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

(10) **DE 695 31 817 T2** 2004.07.15

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 719 689 B1**
(21) Deutsches Aktenzeichen: **695 31 817.9**
(96) Europäisches Aktenzeichen: **95 309 297.0**
(96) Europäischer Anmeldetag: **20.12.1995**
(97) Erstveröffentlichung durch das EPA: **03.07.1996**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **24.09.2003**
(47) Veröffentlichungstag im Patentblatt: **15.07.2004**

(51) Int Cl.[7]: **G06F 11/00**
G06F 11/16

(30) Unionspriorität:

32664594	**28.12.1994**	**JP**
18380395	**20.07.1995**	**JP**

(84) Benannte Vertragsstaaten:
DE, FR

(73) Patentinhaber:
Hitachi, Ltd., Tokio/Tokyo, JP

(74) Vertreter:
Strehl, Schübel-Hopf & Partner, 80538 München

(72) Erfinder:
Sato, Hiroshi, Hitachinaka-shi, Ibaraka 312, JP;
Kanekawa, Nobuyasu, Hitachi-shi, Ibaraki 316, JP;
Nohmi, Makoto, Hitachinaka-shi, Ibaraka 312, JP;
Tashiro, Korefumi, Hitachi-shi, Ibaraki 319-12, JP

(54) Bezeichnung: **Steuereinrichtung mit Fehlersicherheitsfunktion**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung

[0001] Die vorliegende Erfindung betrifft ein Steuergerät mit einer störungssicheren Funktion.

[0002] Mit der Entwicklung der Elektronik wurde eine fortschrittliche und komplizierte Steuerung möglich, um die Energie-(Kraftstoff-)-Ausnutzung, die Funktionsfähigkeit, den Komfort für die Fahrgäste und die Sicherheit sowie die Geschwindigkeit von Transporteinrichtungen wie Flugzeugen, Zügen und Kraftfahrzeugen zu erhöhen.

[0003] Andererseits ist bei solchen Transporteinrichtungen ein sicherer Betrieb erforderlich, und es ist unbedingt notwendig, daß die Steuergeräte zuverlässig sind und gegen Fehler abgesichert (wenn ein Fehler auftritt, darf von einer gefährlichen Stelle kein Ausgangssignal ausgehen).

[0004] Für den Zugbetrieb wird die Sicherheit zum Beispiel durch ein übergeordnetes Überwachungssystem wie die ATC-Einrichtung und die ATS-Einrichtung sichergestellt.

[0005] Wie das Beispiel der ATP-Einrichtung für das Überwachungssystem von Super-Express-Zügen zeigt, werden die Zuverlässigkeit und die Fehlersicherheit überall hoch bewertet.

[0006] Die ATP- und die ATS-Einrichtung für solche Züge umfaßt eine Steuerschaltung und ein Relais mit LSIs mit einer Selbstprüffunktion. Das in diesen Einrichtungen verwendete Hauptsignal ist ein Frequenzsignal, bei dem sich der logische Pegel abwechselnd von H auf L ändert (im folgenden alternierendes Signal genannt).

[0007] Die Steuerschaltung vergleicht und verarbeitet beim Erhalt dieses ATP-Signals das ATP-Sollgeschwindigkeitssignal vom Boden mit dem tatsächlichen Geschwindigkeitssignal des Zuges, steuert mit einem Geschwindigkeitskontrollsignal die Bremskraft, das heißt gibt ein Sollbremssignal entsprechend der Abweichung zwischen dem ATC-Signal und dem tatsächlichen Geschwindigkeitssignal aus, und steuert so die Geschwindigkeit des Zuges.

[0008] Eine Vorrichtung zum Erzeugen eines solchen alternierenden Signals ist in der japanischen Patent-Offenlegungsschrift 57-62702 beschrieben.

[0009] Das US-Patent 4 763 267 beschreibt eine Steuerung mit zwei Prozessoren und einem fehlersicheren Aufbau.

[0010] Es ist ein System zum Übertragen von Informationen über den Zugbetrieb (zum Beispiel die Geschwindigkeit und Position eines Zuges) zwischen einem zentralen Steuerund Befehlsraum und dem Zug und zum Austausch von Informationen zwischen Zügen im Aufbau, um dadurch die Transportdienste zu verbessern und die Steuerung einer dichteren Zugfolge sowie einen schnelleren Zugbetrieb zu ermöglichen.

[0011] Um die dafür gestellten Anforderungen zu erfüllen, reichen die gegenwärtig im Gebrauch befindlichen Steuergeräte und LSIs allein hinsichtlich der Verarbeitungsgeschwindigkeit und der Datenspeicherkapazität nicht aus, weshalb es erforderlich ist, viele periphere Schaltungen hinzuzufügen, womit das Steuergerät jedoch kompliziert wird.

[0012] Die Halbleitertechnologie hat allerdings neulich bemerkenswerte Fortschritte gemacht, und es wurde eine hohe Integration und eine hohe Geschwindigkeit realisiert, so daß ein einziger LSI-Chip eine Steuerschaltung mit verschiedenen Funktionen aufnehmen kann.

[0013] Bei einem entsprechenden LSI-Aufbau besteht jedoch die Gefahr, daß bei einem Fehler im LSI aufgrund eines Kontakts zwischeninneren Leiterbahnen ein falsches Steuersignal ausgegeben wird oder daß durch die Streukapazität zwischen einer offenen Leiterbahn und der benachbarten Leiterbahn ein induziertes Signal ausgegeben wird, wobei das ausgegebene Steuersignal als normal betrachtet wird.

[0014] Die Gebrauchsumgebung für einen LSI ist außerdem bei der Zugsteuerung wesentlich schwieriger als bei anderen LSI-Anwendungen.

[0015] Es ergeben sich ernste Probleme, wenn aufgrund eines inneren Fehlkontaktes oder einer Leiterunterbrechung bei der Herstellung eines LSI oder aufgrund eines Fehlkontaktes oder einer Leitungsunterbrechung im Gebrauch ein falsches Signal ausgegeben wird oder zum Beispiel ein Bremsbefehlsignal, das eine wesentlicher Punkt einer Geschwindigkeitssteuerung ist, nicht ausgegeben wird. Schwere Unfälle können die Folge sein.

[0016] Es ist erforderlich, daß eine LSI für zukünftige verschiedene Steueraufgaben eine Steuerschaltung bildet, die die Funktionen für die Steuerung erfüllt und eine Selbstprüffunktion zum Erfassen eines Fehlers in der LSI selbst und eine störungssichere Steuerung beim Auftreten eines Fehlers aufweist, wobei die Steuerung immer auf der sicheren Seite bleiben soll.

[0017] Aufgabe der vorliegenden Erfindung ist es daher, ein Steuergerät zu schaffen, bei dem die Selbstprüffunktion zum Erfassen von Fehlern verbessert ist und bei der eine störungssichere Funktion für eine Steuerung auf der sicheren Seite möglich ist, wenn ein Fehler auftritt.

[0018] Die Aspekte der vorliegenden Erfindung sind in den anhängenden Patentansprüchen genannt.

[0019] Andere Aufgaben und Aspekte der Erfindung gehen aus der folgenden Beschreibung von Ausführungsformen in Verbindung mit den beiliegenden Zeichnungen hervor. Es zeigen:

[0020] **Fig.** 1 eine Darstellung eines Zuges mit einem ATP-System mit einem erfindungsgemäßen Steuergerät;

[0021] **Fig.** 2 eine Darstellung einer Ausführungsform des erfindungsgemäßen Steuergeräts;

[0022] **Fig.** 3 ein Diagramm zur Datenkonfiguration;

[0023] **Fig.** 4 eine Darstellung des Prinzips für einen Frequenzvergleichsvorgang;

[0024] **Fig.** 5 die Anzahl von Operationen und Fehlersignal-Wellenformen;

[0025] **Fig.** 6 eine Darstellung eines Frequenzver-

gleichers;
[0026] **Fig.** 7 das Zeitdiagramm für den Frequenzvergleicher;
[0027] **Fig.** 8 eine Darstellung einer anderen Ausführungsform der vorliegenden Erfindung;
[0028] **Fig.** 9 ein Diagramm für die Datenkonfiguration in den einzelnen Speichern;
[0029] **Fig.** 10 eine Darstellung von Fehlererfassungssignalen und einer Speicherumschaltoperation;
[0030] **Fig.** 11 eine Darstellung eines Frequenzwandlers;
[0031] **Fig.** 12 eine Darstellung einer Frequenzumwandlungsoperation;
[0032] **Fig.** 13 eine Darstellung eines Frequenzkomparators;
[0033] **Fig.** 14 ein Zeitdiagramm für eine Frequenzvergleichsoperation;
[0034] **Fig.** 15 eine Darstellung für eine Multifrequenzvergleichsoperation;
[0035] **Fig.** 16 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung;
[0036] **Fig.** 17 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung;
[0037] **Fig.** 18 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung;
[0038] **Fig.** 19 eine Darstellung des LSI-Layouts bei der vorliegenden Erfindung;
[0039] **Fig.** 20 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung;
[0040] **Fig.** 21 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung;
[0041] **Fig.** 22 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung;
[0042] **Fig.** 23 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung;
[0043] **Fig.** 24 eine Darstellung einer weiteren Ausführungsform der vorliegenden Erfindung; und
[0044] **Fig.** 25 eine Darstellung eines Beispiels der Programmierung bei der vorliegenden Erfindung.
[0045] Es wird im folgenden der Fall beschrieben, daß die vorliegende Erfindung bei einer automatischen Zugsteuerung ATP angewendet wird.
[0046] Die ATP sorgt entsprechend der Abweichung zwischen einem ATP-Sollgeschwindigkeitssignal vom Boden und der tatsächlichen Geschwindigkeit des Zuges (im folgenden als elektrisches Motorfahrzeug bezeichnet) zu diesem Zeitpunkt für eine Bremskraft und steuert die laufende Geschwindigkeit des elektrischen Motorfahrzeugs so, daß sie kleiner ist als die Sollgeschwindigkeit.
[0047] **Fig.** 1 zeigt ein Blockdiagramm für ein ATP-System bei Anwendung der vorliegenden Erfindung.
[0048] In der **Fig.** 1 bezeichnet das Bezugszeichen **1** die Schienen, auf denen ein elektrisches Motorfahrzeug läuft, **23** den Rumpf des elektrischen Motorfahrzeugs, **2** ein Rad des elektrischen Motorfahrzeugs, **3** einen Drehzahlgenerator, der an der Achse des Rades **2** angebracht ist, um die Drehzahl für den Drehzahlgenerator zu erfassen, **4** einen Wellenformer (oder Signalkonverter) zum Formen der Wellenform der Ausgangsspannung des Drehzahlgenerators **3** und zur Ausgabe eines Geschwindigkeitsfrequenzsignals **5**, das proportional ist zur Geschwindigkeit des elektrischen Motorfahrzeugs, **6** eine Antenne zur Aufnahme eines frequenzmodulierten ATP-Sollgeschwindigkeitssignals vom Boden, **7** einen Fahrzeugempfänger zum Demodulieren des von der Antenne **6** aufgenommenen frequenzmodulierten ATP-Sollgeschwindigkeitssignals und zum Formen und Verstärken der Wellenform, **8** ein ATP-Sollgeschwindigkeitssignal, das das Ausgangssignal des Fahrzeugempfängers **7** ist, **9** eine ATP-Einrichtung zum Steuern der Geschwindigkeit des elektrischen Motorfahrzeugs gemäß dem ATP-Sollgeschwindigkeitssignal und der tatsächlichen Geschwindigkeit des elektrischen Motorfahrzeugs, und **22** eine Bremseinrichtung zum Aufnehmen eines Bremssollsignals **21** von der ATP-Einrichtung und zur Abgabe einer tatsächlichen Bremskraft an das elektrische Motorfahrzeug.
[0049] Mit diesem Aufbau wird die Geschwindigkeit des elektrischen Motorfahrzeugs so gesteuert, daß die Abweichung zwischen der ATP-Sollgeschwindigkeit und der tatsächlichen Geschwindigkeit des elektrischen Motorfahrzeugs zu Null wird, und daß verhindert wird, daß die tatsächliche Geschwindigkeit des elektrischen Motorfahrzeugs die ATP-Sollgeschwindigkeit übersteigt.
[0050] Anhand der **Fig.** 2 wird die störungssichere Technologie des Steuergeräts der genannten ATP-Einrichtung erläutert.
[0051] Das Bezugszeichen **1800** bezeichnet das Steuergerät, **1810** einen ersten Prozessor zur Eingabe und Verarbeitung von ersten Eingangsdaten und zur Ausgabe von ersten Ausgangsdaten **1860** und eines ersten Erfassungssignals **1880**, **1820** einen zweiten Prozessor zur Eingabe und Verarbeitung von zweiten Eingangsdaten und zur Ausgabe von zweiten Ausgangsdaten **1870** und eines zweiten Erfassungssignals **1890** mit invertiertem Vorzeichen, **1830** einen Signalinverter zum Invertieren des Erfassungssignals, das vom zweiten Prozessor verarbeitet wurde, **1900** einen ersten Transmitter zum Übertragen des ersten Erfassungssignals **1880**, das vom ersten Prozessor **1810** ausgegeben wird, an den zweiten Prozessor **1820**, und **1910** einen zweiten Transmitter zum Übertragen des zweiten Erfassungssignals **1890**, das vom zweiten Prozessor ausgegeben wird, über den Signalinverter **1830** an den ersten Prozessor **1810**.
[0052] Die ersten Eingangsdaten und die zweiten Eingangsdaten sind Daten, die von einem Mikrocomputer erzeugt und ausgegeben werden oder die aus einem Speicher ausgelesen werden.
[0053] Hinsichtlich der ersten Eingangsdaten gibt es Fälle, daß es Daten zum Verarbeiten von ersten Ausgangsdaten und Prüfdaten zum Prüfen der obigen Daten sind, und es gibt Fälle, daß es Daten ohne Prüfdaten sind, wobei die Prüfdaten dann vom ersten

Prozessor erzeugt werden, und es wird ein erstes Erfassungssignal erzeugt.

[0054] Hinsichtlich der zweiten Eingangsdaten gibt es Fälle, daß der Dateninhalt der gleiche ist wie bei den ersten Eingangsdaten, und es gibt Fälle, daß der Dateninhalt ein anderer ist wie bei den ersten Eingangsdaten.

[0055] Bei den zweiten Eingangsdaten gibt es Fälle, daß sie wie die ersten Eingangsdaten Prüfdaten enthalten, und es gibt Fälle, daß sie keine Prüfdaten enthalten. Wenn die zweiten Eingangsdaten keine Prüfdaten enthalten, gibt es Fälle, daß die Prüfdaten vom zweiten Prozessor auf die gleiche Weise wie bei den ersten Eingangsdaten erzeugt werden, und Fälle, bei denen vom zweiten Prozessor keine Prüfdaten erzeugt werden.

[0056] Bei der vorliegenden Ausführungsform sind außerdem die ersten Eingangsdaten und die zweiten Eingangsdaten unabhängig voneinander. Es können jedoch leicht die gleichen Eingangsdaten in die beiden Prozessoren eingegeben werden.

[0057] Der erste Prozessor und der zweite Prozessor verarbeiten die ersten bzw. zweiten Eingangsdaten, die sequentiell durch einen seriell arbeitenden Ringtypprozessor eingegeben werden, oder beide Daten werden parallel in einen Mikrocomputer eingegeben und parallel verarbeitet.

[0058] Das erste und das zweite Erfassungssignal zeigen an, ob der erste Prozessor bzw. der zweite Prozessor normal oder unnormal arbeiten. Wenn in der vorgegebenen Zeitspanne alternierende Signale ausgegeben werden, zeigen die Erfassungssignale an, daß die Prozessoren normal arbeiten.

[0059] Diese Erfassungssignale können als Erfassungssignale dafür verwendet werden, ob die Prüfergebnisse für den Inhalt der Eingangsdaten richtig oder falsch sind.

[0060] Durch die ersten Ausgangsdaten und die zweiten Ausgangsdaten wird eine andere Vorrichtung gesteuert.

[0061] Vor der Ausgabe an die andere Vorrichtung wird durch einen Ausgangssignalanteil zum Erfassen einer Übereinstimmung in den ersten Ausgangsdaten und den zweiten Ausgangsdaten oder die Ausgabe von endgültigen Ausgangsdaten unter Verwendung des ersten oder zweiten Erfassungssignals eine weitere, zu steuernde Vorrichtung gesteuert.

[0062] Die Arbeitsweise der Steuerung mit dem beschriebenen Aufbau ist die folgende:

[0063] Das erste Beispiel für die Verarbeitung ist der Fall, daß zuerst das erste Erfassungssignal ausgegeben wird und durch wiederholte Inversion dieses Signals alternierende Signale erzeugt werden.

[0064] Die ersten Eingangsdaten werden in den ersten Prozessor eingegeben und dort verarbeitet und dann die ersten Ausgangsdaten und das erste Erfassungssignal ausgegeben. Das erste Erfassungssignal wird in den zweiten Prozessor eingegeben und dort zusammen mit den zweiten Eingangsdaten verarbeitet, und es werden die zweiten Ausgangsdaten und das zweite Erfassungssignal erzeugt und ausgegeben.

[0065] Das Vorzeichen des zweiten Erfassungssignals wird vom Signalinverter invertiert und das sich ergebende Erfassungssignal in den ersten Prozessor eingegeben. Wenn das vom ersten Prozessor verarbeitete erste Erfassungssignal ein Signal auf dem hohen Pegel (im folgenden mit H bezeichnet) ist, wird das vom zweiten Prozessor verarbeitete zweite Erfassungssignal auch ein H-Pegel-Signal, und wenn das erste Erfassungssignal des ersten Prozessors ein Signal auf niedrigem Pegel (im folgenden mit L bezeichnet) ist, wird das vorn zweiten Prozessor verarbeitete zweite Erfassungssignal ebenfalls ein L-Pegel-Signal.

[0066] Das Vorzeichen des zweiten Erfassungssignals auf dem Pegel H wird durch den Signalinverter invertiert, und das sich ergebende Erfassungssignal mit dem Pegel L wird in den ersten Prozessor eingegeben.

[0067] Wenn das vom ersten Prozessor erarbeitete erste Erfassungssignal ein L-Pegel-Signal ist, wird das L-Pegel-Signal in den zweiten Prozessor eingegeben, und das Erfassungssignal des zweiten Prozessors wird zu einen L-Pegel-Signal. Da jedoch das Vorzeichen dieses Signals durch den Signalinverter invertiert wird, wird das zweite Erfassungssignal als Signal auf dem Pegel H dem ersten Prozessor zugeführt, und wenn der erste Prozessor und der zweite Prozessor normal arbeiten, werden die ersten und zweiten Erfassungssignale zu Signalen, die abwechselnd auf den Pegeln H, L, H, ... liegen.

[0068] Die beschriebene Verarbeitung des Erfassungssignals im ersten und zweiten Prozessor führt zu der Ausgabe eines Signals, das den gleichen Pegel hat wie das eingegebene Signal. Das heißt, wenn ein erstes Erfassungssignal mit dem Pegel H oder L zuerst am ersten Prozessor ausgegeben wird, werden die ersten und zweiten Erfassungssignale zu alternierenden Signalen, solange der erste und der zweite Prozessor normal arbeiten.

[0069] Das zweite Beispiel für die Verarbeitung ist der Fall, daß zwischen dem ersten Erfassungssignal, das vom ersten Prozessor ausgegeben wird, und den Prüfdaten der zweiten Eingangsdaten (oder den aus den zweiten Eingangsdaten erzeugten Prüfdaten) eine logische Verknüpfung ausgeführt wird und vom zweiten Prozessor ein Erfassungssignal ausgegeben wird.

[0070] Dadurch wird mit der vorliegenden Erfindung nicht nur festgestellt, ob der erste und der zweite Prozessor normal arbeiten, sondern es kann auch geprüft werden, ob die ersten oder die zweiten Eingangsdaten normal oder unnormal sind.

[0071] Wenn das erste Erfassungssignal nicht zu den Prüfdaten der zweiten Eingangsdaten paßt, wird vom zweiten Prozessor als Erfassungssignal ein Signal ausgegeben, das auf einem anderen Pegel ist als das erste Erfassungssignal. Dadurch wird das zweite Erfassungssignal nach dem Signalinverter zu einem

Signal, das auf dem gleichen Pegel ist wie das erste Erfassungssignal, so daß kein alternierendes Signal ausgegeben wird.

[0072] Das gleiche gilt für das zweite Erfassungssignal des ersten Prozessors und die Prüfdaten der ersten Eingangsdaten. Bei dieser Verarbeitung wird ein Aufbau mit einer logischen Einheit zur Ausführung einer logischen Verknüpfung im Prozessor verwendet.

[0073] Es wird der Aufbau der ATP-Einrichtung **9** der **Fig.** 1 beschrieben, die die genannten Eigenschaften für die Steuerung in einem automatischen Zugsteuersystem aufweist.

[0074] Die ATP-Einrichtung **9** umfaßt einen ersten Mikrocomputer **10** zum Umwandeln des ATP-Sollgeschwindigkeitssignals **8** in Steuerdaten **12** des ersten Mikrocomputers, einen zweiten Mikrocomputer **11** zum Umwandeln des ATP-Sollgeschwindigkeitssignals **8** in Steuerdaten **13** des zweiten Mikrocomputers durch Ausführen des gleichen Prozesses wie beim ersten Mikrocomputer, eine Logikschaltung **15** des Systems A zum Ausführen von logischen Verknüpfungen zwischen den Steuerdaten **12** des ersten Mikrocomputers **10** und einem Geschwindigkeitsfrequenzsignal **5** und zur Ausgabe eines logischen Ausgangssignals **17** für das System A, eine Logikschaltung **16** des Systems B zum Ausführen von logischen Verknüpfungen zwischen den Steuerdaten **13** des zweiten Mikrocomputers **11** und dem Geschwindigkeitsfrequenzsignal **5** und zur Ausgabe eines logischen Ausgangssignals **18** für das System B, und ein Steuergerät **14** mit einem doppelten logischen Aufbau aus der Logikschaltung **15** des Systems A und der Logikschaltung **16** des Systems B.

[0075] Die Logikschaltung **15** des Systems A umfaßt einen Frequenzwandler **150** zum Umwandeln der Steuerdaten **12** des ersten Mikrocomputers in ein ATP-Geschwindigkeitssollfrequenz(signal) **151**, einen zweiten Transmitter zum Übertragen eines ATP-Geschwindigkeitssollfrequenzsignals **2150** zu einem dritten Prozessor **2030**, einen dritten Transmitter **2200** zum Übertragen eines dritten Erfassungssignals **2160** zu einem vierten Prozessor **2040** und einen vierten Transmitter **2210** zum Übertragen eines vierten Erfassungssignals **2170**, das vom vierten Prozessor **2040** ausgegeben wird, über einen Vorzeicheninverter **2050** zu einem ersten Prozessor **2010**.

[0076] Die ersten Eingangsdaten **2060**, die zweiten Eingangsdaten **2070**, die dritten Eingangsdaten **2080** und die vierten Eingangsdaten **2090** sind Daten, die vom Mikrocomputer erzeugt und ausgegeben werden oder die aus dem Speicher ausgelesen werden.

[0077] Bei den ersten Eingangsdaten **2060** gibt des Fälle, in denen es Daten zum Erarbeiten der ersten Ausgangsdaten **2100** und Prüfdaten zum Prüfen der obigen Daten sind, und Fälle, in denen es keine Prüfdaten gibt, wobei die Prüfdaten dann durch eine Verarbeitung des ersten Prozessors **2010** erzeugt werden, und es wird ein erstes Erfassungssignal **2140** erzeugt.

[0078] Bei den zweiten Eingangsdaten **2070** gibt des Fälle, in denen es Daten zum Erarbeiten der zweiten Ausgangsdaten **2110** und Prüfdaten zum Prüfen der obigen Daten sind, und Fälle, in denen es keine Prüfdaten gibt, wobei die Prüfdaten dann durch eine Verarbeitung des zweiten Prozessors **2020** erzeugt werden, und es wird das zweite Erfassungssignal **2150** erzeugt.

[0079] Bei den dritten Eingangsdaten **2080** gibt des Fälle, in denen es Daten zum Erarbeiten der dritten Ausgangsdaten **2120** und Prüfdaten zum Prüfen der obigen Daten sind, und Fälle, in denen es keine Prüfdaten gibt, wobei die Prüfdaten dann durch eine Verarbeitung des dritten Prozessors **2030** erzeugt werden, und es wird ein drittes Erfassungssignal **2160** erzeugt.

[0080] Bei den vierten Eingangsdaten **2090** gibt des Fälle, in denen es Daten zum Erarbeiten der vierten Ausgangsdaten **2130** und Prüfdaten zum Prüfen der obigen Daten sind, und Fälle, in denen es keine Prüfdaten gibt, wobei die Prüfdaten dann durch eine Verarbeitung des vierten Prozessors **2040** erzeugt werden, und es wird das vierte Erfassungssignal **2170** erzeugt.

[0081] Der erste Prozessor **2010** verarbeitet die ersten Eingangsdaten **2060**, die sequentiell durch einen zugeordneten, seriell arbeitenden Ringtypprozessor eingegeben werden, oder die Daten werden parallel in den Mikrocomputer eingegeben und parallel verarbeitet.

[0082] Der zweite Prozessor **2020** verarbeitet die zweiten Eingangsdaten **2070**, die sequentiell durch einen zugeordneten, seriell arbeitenden Ringtypprozessor eingegeben werden, oder die Daten werden parallel in den Mikrocomputer eingegeben und parallel verarbeitet.

[0083] Der dritte Prozessor **2030** verarbeitet die dritten Eingangsdaten **2080**, die sequentiell durch einen zugeordneten, seriell arbeitenden Ringtypprozessor eingegeben werden, oder die Daten werden parallel in den Mikrocomputer eingegeben und parallel verarbeitet.

[0084] Der vierte Prozessor **2040** verarbeitet die vierten Eingangsdaten **2090**, die sequentiell durch einen zugeordneten, seriell arbeitenden Ringtypprozessor eingegeben werden, oder die Daten werden parallel in den Mikrocomputer eingegeben und parallel verarbeitet.

[0085] Das erste Erfassungssignal **2140** zeigt an, ob der erste Prozessor **2010** normal oder unnormal arbeitet, das zweite Erfassungssignal **2150** zeigt an, ob der zweite Prozessor **2020** normal oder unnormal arbeitet, das dritte Erfassungssignal **2160** zeigt an, ob der dritte Prozessor **2030** normal oder unnormal arbeitet, und das vierte Erfassungssignal **2170** zeigt an, ob der vierte Prozessor **2040** normal oder unnormal arbeitet. Es sind alternierende Signale, die in der vorgegebenen Zeitspanne entsprechend alternieren.

[0086] Diese Erfassungssignale können auch als Signal dafür verwendet werden, ob die Prüfergebnis-

se für den Inhalt der Eingangsdaten richtig oder falsch sind.

[0087] Der erste Prozessor **2010** gibt das Verarbeitungsergebnis beim Prüfen des Inhalts der Eingangsdaten **2060** als Erfassungssignal **2140** aus, der zweite Prozessor **2020** gibt das Verarbeitungsergebnis beim Prüfen des Inhalts der Eingangsdaten **2070** als Erfassungssignal **2150** aus, der dritte Prozessor **2030** gibt das Verarbeitungsergebnis beim Prüfen des Inhalts der Eingangsdaten **2080** als Erfassungssignal **2160** aus, und der vierte Prozessor **2040** gibt das Verarbeitungsergebnis beim Prüfen des Inhalts der Eingangsdaten **2090** als Erfassungssignal **2170** aus.

[0088] Die Ausgangsdaten **2100** des ersten Prozessors **2010**, die Ausgangsdaten **2110** des zweiten Prozessors **2020**, die Ausgangsdaten **2120** des dritten Prozessors **2030** und die Ausgangsdaten **2130** des vierten Prozessors **2040** werden an eine andere Vorrichtung ausgegeben, um sie zu steuern.

[0089] Vor der Ausgabe an die andere Vorrichtung wird durch einen Ausgangssignalanteil zum Beispiel zum Erfassen einer Übereinstimmung in den ersten Ausgangsdaten **2100**, den zweiten Ausgangsdaten **2110**, den dritten Ausgangsdaten **2120** und den vierten Ausgangsdaten **2130**, dem Erfassen einer Übereinstimmung in den ersten Ausgangsdaten **2100** und den zweiten Ausgangsdaten **2110**, dem Erfassen einer Übereinstimmung in den dritten Ausgangsdaten **2120** und den vierten Ausgangsdaten **2130** oder dem weiteren Ausgeben von endgültigen Ausgangsdaten unter Verwendung des ersten Erfassungssignals **2140**, des zweiten Erfassungssignals **2150**, des dritten Erfassungssignals **2160** und des vierten Erfassungssignals **2170** eine weitere, zu steuernde Vorrichtung gesteuert.

[0090] Die Arbeitsweise der Steuerung **2000** mit dem beschriebenen Aufbau ist die folgende:

Die ersten Eingangsdaten **2060** werden dem ersten Prozessor **2010** zugeführt, die zweiten Eingangsdaten **2070** dem zweiten Prozessor **2020**, die dritten Eingangsdaten **2080** dem dritten Prozessor **2030** und die vierten Eingangsdaten **2090** dem vierten Prozessor **2040**, und die Operationen werden entsprechend ausgeführt.

[0091] Die ersten Ausgangsdaten **2100** und das erste Erfassungssignal **2140** werden vom ersten Prozessor **2010** ausgegeben.

[0092] Das erste Erfassungssignal **2140** wird dem zweiten Prozessor **2020** zugeführt und dort zusammen mit den zweiten Eingangsdaten **2070** verarbeitet und daraus die zweiten Ausgangsdaten **2110** und das zweite Erfassungssignal **2150** erzeugt, das zweite Erfassungssignal **2150** wird dem dritten Prozessor **2030** zugeführt und dort zusammen mit den dritten Eingangsdaten **2080** verarbeitet und daraus die dritten Ausgangsdaten **2120** und das dritte Erfassungssignal **2160** erzeugt, das dritte Erfassungssignal **2160** wird dem vierten Prozessor **2040** zugeführt und dort zusammen mit den vierten Eingangsdaten **2090** verarbeitet, um die vierten Ausgangsdaten **2130** und das vierte Erfassungssignal **2170** zu erzeugen, und das Vorzeichen des vierten Erfassungssignals **2170** wird vom Vorzeicheninverter **2050** invertiert und das so entstehende Signal dem ersten Prozessor **2010** zugeführt.

[0093] Wenn das erste Erfassungssignal **2140** des ersten Prozessors **2010** ein Signal auf dem Pegel H ist, ist auch das zweite Erfassungssignal **2150** des zweiten Prozessors **2020** ein H-Pegel-Signal, und wenn das erste Erfassungssignal **2140** ein Signal auf dem Pegel L ist, ist auch das zweite Erfassungssignal **2150** des zweiten Prozessors **2020** ein L-Pegel-Signal.

[0094] Es wird angenommen, daß zuerst die Erfassungssignale aller Prozessoren auf dem Pegel L sind. Das Signal **2210** mit umgekehrtem Vorzeichen aus dem vierten Erfassungssignal **2140** des vierten Prozessors **2040** wird dann zu einem H-Pegel-Signal, das in den ersten Prozessor **2010** eingegeben wird.

[0095] Bei der Verarbeitung durch den ersten Prozessor **2010** wird der Ausgangspegel des ersten Erfassungssignals **2140** daher zu H, bei der folgenden Verarbeitung durch den zweiten Prozessor **2020** wird der Ausgangspegel des zweiten Erfassungssignals **2150** zu H, bei der Verarbeitung durch den dritten Prozessor **2030** wird der Ausgangspegel des dritten Erfassungssignals **2160** zu H, und bei der Verarbeitung durch den vierten Prozessor **2040** wird schließlich der Ausgangspegel des vierten Erfassungssignals **2170** zu H.

[0096] Da das Vorzeichen des Ausgangs am vierten Erfassungssignal **2170** durch den Vorzeicheninverter **2050** invertiert wird, ändert sich der Ausgangspegel des Signals **2210** mit invertiertem Vorzeichen von H auf L.

[0097] Der Ausgangspegel des ersten Erfassungssignals **2140** des ersten Prozessors **2010** ändert sich dann von H nach L, der Ausgangspegel des zweiten Erfassungssignals **2150** des zweiten Prozessors **2020** ändert sich ebenfalls von H nach L, als nächstes ändert sich der Ausgangspegel des dritten Erfassungssignals **2160** des dritten Prozessors **2030** von H nach L, und schließlich ändert sich auch der Ausgangspegel des vierten Erfassungssignals **2170** des vierten Prozessors **2040** von H nach L.

[0098] Da das Vorzeichen des Ausgangs am vierten Erfassungssignal **2170** durch den Vorzeicheninverter **2050** invertiert wird, ändert sich der Ausgangspegel des Signals **2210** mit invertiertem Vorzeichen von H auf L und kehrt zu seinem Ausgangszustand zurück.

[0099] Da wie beschrieben die Erfassungssignale vom ersten Prozessor bis zum vierten Prozessor in einer Schleife verbunden sind, sind die ersten Eingangsdaten bis zu den vierten Eingangsdaten und der Prozessor bis zum vierten Prozessor alle normal. Wenn sie richtig arbeiten, werden das erste Erfassungssignal **2140** bis zum vierten Erfassungssignal **2170** alternierende Signale, die mit einer festen Peri-

ode alternieren. Wenn einer der Prozessoren ausfällt und das entsprechende Erfassungssignal auf H oder L stehenbleibt, endet das Alternieren der Erfassungssignale, die die Schleife bilden.

[0100] Es ist daher vorteilhaft, das Fehlererfassungssignal der Schleife zu überwachen und zum Zeitpunkt des Beendens des Alternierens Notmaßnahmen zu ergreifen. Wenn alle Fehlererfassungssignale überwacht werden, wird der Ausfall eines Prozessors erkannt, und der Fehler läßt sich in kurzer Zeit analysieren.

[0101] In der **Fig.** 20 ist eine Ausführungsform mit vier Prozessoren gezeigt. Wenn Prozessoren in einer beliebigen Anzahl vorgesehen werden, ist es vorteilhaft, ein Fehlererfassungssignal auf die gleiche Weise durchzuleiten, wobei es hinsichtlich der Anzahl von Prozessoren keine Einschränkungen gibt.

[0102] Die **Fig.** 21 zeigt eine andere Ausführungsform der vorliegenden Erfindung. Bei der Ausführung der **Fig.** 21 werden die Prozessoren gemeinsam benutzt, und die Eingabeabschnitte umfassen jeweils einen Parallel-Seriell-Konverter vom Zeitmultiplextyp und die Ausgabeabschnitte einen Seriell-Parallel-Konverter vom Zeitmultiplextyp.

[0103] Das Steuergerät **3000** umfaßt eine erste Steuerschaltung **3010** mit einem Parallel-Seriell-Konverter **3020** vom Zeitmultiplextyp, einem ersten Prozessor **3030** und einem Seriell-Parallel-Konverter **3040** vom Zeitmultiplextyp sowie eine zweite Steuerschaltung **3200** mit einem Parallel-Seriell-Konverter **3210** vom Zeitmultiplextyp, einem zweiten Prozessor **3220** und einem Seriell-Parallel-Konverter **3230** vom Zeitmultiplextyp. Dem Parallel-Seriell-Konverter **3210** der zweiten Steuereinheit **3200** wird das Fehlererfassungssignal **3130** der ersten Steuerschaltung **3010** zugeführt. Das Vorzeichen des Fehlererfassungssignal **3320** der zweiten Steuerschaltung **3200** wird durch einen Vorzeicheninverter **3330** invertiert, und das so erhaltene Signal **3340** mit umgekehrtem Vorzeichen wird dem Parallel-Seriell-Konverter **3020** der ersten Steuereinheit **3010** zugeführt.

[0104] In den Parallel-Seriell-Konverter **3020** der ersten Steuerschaltung **3010** werden erste Eingangsdaten **3050**, zweite Eingangsdaten **3060** und dritte Eingangsdaten **3070** eingegeben, und in den Parallel-Seriell-Konverter **3210** der zweiten Steuerschaltung **3200** werden vierte Eingangsdaten **3240**, fünfte Eingangsdaten **3250** und sechste Eingangsdaten **3260** eingegeben.

[0105] Es wird angenommen, daß die Ausgangspegel des Fehlererfassungssignals **3130** der ersten Steuerschaltung **3010** und des Fehlererfassungssignals **3320** der zweiten Steuerschaltung **3130** zuerst auf L liegen.

[0106] Durch den Parallel-Seriell-Konverter **3020** der ersten Steuerschaltung **3010** werden die ersten Eingangsdaten **3050** in ein serielles Signal **3080** umgewandelt, das vom ersten Prozessor **3030** verarbeitet und als serielles Ausgangssignal **3090** ausgegeben wird, das dann vom Seriell-Parallel-Konverter **3040** in ein paralleles Signal umgewandelt und als Ausgangssignal **3100** ausgegeben wird.

[0107] Durch den Parallel-Seriell-Konverter **3210** der zweiten Steuerschaltung **3200** werden die vierten Eingangsdaten **3240** in ein serielles Signal **3270** umgewandelt, das vom zweiten Prozessor **3220** verarbeitet und als serielles Ausgangssignal **3280** ausgegeben wird, das dann vom Seriell-Parallel-Konverter **3230** in ein paralleles Signal umgewandelt und als Ausgangssignal **3290** ausgegeben wird.

[0108] Auf die gleiche Weise werden die in die erste Steuerschaltung **3010** eingegebenen zweiten Eingangsdaten **3060** durch den ersten Prozessor **3030** verarbeitet und als Ausgangssignal **3110** ausgegeben, die in die zweite Steuerschaltung **3200** eingegebenen fünften Eingangsdaten **3250** durch den zweiten Prozessor **3220** verarbeitet und als Ausgangssignal **3300** ausgegeben, die in die erste Steuerschaltung **3010** eingegebenen dritten Eingangsdaten **3070** durch den ersten Prozessor **3030** verarbeitet und als Ausgangssignal **3120** ausgegeben, und die in die zweite Steuerschaltung **3200** eingegebenen sechsten Eingangsdaten **3260** durch den zweiten Prozessor **3220** verarbeitet und als Ausgangssignal **3310** ausgegeben.

[0109] Wenn das Vorzeichen des Fehlererfassungssignals **3320** der zweiten Steuerschaltung **3200** durch den Vorzeicheninverter **3330** invertiert wird und das Signal **3340** mit invertiertem Vorzeichen vom Pegel H in den Parallel-Seriell-Konverter **3020** der ersten Steuerschaltung **3010** eingegeben wird, wird auf die gleiche Weise wie oben das Fehlererfassungssignal **3130** mit dem Pegel H ausgegeben.

[0110] Da das Fehlererfassungssignal **3130** mit dem Pegel H der ersten Steuerschaltung **3010** in den Parallel-Seriell-Konverter **3210** der zweiten Steuerschaltung **3200** eingegeben wird, wird auf die gleiche Weise wie oben das Fehlererfassungssignal **3320** mit dem Pegel H ausgegeben.

[0111] Das Vorzeichen dieses Fehlererfassungssignals **3320** mit dem Pegel H wird vom Vorzeicheninverter **3330** invertiert, so daß in den Parallel-Seriell-Konverter **3020** der ersten Steuerschaltung **3010** ein Fehlererfassungssignal mit dem Pegel L eingegeben wird und auf die gleiche Weise wie oben das Fehlererfassungssignal **3130** mit dem Pegel H ausgegeben wird.

[0112] Wenn die erste Steuerschaltung **3010** und die Eingangsdaten und die zweite Steuerschaltung **3200** und die Eingangsdaten normal sind, werden das Fehlererfassungssignal **3130** der ersten Steuerschaltung **3010** und das Fehlererfassungssignal **3320** der zweiten Steuerschaltung **3200** zu alternierenden Signalen, die mit einer festen Periode alternieren.

[0113] In der **Fig.** 21 können der erste Prozessor **3020** und der zweite Prozessor **3210** von einem Mikrocomputer gebildet werden. Auf diese Weise können dann durch ein Programm verschiedene Arten ausgewählt werden. Zum Beispiel läßt sich durch ein

Verfahren zum Ausführen des Fehlererfassungsprozesses nach dem Eingangsdatenprozeß oder dem parallelen Ausführen des Eingangsdatenprozesses und des Fehlererfassungsprozesses die gleiche Arbeitsweise wie oben realisieren.

[0114] Es erübrigt sich anzumerken, daß durch Ersetzen des Parallel-Seriell-Konverters durch einen Multiplexer und des Seriell-Parallel-Konverters durch einen Demultiplexer die gleiche Arbeitsweise realisiert werden kann.

[0115] Die **Fig.** 22 zeigt eine andere Ausführungsform der vorliegenden Erfindung. Der Unterschied in der Ausführungsform der **Fig.** 22 ist der, daß der Vorzeicheninverter **2050** in der Ausführungsform der **Fig.** 20 durch einen 1-Bit-Addierer **2220** ersetzt ist. Alles andere bleibt gleich, so daß die Erläuterung der Arbeitsweise davon weggelassen wird und im folgenden nur die Arbeitsweise des Addierers **2220** der **Fig.** 22 beschrieben wird.

[0116] Es wird angenommen, daß der erste Prozessor **2010**, der zweite Prozessor **2020**, der dritte Prozessor **2030** und der vierte Prozessor **2040**, die das Steuergerät **2000** bilden, normal arbeiten und die Ausgangspegel des ersten Erfassungssignals **2140**, des zweiten Erfassungssignals **2150**, des dritten Erfassungssignals **2160** und des vierten Erfassungssignals **2170** auf L liegen.

[0117] Die ersten Eingangsdaten **2060** werden in den ersten Prozessor **2010** eingegeben und von diesem verarbeitet und als erstes Erfassungssignal **2140** mit dem Pegel L ausgegeben. Dieses erste Erfassungssignal **2140** wird in den zweiten Prozessor **2020** eingegeben, dort zusammen mit den zweiten Eingangsdaten **2070** verarbeitet und als zweites Erfassungssignal **2150** mit dem Pegel L ausgegeben. Dieses zweite Erfassungssignal **2150** wird in den dritten Prozessor **2030** eingegeben, dort zusammen mit den dritten Eingangsdaten **2080** verarbeitet und als drittes Erfassungssignal **2160** mit dem Pegel L ausgegeben. Dieses dritte Erfassungssignal **2160** wird in den vierten Prozessor **2040** eingegeben, dort zusammen mit den vierten Eingangsdaten **2090** verarbeitet und als viertes Erfassungssignal **2170** mit dem Pegel L ausgegeben.

[0118] Das vierte Erfassungssignal **2170** wird in den Addierer **2220** eingegeben und zu H-Pegel-Daten addiert, so daß der Ausgangspegel des Ausgangssignals **2210** davon zu H wird. Dieses Signal wird wieder in den ersten Prozessor **2010** eingegeben.

[0119] Dieses H-Pegel-Ausgangssignal **2210** wird dort zusammen mit den ersten Eingangsdaten **2060** verarbeitet, so daß das erste Erfassungssignal **2140** zu einem H-Pegel-Signal wird, woraufhn auf die gleiche Weise das zweite Erfassungssignal **2150** des zweiten Prozessors **2020** zu einem H-Pegel-Signal wird, das dritte Erfassungssignal **2160** des dritten Prozessors **2030** zu einem H-Pegel-Signal wird und das vierte Erfassungssignal **2170** des vierten Prozessors **2040** zu einem H-Pegel-Signal wird.

[0120] Wenn dieses vierte Erfassungssignal **2170** in den Addierer **2220** eingegeben und zu H-Pegel-Daten addiert wird, wird der Ausgangspegel des Ausgangssignals **2210** erneut zu L. Dieses Ausgangssignal **2210** wird wieder zum ersten Prozessor **2010** geführt und zusammen mit den ersten Eingangsdaten verarbeitet. Das heißt, es kehrt zu seinem Anfangszustand zurück.

[0121] Wenn jede Einheit des Steuergeräts **2000** normal arbeitet, werden daher das erste Erfassungssignal **2140**, das zweite Erfassungssignal **2150**, das dritte Erfassungssignal **2160** und das vierte Erfassungssignal **2170**, die eine Schleife bilden, zu alternierenden Signalen, bei denen der H- und L-Pegel mit einer festen Periode abwechselt, so daß es vorteilhaft ist, einen Aufbau zu schaffen, bei dem eines der Signale überwacht wird und bei dem Notmaßnahmen getroffen werden, wenn das Alternieren endet.

[0122] In der **Fig.** 22 ist wie in der **Fig.** 20 ein Beispiel mit vier Prozessoren gezeigt. Es kann jedoch eine beliebige Anzahl von Prozessoren vorgesehen werden, wobei es vorteilhaft ist, diese auf die gleiche Weise zu verbinden. Es gibt keine Beschränkungen hinsichtlich der Anzahl der Prozessoren.

[0123] Die **Fig.** 23 zeigt eine andere Ausführungsform der vorliegenden Erfindung. Der Unterschied zu der **Fig.** 20 ist der, daß in der Ausführungsform der **Fig.** 23 zu einem 2-Bit-Erfassungssignal eines Prozessors die 2-Bit-Daten "0" "1" (die dezimal eine 1 anzeigen) addiert werden, und daß das Ausgangssignal der Addition in den nächsten Prozessor eingegeben und zusammen mit einem dem Prozessor zugeführten Signal verarbeitet wird.

[0124] In der **Fig.** 23 arbeiten der erste Prozessor **2010**, der zweite Prozessor **2020**, der dritte Prozessor **2030** und der vierte Prozessor **2040** sowie der erste Addierer **2220**, der zweite Addierer **2230**, der dritte Addierer **2240** und der erste Subtrahierer **2250** jeweils normal, und der Ausgang des ersten Erfassungssignals **2140**, der einen 2 Bit langen Status anzeigt, ist "0" "0" (dezimal eine 0 anzeigend), der Ausgang des zweiten Erfassungssignals **2150** ist auch "0" "0", der Ausgang des dritten Erfassungssignals **2160** ist ebenfalls "0" "0", und der Ausgang des vierten Erfassungssignals **2170** ist wiederum "0" "0".

[0125] Die ersten Eingangsdaten **2060** werden in den ersten Prozessor eingegeben und von diesem verarbeitet, der die ersten Ausgangsdaten **2100** und das "0" "0" des ersten Erfassungssignals **2140** ausgibt.

[0126] Da zu dem Ausgang "0" "0" des ersten Erfassungssignals **2140** durch den ersten Addierer **2220** die 2-Bit-Daten "0" "1" (die dezimal eine 1 anzeigen) hinzuaddiert werden, wird das Additions-Ausgangssignal **2260** zu "0" "1".

[0127] Das "0" "1" des Additions-Ausgangssignals **2260** wird in den zweiten Prozessor **2020** eingegeben und dort zusammen mit den zweiten Eingangsdaten **2070** verarbeitet, und es werden von diesem die zweiten Ausgangsdaten **2110** und das "0" "1" des zweiten Erfassungssignals **2150** ausgegeben.

[0128] Da zu dem Ausgang "0" "1" des zweiten Erfassungssignals **2150** durch den zweiten Addierer **2230** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2270** zu "1" "0" (was dezimal eine 2 anzeigt).

[0129] Das "1" "0" des Additions-Ausgangssignals **2270** wird in den dritten Prozessor **2030** eingegeben und dort zusammen mit den dritten Eingangsdaten **2080** verarbitet, und es werden von diesem die dritten Ausgangsdaten **2120** und das "1" "0" des dritten Erfassungssignals **2160** ausgegeben.

[0130] Da zu dem Ausgang "1" "0" des dritten Erfassungssignals **2160** durch den dritten Addierer **2240** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2280** zu "1" "1" (was dezimal eine 3 anzeigt).

[0131] Das "1" "1" des Additions-Ausgangssignals **2280** wird in den vierten Prozessor **2040** eingegeben und dort zusammen mit den vierten Eingangsdaten **2090** verarbeitet, und es werden von diesem die vierten Ausgangsdaten **2130** und das "1" "1" des vierten Erfassungssignals **2170** ausgegeben.

[0132] Das "1" "1" des vierten Erfassungssignals **2170** wird in den ersten Subtrahierer **2250** eingegeben und die Daten "1" "0" (dezimal eine 2 anzeigend) werden von den Daten "1" "1" subtrahiert, so daß das Subtraktions-Ausgangssignal **2290** zu "0" "1" wird.

[0133] Das "0" "1" des Subtraktions-Ausgangssignals **2290** wird in den ersten Prozessor **2010** eingegeben und dort zusammen mit den ersten Eingangsdaten **2090** verarbeitet, und es werden von diesem die ersten Ausgangsdaten **2100** und das "0" "1" des ersten Erfassungssignals **2140** ausgegeben.

[0134] Da zu dem "0" "1" des ersten Erfassungssignals **2140** durch den ersten Addierer **2220** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2260** zu "1" "0" (was dezimal eine 2 anzeigt).

[0135] Das "1" "0" des Additions-Ausgangssignals **2260** wird in den zweiten Prozessor **2020** eingegeben und dort zusammen mit den zweiten Eingangsdaten **2070** verarbeitet, und es werden von diesem die zweiten Ausgangsdaten **2110** und das "1" "0" des zweiten Erfassungssignals **2150** ausgegeben.

[0136] Da zu dem "1" "0" des zweiten Erfassungssignals **2150** durch den zweiten Addierer **2230** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2270** zu "1" "1" (was dezimal eine 3 anzeigt).

[0137] Das " 1 " " 1 " des Additions-Ausgangssignals **2270** wird in den dritten Prozessor **2030** eingegeben und dort zusammen mit den dritten Eingangsdaten **2080** verarbeitet, und es werden von diesem die dritten Ausgangsdaten **2120** und das "1" "1" des dritten Erfassungssignals **2160** ausgegeben.

[0138] Da zu dem "1" "1" des dritten Erfassungssignals **2160** durch den dritten Addierer **2240** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2280** zu "0" "0" (was dezimal eine 0 anzeigt).

[0139] Das "0" "0" des Additions-Ausgangssignals **2280** wird in den vierten Prozessor **2040** eingegeben und dort zusammen mit den vierten Eingangsdaten **2090** verarbeitet, und es werden von diesem die vierten Ausgangsdaten **2130** und das "0" "0" des vierten Erfassungssignals **2170** ausgegeben.

[0140] Das "0" "0" des vierten Erfassungssignals **2170** wird in den ersten Subtrahierer **2250** eingegeben und die Daten "1" "0" subtrahiert, so daß das Subtraktions-Ausgangssignal **2290** zu "1" "0" (was dezimal eine 2 anzeigt) wird.

[0141] Das "1" "0" des Subtraktions-Ausgangssignals **2290** wird in den ersten Prozessor **2010** eingegeben und dort zusammen mit den ersten Eingangsdaten **2090** verarbeitet, und es werden von diesem die ersten Ausgangsdaten **2100** und das "1" "0" des ersten Erfassungssignals **2140** ausgegeben.

[0142] Da zu dem "1" "0" des ersten Erfassungssignals **2140** durch den ersten Addierer **2220** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2260** zu "1" "1" (was dezimal eine 3 anzeigt).

[0143] Das "1" "1" des Additions-Ausgangssignals **2260** wird in den zweiten Prozessor **2020** eingegeben und dort zusammen mit den zweiten Eingangsdaten **2070** verarbeitet, und es werden von diesem die zweiten Ausgangsdaten **2110** und das "1" "1" des zweiten Erfassungssignals **2150** ausgegeben.

[0144] Da zu dem "1" "1" des zweiten Erfassungssignals **2150** durch den zweiten Addierer **2230** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2270** zu "0" "0" (was dezimal eine 3 anzeigt).

[0145] Das "0" "0" des Additions-Ausgangssignals **2270** wird in den dritten Prozessor **2030** eingegeben und dort zusammen mit den dritten Eingangsdaten **2080** verarbeitet, und es werden von diesem die dritten Ausgangsdaten **2120** und das "0" "0" des dritten Erfassungssignals **2160** ausgegeben.

[0146] Da zu dem "0" "0" des dritten Erfassungssignals **2160** durch den dritten Addierer **2240** die Daten "0" "1" hinzuaddiert werden, wird das Additions-Ausgangssignal **2280** zu "0" "1" (was dezimal eine 1 anzeigt).

[0147] Das "0" "1" des Additions-Ausgangssignals **2280** wird in den vierten Prozessor **2040** eingegeben und dort zusammen mit den vierten Eingangsdaten **2090** verarbeitet, und es werden von diesem die vierten Ausgangsdaten **2130** und das "0" "1" des vierten Erfassungssignals **2170** ausgegeben.

[0148] Das "0" "1" des vierten Erfassungssignals **2170** wird in den ersten Subtrahierer **2250** eingegeben und die Daten "1" "0" subtrahiert, so daß das Subtraktions-Ausgangssignal **2290** zu "0" "0" (was dezimal eine 0 anzeigt) wird.

[0149] Das "0" "0" des Subtraktions-Ausgangssignals **2290** wird in den ersten Prozessor **2010** eingegeben und dort zusammen mit den ersten Eingangsdaten **2090** verarbeitet, und es werden von diesem die ersten Ausgangsdaten **2100** und das "0" "0" des

ersten Erfassungssignals **2140** ausgegeben.

[0150] Wenn das Steuergerät **2000** so aufgebaut ist, daß zu dem Ausgangssignal des ersten Fehlererfassungssignals des erste Prozessors ein "0" "1" (1 dezimal) hinzuaddiert wird und das Additionssignal in den zweiten Prozessor eingegeben wird, zum Ausgangssignal des zweiten Fehlererfassungssignals ein "0" "1" (1 dezimal) hinzuaddiert wird und das Additionssignal in den dritten Prozessor eingegeben wird, zum Ausgangssignal des dritten Fehlererfassungssignals ein "0" "1" (1 dezimal) hinzuaddiert wird und das Additionssignal in den vierten Prozessor eingegeben wird, und vom Ausgangssignal des vierten Fehlererfassungssignals ein "1" "0" (2 dezimal) subtrahiert und das Subtraktionssignal in den ersten Prozessor eingegeben wird, werden die Fehlererfassungssignale der Prozessoren zu alternierenden Signalen, die mit einer festen Periode abwechseln, wenn die Einheiten des Steuergeräts **2000** normal arbeiten.

[0151] In der **Fig.** 23 ist wie in der **Fig.** 22 ein Beispiel mit vier Prozessoren gezeigt. Es kann jedoch eine beliebige Anzahl von Prozessoren vorgesehen werden, wobei es vorteilhaft ist, diese auf die gleiche Weise zu verbinden. Es gibt keine Beschränkungen hinsichtlich der Anzahl der Prozessoren.

[0152] Die **Fig.** 24 zeigt eine andere Ausführungsform der vorliegenden Erfindung. Die **Fig.** 24 unterscheidet sich von der **Fig.** 21 im Verarbeitungsverfahren für das zweite Erfassungssignal.

[0153] Wenn alle ersten Eingangsdaten **3050** in die erste Steuerschaltung **3010** der **Fig.** 24 eingegeben sind, werden als nächstes die zweiten Eingangsdaten **3060** eingegeben, und wenn alle zweiten Eingangsdaten **3060** eingegeben sind, werden als nächstes die dritten Eingangsdaten **3070** eingegeben, und wenn die dritten Eingangsdaten **3070** alle eingegeben sind, werden als nächstes CRC-Daten **3140** zum Überprüfen der Daten als Fehlererfassungssignal **3160** über einen ersten Addierer **3150** eingegeben.

[0154] Diese Daten werden durch den Parallel-Seriell-Konverter **3020** sequentiell in ein serielles Signal **3080** umgewandelt, in den ersten Prozessor **3030** eingegeben, wie vorgegeben verarbeitet und als serielles Signal **3090** ausgegeben. Dieses serielle Signal **3090** wird durch den Seriell-Parallel-Konverter **3040** umgewandelt und parallel als Ausgangssignal **3100**, Ausgangssignal **3110** und Ausgangssignal **3120** ausgegeben, um eine andere Vorrichtung zu steuern, und außerdem wird das Fehlererfassungssignal **3130** ausgegeben und in den zweiten Addierer **3340** eingegeben.

[0155] Wenn auf die gleiche Weise alle vierten Eingangsdaten **3240** in die zweite Steuerschaltung **3200** eingegeben sind, werden als nächstes die fünften Eingangsdaten **3250** eingegeben, und wenn alle fünften Eingangsdaten **3250** eingegeben sind, werden als nächstes die sechsten Eingangsdaten **3260** eingegeben, und wenn die sechsten Eingangsdaten **3260** alle eingegeben sind, werden als nächstes die CRC-Daten **3330** zum Überprüfen der Daten als Fehlererfassungssignal **3350** über den zweiten Addierer **3340** eingegeben.

[0156] Diese Daten werden durch den Parallel-Seriell-Konverter **3210** sequentiell in ein serielles Signal **3270** umgewandelt, in den zweiten Prozessor **3220** eingegeben, wie vorgegeben verarbeitet und als serielles Signal **3280** ausgegeben. Dieses serielle Signal **3280** wird durch den Seriell-Parallel-Konverter **3230** umgewandelt und parallel als Ausgangssignal **3290**, Ausgangssignal **3300** und Ausgangssignal **3310** ausgegeben, um eine andere Vorrichtung zu steuern, und außerdem wird das Fehlererfassungssignal **3320** ausgegeben, dessen Vorzeichen durch den Vorzeicheninverter **3350** invertiert wird, so daß das Fehlererfassungssignal **3360** mit invertiertem Vorzeichen in den ersten Addierer **3150** eingegeben wird.

[0157] Im Anfangszustand wird angenommen, daß die Ausgangspegel des Fehlererfassungssignals **3130** der ersten Steuerschaltung **3010** und des Fehlererfassungssignals **3320** der zweiten Steuerschaltung **3200** auf L liegen.

[0158] Das Fehlererfassungssignal **3130** auf dem L-Pegel der ersten Steuerschaltung **3010** wird in den zweiten Addierer **3340** der zweiten Steuerschaltung **3200** eingegeben, die CRC-Daten **3330** werden dabei jedoch nicht zerstört. Mit anderen Worten werden in den zweiten Seriell-Parallel-Konverter **3210** die normalen CRC-Daten **3350** eingegeben.

[0159] Der Ausgangspegel des Signals **3370** mit invertiertem Vorzeichen aus dem Fehlererfassungssignal **3320** der zweiten Steuerschaltung **3200** ist dagegen H. Das Signal **3370** mit invertiertem Vorzeichen auf dem Pegel H wird in den ersten Addierer **3150** eingegeben und zerstört die CRC-Daten **3140**.

[0160] Diese zerstörten CRC-Daten **3160** werden in den ersten Parallel-Seriell-Konverter **3020** eingegeben, durch den ersten Prozessor **3030** verarbeitet, und es wird über den ersten Seriell-Parallel-Konverter **3040** das Fehlererfassungssignal **3130** auf dem H-Pegel ausgegeben.

[0161] Dieses Fehlererfassungssignal **3130** auf dem Pegel H wird in den zweiten Addierer **3340** eingegeben, so daß die CRC-Daten **3330** zerstört werden.

[0162] Diese zerstörten CRC-Daten **3350** werden in den zweiten Parallel-Seriell-Konverter **3210** eingegeben, durch den zweiten Prozessor **3220** verarbeitet, und es wird über den zweiten Seriell-Parallel-Konverter **3230** das Fehlererfassungssignal **3320** auf dem H-Pegel ausgegeben.

[0163] Das Vorzeichen dieses Fehlererfassungssignals **3130** auf dem Pegel H wird vom Vorzeicheninverter **3360** invertiert, so daß an den ersten Addierer **3150** ein L-Pegel-Signal ausgegeben wird.

[0164] Da die CRC-Daten **3140** vom ersten Prozessor **3150** nicht zerstört werden, werden in den ersten Parallel-Seriell-Konverter die normalen CRC-Daten

eingegeben. Es wird daher zum Anfangszustand zurückgekehrt und gesteuert.

[0165] Wenn alle Einheiten des Steuergeräts **3000** normal arbeiten, alternieren das Fehlererfassungssignal **3130** und das Fehlererfassungssignal **3320** mit einer festen Periode.

[0166] Die **Fig.** 25 ist ein Beispiel für ein Programm, wenn der erste Prozessor und der zweite Prozessor der **Fig.** 21 und 24 jeweils von einem Mikrocomputer gebildet werden. Die Datenverarbeitung wird nach dem Eingabeprozeß ausgeführt und der Ausgabeprozeß entsprechend dem Ergebnis ausgeführt. Wenn dieser Prozeß endet, beginnt der Diagnoseprozeß zum Erfassen eines Fehlers. Das heißt, daß der Fehlerdiagnoseprozeß nach dem Ausführen des Eingabeprozesses zum Erfassen eines Fehlers ausgeführt wird und wenn dieser Prozeß zu Ende ist, wird der Ausgabeprozeß ausgeführt. Im Normalzustand wird kein Signal ausgegeben. Bei Unnormalitäten wird ein Signal ausgegeben und zum Beispiel ein Notfallprozeß ausgeführt.

[Auswirkungen der Erfindung]

[0167] Erfindungsgemäß wird wie beschrieben ein normaler und ein unnormaler Betrieb eines Steuergeräts mit einer Anzahl von Prozessoren oder logischen Einheiten sicher erfaßt.

[0168] Es kann der Inhalt der Eingangsdaten, die von jedem Prozessor verarbeitet werden, überprüft werden, so daß nicht nur ein normaler oder unnormaler Betrieb der Hardware, sondern auch ein normaler und ein unnormaler Zustand der Software erfaßt werden kann.

[0169] Wenn die vorliegende Erfindung bei einer ATP-Einrichtung angewendet wird, werden die Steuerdaten für die beiden Systeme aus dem ATP-Sollgeschwindigkeitssignal erzeugt, jede Logikeinheit in der ATP-Einrichtung zur Bearbeitung der Steuerdaten ist doppelt vorhanden, und jedes System umfaßt wenigstens zwei CRC-Datensätze zum Überprüfen der Steuerdaten. Durch Auswechseln der entgegengesetzten CRC-Daten für jede Logikeinheit oder die Auswahl von einem der beiden Datensätze entsprechend dem Inhalt eines Fehlererfassungssignals von den beiden Logikeinheiten lassen sich die Steuerdaten und der Betrieb jeder Logikschaltung überprüfen.

[0170] Die ATP-Einrichtung umfaßt einen Frequenzkomparator **152** zum Vergleichen der ATP-Geschwindigkeitssollfrequenz **151** mit der Geschwindigkeitsfrequenz **5** sowie einen Fehlerdetektor **153** zum Überprüfen des Inhalts der Steuerdaten **12** des ersten Mikrocomputers und des Statusses der Logikschaltung **15** auf der Basis eines Fehlererfassungssignals vom System B und zur Ausgabe eines Fehlererfassungssignals **154** für das System A.

[0171] Die Logikschaltung **16** des Systems B umfaßt einen Frequenzkonverter 160 zum Umwandeln der Steuerdaten **13** des zweiten Mikrocomputers in ein ATP-Geschwindigkeitssollfrequenz(signal) **161**, einen Frequenzkomparator **162** zum Vergleichen der ATP-Geschwindigkeitssollfrequenz **151** mit der Geschwindigkeitsfrequenz **5** sowie den Fehlerdetektor **153** zum Überprüfen des Inhalts der Steuerdaten **13** des zweiten Mikrocomputers und des Statusses der Logikschaltung **16** auf der Basis eines Fehlererfassungssignals vom System A und zur Ausgabe eines Fehlererfassungssignals **164** für das System B.

[0172] Das Bezugszeichen **19** bezeichnet eine Logikschaltung zum Erzeugen eines Vorzeicheninversionssignals **165** zum Invertieren des Vorzeichens des Fehlererfassungssignals **164** des Systems B und zu dessen Eingabe in den Fehlerdetektor **153** des Systems A. Zum Beispiel ist es eine Inverterschaltung. Die Logikschaltung **19** zur Vorzeicheninversion ist zwischen die Ausgangsseite des Fehlerdetektors **153** des Systems A und die Eingangsseite des Fehlerdetektors **163** des Systems B geschaltet. Es kann auch das Vorzeichen des Fehlererfassungssignals **154** des Systems A invertiert werden.

[0173] Die ATP-Sollgeschwindigkeit ist eine Grenzgeschwindigkeit, die entsprechend dem Bahnzustand und anderen Zuständen und der Fahrt über die Schienen **1** eingestellt ist, es ist ein frequenzmoduliertes Signal.

[0174] Die ATP-Sollgeschwindigkeit wird mittels der Aufnahmeantenne **6** des fahrenden elektrischen Motorfahrzeugs aufgenommen und im Fahrzeugempfänger **7** gleichzeitig demoduliert, verstärkt und geformt und dann erfaßt.

[0175] Die der Geschwindigkeit des elektrischen Motorfahrzeugs entsprechende Geschwindigkeitsfrequenz **5** wird durch Formen im Wellenformer **4** der Ausgangsspannung des Drehzahlgenerators **3** erfaßt, der direkt mit der Achse des elektrischen Motorfahrzeugs verbunden ist.

[0176] Das ATP-Sollgeschwindigkeitssignal **8** aus dem Fahrzeugempfänger **7** wird in den ersten Mikrocomputer **10** und den zweiten Mikrocomputer **11** der ATP-Einrichtung **9** eingegeben und in Steuerdaten umgewandelt oder verarbeitet.

[0177] Wenn die dem ATP-Sollgeschwindigkeitssignal entsprechenden Daten eine Länge von einem Wort haben, werden n Arten von ATP-Sollgeschwindigkeitssignalen in Daten von n Wörtern umgewandelt.

[0178] Die durch das Umwandeln oder Verarbeiten durch die Mikrocomputer **10** und **11** erzeugten Steuerdaten werden in den in die Mikrocomputer **10** und **11** eingebauten Speichern gespeichert und aufeinanderfolgend als Steuerdaten **12** und **13** ausgegeben.

[0179] Der Speicher, der die Steuerdaten **12** und **13** speichert, kann aus den Speichern bestehen, die in die Mikrocomputer **10** und **11** eingebaut sind, oder aus den Speichern oder Mikrocomputern (in der Zeichnung nicht gezeigt), die in die Logikschaltung **15** und die Logikschaltung **16** eingebaut sind, oder aus einem Speicher, der unabhängig von der ATP-Einrichtung vorgesehen ist.

[0180] Die Steuerdaten **12** und **13** können in einem

Speicher oder in verschiedenen Speichern liegen.

[0181] Die **Fig.** 1 zeigt ein Beispiel, bei dem der obige Speicher in jeden Mikrocomputer eingebaut ist.

[0182] Jeder dieser Speicher kann eine Speichereinrichtung mit einem Cachespeicher sein.

[0183] Der in der **Fig.** 3 gezeigte Speicherinhalt (**1**) wird durch den ersten Mikrocomputer **10** erzeugt und enthält die Daten DI_0 bis DI_{X-1}, die unter den Adressen AI_0 bis AI_{X-1} des Speichers gespeichert sind. Auf die gleiche Weise wird der in der **Fig.** 3 gezeigte Speicherinhalt (**2**) durch den zweiten Mikrocomputer **11** erzeugt und enthält die Daten DI_0 bis DI_{X-1}, die unter den Adressen AI_0 bis DI_{X-1} des Speichers gespeichert sind.

[0184] Am Ende der Daten DI_0 bis DI_{X-1} des ersten Mikrocomputers **10** sind die CRC-Daten D_{CRC1} mit der Länge eines Wortes angefügt, um einen Fehler entdecken zu können. Die Bits niedriger Ordnung bis zu den Bits hoher Ordnung der Daten DI_0 bis DI_{X-1} und D_{CRC1} an den Adressen AI_0 bis AI_n werden seriell ausgegeben. Die ausgegebenen seriellen Daten werden als die Steuerdaten **12** des ersten Mikrocomputers in die Logikschaltung **15** des Systems A eingegeben.

[0185] Auf die gleiche Weise sind am Ende der Daten DI_0 bis DI_{X-1} des zweiten Mikrocomputers **11** die CRC-Daten D_{CRC2} mit der Länge eines Wortes angefügt, um einen Fehler entdecken zu können. Die Bits niedriger Ordnung bis zu den Bits hoher Ordnung der Daten DI_0 bis DI_{X-1} und D_{CRC2} an den Adressen AI_0 bis AI_2 werden seriell ausgegeben. Die ausgegebenen seriellen Daten werden als die Steuerdaten **13** des zweiten Mikrocomputers in die Logikschaltung **16** des Systems B eingegeben.

[0186] Es ist in diesem Fall vorteilhaft, wenn die Daten DI_0 bis DI_{X-1} in beiden Systemen A und B gleich sind, und die vorliegende Ausführungsform zeigt den Fall, daß die Daten gleich sind.

[0187] In der Logikschaltung **15** des Systems A werden die Steuerdaten **12** entsprechend den ATP-Sollgeschwindigkeitssignalen durch den Frequenzkonverter **150** vom seriellen Typ (eine später noch beschriebene arithmetische Ringschaltung) in n Arten von ATP-Sollgeschwindigkeitsfrequenzen **151** umgewandelt und seriell ausgegeben.

[0188] Die seriell ausgegebenen Signale werden in den Frequenzkomparator **152** (wird später noch beschrieben) eingegeben.

[0189] Auf die gleiche Weise werden in der Logikschaltung **16** des Systems B die Steuerdaten **13** entsprechend den ATP-Sollgeschwindigkeitssignalen durch den Frequenzkonverter **160** (eine später noch beschriebene arithmetische Ringschaltung) in n Arten von ATP-Sollgeschwindigkeitsfrequenzen **161** umgewandelt und seriell ausgegeben.

[0190] Die seriell ausgegebenen Signale werden in den Frequenzkomparator **162** (wird später noch beschrieben) eingegeben.

[0191] In diesem Fall sind die Daten D_{CRC1} des Systems A und die Daten D_{CRC2} des Systems B CRC-Daten, die jeweils durch wenigstens zwei verschiedene erzeugende Polynomausdrücke erzeugt werden, das heißt D_{CRC10} und D_{CRC11} für das System A und D_{CRC20} und D_{CRC21} für das System B. Bei diesen CRC-Daten können zwei CRC-Datensätze aus einem Satz CRC-Daten erzeugt werden.

[0192] Diese CRC-Daten werden von den Frequenzkonvertern **150** und **160** nicht frequenzkonveniert.

[0193] Die Geschwindigkeitsfrequenz **5** wird am anderen Eingang des Frequenzkomparators **152** des Systems A eingegeben und mit der ATP-Sollgeschwindigkeitsfrequenz **151** verglichen. Das Ergebnis des Vergleichs wird seriell als Ausgangssignal **17** ausgegeben.

[0194] Auf die gleiche Weise wird die Geschwindigkeitsfrequenz **5** am anderen Eingang des Frequenzkomparators **162** des Systems B eingegeben und mit der ATP-Sollgeschwindigkeitsfrequenz **161** verglichen. Das Ergebnis des Vergleichs wird seriell als Ausgangssignal **18** ausgegeben.

[0195] Der Vergleicher **20** vergleicht das Ausgangssignal **17** der Logikschaltung **15** mit dem Ausgangssignal **18** der Logikschaltung **16**, gleicht die Signale einander an und gibt ein angepaßtes Signal als Bremsbefehl **21** an die Bremseinrichtung **22** aus.

[0196] Die **Fig.** 4 zeigt eine Darstellung des Prinzips der Frequenzvergleichsoperation, wenn ein elektrisches Motorfahrzeug an einer vorgegebenen Stelle anhält. Es wird die Beziehung zwischen der ATP-Sollgeschwindigkeitsfrequenz, der Geschwindigkeitsfrequenz des elektrischen Motorfahrzeugs und dem Bremsbefehl erläutert.

[0197] In der **Fig.** 4 zeigt die Ordinatenachse die ATP-Sollgeschwindigkeitsfrequenz f_{ATP} und die Abszissenachse die Zeit t an, und das Symbol f_V gibt die Geschwindigkeitsfrequenz des elektrischen Motorfahrzeugs an.

[0198] Die Beziehung zwischen der ATP-Sollgeschwindigkeitsfrequenz f_{ATP0} und der Geschwindigkeitsfrequenz f_V des elektrischen Motorfahrzeugs zum Zeitpunkt t_0 ist $f_{ATPO} > f_V$. In der Zeitspanne von der Zeit t_0 bis zur Zeit t_1 werden an das elektrische Motorfahrzeug weder ein Fahrbefehl noch ein Bremsbefehl ausgegeben, und das elektrische Motorfahrzeug ist in einem Zustand, in dem es im Dahinrollen langsamer wird.

[0199] Wenn sich in diesem Zustand zum Zeitpunkt t_1 die ATP-Sollgeschwindigkeitsfrequenz von f_{ATP0} auf f_{ATP1} ändert, wird f_{ATP1} kleiner als f_V, so daß vom Vergleicher **20** das Signal B_1 als Bremsbefehlsignal **21** ausgegeben wird, das der Bremseinrichtung **22** zugeführt wird.

[0200] Von der Bremseinrichtung **22** wird die dem Befehl entsprechende Bremskraft an das elektrische Motorfahrzeug ausgegeben, und das elektrische Motorfahrzeug beginnt mit der Abbremsung.

[0201] Wenn sich zum Zeitpunkt t_2 die ATP-Sollgeschwindigkeitsfrequenz von fahl auf f_{ATP2} ändert, wird f_{ATP2} kleiner als f_V, so daß das Bremsbefehlsignal **21** vom Vergleicher **20** zu B_2 wird und das elektrische

Motorfahrzeug stärker bremst.

[0202] Wenn sich auf die gleiche Weise zu den Zeitpunkten t_3 bis t_5 die ATP-Sollgeschwindigkeitsfrequenz von f_{ATP3} auf f_{ATP5} ändert, werden vom Vergleicher **20** als Bremsbefehlsignale **21** die Signale B_3 bis B_5 ausgegeben. Diese Bremsbefehlsignale **21** werden der Bremseinrichtung **22** zugeführt, und die Bremseinrichtung **22** gibt die jeweils vorgegebene Bremskraft an das elektrische Motorfahrzeug aus.

[0203] In der **Fig.** 1 werden die Bremsbefehlsignale **21** B_1 bis B_5 seriell vom Vergleicher **20** wie in der **Fig.** 3 gezeigt ausgegeben, und die Bremseinrichtung **22** dekodiert sie. Die Bremseinrichtung **22** ist so gesteuert, daß die in der **Fig.** 4 gezeigte Bremskraft in Wirkung tritt.

[0204] Wie oben erwähnt ist die ATP-Einrichtung eine Überwachungseinrichtung zum Abgeben einer Bremskraft an ein elektrisches Motorfahrzeug durch ein ATP-Sollgeschwindigkeitssignal vom Boden, Steuern der Geschwindigkeit des elektrischen Motorfahrzeugs und Anhalten des elektrischen Motorfahrzeugs an einer vorgegebenen Haltestelle.

[0205] Wenn jedoch einer der Mikrocomputer **10** und **11**, eine der Logikschaltungen **15** und **16** oder der Vergleicher **20** der ATP-Einrichtung ausfällt und der vorgegebene Bremsbefehl nicht ausgegeben wird, ergibt sich eine ernste Situation, so daß es erforderlich ist, das elektrische Motorfahrzeug immer sicher zu stoppen, wenn in einer der genannten Schaltungen und Einrichtungen ein Fehler erfaßt wird.

[0206] Im folgenden wird die Fehlererfassungsfunktion erläutert.

[0207] Die in die Logikschaltung **15** des Systems A eingegebenen Steuerdaten **12** werden vom Fehlerdetektor **153** geprüft, und das Ausgangssignal **154** davon wird in den Fehlerdetektor des Systems B eingegeben.

[0208] Die in die Logikschaltung **16** des Systems B eingegebenen Steuerdaten **13** werden vom Fehlerdetektor **163** geprüft, und das Vorzeichen des Ausgangssignals **164** davon wird durch den Inverter **19** invertiert, und das Signal wird als Fehlererfassungssignal **165** mit invertiertem Vorzeichen in den Fehlerdetektor **153** des Systems A eingegeben.

[0209] Es wird in diesem Fall angenommen, daß der Signalspannungspegel, wenn die Schaltung nicht in Betrieb ist, L ist, und daß der Signalspannungspegel, wenn die Schaltung in Betrieb ist, H ist.

[0210] Wenn die Steuerdaten **12** und **13**, die in die Logikschaltungen **15** und **16** der Systeme A und B eingegeben werden, normal sind, liegen die Ausgangssignale **154** und **164** der Fehlerdetektoren **153** und **163** auf L. Das Vorzeichen des Ausgangssignals **164** des Fehlerdetektors **163** des Systems B wird durch den Inverter **19** invertiert, so daß das Fehlererfassungssignal **165** mit invertiertem Vorzeichen zu H wird.

[0211] Von den in den Fehlerdetektor **153** der Logikschaltung **15** eingegebenen CRC-Daten wird durch das Fehlererfassungssignal **165** mit invertiertem Vorzeichen auf dem Pegel H einer der beiden CRC-Datensätze D_{CRC10} und D_{CRC11} ausgewählt und in den Fehlerdetektor **153** eingegeben, oder die zuerst eingegebenen CRC-Daten werden zu den anderen CRC-Daten geändert. Zum Beispiel wird D_{CRC10} kontrolliert zu D_{CRC11} geändert.

[0212] Vorzugsweise sind die in den einzelnen CRC-Datensätzen enthaltenen Daten in den beiden Systemen A und B gleich. Bei der vorliegenden Ausführungsform ist es eine vorgegebene Bedingung, daß alle Daten gleich sind. Das heißt, daß die Daten DI_0 bis DI_{x-1} in den CRC-Daten D_{CRC10}, D_{CRC10}, D_{CRC20} und D_{CRC21} die gleichen sind.

[0213] Die in den Fehlerdetektor **163** der Logikschaltung **16** eingegebenen CRC-Daten ändern sich jedoch nicht zu den anderen CRC-Daten, da das Fehlererfassungssignal **154** im L-Zustand ist.

[0214] Die Verarbeitung beginnt in diesem Zustand.

[0215] Es wird im vorliegenden Fall angenommen, daß die CRC-Daten der Steuerdaten **12** und **13**, die in jedem der Fehlerdetektoren zuerst ausgewählt werden, im System A D_{CRC10} sind und im System B D_{CRC20} sind.

[0216] Die **Fig.** 5 zeigt die Beziehung der Operationswellen zwischen der Anzahl von Operationen, dem Ausgangssignal **154** des Fehlerdetektors **153** und dem Ausgangssignal **164** des Fehlerdetektors **163**.

[0217] Bei der ersten Operation werden die Steuerdaten **12** vom Mikrocomputer **10** der Logikschaltung **15** des Systems A zugeführt, und das ATP-Sollgeschwindigkeitssignal wird durch die Logikschaltung **15** in die vorgegebene Sollgeschwindigkeitsfrequenz f_{ATPn} umgewandelt und als Ausgangssignal **17** ausgegeben.

[0218] Aus den CRC-Daten dieser Steuerdaten **12** wird D_{CRC11}, das nicht korrekte CRC-Daten darstellt, für den Fehlerdetektor **153** durch das Fehlererfassungssignal **165** mit invertiertem Vorzeichen auf dem Pegel H des Systems B ausgewählt. Im Fehlerdetektor **153** wird daher ein Fehler erfaßt, und das Fehlererfassungssignal **154** geht auf H.

[0219] Andererseits werden die Steuerdaten **13** vom Mikrocomputer **11** der Logikschaltung **16** des Systems B zugeführt, und das ATP-Sollgeschwindigkeitssignal wird von der Logikschaltung **16** in die vorgegebene Sollgeschwindigkeitsfrequenz f_{ATPn} umgewandelt und als Ausgangssignal **18** ausgegeben. Die CRC-Daten dieser Steuerdaten **13** bleiben unverändert D_{CRC20}, da das Fehlererfassungssignal **154** des Systems im vorherigen Zustand ist, das heißt im Zustand L. Das Fehlererfassungssignal **164** des Fehlerdetektors **163** bleibt daher unverändert auf L, und das Fehlererfassungssignal **165** mit umgekehrtem Vorzeichen geht in den H-Zustand über.

[0220] Am Ende der ersten Operation liegt das Fehlererfassungssignal **154** des Systems A damit auf H und das Fehlererfassungssignal **164** auf L.

[0221] Bei der zweiten Operation wird das ATP-Soll-

geschwindigkeitssignal in die vorgegebene Sollgeschwindigkeitsfrequenz f_{ATPn} umgewandelt, und die Ausgangssignale **17** und **18** bleiben unverändert. Für die CRC-Daten der Steuerdaten **12**, die in die Logikschaltung **15** des Systems A eingegeben werden, wird vom H-Pegel-Fehlererfassungssignal **165** mit invertiertem Vorzeichen des Systems B jedoch D_{CRC11} ausgewählt, und das Fehlererfassungssignal **154** der Logikschaltung **15** wird auf dem H-Zustand gehalten, wie es oben erwähnt wurde.

[0222] Für die CRC-Daten der Steuerdaten **13**, die in die Logikschaltung **16** des Systems B eingegeben werden, wird vom Fehlererfassungssignal **154**, das im H-Zustand ist, der der endgültige Zustand der ersten Operation des Systems A ist, D_{CRC21} ausgewählt.

[0223] Im Fehlerdetektor **163** wird daher ein Fehler erfaßt, und das Fehlererfassungssignal **164** ändert sich von L auf H und das Fehlererfassungssignal **165** mit invertiertem Vorzeichen von H auf L.

[0224] Am Ende der zweiten Operation liegt damit das Fehlererfassungssignal **154** des Systems A auf H und das Fehlererfassungssignal **164** auch auf H.

[0225] Bei der dritten Operation wird das ATP-Sollgeschwindigkeitssignal in die vorgegebene Sollgeschwindigkeitsfrequenz f_{ATPn} umgewandelt, und die Ausgangssignale **17** und **18** bleiben unverändert. Die CRC-Daten des Systems A ändern sich jedoch von D_{CRC11} auf D_{CRC10}, da das Fehlererfassungssignal **165** mit invertiertem Vorzeichen des Systems B auf L liegt.

[0226] Das Fehlererfassungssignal **154** der Logikschaltung **15** ändert sich daher von H auf L.

[0227] Für die CRC-Daten der Steuerdaten **13**, die in die Logikschaltung **16** des Systems B eingegeben werden, wird vom Fehlererfassungssignal **154**, das im H-Zustand ist, der der endgültige Zustand der zweiten Operation des Systems A ist, D_{CRC21}, ausgewählt.

[0228] Im Fehlerdetektor **163** wird daher ein Fehler erfaßt, und das Fehlererfassungssignal **164** bleibt im H-Zustand, und das Fehlererfassungssignal **165** mit invertiertem Vorzeichen bleibt ebenfalls unverändert im L-Zustand.

[0229] Am Ende der dritten Operation liegt damit das Fehlererfassungssignal **154** des Systems A auf L und das Fehlererfassungssignal **164** auf H.

[0230] Bei der vierten Operation wird das ATP-Sollgeschwindigkeitssignal in die vorgegebene Sollgeschwindigkeitsfrequenz f_{ATPn} umgewandelt, und die Ausgangssignale **17** und **18** bleiben unverändert. Da das Fehlererfassungssignal **165** mit invertiertem Vorzeichen des Systems B, das in die Logikschaltung **15** des Systems A eingegeben wird, auf L liegt, bleiben die CRC-Daten unverändert auf D_{CRC10}

[0231] Das Fehlererfassungssignal **154** der Logikschaltung **15** bleibt daher unverändert auf L.

[0232] Für die CRC-Daten der Steuerdaten **13**, die in die Logikschaltung **16** des Systems B eingegeben werden, wird D_{CRC20} ausgewählt, da das Fehlererfassungssignal **154** des Systems A im L-Zustand ist, der der endgültige Zustand der dritten Operation ist.

[0233] Das Fehlererfassungssignal **164** ändert sich damit von H auf L, und das Fehlererfassungssignal **165** mit invertiertem Vorzeichen ändert sich von L auf H.

[0234] Am Ende der vierten Operation liegt das Fehlererfassungssignal **154** des Systems A auf L und das Fehlererfassungssignal **164** auf L.

[0235] Bei der fünften Operation wird das ATP-Sollgeschwindigkeitssignal in die vorgegebene Sollgeschwindigkeitsfrequenz f_{ATPn} umgewandelt, und die Ausgangssignale **17** und **18** bleiben unverändert. Da das Fehlererfassungssignal **165** mit invertiertem Vorzeichen des Systems B, das in die Logikschaltung **15** des Systems A eingegeben wird, auf H liegt, wird für die CRC-Daten D_{CRC11} ausgewählt.

[0236] Das Fehlererfassungssignal **154** der Logikschaltung **15** ändert sich damit von L auf H.

[0237] Für die CRC-Daten der Steuerdaten **13**, die in die Logikschaltung **16** des Systems B eingegeben werden, wird D_{CRC20} ausgewählt, da das Fehlererfassungssignal **154** des Systems A im L-Zustand ist, der der endgültige Zustand der vierten Operation ist. Im Ergebnis wird das Fehlererfassungssignal **164** des Fehlerdetektors **163** im L-Zustand gehalten, und das Fehlererfassungssignal **165** mit invertiertem Vorzeichen wird im H-Zustand gehalten.

[0238] Der Prozeß kehrt zum Status der ersten Operation zurück, und die obigen Operationen werden wiederholt.

[0239] Bei dem genannten Beispiel erfolgt die Steuerung so, daß, wenn das Fehlererfassungssignal **154** und das Fehlererfassungssignal **165** mit invertiertem Vorzeichen im L-Zustand sind, im System A D_{CRC10} ausgewählt wird und im System B D_{CRC20}, wenn sie im H-Zustand sind, im System A D_{CRC11} ausgewählt wird und im System B D_{CRC21}. Die Beziehung zwischen dem Fehlererfassungssignal und der Steuerung der Auswahl der CRC-Daten kann in Abhängigkeit von der Anwendung verändert werden.

[0240] Hier wurde zwar angenommen, daß D_{CRC10} im System A und D_{CRC20} im System B die korrekten CRC-Daten sind und D_{CRC11} im System A und D_{CRC21} im System B die nicht korrekten CRC-Daten sind; es kann jedoch auch umgekehrt vorgegangen werden. In diesem Fall umfaßt korrekt und nicht korrekt, daß die erzeugenden Polynomausdrücke der CRC-Prüfdaten anders sind.

[0241] Wie erwähnt werden die CRC-Daten D_{CRC2} des Systems B durch das Fehlererfassungssignal **154** der Logikschaltung **15** des Systems A kontrolliert und die CRC-Daten D_{CRC1}, des Systems A durch das Fehlererfassungssignal **165** mit invertiertem Vorzeichen bzw. das Fehlererfassungssignal **164** der Logikschaltung **16** des Systems B.

[0242] Wenn alle Schaltungen wie die Mikrocomputer **10** und **11** und die Logikschaltungen **15** und **16** und die Steuerdaten normal sind, alternieren daher die Fehlererfassungssignale **154** und **164** mit einer festen Periode.

[0243] Die Logikschaltung **15**, das Fehlererfassungssignal **154**, die Logikschaltung **16** und das Fehlererfassungssignal **164** bilden eine Fehlererfassungsschleife, und ein Fehler in dieser Schleife kann durch ein Verfahren zum Überwachen eines der Fehlererfassungssignale **154** und **164** erfaßt werden.

[0244] Es kann jedoch kein Fehler erfaßt werden, der in einem Abschnitt außerhalb dieser Fehlererfassungsschleife auftritt. Um zu verhindern, daß aufgrund eines Versagens in einem Abschnitt außerhalb der Fehlererfassungsschleife ein falsches Signal ausgegeben wird oder ein richtiges Signal nicht ausgegeben wird, werden das Fehlererfassungssignal **154** des Systems A und das Fehlererfassungssignal **164** des Systems B durch den Vergleicher **20** abgeglichen.

[0245] Die **Fig.** 4 zeigt, daß ein elektrisches Motorfahrzeug an einer vorgegebenen Stelle anhält, wobei es aufgrund eines Bremsbefehls B_5 an der vorgegebenen Stelle anhält, und es ist der Bremsbefehl **21** und die Bremskraft gezeigt, wenn die Bremse zu Beginn zum Beispiel durch einen Ausfall der Logikschaltung **15** des Systems A ausgeschaltet wird.

[0246] Wenn das Alternieren des Fehlererfassungssignals **154** des Systems A und des Fehlererfassungssignals **164** des Systems B endet, endet auch das Alternieren eines Fehlererfassungsabgleich-Ausgangssignals **24**. Im Ergebnis wird von der Bremseinrichtung **21** ein Notbremssignal EB ausgegeben, durch das die maximale Bremskraft an das elektrische Motorfahrzeug angelegt wird.

[0247] Dieses Notbremssignal EB wirkt, wie in der **Fig.** 4 gezeigt, vor den anderen Bremsbefehlen B_1 bis B_5.

[0248] Durch das Überwachen des Alternierens eines der Fehlererfassungssignale **154** und **164** ist es möglich festzustellen, welches der doppelt vorhandenen Systeme A und B ausgefallen ist, und der Fehler kann schnell analysiert werden, und es läßt sich ein wirkungsvolles störungssicheres System aufbauen.

[0249] Die **Fig.** 6 zeigt einen störungssicheren 1-Bit-Vergleicher und die **Fig.** 7 die Operationswellenformen davon.

[0250] In der **Fig.** 6 bezeichnet das Bezugszeichen **100** einen Vorzeicheninverter, **101** ein erstes Flip-Flop, **102** ein zweites Flip-Flop, **103** ein exklusives ODER-Gatter, **104** ein Abgleich-Ausgangssignal, **154** das Fehlererfassungssignal der Logikschaltung **15** der **Fig.** 1 und **164** das Fehlererfassungssignal der Logikschaltung **16** der **Fig.** 1.

[0251] Da die Fehlererfassungssignale **154** und **164** mit einer festen Periode alternieren, wird das Fehlererfassungssignal **154** an den Takteingang CK_1 des ersten Flip-Flops **101** angelegt.

[0252] Wenn das Vorzeichen des Fehlererfassungssignals **164** vom Inverter **100** invertiert wird und das Signal dann an den Takteingang CK_2 des zweiten Flip-Flops **102** angelegt wird, ergeben sich ein Ausgangssignal Q_1 des ersten Flip-Flops **101** und ein negatives (invertiertes) Ausgangssignal Q_2 des zweiten Flip-Flops **102**, wie es in der **Fig.** 6 gezeigt ist.

[0253] Wenn das Ausgangssignal Q_1 des ersten Flip-Flops **101** und das negative Ausgangssignal Q_2 des zweiten Flip-Flops **102** an das exklusive ODER-Gatter **103** angelegt werden, wie es in der **Fig.** 6 gezeigt ist, wird ein exklusives ODER-Ausgangssignal **104** erhalten.

[0254] Wenn das Alternieren eines der Fehlererfassungssignale **154** und **164** endet oder einer der Flip-Flops ausfällt und der Ausgangsanschluß davon auf H oder L stehenbleibt, endet auch das Alternieren des Ausgangssignals **104** des exklusiven ODER-Gatters **103**.

[0255] Da der Vergleicher immer zwei Signale abgleicht, alterniert sein Ausgang im normalen Zustand immer. Durch Überwachen des Ausgangssignals **104** kann daher nicht nur hinsichtlich der Schleife der Systeme A und B, sondern der gesamten ATP-Einrichtung festgestellt werden, ob sie normal oder unnormal arbeiten. Anders gesagt ist es vorteilhaft, nur dieses Ausgangssignal **104** zu überwachen.

[0256] Die **Fig.** 6 zeigt einen störungssicheren Vergleicher zum Abgleichen eines 1-Bit-Ausgangs. Um eine Vielzahl von Bits störungssicher abzugleichen, ist es vorteilhaft, nur die Abgleichschaltung vorzusehen, die in der **Fig.** 6 gezeigt ist. Der Vergleicher der **Fig.** 1 ist dahingehend der Schaltung der **Fig.** 6 gleichwertig, daß er eine Anzahl von eingebauten Bits aufweist.

[0257] Es ist vorteilhaft, wenn eine Steuerung oder ein System zur Aufnahme eines Ausgangssignals von der ATP-Einrichtung einen Aufbau zur Ausgabe eine Notsteuersignals aufweist, wenn die Ausgabe von alternierenden Signalen des Vergleichers **20** endet, so daß es möglich ist, ein störungssicheres System aufzubauen.

[0258] Das Vorstehende ist eine Erläuterung einer Ausführungsform, bei der der Bremsbefehl **21** des Vergleichers **20** seriell ausgegeben wird. Es ist jedoch auch möglich, die Signale B_1, B_2, B_3, B_4 und B_5 des Bremsbefehls **21** parallel an die Bremseinrichtung **22** auszugeben. Mit anderen Worten ist es vorteilhaft, wenn der Vergleicher **20** so aufgebaut ist, daß der Befehl ein Signal für die Bremseinrichtung **22** ist.

[0259] Wie erwähnt ist es erforderlich, eine bestimmte Speicherlese-Signalleitung, eine Schreibsignal-Steuerleitung und eine Adressensignalerzeugungsleitung im Logikelement vorzusehen, das das Steuergerät **14** bildet, wenn der Speicher für die Steuerdaten nicht in den Mikrocomputer eingebaut ist, sondern in der ATP-Einrichtung oder unabhängig installiert ist. Es ist jedoch auch möglich, daß der Mikrocomputer und das Steuergerät über den Speicher nur Informationen übertragen und erhalten, und der Mikrocomputer kann dann für andere Steueraufgaben verwendet werden, so daß die Auslastung des Mikrocomputers erhöht werden kann.

[0260] Bei der vorliegenden Ausführungsform wurde ein Beispiel beschrieben, bei dem für eine Logik-

schaltung zwei CRC-Datensätze vorgesehen sind, zwischen denen durch ein Fehlererfassungssignal von einer anderen Logikschaltung umgeschaltet wird. Wie in der **Fig.** 18 erwähnt, ist es jedoch leicht möglich, daß der Inhalt der Daten nicht geprüft wird und nur alternierende Signale erzeugt werden und nur ausgegeben wird, ob die beiden Logikschaltungen normal arbeiten oder nicht.

[0261] Anhand der **Fig.** 8 wird nun eine weitere Ausführungsform beschrieben.

[0262] In der **Fig.** 8 ist wie in der **Fig.** 1 der gleiche Schaltkreis als duplizierte Systeme A und B auf einem Chip ausgebildet. Wie erwähnt ist die ATP-Einrichtung eine Überwachungseinrichtung, die für den sicheren Betrieb eines elektrischen Motorfahrzeugs erforderlich ist, und es darf nie der Fall eintreten, daß aufgrund eines Versagens in der ATP-Einrichtung ein falsches Signal ausgegeben wird oder das vorgegebene Signal nicht ausgegeben wird. Die Fehlererfassungsfunktion ist daher bei der Ausführungsform der **Fig.** 8 weiter ausgebildet.

[0263] In der **Fig.** 8 haben gleiche Codes wie in der **Fig.** 1 die gleiche Funktion, so daß diesbezüglich keine erneute Erläuterung erfolgt.

[0264] In der **Fig.** 8 bezeichnet das Bezugszeichen **30** einen ersten Speicher, der in der Logikschaltung **15** des Systems A untergebracht ist, **31** einen zweiten Speicher, **32** CRC-Daten der Steuerdaten **120**, die im ersten Speicher **30** gespeichert sind, **33** CRC-Daten der Steuerdaten **121**, die im zweiten Speicher **31** gespeichert sind, **34** einen Umschalter für den ersten Speicher **30** und den zweiten Speicher **31**, **35** das Ausgangssignal des Speicherumschalters **34**, **36** eine erste CRC-Prüfschaltung für die Steuerdaten **120**, die im ersten Speicher **30** gespeichert sind, **37** eine zweite CRC-Prüfschaltung für die Steuerdaten **121**, die im zweiten Speicher **31** gespeichert sind, **38** eine erste ODER-Schaltung zum Addieren des Ausgangssignals **35** des Umschalters **34** und des Ausgangssignals der zweiten CRC-Prüfschaltung **37**, und **39** eine zweite ODER-Schaltung zum Addieren des Ausgangssignals der ersten CRC-Prüfschaltung **36** und der Geschwindigkeitsfrequenz **5**, die der gegenwärtigen Geschwindigkeit des elektrischen Motorfahrzeugs entspricht.

[0265] Das Bezugszeichen **40** bezeichnet einen ersten Speicher, der in der Logikschaltung **16** des Systems B untergebracht ist, **41** einen zweiten Speicher, **42** CRC-Daten der Steuerdaten **130**, die im ersten Speicher **40** gespeichert sind, **43** CRC-Daten der Steuerdaten **131**, die im zweiten Speicher **41** gespeichert sind, **44** einen Umschalter für den ersten Speicher **40** und den zweiten Speicher **41**, **45** das Ausgangssignal des Speicherumschalters **44**, **46** eine erste CRC-Prüfschaltung für die Steuerdaten **130**, die im ersten Speicher **40** gespeichert sind, **47** eine zweite CRC-Prüfschaltung für die Steuerdaten **131**, die im zweiten Speicher **41** gespeichert sind, **48** eine erste ODER-Schaltung zum Addieren des Ausgangssignals **45** des Umschalters **44** und des Ausgangssignals der zweiten CRC-Prüfschaltung **47**, und **49** eine zweite ODER-Schaltung zum Addieren des Ausgangssignals der ersten CRC-Prüfschaltung **46** und der Geschwindigkeitsfrequenz **5**, die der gegenwärtigen Geschwindigkeit des elektrischen Motorfahrzeugs entspricht.

[0266] Das ATP-Sollgeschwindigkeitssignal **8** wird von den Mikrocomputern **10** und **11** verarbeitet, und die entsprechenden Steuerdaten werden in die Logikschaltungen der Systeme A und B des 1-Chip-Logikelements **14** eingegeben.

[0267] Die ersten Steuerdaten **120** vom Mikrocomputer **10** werden im ersten Speicher **30** des Systems A gespeichert und die zweiten Steuerdaten **121** im zweiten Speicher **31** des Systems A, und die dritten Steuerdaten **130** vom Mikrocomputer **11** werden im ersten Speicher **40** des Systems B gespeichert und die zweiten Steuerdaten **131** im zweiten Speicher **41** des Systems B.

[0268] Die **Fig.** 9 zeigt die Daten, die in den Speichern **30**, **31**, **40** und **41** von den Mikrocomputern **10** und **11** gespeichert werden.

[0269] Der Mikrocomputer **10** wandelt das ATP-Sollgeschwindigkeitssignal **5** in die n-Wort-Daten D_{0X} bis D_{14Y} um, verarbeitet und erhält die CRC-Daten **32** (die Daten D_{CRCX1}) der Daten D_{0X} bis D_{14Y}, wobei der erzeugende Polynomausdruck dabei auf G0(X) gesetzt ist, und speichert die Daten in (1) der **Fig.** 9, wobei die CRC-Daten **32** (die Daten D_{CRCX1}) an den Adressen A0S bis A15S des ersten Speichers **30** zu den Daten D_{0X} bis D_{14X} hinzugefizgt werden.

[0270] Der Mikrocomputer **10** verarbeitet und gewinnt die CRC-Daten **33** (die Daten D_{CRCX2}) der Daten D_{0X} bis D_{14X}, wobei der erzeugende Polynomausdruck dabei auf G1(X) gesetzt ist, und speichert die Daten in (2) der **Fig.** 9, wobei die CRC-Daten **33** (die Daten D_{CRCX2}) an den Adressen A0T bis A15T des zweiten Speichers **31** zu den Daten D_{0X} bis D_{14X} hinzugefügt werden.

[0271] Der Mikrocomputer **11** wandelt das ATP-Sollgeschwindigkeitssignal **5** in die n-Wort-Daten D_{0Y} bis D_{14Y} um, verarbeitet und erhält die CRC-Daten **42** (die Daten D_{CRCY1}) der Daten D_{0Y} bis D_{14Y}, wobei der erzeugende Polynomausdruck dabei auf G2(X) gesetzt ist, und speichert die Daten in (3) der **Fig.** 9, wobei die CRC-Daten **42** (die Daten D_{CRCY1}) an den Adressen A0U bis A15U des ersten Speichers **40** zu den Daten D_{0Y} bis D_{14Y} hinzugefügt werden.

[0272] Der Mikrocomputer **11** verarbeitet und gewinnt die CRC-Daten **43** (die Daten D_{CRCY2}) der Daten D_{0Y} bis D_{14Y}, wobei der erzeugende Polynomausdruck dabei auf G3(X) gesetzt ist, und speichert die Daten in (4) der **Fig.** 9, wobei die CRC-Daten **43** (die Daten D_{CRCY2}) an den Adressen A0V bis A15V des zweiten Speichers **41** zu den Daten D_{0Y} bis D_{14Y} hinzugefügt werden.

[0273] Die unter den Adressen A0S bis Al5S des ersten Speichers **30** des Systems A und den Adressen A0T bis A15T des zweiten Speichers **31** gespeicherten Daten werden von parallelen Daten in seriel-

le Daten umgewandelt, aufeinanderfolgend von den Bits niedriger Ordnung bis zu den Bits hoher Ordnung ausgelesen und in den Umschalter **34** eingegeben.

[0274] Auf die gleiche Weise werden die unter den Adressen A0U bis A15U des ersten Speichers **40** des Systems B und den Adressen A0V bis A15V des zweiten Speichers **41** gespeicherten Daten von parallelen Daten in serielle Daten umgewandelt, aufeinanderfolgend von den Bits niedriger Ordnung bis zu den Bits hoher Ordnung ausgelesen und in den Umschalter **17** eingegeben.

[0275] Die Periode vom Auslesen der Adressen A0n bis A15n bis zum Verarbeiten durch die Frequenzkomparatoren **152** und **162** ist eine Verarbeitungsperiode.

[0276] Die erste CRC-Prüfschaltung **36** des Systems A ist eine Schaltung, die dem erzeugenden Polynomausdruck G0(X) entspricht, und die zweite CRC-Prüfschaltung **37** ist eine Schaltung, die dem erzeugenden Polynomausdruck G1(X) entspricht.

[0277] Die dritte CRC-Prüfschaltung **46** des Systems B ist eine Schaltung, die dem erzeugenden Polynomausdruck G2(X) entspricht, und die vierte CRC-Prüfschaltung **47** ist eine Schaltung, die dem erzeugenden Polynomausdruck G3(X) entspricht.

[0278] Es wird angenommen, daß das Fehlererfassungssignal **154** des Systems A und das Fehlererfassungssignal **164** des Systems B zuerst im L-Zustand sind. Das Fehlererfassungssignal **165** mit invertiertem Vorzeichen des Systems B ist daher auf H.

[0279] Durch dieses Fehlererfassungssignal **165** mit invertiertem Vorzeichen des Systems B im H-Zustand wird der Umschalter **34** des Systems A auf die Seite zum Auslesen der Daten des ersten Speichers **30** geschaltet, und der Umschalter **44** des Systems B wird durch das Signal **154** des Systems A mit invertiertem Vorzeichen im L-Zustand auf die Seite zum Auslesen der Daten des zweiten Speichers **41** geschaltet, und es wird angenommen, daß alle Daten und die interne Schaltung normal sind.

[0280] Die Daten D_{0X} bis D_{14X} an den Adressen A0S bis A14S des ersten Speichers **30** des Systems A und die Daten D_{0X} bis D_{14X} an den Adressen A0T bis A14T des zweiten Speichers **31** werden über den Umschalter **34** und die erste Logikschaltung **38** in den Frequenzkonverter **150** eingegeben und entsprechend den jeweiligen Daten in die ATP-Sollgeschwindigkeitsfrequenz f_{rn} umgewandelt, während die Daten D_{0X} bis D_{14X} an den Adressen A0S bis A14S und A0T bis A14T in die erste CRC-Prüfschaltung **36** und die zweite CRC-Prüfschaltung **37** eingegeben werden und durch die CRC-Daten D_{CRCX1} und D_{CRCX2} an den Adressen A15S und A15T geprüft werden.

[0281] Der Aufbau ist also derart, daß kein Zwischenprüfergebnis für die erste Prüfschaltung **36** und die zweite Prüfschaltung **37** ausgegeben wird, bevor nicht alle die CRC-Daten D_{CRCX1} und D_{CRCX2} an den Adressen A15S und A15T ausgelesen und geprüft wurden.

[0282] Die CRC-Daten D_{CRCX1} und D_{CRCX2} an den Adressen A15S und A15T sind so strukturiert, daß sie nicht frequenzkonvertiert werden.

[0283] Auf die gleiche Weise werden die Daten D_{0Y} bis D_{14Y} an den Adressen A0U bis A14U des ersten Speichers **40** des Systems B und die Daten D_{0Y} bis D_{14Y} an den Adressen A0V bis A14V des zweiten Speichers **41** über den Umschalter **44** und die erste Logikschaltung **48** in den Frequenzkonverter **160** eingegeben und entsprechend den jeweiligen Daten in die ATP-Sollgeschwindigkeitsfrequenz f_{rn} umgewandelt, während die Daten D_{0Y} bis D_{14Y} an den Adressen A0U bis A14U und A0V bis A14V in die erste CRC-Prüfschaltung **46** und die zweite CRC-Prüfschaltung **47** eingegeben werden und durch die CRC-Daten D_{CRCY1} und D_{CRCY2} an den Adressen A15U und A15V geprüft werden.

[0284] Der Aufbau ist also derart, daß kein Zwischenprüfergebnis für die erste Prüfschaltung **46** und die zweite Prüfschaltung **47** ausgegeben wird, bevor nicht alle die CRC-Daten D_{CRCY1} und D_{CRCY2} an den Adressen A15U und A15V ausgelesen und geprüft wurden.

[0285] Die CRC-Daten D_{CRCY1} und D_{CRCY2} an den Adressen A15U und A15V sind so strukturiert, daß sie nicht frequenzkonvertiert werden.

[0286] Zuerst werden die Ein-Wort-Daten an der Adresse A0S des ersten Speichers **30** des Systems A über die ODER-Schaltung in den Frequenzkonverter **150** eingegeben und entsprechend den Ein-Wort-Daten in die ATP-Sollgeschwindigkeitsfrequenz f_{r0} umgewandelt.

[0287] Da die erste CRC-Prüfschaltung **36** und die zweite CRC-Prüfschaltung **37** die Daten prüfen, sind die Ausgangssignale davon auf L.

[0288] Dann werden die Ein-Wort-Daten an der Adresse A1S über die ODER-Schaltung **38** in den Frequenzkonverter **150** eingegeben und entsprechend den Ein-Wort-Daten an der Adresse A1S in die ATP-Sollgeschwindigkeitsfrequenz f_{r1} umgewandelt. Die Eingangsfrequenz f_{i1} die ein Ausgangssignal der ODER-Schaltung **39** ist, ist L.

[0289] Danach werden auf die gleiche Weise die Ein-Wort-Daten bis hinauf zur Adresse A14S entsprechend den jeweiligen Daten in die ATP-Sollgeschwindigkeitsfrequenz f_{r14} umgewandelt.

[0290] Die Ein-Wort-CRC-Daten D_{CRCX1} (die Daten vom erzeugenden Polynomausdruck GO(X)) an der letzten Adresse A15S werden von der ersten CRC-Prüfschaltung **36** (der Schaltung, die dem erzeugenden Polynomausdruck GO(X) entspricht) als normal betrachtet, so daß das Ausgangssignal davon auf L ist. Da jedoch die Ein-Wort-CRC-Daten D_{CRCX1} von der zweiten CRC-Prüfschaltung **37** (der Schaltung, die dem erzeugenden Polynomausdruck G1(X) entspricht), als unnormal betrachtet werden, ist das Ausgangssignal davon auf H.

[0291] Das Ausgangssignal dieser zweiten CRC-Prüfschaltung wird über die ODER-Schaltung **38** als Maximaldaten in den Frequenzkonverter **150** eingegeben und in die Maximalfrequenz f_{r15} umge-

wandelt.

[0292] Andererseits ist die Eingangsfrequenz f_{i15}, die ein Ausgangssignal der ODER-Schaltung **39** ist, auf L.

[0293] Der Frequenzkomparator **152** vergleicht die den Daten entsprechende Frequenz f_{rn} an jeder Adresse, die auf diese Weise erhalten wird, mit der Eingangsfrequenz f_{in}.

[0294] Im Frequenzkomparator **152** wird, wenn die Beziehung zwischen der ATP-Sollgeschwindigkeitsfrequenz frn und der Additionsfrequenz f_{in} gleich $f_{rn} > f_{in}$ ist, zu dem internen Zählwert 1 hinzuaddiert (in der Zeichnung nicht gezeigt), und wenn die Beziehung $f_{rn} < n$ ist, wird 1 vom Zählwert subtrahiert, und wenn $f_{rn} = f_{in}$ ist, wird nichts zum Zählwert addiert und nichts davon subtrahiert.

[0295] In der ersten Verarbeitungsperiode wird entsprechend den Adressen A0S bis A14S 1 zum Zählwert hinzuaddiert. In der zweiten Verarbeitungsperiode werden die Daten erneut aus der Adresse A0S ausgelesen und der Zählwert des Frequenzkomparators **152** auf die gleiche Weise wie oben hochgezählt.

[0296] Wenn die vierte Verarbeitungsperiode verstrichen ist und der Zählwert des Frequenzkomparators **152** den Wert 4 erreicht hat, wird das Ausgangssignal **17** entsprechend den einzelnen Adressen von A0S bis A15S erhalten.

[0297] Die **Fig.** 10 zeigt die Beziehung zwischen der Operation des Zählers des Frequenzkomparators **152** aufgrund des CRC-Prüfergebnisses an der Adresse A15S des Systems A, der Operation des Zählers des Frequenzkomparators **162** aufgrund des CRC-Prüfergebnisses an der Adresse A15U des Systems B und das Vorzeicheninversionssignal **165**, das das Vorzeichen des Fehlererfassungssignals **164** invertiert.

[0298] Wenn der Zählwert durch einen Hochzählimpuls den Wert 4 erreicht, wird das Ausgangssignal **17** des Frequenzkomparators **152** erhalten. Da dieses Ausgangssignal **17** seriell ausgegeben wird, erfaßt der Frequenzkomparator **152** das Fehlererfassungssignal 154 im H-Zustand entsprechend der Adresse A15S synchron zum Zeitgebersignal (in der Zeichnung nicht gezeigt) und gibt es in den Speicherumschalter **44** des Systems B.

[0299] Der Umschalter **44** schaltet zeitlich exakt zum Beginn der nächsten Verarbeitungsperiode vom Speicher **41** auf den Speicher 40 um, so daß der Speicher des Systems B genau zu Beginn der sechsten Verarbeitungsperiode umgeschaltet wird.

[0300] Die an den Adressen A0U bis A14U des Speichers **40** gespeicherten Daten sind exakt die gleichen wie die an den Adressen A0V bis A14V des Speichers **41** gespeicherten Daten, so daß auch dann, wenn vom Speicher **41** auf den Speicher **40** umgeschaltet wird, sich die ATP-Sollgeschwindigkeitsfrequenz f_{rn} nicht ändert und sich auch die Eingangsfrequenz f_{in} nicht ändert.

[0301] Die Ausgangsfrequenz des Frequenzkonverters **160** ändert sich daher nicht. Nur die CRC-Daten **42** (D_{CRCY1}), die an der Adresse A15U gespeichert sind, und die CRC-Daten **43** (D_{CRCY2}), die an der Adresse A15V gespeichert sind, sind verschieden, so daß die Operation beschrieben wird, wenn die Daten an den Adressen A15U und A15V ausgelesen werden.

[0302] Wenn vom Speicher **41** auf den Speicher **40** umgeschaltet wird, werden die CRC-Daten **42** (die Daten vom erzeugenden Polynomausdruck) ausgelesen. Da die CRC-Daten **42** von der ersten CRC-Prüfschaltung **46** (der Schaltung, die dem erzeugenden Polynomausdruck G2(X) entspricht) als normal betrachtet werden, ist das. Ausgangssignal davon auf L. Da sie jedoch von der zweiten CRC-Prüfschaltung **47** (der Schaltung, die dem erzeugenden Polynomausdruck G3(X) entspricht), als unnormal betrachtet werden, ist das Ausgangssignal davon auf H.

[0303] Da das Ausgangssignal der ersten CRC-Prüfschaltung **46** auf L ist, ist das Ausgangssignal der ODER-Schaltung **49** ebenfalls auf L, und die Eingangsfrequenz f_{i15} ist gleichfalls auf L.

[0304] Andererseits wird das H-Zustands-Ausgangssignal der zweiten CRC-Prüfschaltung **47** über die ODER-Schaltung **48** in den Frequenzkonverter **160** eingegeben und in die Maximalfrequenz f_{r15} umgewandelt.

[0305] Diese Maximalfrequenz f_{r15} und die Eingangsfrequenz f_{i15} werden in den Frequenzkomparator **162** eingegeben und dort verglichen. Da die Beziehung zwischen der Maximalfrequenz f_{r15} und der Eingangsfrequenz f_{i15} gleich $f_{r15} > f_{i15}$ ist, wird der Zählwert des Frequenzkomparators **162** auf 1 hochgezählt.

[0306] Danach wird auf die gleiche Weise wie beim System A der Zählwert des Frequenzkomparators **162** hochgezählt, und wenn der Zählwert **4** erreicht, wird das Ausgangssignal **18** für die Adressen von A0U bis A15U erhalten.

[0307] Der Frequenzkomparator **162** erfaßt das Fehlererfassungssignal **164** im H-Zustand für die Adresse A15U dieses Ausgangssignals **18** synchron zum Zeitgebersignal (in der Zeichnung nicht gezeigt) und addiert das L-Zustands-Vorzeicheninversionssignal **165**, dessen Vorzeichen vom Inverter **19** invertiert wurde, in den Speicherumschalter **34** des Systems A.

[0308] Wenn das Vorzeicheninversionssignal **165** eingegeben wird, schaltet der Speicherumschalter **34** des Systems A für die nächste 11. Periode vom Speicher **30** auf den Speicher **31** um.

[0309] In den Speichern **30** und **31** sind nur die CRC-Daten **32** (D_{CRCX1}), die an der Adresse A15S gespeichert sind, und die CRC-Daten **33** (D_{CRCX2}), die an der Adresse A15T gespeichert sind, voneinander verschieden, so daß sich die Frequenz bei der Umschaltung der Speicher nicht ändert.

[0310] Da die CRC-Daten **33** des Speichers **31** Daten vom erzeugenden Polynomausdruck G1(X) sind, werden sie von der CRC-Prüfschaltung **36** als unnor-

mal betrachtet, und das Ausgangssignal davon geht auf H. Dieses Ausgangssignal wird über die zweite ODER-Schaltung **39** als Maximalfrequenz f_{i15} in den Frequenzkonverter **152** eingegeben.

[0311] Da andererseits die CRC-Daten **33** des Speichers **31** Daten vom erzeugenden Polynomausdruck G1(X) sind, werden sie von der CRC-Prüfschaltung **37** als normal betrachtet, und das Ausgangssignal davon geht auf L, das Ausgangssignal der ODER-Schaltung **38** geht auch auf L, und auch die Bezugsfrequenz f_{r15} des Frequenzkonverters **150** wird zu L.

[0312] Die Beziehung zwischen der Bezugsfrequenz f_{i15} und der Eingangsfrequenz f_{i15} ist daher $f_{r15} < f_{i15}$, und es wird ein Abwärtszählimpuls ausgegeben, so daß vom Zählwert des Frequenzkomparators **152** Eins abgezogen wird und sich der Zählwert von 4 auf 3 ändert.

[0313] Wenn der Zählwert des Frequenzkomparators **152** Null wird, geht das Ausgangssignal **17** des Frequenzkomparators **152** auf L, und das Fehlererfassungssignal **154** für die Adresse A15T wird zu L.

[0314] Durch den gleichen Vorgang wird in der Folge durch das Fehlererfassungssignal des Frequenzkomparators des eigenen Systems auf den Speicher des anderen Systems umgeschaltet.

[0315] Wenn die Speicher **30** und **31** des Systems A, die daraus ausgelesenen Daten, die beiden CRC-Prüfschaltungen **36** und **37**, die Speicher **40** und **41** des Systems B, die daraus ausgelesenen Daten und die beiden CRC-Prüfschaltungen **46** und **47** wie oben erwähnt normal sind, wechseln die Fehlererfassungssignale **154** und **164** jedesmal beim Wechseln des Speichers alternierend auf H oder L.

[0316] Wenn zum Beispiel in den Daten des Speichers **30** des Systems A ein Fehler auftritt, passiert folgendes. Der Datenfehler des Speichers **30** wird durch die CRC-Prüfschaltung **36** erfaßt, und das Ausgangssignal davon geht auf H.

[0317] Andererseits werden von Beginn an CRC-Daten von einem anderen erzeugenden Polynomausdruck in die CRC-Prüfschaltung **37** eingegeben, so daß das Ausgangssignal davon auch auf H ist. Da die Ausgangssignale beider Prüfschaltungen auf H sind, ändert sich weder der Zählwert des Frequenzkomparators **152** noch das Ausgangssignal.

[0318] Das Fehlererfassungssignal **154** für das Umschalten des Speichers **41** des Systems B auf den Speicher **41** bleibt damit fest auf H oder L, was der Zustand ist, wenn ein Fehler auftritt, und der Speicher kann nicht von **41** auf **40** umgeschaltet werden.

[0319] Im Ergebnis wird der Zählwert des Frequenzkomparators **162** nicht hochgezählt, und das Fehlererfassungssignal bleibt immer im L-Zustand. Das Fehlererfassungssignal **165** mit invertiertem Vorzeichen ist daher auf H fixiert, und es wird nicht vom Speicher **30** des Systems A auf den Speicher **31** umgeschaltet, und die Ausgangssignale der CRC-Prüfschaltungen **36** und **37** bleiben auf H.

[0320] Es ist dies ein Beispiel dafür, wenn in den Daten ein Fehler auftritt. Aber auch wenn eine der Schaltungen ausfällt, bleiben als Folge davon die Zählwerte der Frequenzkomparatoren **152** und **162** fixiert, und das Alternieren der Fehlererfassungssignale **154** und **164** endet, so daß vorzugsweise das Alternieren der Fehlererfassungssignale überwacht wird.

[0321] Wenn ein solches Verfahren zum Erzeugen eines Signals auf die ATP-Einrichtung eines Zuges angewendet wird, kann es als Mittel zum Ausgeben eines Notbremsbefehls verwendet werden, wenn das Alternieren der Fehlererfassungssignale **154** und **164** endet.

[0322] Wenn diese Schaltung eine LSI umfaßt und einschließlich der peripheren Schaltungen auf einem Chip integriert wird, läßt sich eine Miniaturisierung und eine Massenproduktion realisieren, bei der die Zuverlässigkeit der ATP-Einrichtung erhalten bleibt.

[0323] Da wie erwähnt die ATP-Einrichtung zusammen mit dem Fehlererfassungssignal **154** und dem Fehlererfassungssignal **164** eine Fehlererfassungsschleife bildet, kann durch Überwachen eines der Fehlererfassungssignale **154** und **164** ein Fehler innerhalb der Schleife erfaßt werden. Ein außerhalb der Fehlererfassungsschleife auftretender Fehler kann damit jedoch nicht erfaßt werden. Um zu verhindern, daß aufgrund eines Fehlers an einer Stelle außerhalb der Fehlererfassungsschleife ein falsches Signal ausgegeben wird oder ein richtiges Signal nicht ausgegeben wird, werden die Fehlererfassungssignale **154** und **164** vom Vergleicher **20** abgeglichen. Wenn eine Übereinstimmung vorliegt, gibt der Vergleicher **20** ein Signal aus, und wenn keine Übereinstimmung vorliegt, gibt der Vergleicher **20** ein Notsteuersignal aus.

[0324] Der Vergleicher **20** vergleicht das Ausgangssignal **17** des Frequenzkomparators **152** des Systems A und das Ausgangssignal **18** des Frequenzkomparators **162** des Systems B. Wie erwähnt vergleicht, da das Ausgangssignal **17** und das Ausgangssignal **18** seriell in der Reihenfolge der Adressensignale ausgegeben werden, der Vergleicher **20** das Signal Adresse für Adresse, und er gibt nur ein Vergleichssignal aus. Wenn keine Übereinstimmung vorliegt, kann der Vergleicher **20** ein Notsteuersignal ausgeben oder eine Anzeige an der Anzeigeeinheit.

[0325] Es läßt sich mit diesem Aufbau damit ein störungssicheres System herstellen.

[0326] Wie erwähnt werden erfindungsgemäß zwei CRC-Datensätze, bei denen verschiedene erzeugende Polynomausdrücke angewendet wurden, auf einen Datensatz angewendet, und es werden zwei Systeme von Steuerdaten, zu denen die CRC-Daten addiert werden, erstellt, und es sind zwei Arten von Prüfschaltungen für die verschiedenen erzeugenden Polynomausdrücke vorgesehen.

[0327] Das Verfahren umfaßt die gleichzeitige Eingabe der beiden Systeme von Steuerdaten in die beiden Arten von Prüfschaltungen und das Umschalten der beiden Datensysteme unter Verwendung des Er-

gebnisses, das durch einen Vergleich der Ausgangssignale der beiden Arten von Prüfschaltungen erhalten wird.

[0328] Es läßt sich damit nicht nur ein Fehler in den Daten, sondern auch ein Fehler in den Prüfschaltungen erfassen, und außerdem werden die Ausgangssignal verglichen, und nur wenn eine Übereinstimmung vorliegt, werden sie ausgegeben. Ein Vorteil ist, daß unter Verwendung eines Chips, in denen jede der Schaltungen aus einer LSI besteht, ein sichereres störungsfreies System aufgebaut werden kann.

[0329] Im Folgenden wird die Arbeitsweise des Frequenzkonverters **150** (**160**) der **Fig.** 1 und 8 erläutert.

[0330] Die **Fig.** 11 zeigt einen Frequenzkonverter zum Umwandeln von digitalen Daten in eine Frequenz und die **Fig.** 12 ein Ablaufdiagramm für die Arbeitsweise des Frequenzkonverters.

[0331] In der **Fig.** 11 bezeichnet das Bezugszeichen **50** einen Speicher, **51** ein Taktsignal C_N, **52** ein Datenregister, **53** das Ausgangssignal des Datenregisters **52**, **54** alle 1-Bit-Addierer, **55** den Additionsausgang aller Addierer **54**, **56** ein Taktsignal C_P, **57** ein Prozeßschieberegister, **58** ein Ausgangssignal des Scheiberegisters **57**, **59** ein erstes Flip-Flop zur zeitlichen Einstellung, **60** das Ausgangssignal des Flip-Flops **59**, **61** ein Träger-Ausgangssignal aller Addierer **54**, **62** ein zweites Flip-Flop zum Festhalten des Träger-Ausgangssignals **61**, **63** das Ausgangssignal des Flip-Flops **62**, **64** ein Zeitgebersignal, **65** eine UND-Schaltung zum Entfernen des vorgegebenen Ausgangssignals aus dem Träger-Ausgangssignal **61**, und **66** das Ausgangssignal des Frequenzkonverters.

[0332] Zur einfacheren Erläuterung der Arbeitsweise wird angenommen, daß die Datenlänge 4 Bit ist, daß Daten mit dem gleichen Wert periodisch aus dem Speicher **50** ausgelesen werden, und daß die Daten des Datenregisters **52** und des Schieberegisters **57** in der Richtung des Pfeiles oberhalb des jeweiligen Registers verschoben werden.

[0333] Das Taktsignal C_N **51** und das Taktsignal C_P **56** sind Taktsignale, die sich in der Phase um 90° unterscheiden, wie es bei (1) und (2) in der **Fig.** 12 gezeigt ist.

[0334] Im Anfangszustand sind die Anfangswerte im Datenregister **52** und im Schieberegister **57** gleich 02 (das heißt, daß der binäre Datenwert Null ist).

[0335] Es wird angenommen, daß der Datenwert **82** (der binäre Datenwert **8**) wie bei (3) in der **Fig.** 12 gezeigt mit der führenden Flanke des Taktsignals C_P **56** aus dem Speicher **50** ausgelesen wird. Da diese Daten mit der führenden Flanke des Taktsignals C_N **51** in das Datenregister **52** eingegeben werden, sind das höchstwertige Bit DR_3 bis zum niedrigstwertigen Bit DR_0 wie bei (4) in der **Fig.** 12 gezeigt angeordnet.

[0336] Wenn die bei (4) in der **Fig.** 12 gezeigten Daten viermal mit der führenden Flanke des Taktsignals C_N **51** verschoben werden, ist der Wert des niedrigstwertigen Bits DR_0 des Datenregisters **52** wie bei (7) in der **Fig.** 12 gezeigt und wird zum Eingangsanschluß A aller Addierer **54** als Ausgangssignal **53** addiert.

[0337] Dabei werden das Ausgangssignal **60** des ersten Flip-Flops **59** und das Ausgangssignal **63** des zweiten Flip-Flops **62** nicht zu den Eingangsanschlüssen B und C aller Addierer **54** addiert, so daß das Ausgangssignal **55** am Additionsausgangsanschluß $\sum$ aller Addierer **54** auf H geht, wie es bei (8) in der **Fig.** 12 gezeigt ist, und zum Schieberegister **57** hinzuaddiert wird. Das Ausgangssignal **61** am Träger-Ausgangsanschluß C_r aller Addierer **54** wird L, wie es bei (14) in der **Fig.** 12 gezeigt ist.

[0338] Da das Ausgangssignal **55** aller Addierer **54** mit der führenden Flanke des Taktsignals C_P **56**, das bei (2) in der **Fig.** 12 gezeigt ist, in das Schieberegister **57** eingegeben wird, ist das niedrigstwertige Bit SR_0 des Schieberegister **57** wie bei (**9**) in der **Fig.** 12 gezeigt.

[0339] Der Datenwert **82** (dezimal 8) wird mit der führenden Flanke des gleichen Taktsignals C_P **56** aus dem Speicher **50** ausgelesen und mit der führenden Flanke des Taktsignals C_N 51 in das Datenregister **52** geschoben.

[0340] Die Daten des niedrigstwertigen Bits SR_0 des Schieberegisters **57** werden mit der fühenden Flanke des Taktsignals C_P 56 im Schieberegister **57** verschoben, und das höchstwertige Bit SR_3 wird in der Mitte der 7. Verschiebung zu H und das Ausgangssignal **58** des Schieberegister **57** ebenfalls zu H.

[0341] Da das Ausgangssignal **58** mit der führenden Flanke des Taktsignals C_P **56** in den ersten Flip-Flop **59** eingegeben wird, wird das Ausgangssignal **60** davon mit der B. Verschiebung zu H, wie es bei (**13**) in der **Fig.** 12 gezeigt ist, und wird an den Eingangsanschluß B aller Addierer **54** ausgegeben.

[0342] Das geringstwertige Bit DR_0 des Datenregisters **52** wird bei der 8. Verschiebung ebenfalls zu H und wird in den Eingangsanschluß A aller Addierer **54** eingegeben.

[0343] Im Ergebnis wird das Ausgangssignal **55** am Additionsausgangsanschluß $\sum$ aller Addierer **54** zu L, auch wenn das Ausgangssignal **61** des Träger-Ausgangsanschlusses C_r zu H wird. Dieses Ausgangssignal **61** wird mit der führenden Flanke des nächsten Taktsignals C_N **51** in den zweiten Flip-Flop eingegeben, und das bei (**15**) in der **Fig.** 12 gezeigte Ausgangssignal **63** wird in den Eingangsanschluß C aller Addierer **54** gegeben.

[0344] Das Ausgangssignal **55** am Additionsausgangsanschluß $\sum$ aller Addierer **54** geht damit auf H und wird in das Schieberegister **57** geschoben.

[0345] Da sich die erläuterte Operation periodisch wiederholt, bewegen sich die in der **Fig.** 12 gezeigten Daten im Kreis durch das Datenregister **52** und das Schieberegister **57**. Durch das Hinzuaddieren des Träger-Ausgangssignals **61** aller Addierer **54**, das dabei erzeugt wird, zum UND-Gatter und Zuführen mit dem Zeitgebersignal **64** wird in der festen Periode wie bei (**16**) in der **Fig.** 12 gezeigt das Ausgangssignal **66** erhalten.

[0346] Diese Erläuterung betrifft den Vorgang, wenn eine Art von digitalen Daten in eine Frequenz umgewandelt wird. Um zwei Arten von Frequenzen zu erhalten, werden aus dem Speicher **50** abwechselnd verschiedene Daten ausgelesen und verarbeitet. Dabei sollte die Anzahl der Bits des Schieberegisters **57** auf 8 Bit erhöht werden, und es sollten zwei UND-Gatter und ein Zeitgebersignal so vorgesehen werden, daß aus dem Träger-Ausgangssignal **61** aller Addierer **54** die vorgegebene Frequenz herausgenommen wird. Durch dieses Ändern der Bitlänge des Schieberegisters **57** kann der Frequenzkonverter der **Fig.** 10 viele Frequenzen erzeugen.

[0347] Es wird nun die Frequenzvergleichsoperation der **Fig.** 1 und 8 erläutert. In der **Fig.** 13 ist ein Frequenzkomparator gezeigt und in der **Fig.** 14 ein Operations-Zeitdiagramm. In der **Fig.** 13 bezeichnen die Bezugszeichen **70** bis **73** Flip-Flops, **74** bis **79** UND-Gatter, P und M zu vergleichende Eingangsfrequenzsignale und **80** das Ausgangssignal des Ergebnisses des Frequenzvergleichs.

[0348] Zum Beispiel invertiert und betätigt bei der Eingabe des P-Signals dieses das eine derjenigen Flip-Flops **70** bis **73**, das dem Eingang am nächsten liegt, auf Q = H, und wenn das M-Signal eingegeben wird, invertiert und betätigt dieses das eine derjenigen Flip-Flops **70** bis **73**, bei dem Q = H ist, und das dem Eingang am nächsten auf Q = L.

[0349] Wenn das Eingangssignal P wie in der **Fig.** 14 gezeigt kontinuierlich eingegeben wird, arbeitet das Flip-Flop **70** synchron mit der hinteren Flanke des ersten Impulses P_1 des Signals P, und Q_0 wird zu H, und wenn der zweite Impuls P_2 eingegeben wird, arbeitet das Flip-Flop **71** synchron mit der hinteren Flanke des Ausgangs des UND-Gatters **74** und Q_1 geht auf H, und wenn der dritte Impuls P_3 eingegeben wird, arbeitet das Flip-Flop **72** synchron mit der hinteren Flanke des Ausgangssignals des UND-Gatters **76** und Q_2 geht auf H, und wenn der vierte Impuls P_4 des Signals P eingegeben wird, arbeitet das Flip-Flop **73** synchron mit der hinteren Flanke des Ausgangssignals des UND-Gatters **78** und Q_3 wird zu H.

[0350] Wenn das Eingangssignal M kontinuierlich eingegeben wird, arbeitet das Flip-Flop **70** synchron mit der hinteren Flanke des ersten Impulses M_1 des Signals M, und Q_0 ändert sich von H auf L, und wenn der zweite Impuls M_2 eingegeben wird, arbeitet das Flip-Flop **71** synchron mit der hinteren Flanke des UND-Gatters **75** und Q_1 ändert sich von H auf L, und wenn der dritte Impuls M_3 eingegeben wird, arbeitet das Flip-Flop **72** synchron mit der hinteren Flanke des UND-Gatters **77** und Q_2 ändert sich von H auf L, und wenn der vierte Impuls M_4 eingegeben wird, arbeitet das Flip-Flop **73** synchron mit der hinteren Flanke des UND-Gatters **79** und Q_3 ändert sich von H auf L.

[0351] Wenn die Anzahl der Impulse groß ist, kann so der Ausgangszustand des Flip-Flops der folgenden Stufe invertiert werden, und wenn P > M ist, wird Q_3 des Flip-Flops FF_3 zu H, das heißt, das Ausgangssignal **80** des Frequenzvergleichs wird zu H. Wenn P < M ist, wird das Ausgangssignal **80** zu L.

[0352] Für einen Multifrequenzvergleich wird der Aufbau der **Fig.** 15 verwendet. In der **Fig.** 15 bezeichnet das Bezugszeichen **90** ein Schieberegister mit einer Anzahl von Bits, die einer Länge von 4 Bits gleichwertig ist, was zu den Flip-Flops **70** bis **73** in der **Fig.** 13 gleichwertig ist, und **91** bezeichnet einen Addierer mit einer Funktion, die es ermöglicht, daß die Daten auf dem Schieberegister **90** die gleiche Operation ausführen wie bei der Inversion der Flip-Flops **70** bis **73** der **Fig.** 13.

[0353] Das Symbol I_n bezeichnet einen Zeitschlitz, in den 4 Bit gruppiert sind, und der Zeitschlitz gibt das Signal P an den +A–Anschluß des Addierers **91** und das Signal M an den -C-Anschluß. Es wird angenommen, daß P > M ist.

[0354] In der Frequenzvergleichsoperation wird das Vorhandensein oder die Abwesenheit eines Impulses festgestellt, so daß in diesem Zustand P = H und M = L ist. Da das Vorzeicheninversionssignal H von M in den -C-Anschluß eingegeben und addiert wird, wird das Träger-Ausgangssignal C_r zu H, und dieser Wert wird zu dem I_0-Zeitschlitz des Schieberegisters **90** addiert.

[0355] Die Daten dieses I_0-Zeitschlitzes bewegen sich durch das Schieberegister, werden am I_0-Zeitschlitz in der nächsten Periode in den +B-Anschluß des Addierers **91** eingegeben und mit dem P-Eingang und dem M-Eingang zusammenaddiert, so daß das Träger-Ausgangssignal C_r des Addierers auch in diesem Fall zu H wird und sich Daten mit dem gleichen Wert wie die vorherigen durch das Schieberegister **90** bewegen.

[0356] Wenn P < M ist, wird ein Signal L addiert, bei dem die Vorzeichen von P = L und M = H invertiert sind, so daß das Träger-Ausgangssignal C_r zu L wird, das heißt subtrahiert wird und der Wert zum I_0-Zeitschlitz des Schieberegisters **90** hinzuaddiert wird. Das heißt, daß sich durch das Schieberegister **90** ein L bewegt.

[0357] Wenn P = M = H ist, wird ein Signal L addiert, bei dem die Vorzeichen von P = L und M = H invertiert sind. Der Wert des Träger-Ausgangssignals C_r variiert mit den Daten des I_0-Zeitschlitzes, die sich durch das Schieberegister **90** bewegen. Wenn die Daten des I_0-Zeitschlitzes auf L sind, geht das Träger-Ausgangssignal C_r auch auf L, und wenn die Daten des I_0-Zeitschlitzes auf H sind, wird das Träger-Ausgangssignal C_r ein H. Das heißt, daß sich die Daten des I_0-Zeitschlitzes, die sich durch das Schieberegister **90** bewegen, nicht ändern.

[0358] Wenn P = M = L ist, wird ein Signal H addiert, bei dem die Vorzeichen von P = L und M = H invertiert sind. Auch in diesem Fall variiert der Wert des Träger-Ausgangssignals C_r mit den Daten des I_0-Zeitschlitzes, die sich durch das Schieberegister **90** bewegen. Wenn die Daten des I_0-Zeitschlitzes auf L sind, geht das Träger-Ausgangssignal C_r auch auf L, und wenn die Daten des I_0-Zeitschlitzes auf H sind,

wird das Träger-Ausgangssignal C_r ein H. Das heißt, daß sich die Daten des I_0-Zeitschlitzes, die sich durch das Schieberegister **90** bewegen, nicht ändern.

[0359] Auf diese Weise wird die Frequenzvergleichsoperation ausgeführt.

[0360] Die **Fig.** 16 zeigt eine andere Ausführungsform der vorliegenden Erfindung. Bei der Ausführungsform der **Fig.** 8 weist das System A zwei Schaltungen mit den Speichern **30** und **31** in der Logikschaltung **15** auf, und das Ausgangssignal **35** wird durch Auswählen der Daten, die von den Speichern ausgegeben werden, mittels des Umschalters **34** erhalten. Auch das System B weist zwei Schaltungen mit den Speichern **40** und **41** in der Logikschaltung **16** auf, und das Ausgangssignal **45** wird durch Auswählen der Daten, die von den Speichern ausgegeben werden, mittels des Umschalters **44** erhalten.

[0361] Der Unterschied der Ausführungsform der vorliegenden Erfindung der **Fig.** 16 von der der **Fig.** 8 ist der, daß jeder der Speicher in den Logikschaltungen **15** und **16** der Systeme A und B aus einer Schaltung besteht, und daß die CRC-Daten, die zu der letzten Adresse jedes Speichers gehören, in einem anderen Bereich gespeichert werden und durch das erwähnte Fehlererfassungssignal umgeschaltet werden.

[0362] Das heißt, daß die CRC-Daten D_{CRX1}, die vom Mikrocomputer **10** mit dem gleichen Verfahren wie erwähnt berechnet werden, im Speicher **32** des Systems A gespeichert werden, D_{CRCX2} wird im Speicher **33** gespeichert, und Dc_{RCY1}, das vom Mikrocomputer **11** berechnet wird, wird im Speicher **42** des Systems B gespeichert und D_{CRCY2} im Speicher **43**.

[0363] Im System A werden die aus dem Speicher **30** ausgelesenen Daten durch die Logikschaltung **200** zu den CRC-Daten **32** hinzuaddiert, und die aus dem Speicher **30** ausgelesenen Daten werden durch die Logikschaltung **201** zu den CRC-Daten **33** hinzuaddiert und in den Umschalter **34** eingegeben.

[0364] Im System B werden die aus dem Speicher **40** ausgelesenen Daten durch die Logikschaltung **202** zu den CRC-Daten **42** hinzuaddiert, und die aus dem Speicher **40** ausgelesenen Daten werden durch die Logikschaltung **203** zu den CRC-Daten **43** hinzuaddiert und in den Umschalter **44** eingegeben. Die Schaltzeit für die Umschalter **34** und **44** ist die gleiche wie erwähnt.

[0365] Bei einer anderen Ausführungsform der vorliegenden Erfindung, die in der **Fig.** 16 gezeigt ist, kann die Anzahl an Speichern, die die maximale Anzahl von Gattern benötigen, auf die Hälfte reduziert werden, was nicht nur erheblich zu einer Verbesserung der Zuverlässigkeit beiträgt, sondern auch große Auswirkungen auf den Energieverbrauch der LSI hat.

[0366] Die **Fig.** 17 zeigt eine weitere Ausführungsform der vorliegenden Erfindung. Die **Fig.** 17 zeigt auch ein Beispiel, bei dem jeder eingebaute Speicher eine Schaltung umfaßt. Der Unterschied zwischen der **Fig.** 8 und der **Fig.** 16 ist der, daß in der **Fig.** 8 das Ausgangssignal der Speicher vom Umschalter umgeschaltet wird und in der **Fig.** 16 die CRC-Daten umgeschaltet werden. Die CRC-Daten D_{CRCX1} die vom Mikrocomputer **10** mit dem gleichen Verfahren wie erwähnt berechnet werden, werden im Speicher **32** des Systems A gespeichert, und D_{CRCX2} wird im Anschluß an den Speicher **32** im Speicher **33** gespeichert. D_{CRCY1} das vom Mikrocomputer **11** berechnet wird, wird im Speicher **42** des Systems B gespeichert und D_{CRCY2} im Anschluß an den Speicher **42** im Speicher **43**.

[0367] Die CRC-Daten werden von den Umschaltern **204** und **205** unter Verwendung des Fehlererfassungssignals **154** und des Fehlererfassungssignals mit invertiertem Vorzeichen umgeschaltet. Auch bei der Ausführungsform der **Fig.** 17 kann die Anzahl an Speichern, die die maximale Anzahl von Gattern benötigen, auf die Hälfte reduziert werden, was nicht nur erheblich zu einer Verbesserung der Zuverlässigkeit beiträgt, sondern auch große Auswirkungen auf den Energieverbrauch der LSI hat.

[0368] Die **Fig.** 18 zeigt eine weitere Ausführungsform der vorliegenden Erfindung. Die **Fig.** 18 zeigt auch ein Beispiel, bei dem jeder eingebaute Speicher eine Schaltung umfaßt. Der Unterschied zur **Fig.** 17 ist der, daß die CRC-Daten von den Umschaltern umgeschaltet werden und dann das Ausgangssignal jedes Speichers und das Ausgangssignal jedes Umschalters durch den Addierer addiert werden.

[0369] Der Umschalter **34** des Systems A schaltet die CRC-Daten **32** und die CRC-Daten **33** unter Verwendung des Fehlererfassungssignals **165** mit invertiertem Vorzeichen um, und der Umschalter **44** des Systems B schaltet die CRC-Daten **42** und die CRC-Daten **43** unter Verwendung des Fehlererfassungssignals **154** um. Auch bei der Ausführungsform der vorliegenden Erfindung der **Fig.** 18 kann die Anzahl an Speichern, die die maximale Anzahl von Gattern benötigen, auf die Hälfte reduziert werden, was nicht nur erheblich zu einer Verbesserung der Zuverlässigkeit beiträgt, sondern auch große Auswirkungen auf den Energieverbrauch der LSI hat.

[0370] Die **Fig.** 19 ist eine schematische Layoutdarstellung einer weiteren Ausführungsform der vorliegenden Erfindung, die in der **Fig.** 8 gezeigt ist. Das Bezugszeichen **300** bezeichnet ein Businterface zum Übertragen und Aufnehmen von Daten zu und vom Mikrocomputer **10**, **301** und **302** Speicher zum Speichern von Daten, **303** einen Prozessor wie einen Speicherumschalter, einen Frequenzkonverter und einen Frequenzkomparator, **304** bezeichnet ein Businterface zum Übertragen und Aufnehmen von Daten zu und vom Mikrocomputer **11**, **305** und **306** Speicher zum Speichern von Daten, **307** einen Prozessor wie einen Speicherumschalter, einen Frequenzkonverter und einen Frequenzkomparator, und 308 einen Vergleicher für das Ausgangssignal des Prozessors **303** des Systems A und das Ausgangssignal des Prozessors **307** des Systems B.

[0371] Wenn die duplizierten Systeme A und B von-

einander entfernt angeordnet werden, können Auswirkungen beim Ausfall eines Systems auf das andere System verhindert werden, es kann zum Beispiel verhindert werden, daß, obwohl ein System ausgefallen ist, ein Signal ausgegeben wird, als ob es normal arbeiten würde.

[0372] Wenn die Schaltungen, die die Vergleicheinheit bilden, voneinander entfernt angeordnet werden und der Leitungsabstand vergrößert wird, können Auswirkungen des Ausfalls eines der Vergleicher auf die anderen Vergleicher verhindern werden, und es kann verhindert werden, daß ein Befehlssignal nicht ausgegeben wird oder ausgegeben wird.

[0373] In der **Fig.** 20 ist eine andere Ausführungsform der vorliegenden Erfindung dargestellt, die im folgenden erläutert wird. Die **Fig.** 20 ist eine Darstellung einer Ausführungsform, bei der eine Anzahl von Prozessor der Art, wie sie in der **Fig.** 2 gezeigt sind, vorgesehen ist.

[0374] Das Bezugszeichen **2000** bezeichnet ein Steuergerät, **2010** einen ersten Prozessor zum Eingeben und Verarbeiten von ersten Eingangsdaten und zum Ausgeben von ersten Ausgangsdaten **2100** und eines ersten Erfassungssignals **2140**, **2020** einen zweiten Prozessor zur Ausgabe von zweiten Ausgangsdaten **2110** und eines zweiten Erfassungssignals **2150**, **2030** einen dritten Prozessor zur Ausgabe dritter Ausgangsdaten **2120** und eines dritten Erfassungssignals **2160**, **2040** einen vierten Prozessor zur Ausgabe vierter Ausgangsdaten **2130** und eines vierten Erfassungssignals **2170**, **2180** einen ersten Transmitter zum Übertragen des ersten Erfassungssignals **2140**, das vom ersten Prozessor **2010** ausgegeben wird, zum zweiten Prozessor **2020**, **2190** einen zweiten Transmitter zum Übertragen des zweiten Erfassungssignals **2150**, das vom zweiten Prozessor **2020** ausgegeben wird, zum dritten Prozessor **2030**, und **2200** einen dritten Transmitter zum Übertragen des dritten Erfassungssignals **2160**, das vom dritten Prozessor **2030** ausgegeben wird, zum den vierten Prozessor **2040**.

[0375] Durch die Verwendung eines solchen duplizierten Aufbaus werden durch die Fehlererfassungssignale die CRC-Daten des entsprechenden Detektors nur dann umgeschaltet, wenn alle Daten, Schaltungen und Elemente normal arbeiten, es wird ein Ausgangssignal zum Steuern des zu steuernden Objekts ausgegeben, und wenn in einem Teil ein Fehler entdeckt wird, wird kein Ausgangssignal ausgegeben. Beim Auftreten eines Fehlers wird somit eine fehlersicher Funktion zum Steuern auf der sicheren Seite möglich.

[0376] Gemäß der vorstehend beschriebenen Erfindung wird ein Steuergerät mit einer extrem hohen Ausfallsicherheit und ein System mit diesem Steuergerät geschaffen.

Patentansprüche

1. Steuergerät mit mehreren Prozessoren zum Eingeben von Eingangssignalen und zum Ausgeben von Ausgangssignalen, die durch das Ausführen mehrerer Operationen erhalten werden, wobei das Steuergerät ein Ausgangssignal eines ersten Prozessors unter den mehreren Prozessoren in zweite Prozessoren eingibt, die mindestens einen zweiten Prozessor unter den mehreren Prozessoren beinhalten, und Ausgangssignale der zweiten Prozessoren in den ersten Prozessor eingibt, **dadurch gekennzeichnet**, daß die Polarität eines der Ausgangssignale der zweiten Prozessoren zu der der Eingangssignale des ersten Prozessors invertiert ist.

2. Steuergerät nach Anspruch 1, mit weiterhin einem ersten seriellen Konverter zum Konvertieren eines parallelen Eingangssignals in ein serielles Signal, mit dem ersten Prozessor zum Verarbeiten eines Ausgangssignals des ersten seriellen Konverters als Eingangssignal, einem ersten Parallel-Konverter zum Umwandeln eines seriellen Ausgangssignals des ersten Prozessors in ein Parallel-Signal, mit einem zweiten seriellen Konverter zum Konvertieren eines parallelen Eingangssignals in ein serielles Signal, mit dem zweiten Prozessor zum Verarbeiten eines Ausgangssignals des zweiten seriellen Konverters als Eingangssignal, und mit einem zweiten Parallel-Konverter zum Konvertieren eines seriellen Ausgangssignals des zweiten Prozessors in ein Parallel-Signal, wobei das Steuergerät eines der Ausgangssignale des ersten Parallel-Konverters in den zweiten seriellen Konverter zusammen mit anderen parallelen Eingangssignalen eingibt, um es in serielle Signale umzuwandeln, und wobei eines der in den ersten seriellen Konverter eingegebenen parallelen Eingangssignale in einem Polaritätsumwandler die Polarität eines der Ausgangssignale des zweiten Parallel-Konverters in die der anderen parallele Eingangssignale umwandelt, die in den ersten seriellen Konverter eingegeben werden.

3. Steuergerät nach Anspruch 1, mit weiterhin einer Addiervorrichtung, um ein Ausgangssignal eines der mehreren Prozessoren und einen optionalen Wert zu addieren, wobei das Steuergerät entsprechend Eingangssignale in die mehreren Prozessoren, eines der Ausgangssignale aus dem zweiten Prozessor in die Addiervorrichtung und ein Ausgangssignal der Addiervorrichtung in den ersten Prozessor eingibt.

4. Steuergerät nach Anspruch 1, mit weiterhin einer Addiervorrichtung, um ein Ausgangssignal des ersten Prozessors und einen optionalen Wert zu addieren, und mit einer Subtrahiervorrichtung, um einen optionalen Wert von einem Ausgangssignal des zweiten Prozessors zu subtrahieren, wobei das Steuergerät Eingangssignale in den ersten bzw. zweiten Prozessor, eines der Ausgangssignale des ersten Prozessors in die Addiervorrichtung, eines der Ausgangssignale der Addiervorrichtung in den zweiten

Prozessor, eines der Ausgangssignale des zweiten Prozessors in die Subtrahiervorrichtung und ein Ausgangssignal der Subtrahiervorrichtung in den ersten Prozessor eingibt.

5. Steuergerät nach Anspruch 2, mit weiterhin einem ersten Zustands-Diskriminations-Signal, um die Zustände des ersten seriellen Konverters, des ersten Prozessors und des ersten Parallel-Konverters zu unterscheiden, mit einem zweiten Zustands-Diskriminations-Signal, um die Zustände des zweiten seriellen Konverters, des zweiten Prozessors und des zweiten Parallel-Konverters zu unterscheiden, mit einem ersten Fehlererfassungssignal zum Anzeigen der Zustände des ersten seriellen Konverters, des ersten Prozessors und des ersten Parallel-Konverters durch das erste Zustands-Diskriminations-Signal, mit einem zweitem Fehlererfassungssignal zum Anzeigen der Zustände des zweiten seriellen Konverters, des zweiten Prozessors und des zweiten Parallel-Konverters durch das zweite Zustands-Diskriminations-Signal, mit einem Polaritätsinversionsfehlererfassungssignal, um das zweite Fehlererfassungssignal in einen Polaritätsinverter einzugeben und um seine Ausgangspolarität in dem Polaritätsinverter zu invertieren, mit einer ersten Addiervorrichtung, um das erste Zustands-Diskriminations-Signal und das Polaritätsinversionsfehlererfssungssignal zu addieren, und mit einer zweiten Addiervorrichtung, um das zweite Zustands-Diskriminations-Signal und das erste Fehlererfassungssignal zu addieren, wobei das Steuergerät ein Ausgabesignal der ersten Addiervorrichtung zusammen mit parallelen Eingangssignalen in den ersten seriellen Konverter und ein Ausgangssignal der zweiten Addiervorrichtung zusammen mit parallelen Eingangssignalen in den zweiten seriellen Konverter eingibt.

6. Steuergerät nach Anspruch 1, wobei der erste Prozessor ein erstes Ausgangssignal, das durch die Eingabe und die Verarbeitung eines ersten Eingangs- und eines ersten Erfassungssignals erhalten wird, ausgibt, der zweite Prozessor ein zweites Ausgangssignal, das durch die Eingabe und die Verarbeitung eines zweiten Eingangs- und eines zweiten Erfassungssignals erhalten wird, ausgibt, ein erster Transmitter zum Übertragen des ersten Erfassungssignals an den zweiten Prozessor, und ein zweiter Transmitter zum Übertragen des zweiten Erfassungssignals des zweiten Prozessors an den ersten Prozessor vorhanden sind, und der zweite Prozessor das zweite Erfassungssignal auf der Basis des über den ersten Transmitter eingegebenen ersten Erfassungssignals ausgibt, und der zweite Transmitter das zweite Erfassungssignal, dessen Polarität zu der des ersten Eingangssignals des ersten Prozessors invertiert ist, ausgibt, und der erste Prozessor das erste Erfassungssignal auf der Basis des polaritätsinvertierten Signals des zweiten Transmitters ausgibt.

7. Steuergerät nach Anspruch 5, wobei der zweite Transmitter eine logische Einheit aufweist, um die Polarität des zweiten Erfassungssignals, das von dem zweiten Prozessor ausgegeben wird, in die des ersten Eingangssignals zu invertieren.

8. Steuergerät nach Anspruch 6, wobei der zweite Prozessor eine logische Einheit aufweist, um die Polarität des zweiten Erfassungssignals, das auf der Basis des ersten Erfassungssignals verarbeitet wird; in die des ersten Eingangssignals zu invertieren.

9. Steuergerät nach einem der Ansprüche 6 bis 8, wobei das erste oder das zweite Eingangssignal Verarbeitungsdaten des ersten und des zweiten Prozessors und Überprüfungsdaten zum Überprüfen der Operationen des ersten und zweiten Prozessors umfaßt.

10. Steuergerät nach Anspruch 9, wobei die Verarbeitungsdaten des ersten Eingangssignals die gleichen sind wie die des zweiten Eingangssignals.

11. Steuergerät nach Anspruch 9, wobei die Überprüfungsdaten den Inhalt der ersten oder der zweiten Verarbeitungsdaten überprüfen.

12. Steuergerät nach Anspruch 9, wobei der erste oder der zweite Prozessor eine logische Einheit aufweist, um eine Logik für das erste oder das zweite Erfassungssignal und die Überprüfungsdaten des ersten oder des zweiten Eingangssignals zu implementieren.

Es folgen 24 Blatt Zeichnungen

Anhängende Zeichnungen

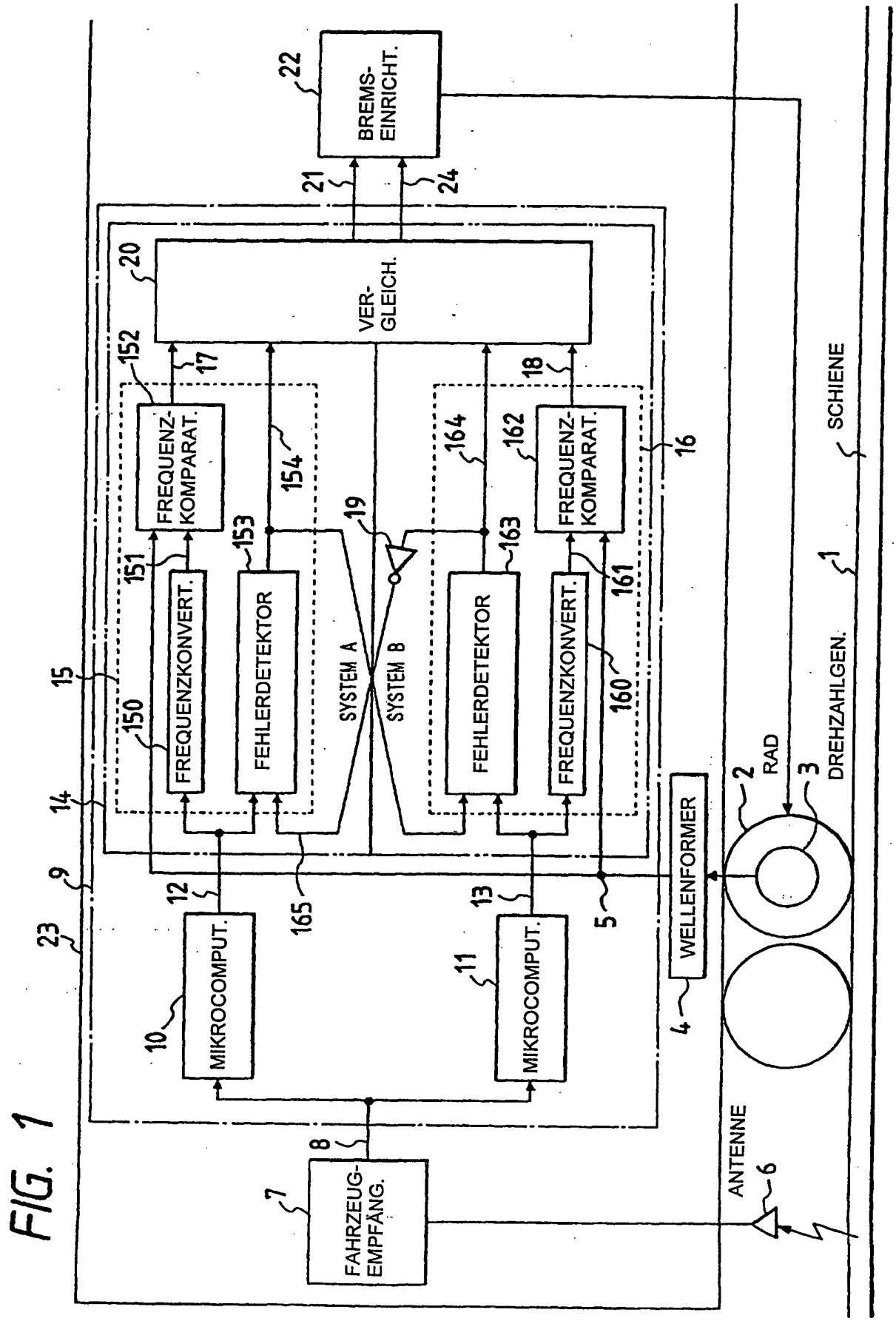


FIG. 1

FIG. 2

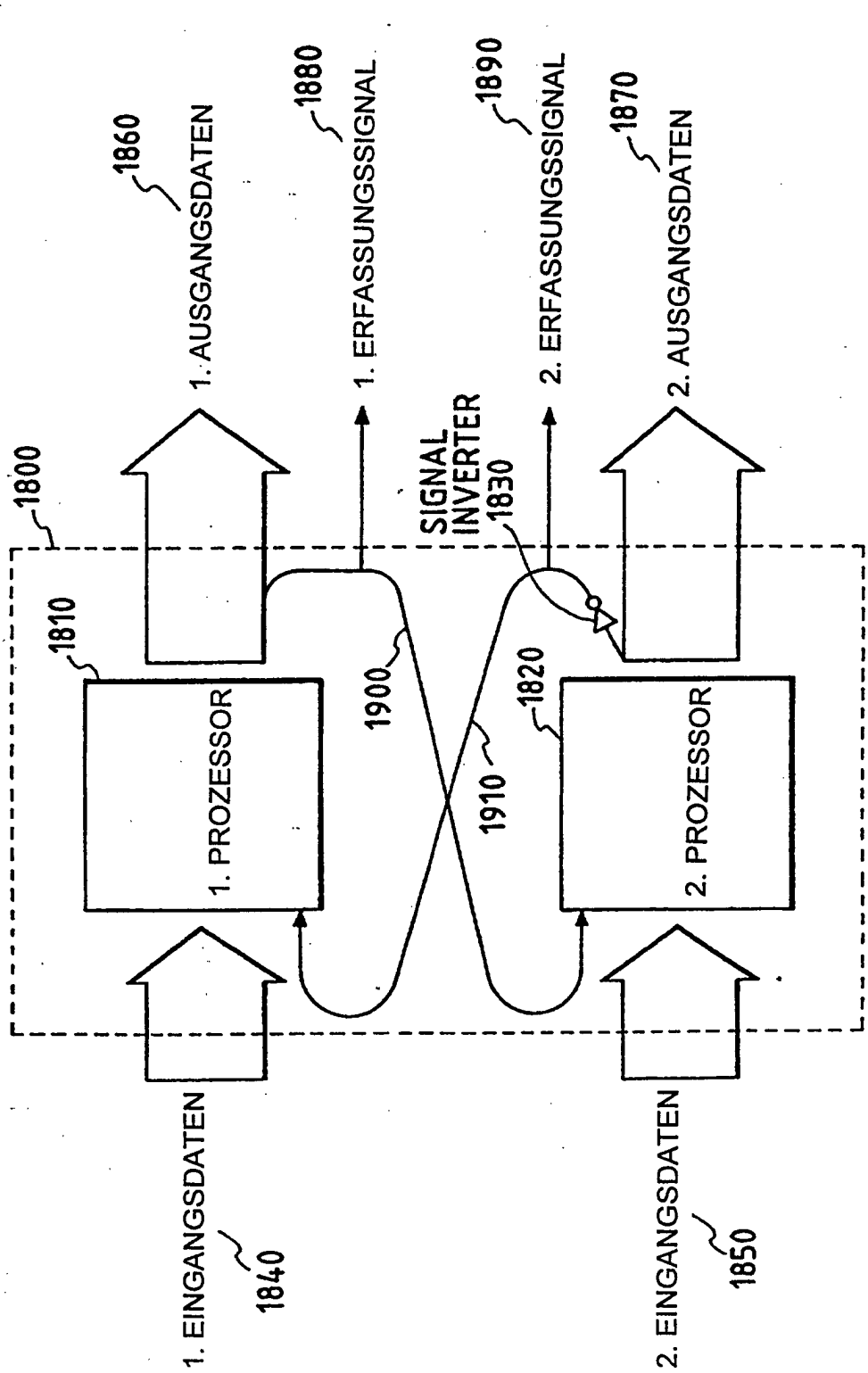

1800
1810
1. PROZESSOR
1820
2. PROZESSOR
1830
SIGNAL INVERTER
1840
1. EINGANGSDATEN
1850
2. EINGANGSDATEN
1860
1. AUSGANGSDATEN
1870
2. AUSGANGSDATEN
1880
1. ERFASSUNGSSIGNAL
1890
2. ERFASSUNGSSIGNAL
1900
1910

FIG. 3

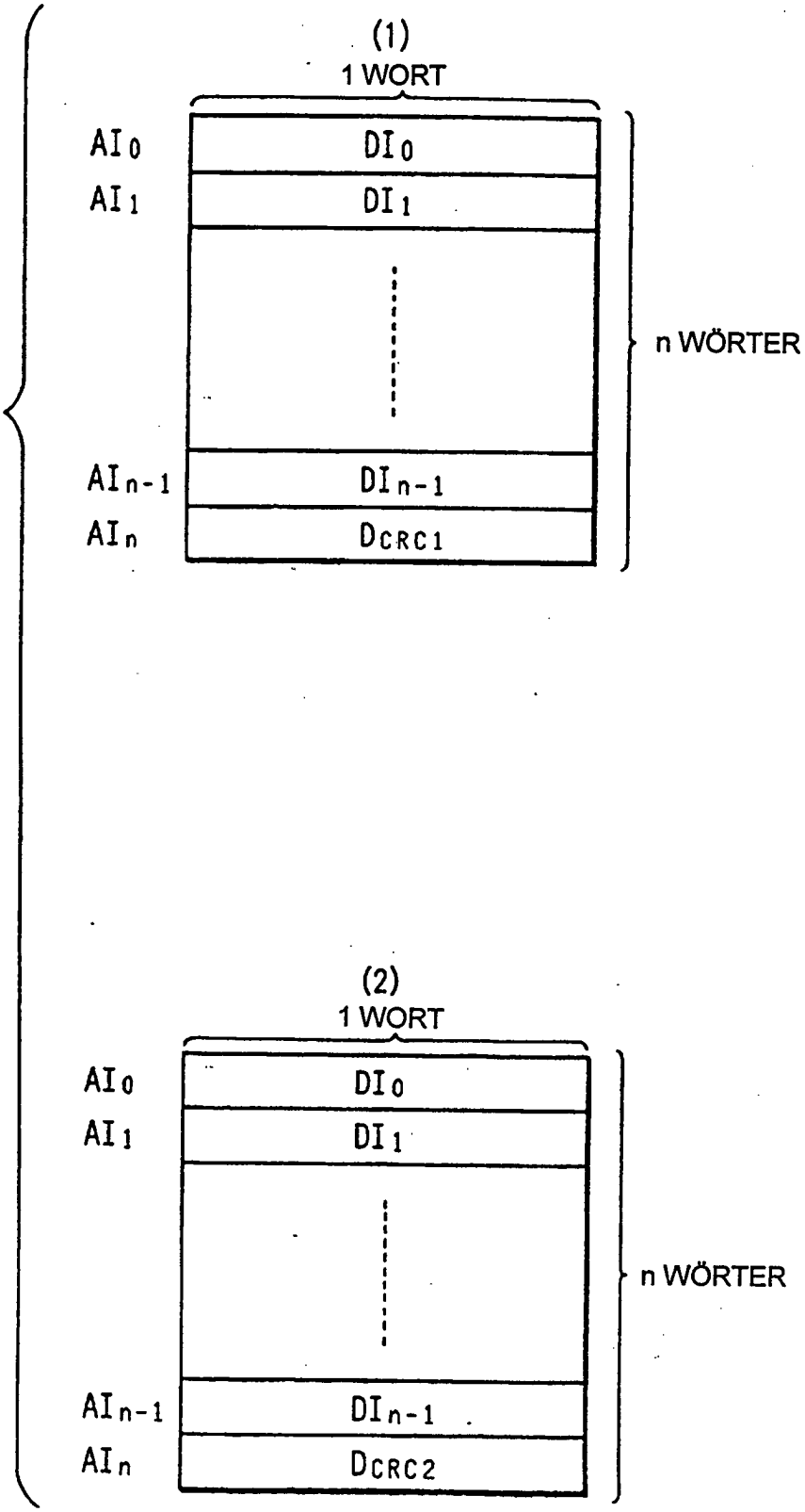

(1)
1 WORT
AI0
DI0
AI1
DI1
n WÖRTER
AIn-1
DIn-1
AIn
DCRC1
(2)
1 WORT
AI0
DI0
AI1
DI1
n WÖRTER
AIn-1
DIn-1
AIn
DCRC2

FIG. 4

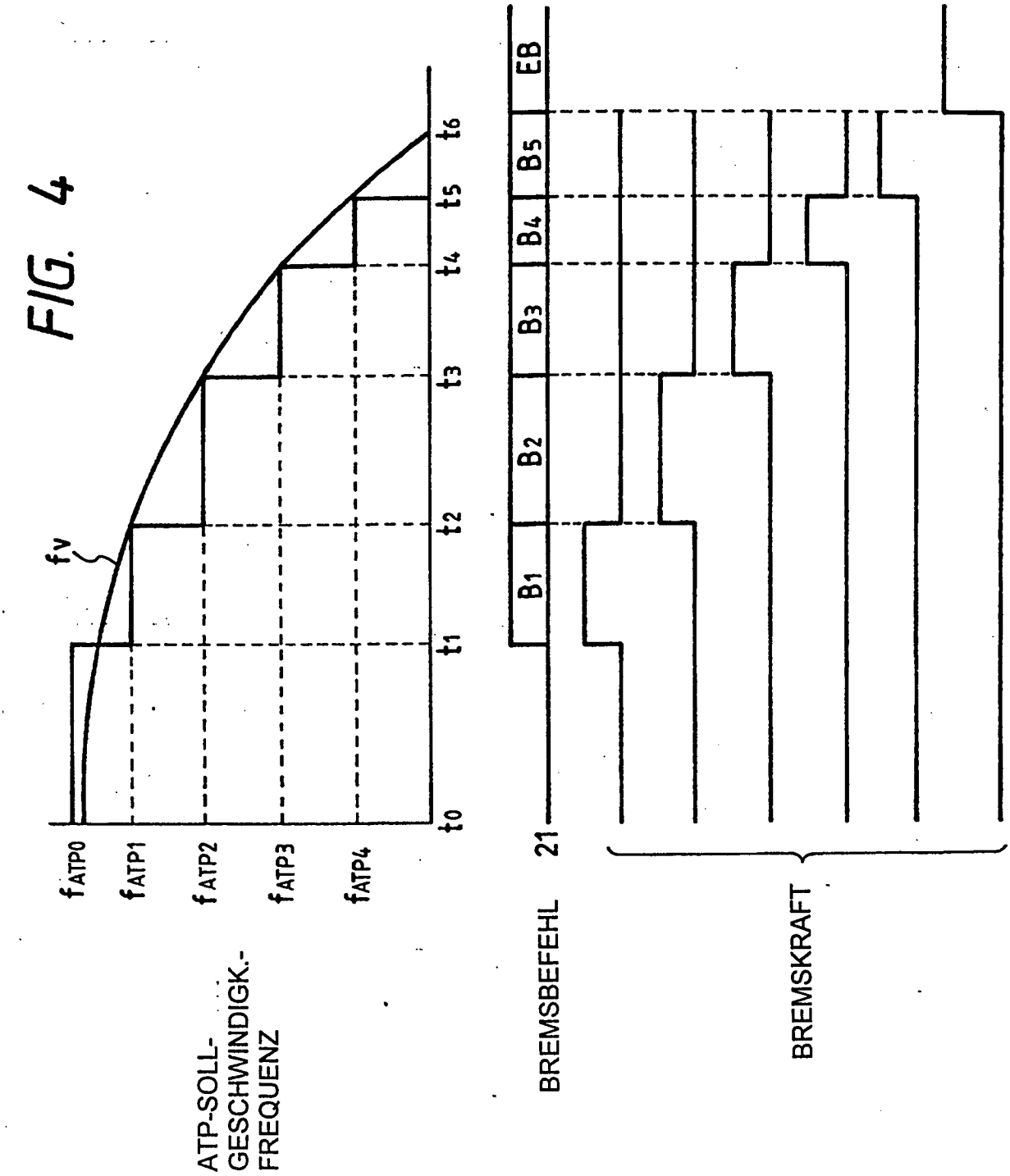

fv
t0
t1
t2
t3
t4
t5
t6
fATP0
fATP1
fATP2
fATP3
fATP4
ATP-SOLL-GESCHWINDIGK.-FREQUENZ
B1
B2
B3
B4
B5
EB
21
BREMSBEFEHL
BREMSKRAFT

FIG. 5

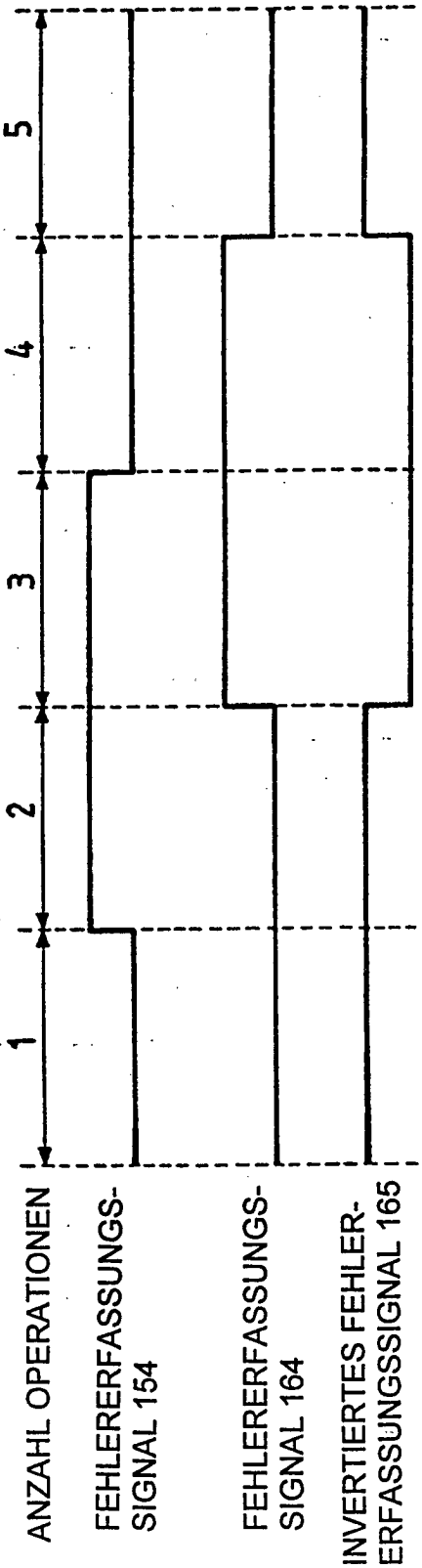

1
2
3
4
5
ANZAHL OPERATIONEN
FEHLERERFASSUNGS-SIGNAL 154
FEHLERERFASSUNGS-SIGNAL 164
INVERTIERTES FEHLER-ERFASSUNGSSIGNAL 165

FIG. 6

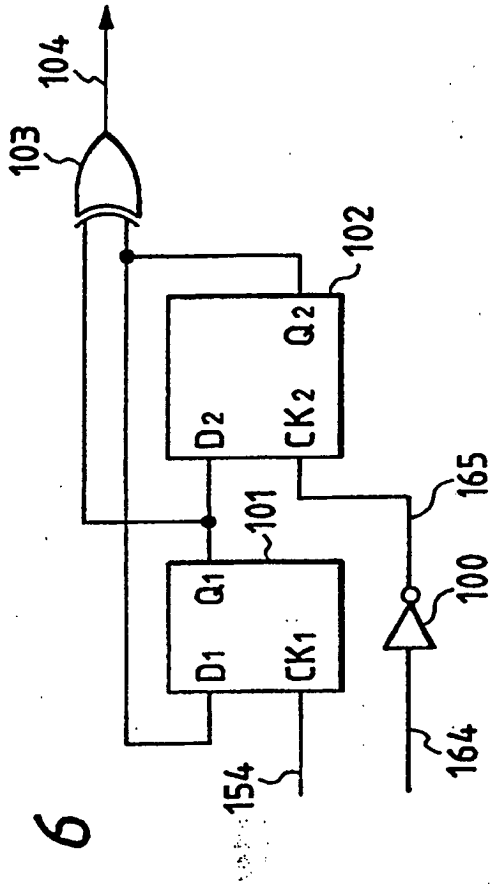

104
103
102
Q2
D2
CK2
165
101
100
Q1
D1
CK1
154
164

FIG. 7

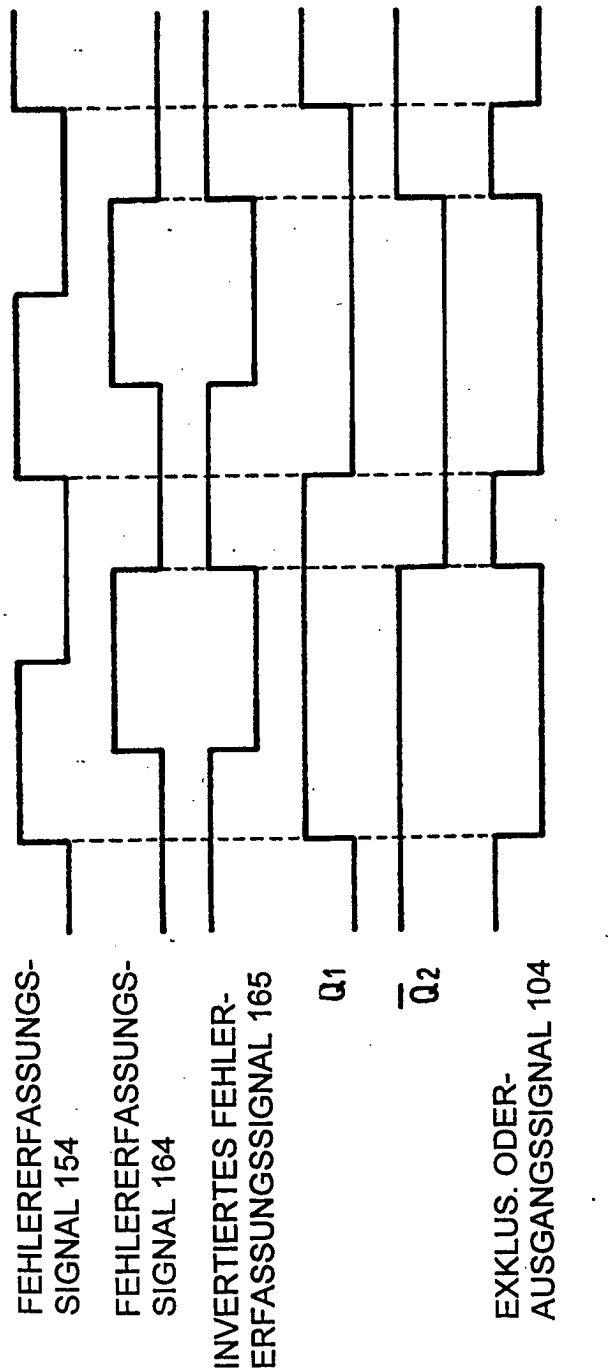

FEHLERERFASSUNGS-SIGNAL 154
FEHLERERFASSUNGS-SIGNAL 164
INVERTIERTES FEHLER-ERFASSUNGSSIGNAL 165
Q1
$\overline{Q2}$
EXKLUS. ODER-AUSGANGSSIGNAL 104

FIG. 8

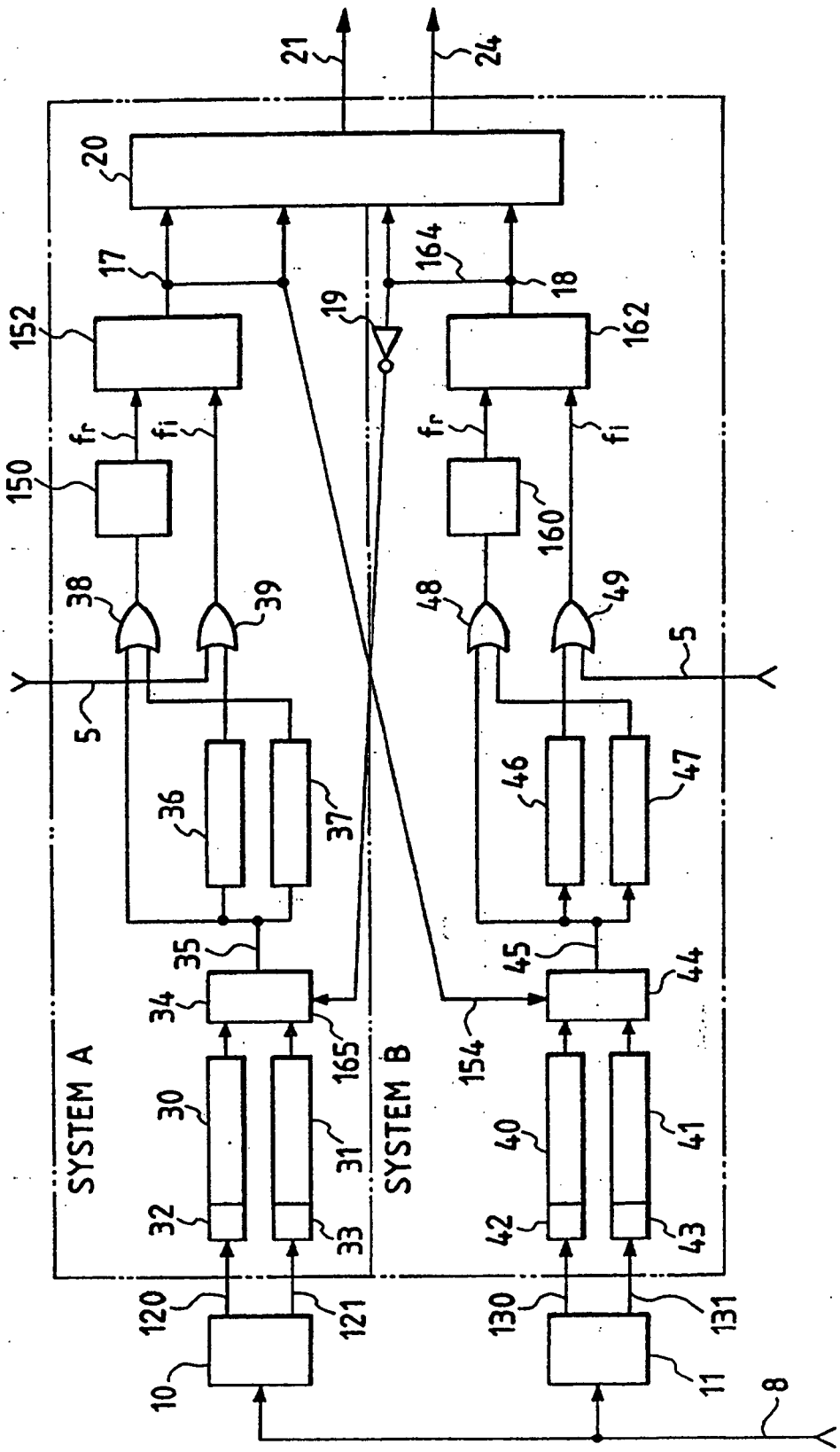

21
24
20
17
164
18
19
152
162
fr
fi
150
160
38
39
48
49
5
Y
36
37
46
47
35
45
34
44
165
154
SYSTEM A
SYSTEM B
30
31
40
41
32
33
42
43
120
121
130
131
10
11
8

FIG. 9

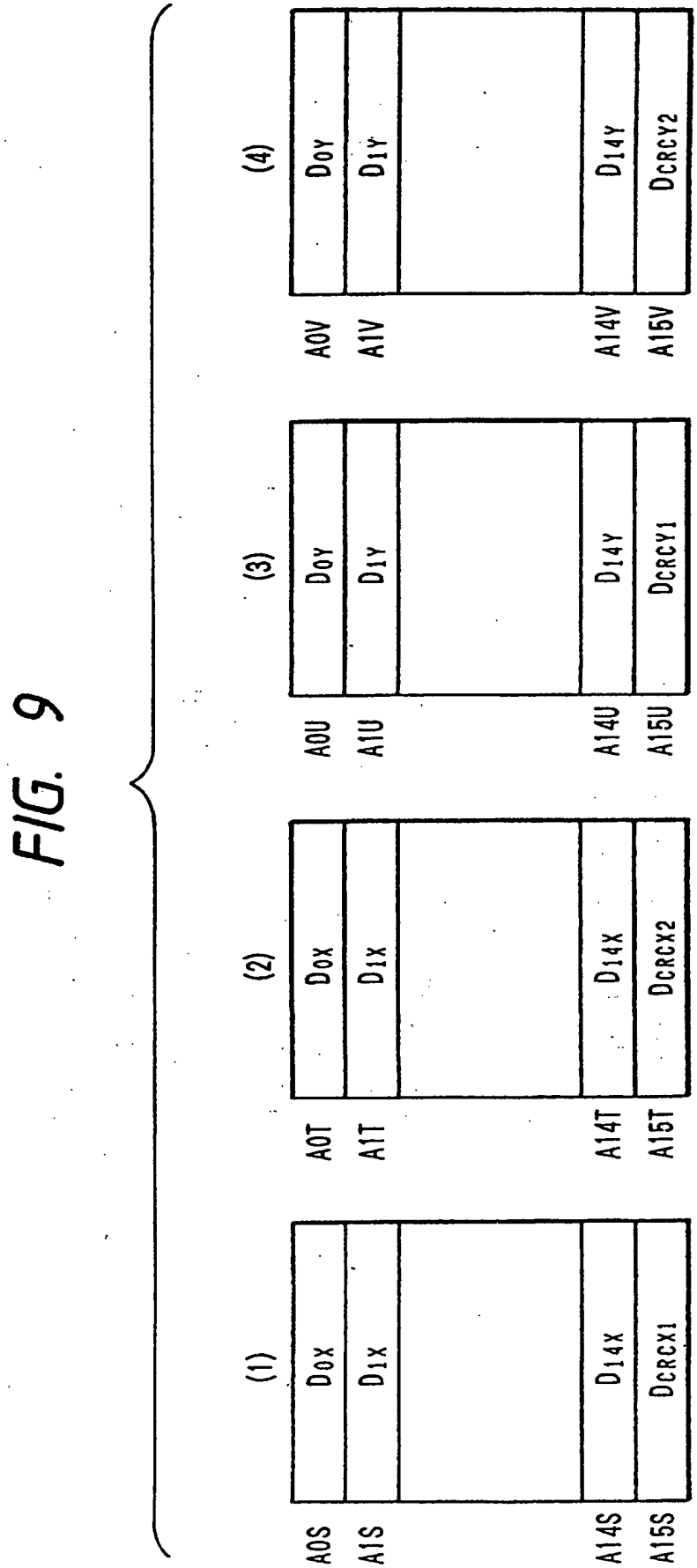

(1)
D0X
D1X
D14X
DCRCX1
A0S
A1S
A14S
A15S
(2)
D0X
D1X
D14X
DCRCX2
A0T
A1T
A14T
A15T
(3)
D0Y
D1Y
D14Y
DCRCY1
A0U
A1U
A14U
A15U
(4)
D0Y
D1Y
D14Y
DCRCY2
A0V
A1V
A14V
A15V

FIG. 10

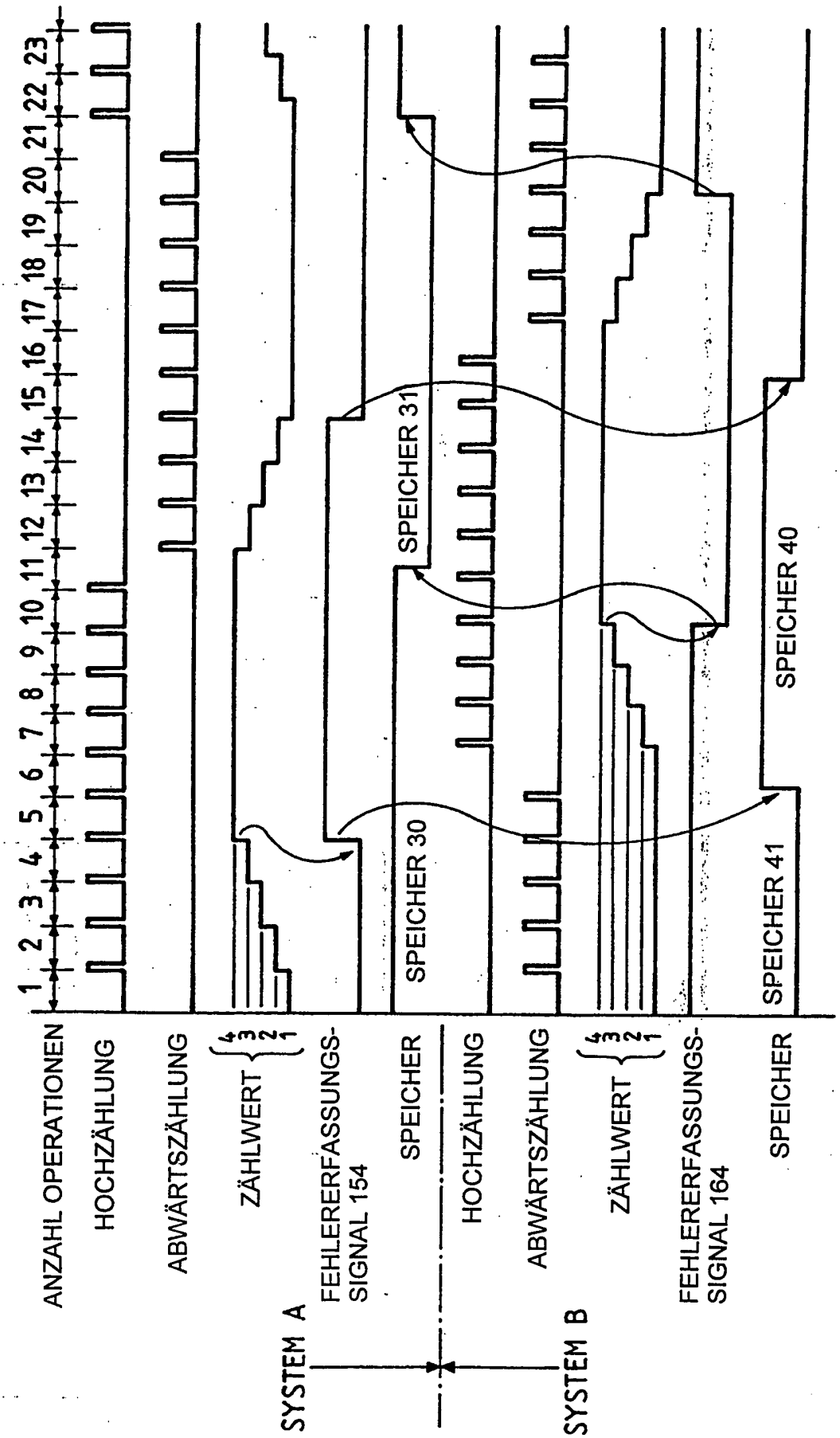

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
ANZAHL OPERATIONEN
HOCHZÄHLUNG
ABWÄRTSZÄHLUNG
ZÄHLWERT
4
3
2
1
FEHLERERFASSUNGS-SIGNAL 154
SPEICHER
SPEICHER 30
SPEICHER 31
SYSTEM A
HOCHZÄHLUNG
ABWÄRTSZÄHLUNG
ZÄHLWERT
4
3
2
1
FEHLERERFASSUNGS-SIGNAL 164
SPEICHER
SPEICHER 40
SPEICHER 41
SYSTEM B

FIG. 11

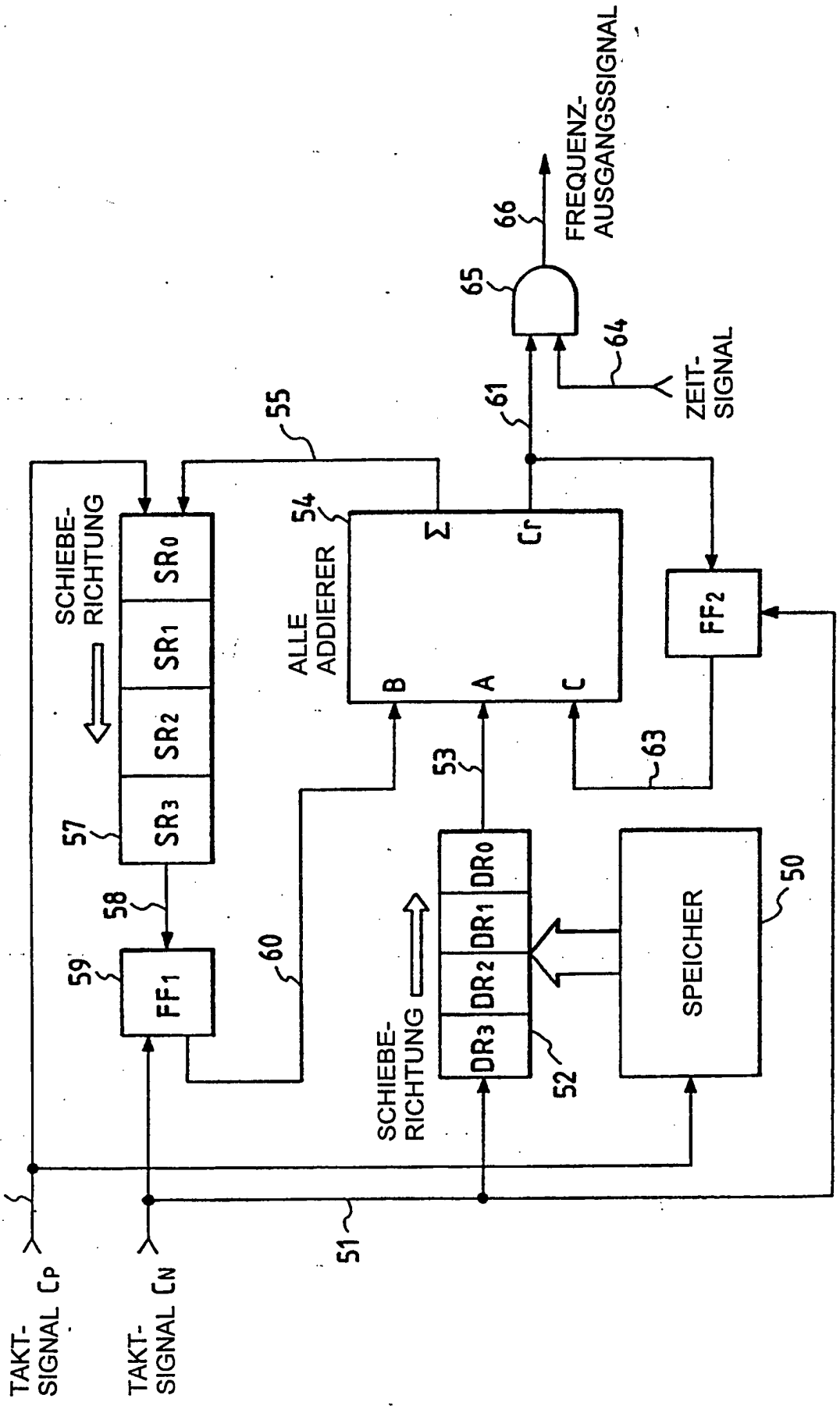

FREQUENZ-
AUSGANGSSIGNAL
66
65
64
ZEIT-
SIGNAL
61
55
54
ALLE
ADDIERER
Σ
Cr
B
A
C
SCHIEBE-
RICHTUNG
SR0
SR1
SR2
SR3
57
58
59
FF1
60
FF2
63
53
DR0
DR1
DR2
DR3
SCHIEBE-
RICHTUNG
52
SPEICHER
50
56
51
TAKT-
SIGNAL CP
TAKT-
SIGNAL CN

FIG. 12

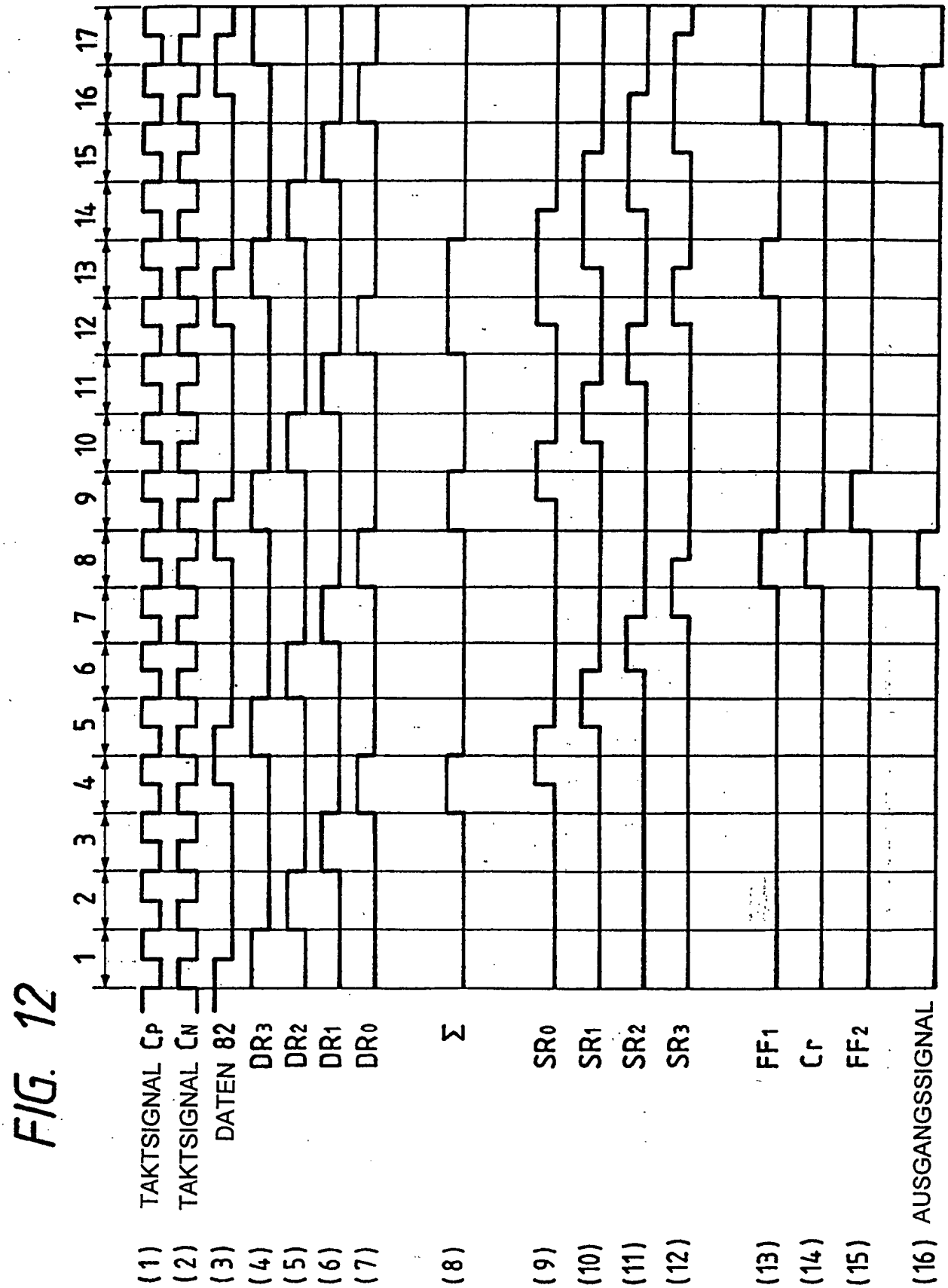

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
(1) TAKTSIGNAL C_P
(2) TAKTSIGNAL C_N
(3) DATEN 82
(4) DR_3
(5) DR_2
(6) DR_1
(7) DR_0
(8) Σ
(9) SR_0
(10) SR_1
(11) SR_2
(12) SR_3
(13) FF_1
(14) C_r
(15) FF_2
(16) AUSGANGSSIGNAL

FIG. 13

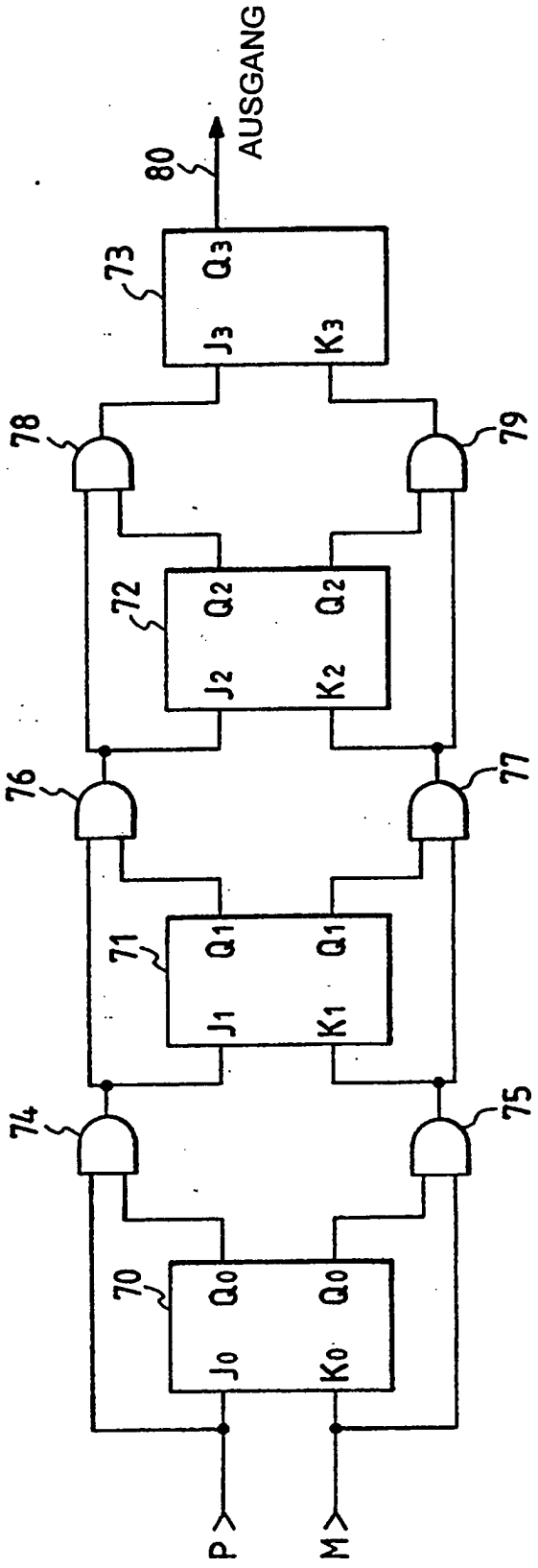

AUSGANG
80
73
Q3
J3
K3
78
79
72
Q2
Q2
J2
K2
76
77
71
Q1
Q1
J1
K1
74
75
70
Q0
Q0
J0
K0
P
M

FIG. 14

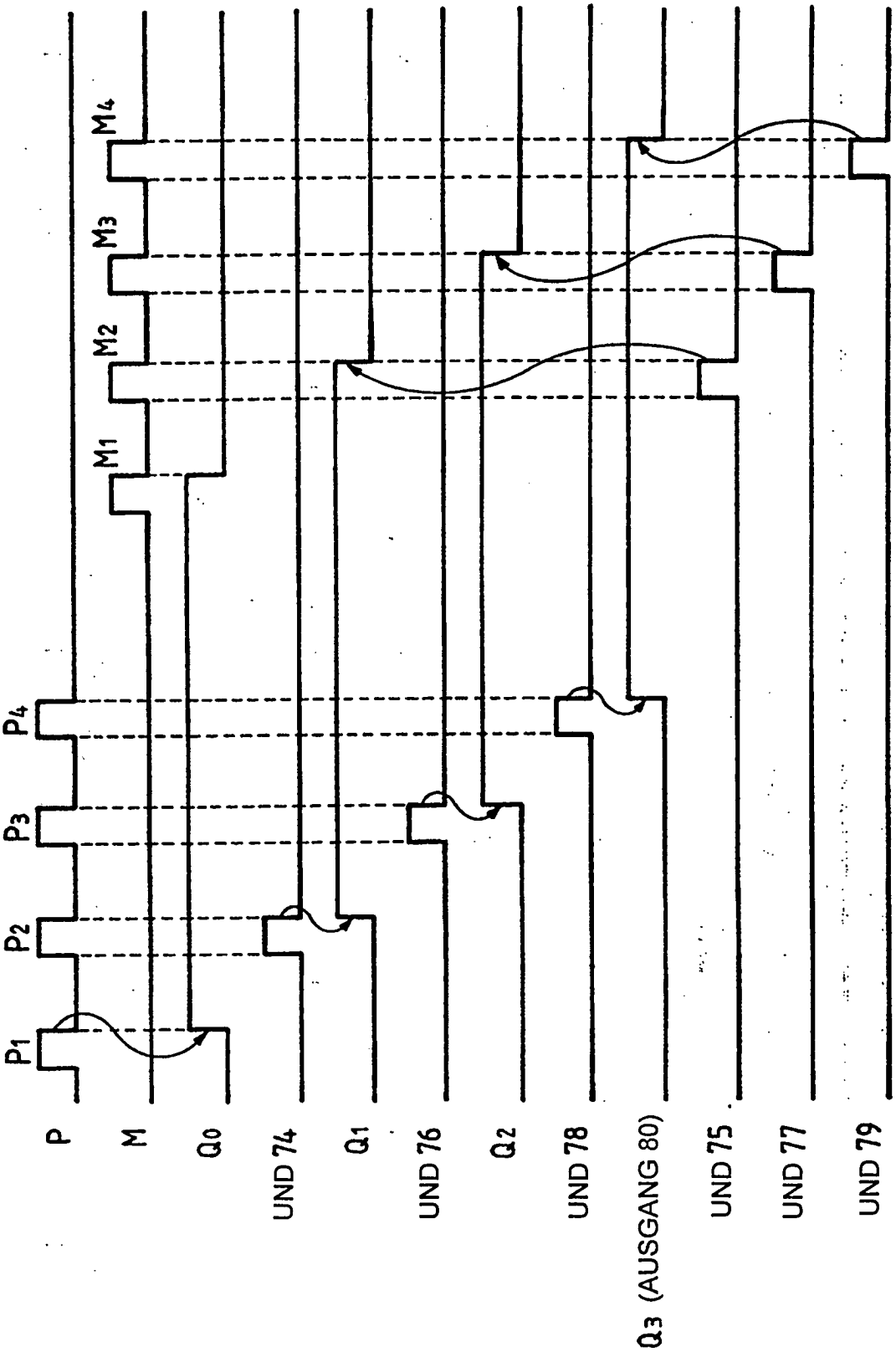

P1
P2
P3
P4
M1
M2
M3
M4
P
M
Q0
UND 74
Q1
UND 76
Q2
UND 78
Q3 (AUSGANG 80)
UND 75
UND 77
UND 79

FIG. 15

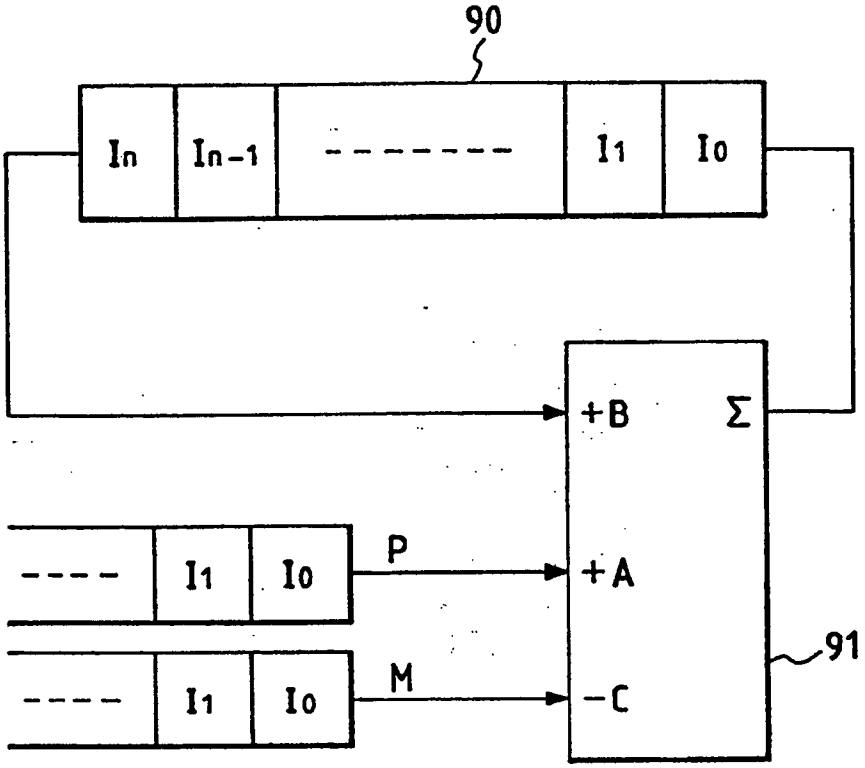


FIG. 16

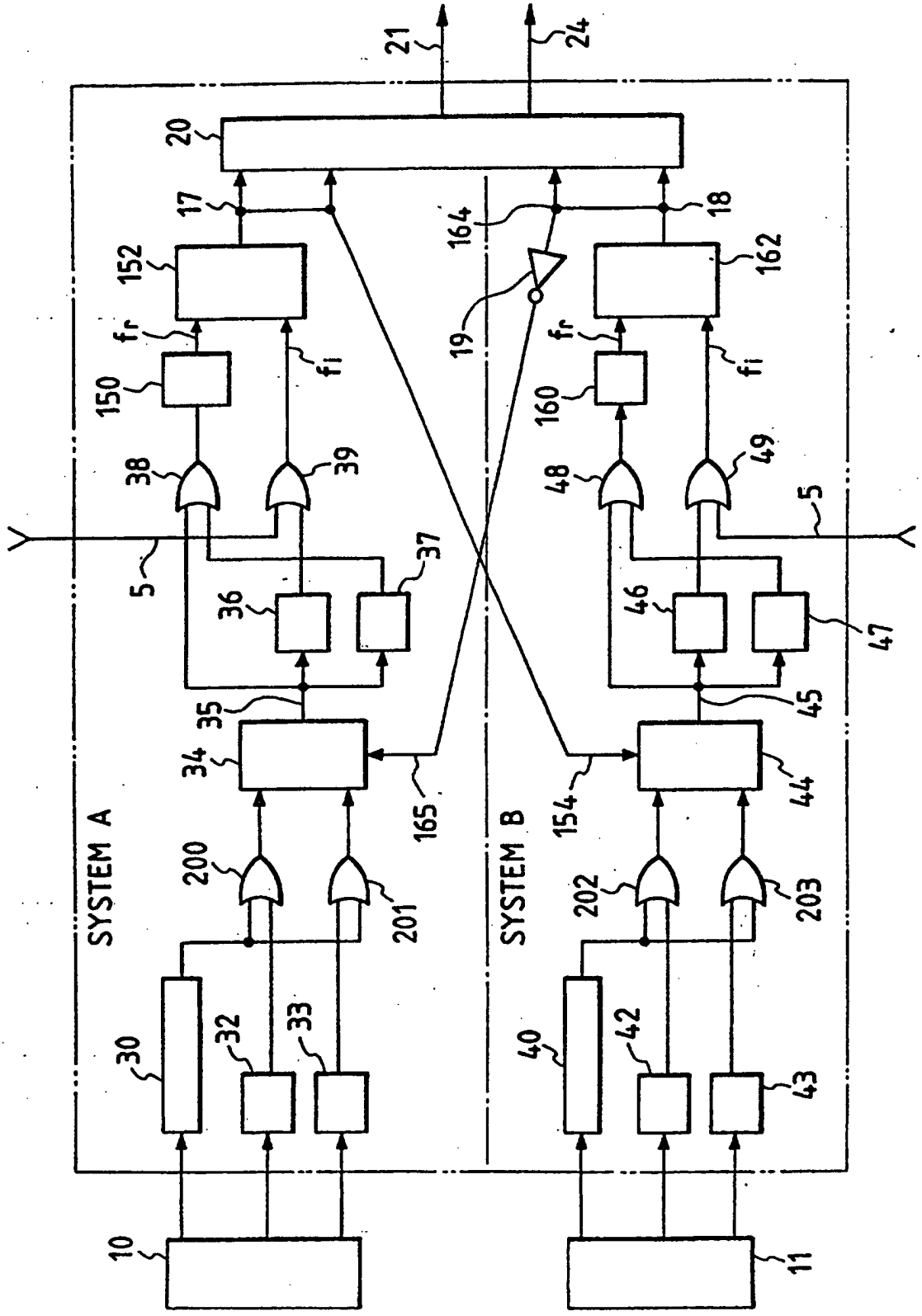

SYSTEM A
SYSTEM B
21
24
20
17
18
164
152
162
fr
fi
19
150
160
38
39
48
49
5
37
36
46
47
35
45
34
44
165
154
200
201
202
203
30
32
33
40
42
43
10
11

FIG. 17

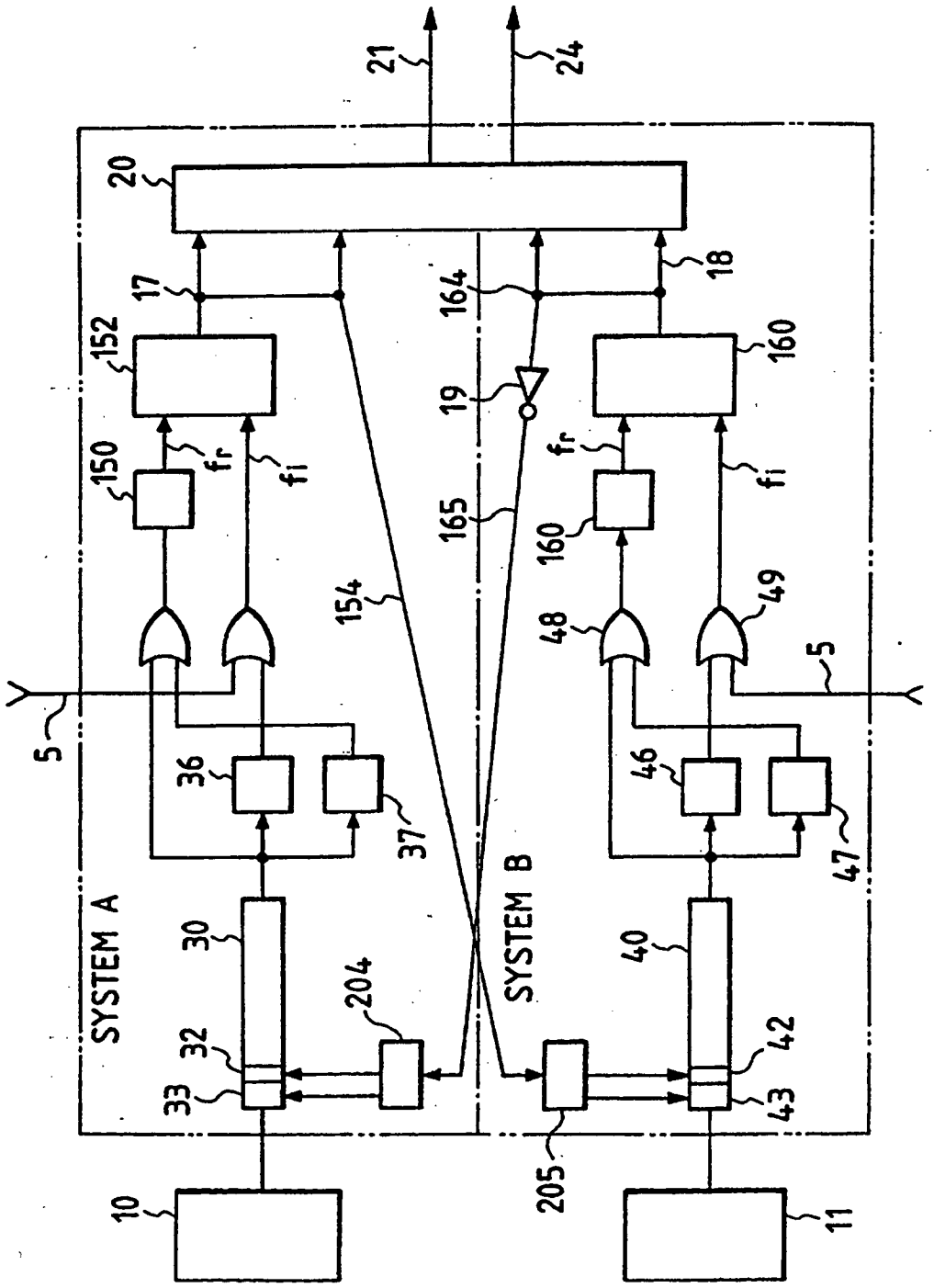

SYSTEM A
SYSTEM B
10
11
20
21
24
17
18
19
150
152
160
154
164
165
fr
fi
5
30
32
33
36
37
40
42
43
46
47
48
49
204
205

FIG. 18

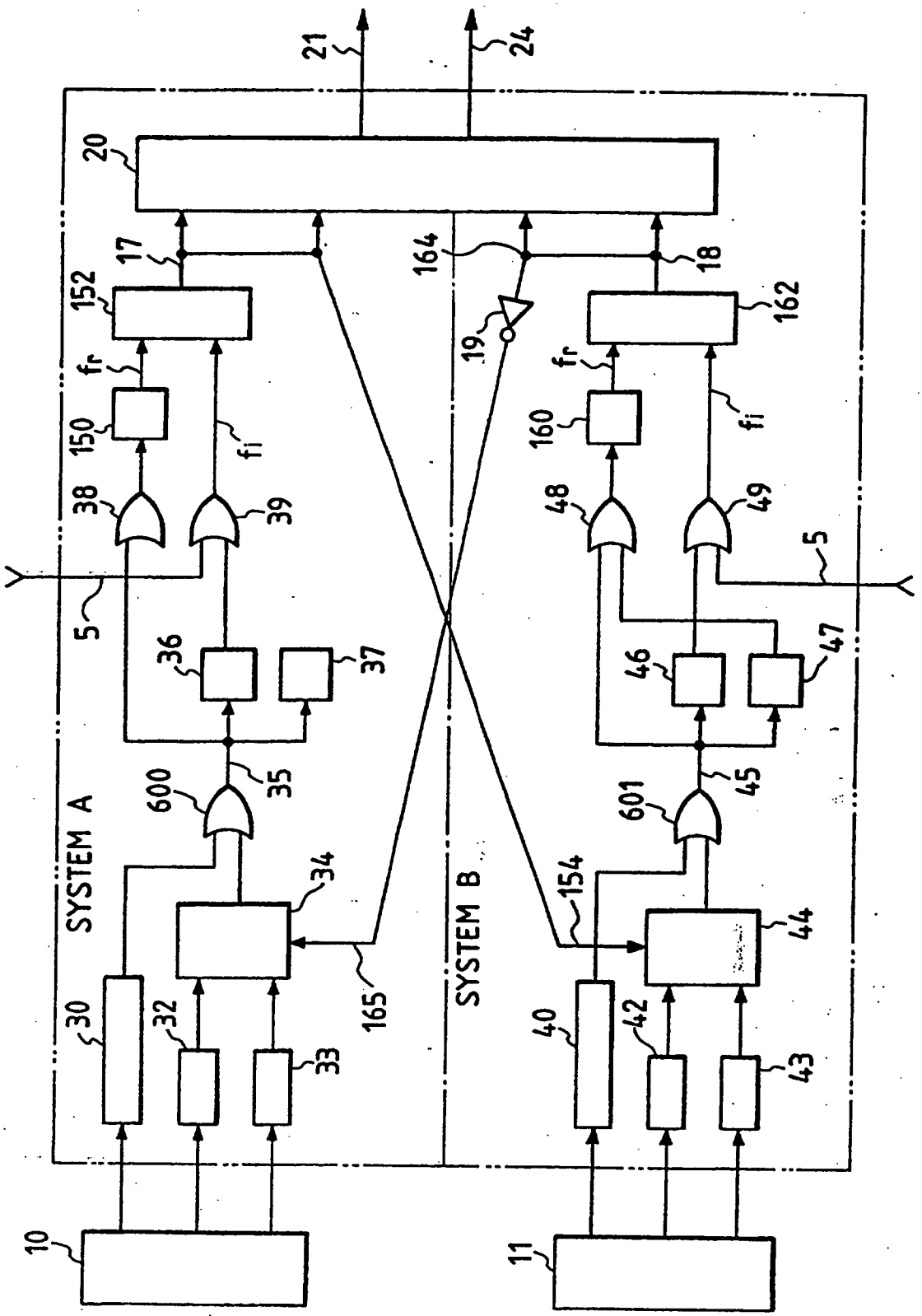

SYSTEM A
SYSTEM B
21
24
20
17
18
164
152
162
19
fr
fi
150
160
38
39
48
49
5
Y
36
37
46
47
600
601
35
45
34
154
44
165
30
32
33
40
42
43
10
11

FIG. 19

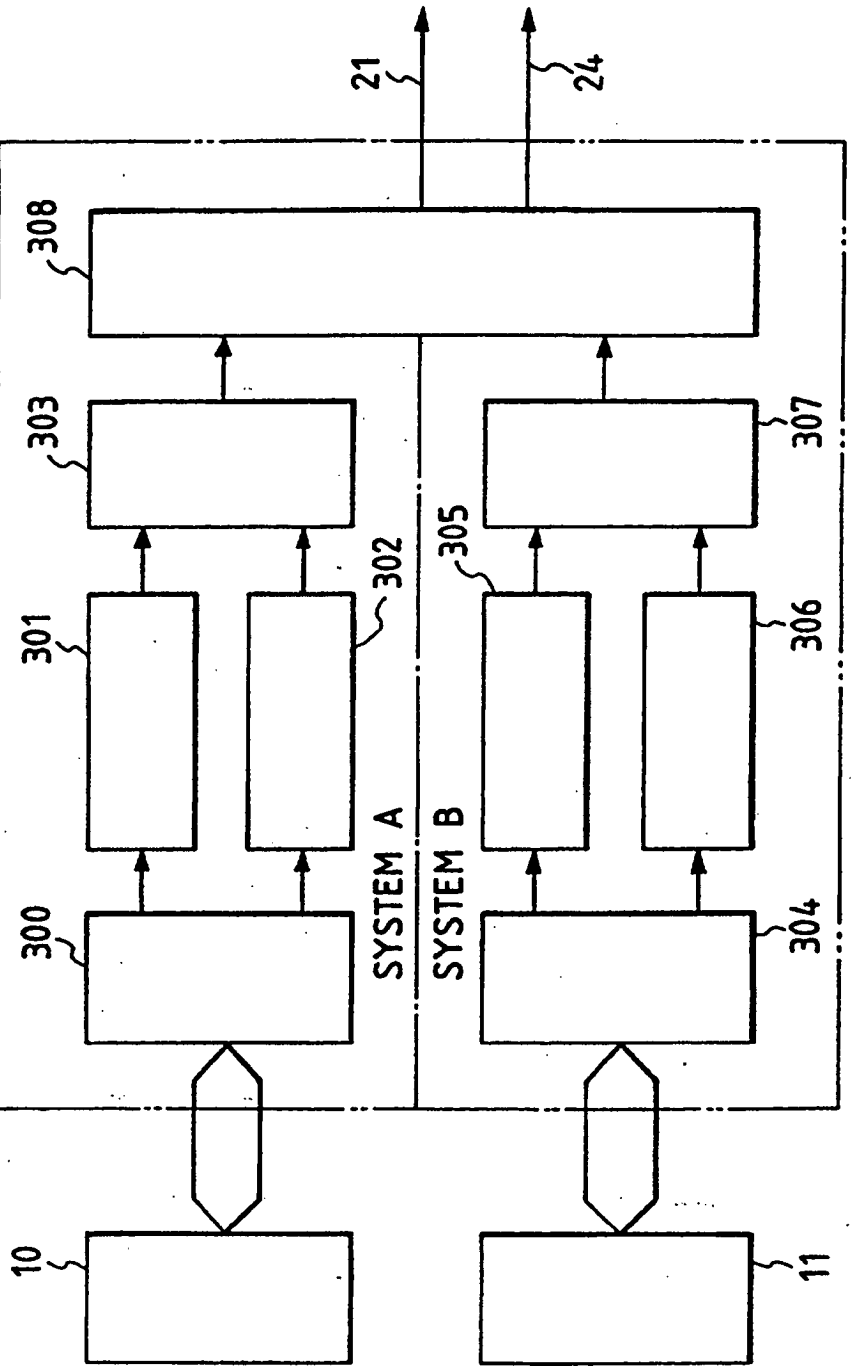

21
24
308
303
307
14
305
302
301
306
SYSTEM A
SYSTEM B
300
304
10
11

FIG. 20

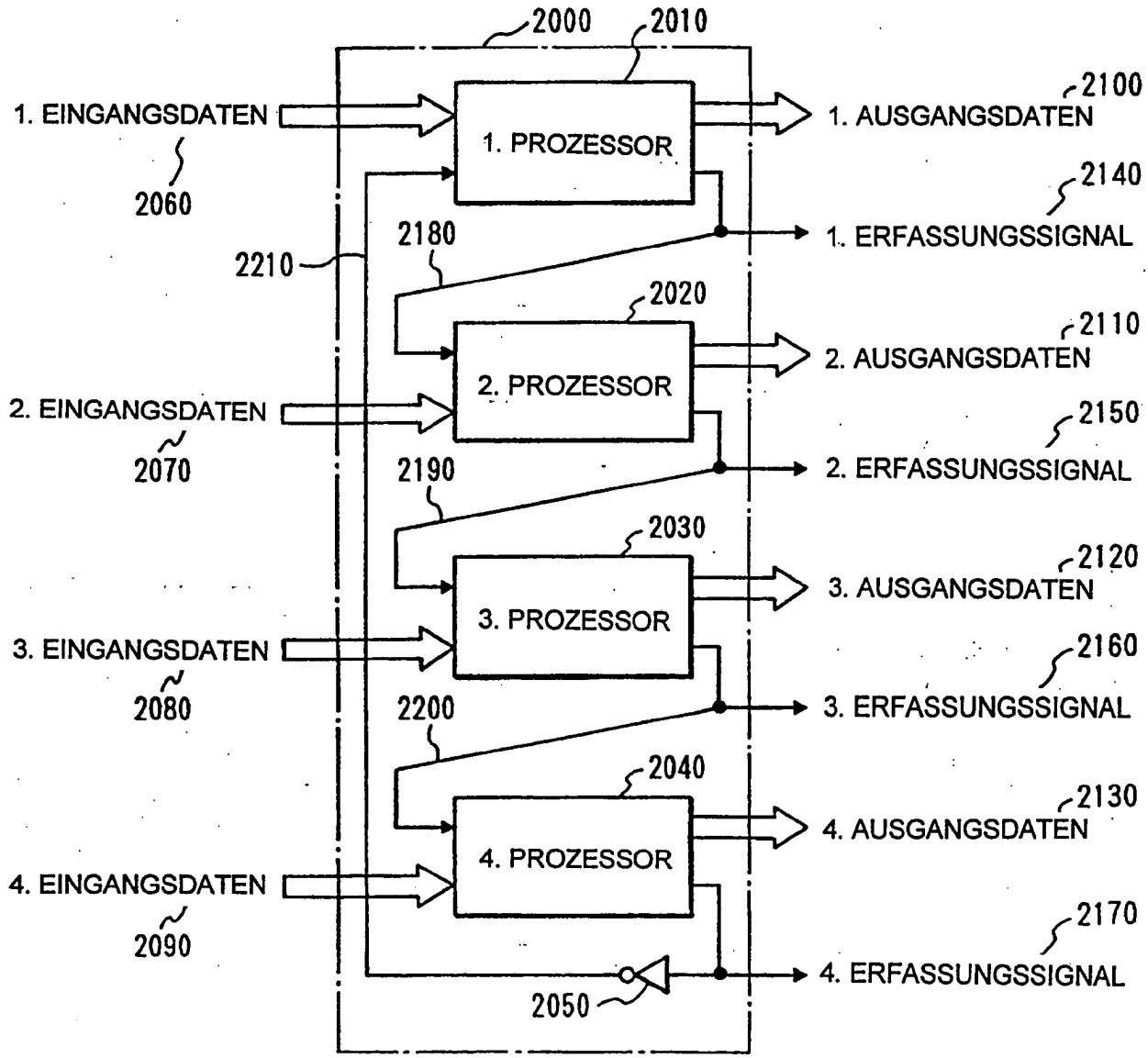

2000
2010
1. EINGANGSDATEN
2060
1. PROZESSOR
2100
1. AUSGANGSDATEN
2140
1. ERFASSUNGSSIGNAL
2180
2210
2020
2. PROZESSOR
2110
2. AUSGANGSDATEN
2. EINGANGSDATEN
2070
2150
2. ERFASSUNGSSIGNAL
2190
2030
3. PROZESSOR
2120
3. AUSGANGSDATEN
3. EINGANGSDATEN
2080
2160
3. ERFASSUNGSSIGNAL
2200
2040
4. PROZESSOR
2130
4. AUSGANGSDATEN
4. EINGANGSDATEN
2090
2170
4. ERFASSUNGSSIGNAL
2050

FIG. 21

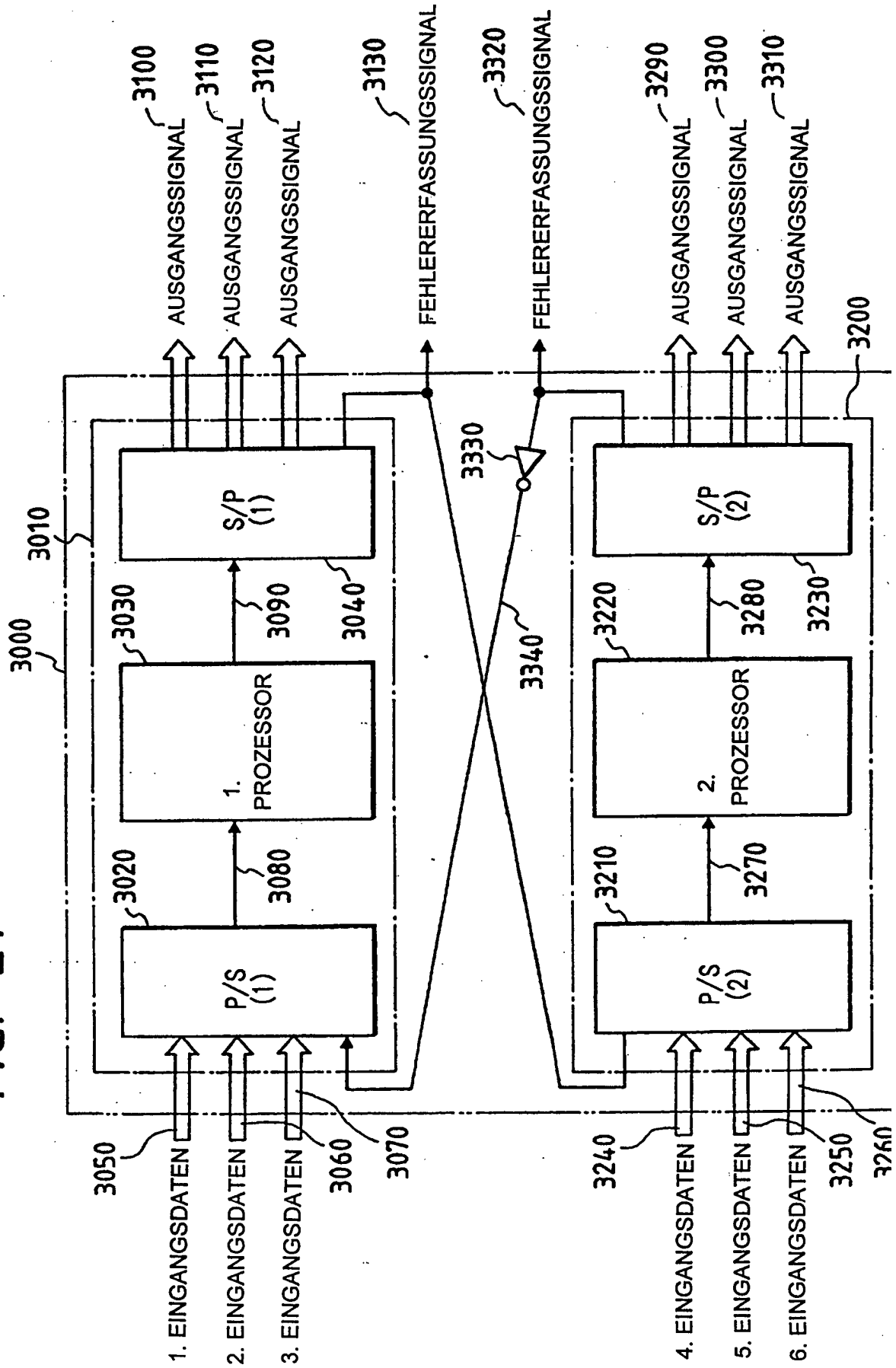

3100
AUSGANGSSIGNAL
3110
AUSGANGSSIGNAL
3120
AUSGANGSSIGNAL
3130
FEHLERERFASSUNGSSIGNAL
3320
FEHLERERFASSUNGSSIGNAL
3290
AUSGANGSSIGNAL
3300
AUSGANGSSIGNAL
3310
AUSGANGSSIGNAL
3200
3010
3000
3030
3090
3040
3330
3340
3220
3280
3230
S/P (1)
S/P (2)
1. PROZESSOR
2. PROZESSOR
3020
3080
3210
3270
P/S (1)
P/S (2)
3050
3060
3070
3240
3250
3260
1. EINGANGSDATEN
2. EINGANGSDATEN
3. EINGANGSDATEN
4. EINGANGSDATEN
5. EINGANGSDATEN
6. EINGANGSDATEN

FIG. 22

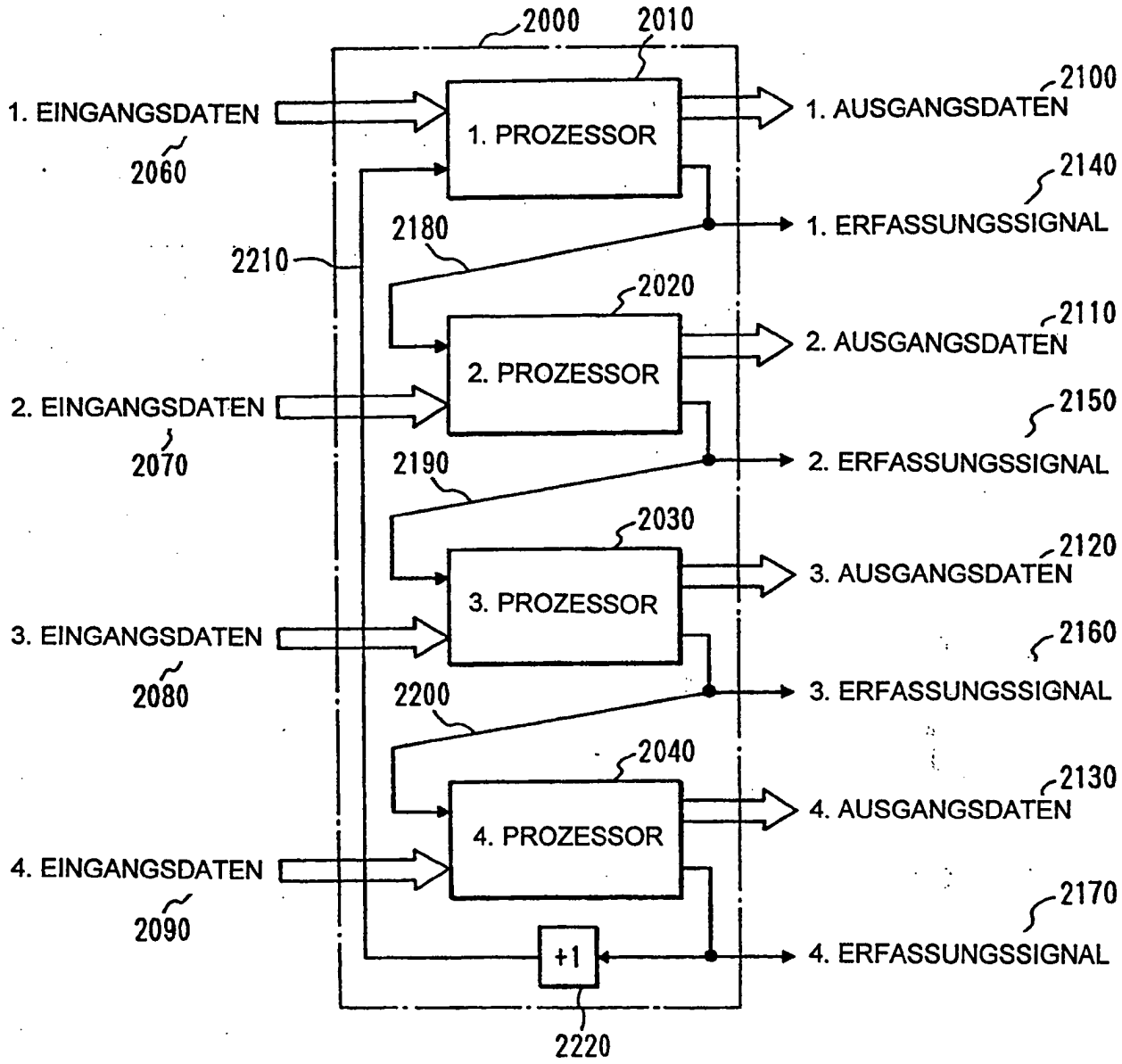

2000
2010
1. EINGANGSDATEN
2060
1. PROZESSOR
2100
1. AUSGANGSDATEN
2140
1. ERFASSUNGSSIGNAL
2180
2210
2020
2. PROZESSOR
2110
2. AUSGANGSDATEN
2. EINGANGSDATEN
2070
2150
2. ERFASSUNGSSIGNAL
2190
2030
3. PROZESSOR
2120
3. AUSGANGSDATEN
3. EINGANGSDATEN
2080
2160
3. ERFASSUNGSSIGNAL
2200
2040
4. PROZESSOR
2130
4. AUSGANGSDATEN
4. EINGANGSDATEN
2090
2170
+1
4. ERFASSUNGSSIGNAL
2220

FIG. 23

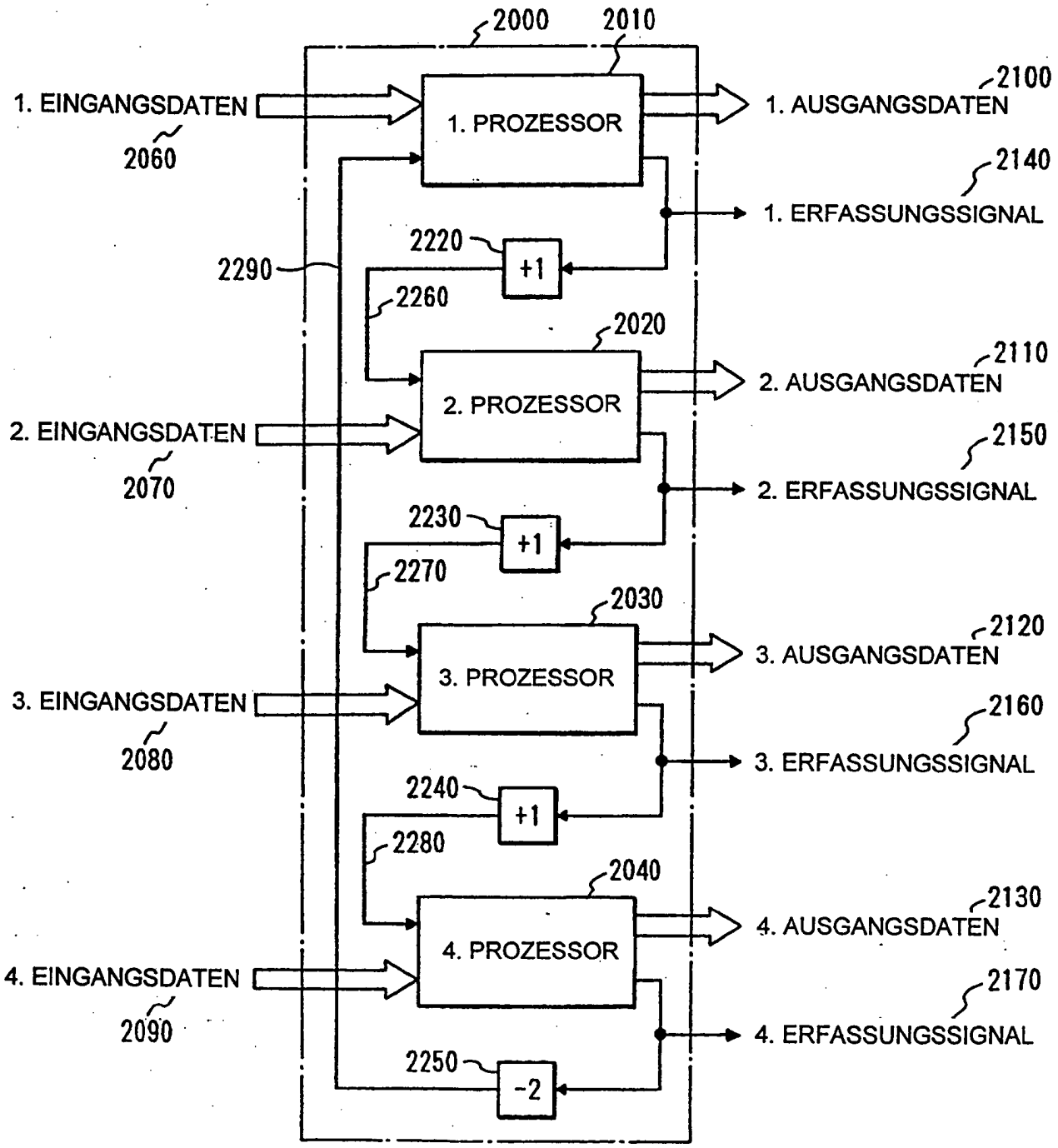

2000
2010
1. EINGANGSDATEN
2060
1. PROZESSOR
1. AUSGANGSDATEN
2100
2140
1. ERFASSUNGSSIGNAL
2220
+1
2290
2260
2020
2. PROZESSOR
2. AUSGANGSDATEN
2110
2. EINGANGSDATEN
2070
2150
2. ERFASSUNGSSIGNAL
2230
+1
2270
2030
3. PROZESSOR
3. AUSGANGSDATEN
2120
3. EINGANGSDATEN
2080
2160
3. ERFASSUNGSSIGNAL
2240
+1
2280
2040
4. PROZESSOR
4. AUSGANGSDATEN
2130
4. EINGANGSDATEN
2090
2170
4. ERFASSUNGSSIGNAL
2250
-2

FIG. 24

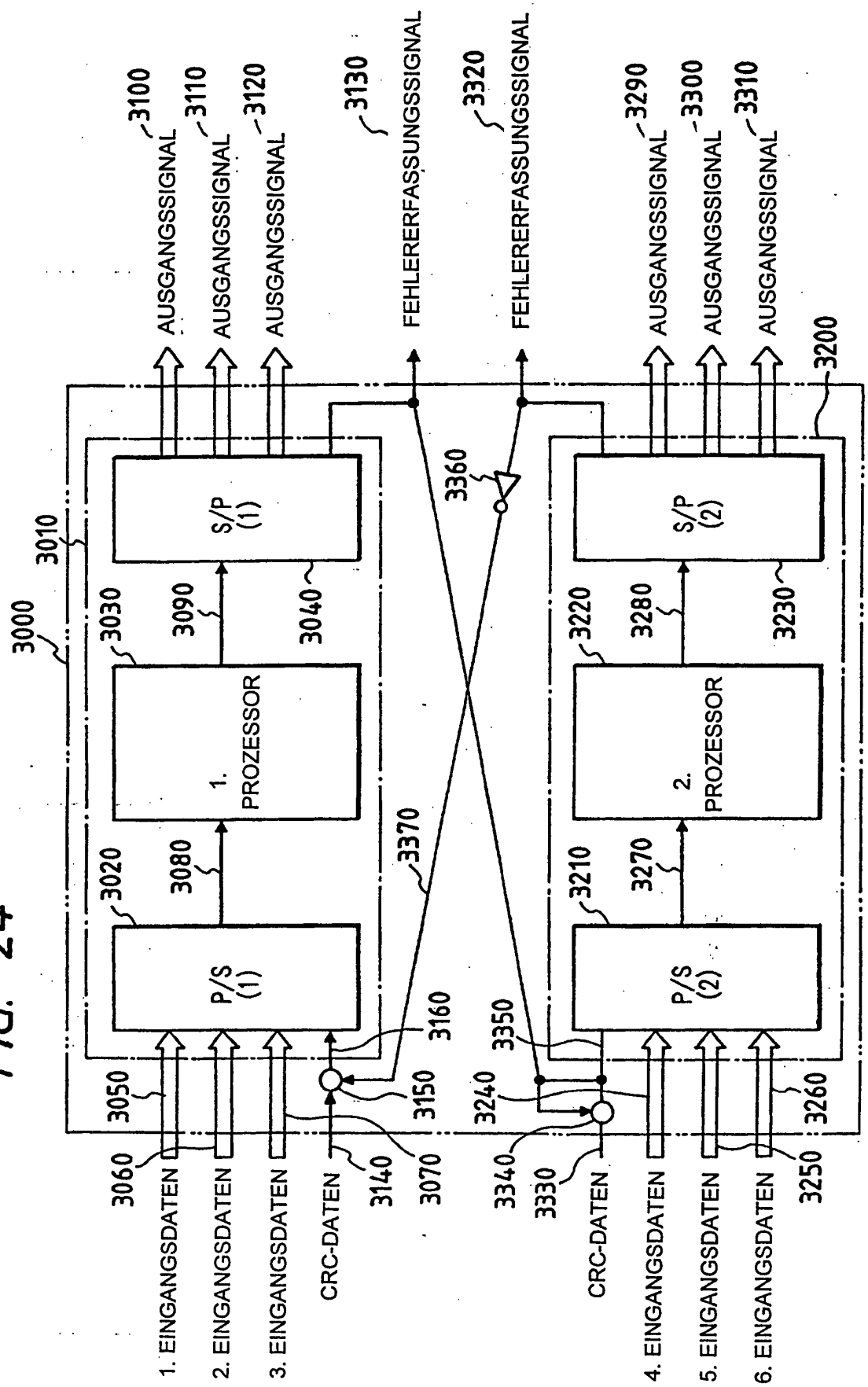
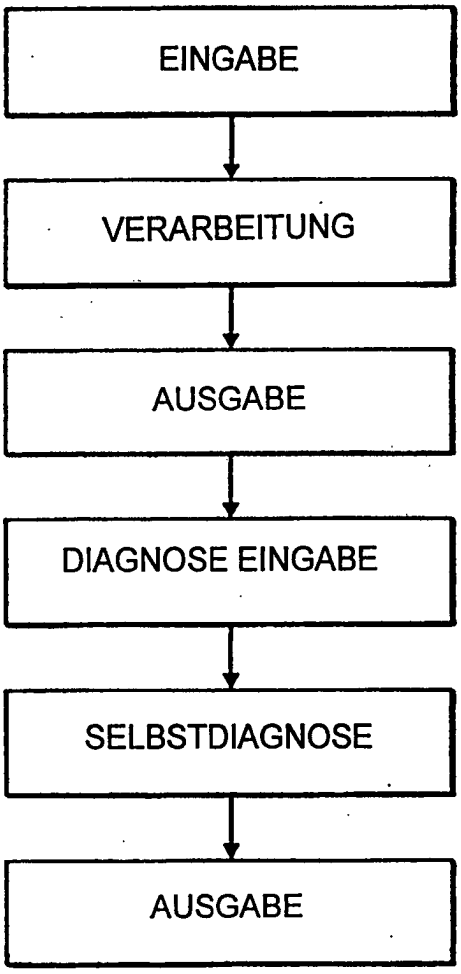

3100
AUSGANGSSIGNAL
3110
AUSGANGSSIGNAL
3120
AUSGANGSSIGNAL
3130
FEHLERERFASSUNGSSIGNAL
3320
FEHLERERFASSUNGSSIGNAL
3290
AUSGANGSSIGNAL
3300
AUSGANGSSIGNAL
3310
AUSGANGSSIGNAL
3200
3000
3010
S/P (1)
S/P (2)
3030
3090
3040
3220
3280
3230
1. PROZESSOR
2. PROZESSOR
3020
3080
3210
3270
P/S (1)
P/S (2)
3360
3370
3160
3350
3050
3150
3240
3260
3060
3140
3070
3340
3330
3250
1. EINGANGSDATEN
2. EINGANGSDATEN
3. EINGANGSDATEN
CRC-DATEN
CRC-DATEN
4. EINGANGSDATEN
5. EINGANGSDATEN
6. EINGANGSDATEN

FIG. 25


EINGABE
VERARBEITUNG
AUSGABE
DIAGNOSE EINGABE
SELBSTDIAGNOSE
AUSGABE