden, dass in Schritt S4200 (Schritt S4203) alle der drei Datenblöcke von einer Gruppe nicht zueinander identisch sind. In diesem Fall bestimmt die CPU **111** wahrscheinlich, dass die Daten, die von der Verwaltungszentrale **200** gesendet werden, nicht autorisierte Daten und/oder illegal geänderte Daten sind, um dadurch die Datenextraktionsaufgabe auf ein derartiges Bestimmen zu verlassen, dass die Majoritätsregelüberprüfung nicht ohne Einsetzen des Majoritätsregelmerker gebildet ist.

[0345] In Schritt S135 (siehe **Fig.** 7) wird deshalb die Bestimmung, ob die Umprogrammierungsfaktordaten dekodiert sind, abhängig von dem Zustand des Majoritätsregelmerkers ausgeführt.

[0346] Wie es zuvor beschrieben worden ist, kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß der Ausgestaltung des dritten Ausführungsbeispiels im Wesentlichen die gleiche Effekt wie die ersten und zweiten Effekte gemäß dem ersten Ausführungsbeispiel erzielen.

[0347] Weiterhin kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß der Ausgestaltung des dritten Ausführungsbeispiels den hier im weiteren Verlauf beschriebenen vierten Effekt erzielen.

[0348] Genauer gesagt kann die CPU **111** als den vierten Effekt, wenn es bestimmt wird, dass einer der Datenblöcke identisch zu einem anderen der Datenblöcke ist, die gepaarten Datenblöcke (den einen und anderen der Datenblöcke) als das Paar der Datenblöcke bestimmen, dessen Anzahl von identischen Blöcken in den entsprechenden drei Datenblöcken die größte darin ist. Dies lässt zu, dass die Last der CPU **111** verringert wird.

Viertes Ausführungsbeispiel

[0349] Ein in ein Fahrzeug eingebautes Steuersystem gemäß einem vierten Ausführungsbeispiel der vorliegenden Erfindung wird hier im weiteren Verlauf beschrieben. Das in ein Fahrzeug eingebaute Steuersystem des vierten Ausführungsbeispiels weist im Wesentlichen den gleichen Aufbau wie den des in ein Fahrzeug eingebauten Steuersystems des dritten Ausführungsbeispiels auf. Aus diesem Grund sind ähnliche Bezugszeichen ähnlichen Teilen in den in ein Fahrzeug eingebauten Steuersystemen gemäß den dritten und vierten Ausführungsbeispielen zugewiesen, so dass Beschreibungen der Teile des in ein Fahrzeug eingebauten Steuersystems des vierten Ausführungsbeispiels weggelassen werden.

[0350] Im Wesentlichen ähnlich dem dritten Ausführungsbeispiel ist in dem vierten Ausführungsbeispiel das in ein Fahrzeug eingebaute Steuersystem **100** dazu ausgelegt:
eine Mehrzahl von Elementen der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) zu empfangen, zu welchen der gleiche Identifizierer hinzugefügt ist, welcher von der Verwaltungszentrale **200** gesendet worden ist;
die empfangenen Elemente der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) gegeneinander zu überprüfen; und
als gültig gemachte Daten mindestens eines der Elemente der Umprogrammierungsdaten zu extrahieren, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass mindestens eines der Elemente der Umprogrammierungsdaten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element erfüllt.

[0351] Es ist anzumerken, dass in dem vierten Ausführungsbeispiel die Verwaltungszentrale **200** dazu ausgelegt ist, die Elemente der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) zu dem in ein Fahrzeug eingebauten Steuersystem **100** gemäß dem Verfahren und Algorithmus ähnlich dem dritten Ausführungsbeispiel (siehe die Fig. 13A und Fig. 13B) zu senden.

[0352] Genauer gesagt liest in dem vierten Ausführungsbeispiel beim Senden der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten die Verwaltungszentrale **200** aufeinanderfolgend die Umprogrammierungsfaktordaten und alle der ersten und bis dritten Pseudodaten in der Reihenfolge von dem Block [1] zu dem Block [e] unter den folgenden ersten bis dritten Bedingungen:
Die erste Bedingung ist, dass fünf Datenblöcke in jedem der Datenblöcke [1] bis [e] kombiniert werden, um ausgelesen zu werden.

[0353] Die zweite Bedingung ist, dass jede kombinierte Gruppe von fünf Blöcken mindestens welche der ersten bis dritten Pseudodaten beinhaltet.

[0354] Die dritte Bedingung ist, dass die Anzahl von Datenblöcken der Umprogrammierungsfaktordaten eine Majorität der Gesamtzahl (fünf) der fünf identischen Blöcke in einer Gruppe aufweist.

[0355] Zum Beispiel wird, wie es in Fig. 22 dargestellt ist, nach einem dreifachen Auslesen von einem Block [1] aus den Umprogrammierungsfaktordaten ein Block [1] aus allen der ersten und zweiten Pseudodaten extrahiert. Das dreifache Auslesen von Block [1] der Umprogrammierungsfaktordaten wird mit den entsprechenden ausgelesenen Blöcken [1] der ersten und zweiten Pseudodaten kombiniert, um eine Gruppe G1 auszubilden.

[0356] Da die Gruppe G1 aus den fünf Blöcken [1] besteht, ist die erste Bedingung erfüllt, und da die

Gruppe G1 die Blöcke [1] der ersten und zweiten Pseudodaten beinhaltet, ist die zweite Bedingung erfüllt. Weiterhin weist die Anzahl (3) von Datenblöcken [1] der Umprogrammierungsfaktordaten eine Majorität der Gesamtzahl (fünf) der fünf identischen Blöcke in einer Gruppe auf (siehe Fig. 22).

[0357] Dies lässt zu, dass der Satz der Gruppe G1 zu einer Gruppe Ge erzeugt wird. Die Gruppe G1 besteht aus den fünf kombinierten Blöcken DB1 bis DB5 der Datenblöcke [1] bis [5] der Umprogrammierungsfaktordaten und den ersten bis vierten Pseudodaten. Die Gruppe Ge besteht aus den fünf kombinierten Blöcken DBSe-4 bis DB5e der Datenblöcke [e] der Umprogrammierungsfaktordaten und der ersten Pseudodaten, die zu erzeugen sind.

[0358] Die Verwaltungszentrale **200** kategorisiert Datenblöcke DB1, DB2, DB3,..., DB5e derart, dass eine Mehrzahl der Datenrahmen DF die Datenblöcke DB1, DB2, DB3,..., DB5e enthält. Als Nächstes sendet die Verwaltungszentrale **200** aufeinanderfolgend die Blöcke von Datenrahmen DF zu der in ein Fahrzeug eingebauten Steuereinheit **100** in einem Paket bzw. in einem Block.

[0359] Wenn die Elemente der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten von der Verwaltungszentrale **200** auf die zuvor beschriebene Sendeweise gesendet werden, speichert die Motorsteuereinheit **110**, wie es in Fig. 22 dargestellt ist, die Datenblöcke [1] bis [5e] in einem ersten Speicherbereich **114i** des Datenspeichers **114** als die Datenblöcke DB1 bis DB5e.

[0360] In der Datenextraktionsaufgabe bezüglich des ersten Speicherbereichs **114i** extrahiert die CPU **111** Elemente von Umprogrammierungsfaktordaten in einem Schritt S134 mit einer Gültigkeit auf die folgende Weise.

[0361] Genauer gesagt führt die CPU **111** in dem vierten Ausführungsbeispiel, wie es in Fig. 22 dargestellt ist, eine Majoritätsbelegungsüberprüfung von Elementen der Umprogrammierungsfaktordaten und denjenigen der ersten bis dritten Pseudodaten durch, um zu bestimmen, ob es eine Majorität von identischen Blöcken in einer Gruppe (fünf kombinierten Elementen) der Umprogrammierungsfaktordaten und der ersten bis dritten Pseudodaten jeder Gruppe gibt.

[0362] Als ein Ergebnis der Majoritätsbelegungsüberprüfung wird, wenn es bestimmt wird, dass es die Majorität von identischen Blöcken in einer Gruppe (fünf kombinierten Elementen) gibt, die Majoritätsbelegungsüberprüfung gebildet, anders ausgedrückt, die vorbestimmte Regel ist erfüllt. Dann extrahiert die CPU **111** als die gültig gemachten Daten mindestens eines der identischen Datenelemente (Blöcke) von jeder Gruppe aus den fünf Datenelementen von diesen. Danach kombiniert die CPU **111**, wie es in Fig. 22 dargestellt, die extrahierten Datenblöcke DB1, DB7,..., DB5e-4, um dadurch diese in einem zweiten Speicherbereich **114j** des Datenspeichers **114** zu speichern. Dies lässt zu, dass die Umprogrammierungsfaktordaten in dem zweiten Speicherbereich **114j** des Datenspeichers **114** dekodiert werden.

[0363] Fig. 23 stellt schematisch ein bestimmtes Beispiel der Datenextraktionsaufgabe in Schritt S134 gemäß dem vierten Ausführungsbeispiel dar.

[0364] Es ist anzumerken, dass, wie es in Fig. 23 dargestellt ist, die Datenblöcke [1] bis [5e] (siehe Fig. 22) durch [5n-4 (n ist eine Ganzzahl)], [5n-3], [5n-2], [5n-1] oder [5n] für jeden Block ausgedrückt sind.

[0365] Wie es in Fig. 23 dargestellt ist, ordnet die CPU **111** "1" der Variablen n in Schritt S5001 zu.

[0366] Als Nächstes wiederholt die CPU **111** den folgenden Satz von Schritten S5100, S5200, S5300 und S5600, was zulässt, dass jede Gruppe, die aus den fünf Datenblöcken [5n-4] bis [5n] der Umprogrammierungsfaktordaten und den ersten bis dritten Pseudodaten für jede Gruppe besteht, der Majoritätsbelegungsüberprüfung unterzogen wird, die vorhergehend dargelegt worden ist:
In Schritt S5100 führt die CPU **111** eine Majoritätsbelegungsüberprüfungsaufgabe durch, um:
die fünf Blöcke [5n-4] bis [5n] jeder Gruppe der Majoritätsbelegungsüberprüfung zu unterziehen, und
einen Majoritätsbelegungsmerker unter einer Bedingung zu setzen, dass die Majoritätsbelegungsüberprüfung gebildet wird.

[0367] In Schritt S5200 überprüft die CPU **111** den Zustand des Majoritätsbelegungsmerkers; und
In Schritt S5300 bestimmt die CPU **111**, ob alle der Datenblöcke [1] bis [5e] der Majoritätsbelegungsüberprüfung unterzogen werden.

[0368] In Schritt S5600 addiert die CPU **111** [1] zu der Variablen [n].

[0369] In Fig. 23 ist ein Schritt S5100 äquivalent zu Schritt S3100 gemäß dem dritten Ausführungsbeispiel. Genauer gesagt wird, wie es in Fig. 16A dargestellt ist, der Datenblock [5n-4] in den fünf Datenblöcken [5n-4] bis [5n] als eine Referenz gegen die verbleibenden Datenblöcke darin verwendet, und wird es bestimmt, ob jeder der verbleibenden Datenblöcke [5n-3] bis [5n] identisch zu dem Referenzdatenblock [5n-4] ist.

[0370] Nach dem Identitätsvergleich unter Verwendung des Datenblocks [5n-4] als eine Referenz wird, wie es in Fig. 16B dargestellt ist, der Datenblock

[5n-3] als eine Referenz gegenüber den verbleibenden Datenblöcken darin verwendet, und wird es bestimmt, ob alle der verbleibenden Datenblöcke [5n-4] und [5n-2] bis [5n] identisch zu dem Referenzdatenblock [5n-3] sind.

[0371] Nach dem Identitätsvergleich unter Verwendung des Datenblocks [5n-3] als eine Referenz wird, wie es in Fig. 16C dargestellt ist, der Datenblock [5n-2] als eine Referenz gegenüber den verbleibenden Datenblöcken darin verwendet, und wird es bestimmt, ob jeder der verbleibenden Datenblöcke [5n-4], [5n-3], [5n-1] und [5n] identisch zu dem Referenzdatenblock [5n-2] ist.

[0372] Nach dem Identitätsvergleich unter Verwendung des Datenblocks [5n-2] als eine Referenz wird, wie es in Fig. 16D dargestellt ist, der Datenblock [5n-1] als eine Referenz gegenüber den verbleibenden Datenblöcken darin verwendet, und wird es bestimmt, ob jeder der verbleibenden Datenblöcke [5n-4], [5n-3], [5n-2] und [5n] identisch zu dem Referenzdatenblock [5n-1] ist.

[0373] In dem vierten Ausführungsbeispiel wird es, wie es in Fig. 16A dargestellt ist, als ein Beispiel als ein Ergebnis eines Bestimmens, ob jeder der verbleibenden Datenblöcke [5n-3] bis [5n] identisch zu dem Referenzdatenblock [5n-4] ist, wenn die identische Anzahl "2" erzielt wird, bestimmt, dass es die Majorität von identischen Blöcken in den Datenblöcken [5n-4] bis [5n] gibt. Dann wird der Datenblock [5n-4] als die gültig gemachten Daten aus den fünf Datenblöcken [5n-4] bis [5n] extrahiert und wird deshalb der Majoritätsbelegungsmerker gesetzt; dieser Majoritätsbelegungsmerker stellt dar, dass die Majoritätsbelegungsüberprüfung gebildet ist.

[0374] In Schritt S5300 wird es unter der Bedingung, dass der Majoritätsbelegungsmerker gesetzt ist (siehe Schritt S5200) bestimmt, ob alle Blöcke [1] bis [5e] der Majoritätsbelegungsüberprüfung unterzogen werden. Wenn es bestimmt wird, dass alle Blöcke [1] bis [5e] nicht der Majoritätsbelegungsüberprüfung unterzogen werden, wird "1" zu der Variablen "n" durch die CPU **111** in Schritt S5600 addiert und wird danach der Schritt S5100 erneut ausgeführt, so dass die fünf Datenblöcke der nächsten Gruppe der Majoritätsbelegungsüberprüfung unterzogen werden.

[0375] Als ein Ergebnis von wiederholten Ausführungen des Satzes von Schritten S5100, S5200, S5300 und S5600 führt die CPU **111** in Schritt S5300, wenn es bestimmt wird, dass alle der Datenblöcke [1] bis [5e] bereits der Majoritätsbelegungsüberprüfung unterzogen worden sind, den Satz von Schritten S5400 und S5500 aus.

[0376] Genauer gesagt liest die CPU **111** in Schritt S5400, wie es in Fig. 22 dargestellt ist, die Datenblöcke DB1, DB7,..., DB5e-4, die extrahiert worden sind, als die gültig gemachten Daten aus den Datenblöcken DB1 bis DB5e der Umprogrammierungsfaktordaten und der Pseudodaten für jede Gruppe aus. Als Nächstes kombiniert die CPU **111** aufeinanderfolgend die ausgelesenen Datenblöcke DB1, DB7,..., DB5e-4, um die Umprogrammierungsfaktordaten zu dekodieren und den kombinierten Satz der Datenblöcke DB1, DB7,..., DB5e-4 in dem zweiten Speicherbereich **114f** des Datenspeichers **114** zu speichern.

[0377] Wenn die Umprogrammierungsfaktordaten als ein Ergebnis der Datenextraktionsaufgabe dekodiert werden, wird der Majoritätsbelegungsmerker in zum Beispiel dem Datenspeicher **114** gesetzt. Dies lässt zu, dass die CPU **111** auf der Grundlage des Zustands des Majoritätsbelegungsmerkers bestimmt, ob die Umprogrammierungsfaktorsdaten dekodiert werden, und danach verlässt die CPU **111** die Datenextraktionsverarbeitung.

[0378] Als ein Ergebnis eines wiederholten Ausführens des Satzes von Schritten S5100, S5200, S5300 und S5600 kann die CPU **111** wahrscheinlich in Schritt S5200, wenn der Majoritätsbelegungsmerker zurückgesetzt werden kann, bestimmen, dass die Daten, die von der Verwaltungszentrale **200** gesendet werden, nicht autorisierte Daten und/oder illegal geänderte Daten sind. Daher verlässt die CPU **111** die Datenextraktionsaufgabe, wenn sie bestimmt, dass der Majoritätsbelegungsmerker zurückgesetzt ist.

[0379] In Schritt S135 (siehe **Fig.** 7) wird deshalb die Bestimmung, ob die Umprogrammierungsfaktordaten dekodiert werden, abhängig von dem Zustand des Majoritätsbelegungsmerkers ausgeführt.

[0380] Fig. 24 ist ein Flussdiagramm, das schematisch ein Beispiel des Algorithmus der Majoritätsbelegungsüberprüfung in Schritt S5100 von Fig. 23 darstellt.

[0381] Wie es zuvor beschrieben worden ist, werden in der Majoritätsbelegungsüberprüfung, wie es in den Fig. 16A bis Fig. 16D dargestellt ist, die kombinierten fünf Datenblöcke [5n-4] des [5n] jeder Gruppe mit einem der Blöcke [5n-4] bis [5n] als eine Referenz, die zu ändern ist, verglichen. Wenn es bestimmt wird, dass es die Majorität von identischen Blöcken in den Datenblöcken [5n-4] bis [5n] gibt, wird der Vergleich beendet und nur der Majoritätsbelegungsüberprüfungsmerker gesetzt.

[0382] In dem vierten Ausführungsbeispiel werden, wie es in Fig. 24 dargestellt ist, Stellgrößen "j", "k" und eine Mehrzahl von identischen Anzahlzählern auf die hier im weiteren Verlauf beschriebene Weise manipuliert, um die Majoritätsbelegungsüberprüfung zu realisieren. Es ist anzumerken, dass die inneren

Zähler der CPU **111** als die Mehrzahl von identischen Anzahlzählern verwendet werden können, aber äußere Zähler als die Mehrzahl von identischen Anzahlzählern vorbereitet und verwendet werden können.

[0383] Wie es in Fig. 24 dargestellt ist, löscht die CPU **11** bei der Majoritätsbelegungsüberprüfung den Zählwert von jedem der identischen Anzahlzähler in Schritt S5101.

[0384] Als Nächstes setzt die CPU **111** den Majoritätsbelegungsmerker zurück, setzt die Variable "j" auf "0" und setzt die Variable "k" auf "1" in Schritten S5102 und S5103.

[0385] Danach wiederholt die CPU **111** den folgenden Satz von Schritten S5104 bis S5108. Dies lässt zu, dass die fünf Datenblöcke [5n-4] bis [5n] den Identitätsvergleich auf der Grundlage des Datenblocks [5n-4] als eine Referenz gegenüber den verbleibenden Datenblöcken darin unterzogen werden, um dadurch zu bestimmen, ob jeder der verbleibenden Datenblöcke [5n-3] bis [5n] identisch zu dem Referenzdatenblock [5n-4] ist:
In Schritt S5104 berechnet die CPU **111** die Summe der Variablen "j" und "k", um den Summenwert mit der Kombinationsanzahl "5" der fünf verglichenen Blöcke zu vergleichen.

[0386] In Schritt S5105 vergleicht die CPU **111** den Datenblock [5n-4+j] mit dem Datenblock [5n-4+j+k] unter der Bedingung, dass der Summenwert weniger als die Kombinationsanzahl "5" ist, als ein Ergebnis des Vergleichs in Schritt S5104.

[0387] In Schritt S5106 zählt die CPU **111** den Zählwert des identischen Anzahlzählers [5n-4+j], der die identische Anzahl (Vergleichsergebnis) des Datenblocks [5n-4+j] darstellt, unter der Bedingung, dass der Datenblock [5n-4+j] identisch zu dem Datenblock [5n-4+j+k] ist, als ein Ergebnis des Vergleichs in Schritt S5105.

[0388] In Schritt S5107 bestimmt die CPU **111**, ob der Zählwert des identischen Anzahlzählers [5n-4+j], der in Schritt S5106 aufwärts gezählt wird, ein Bilden der Majoritätsbelegungsüberprüfung darstellt.

[0389] In Schritt S5108 gibt die CPU **111** "1" zu der Variable "k".

[0390] Zum Beispiel wird es in Schritten S5101 bis S5103, wenn der Zählwert des identischen Zählers [5n-4+j] gelöscht worden ist und die Variablen "j" und "k" auf "0" bzw. "1" gesetzt worden ist, in Schritt S5104 bestimmt, dass der Summenwert der Variablen "j" und "k" kleiner als der Referenzwert "5" ist.

[0391] Als Nächstes wird in Schritt S5105 der Datenblock [5n-4], der als eine Referenz verwendet wird, mit dem Datenblock [5n-3] verglichen, um zu bestimmen, ob der Datenblock [5n-4] identisch zu dem Datenblock [5n-3] ist.

[0392] Wenn es bestimmt wird, dass der Datenblock [5n-4] identisch zu dem Datenblock [5n-3] ist, wird in Schritt S5106 der Zählwert des identischen Anzahlzählers [5n-4], der die identische Anzahl des Datenblocks [5n-4] darstellt, aufwärts gezählt (inkrementiert), so dass der Zählwert "1" des identischen Anzahlzählers [5n-4] erzielt wird.

[0393] In Schritt S5107 wird es bestimmt, ob der Zählwert "1" des identischen Anzahlzählers [5n-4] ein Bilden der Majoritätsbelegungsüberprüfung darstellt.

[0394] Genauer gesagt wird es, wie es in Fig. 16A dargestellt ist, wenn die identische Anzahl des Datenblocks [5n-4], d. h. der Zählwert des identischen Anzahlzählers [5n-4], gleich oder größer als "2" ist, bestimmt, dass die Majorität von Datenblöcken in fünf Datenblöcken zueinander identisch ist. Bezüglich des Datenblocks [5n-4] wird es deshalb, da der Zählwert "1" des identischen Anzahlzählers [5n-4] ist, bestimmt, dass die Majoritätsbelegungsüberprüfung nicht gebildet ist. Dies führt dazu, dass "1" zu der Variable "k" in Schritt S5108 addiert wird.

[0395] In Schritt S5105 kann es bestimmt werden, dass der Datenblock [5n-4] nicht identisch zu dem Datenblock [5n-3] ist. In diesem Fall geht die CPU **111** ohne ein Ausführen des Zählermanipulationsvorgangs und der gleichen in Schritten S5100 und S5107 zu Schritt S5108. Das heißt, die CPU **111** addiert "1" als Reaktion auf den Zeitpunkt, zu dem es bestimmt wird, dass der Datenblock [5n-4] nicht identisch zu dem Datenblock [5n-3] ist, zu der Variablen "k".

[0396] Nach einem Ausführen des Vorgangs in Schritt S5108, der zuvor dargelegt worden ist, wird der Satz von aufeinanderfolgenden Schritten S5104 bis S5108 erneut ausgeführt. Es ist anzumerken, dass bei diesem Ausführen der Vorgänge in Schritten S5104 bis S5108, da "1" in Schritt S5108 zu der Variablen "k" addiert wird, der Datenblock [5n-4], der als eine Referenz verwendet wird, mit dem Datenblock [5n-2] verglichen wird, um zu bestimmen, ob der Datenblock [5n-4] identisch zu dem Datenblock [5n-2] ist.

[0397] Genauer gesagt wird zu jeder Zeit, zu der "1" zu der Variablen "k" addiert wird, das Ziel eines Identitätsvergleichs des Referenzdatenblocks [5n-4] aufeinanderfolgend in der Reihenfolge des Datenblocks [5n-3], des Datenblocks [5n-2], des Datenblocks [5n-1] und des Datenblocks [5n] in Schritt S5104 verschoben. Der Zählwert des identischen Anzahlzählers [5n-4], der durch die Vorgänge in Schritten S5104 bis S5106 manipuliert wird, zeigt letztlich die

Anzahl von identischen Blöcken in einer Gruppe in dem Fall eines Verwendens des Datenblocks [5n-4] als die Referenz, wie zum Beispiel " 2" in Fig. 16A als ein Beispiel in Schritt S5106.

[0398] Als ein Ergebnis des wiederholten Ausführens der Vorgänge in Schritt S5104 bis S5108 geht der Summenwert der Variablen "j" und "k" in Schritt S5104 zu "5", so dass es bestimmt wird, dass der Summenwert gleich oder größer als der Referenzwert "5" ist. Dies bewirkt, dass das Ausführen der CPU zu Schritt S5111 geht.

[0399] Genauer gesagt ist in diesem Fall die Variable "k" "5", so dass der Identitätsvergleich zwischen den fünf kombinierten Datenblöcken [5n-4] bis [5n] unter Verwendung des Datenblocks [5n-4] als eine Referenz beendet worden ist. In Schritt S5111 wird deshalb, wie es in Fig. 16B dargestellt ist, als eine Aufgabe zum Verschieben des Datenblocks, der als eine Referenz gegenüber den verbleibenden Datenblöcken verwendet wird, von dem Datenblock [5n-4] zu dem Datenblock [5n-3] "1" zu der Variablen "j" addiert. Parallel zu der Additionsaufgabe wird die Variable "k" auf "1" gesetzt, anders ausgedrückt wird gelöscht, um den Identitätsvergleich zwischen den fünf kombinierten Datenblöcken [5n-4] bis [5n] unter Verwendung des Datenblocks [5n-3] als eine Referenz auszuführen.

[0400] Als Nächstes werden die Vorgänge in Schritten S5104 bis S5108 wiederholt erneut unter der Bedingung ausgeführt, dass die Variable "j" niedriger als der Referenzwert "4" ist, der durch Subtrahieren von "1" von dem vorhergehenden Referenzwert "5" erzielt wird. Dies lässt den Identitätsvergleich zwischen den fünf kombinierten Datenblöcken [5n-4] bis [5n] unter Vewendung des Datenblocks [5n-3] als eine Referenz zu.

[0401] Es ist anzumerken, dass in dem vierten Ausführungsbeispiel ähnlich dem dritten Ausführungsbeispiel der Identitätsvergleich zwischen ausgewählten zwei Blöcken in den Blöcken [5n-4] des [5n] ohne Verdopplung ausgeführt wird.

[0402] Nach dem aufeinanderfolgenden wiederholten Ausführen des Satzes von Vorgängen in Schritten S5104 bis S5108, die aufeinanderfolgend die Datenblöcke [5n-3], [5n-2] und [5n-1] als Referenzen verwenden, durch die Vorgänge in Schritt S5111 und S5112 wird es letztlich in Schritt S5112 bestimmt, dass die Variable "j" nicht kleiner als "4" ist. Dann wird der Identitätsvvergleich zwischen den fünf kombinierten Datenblöcken [5n-4] bis [5n] beendet, so dass der Identitätsvergleich in Schritt S5100 zu dem Zeitpunkt beendet ist, zu dem es bestimmt wird, dass die Variable "j" nicht weniger als "4" ist.

[0403] Wie es vorhergehend dargelegt worden ist, unterbricht die CPU **11** jedoch als ein Ergebnis eines wiederholten Ausführens der Vorgänge in Schritten S5104 bis S5108 und Schritten S5111 und S5112, wenn es bestimmt wird, dass der Zählwert des identischen Anzahlzählers [5n-4] "2" erreicht, so dass die Majorität von Datenblöcken in den fünf kombinierten Datenblöcken zueinander identisch sind, ein Ausführen der Reihe von Vorgängen in Schritten S5104 bis S5108 und Schritten S5111 und S5112.

[0404] Nach dem Unterbrechen speichert die CPU **111** den Datenblock [5n-4] in zum Beispiel dem Datenspeicher **114** als die gültig gemachten Daten, die die fünf kombinierten Datenblöcke belegen, in einer Gruppe in Schritt S5121. Danach setzt die CPU **111** den Majoritätsbelegungsmerker, der ein Bilden der Majoritätsüberprüfungsbelegung anzeigt, worauf die Majoritätsbelegungsüberprüfung verlassen wird. Danach bestimmt die CPU **111** in Schritt S5200 (siehe Fig. 23), ob die Datenextraktionsaufgabe in Schritt S134 zu beenden ist.

[0405] Wie es zuvor beschrieben worden ist, wird, wenn "1" zu der Variablen "n" durch die CPU **111** in Schritt S5600 addiert wird, der Schritt S5100 (siehe Fig. 24) erneut ausgeführt, so dass die fünf Datenblöcke der letzten Gruppe der Majoritätsbelegungsüberprüfung unterzogen werden.

[0406] Wie es zuvor beschrieben worden ist, kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß dem vierten Ausführungsbeispiel im Wesentlichen die gleichen Effekte wie die ersten und zweiten Effekte gemäß dem ersten Ausführungsbeispiel erzielen.

[0407] Weiterhin kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß dem vierten Ausführungsbeispiel den hier im weiteren Verlauf beschriebenen fünften Effekt erzielen.

[0408] Genauer gesagt ist es als der fünfte Effekt, wenn es bestimmt wird, dass die Majorität von identischen Blöcken in den fünf kombinierten Blöcken in einer Gruppe zueinander identisch sind, möglich, einen der identischen Blöcke als gültig gemachte Daten zu speichern, was die Last der CPU **11** verringert.

[0409] In dem vierten Ausführungsbeispiel wird die Datenextraktionsaufgabe in Schritt S134 unter Verwendung der Majoritätsbelegungsüberprüfung in Übereinstimmung mit den Verfahren und Algorithmen ausgeführt, die in den Fig. 23 und Fig. 24 gezeigt sind. Es ist anzumerken, dass als eine Ausgestaltung des vierten Ausführungsbeispiels jede Gruppe aus drei Datenblöcken besteht, so dass die Majoritätsbelegungsüberprüfung an den drei Blöcken ihrer Gruppe angewendet wird.

[0410] Bei dieser Ausgestaltung kann die Datenex-

traktionsaufgabe in Schritt S134 unter Verwendung der Majoritätsbelegungsüberprüfung in Übereinstimmung mit den Verfahren und Algorithmus, die in Fig. 26 dargestellt sind, an Stelle der Fig. 23 und Fig. 24 ausgeführt werden.

[0411] Genauer gesagt ist bei der Ausgestaltung die Verwaltungszentrale **200** dazu ausgelegt, die Elemente der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) in Übereinstimmung mit den Verfahren und Algorithmus ähnlich dem zweiten Ausführungsbeispiel (siehe die Fig. 10A und Fig. 10B) zu dem in ein Fahrzeug eingebauten Steuersystem **100** zu senden.

[0412] Wie es in Fig. 25 dargestellt ist, teilt die Motorsteuereinheit **110** die Elemente der Umprogrammierungsfaktordaten und der Spiegeldaten in Blöcke [1] bis [3f] und speichert die Datenblöcke [1] bis [3f] in einem ersten Speicherbereich **114k** des Datenspeichers **114** als Datenblöcke DB1 bis DB3f (f ist eine Ganzzahl).

[0413] Genauer gesagt ist der Satz von Gruppen G1 bis Gf aufeinanderfolgend in dem ersten Speicherbereich **114g** des Datenspeichers **114** aufeinanderfolgend gespeichert worden. Die Gruppe G1 besteht aus dem Paar DB1 und DB2 des Datenblocks [1] der Umprogrammierungsfaktordaten und dem Datenblock DB3 (Datenblock [1]) der Pseudodaten. Weiterhin besteht die Gruppe Gf aus dem Paar DB3f-2 und DB3f-1 des Datenblocks (f) der Umprogrammierungsfaktordaten und des Datenblocks DB3f (Datenblock [f]) der Pseudodaten.

[0414] Bei der Ausgestaltung führt die CPU **111** die Datenextraktion in Schritt S134 bezüglich den Datenblöcken DB1 bis DB3f unter Verwendung der Majoritätsbelegungsüberprüfung in Übereinstimmung mit dem Verfahren in der Algorithmusaufgabe aus, die in Fig. 26 dargestellt sind.

[0415] Wie es in Fig. 26 dargestellt ist, ordnet die CPU **111** in der Datenextraktionsverarbeitung der Variablen n "1" in Schritt S6100 zu.

[0416] Als Nächstes wiederholt die CPU **111** den folgenden Satz von Schritten S6200, S6300, S6400 und S6700, was zulässt, dass jede Gruppe, die aus den Datenblöcken [3n-2], [3n-1] und [3n] der Umprogrammierungsfaktordaten und den Pseudodaten besteht, der Majoritätsbelegungsüberprüfung unterzogen wird, wie es vorhergehend dargelegt worden ist:
In Schritt S6200 führt die CPU **111** eine Majoritätsbelegungsüberprüfungsaufgabe durch, um:
die drei Blöcke [3n-2] bis [3n] jeder Gruppe der Majoritätsbelegungsüberprüfung zu unterziehen, und
einen Majoritätsbelegungsmerker unter der Bedingung zu setzen, dass die Majoritätsbelegungsüberprüfung gebildet ist.

[0417] In Schritt S6300 extrahiert die CPU **111** einen Datenblock, der die Majorität von den identischen Blöcken in den drei Datenblöcken belegt, auf der Grundlage des Ergebnisses einer Majoritätsbelegungsüberprüfung in Schritt S6200.

[0418] In Schritt S6400 bestimmt die CPU **111**, ob alle Datenblöcke [1] bis [3f] der Majoritätsbelegungsüberprüfung unterzogen werden.

[0419] In Schritt S6700 addiert die CPU **111** [1] zu der Variablen [n].

[0420] Zum Beispiel wird in Schritt S6100 [1] zu der Variablen [n] addiert und wird danach in Schritt S6200 die Majoritätsbelegungsüberprüfung für die Datenblöcke [1] bis [3], die den Datenblöcken DB1 bis DB3 in den Datenblöcken [1] bis [3b] entsprechen, durchgeführt.

[0421] Genauer gesagt bestimmt die CPU **111** in Schritt S6201 von Schritt S6200, ob der Datenblock DB1 identisch zu dem Datenblock DB2 ist. In Schritt S6201 bestimmt die CPU **111**, wenn es bestimmt wird, dass der Datenblock DB1 nicht identisch zu dem Datenblock DB2 ist (die Bestimmung in Schritt S6201 ist NEIN), ob der Datenblock DB1 identisch zu dem Datenblock DB3 ist, in Schritt S6202. Wenn es bestimmt wird, dass der Datenblock DB1 nicht identisch zu dem Datenblock DB3 ist (die Bestimmung in Schritt S6202 ist NEIN), bestimmt die CPU **111**, ob der Datenblock DB2 identisch zu dem Datenblock DB3 ist, in Schritt S6203.

[0422] Bei der Ausgestaltung führt die CPU **111** die Majoritätsbelegungsüberprüfung für die Datenblöcke [3n-2] bis [3n] in jeder Gruppe durch. Aus diesem Grund kann die CPU in Schritt 6200, wenn es bestimmt wird, dass einer der Datenblöcke identisch zu einem anderen der Datenblöcke ist, die gepaarten Datenblöcke (den einen und anderen der Datenblöcke) als das Paar der Datenblöcke bestimmen, die die Majorität von identischen Blöcken in den entsprechenden drei Datenblöcken belegen.

[0423] Die CPU **111** geht daher zu Schritt S6300, wenn es bestimmt wird, dass es einen Datenblock gibt, der die Majorität von identischen Blöcken in den entsprechenden drei Datenblöcken belegt, nach einem Erfassen der gepaarten identischen Datenblöcke in jeder Gruppe.

[0424] In Schritt S6300 extrahiert die CPU **111** ein Datenelement aus den gepaarten identischen Datenelementen auf der Grundlage des Ergebnisses der Majoritätsbelegungsüberprüfung.

[0425] Genauer gesagt extrahiert die CPU **111** in Schritt S6301 von Schritt S6300 den Datenblock DB1 aus den drei Datenelementen DB1 bis DB3, wenn es

bestimmt in Schritt S6201 oder S6202 wird, dass der Datenblock DB1 identisch zu mindestens einem von anderen Datenblöcken DB2 und DB3 ist.

[0426] Andererseits extrahiert die CPU **111** in Schritt S6302 von Schritt S6300 den Datenblock DB2 aus den drei Datenelementen DB1 bis DB3, wenn es in Schritt S6203 bestimmt wird, dass der Datenblock DB2 identisch zu dem Datenblock DB3 ist.

[0427] Nachdem die Majoritätsbelegungsüberprüfung für die Blöcke [1] bis [3] gebildet worden ist, führt die CPU **111** die Schritte S6400 und S6700 aus, die erforderlich sind, um die nächste Gruppe der drei Blöcke [4] bis [6] auszuführen.

[0428] Nachdem die Aufgabe in Schritt S6700 beendet worden ist, führt die CPU den Satz von Schritten S6200, S6300, S6400 und S6700 erneut aus. Da "1" zu der Variablen "n" in Schritt S6700 addiert wird, wird die Majoritätsbelegungsüberprüfung in den drei Blöcken DB4 bis DB6 angewendet, die den Blöcken [4] bis [6] entsprechen.

[0429] Genauer gesagt wird bei jeder Addition der Variable "n" zu "1" in Schritt S6700 der Satz von drei Blöcken, an welchen die Majoritätsbelegungsüberprüfung angewendet wird, aufeinanderfolgend in der Reihenfolge von den drei Blöcken [4] bis [6] zu den drei Blöcken [3f-2] bis [3f] in Schritt S6200 geändert.

[0430] Als ein Ergebnis von wiederholten Ausführungen des Satzes von Schritten S6200, S6300, S6400 und S6700 führt die CPU **111** in dem Schritt S6400, wenn es bestimmt wird, dass alle der Datenblöcke [1] bis [3f] der Majoritätsbelegungsüberprüfung unterzogen werden, den Satz von Schritten S6500 und S6600 aus.

[0431] Genauer gesagt liest die CPU **111** in Schritt S6500, wie es in Fig. 25 dargestellt ist, die Datenblöcke DB1, DB4,..., DB3f-2, die extrahiert worden sind, als die gültig gemachten Daten aus den Datenblöcken DB1 bis DB3f der Umprogrammierungsfaktordaten und der Pseudodaten aus. Als Nächstes kombiniert die CPU **111** aufeinanderfolgend die ausgelesenen Datenblöcke DB1, DB4,..., DB3f-2, um die Umprogrammierungsfaktordaten zu dekodieren und den kombinierten Satz der Datenblöcke DB1, DB4,..., DB3f-2 in einem zweiten Speicherbereich **114m** des Datenspeichers **114** zu speichern.

[0432] Wenn die Umprogrammierungsfaktordaten als ein Ergebnis der Datenextraktionsaufgabe dekodiert werden, setzt die CPU **111** den Datenbelegungsmerker in zum Beisp dem Datenspeicher **114** in Schritt S6600. Dieser Majoritätsbelegungsmerker stellt dar, dass die Majoritätsbelegungsüberprüfung gebildet ist. Dies lässt zu, dass die CPU **111** auf der Grundlage des Zustands des Majoritätsbelegungsmerkers bestimmt, ob die Umprogrammierungsfaktordaten dekodiert werden, und danach verlässt die CPU **111** die Datenextraktionsverarbeitung.

[0433] Als ein Ergebnis eines wiederholten Ausführens des Satzes von Schritten S6200, S6300, S6400 und S6700, kann es bestimmt werden, dass in Schritt S6200 (Schritt S6203) alle der drei Blöcke von einer Gruppe nicht zueinander identisch sind. In diesem Fall bestimmt die CPU **111** wahrscheinlich, dass die Daten, die von der Verwaltungszentrale **200** gesendet werden, nicht aktualisierte Daten und/oder illegal abgeänderte Daten sind, um dadurch die Datenextraktionsaufgabe nach einem derartigen Bestimmen zu verlassen, dass die Majoritätsbelegungsüberprüfung nicht ohne ein Setzen des Majoritätsbelegungsmerkers gebildet worden ist.

[0434] In Schritt S135 (siehe **Fig.** 7) wird deshalb die Bestimmung, ob die Umprogrammierungsfaktordaten dekodiert werden, abhängig von dem Zustand des Majoritätsbelegungsmerkers ausgeführt.

[0435] Wie es zuvor beschrieben worden ist, kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß der Ausgestaltung des vierten Ausführungsbeispiels im Wesentlichen die gleichen Effekte wie die ersten und zweiten Effekte gemäß dem ersten Ausführungsbeispiel erzielen.

[0436] Weiterhin kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß der Ausgestaltung des vierten Ausführungsbeispiels den fünften Effekt erzielen, der hier im weiteren Verlauf beschrieben wird.

[0437] Genauer gesagt kann die CPU **111** als den fünften Effekt, wenn es bestimmt wird, dass einer der Datenblöcke identisch zu einem anderen der Datenblöcke ist, die gepaarten Datenblöcke (den ein oder anderen der Datenblöcke) als das Paar der Datenblöcke bestimmen, die die Majorität von identischen Datenblöcken in den drei kombinierten Blöcken belegen. Dies lässt zu, dass die Last der CPU **111** verringert wird.

Fünftes Ausführungsbeispiel

[0438] Ein in ein Fahrzeug eingebautes Steuersystem gemäß einem fünften Ausführungsbeispiel der vorliegenden Erfindung wird hier im weiteren Verlauf beschrieben. Das in eine Fahrzeug eingebaute Steuersystem des fünften Ausführungsbeispiels weist im Wesentlichen den gleichen Aufbau wie den des in ein Fahrzeug eingebauten Steuersystems des dritten Ausführungsbeispiels auf. Aus diesem Grund sind ähnliche Bezugszeichen ähnlichen Teilen in den in ein Fahrzeug eingebauten Steuersystemen gemäß dem vierten und fünften Ausführungsbeispiel zugewiesen, so dass Beschreibungen der Teile des in ein

Fahrzeug eingebauten Steuersystems des fünften Ausführungsbeispiels weggelassen werden.

[0439] Im Wesentlichen ähnlich dem vierten Ausführungsbeispiel ist in dem fünften Ausführungsbeispiel das in ein Fahrzeug eingebaute Steuersystem **100** dazu ausgelegt:
eine Mehrzahl von Elementen der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) zu empfangen, zu welchen der gleiche Identifizierer hinzugefügt worden ist, welcher von der Verwaltungszentrale **200** gesendet worden ist;
die empfangenen Elemente der Umprogrammierungsdaten (der Umprogrammierungsfaktordaten und der Pseudodaten) gegeneinander zu überprüfen; und
mindestens eines der Elemente der Umprogrammierungsdaten als gültig gemachte Daten zu extrahieren, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass mindestens eines der Elemente der Umprogrammierungsdaten irgendeine von ersten bis vierten vorbestimmten Regeln gegenüber mindestens einem verbleibenden Element erfüllt.

[0440] Es ist anzumerken, dass in dem fünften Ausführungsbeispiel das in ein Fahrzeug eingebaute Steuersystem **100** die folgenden ersten bis vierten Bestimmungseinheiten beinhaltet, die im Detail in dem ersten bis vierten Ausführungsbeispiel beschrieben sind:
Die erste Bestimmungseinheit (Spiegelbestimmungseinheit) ist dazu ausgelegt, zu bestimmen, wenn die Mehrzahl von Elementen von Umprogrammierungsdaten, die aus bestimmten Daten und den Spiegeldaten von diesen bestehen, die das Komplementär der bestimmten Daten sind, empfangen worden ist, dass die erste vorbestimmte Regel erfüllt ist, wenn die empfangenen bestimmten Daten und die Spiegeldaten das Komplementär zueinander sind (das erste Ausführungsbeispiel).

[0441] Die zweite Bestimmungseinheit (Identitätsbestimmungseinheit) ist dazu ausgelegt, zu bestimmen, wenn die Mehrzahl von Elementen von Umprogrammierungsdaten, von denen jedes Element aus mindestens drei oder mehr Datenblöcken besteht, empfangen worden ist, dass die zweite vorbestimmte Regel gebildet ist, wenn mindestens zwei der drei oder mehr Datenblöcke von jedem Element der Umprogrammierungsdaten zueinander identisch sind (das zweite Ausführungsbeispiel).

[0442] Die dritte Bestimmungseinheit (Majoritätsregelbestimmungseinheit) ist dazu ausgelegt, zu bestimmen, wenn die Mehrzahl von Elementen von Umprogrammierungsdaten, wobei jedes Element aus drei oder mehr Datenblöcken besteht, empfangen worden ist, dass die dritte vorbestimmte Regel gebildet ist, wenn es mindestens einen Datenblock gibt, dessen Anzahl von Blöcken, die in den drei oder mehr Datenblöcken identisch sind, die größte ist (das dritte Ausführungsbeispiel).

[0443] Die vierte Bestimmungseinheit (Majoritätsbelegungsbestimmungseinheit) ist dazu ausgelegt, zu bestimmen, wenn die Mehrzahl von Elementen von Umprogrammierungsdaten, von denen jedes Element aus drei oder mehr Datenblöcken besteht, empfangen worden ist, dass die vierte vorbestimmte Regel gebildet ist, wenn die Majorität in drei oder mehr Datenblöcke zueinander identisch ist (das vierte Ausführungsbeispiel).

[0444] Beim Extrahieren von mindestens einem der Elemente der Umprogrammierungsdaten ist das in ein Fahrzeug eingebaute Steuersystem **100** dazu ausgelegt, dynamisch eine Auswahl von mindestens einer der ersten bis vierten Einheiten im Zusammenwirken mit der Verwaltungszentrale **200** zu bestimmen. Genauer gesagt ist das in ein Fahrzeug eingebaute Steuersystem **100** dazu ausgelegt, dynamisch eine Auswahl von irgendeiner der ersten bis vierten Bestimmungseinheiten auf der Grundlage einer Anweisung zu bestimmen, die von der Verwaltungszentrale **200** gesendet wird.

[0445] Gemäß dem Aufbau des in ein Fahrzeug eingebauten Steuersystems **100** ist es auch dann, wenn eine der ersten bis vierten vorbestimmten Regeln allgemein bekannt ist, möglich, den Sicherheitspegel aufrecht zu erhalten, der der Datenumprogrammierung zugehörig ist. Weiterhin ist das in ein Fahrzeug eingebaute Steuersystem **100** dazu ausgelegt, dynamisch eine Auswahl von irgendeiner der ersten bis vierten Bestimmungseinheiten im Zusammenwirken mit der Verwaltungszentrale **200** zu bestimmen, was es schwierig macht, zu dekodieren, wie die Umprogrammierungsfaktordaten zu dekodieren sind. Ein Umprogrammieren auf der Grundlage von nicht aktualisierten Daten und/oder illegal geänderten Daten kann deshalb sicher vermieden werden.

[0446] Fig. 27 ist ein Flussdiagramm, das schematisch die Auswahlverarbeitung zeigt, die durch zum Beispiel die Motorsteuereinheit **110** im Zusammenwirken mit der Verwaltungszentrale **200** auszuführen ist. In dem fünften Ausführungsbeispiel können die Kommunikationen zwischen der Verwaltungszentrale **200** und der Motorsteuereinheit **110** über die Hauptsteuereinheit **140** gebildet werden. Weiterhin kann die Auswahlverarbeitung jede vorbestimmte Periode wiederholt werden.

[0447] Wenn von der Verwaltungszentrale **200** auf der Grundlage von irgendeiner der ersten bis vierten Bestimmungseinheiten bestimmt wird, wie die Umprogrammierungsfaktordaten zu extrahieren sind, wird die Auswahlinformation, die irgendeine der ers-

ten bis vierten Bestimmungseinheiten anzeigt, die von der Motorsteuereinheit **110** ausgewählt werden sollte, von der Verwaltungszentrale **200** gesendet. Zum Beispiel kann als die Auswahlinformation ein Identifizierer verwendet werden. Der Identifizierer kann irgendeine der ersten bis vierten Bestimmungseinheiten identifizieren.

[0448] Beim Auswählen irgendeiner der ersten bis vierten Bestimmungseinheiten wartet die CPU **111** der Motorsteuereinheit **110**, bis sie Daten über den Kommunikationsbus **101** in Schritt S141 empfängt.

[0449] Wenn Daten empfangen werden, ist die Bestimmung in Schritt S141 JA, so dass die CPU **111** in Schritt S142 auf der Grundlage des Identifizierers, der in den empfangenen Daten enthalten ist, identifiziert, von welchem Typ die empfangenen Daten sind.

[0450] Wenn es identifiziert wird, dass die empfangenen Daten die Auswahlinformation sind (die Bestimmung in Schritt S142 ist JA), wählt die CPU **111** irgendeine der ersten bis vierten Bestimmungseinheiten auf der Grundlage der empfangenen Auswahlinformation (Identifizierer) aus.

[0451] In der Datenextraktionsaufgabe in Schritt S134 ist es deshalb möglich, die empfangenen Elemente von Umprogrammierungsdaten unter Verwendung der ausgewählten der ersten bis vierten Bestimmungseinheiten zu überprüfen, um dadurch als gültig gemachte Daten mindestens eines der Elemente der Umprogrammierungsdaten zu extrahieren, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass das mindestens eine der Elemente der Umprogrammierungsdaten eine entsprechende der ersten bis vierten Regeln gegenüber mindestens einem verbleibendem Element erfüllt.

[0452] Zum Beispiel können erste bis vierte Identifizierer, wie zum Beispiel "0 × 01 ", "0 × 02", "0 × 03" und "0 × 04", die jeweils den ersten bis vierten Bestimmungseinheiten entsprechen, als der Identifizierer verwendet werden. In diesem Fall kann jede der ECUs **110**, **120** und **130** mit einer Tabelle T versehen sein, die in Fig. 28 dargestellt ist. Die Tabelle T stellt die Beziehung zwischen den ersten bis vierten Identifizierern und den ersten bis vierten Bestimmungseinheiten dar.

[0453] Genauer gesagt ist es möglich, dass jede der ECUs **110**, **120** und **130** einfach irgendeine der ersten bis vierten Bestimmungseinheiten durch Beziehen auf die Tabelle T oder Verwendung des Identifizierers auswählt, der in den empfangenen Daten enthalten ist.

[0454] Wie es zuvor beschrieben worden ist, kann das in ein Fahrzeug eingebaute Steuersystem **100** gemäß dem fünften Ausführungsbeispiel im Wesentlichen die gleichen Effekte wie die ersten und zweiten Effekte gemäß dem ersten Ausführungsbeispiel, wie den dritten Effekt gemäß dem dritten Ausführungsbeispiel und wie den fünften Effekt gemäß dem vierten Ausführungsbeispiel erzielen.

[0455] Weiterhin kann das in eine Fahrzeug eingebaute Steuersystem **100** gemäß dem fünften Ausführungsbeispiel den hier im weiteren Verlauf beschriebenen siebten Effekt erzielen.

[0456] Genauer gesagt ist es als der siebte Effekt auch dann, wenn eine der ersten bis vierten vorbestimmten Regeln im Allgemeinen bekannt ist, möglich, den Sicherheitspegel aufrecht zu erhalten, der der Datenumprogrammierung zugehörig ist. Weiterhin machen es das in ein Fahrzeug eingebaute Steuersystem **100** und die Verwaltungszentrale **200** schwierig, zu dekodieren, wie die Umprogrammierungsfaktordaten zu extrahieren sind. Ein Umprogrammieren auf der Grundlage von nicht aktualisierten Daten und/oder illegal geänderten Daten kann deshalb sicher vermieden werden.

[0457] In dem fünften Ausführungsbeispiel lässt ein Verwenden einer Bestimmungseinheit, die in der Ausgestaltung des dritten Ausführungsbeispiels beschrieben ist, als die Majoritätsregelbestimmungseinheit zu, dass der vierte Effekt erzielt wird. Ähnlich lässt ein Verwenden einer Bestimmungseinheit, die in der Ausgestaltung des vierten Ausführungsbeispiels beschrieben ist, als die Majoritätsbelegungsbestimmungseinheit zu, dass der sechste Effekt erzielt wird.

[0458] Die ersten bis vierten Ausführungsbeispiele können innerhalb des Umfangs der vorliegenden Erfindung geändert werden. Einige Änderungen werden hier im weiteren Verlauf als Beispiele beschrieben.

[0459] Die Datenextraktionsaufgabe zum Extrahieren mindestens eines der Elemente der Umprogrammierungsdaten der gültig gemachten Daten, wenn es auf der Grundlage des überprüften Ergebnisses bestimmt wird, dass das mindestens eine der Elemente der Umprogrammierungsdaten eine vorbestimmte Regel gegenüber einem verbleibenden Element erfüllt, kann parallel zu einem Empfangen der Elemente der Umprogrammierungsdaten ohne Speichern von diesen durchgeführt werden.

[0460] In dem fünften Ausführungsbeispiel sendet die Verwaltungszentrale **200** die Auswahlinformation, aber eine andere Einheit, wie zum Beispiel die Hauptsteuereinheit **140**, kann die Auswahlinformation senden. In diesem Fall kann die Verwaltungszentrale **200** die Auswahlinformation empfangen und auf der Grundlage von irgendeiner der ersten bis vierten Bestimmungseinheiten bestimmen, wie die Umpro-

grammierungsfaktordaten zu extrahieren sind.

[0461] In dem fünften Ausführungsbeispiel kann es möglich sein, irgendeine der ersten bis vierten Bestimmungseinheiten auf der Grundlage einer Information, die einen Zeitpunkt, wie zum Beispiel Datum und/oder Stunde, anzeigt, als die Auswahlinformation auszuwählen, in welcher die Elemente von Umprogrammierungsdaten empfangen worden sind.

[0462] Zum Beispiel können die folgenden Reste, die durch Teilen durch "4" erzielt werden, das Datum, zu welchem die Elemente von Umprogrammierungsdaten empfangen worden sind, den ersten bis vierten Bestimmungseinheiten zugewiesen werden; diese "4" entspricht der Anzahl von Typen der Bestimmungseinheiten:

(a) der Rest "0", wie zum Beispiel das Datum von 4 oder 8, ist der ersten Bestimmungseinheit zugewiesen.

(b) der Rest "1", wie zum Beispiel das Datum von 1 oder 0 ist der zweiten Bestimmungseinheit zugewiesen.

(c) der Rest "2", wie zum Beispiel das Datum von 2 oder 6 ist der dritten Bestimmungseinheit zugewiesen.

(d) der Rest "3", wie zum Beispiel das Datum von 3 oder 7 ist der vierten Bestimmungseinheit zugewiesen.

[0463] In diesem Fall kann die Verwaltungszentrale **200** auf der Grundlage von irgendeiner der ersten bis vierten Bestimmungseinheiten in Übereinstimmung mit der Information, die ein Datum und/oder eine Stunde anzeigt, zu welchen die Elemente von Umprogrammierungsdaten empfangen worden sind, bestimmen, wie die Umprogrammierungsfaktordaten zu extrahieren sind.

[0464] In dem fünften Ausführungsbeispiel kann das in eine Fahrzeug eingebaute Steuersystem **100** mit mindestens zwei der ersten bis vierten Bestimmungseinheiten ausgestattet sein, um dadurch dynamisch irgendeine der mindestens zwei der ersten bis vierten Bestimmungseinheiten auszuwählen.

[0465] In jedem der ersten bis vierten Ausführungsbeispiele ist die Motorsteuereinheit **1110** dazu ausgelegt, die Datenextraktionsaufgabe zum Extrahieren von mindestens einem der Elemente der Umprogrammierungsdaten als gültig gemachte Daten auszuführen, wenn es bestimmt wird, dass mindestens eines der Elemente der Umprogrammierungsdaten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element erfüllt. Die vorliegende Erfindung ist nicht auf diese Aufgabe beschränkt.

[0466] Genauer gesagt kann die Hauptsteuereinheit **140** die Datenextraktionsaufgabe ausführen, bevor die Elemente der Umprogrammierungsdaten zu der Motorsteuereinheit **110** gesendet werden.

[0467] Die Datenextraktionsaufgabe, die von der Hauptsteuereinheit **140** auszuführen ist, wird hier im weiteren Verlauf beschrieben. Es ist anzumerken, dass die Datenextraktionsaufgabe jede vorbestimmte Periode wiederholt von der Hauptsteuereinheit **140** ausgeführt werden kann.

[0468] Wie es in Fig. 29 dargestellt ist, wartet die CPU **141** in Schritt S201, bis sie Elemente von Daten über den Kommunikationsbus **101** empfängt.

[0469] Wenn sie die Elemente von Umprogrammierungsdaten in einem Block über den Kommunikationsbus **201** empfängt, bestimmt die CPU **141**, ob der empfangene Identifizierer in jedem Element der Umprogrammierungsdaten und der bestimmte Codes, wie zum Beispiel ein VIN-Code, des entsprechenden Fahrzeugs übereinstimmen, in Schritt S202.

[0470] Wenn es bestimmt, dass der empfangene Identifizierer von jedem Element mit dem bestimmten Code des entsprechenden Fahrzeugs übereinstimmt, bestimmt die CPU **141** auf der Grundlage einer Empfangshistorie, die in dem Datenspeicher **144** gespeichert ist, ob die empfangenen Elemente der Umprogrammierungsdaten äquivalent zu Elementen sind, die bereits empfangen worden sind, um in dem Datenspeicher **144** gespeichert zu werden, in Schritt S203.

[0471] Wenn es bestimmt wird, dass die empfangenen Elemente der Umprogrammierungsdaten noch nicht empfangen worden sind (die Bestimmung in Schritt S203 ist NEIN), speichert die CPU **141** die empfangenen Elemente in dem Datenspeicher **144** als die Empfangshistorie in Schritt S204 in einem Block. Genauer gesagt speichert die CPU **141** die empfangenen Datenelemente in dem Datenspeicher **144** derart einem Block, dass die Datenelemente einander zugeordnet sind.

[0472] Dann führt die CPU **141** die Datenextraktionsaufgabe in Schritt S205 (siehe Schritt S134 in **Fig.** 7) aus, und, wenn die Umprogrammierungsfaktordaten von der Datenextraktionsaufgabe in Schritt S205 extrahiert werden, beendet die CPU **141** die Datenextraktionsverarbeitung, die in Fig. 29 dargestellt ist.

[0473] Wenn kein Element von Umprogrammierungsfaktordaten von der Datenextraktionsaufgabe in Schritt S205 extrahiert wird, liefert die CPU **141** eine Anforderung zum erneuten Senden von Elementen der Umprogrammierungsdaten in Schritt S207 zu der Verwaltungszentrale **200**.

[0474] Genauer gesagt ist die Hauptsteuereinheit **140** dazu ausgelegt, das extrahierte mindestens eine

Element der Umprogrammierungsdaten zu der Motorsteuereinheit **110** zu senden.

[0475] In diesem Fall kann die CPU **111** der Motorsteuereinheit **110** eine Datenumprogrammierungsaufgabe auf der Grundlage des extrahierten Elements, das von der Hauptsteuereinheit **140** gesendet wird, in Übereinstimmung mit dem Algorithmus ausführen, der als ein Beispiel in Fig. 30 dargestellt ist.

[0476] Genauer gesagt wartet die CPU **111** bis zu einem Empfangen von Daten in Schritt S231.

[0477] Wenn Daten empfangen werden, bestimmt die CPU **111**, ob die empfangenen Daten ein Element der Umprogrammierungsfaktordaten sind, auf der Grundlage des Identifizierers, der den empfangenen Daten hinzugefügt ist, in Schritt S232.

[0478] Wenn es bestimmt wird, dass die empfangenen Daten ein Element der Umprogrammierungsfaktordaten sind, speichert die CPU **111** die empfangenen Daten in dem Datenspeicher **114** in Schritt S233 und programmiert das entsprechende mindestens eine Datenelement, das in dem Programmspeicher **113** gespeichert ist, auf der Grundlage der empfangenen Daten in Schritt S234 um. Danach sendet die CPU **111** ein Datenumprogrammierungs-Beendigungssignal, das ein Beenden einer Datenumprogrammierung anzeigt, in Schritt S235 zu der Hauptsteuereinheit **140**.

[0479] Wie es zuvor beschrieben worden ist, kann die Hauptsteuereinheit **140** die Datenextraktionsaufgabe parallel zu einem Empfangen der Mehrzahl von Elementen von Umprogrammierungsdaten ausführen. In diesem Fall beseitigt die CPU **141** bevorzugt, wenn mindestens einer von Datenblöcken, der eine vorbestimmte Regel erfüllt, aus Datenblöcken in einer Gruppe extrahiert wird, ein Empfangen der verbleibenden Datenblöcke ausgenommen der Datenblöcke, die die vorbestimmte Regel erfüllen, und startet, Datenblöcke in der nächsten Gruppe zu überprüfen. Diese ermöglicht es, wirksam den Datenspeicher **144** zu verwenden. Eine derartige Datenextraktionsaufgabe, die von der Hauptsteuereinheit **140** auszuführen ist, wird hier im weiteren Verlauf beschrieben.

[0480] Es ist anzumerken, dass diese Ausgestaltung an dem zweiten bis fünften Ausführungsbeispiel angewendet werden kann, in welchem drei oder mehr Datenblöcke in einer Gruppe von der Verwaltungszentrale **200** gesendet werden. In diesem Fall wird die Ausgestaltung an der Identitätsbestimmungseinheit gemäß dem zweiten Ausführungsbeispiel als ein Beispiel angewendet. Die Datenextraktionsaufgabe, die von der Hauptsteuereinheit **140** gemäß der Ausgestaltung auszuführen ist, wird hier im weiteren Verlauf unter Bezugnahme auf die Fig. 31 und Fig. 32 beschrieben.

[0481] In der Verwaltungszentrale **200** werden, wie es in Fig. 31 dargestellt ist, die Umprogrammierungsfaktordaten und Pseudodaten in eine Mehrzahl von Datenblöcken [1] bis [g] geteilt, wobei "g" eine vorbestimmte Ganzzahl darstellt.

[0482] Eine Gruppe G1 besteht aus den fünf kombinierten Blöcken DB1 bis DB5 der Datenblöcke [1], die Pseudodaten beinhalten. Ähnlich besteht eine Gruppe G2 aus den fünf kombinierten Datenblöcken DB6 bis DB10 der Datenblöcke [2], die Pseudodaten beinhalten,..., besteht eine Gruppe Gg aus den fünf kombinierten Datenblöcken DB5g-4 bis 5g der Datenblöcke [g], die Pseudodaten beinhalten.

[0483] Die Verwaltungszentrale **200** kategorisiert die Datenblöcke DB1, DB2, DB3,..., DB5g derart, dass eine Mehrzahl der Datenrahmen DF die Datenblöcke DB1, DB2, DB3,..., DB5g enthalten. Als Nächstes sendet die Verwaltungszentrale **200** aufeinanderfolgend die Mehrzahl von Datenrahmen DF derart zu der in ein Fahrzeug eingebauten Steuereinheit **100** in einem Paket bzw. in einem Block, dass die Datenblöcke in jeder Gruppe miteinander gepaart sind. Die Verwaltungszentrale **200** sendet ebenso aufeinanderfolgend die Mehrzahl von Datenrahmen DF derart zu der in ein Fahrzeug eingebauten Steuereinheit **100**, dass ein Block der Umprogrammierungsfaktordaten, der in den fünf kombinierten Datenblöcken ihrer Gruppe enthalten ist, von den verbleibenden fünf Datenblöcken gesendet wird.

[0484] In der Hauptsteuereinheit **140** wartet die CPU **141**, wie es in Fig. 32 dargestellt ist, bis sie einen Datenblock in Schritt S6001 empfängt.

[0485] Wenn sie den Datenblock [1] empfängt, schreitet die CPU **141** zu Schritt S6002 fort und bestimmt, ob der Identifizierer, der dem Datenblock hinzugefügt ist, mit dem bestimmten Code, wie zum Beispiel dem VIN-Code, des entsprechenden Fahrzeugs übereinstimmt.

[0486] Wenn es bestimmt wird, dass der Identifizierer, der dem Datenblock hinzugefügt ist, mit dem bestimmten Code des entsprechenden Fahrzeugs übereinstimmt (die Bestimmung in Schritt S6002 ist JA), schreitet die CPU **141** zu Schritt S6003 fort.

[0487] In Schritt S6003 bestimmt die CPU **141**, ob die empfangene Anzahl "k" eines Empfangens des Datenblocks kleiner als "5" ist, was der Anzahl von gepaarten Blöcken in jeder Gruppe entspricht.

[0488] Es ist anzumerken, dass die Anzahl "k" auf ein Empfangen des Datenblocks [1] auf "0" gesetzt werden kann und deshalb die CPU **141** bestimmt, dass die Anzahl "k" eines Empfangen des Datenblocks kleiner als "5" ist. In diesem Fall führt die CPU **141** den Satz von Schritten S6004 bis S6008 aus, um

eine Identitätsvergleichsaufgabe zwischen den fünf Datenblöcken [1] bis [5] auszuführen:
In Schritt S6004 überprüft die CPU **141** den Zustand des Identitätsmerkers, der auf ein Bilden der Identitätsüberprüfung gesetzt wird.

[0489] In Schritt S6005 bestimmt die CPU **141**, ob Datenblöcke in dem Datenspeicher **144** gespeichert sind, unter der Bedingung, dass der Identitätsmerker zurückgesetzt ist.

[0490] In Schritt S6006 speichert die CPU **141** den empfangenen Datenblock in dem Datenspeicher **144** und der Bedingung, dass keine Datenblöcke in dem Datenspeicher **144** gespeichert sind.

[0491] In Schritt S6007 setzt die CPU **141** einen Speichermerker, der anzeigt, dass Datenblöcke in dem Datenspeicher **144** gespeichert sind, unter der Bedingung, dass keine Datenblöcke in dem Datenspeicher **144** gespeichert sind.

[0492] In Schritt S6008 addiert die CPU **141** "1" zu der empfangenen Anzahl "k".

[0493] Genauer gesagt wird der empfangene Datenblock [**1**] in dem Datenspeicher **114** gespeichert (siehe Schritt S6006), wird der Speichermerker gesetzt (siehe Schritt S6007) und wird die empfangene Anzahl "k" auf "1" manipuliert.

[0494] Wenn ein Datenblock [2] neu empfangen wird, bestimmt die CPU **141**, da der Speichermerker gesetzt ist, dass der Datenblock im Voraus in dem Datenspeicher **114** gespeichert ist.

[0495] Dann führt die CPU **141** als eine Identitätsüberprüfung zwischen dem Datenblock, der in dem Datenspeicher **144** gespeichert ist, und dem neu empfangenen Datenblock, die vorliegenden Schritte S6011 und S6012 aus:
In Schritt S6011 bestimmt die CPU **141**, ob der Datenblock, der in dem Datenspeicher **144** gespeichert ist, identisch zu dem neu empfangenen Datenblock ist.

[0496] In Schritt S6012 setzt die CPU **141** den Identitätsmerker unter der Bedingung, dass der Datenblock, der in dem Datenspeicher **144** gespeichert ist, identisch zu dem neu empfangenen Datenblock ist.

[0497] Genauer gesagt sind, wie es in Fig. 31 dargestellt ist, der Datenblock [1] und der neu empfangene Datenblock [2] zueinander identisch (siehe Schritt S6011), so dass der Identitätsmerker gesetzt wird (siehe Schritt S6012). Danach wird "1" zu der empfangenen Anzahl "k" zugefügt, so dass die empfangene Anzahl "k" zu "2" geht.

[0498] Wenn die Datenblöcke [3] bis [5] empfangen werden (siehe Schritt S600), wird der Zustand des Identitätsmerker überprüft, so dass das Empfangen der Datenblöcke [3] bis [5] ausgelöscht wird. Das heißt, die CPU **141** addiert weiterhin "1" zu der empfangenen Anzahl "k", wenn der Identitätsmerker überprüft wird (siehe Schritt S6008).

[0499] Im Gegensatz dazu wird es, wenn die Identitätsüberprüfung zwischen den zwei kombinierten Datenblöcken in einer Gruppe beendet ist und danach der Datenblock [6] empfangen wird (S6001), bestimmt, dass die empfangene Anzahl "k" nicht kleiner als "5" ist. In diesem Fall führt die CPU, um fünf kombinierte Datenblöcke der letzten Gruppe dem Identitätsvergleich zu unterziehen, die folgenden Schritte S6021 bis S6025 und S6031 aus.

[0500] In Schritt S6021 löscht die CPU **141** die empfangene Anzahl "k".

[0501] In Schritt S6022 überprüft die CPU **141** den Zustand des Identitätsmerkers.

[0502] In Schritt S6023 setzt die CPU **141** den Identitätsmerker unter der Bedingung zurück, dass der Identitätsmerker in Schritt S6022 gesetzt worden ist.

[0503] In Schritt S6024 speichert die CPU **141** den empfangenen Datenblock in dem Datenspeicher **144** unter der Bedingung, dass der Identitätsmerker in Schritt S6022 gesetzt worden ist.

[0504] In Schritt S6025 setzt die CPU **141** die empfangene Anzahl "k" auf "1" unter der Bedingung, dass der Identitätsmerker in Schritt S6022 gesetzt worden ist.

[0505] In Schritt S6031 liefert die CPU **141** eine Anforderung zum erneuten Senden der Umprogrammierungsdaten unter der Bedingung zu der Verwaltungszentrale **200**, dass der Identitätsmerker in Schritt S6022 zurückgesetzt worden ist.

[0506] Genauer gesagt wird der Identitätsmerker zurückgesetzt (siehe Schritt S6022), wird der Datenblock [6] in dem Datenspeicher **144** gespeichert (siehe Schritt S6024) und wird die empfangene Anzahl "k" auf "1" gesetzt (siehe Schritt S6025). Danach wird es, wenn der Datenblock [7] erneut empfangen wird (siehe Schritt S6001), in Schritt S6003 bestimmt, dass die empfangene Anzahl "k" kleiner als "5" ist. Danach werden die Schritte S6004 bis S6008, S6011 und S6012 erneut wiederholt, so dass die Identitätsüberprüfung zwischen den Datenblöcken [6] bis [10] durchgeführt wird, wie es in Fig. 31 dargestellt ist.

[0507] Ein wiederholtes Ausführen des Satzes von Schritten S6001 bis S6008, S6011 und S6012 und S6012 bis S6025 lässt ein Realisieren eines Beispiels der Datenextraktionsaufgabe zu, die in Fig. 31

dargestellt ist.

[0508] In den zweiten bis fünften Ausführungsbeispielen können drei oder mehr Elemente der Umprogrammierungsdaten als das Ziel für die Identitätsüberprüfung, die Majoritätsregelüberprüfung und die Majoritätsbelegungsüberprüfung verwendet werden.

[0509] In dem zweiten und fünften Ausführungsbeispiel kann es möglich sein, zu bestimmen, dass die Identitätsüberprüfung gebildet wird, wenn mindestens drei Blöcke in drei oder mehr Blöcken in einer Gruppe zueinander identisch sind.

[0510] Als ein Ergebnis der Majoritätsregelüberprüfung ist es, wenn es bestimmt wird, dass es eine Mehrzahl von Blöcken gibt, deren Anzahl von Blöcken, die darin in einer Gruppe identisch sind, die gleiche zueinander sind und die größte darin ist, möglich, die Datenextraktionsaufgabe in Übereinstimmung mit verschiedenen Verfahren ausgenommen für diejenigen, die in dem dritten Ausführungsbeispiel beschrieben sind, zu halten. Dies ermöglicht es, die Zuverlässigkeit eines Umprogrammierens für Daten zu verbessern.

[0511] Bei der Majoritätsregelüberprüfung und/oder Majoritätsbelegungsüberprüfung ist es möglich, eine Identitätsüberprüfung von bestimmten Datenblöcken erneut auszuführen, deren Identitätsüberprüfung bereits durchgeführt worden ist.

[0512] Die Verwaltungszentrale **200** kann die Pseudodaten beim Senden der Elemente der Umprogrammierungsdaten ohne Speichern von diesen erzeugen.

[0513] Wenn der Identitätsvergleich an Datenblöcken angewendet wird, ist es möglich, diesen in irgendeiner Datenlänge, wie zum Beispiel 2 Bytes, auszuführen.

[0514] Es ist möglich, die Datenblöcke in irgendeiner Einheit ausgenommen in Datenblöcken zu senden.

[0515] Es ist möglich, die Elemente von Umprogrammierungsdaten in irgendeiner Einheit ausgenommen in Datenblöcken zu überprüfen.

[0516] Das in ein Fahrzeug eingebaute Steuersystem gemäß jedem der ersten bis fünften Ausführungsbeispiele kann an einem System angewendet werden, das aus einer Mehrzahl von ECUs besteht, die dazwischen über eine besondere Leitung kommunizierbar verbunden sind.

[0517] Eine andere Verarbeitung ausgenommen der Spiegelüberprüfung, der Identitätsüberprüfung, der Majoritätsregelüberprüfung und der Majoritätsbelegungsüberprüfung kann verwendet werden, um die Datenextrahierungsaufgabe zu realisieren, die dazu ausgelegt ist, die Elemente der Umprogrammierungsdaten zu überprüfen und als gültig gemachte Daten mindestens eines der Elemente der Umprogrammierungsdaten auf der Grundlage des überprüften Ergebnisses zu extrahieren.

[0518] Ein Algorithmus zum Bewirken, dass die CPU eine andere Verarbeitung ausführt, kann frei bestimmt werden.

[0519] Die Verwaltungszentrale **200** kann alle Elemente von Umprogrammierungsdaten als gültig gemachte Daten senden.

[0520] In jedem der ersten bis fünften Ausführungsbeispiele wird die vorliegende Erfindung an einem in ein Fahrzeug eingebauten Steuersystem angewendet. Die vorliegende Erfindung ist jedoch nicht auf die Anwendung beschränkt. Genauer gesagt kann die vorliegende Erfindung an Umprogrammierungssystemen zum Umprogrammieren von Daten angewendet werden, die darin gespeichert sind. In jedem der ersten bis fünften Ausführungsbeispiele und ihren Ausgestaltungen können die Elemente, die in den Einheiten vorgesehen sind, als Mikrocomputer realisiert werden.

[0521] Einige der Elemente, die alle der Einheiten gemäß den ersten bis fünften Ausführungsbeispielen und ihren Ausgestaltungen bilden, können unabhängig von einem Mikrocomputer vorgesehen sein oder können als dedizierte Hardwarevorrichtung, wie zum Beispiel kundenspezifische LSI- (Large-Scale Integration)-Schaltungen realisiert sein.

[0522] Weiterhin werden Fachleute erkennen, dass die vorliegende Erfindung im Stande ist, als Programmprodukte, zum Beispiel die Programme, die in den Programmspeichern in einer Vielzahl von Formen gespeichert sind, verteilt zu werden. Es ist ebenso wichtig, anzumerken, dass die vorliegende Erfindung unberücksichtigt des besonderen Typs eines Signal tragenden Mediums gleichermaßen angewendet werden kann, um tatsächlich die Verteilung auszuführen. Beispiele von geeigneten Signal tragenden Medien beinhalten Medien eines aufzeichenbaren Typs, die zum Beispiel CD-ROMs und DVD-ROMs, und Medien eines Übertragungstyps, wie zum Beispiel digitale und analoge Kommunikationsverbindungen, beinhalten.

[0523] Obgleich beschrieben worden ist, was derzeit als die Ausführungsbeispiele und ihre Ausgestaltungen der vorliegenden Erfindung erachtet wird, versteht es sich, dass verschiedene Ausgestaltungen, welche noch nicht beschrieben worden sind, darin durchgeführt werden können, und es beabsichtigt ist, in den beiliegenden Ansprüchen alle derartigen Aus-

gestaltungen abzudecken, wie sie innerhalb des wahren Geists und Umfangs der Erfindung fallen.

Patentansprüche

1. Ein Datenumprogrammierungssystem, welches mit einem umschreibbaren Speicher kommunizieren kann, in welchem Daten im Voraus gespeichert sind, wobei die gespeicherten Daten auf der Grundlage von gültig gemachten Daten umprogrammierbar sind, wobei das Datenumprogrammierungssystem aufweist:
eine Empfangseinheit, die dazu ausgelegt ist, eine Mehrzahl von Elementen von Daten zur Verwendung beim Umprogrammieren zu empfangen, wobei die Mehrzahl von Elementen von Daten zu der Empfangseinheit gesendet werden, wobei der Mehrzahl von Elementen von Daten jeweils der gleiche Identifizierer hinzugefügt ist; und
eine Datenextraktionseinheit, die dazu ausgelegt ist, die empfangene Mehrzahl von Elementen von Daten zu überprüfen und mindestens ein Element der Daten als die gültig gemachten Daten auf ein Bestimmen auf der Grundlage des überprüften Ergebnisses zu extrahieren, dass das mindestens eine Element der Daten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element der Daten erfüllt.

2. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei das Datenumprogrammierungssystem in ein Fahrzeug eingebaut ist und die Daten, die in dem umschreibbaren Speicher gespeichert sind, ein Steuerprogramm und Steuerdaten beinhalten, die zum Steuern von mindestens einer Vorrichtung verwendet werden, die in das Fahrzeug eingebaut ist.

3. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, mindestens ein Element von nicht gültig gemachten Daten beinhaltet, wobei das mindestens eine Element von nicht gültig gemachten Daten unvereinbar mit dem mindestens einen Element ist, das von der Extraktionseinheit extrahiert wird.

4. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei
die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, mindestens ein Element von bestimmten Daten und mindestens ein Element von Spiegeldaten beinhaltet, wobei das mindestens eine Element von bestimmten Daten und das mindestens eine Element von Spiegeldaten zueinander komplementär sind, wobei das mindestens eine Element von bestimmten Daten und das mindestens eine Element von Spiegeldaten zu der Empfangseinheit gesendet werden, während sie miteinander gepaart sind, und die Datenextraktionseinheit dazu ausgelegt ist: das mindestens eine Element von bestimmten Daten und das mindestens
eine Element von Spiegeldaten zu überprüfen, um zu erfassen, dass das mindestens eine Element von bestimmten Daten komplementär zu dem mindestens einen Element von Spiegeldaten ist, um dadurch zu bestimmen, dass die mindestens einen bestimmten Daten die vorbestimmte Regel gegenüber dem mindestens einen Element der Spiegeldaten erfüllen; und
als die gültig gemachten Daten das mindestens eine Element der bestimmten Daten auf der Grundlage des Bestimmens zu extrahieren.

5. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei
die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen werden, drei oder mehr Elementen von Daten beinhaltet, wobei mindestens zwei der drei oder mehr Elemente von Daten zueinander identisch sind, wobei die Datenextraktionseinheit dazu ausgelegt ist:
die drei oder mehr Elemente von Daten zu überprüfen, um zu erfassen, dass mindestens zwei der drei oder mehr Elemente von Daten zueinander identisch sind, um dadurch zu bestimmen, dass eines der mindestens zwei identischen Elemente von Daten die vorbestimmte Regel gegenüber mindestens einem anderen der mindestens zwei identischen Elementen von Daten erfüllt; und
als die gültig gemachten Daten das eine der mindestens zwei identischen Elemente der Daten auf der Grundlage des Bestimmens extrahiert.

6. Ein Datenumprogrammierungssystem nach Anspruch 5, wobei die Datenextraktionseinheit dazu ausgelegt ist, die mindestens zwei der drei oder mehr Elemente von Daten auszulöschen, nachdem es bestimmt worden ist, dass die mindestens zwei der drei oder mehr Elemente zueinander identisch sind.

7. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei
die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen werden, drei oder mehr Elemente von Daten beinhaltet, wobei eine Majorität der drei oder mehr Elemente von Daten zueinander identisch sind, und die Datenextraktionseinheit dazu ausgelegt ist:
die drei oder mehr Elemente von Daten zu überprüfen, um zu erfassen, dass die Majorität der drei oder mehr Elemente von Daten zueinander identisch sind, um dadurch zu bestimmen, dass mindestens eines der Majorität der drei oder mehr Elemente von Daten die vorbestimmte Regel gegenüber einem anderen mindestens einem der Majorität der drei oder mehr Elemente von Daten erfüllt; und
als die gültig gemachten Daten das mindestens eine der Majorität der drei oder mehr Elemente von Daten auf der Grundlage des Bestimmens extrahiert.

8. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, drei oder mehr Elemente von Daten beinhaltet, wobei die drei oder mehr Elemente von Daten mindestens ein erstes Element beinhalten, dessen Anzahl von Elementen zu denen in den drei oder mehr Elementen von Daten identisch ist, wobei die Datenextraktionseinheit dazu ausgelegt ist:
die drei oder mehr Elemente von Daten zu überprüfen, um zu bestimmen, dass das mindestens eine erste Element von Daten, dessen Anzahl von Elementen zu denen in den drei oder mehr Elementen von Daten identisch ist, darin das größte ist, um dadurch zu bestimmen, dass das mindestens eine erste Element von Daten die vorbestimmte Regel gegenüber dem verbleibenden mindestens einem der Elemente erfüllt, das dazu identisch ist;
als die gültig gemachten Daten das mindestens eine erste Element der Daten auf der Grundlage des Bestimmens extrahiert.

9. Ein Datenumprogrammierungssystem nach Anspruch 8, wobei die Datenextraktionseinheit dazu ausgelegt ist, zu bestimmen, wenn es zusätzlich zu dem mindestens einen ersten Element mindestens ein zweites Element von Daten erfasst, dessen Anzahl von Elementen zu denen in den drei oder mehr Elementen von Daten das größte darin ist, dass das mindestens eine erste Element der drei oder mehr Elemente von Daten nicht die vorbestimmte Regel erfüllt.

10. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei die vorbestimmte Regel aus einer Mehrzahl von Regeln besteht, wobei die Datenextraktionseinheit aufweist:
eine Sendeeinheit, die dazu ausgelegt ist, die Mehrzahl von Elementen von Daten zu der Empfangseinheit zu senden;
eine Mehrzahl von Bestimmungseinheiten, die mit der Sendeeinheit kommunizieren können und dazu ausgelegt sind, auf der Grundlage des überprüften Ergebnisses zu bestimmen, ob das mindestens eine Element der Daten mindestens eine der Mehrzahl von Regeln gegenüber mindestens einem verbleibenden Element der Daten erfüllt; und
eine Auswahleinheit, die dazu ausgelegt ist, eine der Mehrzahl von Bestimmungseinheiten im Zusammenwirken mit der Sendeeinheit auszuwählen.

11. Ein Datenumprogrammierungssystem nach Anspruch 10, wobei die Sendeeinheit dazu ausgelegt ist, weiterhin zu der Datenextraktionseinheit eine Auswahlinformation zu senden, die irgendeine der Mehrzahl von Bestimmungseinheiten anzeigt, die ausgewählt werden sollte, und die Auswahleinheit dazu ausgelegt ist, die mindestens eine der Mehrzahl von Bestimmungseinheiten auf der Grundlage der Auswahlinformation auszuwählen, die von der Sendeeinheit gesendet wird.

12. Ein Datenumprogrammierungssystem nach Anspruch 10, wobei das Datenumprogrammierungssystem in ein Fahrzeug eingebaut ist, wobei das Fahrzeug eine Steuereinheit beinhaltet, die darin eingebaut ist, und die Datenextraktionseinheit mit der Steuereinheit kommunizieren kann und dazu ausgelegt ist, eine der Mehrzahl von Bestimmungseinheiten auf der Grundlage einer Auswahlinformation auszuwählen, wobei die Auswahlinformation irgendeine der Mehrzahl von Bestimmungseinheiten anzeigt, die ausgewählt werden sollte, und von der Steuereinheit zu der Datenextraktionseinheit gesendet wird.

13. Ein Datenumprogrammierungssystem nach Anspruch 10, wobei die Datenextraktionseinheit dazu ausgelegt ist, eine der Mehrzahl von Bestimmungseinheiten auf der Grundlage einer Auswahlinformation auszuwählen, wobei die Auswahlinformation einen Zeitpunkt anzeigt, zu welchem die Mehrzahl von Elementen von Daten empfangen wird.

14. Ein Datenumprogrammierungssystem nach Anspruch 10, wobei die Mehrzahl von Bestimmungseinheiten beinhaltet:
eine Spiegelbestimmungseinheit, die dazu ausgelegt ist, wenn die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, mindestens ein Element von bestimmten Daten und mindestens ein Element von Spiegeldaten beinhaltet, die zueinander komplementär sind und zu der Empfangseinheit gesendet werden, während sie miteinander gepaart sind, das mindestens ein Element von bestimmten Daten und das mindestens ein Element von Spiegeldaten zu überprüfen, um zu erfassen, dass das mindestens eine Element von bestimmten Daten komplementär zu den mindestens einem Element von Spiegeldaten ist, um dadurch zu bestimmen, dass die mindestens einen bestimmten Daten die vorbestimmte Regel gegenüber dem mindestens einen Element der Spiegeldaten erfüllen;
eine Identitätsbestimmungseinheit, die dazu ausgelegt ist, wenn die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, drei oder mehr Elemente von Daten beinhaltet, wobei mindestens zwei der drei oder mehr Elemente von Daten zueinander identisch sind, die drei oder mehr Elemente von Daten zu überprüfen, um zu bestimmen, dass mindestens zwei der drei oder mehr Elemente von Daten zueinander identisch sind, um dadurch zu bestimmen, dass eines der mindestens zwei identischen Elemente von Daten die vorbestimmte Regel gegenüber mindestens ein anderem der mindestens zwei identischen Elemente von Daten erfüllt;
eine Majoritätsbelegungs-Bestimmungseinheit, die dazu ausgelegt ist, wenn die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, drei oder mehr Elementen von Daten beinhaltet, wobei eine Majorität der drei oder mehr Ele-

mente von Daten zueinander identisch ist, die drei oder mehr Elemente von Daten zu überprüfen, um zu erfassen, dass die Majorität der drei oder mehr Elemente von Daten zueinander identisch ist, um dadurch zu bestimmen, dass mindestens eines der Mehrzahl der drei oder mehr Elemente von Daten die vorbestimmte Regel gegenüber einem anderen mindestens einem der Majorität der drei oder mehr Elemente von Daten erfüllt; und
eine Majoritätsregel-Bestimmungseinheit, die dazu ausgelegt ist, wenn die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, drei oder mehr Elemente von Daten beinhaltet, wobei die drei oder mehr Elemente von Daten mindestens ein erstes Element beinhalten, dessen Anzahl von Elementen zu denen in den drei oder mehr Elementen von Daten identisch ist, die drei oder mehr Elemente von Daten zu überprüfen, um zu erfassen, dass das mindestens eine erste Element von Daten, dessen Anzahl von Elementen zu denen in den drei oder mehr Elementen von Daten identisch ist, darin das größte ist, um dadurch zu bestimmen, dass das mindestens eine Element von Daten die vorbestimmte Regel gegenüber dem verbleibenden mindestens einem der Elemente erfüllt, das darin identisch ist.

15. Ein Datenumprogrammierungssystem nach Anspruch 14, wobei, wenn die Identitätsbestimmungseinheit aus der Mehrzahl von Bestimmungseinheiten ausgewählt wird, die Datenextraktionseinheit dazu ausgelegt ist, ein Empfangen der mindestens zwei der drei oder mehr Elemente von Daten auszulöschen, nachdem bestimmt worden ist, dass die mindestens zwei der drei oder mehr Elemente zueinander identisch sind.

16. Ein Datenumprogrammierungssystem nach Anspruch 14, wobei, wenn die Majoritätsregel-Bestimmungseinheit aus der Mehrzahl von Bestimmungseinheiten ausgewählt wird, die Datenextraktionseinheit dazu ausgelegt ist, in einem Fall eines Erfassens mindestens eines zweiten Elements von Daten, dessen Anzahl von Elementen zu denen in den drei oder mehr Elementen von Daten das größte darin ist, zusätzlich zu dem mindestens einen ersten Element, zu bestimmen, dass das mindestens eine erste Element der drei oder mehr Elemente von Daten nicht die vorbestimmte Regel erfüllt.

17. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei jedes der Elemente von Daten, die von der Empfangseinheit empfangen werden, im Voraus in eine Mehrzahl von Blöcken geteilt wird und aufeinanderfolgend in einem Block auf eine vorbestimmte Sendeweise gesendet wird zu der Empfangseinheit, wobei die vorbestimmte Sendeweise von der vorbestimmten Regel bestimmt wird und die Datenextraktionseinheit dazu ausgelegt ist:
in einem Block jedes der empfangenen Mehrzahl von Elementen von Daten zu überprüfen;
eine Mehrzahl von Blöcken des mindestens einen Elements der Daten als überprüfte Blöcke nach einem Bestimmen auf der Grundlage des überprüften Ergebnisses in einem Block zu extrahieren, dass die Mehrzahl von Blöcken des mindestens einen Elements von Daten die vorbestimmte Regel gegenüber einer Mehrzahl von Blöcken von mindestens einem verbleibenden Element von Daten erfüllt; und
die extrahierten Blöcke des mindestens einen Elements von Daten zu kombinieren, um das mindestens eine Element von Daten als die gültig gemachten Daten zu dekodieren.

18. Ein Datenumprogrammierungssystem nach Anspruch 17, wobei die Mehrzahl von Elementen von Daten, die von der Empfangseinheit empfangen wird, drei oder mehr Elemente von Daten beinhaltet, wobei mindestens zwei der drei oder mehr Elemente von Daten zueinander identisch sind, wobei jedes der Elemente von Daten, das von der Empfangseinheit empfangen wird, im Voraus derart in eine Mehrzahl von Blöcken geteilt wird, dass jeder Block der mindestens zwei der drei oder mehr Elemente und jeder Block des verbleibenden mindestens einen der drei oder mehr Elemente kombiniert werden, um eine Gruppe zu erzeugen, und die drei oder mehr Elemente von Daten aufeinanderfolgend zu der Empfangseinheit in einer Gruppe gesendet werden.

19. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei jedes der Mehrzahl von Elementen von Daten als ein Datenrahmen verarbeitet wird, wenn es in einem Paket zu der Empfangseinheit gesendet wird, wobei der Datenrahmen ein Datenfeld aufweist, in welchem jedes der Mehrzahl von Elementen von Daten enthalten ist, wobei jedes der Mehrzahl von Elementen, das in dem Datenfeld enthalten ist, im Voraus in eine Mehrzahl von Blöcken geteilt ist und die Datenextraktionseinheit dazu ausgelegt ist, die Blöcke von einem Element der Mehrzahl von Elementen von Daten und diejenigen eines anderen Elements der Mehrzahl von Elementen von Daten zu überprüfen.

20. Ein Datenumprogrammierungssystem nach Anspruch 1, wobei das Datenumprogrammierungssystem aus einer Mehrzahl von elektronischen Steuereinheiten besteht, die in ein Fahrzeug eingebaut sind, wobei die Mehrzahl von elektronischen Steuereinheiten über einen Kommunikationsbus miteinander kommunizierbar verbunden sind und im Zusammenwirken ein verteiltes Steuern für das Fahrzeug durchführen, wobei die Empfangseinheit und die Datenextraktionseinheit in mindestens eine der elektronischen Steuereinheiten eingebaut sind.

21. Ein Datenumprogrammierungs-Programmprodukt, das in einen Speicher eines Computers eingebettet ist, der mit einem umschreibbaren Speicher kommunizieren kann, in welchem im Voraus Daten

gespeichert sind, wobei die gespeicherten Daten auf der Grundlage von gültig gemachten Daten umprogrammierbar sind, wobei das Datenumprogrammierungs-Programmprodukt aufweist:
eine erste Einrichtung zum Anweisen eines Computers, eine Mehrzahl von Elementen von Daten zum Verwenden beim Umprogrammieren zu empfangen, wobei die Mehrzahl von Elementen von Daten zu dem Computer gesendet wird, wobei der Mehrzahl von Elementen von Daten jeweils der gleiche Identifizierer hinzugefügt ist; und
eine zweite Einrichtung zum Anweisen eines Computers, die empfangene Mehrzahl von Elementen von Daten zu überprüfen und mindestens ein Element der Daten als die gültig gemachten Daten nach einem Bestimmen auf der Grundlage des überprüften Ergebnisses zu extrahieren, dass das mindestens eine Element der Daten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element der Daten erfüllt.

22. Ein Verfahren zum Umprogrammieren von Daten, die im Voraus in einem umschreibbaren Speicher gespeichert sind, auf der Grundlage von gültig gemachten Daten, wobei das Verfahren umfasst:
Empfangen einer Mehrzahl von Elementen von Daten zum Verwenden beim Umprogrammieren, wobei die Mehrzahl von Elementen von Daten derart gesendet wird, dass der Mehrzahl von Elementen von Daten jeweils der gleiche Identifizierer hinzugefügt ist;
Überprüfen der empfangenen Mehrzahl von Elementen von Daten; und
Extrahieren mindestens eines Elements der Daten als die gültig gemachten Daten nach einem Bestimmen, dass das mindestens eine Element der Daten eine vorbestimmte Regel gegenüber mindestens einem verbleibenden Element der Daten erfüllt, auf der Grundlage des überprüften Ergebnisses.

Es folgen 30 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1

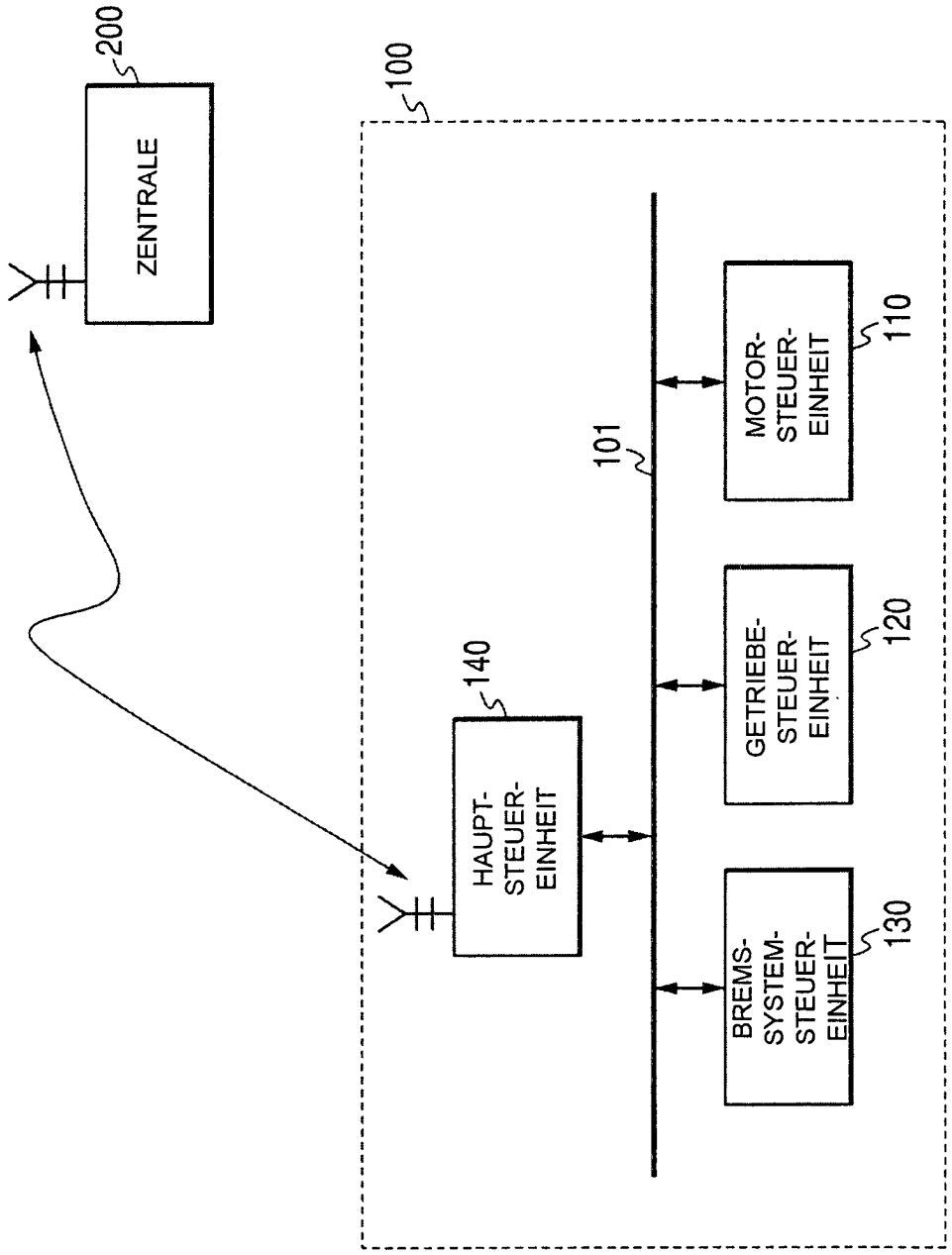


FIG. 2

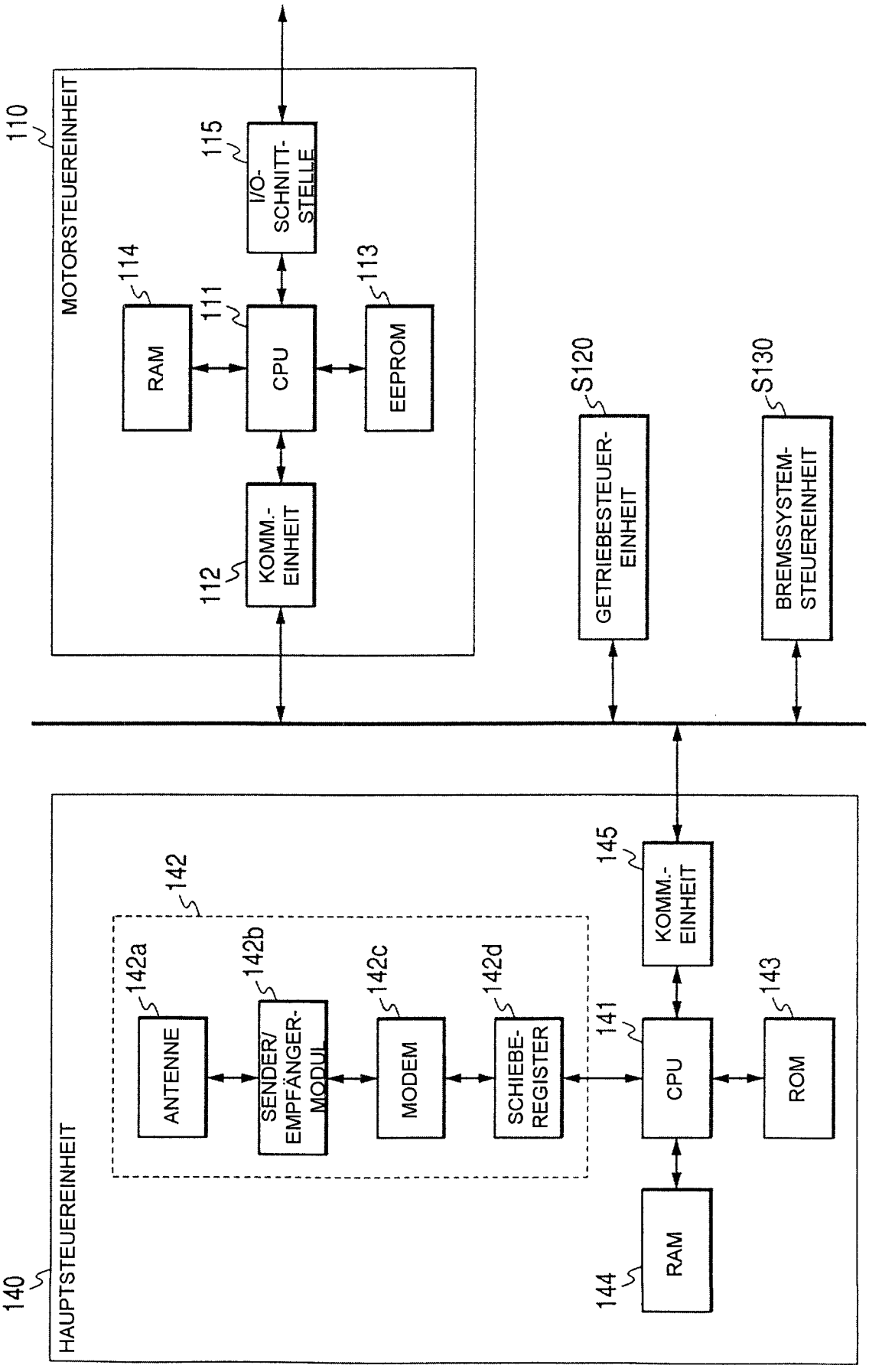

110
MOTORSTEUEREINHEIT
115
I/O-SCHNITT-STELLE
114
RAM
111
CPU
113
EEPROM
112
KOMM.-EINHEIT
S120
GETRIEBESTEUER-EINHEIT
S130
BREMSSYSTEM-STEUEREINHEIT
140
HAUPTSTEUEREINHEIT
142
142a
ANTENNE
142b
SENDER/EMPFÄNGER-MODUL
142c
MODEM
142d
SCHIEBE-REGISTER
145
KOMM.-EINHEIT
141
CPU
143
ROM
144
RAM

FIG. 3A

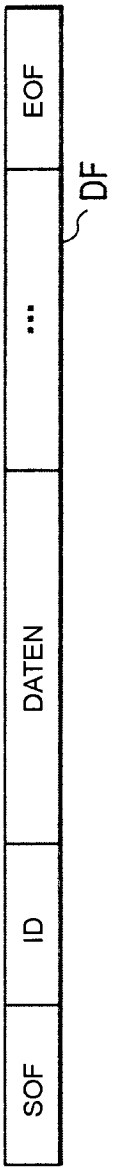

SOF
ID
DATEN
...
EOF
DF

FIG. 3B

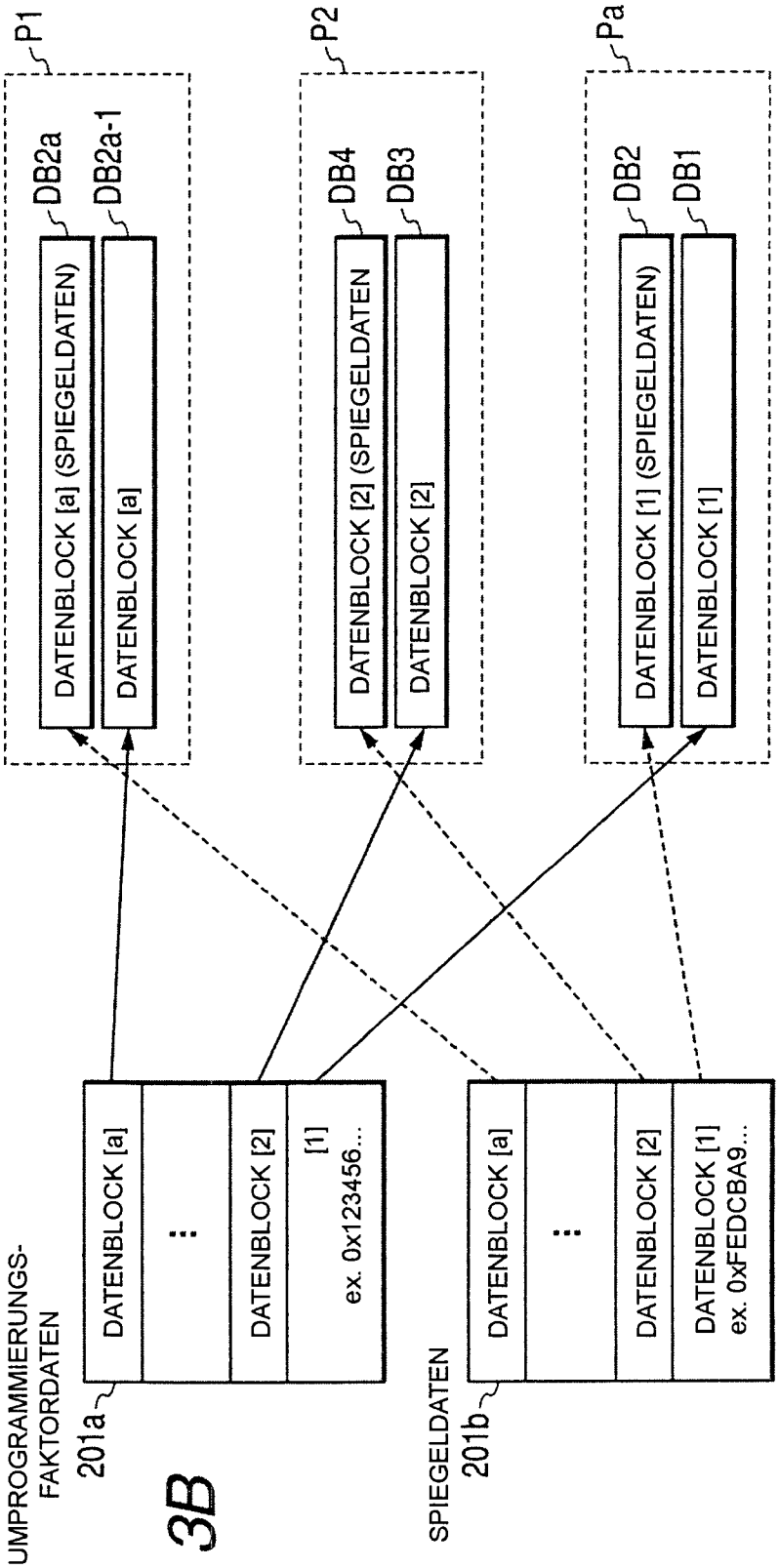

P1
P2
Pa
DB2a
DB2a-1
DB4
DB3
DB2
DB1
DATENBLOCK [a] (SPIEGELDATEN)
DATENBLOCK [a]
DATENBLOCK [2] (SPIEGELDATEN
DATENBLOCK [2]
DATENBLOCK [1] (SPIEGELDATEN)
DATENBLOCK [1]
UMPROGRAMMIERUNGS-FAKTORDATEN
201a
DATENBLOCK [a]
DATENBLOCK [2]
[1]
ex. 0x123456...
SPIEGELDATEN
201b
DATENBLOCK [a]
DATENBLOCK [2]
DATENBLOCK [1]
ex. 0xFEDCBA9...

FIG. 4

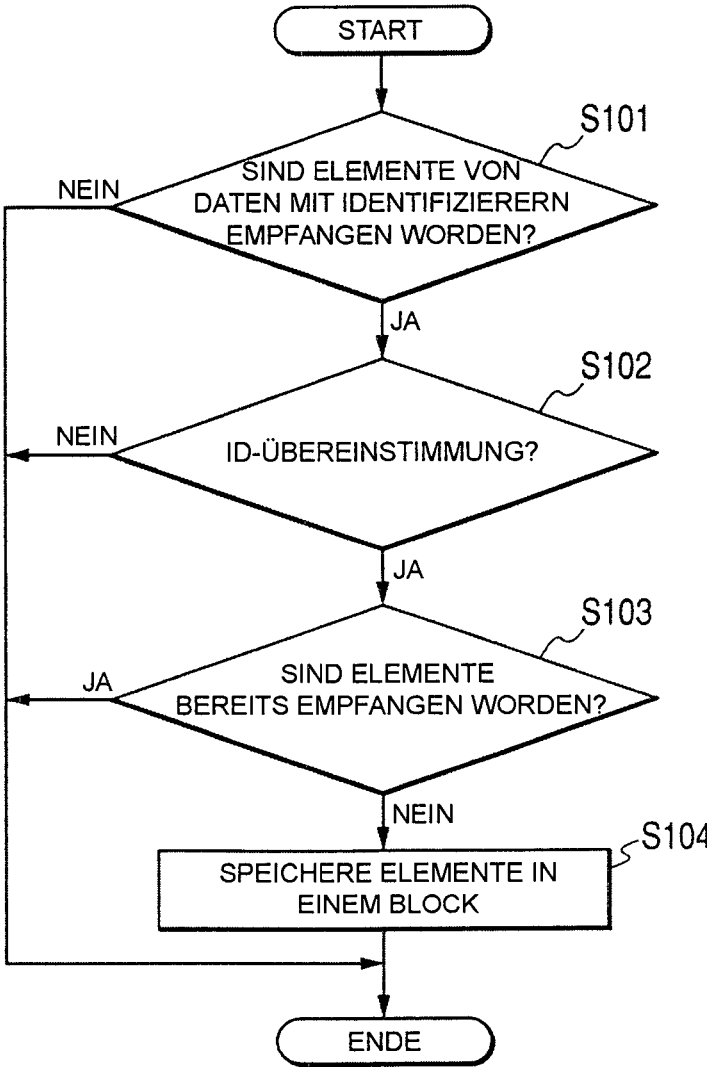

START
S101
SIND ELEMENTE VON DATEN MIT IDENTIFIZIERERN EMPFANGEN WORDEN?
NEIN
JA
S102
ID-ÜBEREINSTIMMUNG?
NEIN
JA
S103
SIND ELEMENTE BEREITS EMPFANGEN WORDEN?
JA
NEIN
S104
SPEICHERE ELEMENTE IN EINEM BLOCK
ENDE

FIG. 5

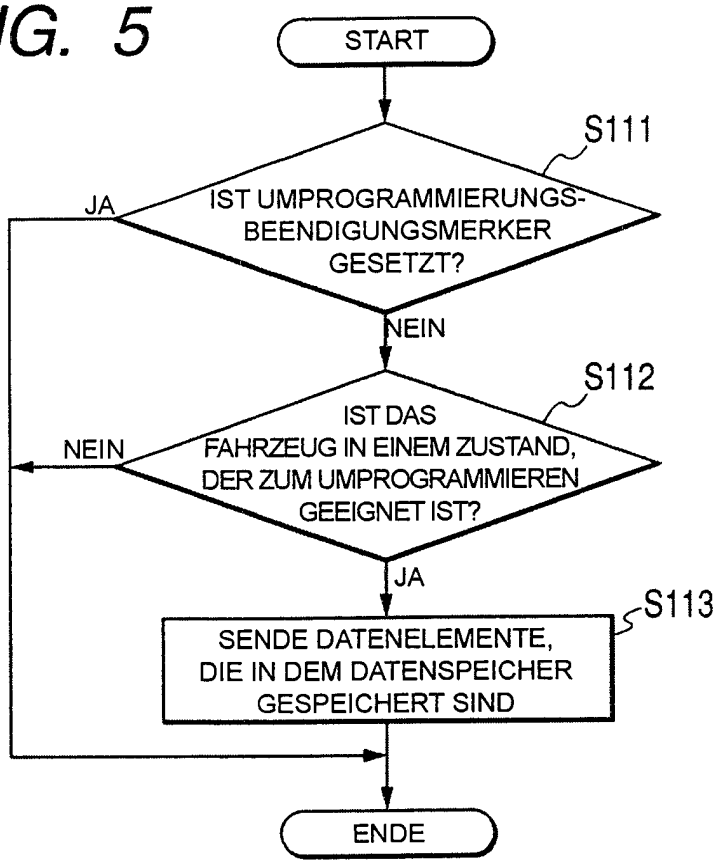

START
S111
IST UMPROGRAMMIERUNGS-BEENDIGUNGSMERKER GESETZT?
JA
NEIN
S112
IST DAS FAHRZEUG IN EINEM ZUSTAND, DER ZUM UMPROGRAMMIEREN GEEIGNET IST?
NEIN
JA
S113
SENDE DATENELEMENTE, DIE IN DEM DATENSPEICHER GESPEICHERT SIND
ENDE

FIG. 6

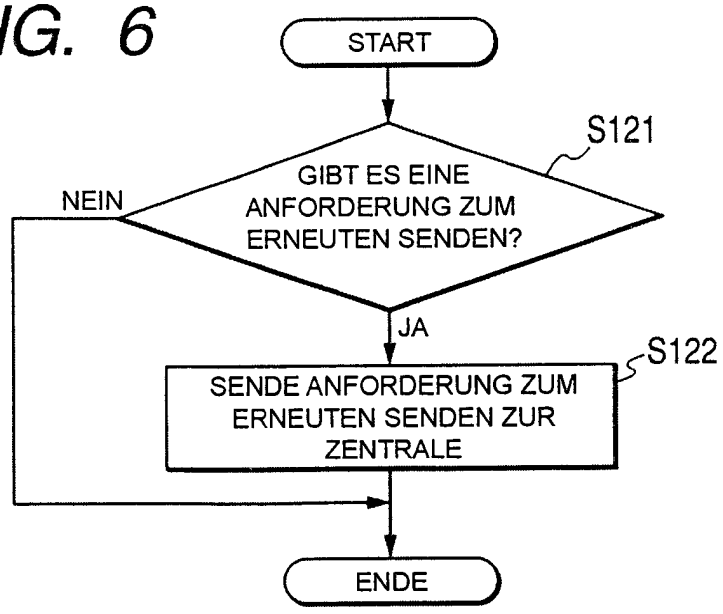

START
S121
GIBT ES EINE ANFORDERUNG ZUM ERNEUTEN SENDEN?
NEIN
JA
S122
SENDE ANFORDERUNG ZUM ERNEUTEN SENDEN ZUR ZENTRALE
ENDE

FIG. 7

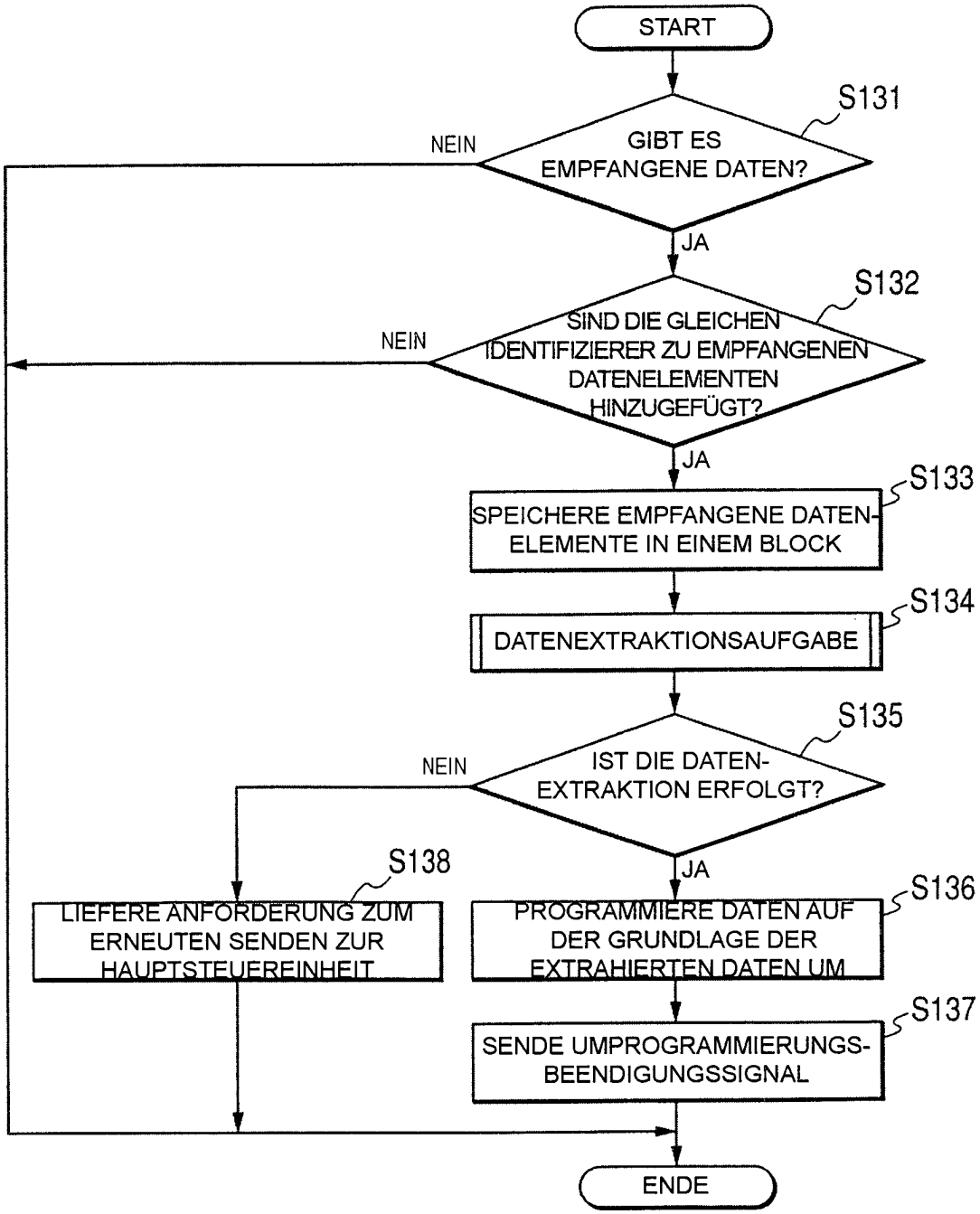

START
S131
GIBT ES EMPFANGENE DATEN?
NEIN
JA
S132
SIND DIE GLEICHEN IDENTIFIZIERER ZU EMPFANGENEN DATENELEMENTEN HINZUGEFÜGT?
NEIN
JA
S133
SPEICHERE EMPFANGENE DATEN-ELEMENTE IN EINEM BLOCK
S134
DATENEXTRAKTIONSAUFGABE
S135
IST DIE DATEN-EXTRAKTION ERFOLGT?
NEIN
JA
S138
LIEFERE ANFORDERUNG ZUM ERNEUTEN SENDEN ZUR HAUPTSTEUEREINHEIT
S136
PROGRAMMIERE DATEN AUF DER GRUNDLAGE DER EXTRAHIERTEN DATEN UM
S137
SENDE UMPROGRAMMIERUNGS-BEENDIGUNGSSIGNAL
ENDE

FIG. 8

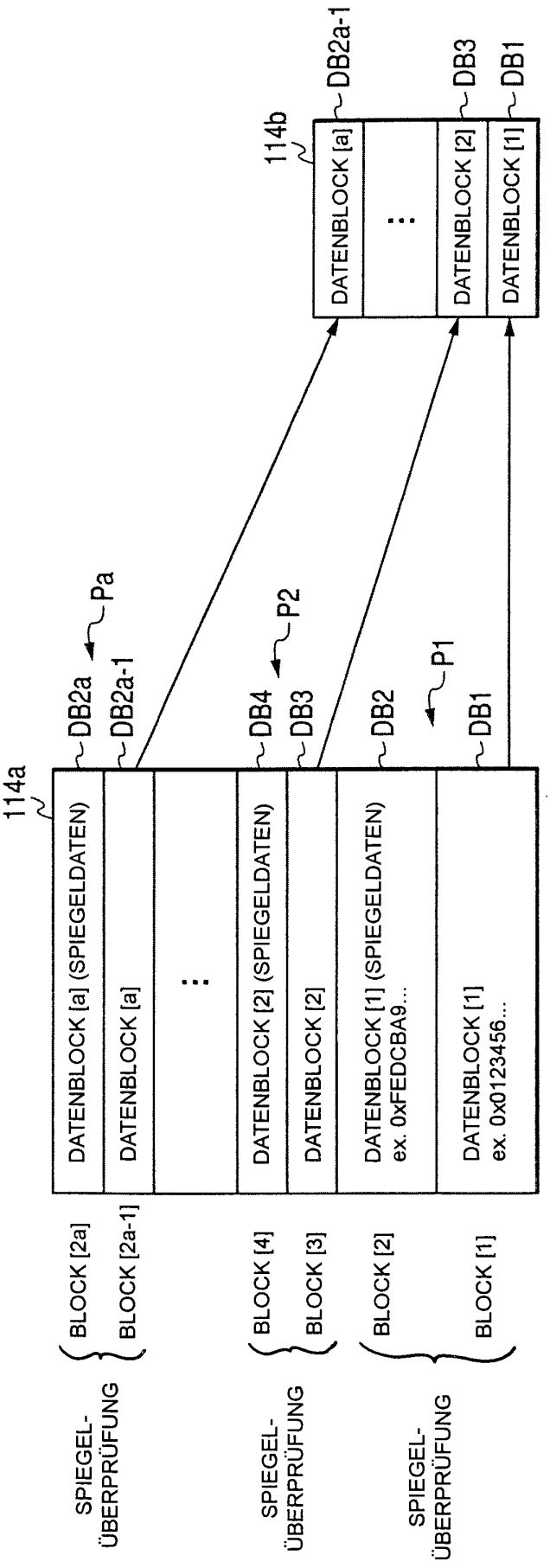

114a
DATENBLOCK [a] (SPIEGELDATEN)
DATENBLOCK [a]
DATENBLOCK [2] (SPIEGELDATEN)
DATENBLOCK [2]
DATENBLOCK [1] (SPIEGELDATEN)
ex. 0xFEDCBA9...
DATENBLOCK [1]
ex. 0x0123456...
DB2a
DB2a-1
DB4
DB3
DB2
DB1
Pa
P2
P1
114b
DATENBLOCK [a]
DATENBLOCK [2]
DATENBLOCK [1]
DB2a-1
DB3
DB1
BLOCK [2a]
BLOCK [2a-1]
BLOCK [4]
BLOCK [3]
BLOCK [2]
BLOCK [1]
SPIEGEL-ÜBERPRÜFUNG
SPIEGEL-ÜBERPRÜFUNG
SPIEGEL-ÜBERPRÜFUNG

FIG. 9

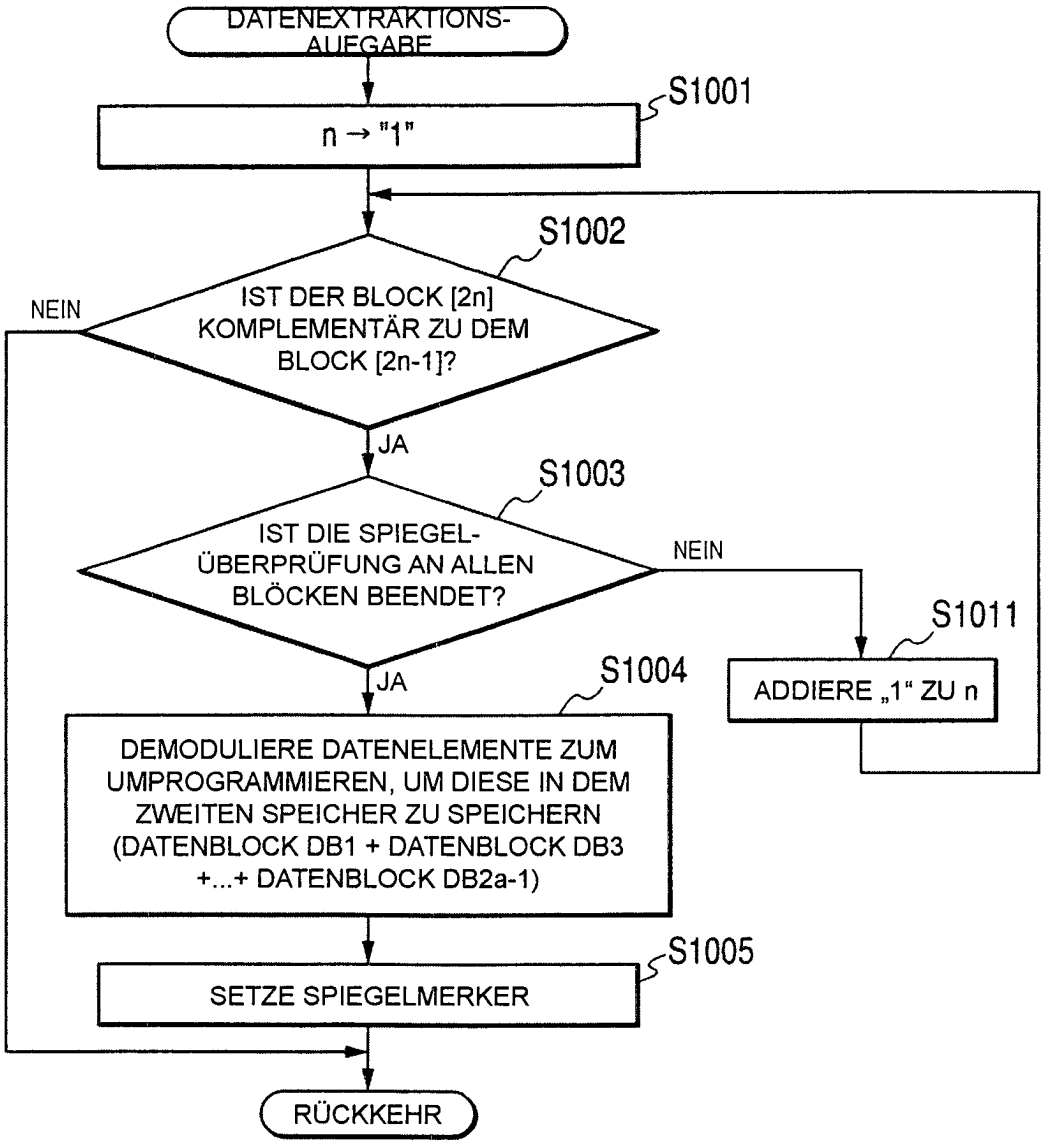

DATENEXTRAKTIONS-AUFGABE
S1001
n → "1"
S1002
IST DER BLOCK [2n] KOMPLEMENTÄR ZU DEM BLOCK [2n-1]?
NEIN
JA
S1003
IST DIE SPIEGEL-ÜBERPRÜFUNG AN ALLEN BLÖCKEN BEENDET?
NEIN
S1011
ADDIERE „1" ZU n
JA
S1004
DEMODULIERE DATENELEMENTE ZUM UMPROGRAMMIEREN, UM DIESE IN DEM ZWEITEN SPEICHER ZU SPEICHERN (DATENBLOCK DB1 + DATENBLOCK DB3 +...+ DATENBLOCK DB2a-1)
S1005
SETZE SPIEGELMERKER
RÜCKKEHR

FIG. 10A

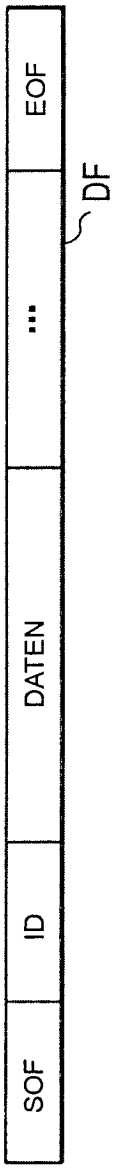


FIG. 10B

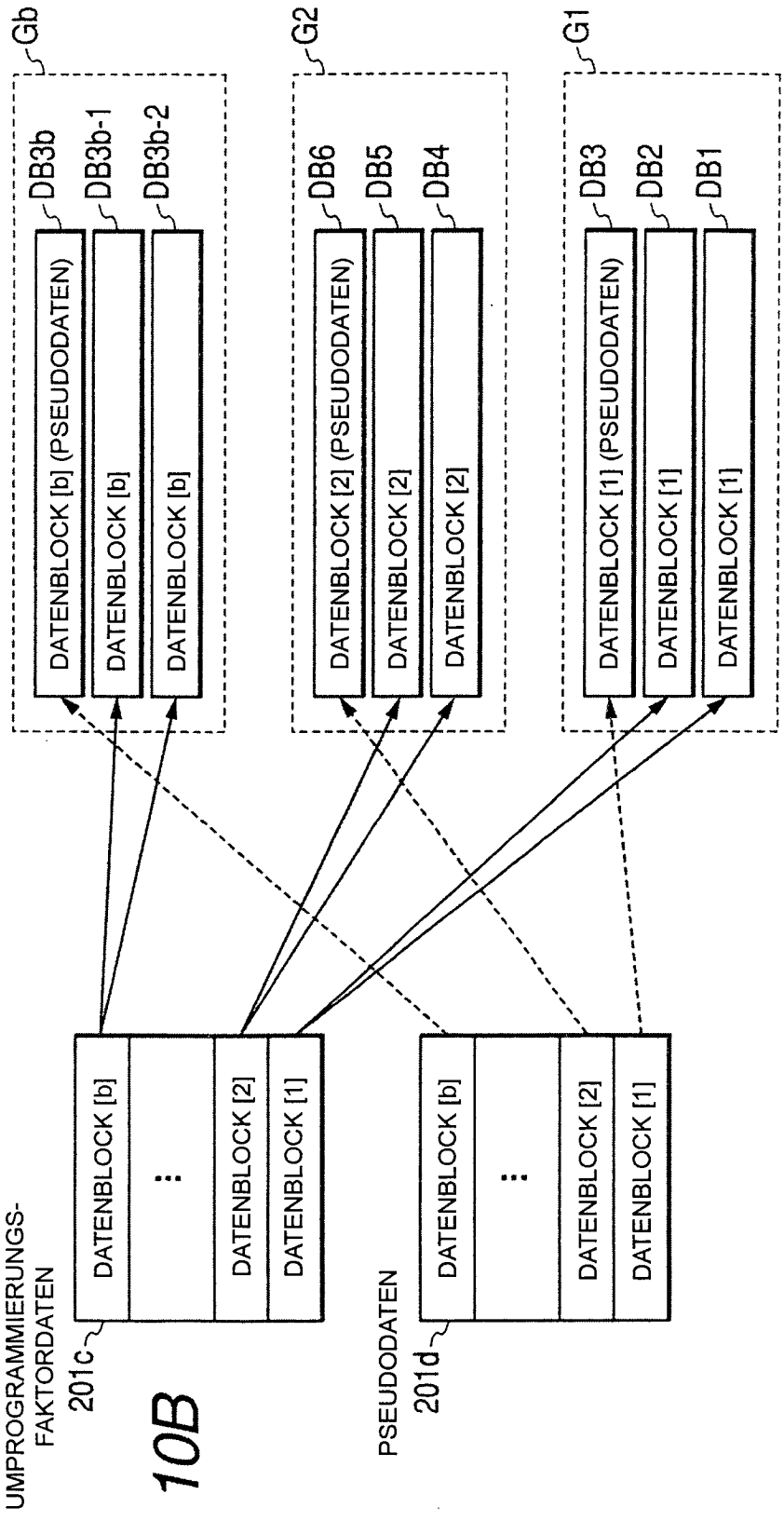


FIG. 11

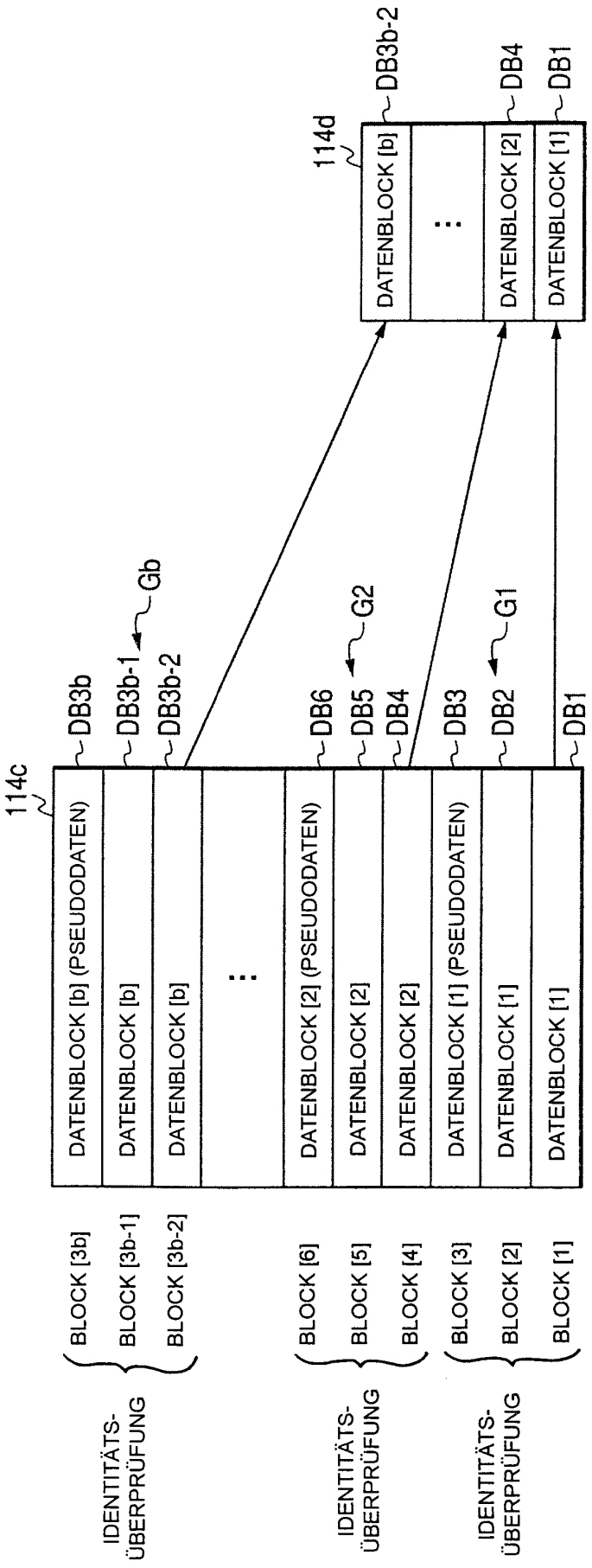

114c
DATENBLOCK [b] (PSEUDODATEN)
DATENBLOCK [b]
DATENBLOCK [b]
DATENBLOCK [2] (PSEUDODATEN)
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [1] (PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1]
DB3b
DB3b-1
DB3b-2
DB6
DB5
DB4
DB3
DB2
DB1
Gb
G2
G1
114d
DATENBLOCK [b]
DATENBLOCK [2]
DATENBLOCK [1]
DB3b-2
DB4
DB1
BLOCK [3b]
BLOCK [3b-1]
BLOCK [3b-2]
BLOCK [6]
BLOCK [5]
BLOCK [4]
BLOCK [3]
BLOCK [2]
BLOCK [1]
IDENTITÄTS-ÜBERPRÜFUNG
IDENTITÄTS-ÜBERPRÜFUNG
IDENTITÄTS-ÜBERPRÜFUNG

FIG. 12

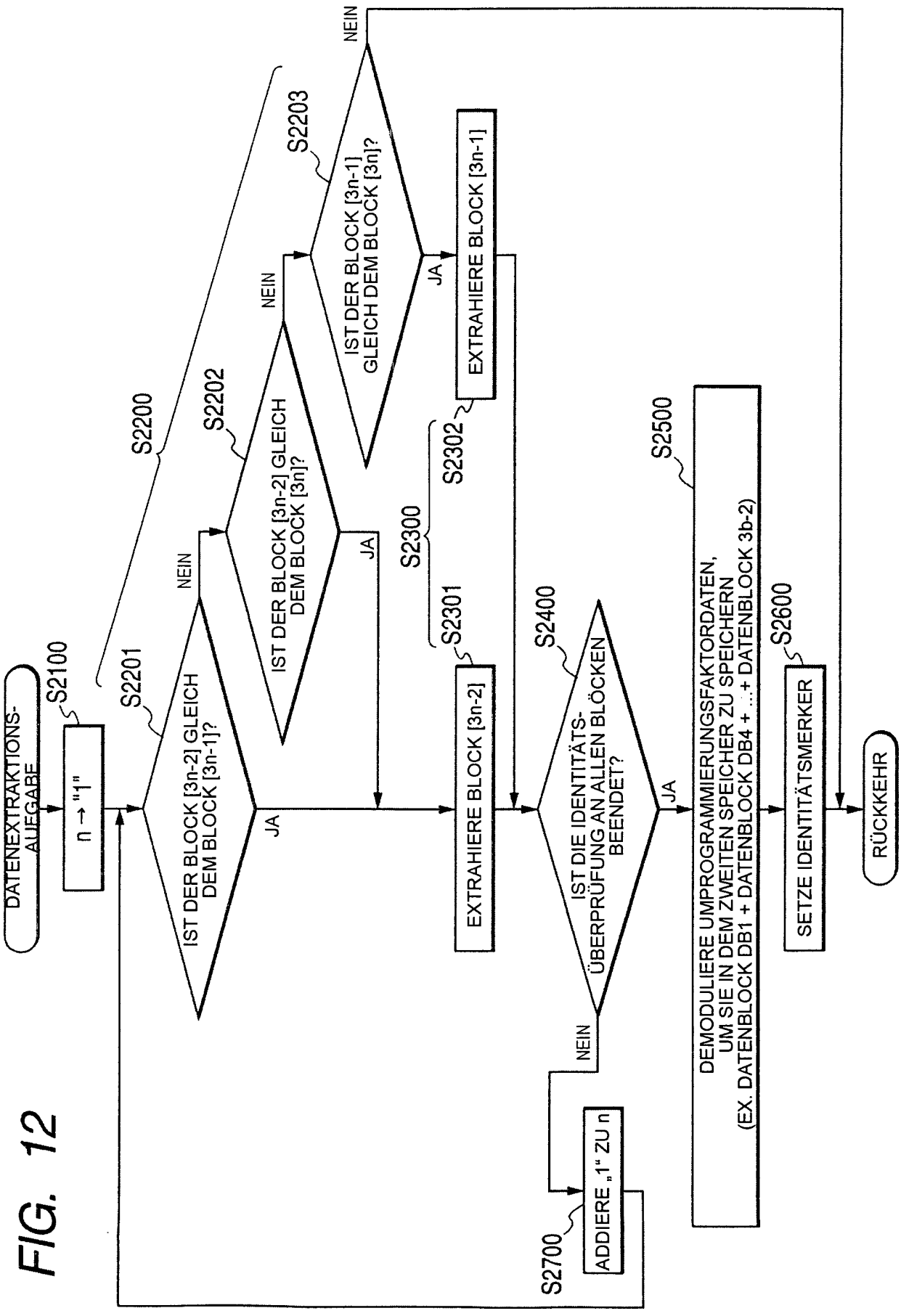

DATENEXTRAKTIONS-AUFGABE
S2100
n → "1"
S2200
S2201
IST DER BLOCK [3n-2] GLEICH DEM BLOCK [3n-1]?
NEIN
JA
S2202
IST DER BLOCK [3n-2] GLEICH DEM BLOCK [3n]?
NEIN
JA
S2203
IST DER BLOCK [3n-1] GLEICH DEM BLOCK [3n]?
NEIN
JA
S2300
S2301
EXTRAHIERE BLOCK [3n-2]
S2302
EXTRAHIERE BLOCK [3n-1]
S2400
IST DIE IDENTITÄTS-ÜBERPRÜFUNG AN ALLEN BLÖCKEN BEENDET?
NEIN
JA
S2700
ADDIERE „1" ZU n
S2500
DEMODULIERE UMPROGRAMMIERUNGSFAKTORDATEN, UM SIE IN DEM ZWEITEN SPEICHER ZU SPEICHERN (EX. DATENBLOCK DB1 + DATENBLOCK DB4 + ...+ DATENBLOCK 3b-2)
S2600
SETZE IDENTITÄTSMERKER
RÜCKKEHR

FIG. 13A

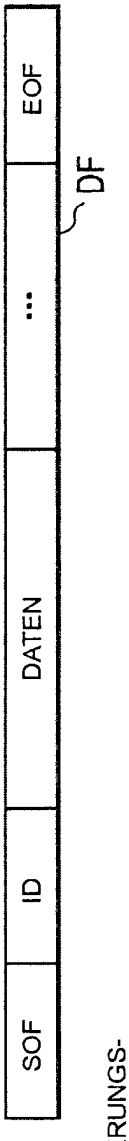


FIG. 13B

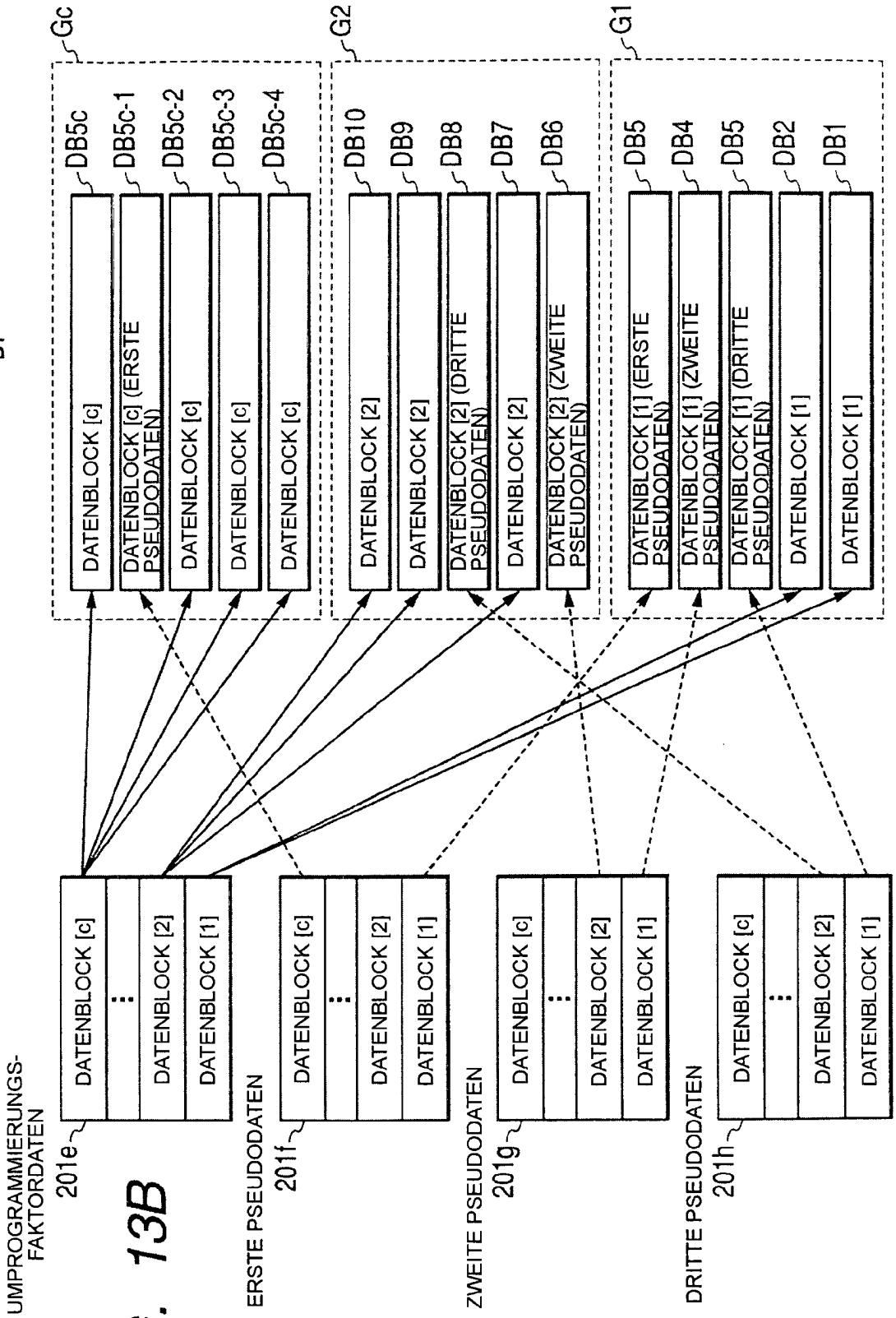


FIG. 14

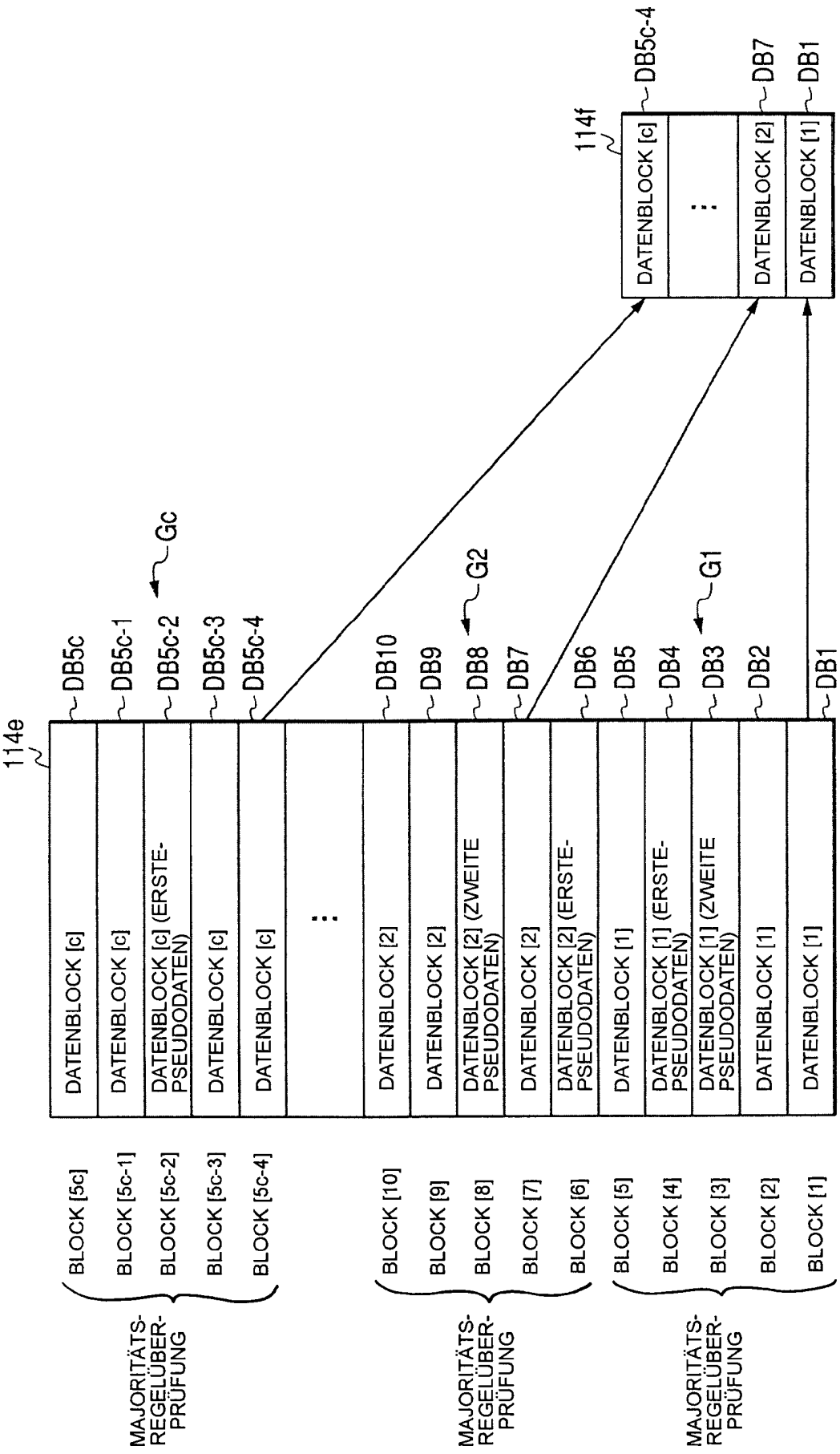

114e
DB5c
DB5c-1
DB5c-2
DB5c-3
DB5c-4
Gc
DB10
DB9
DB8
DB7
G2
DB6
DB5
DB4
DB3
DB2
DB1
G1
DATENBLOCK [c]
DATENBLOCK [c]
DATENBLOCK [c] (ERSTE-PSEUDODATEN)
DATENBLOCK [c]
DATENBLOCK [c]
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [2] (ZWEITE PSEUDODATEN)
DATENBLOCK [2]
DATENBLOCK [2] (ERSTE-PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1] (ERSTE-PSEUDODATEN)
DATENBLOCK [1] (ZWEITE PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1]
114f
DB5c-4
DB7
DB1
DATENBLOCK [c]
DATENBLOCK [2]
DATENBLOCK [1]
BLOCK [5c]
BLOCK [5c-1]
BLOCK [5c-2]
BLOCK [5c-3]
BLOCK [5c-4]
BLOCK [10]
BLOCK [9]
BLOCK [8]
BLOCK [7]
BLOCK [6]
BLOCK [5]
BLOCK [4]
BLOCK [3]
BLOCK [2]
BLOCK [1]
MAJORITÄTS-REGELÜBER-PRÜFUNG
MAJORITÄTS-REGELÜBER-PRÜFUNG
MAJORITÄTS-REGELÜBER-PRÜFUNG

FIG. 15

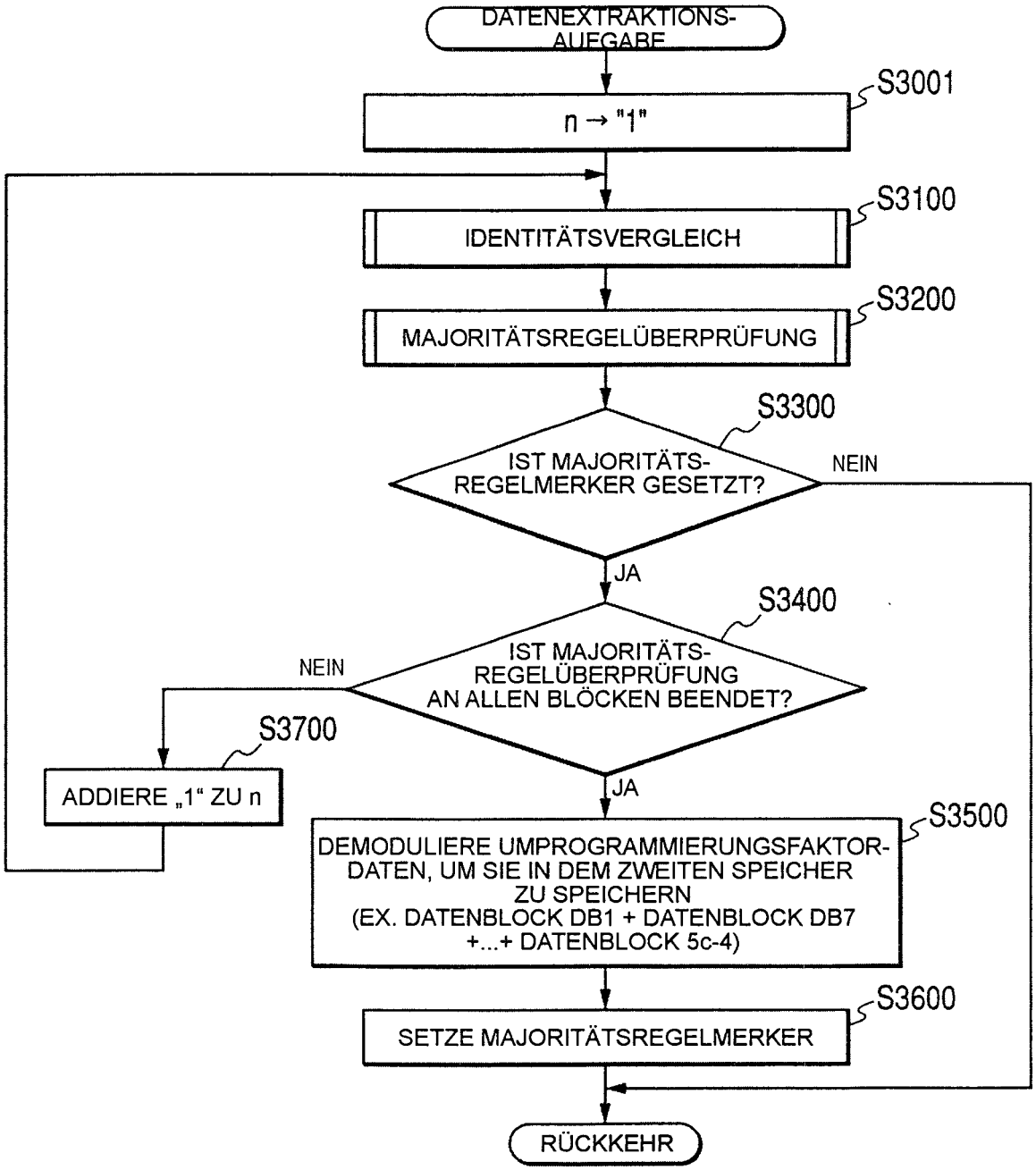

DATENEXTRAKTIONS-AUFGABE
S3001
n → "1"
S3100
IDENTITÄTSVERGLEICH
S3200
MAJORITÄTSREGELÜBERPRÜFUNG
S3300
IST MAJORITÄTS-REGELMERKER GESETZT?
NEIN
JA
S3400
IST MAJORITÄTS-REGELÜBERPRÜFUNG AN ALLEN BLÖCKEN BEENDET?
NEIN
S3700
ADDIERE „1" ZU n
JA
S3500
DEMODULIERE UMPROGRAMMIERUNGSFAKTOR-DATEN, UM SIE IN DEM ZWEITEN SPEICHER ZU SPEICHERN (EX. DATENBLOCK DB1 + DATENBLOCK DB7 +...+ DATENBLOCK 5c-4)
S3600
SETZE MAJORITÄTSREGELMERKER
RÜCKKEHR

FIG. 16A

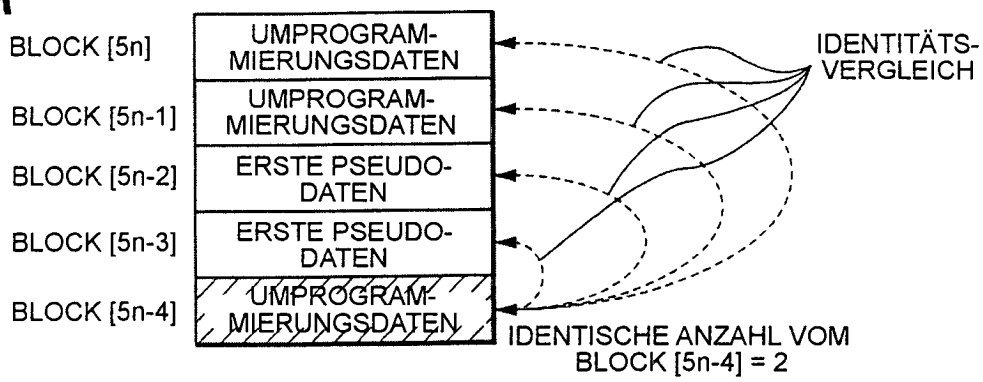


FIG. 16B

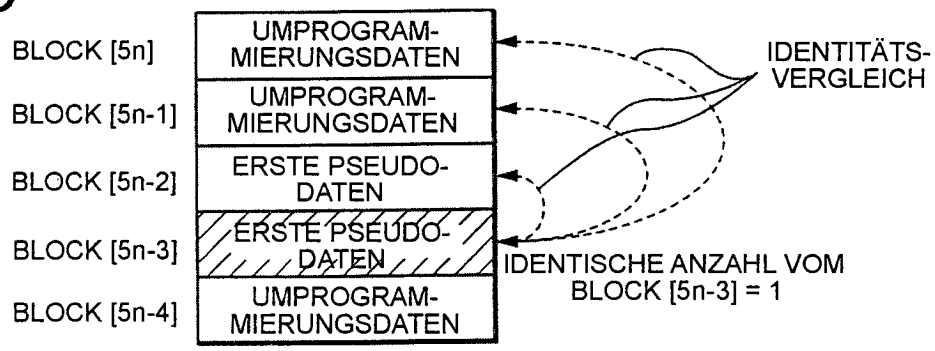


FIG. 16C

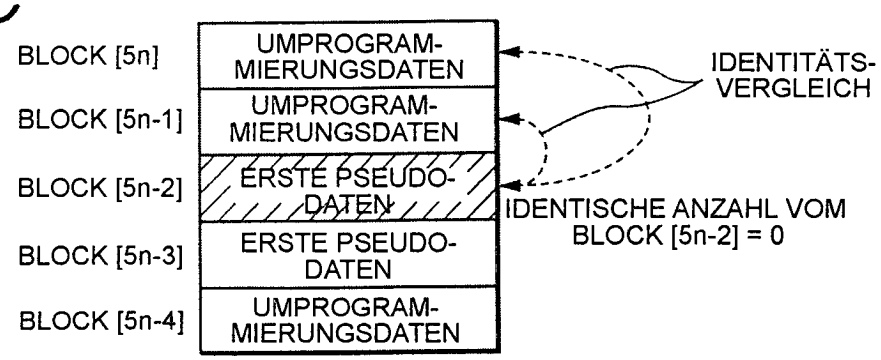


FIG. 16D

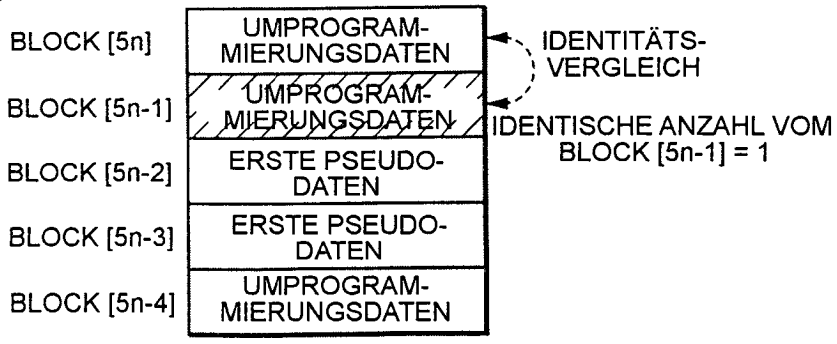


FIG. 17

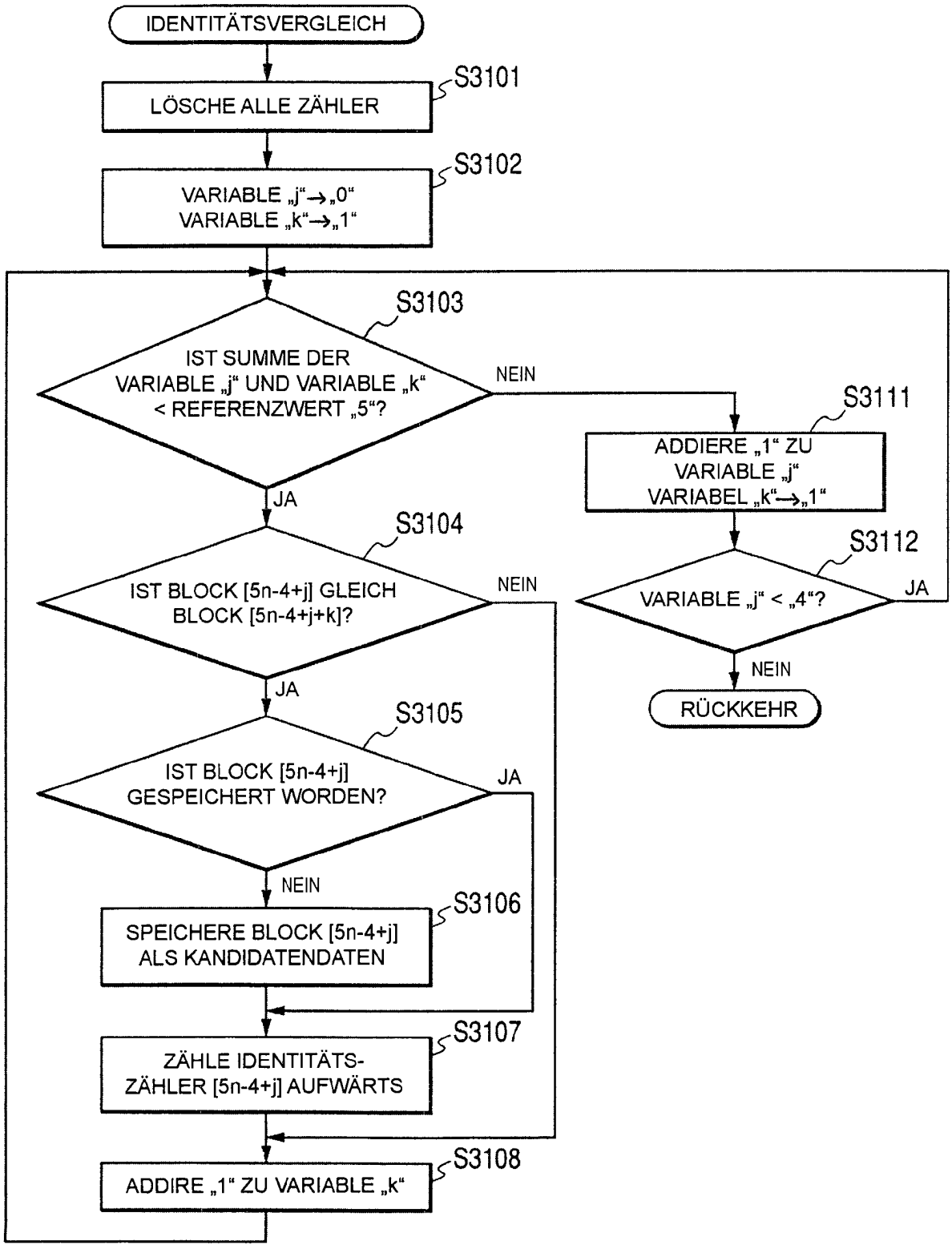

IDENTITÄTSVERGLEICH
S3101
LÖSCHE ALLE ZÄHLER
S3102
VARIABLE „j"→„0"
VARIABLE „k"→„1"
S3103
IST SUMME DER VARIABLE „j" UND VARIABLE „k" < REFERENZWERT „5"?
NEIN
JA
S3111
ADDIERE „1" ZU VARIABLE „j"
VARIABEL „k"→„1"
S3112
VARIABLE „j" < „4"?
JA
NEIN
RÜCKKEHR
S3104
IST BLOCK [5n-4+j] GLEICH BLOCK [5n-4+j+k]?
NEIN
JA
S3105
IST BLOCK [5n-4+j] GESPEICHERT WORDEN?
JA
NEIN
S3106
SPEICHERE BLOCK [5n-4+j] ALS KANDIDATENDATEN
S3107
ZÄHLE IDENTITÄTS-ZÄHLER [5n-4+j] AUFWÄRTS
S3108
ADDIRE „1" ZU VARIABLE „k"

FIG. 18

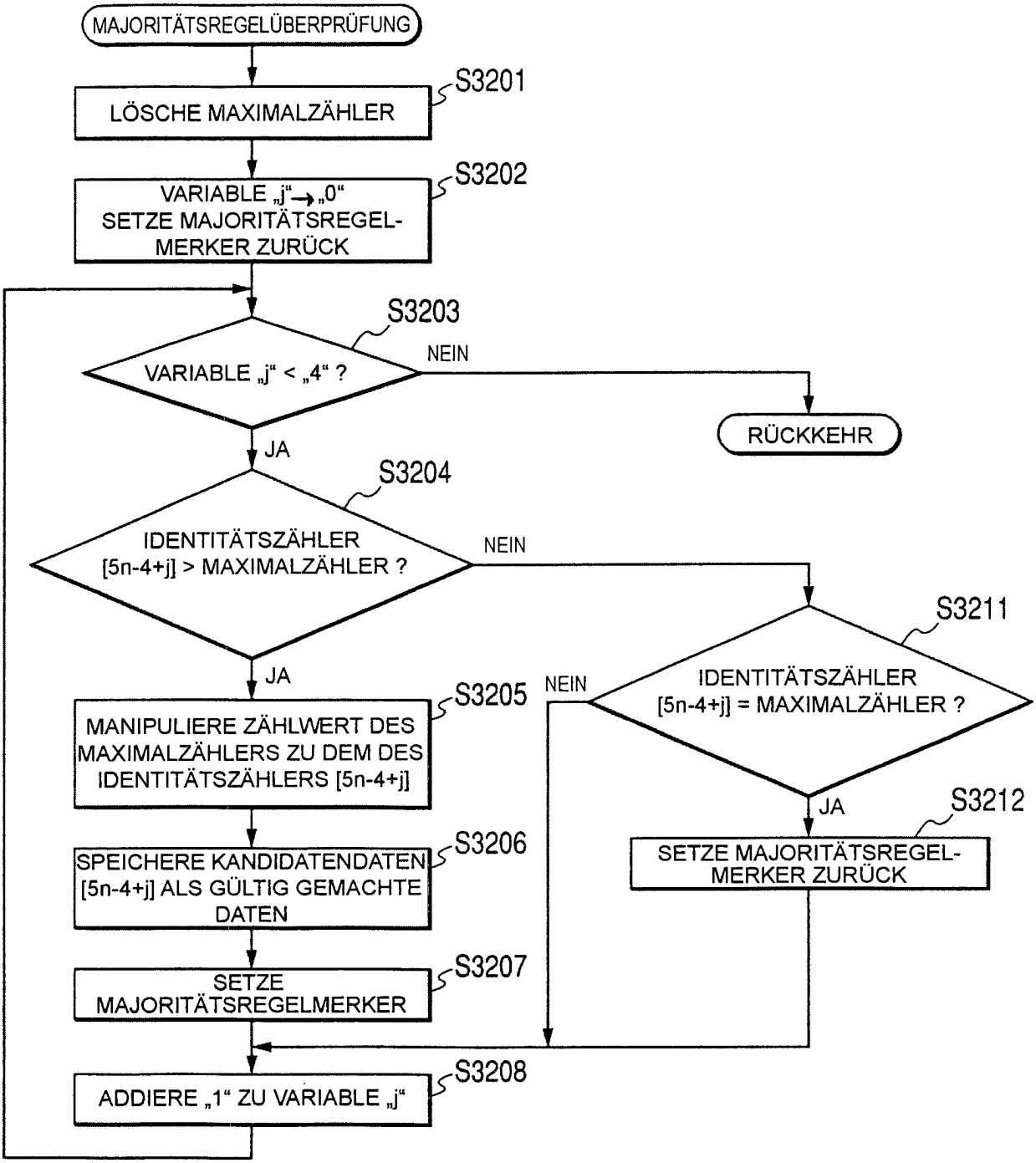

MAJORITÄTSREGELÜBERPRÜFUNG
S3201
LÖSCHE MAXIMALZÄHLER
S3202
VARIABLE „j“→„0“
SETZE MAJORITÄTSREGEL-
MERKER ZURÜCK
S3203
VARIABLE „j“ < „4“ ?
NEIN
RÜCKKEHR
JA
S3204
IDENTITÄTSZÄHLER
[5n-4+j] > MAXIMALZÄHLER ?
NEIN
S3211
JA
NEIN
IDENTITÄTSZÄHLER
[5n-4+j] = MAXIMALZÄHLER ?
S3205
MANIPULIERE ZÄHLWERT DES
MAXIMALZÄHLERS ZU DEM DES
IDENTITÄTSZÄHLERS [5n-4+j]
JA
S3212
SETZE MAJORITÄTSREGEL-
MERKER ZURÜCK
S3206
SPEICHERE KANDIDATENDATEN
[5n-4+j] ALS GÜLTIG GEMACHTE
DATEN
S3207
SETZE
MAJORITÄTSREGELMERKER
S3208
ADDIERE „1“ ZU VARIABLE „j“

FIG. 19

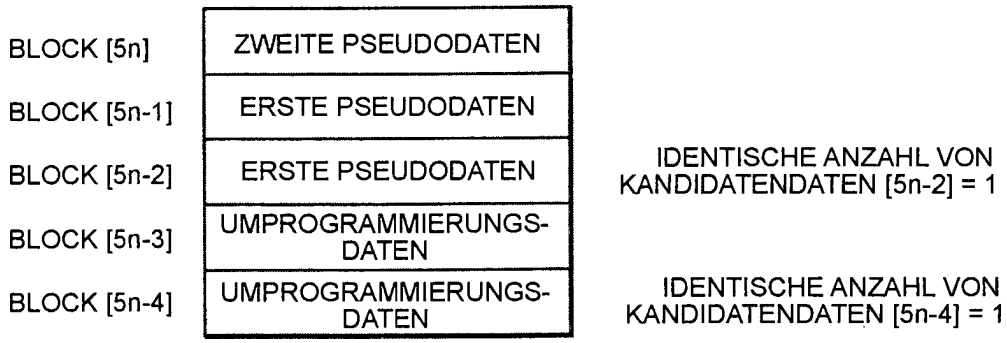
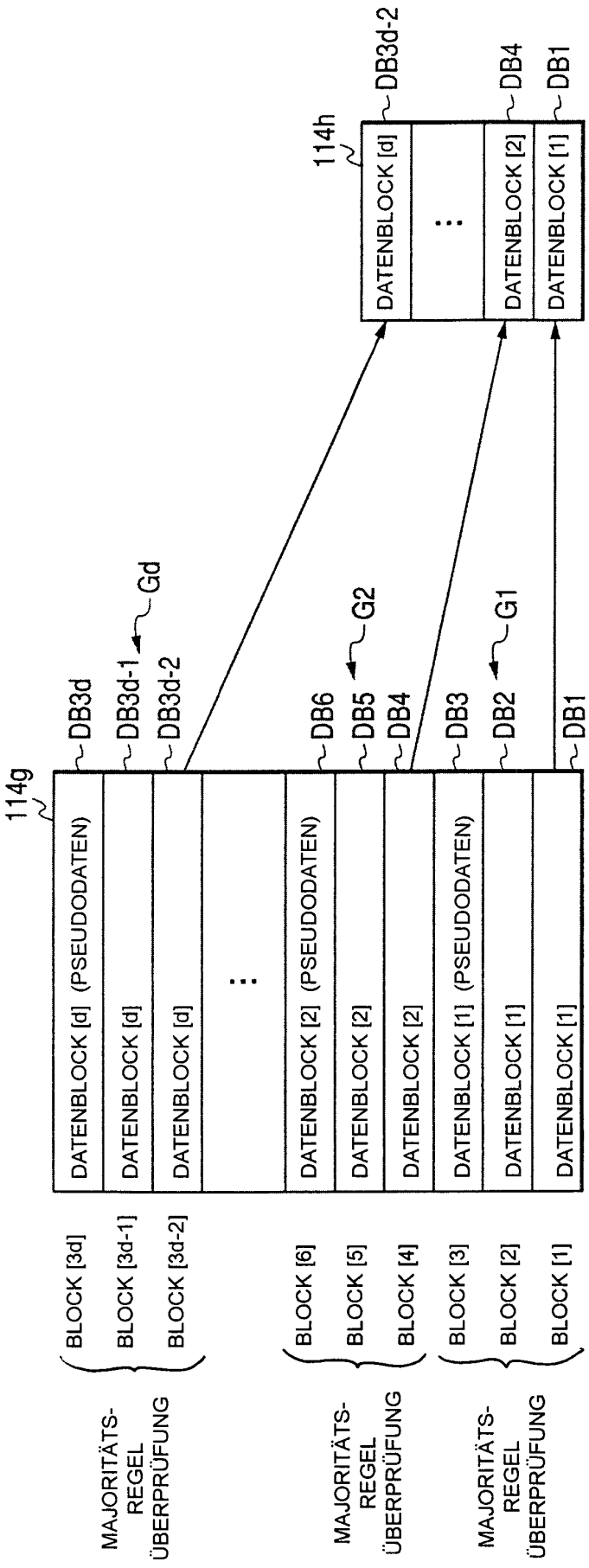

BLOCK [5n]
ZWEITE PSEUDODATEN
BLOCK [5n-1]
ERSTE PSEUDODATEN
BLOCK [5n-2]
ERSTE PSEUDODATEN
IDENTISCHE ANZAHL VON KANDIDATENDATEN [5n-2] = 1
BLOCK [5n-3]
UMPROGRAMMIERUNGS-DATEN
BLOCK [5n-4]
UMPROGRAMMIERUNGS-DATEN
IDENTISCHE ANZAHL VON KANDIDATENDATEN [5n-4] = 1

FIG. 20


114g
DATENBLOCK [d] (PSEUDODATEN)
DATENBLOCK [d]
DATENBLOCK [d]
DATENBLOCK [2] (PSEUDODATEN)
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [1] (PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1]
DB3d
DB3d-1
DB3d-2
DB6
DB5
DB4
DB3
DB2
DB1
Gd
G2
G1
114h
DATENBLOCK [d]
DATENBLOCK [2]
DATENBLOCK [1]
DB3d-2
DB4
DB1
BLOCK [3d]
BLOCK [3d-1]
BLOCK [3d-2]
BLOCK [6]
BLOCK [5]
BLOCK [4]
BLOCK [3]
BLOCK [2]
BLOCK [1]
MAJORITÄTS-REGEL ÜBERPRÜFUNG
MAJORITÄTS-REGEL ÜBERPRÜFUNG
MAJORITÄTS-REGEL ÜBERPRÜFUNG

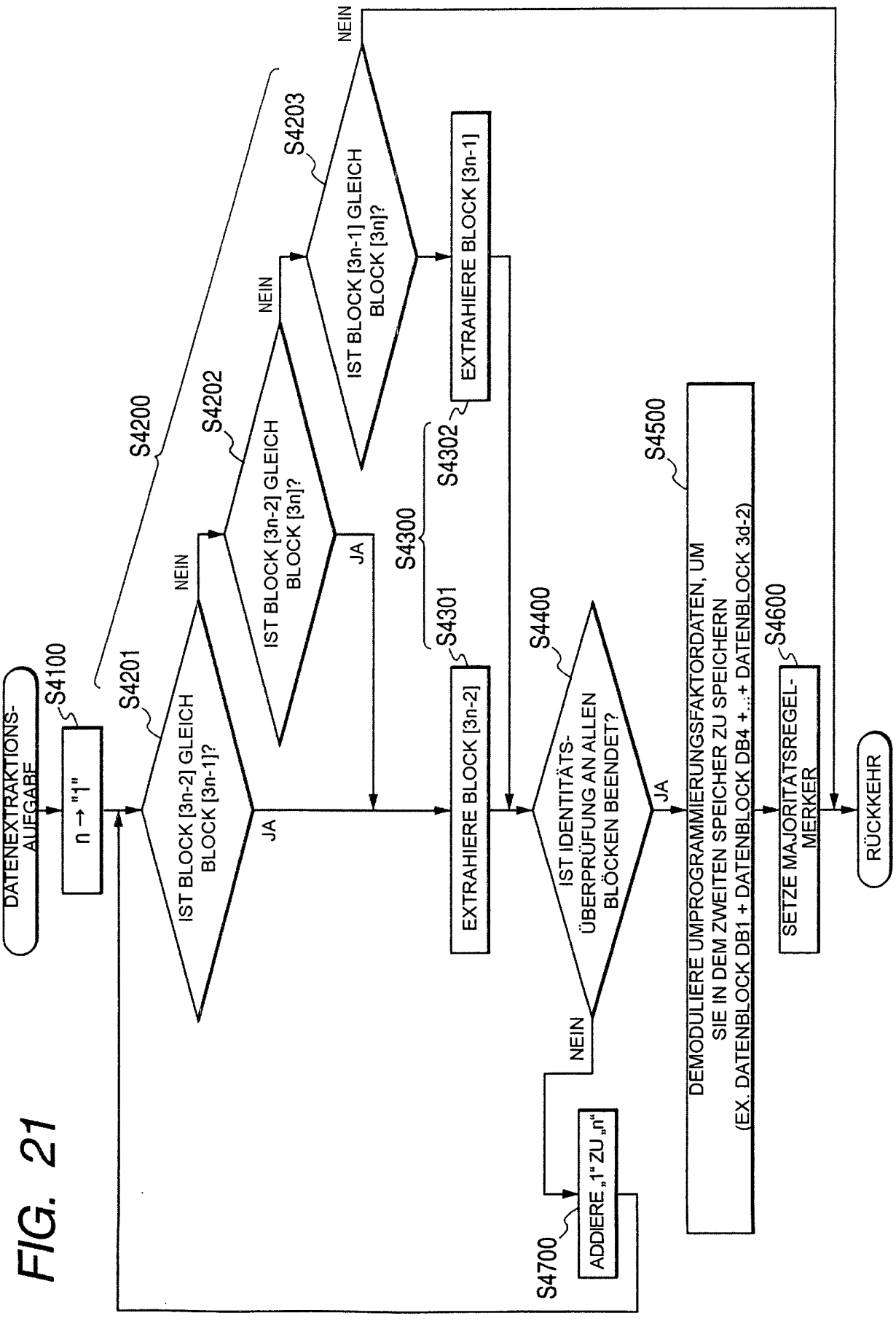

DATENEXTRAKTIONS-AUFGABE
S4100
n → "1"
S4200
S4201
IST BLOCK [3n-2] GLEICH BLOCK [3n-1]?
NEIN
JA
S4202
IST BLOCK [3n-2] GLEICH BLOCK [3n]?
NEIN
JA
S4203
IST BLOCK [3n-1] GLEICH BLOCK [3n]?
NEIN
S4300
S4301
EXTRAHIERE BLOCK [3n-2]
S4302
EXTRAHIERE BLOCK [3n-1]
S4400
IST IDENTITÄTS-ÜBERPRÜFUNG AN ALLEN BLÖCKEN BEENDET?
NEIN
JA
S4700
ADDIERE „1" ZU „n"
S4500
DEMODULIERE UMPROGRAMMIERUNGSFAKTORDATEN, UM SIE IN DEM ZWEITEN SPEICHER ZU SPEICHERN (EX. DATENBLOCK DB1 + DATENBLOCK DB4 +...+ DATENBLOCK 3d-2)
S4600
SETZE MAJORITÄTSREGEL-MERKER
RÜCKKEHR

FIG. 21

FIG. 22

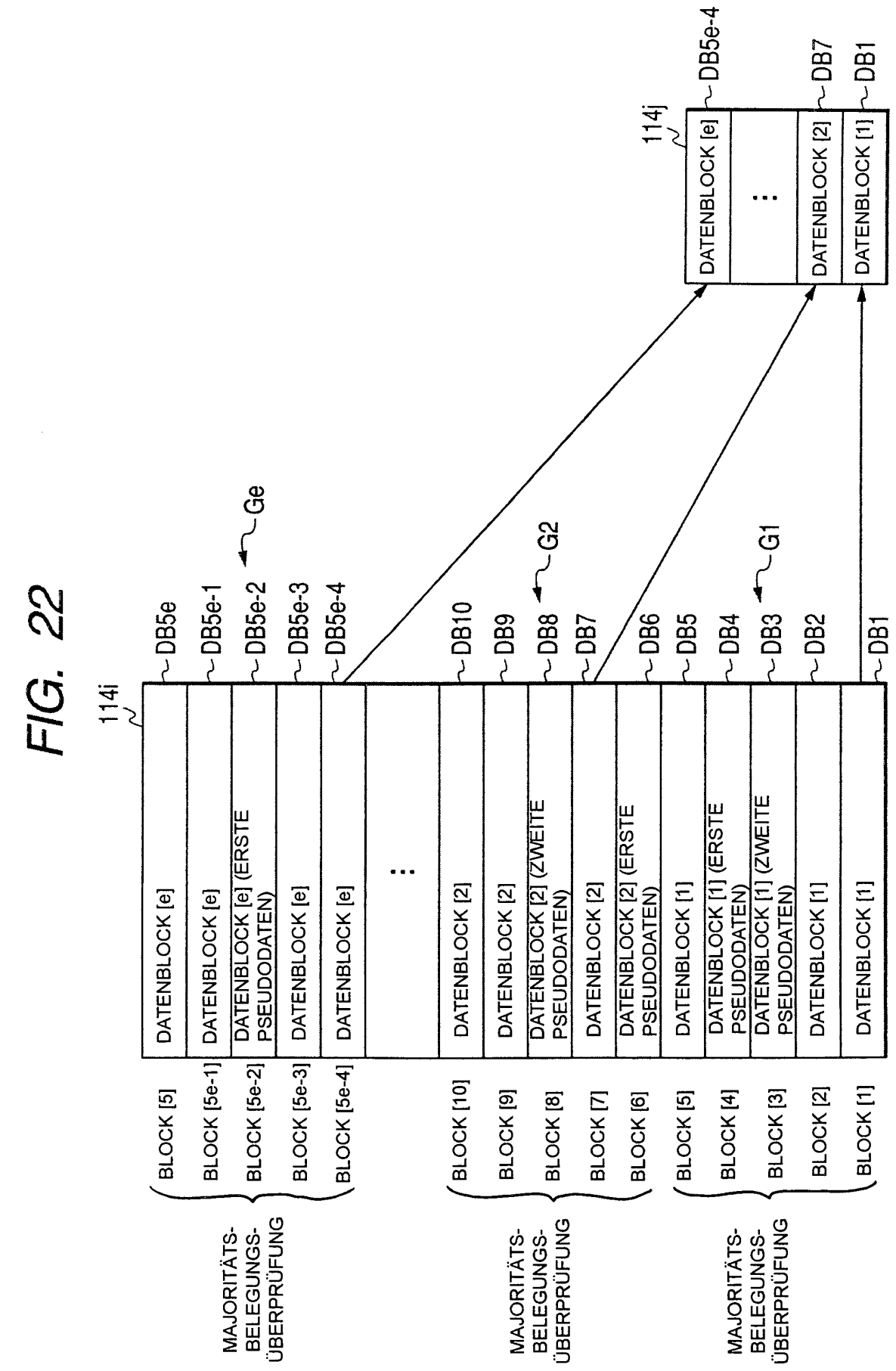

114i
DB5e
DB5e-1
DB5e-2
DB5e-3
DB5e-4
Ge
DB10
DB9
DB8
DB7
DB6
G2
DB5
DB4
DB3
DB2
DB1
G1
DATENBLOCK [e]
DATENBLOCK [e]
DATENBLOCK [e] (ERSTE PSEUDODATEN)
DATENBLOCK [e]
DATENBLOCK [e]
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [2] (ZWEITE PSEUDODATEN)
DATENBLOCK [2]
DATENBLOCK [2] (ERSTE PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1] (ERSTE PSEUDODATEN)
DATENBLOCK [1] (ZWEITE PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1]
BLOCK [5]
BLOCK [5e-1]
BLOCK [5e-2]
BLOCK [5e-3]
BLOCK [5e-4]
BLOCK [10]
BLOCK [9]
BLOCK [8]
BLOCK [7]
BLOCK [6]
BLOCK [5]
BLOCK [4]
BLOCK [3]
BLOCK [2]
BLOCK [1]
MAJORITÄTS-BELEGUNGS-ÜBERPRÜFUNG
MAJORITÄTS-BELEGUNGS-ÜBERPRÜFUNG
MAJORITÄTS-BELEGUNGS-ÜBERPRÜFUNG
114j
DB5e-4
DB7
DB1

FIG. 23

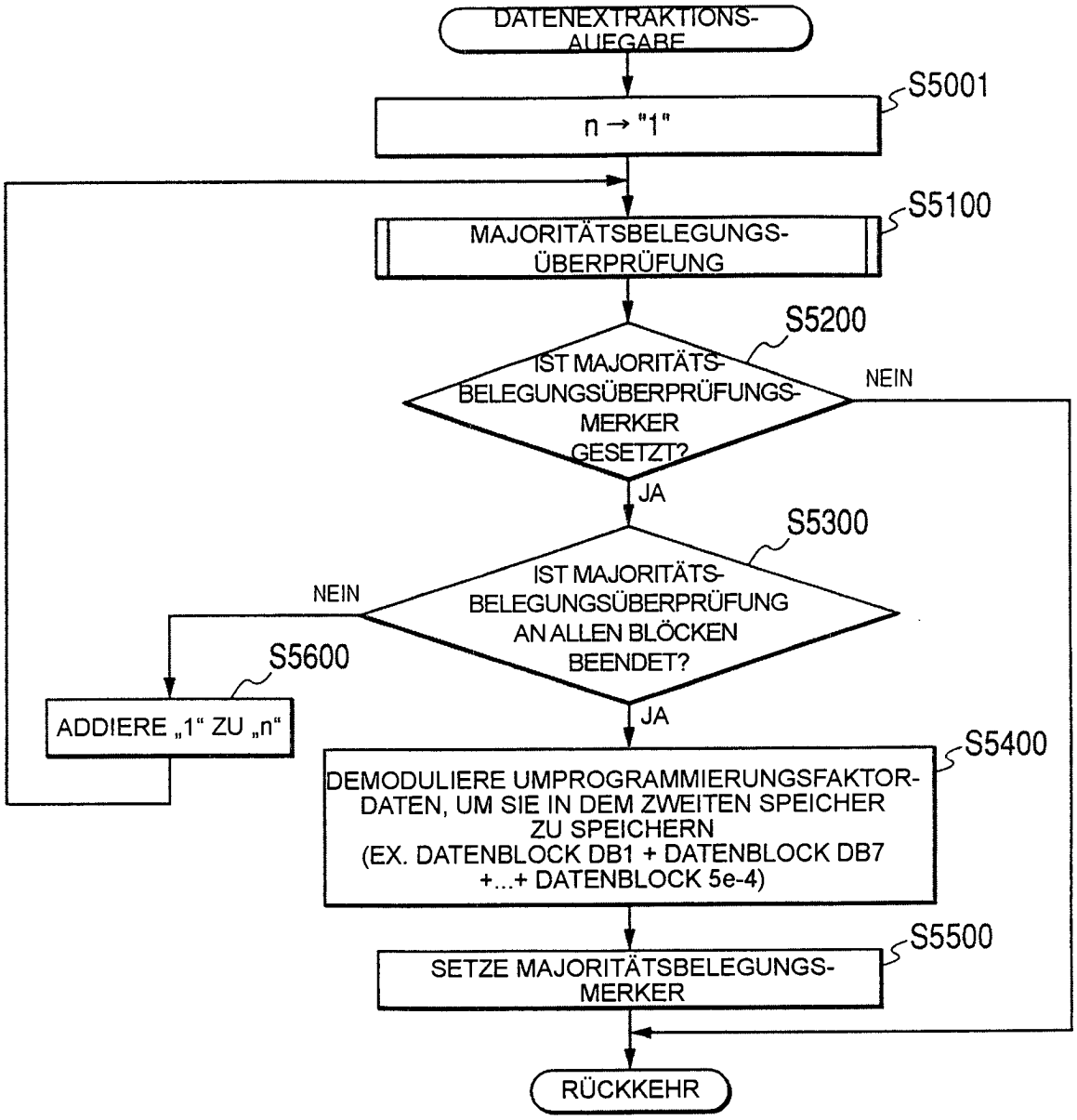

DATENEXTRAKTIONS-AUFGABE
S5001
n → "1"
S5100
MAJORITÄTSBELEGUNGS-ÜBERPRÜFUNG
S5200
IST MAJORITÄTS-BELEGUNGSÜBERPRÜFUNGS-MERKER GESETZT?
NEIN
JA
S5300
IST MAJORITÄTS-BELEGUNGSÜBERPRÜFUNG AN ALLEN BLÖCKEN BEENDET?
NEIN
S5600
ADDIERE „1" ZU „n"
JA
S5400
DEMODULIERE UMPROGRAMMIERUNGSFAKTOR-DATEN, UM SIE IN DEM ZWEITEN SPEICHER ZU SPEICHERN (EX. DATENBLOCK DB1 + DATENBLOCK DB7 +...+ DATENBLOCK 5e-4)
S5500
SETZE MAJORITÄTSBELEGUNGS-MERKER
RÜCKKEHR

FIG. 24

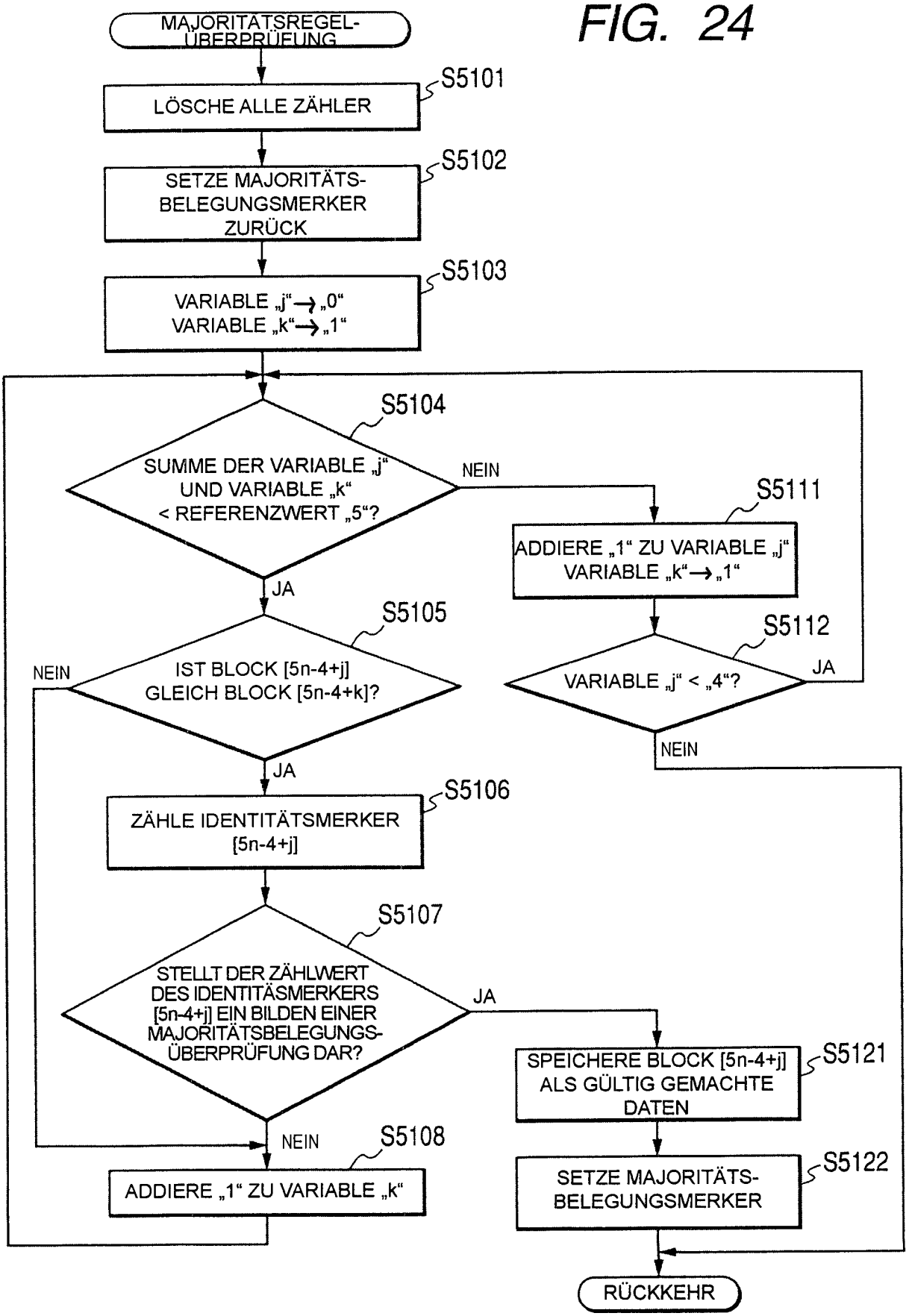
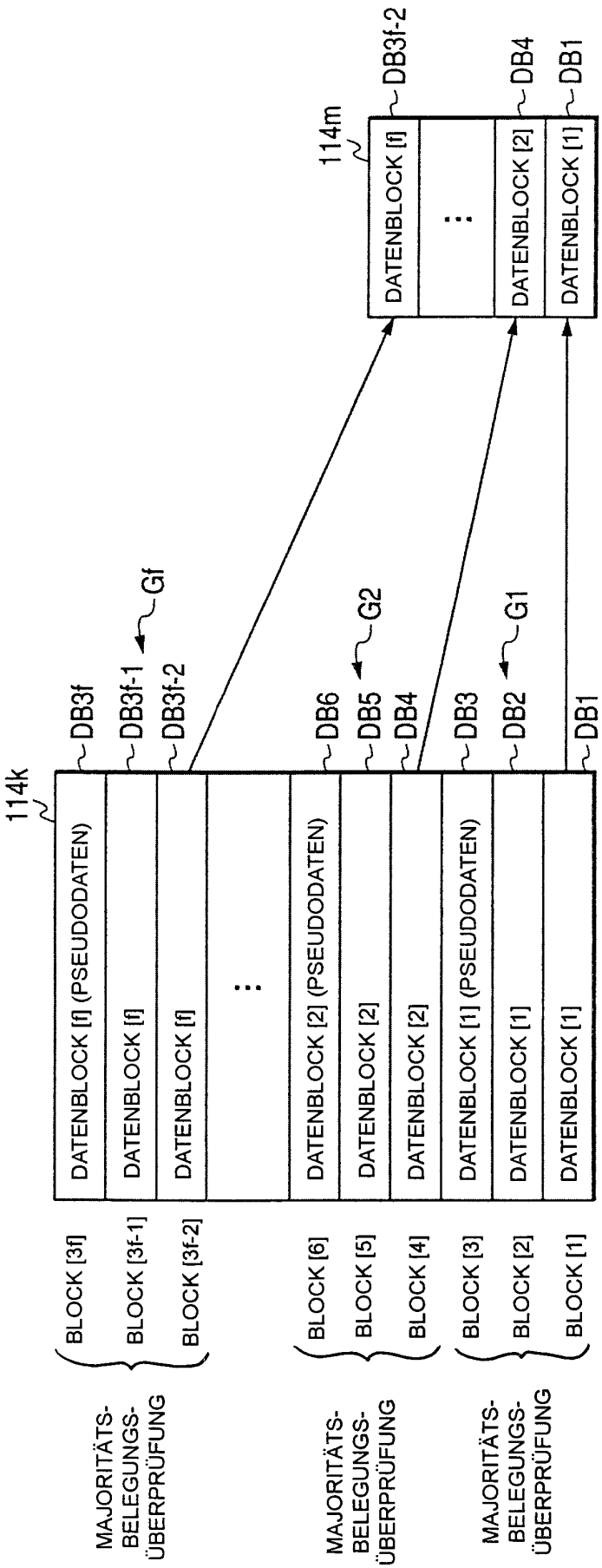

MAJORITÄTSREGEL-ÜBERPRÜFUNG
S5101
LÖSCHE ALLE ZÄHLER
S5102
SETZE MAJORITÄTS-BELEGUNGSMERKER ZURÜCK
S5103
VARIABLE „j“ → „0“
VARIABLE „k“ → „1“
S5104
SUMME DER VARIABLE „j“ UND VARIABLE „k“ < REFERENZWERT „5“?
NEIN
JA
S5111
ADDIERE „1“ ZU VARIABLE „j“
VARIABLE „k“ → „1“
S5112
VARIABLE „j“ < „4“?
JA
NEIN
S5105
IST BLOCK [5n-4+j] GLEICH BLOCK [5n-4+k]?
NEIN
JA
S5106
ZÄHLE IDENTITÄTSMERKER [5n-4+j]
S5107
STELLT DER ZÄHLWERT DES IDENTITÄSMERKERS [5n-4+j] EIN BILDEN EINER MAJORITÄTSBELEGUNGS-ÜBERPRÜFUNG DAR?
JA
NEIN
S5121
SPEICHERE BLOCK [5n-4+j] ALS GÜLTIG GEMACHTE DATEN
S5108
ADDIERE „1“ ZU VARIABLE „k“
S5122
SETZE MAJORITÄTS-BELEGUNGSMERKER
RÜCKKEHR

FIG. 25


114k
DATENBLOCK [f] (PSEUDODATEN)
DATENBLOCK [f]
DATENBLOCK [f]
DATENBLOCK [2] (PSEUDODATEN)
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [1] (PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1]
DB3f
DB3f-1
DB3f-2
DB6
DB5
DB4
DB3
DB2
DB1
Gf
G2
G1
114m
DATENBLOCK [f]
DATENBLOCK [2]
DATENBLOCK [1]
DB3f-2
DB4
DB1
BLOCK [3f]
BLOCK [3f-1]
BLOCK [3f-2]
BLOCK [6]
BLOCK [5]
BLOCK [4]
BLOCK [3]
BLOCK [2]
BLOCK [1]
MAJORITÄTS-BELEGUNGS-ÜBERPRÜFUNG
MAJORITÄTS-BELEGUNGS-ÜBERPRÜFUNG
MAJORITÄTS-BELEGUNGS-ÜBERPRÜFUNG

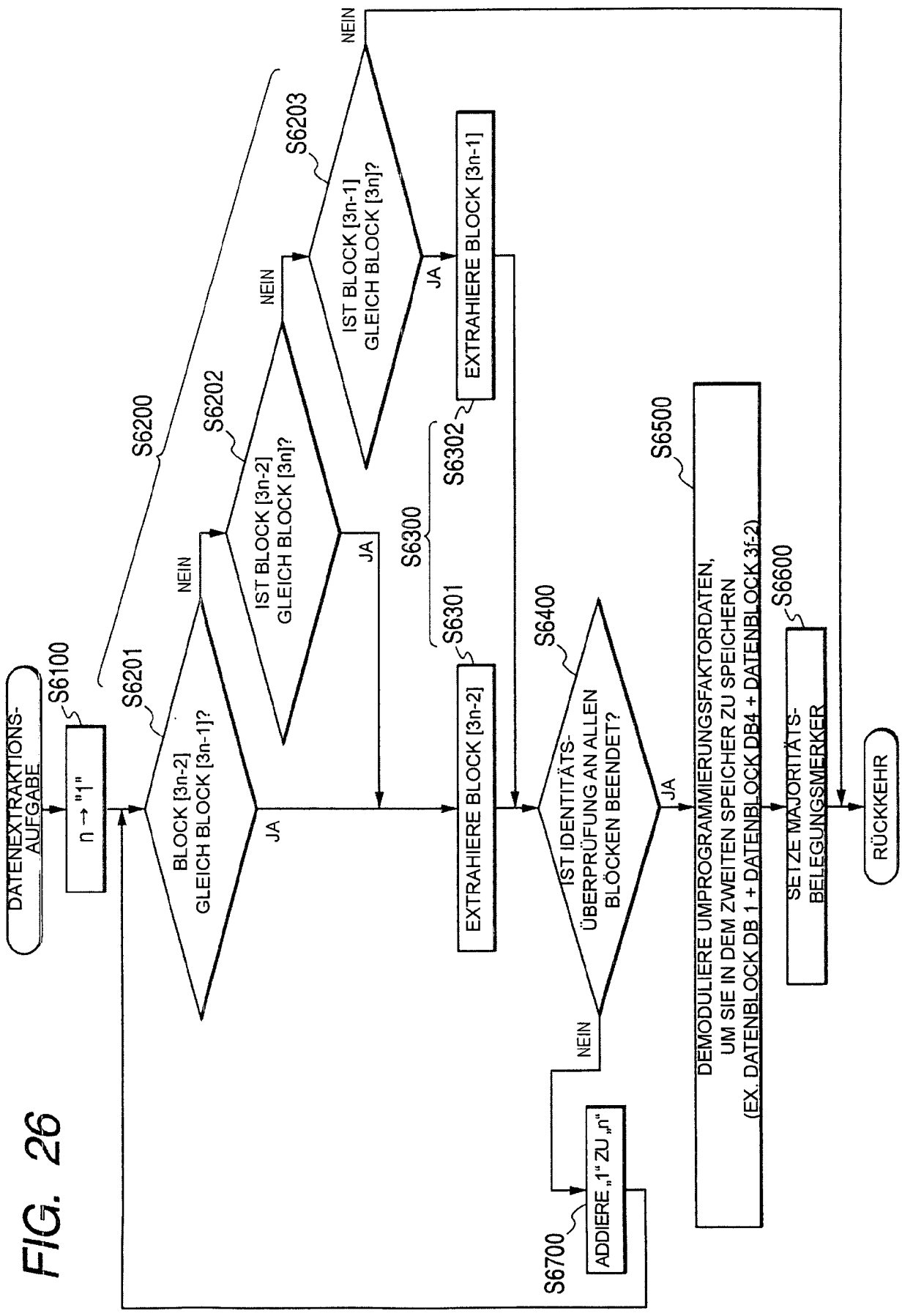

DATENEXTRAKTIONS-AUFGABE
S6100
n → "1"
S6200
S6201
BLOCK [3n-2] GLEICH BLOCK [3n-1]?
NEIN
JA
S6202
IST BLOCK [3n-2] GLEICH BLOCK [3n]?
NEIN
JA
S6203
IST BLOCK [3n-1] GLEICH BLOCK [3n]?
NEIN
JA
S6300
S6301
EXTRAHIERE BLOCK [3n-2]
S6302
EXTRAHIERE BLOCK [3n-1]
S6400
IST IDENTITÄTS-ÜBERPRÜFUNG AN ALLEN BLÖCKEN BEENDET?
NEIN
JA
S6500
DEMODULIERE UMPROGRAMMIERUNGSFAKTORDATEN, UM SIE IN DEM ZWEITEN SPEICHER ZU SPEICHERN (EX. DATENBLOCK DB 1 + DATENBLOCK DB4 + DATENBLOCK 3f-2)
S6600
SETZE MAJORITÄTS-BELEGUNGSMERKER
S6700
ADDIERE „1" ZU „n"
RÜCKKEHR

FIG. 26

FIG. 27

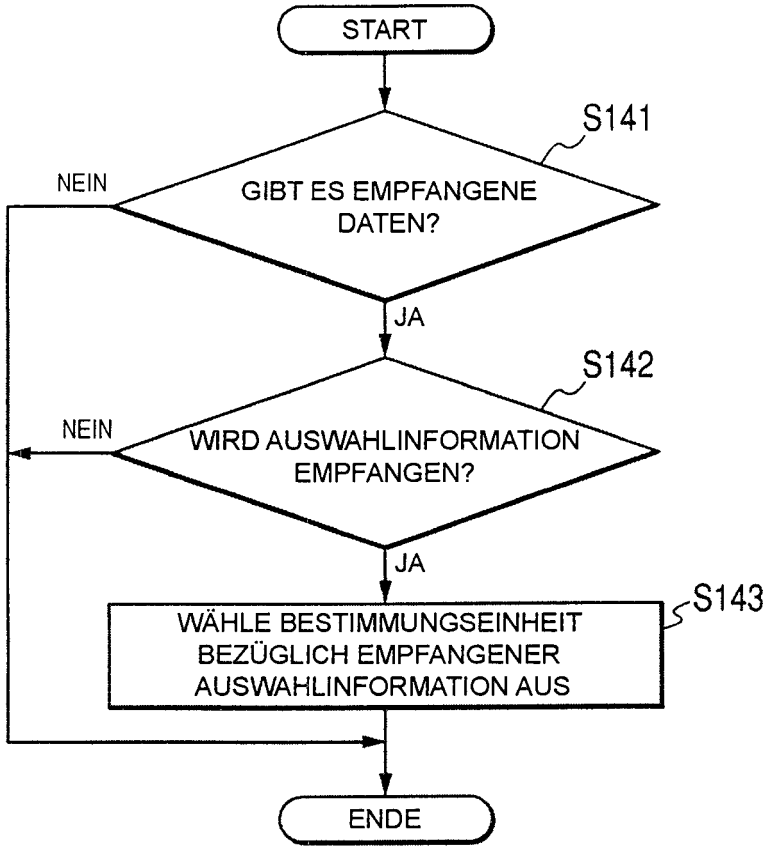


FIG. 28

T

IDENTIFIZIERER (ID)	BESTIMMUNGSEINHEIT
0x01	SPIEGELBESTIMMUNGSEINHEIT
0x02	IDENTITÄTSBESTIMMUNGSEINHEIT
0x03	MAJORITÄTSREGEL-BESTIMMUNGSEINHEIT
0x04	MAJORITÄTSBELEGUNGS-BESTIMMUNGSEINHEIT

FIG. 29

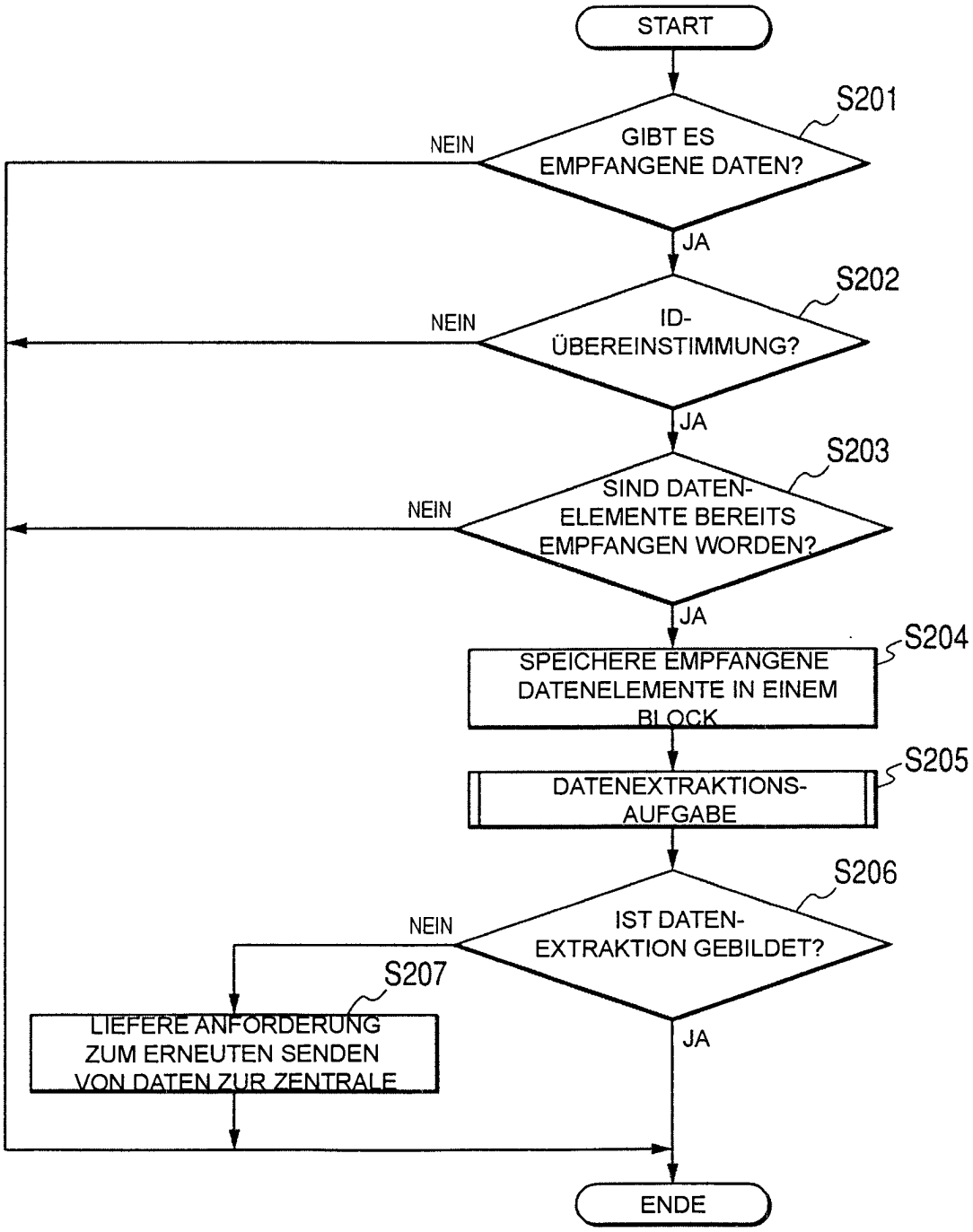

START
S201
GIBT ES EMPFANGENE DATEN?
NEIN
JA
S202
ID-ÜBEREINSTIMMUNG?
NEIN
JA
S203
SIND DATEN-ELEMENTE BEREITS EMPFANGEN WORDEN?
NEIN
JA
S204
SPEICHERE EMPFANGENE DATENELEMENTE IN EINEM BLOCK
S205
DATENEXTRAKTIONS-AUFGABE
S206
IST DATEN-EXTRAKTION GEBILDET?
NEIN
JA
S207
LIEFERE ANFORDERUNG ZUM ERNEUTEN SENDEN VON DATEN ZUR ZENTRALE
ENDE

FIG. 30

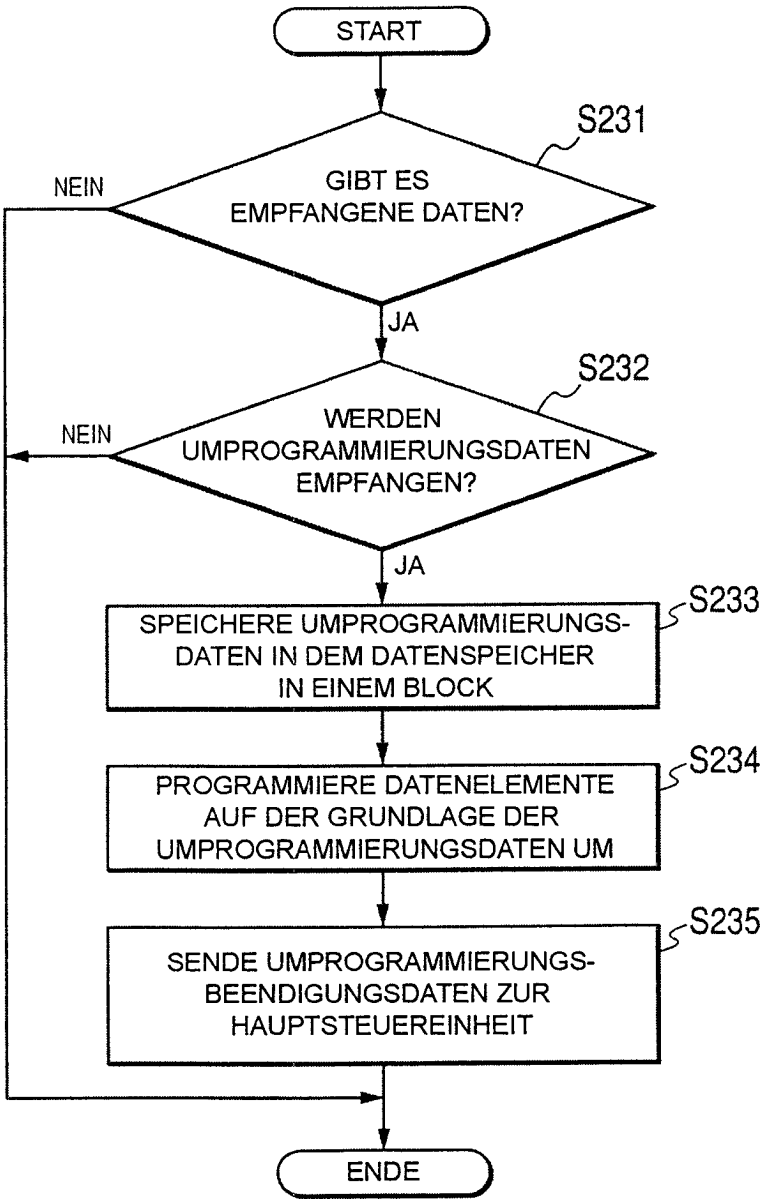

START
S231
GIBT ES EMPFANGENE DATEN?
NEIN
JA
S232
WERDEN UMPROGRAMMIERUNGSDATEN EMPFANGEN?
NEIN
JA
S233
SPEICHERE UMPROGRAMMIERUNGS-DATEN IN DEM DATENSPEICHER IN EINEM BLOCK
S234
PROGRAMMIERE DATENELEMENTE AUF DER GRUNDLAGE DER UMPROGRAMMIERUNGSDATEN UM
S235
SENDE UMPROGRAMMIERUNGS-BEENDIGUNGSDATEN ZUR HAUPTSTEUEREINHEIT
ENDE

FIG. 31

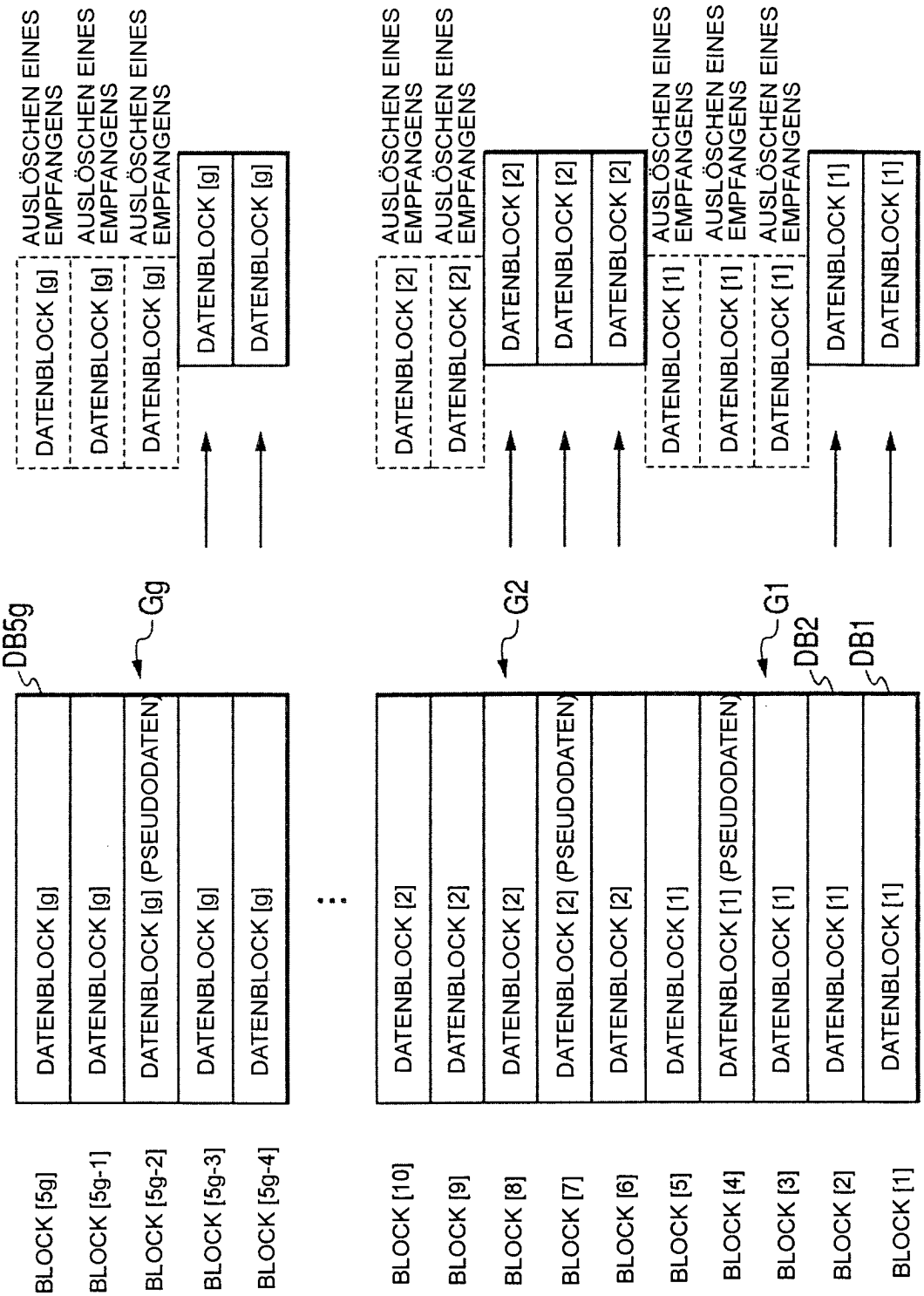
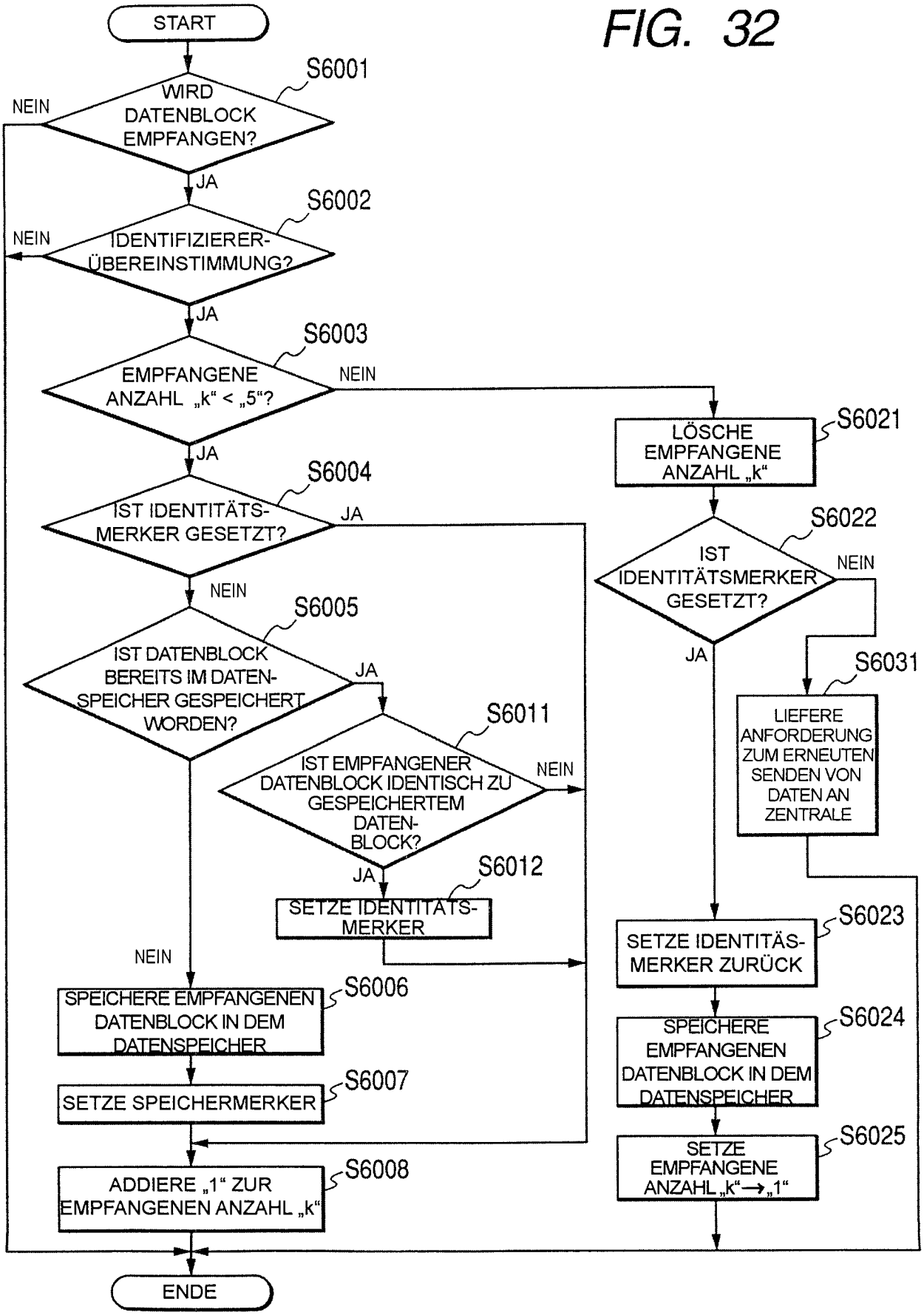

BLOCK [5g]
BLOCK [5g-1]
BLOCK [5g-2]
BLOCK [5g-3]
BLOCK [5g-4]
DATENBLOCK [g]
DATENBLOCK [g]
DATENBLOCK [g] (PSEUDODATEN)
DATENBLOCK [g]
DATENBLOCK [g]
DB5g
Gg
AUSLÖSCHEN EINES EMPFANGENS
AUSLÖSCHEN EINES EMPFANGENS
AUSLÖSCHEN EINES EMPFANGENS
DATENBLOCK [g]
DATENBLOCK [g]
DATENBLOCK [g]
DATENBLOCK [g]
DATENBLOCK [g]
BLOCK [10]
BLOCK [9]
BLOCK [8]
BLOCK [7]
BLOCK [6]
BLOCK [5]
BLOCK [4]
BLOCK [3]
BLOCK [2]
BLOCK [1]
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [2] (PSEUDODATEN)
DATENBLOCK [2]
DATENBLOCK [1]
DATENBLOCK [1] (PSEUDODATEN)
DATENBLOCK [1]
DATENBLOCK [1]
DATENBLOCK [1]
G2
G1
DB2
DB1
AUSLÖSCHEN EINES EMPFANGENS
AUSLÖSCHEN EINES EMPFANGENS
AUSLÖSCHEN EINES EMPFANGENS
AUSLÖSCHEN EINES EMPFANGENS
AUSLÖSCHEN EINES EMPFANGENS
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [1]
DATENBLOCK [1]
DATENBLOCK [1]
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [2]
DATENBLOCK [1]
DATENBLOCK [1]
DATENBLOCK [1]

FIG. 32


START
S6001
WIRD DATENBLOCK EMPFANGEN?
NEIN
JA
S6002
IDENTIFIZIERER-ÜBEREINSTIMMUNG?
NEIN
JA
S6003
EMPFANGENE ANZAHL „k“ < „5“?
NEIN
JA
S6004
IST IDENTITÄTS-MERKER GESETZT?
JA
NEIN
S6005
IST DATENBLOCK BEREITS IM DATEN-SPEICHER GESPEICHERT WORDEN?
JA
S6011
IST EMPFANGENER DATENBLOCK IDENTISCH ZU GESPEICHERTEM DATEN-BLOCK?
NEIN
JA
S6012
SETZE IDENTITÄTS-MERKER
NEIN
S6006
SPEICHERE EMPFANGENEN DATENBLOCK IN DEM DATENSPEICHER
S6007
SETZE SPEICHERMERKER
S6008
ADDIERE „1“ ZUR EMPFANGENEN ANZAHL „k“
ENDE
S6021
LÖSCHE EMPFANGENE ANZAHL „k“
S6022
IST IDENTITÄTSMERKER GESETZT?
NEIN
JA
S6031
LIEFERE ANFORDERUNG ZUM ERNEUTEN SENDEN VON DATEN AN ZENTRALE
S6023
SETZE IDENTITÄS-MERKER ZURÜCK
S6024
SPEICHERE EMPFANGENEN DATENBLOCK IN DEM DATENSPEICHER
S6025
SETZE EMPFANGENE ANZAHL „k“→„1“