



SCHWEIZERISCHE EIDGENOSSENSCHAFT
BUNDESAMT FÜR GEISTIGES EIGENTUM

⑪ CH 658 351 A5

⑤① Int. Cl.4: H 03 M 13/02

Erfindungspatent für die Schweiz und Liechtenstein
Schweizerisch-liechtensteinischer Patentschutzvertrag vom 22. Dezember 1978

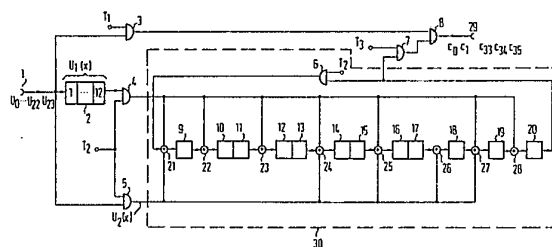
⑫ PATENTSCHRIFT A5

⑳① Gesuchsnummer:	1675/82	⑦② Inhaber:	International Standard Electric Corporation, New York/NY (US)
⑳② Anmeldungsdatum:	11.02.1982		
⑳③ Priorität(en):	11.02.1981 DE 3104762	⑦③ Erfinder:	Roth, Dieter, Pforzheim (DE) Schorb, Herbert, Karlsruhe (DE) Kloss, Herbert, Wörth (DE)
⑳④ Patent erteilt:	31.10.1986		
④⑤ Patentschrift veröffentlicht:	31.10.1986	⑦④ Vertreter:	Dipl.-El.-Ing. Hans F. Bucher, Bern

⑤④ Verfahren zur binären Datenübertragung mit Fehlererkennung und -korrektur.

⑤⑦ In einem System zur binären Datenübertragung wird ein Code mit aus n-Bits bestehenden Codewörtern eines quasizyklischen Code mit k Informationsbits und n-k Prüfbits verwendet.

Die Codewörter haben $3i$ Bits für n und $2i$ Bits für k, wobei i eine ganze Zahl grösser oder gleich 4 ist. Der dazu verwendete Coder weist einen Nutzdateneingang (1), einen Ausgang (29) und drei Takteingänge (T_1 - T_3) auf. Ankommende Daten gelangen teils an einen Pufferspeicher (2) und über Logikmittel (3, 8) an den Ausgang, teils über weitere Logikmittel (3, 4) an eine Coderschaltung (30). Diese besteht aus modulo-2-Addierern (21 - 28) und Schieberegisterstreifen (9 - 20).



PATENTANSPRÜCHE

1. Verfahren zur binären Datenübertragung mit einem Coder und einem Decoder mit Fehlererkennungs- und -korrektur-Einrichtung, dadurch gekennzeichnet, dass der Coder das zu übertragende binäre Eingangssignal mittels eines $k/2$ -zelligen Schieberegisters (30) in n Bit umfassende Codewörter eines quasizyklischen Codes mit k Informationsbits und der Decoder diesen empfangenen Code in das übertragene binäre Ausgangssignal umsetzt, dass in der Fehlererkennungs- und -korrektur-Einrichtung aus jedem Codewort mittels eines $k/2$ -zelligen Schieberegisters (46) ein Syndrom berechnet wird, dessen Wert als Adresse eines Festwertspeichers (45) mit 2^{n-k} Speicherplätzen verwendet wird, wodurch entweder eine Information ausgegeben wird, die die Position eines fehlerhaften Bits in den Informationsbits angibt und diese Information in einem Umcoder (47) in ein k Bit umfassendes Fehlermuster umgesetzt wird oder eine Information ausgegeben wird, die die Korrektur beendet, dass durch modulo-2-Addition des Fehlermusters zu den Informationsbits korrigierte Informationsbits erzeugt werden, dass nach Berechnung neuer Syndrome aus den korrigierten Informationsbits in jeweils weiteren Korrekturschritten jeweils ein weiterer Fehler in den Informationsbits korrigiert wird und dass nach einer wählbaren Anzahl von derartigen Korrekturschritten oder beim Auffinden der Information zum Beenden der Korrektur das Korrekturverfahren beendet und das korrigierte Ausgangssignal ausgegeben wird.
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass zur Einsparung von Speicherplatz in einem Festwertspeicher (34) das mit dem $k/2$ -zelligen Schieberegister (46) berechnete Syndrom in einem weiteren Schieberegister (31) verarbeitet wird und das Verarbeitungsergebnis als Adresse des Festwertspeichers (34) dient, unter der in diesem Festwertspeicher das Ende der Korrekturschritte vermerkt ist oder mit $\log_2 k$ Bits die Position eines fehlerhaften Bits in der Nutzinformation angegeben ist und diese in dem Umcoder (33) und im Schieberegister (31) in das zugehörige k Bits umfassende Fehlermuster umgesetzt wird.
3. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass der quasizyklische Code aus Codewörtern mit $3i$ Bits für n und $2i$ Bits für k gebildet ist, wobei i eine ganze Zahl grösser oder gleich 4 ist.
4. Verfahren nach Anspruch 3, dadurch gekennzeichnet, dass der quasizyklische Code aus Codewörtern mit $n = 36$ Bits, von denen $k = 24$ Bits Informationsbits und 12 Bits Prüfbits sind, besteht und eine Redundanz-Matrix aus zwei gleich grossen zyklischen Teilmatrizen zusammengesetzt ist.
5. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass bei einem Simplex-Sende/Empfangsgerät das Schieberegister (46) im Decoder als Schieberegister (30) im Coder eingesetzt wird.
6. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass der Decoder in einer ersten Taktphase (I) mit einem langsameren Takt als in einer zweiten Taktphase (II) betrieben wird.
7. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass die Anzahl der Korrekturschritte drei beträgt.

Die Erfindung bezieht sich auf ein Verfahren zur binären Datenübertragung mit einem Coder und einem Decoder mit Fehlererkennungs- und -korrektur-Einrichtung.

Bei solchen Verfahren besteht aufgrund unvermeidlicher Störungen immer eine gewisse Wahrscheinlichkeit für den Empfang von falschen Zeichen. Um bei der Decodierung verfälschte Binärinformation zu erkennen und zu korri-

gieren, ist eine Vielzahl von Verfahren bekannt geworden, so z.B. in dem Buch «Error-Correcting-Codes» von W.W. Peterson und E.I. Weldon, The M.I.T. Press, Cambridge, Mass./USA second edition, 1972 auf den Seiten 289 bis 308.

- 5 Mit den dort beschriebenen Decodierungsverfahren von binären Bose-Chaudhuri-Hocquenghem-Codes (BCH-Codes), die auf dem Berlekamp-Massey-Algorithmus beruhen, können in einem Codewort in Abhängigkeit von Minimalabstand m zwischen allen Codewörtern alle höchstens f -fachen Fehler korrigiert werden. Es gilt dabei $m = 2f + 1$.

Verwendet man binäre BCH-Codes, so lassen sich die Coder und Decoder aus einfachen Schieberegisterschaltungen aufbauen, da BCH-Codes sogenannte zyklische Codes sind.

- 15 Decoder-Schaltungen für verkürzte BCH-Codes, die nicht mehr zyklisch sind, erfordern dagegen einen zusätzlichen Aufwand in Form von etwas komplizierteren logischen Schaltungen und deren Steuerung.

Es ist nun Aufgabe der Erfindung, ein Verfahren der eingangs genannten Art anzugeben, das im Decoder die potentielle Fehlerkorrekturfähigkeit des Codes ausnutzt, minimalen Speicherplatz benötigt und dabei die Anzahl der Korrekturschritte möglichst klein hält.

- 25 Die Lösung dieser Aufgabe erfolgt mit den im Anspruch 1 angegebenen Mitteln. Vorteilhaft Weiterbildungen können den abhängigen Ansprüchen entnommen werden.

Durch die Fehlerkorrekturfähigkeit des Decoders wird bei gleichem Verhältnis zwischen binärer Prüf- und Nutzinformation eine grössere Übertragungssicherheit als mit den bekannten BCH-Codes bei gleichen Parametern n und k erreicht. Bei gleichen Forderungen an die Übertragungssicherheit wie bei BCH-Codes ist die Nutzinformation in einem Codewort zu Lasten der Prüfinformation grösser. Dies führt dazu, dass die Menge an Nutzinformation, die in der Zeiteinheit übertragen werden kann, zunimmt.

Durch die schrittweise Fehlerkorrektur kommt man zudem mit etwa einem Siebentel des die Korrekturinformation enthaltenen Speicherplatzes aus wie bei den bekannten Verfahren.

- 40 Ein weiterer Vorteil besteht in der Verwendung quasizyklischer Codes, weil sich dadurch die Codierung mit relativ einfachen Schieberegisterschaltungen, ähnlich wie bei zyklischen Codes, realisieren lässt. Ausserdem kann die Codierung als wesentlicher Teil zur Berechnung der Syndrome im Decoder verwendet werden.

Bei der Decodierung des verwendeten Codes werden alle bis zu zweifachen Fehler und ein erheblicher Teil der dreifachen Fehler im Informationsteil des empfangenen Codewortes korrigiert. Das Decoderschema ist so ausgelegt, dass sich von den korrigierbaren 4fachen und 5fachen Fehlern im empfangenen Wort höchstens 3fache Fehler im Informationsteil befinden können und damit diese Fehler mit höchstens 3 Schritten korrigiert werden. Vergleichbare BCH-Codes können mit einer auf dem Berlekamp-Massey-Algorithmus beruhenden Schaltung nur alle höchstens zweifachen Fehler korrigieren. Weiterhin werden für solche zyklischen BCH-Codes im Decoder komplizierte Rechen- und Logikschaltungen benötigt.

- 60 Ein zusätzlicher Vorteil ergibt sich aus der Tatsache, dass von allen möglichen korrigierbaren Fehlermustern sich höchstens 3 Bitfehler im informationstragenden Teil des Codewortes befinden.

Da eine klare zeitliche Trennung von zwei verschiedenen Taktphasen möglich ist, kann dadurch eine Beschleunigung der Decodierung erreicht werden.

Die Erfindung wird anhand von Zeichnungen eines Ausführungsbeispiels näher erläutert. Es zeigt:

Fig. 1 ein Blockschaltbild eines Coders für einen quasizyklischen (36,24)-Code;

Fig. 2 im Blockschaltbild des Coders gemäss Fig. 1 erwähnte Takte T_1 , T_2 und T_3 ;

Fig. 3 ein Blockschaltbild eines Decoders für den quasizyklischen (36,24)-Code;

Fig. 4 im Blockschaltbild des Decoders gemäss Fig. 3 erwähnte Takte T_4 bis T_9 und

Fig. 5 ein Blockschaltbild eines anderen Schaltkreises im gestrichelt umrahmten Teil der Fig. 3.

Die Erfindung wird an einem Ausführungsbeispiel mit einem quasizyklischen (36,24)-Code beschrieben. Dieser Code besteht aus $n = 36$ Bits je Codewort mit $k = 24$ Bits als Nutzinformation und $n-k = 12$ Bits als Prüfinformation. Der Code heisst quasizyklisch, weil sich die Redundanz-Matrix in zyklische Teilmatrizen zerlegen lässt. Allgemein gesagt besteht der quasizyklische Code aus Codewörtern mit $3i$ Bits für n und $2i$ Bits für k , wobei i eine ganze Zahl grösser oder gleich 4 ist.

Obwohl der quasizyklische (36,24)-Code kein zyklischer Code ist und der Code zunächst nur in Form der Generatormatrix vorliegt, muss zur Bildung eines Codewortes keine in der Coder-Schaltung nur sehr aufwendig realisierbare Matrizenmultiplikation durchgeführt werden. Durch die speziellen Eigenschaften des Codes und die spezielle Zerlegung des Informationsteils eines Codewortes genügt es, ein rückgekoppeltes Schieberegister mit einem Zusatzspeicher zu verwenden.

Im Decoder wird das Syndrom des empfangenen Wortes berechnet. Unter einem Syndrom versteht man einen Zeilenvektor mit r Komponenten, der dadurch entsteht, dass man ein Codewort eines (n,k) -Codes als $(1,n)$ -Matrix (Codevektor) auffasst und diesen Codevektor mit der transponierten (r,n) Paritäts-Prüfmatrix multipliziert.

Für r gilt dabei $r = n-k$. Ist das Syndrom nicht gleich dem Nullvektor, besteht eine Unvereinbarkeit zwischen codierter und decodierter Information, wie sie zum Beispiel durch Übertragungsfehler hervorgerufen wird.

Für den verwendeten Code ergeben sich 2^{12} verschiedene mögliche Syndrome. Jedem der $2^{12} = 4096$ Syndrome ist die Position eines Bitfehlers im Informationsteil des empfangenen Wortes oder Null zugeordnet. Dabei gibt der Wert des Syndroms die Adresse eines Speicherplatzes an, der normalerweise die Position eines falsch übertragenen Bits enthält. Enthält der Speicherplatz das Nullwort, so bedeutet dies, dass höchstens im Redundanzteil noch Bitfehler liegen können. Weitere Syndromberechnungen werden dann nicht mehr durchgeführt. Da nach der Decodierung lediglich der Informationsteil von 24 Bit von Interesse ist, genügen 5 Bits zur Kennzeichnung der Position eines Bitfehlers, was sich aus $\log_2 k$ errechnet, als diejenige ganze Zahl, die grösser oder höchstens gleich diesem Wert ist. Nach der eventuell erfolgten Korrektur eines Bitfehlers wird erneut das Syndrom des korrigierten Wortes berechnet und ein wie oben beschriebener Vorgang durchgeführt. Nach höchstens drei Korrekturschritten, das heisst nach dreimaliger Berechnung des Syndroms, sind im Informationsteil des empfangenen Wortes alle korrigierbaren Fehler beseitigt worden. Das bedeutet, dass im Informationsteil bis zu 3 Bitfehler korrigiert wurden.

Der allgemeine Weg zur Codierung von binärer Nutzinformation eines Codewortes besteht darin, dieses als Vektor aufzufassen und mit der Generatormatrix zu multiplizieren. Bei zyklischen Codes besteht eine Analogie zwischen Vektor und Polynomdarstellung, was die Codierung und Decodierung dieser Codes wesentlich erleichtert. Die einfachen Schieberegisterschaltungen, wie sie auch in der Erfindung beim Coder

und Decoder verwendet werden, erlauben nämlich unter anderem die Addition, Multiplikation und Division von Polynomen und damit die einfache Codierung von zyklischen bzw. quasizyklischen Codes.

Zur Codierung der Nutzinformation des verwendeten quasizyklischen (36,24)-Code, das heisst der Nutzbits $U_0, U_1, \dots, U_{23}$, müsste man diese daher mit der Generatormatrix G multiplizieren, da dieser Code nicht zyklisch ist.

Zerlegt man aber die Nutzinformation in ihre beiden Hälften $U_0, U_1, \dots, U_{11}$ und $U_{12}, U_{13}, \dots, U_{23}$, ist auch hier der Übergang zur Polynomdarstellung möglich. Dann ergibt sich für die beiden Hälften der Nutzinformation $U_1(x)$ und $U_2(x)$:

$$U_1(x) = \sum_{i=0}^{11} U_{12+i} \cdot x^i$$

$$U_2(x) = \sum_{i=0}^{11} U_i x^i$$

Das Codepolynom $c(x)$ hat dann die Darstellung:

$$c(x) = U_1(x) \cdot x^{24} + U_2(x) \cdot x^{12} + p(x)$$

Mit $p(x)$ als Prüfpolyom, das für den gewählten Code durch die Beziehung:

$$p(x) = [U_1(x) \cdot g_1(x) + U_2(x) \cdot g_2(x)] / (1 + x^{12})$$

gebildet wird.

Aufgrund des gewählten Codes ergeben sich

$g_1(x)$ und $g_2(x)$ zu:

$$g_1(x) = 1 + x + x^3 + x^5 + x^7 + x^{10} + x^{11}$$

$$g_2(x) = 1 + x^5 + x^7 + x^9 + x^{10}$$

Fig. 1 zeigt das Blockschaltbild des Coders und Fig. 2 die dazugehörigen Takte T_1 , T_2 und T_3 , an deren Zeitachse die fortlaufenden jeweiligen Bittakte angegeben sind.

Die ersten 12 Bits (Bit U_0 bis U_{11}) der an einem Eingang 1 liegenden Nutzinformation $U_1(x)$, die von einer nicht dargestellten Datenquelle kommt, gelangen in einen Pufferspeicher 2 und gleichzeitig über ein UND-Gatter 3 mit dem Takt T_1 und einem nachfolgenden ODER-Gatter 8 direkt zu einem Ausgang 29. Der Pufferspeicher erlaubt die gleichzeitige Verarbeitung von zwei zyklischen Codehälften. Die nächsten 12 Bits (Bit U_{12} bis U_{23}) der Nutzinformation $U_2(x)$ gelangen aufgrund des Taktes T_2 über ein UND-Gatter 5 an einen modulo-2-Addierer 21. Gleichzeitig gelangt $U_1(x)$ aus dem Pufferspeicher 2 über ein UND-Gatter 4 auch an den modulo-2-Addierer 21, dessen Ergebnis gelangt an eine erste Stufe 9 eines Schieberegisters 30. Ausserdem gelangt $U_2(x)$ wieder über das UND-Gatter 3 und das ODER-Gatter 8 an den Ausgang 29. Die erste Stufe 9 mit den folgenden Stufen 10 bis 20 und der modulo-2-Addierer 21 mit den nachfolgenden modulo-2-Addierern 22 bis 28 stellen eine Codierschaltung dar. Die Codierschaltung multipliziert gleichzeitig die Polynome $U_1(x)$ mit $g_1(x)$ und $U_2(x)$ mit $g_2(x)$, addiert die Multiplikationsergebnisse und bildet als Ergebnis das Prüfpolyom $p(x)$ durch Division mit $d(x) = 1 + x^{12}$ (UND-Gatter 6, das den Ausgang der Stufe 20 mit dem modulo-2-Addierer 21 verbindet und das vom Takt T_2 gesteuert wird). Am Ende des Taktes T_2 steht das erste der 12 Prüfbits in der Schieberegisterstufe 20, nach weiteren 12 Bittakten sind die 12 Prüfbits über ein an die Stufe 20 angeschlossenes UND-Gatter 7 (Takt T_3) und das ODER-Gatter 8 der Nutzinformation angefügt worden. Am Ausgang 29 tritt auf diese Weise das Codepo-

lynom $c(x)$ bzw. die Bits c_0 bis c_{35} auf. Die sich anschließenden Sendeeinrichtungen und der Übertragungskanal sind nicht dargestellt, ebenso wie der erforderliche Taktgenerator.

Fig. 3 zeigt das Blockschaltbild des zyklischen Decoders und Fig. 4 die dazugehörigen Takte T_4 bis T_9 , an deren Zeitachse die fortlaufenden, jeweiligen Bittakte angegeben sind.

Die vor einem Eingang 40 liegenden Empfängerteile sind nicht dargestellt. Der Taktgenerator ist mit 39 bezeichnet. Vom Eingang 40 her gelangen die ersten 12 Bits des empfangenen 36 Bit-Wortes wegen des Taktes T_4 bzw. T_5 gleichzeitig über ein UND-Gatter 48 in einen Pufferspeicher 44 und über ein UND-Gatter 49 in ein Vorregister 41 mit einem Serienausgang und 12 bitparallelen Ausgängen. Der Pufferspeicher 44 erlaubt die gleichzeitige Verarbeitung von zwei zyklischen Codehälften.

Ein zweipoliger Umschalter 62 ist mit seinem ersten beweglichen Arm mit dem Ausgang eines UND-Gatters 51 und mit seinem zweiten beweglichen Arm mit dem Eingang eines UND-Gatters 52 verbunden, an die ausserdem der Takt T_7 anliegt. Der erste Ruhekontakt des Umschalters 62 ist mit dem Ausgang des Pufferspeichers 44 und der zweite Ruhekontakt mit dem Eingang 40 verbunden. Der erste Arbeitskontakt ist mit dem Ausgang eines aus zwei gleichen Hälften bestehenden, je 12 Bit langen Summenregisters 42 und der zweite Arbeitskontakt ist mit dem Mittenabgriff des Summenregisters 42 verbunden. Der Umschalter 62 wird vom Takt T_6 umgesteuert. Während des Taktes T_6 ist der zweipolige Umschalter 62 in der oberen Stellung und wegen des Taktes T_7 liegen die jeweiligen ersten und zweiten Informationshälften $U_1(x)$ bzw. $U_2(x)$ gleichzeitig über die UND-Gatter 51 bzw. 52 an den Eingängen eines rückgekoppelten Schieberegisters 46, welches sich von dem in Fig. 1 gezeigten Schieberegister 30 nur dadurch unterscheidet, dass jede seiner Stufen zusätzlich noch einen bitparallelen Ausgang hat. Mit Hilfe dieses Schieberegisters 46 werden während des Taktes T_7 die zu den empfangenen Informationsbits gehörenden 12 Prüfbits berechnet. Das Syndrom des empfangenen Wortes erhält man durch Addition dieser 12 Prüfbits zu den im Vorregister 41 stehenden 12 Prüfbits des empfangenen Wortes. Die Addition erfolgt bitparallel mit 12 modulo-2-Addierern 60, die an die bitparallelen Ausgänge des Schieberegisters 46 angeschaltet sind.

Während des Taktes T_8 gelangt das so erhaltene 12 Bit-Wort durch 12 UND-Gatter 57 an den Adresseneingang eines Festwertspeichers 45. Unter der damit aufgerufenen Adresse steht die Position des fehlerhaften Informationsbits im empfangenen Wort, das den kleinsten Abstand vom Redundanzteil hat. Da bei dem Fehlerkorrekturverfahren nur Bitfehler im Informationsteil korrigiert werden, genügen zur binären Darstellung der möglichen Bitpositionen eines Bitfehlers 5 Bit, wie schon früher erläutert. Diese 5 Bits stehen am Datenausgang 67 des Festwertspeichers 45 während des Taktes T_8 zur Verfügung. In einem nachfolgenden Umcoder 47 werden diese 5 Bits in ein 24 Bit-Fehlermuster umgewandelt, das nur eine einzige Eins an der durch die 5 Bits adressierten Bitposition aufweist. Dieses Fehlermuster und der Inhalt eines aus zwei gleichen Hälften bestehenden, je 12 Bit langen Sicherstellungsregisters 43 werden über je 12 modulo-2-Addierer 58 und 59 parallel in das Summenregister 42 addiert, dessen Eingang über ein UND-Gatter 50 mit dem Ausgang des Vorregisters verbunden ist. Der zweite Eingang des UND-Gatters 50 ist mit dem Takt T_4 verbunden. Zuvor ergab sich der Inhalt des Sicherstellungsregisters 43 aus dem Informationsteil des empfangenen Wortes, der während des Taktes T_9 über je 12 UND-Gatter 53 und 54 aus dem Summenregister 42 kopiert wurde. Sofern das empfangene Wort ein korrigierbares Fehlermuster enthält, ist der im Summenregister 42 abgespeicherte Informationsteil um den ersten von eventuell meh-

ren Bitfehlern korrigiert worden. Während des folgenden Taktes T_9 wird der Inhalt des Summenregisters 42 wiederum in das Sicherstellungsregister 43 kopiert.

Mit dem korrigierten Informationsteil wird über das Schieberegister 46 nach dem oben beschriebenen Ablauf ein neues Syndrom berechnet, wobei der Umschalter 62 jetzt in der unteren Stellung ist. Hat das Syndrom den Wert Null, so steht am Datenausgang 67 des Festwertspeichers 45 ebenfalls ein Nullwort oder ein Hinweis auf Bitfehler, die sich ausschliesslich im Redundanzteil befinden. Der Ausgang eines NOR-Gatters 61, das am Datenausgang 67 angeschlossen ist, liefert dann eine logische «1», die die Ausgänge von je 12 UND-Gattern 55 und 56 freigibt, die mit ihren jeweils anderen Eingängen mit den Ausgängen des Sicherstellungsregisters 43 verbunden sind und die die bereits wieder in das Sicherstellungsregister 43 kopierte Nutzinformation am Ausgang 68 zur Verfügung stellen. Die weitere Verarbeitung der Nutzinformation ist nicht dargestellt.

Hat das Syndrom nicht den Wert Null und weist es ausserdem auf weitere Bitfehler im Informationsteil hin, so wird über den umgeschalteten Umschalter 62 mit dem korrigierten Informationsteil der oben beschriebene Vorgang noch einmal wiederholt.

Mit einem Zähler 63 wird die Anzahl der Fehler im empfangenen Wort über die Anzahl der Korrekturzyklen erfasst, der Takt T_9 löst die Zählung aus und stellt den Zählimpuls für den Zähler 63 dar. Über einen Ausgang 66 kann der Zählerstand gleichzeitig mit der korrigierten Nutzinformation ausgegeben werden.

Diese Zählschaltung kann mit einem durch einen Schalter 64 einstellbaren Vergleicher 65 erweitert werden. Wenn das empfangene Wort einen Fehler enthält, der mehr Korrekturzyklen erfordert als mit dem Schalter 64 eingestellt sind, $- \text{Zahl der Korrekturzyklen} = \text{Schalterstellung (Fehlerzahl)} + 1$ - wird die Korrektur abgebrochen und über ein UND-Gatter 69 ein Signal für Decoderversagen an einen weiteren Ausgang 70 ausgegeben. Die Korrekturfähigkeit des Decoders kann damit so eingestellt werden, dass kein oder maximal ein, zwei oder drei Fehler im informationstragenden Teil des empfangenen Codewortes korrigiert werden.

Wird die eingestellte Fehlerzahl erreicht, so gibt der Vergleicher 65 über ein ODER-Gatter 71 an den Taktgenerator 39 ein Rücksetzsignal R, welches diesen in seinen Ausgangszustand zurückstellt. Auch wenn der Decoder schon mit weniger Fehlern als eingestellt ein Wort korrigiert hat, wird der Taktgenerator durch das Signal des ODER-Gatters 61 über das ODER-Gatter 71 rückgesetzt. Mit dem dann vom Taktgenerator 39 abgegebenen Rücksetzsignal RS werden ausserdem alle Arbeitsregister, mit Ausnahme vom Sicherstellungsregister 43, gelöscht.

Nach Beendigung des Taktsignals T_4 ist eine erste Taktphase, die Taktphase I beendet, während der die Ablaufsteuerung des Decoders synchron mit dem Empfangsbittakt sein muss. In einer anschliessenden zweiten Taktphase, der Taktphase II kann der Ablauf schneller erfolgen, so dass sich eine Verkürzung der Decodierzeit ergibt. Dies kann beispielsweise durch Erhöhen der Frequenz des Taktgenerators erfolgen.

Bei simplex betriebenen Sende/Empfangsgeräten mit den beschriebenen Codern und Decodern kann das Schieberegister 46 im Decoder auch als Schieberegister 30 im Coder benutzt werden (nicht dargestellt).

Diese Art der Decodierung erfordert eine Speicherkapazität von 4096 Speicherplätzen zu je 5 Bit. Ist der Speicherplatzbedarf eine kritische Grösse, so besteht eine Möglichkeit zur Halbierung der 4096 Speicherplätze.

Zur Halbierung des Speicherplatzbedarfs ist ein relativ geringer Aufwand an Zusatzlogik und ein zusätzlicher Zeit-

aufwand von 12 Schiebetakten pro Korrekturschritt erforderlich. Dafür fällt der Umcoder etwas einfacher aus als der in Fig. 3.

Mit dem Takt T_8 wurde bisher das Syndrom als Adresse für den Festwertspeicher 45 freigegeben. Der Takt T_8 gibt nun das Syndrom frei zur bitparallelen Übernahme in ein rückgekoppeltes Schieberegister. In diesem Schieberegister wird die am weitesten rechts stehende Bitposition auf das Erscheinen einer binären Eins überprüft. In diesem Schieberegister wird nun das Syndrom solange zyklisch verschoben, bis nach spätestens elf Schritten in der überwachten Bitposition eine binäre Eins steht, oder es wird nach zwölf Schritten abgebrochen, falls das Syndrom Null ist. Ist das Syndrom nicht Null, so geben die elf rechtsbündig im Schieberegister stehenden Bits die Adresse eines Festwertspeichers an, in dem Null oder mit jeweils fünf Bits gekennzeichnet, die Position eines Bitfehlers im 24 Bit umfassenden Informationsteil steht. Die fünf Bits für die fehlerhafte Bitposition werden so zugeordnet, dass z.B. beim Wert Eins des fünften Bits der Fehler in der 1. Hälfte des Informationsteils liegt und beim Wert Null des fünften Bits der Fehler in der zweiten Hälfte des Informationsteils liegt (oder umgekehrt).

Im Umcoder werden die vier restlichen Bits in ein 12-Bit-Wort umgewandelt und dieses in das Schieberegister eingeschrieben. Im Schieberegister werden die zu zwölf fehlenden zyklischen Vertauschungsschritte durchgeführt. Die Differenz der zu zwölf fehlenden Vertauschungsschritte ist bei der Verschiebung des Syndroms und der Überwachung der am weitesten rechts stehenden Bitposition des Schieberegisters von einem Zähler erfasst worden.

Nach dieser Anzahl von zyklischen Vertauschungsschritten steht im Schieberegister das Fehlerwort, wobei das fünfte Bit aus dem Festwertspeicher angibt, ob dieses Fehlerwort zur ersten oder zweiten Hälfte der Nutzinformation gehört. Durch bitparallele modulo-2-Addition wird anschliessend die Korrektur der Nutzinformation vorgenommen. Die restlichen Schritte verlaufen so, wie es bereits beschrieben wurde.

Das Blockschaltbild eines Ausführungsbeispiels für die Reduzierung des Speicherplatzbedarfes auf 2047 Speicherplätze zu je 5 Bit zeigt die Fig. 5. Es kann ersatzweise in Fig. 3 an der gestrichelt umrahmten Stelle eingefügt werden.

Zunächst wird das Syndrom in das 12zellige Schieberegister 31 bitparallel übernommen. Der Ausgang des Schieberegisters wird von der Zusatzlogik 32 auf das Erscheinen einer binären Eins überwacht. Falls der Ausgang des Schieberegisters eine binäre Eins führt, wird von der Zusatzlogik 32 kein Verschiebetakt geliefert. Ist der Ausgang des Schieberegisters Null, so wird der Inhalt des Schieberegisters 31 um jeweils eine Stelle verschoben, bis eine binäre Eins erscheint. Die Anzahl der Stellenverschiebungen des Inhalts des Schieberegisters, bis wieder eine binäre Eins an seinem Ausgang auftritt, liefert in einem auf die Zahl zwölf voreingestellten Zähler in der Zusatzlogik die Differenz der nachher mit dem

Fehlermuster durchzuführenden zyklischen Verschiebeschritte.

Sowie im Verlauf der Syndromverschiebungen eine binäre Eins am Ausgang des Schieberegisters 31 auftritt, wird der Verschiebetakt sowohl für das Schieberegister als auch für den voreingestellten Zähler in der Zusatzlogik weggenommen.

Durch ein Freigabesignal von der Zusatzlogik 32 gelangt jetzt das zyklisch verschobene Syndrom mit den ersten elf rechtsbündig stehenden Bits als Adresse auf den Festwertspeicher 34. Die zu diesen elf Bits gehörende Bitposition wird an den Umcoder 33 weitergegeben, der daraus ein 12-Bit-Wort als zugehöriges Fehlermuster bildet. Dieses Fehlermuster wird in einen Zwischenspeicher 35 übernommen. Gleichzeitig setzt die fünfte Bitposition aus dem Festwertspeicher 34 ein RS-Flipflop 36.

Umschalter U_1 bis U_{12} verbinden die Eingänge des Schieberegisters 31 entweder mit den Ausgängen der modulo-2-Addierer zur Übernahme des Syndroms oder mit den Ausgängen des Zwischenspeichers 35 zur Übernahme des Fehlermusters. Nach der Übernahme des Fehlermusters in den Zwischenspeicher 35 werden die Umschalter U_1 bis U_{12} durch ein Steuersignal aus der Zusatzlogik 32 umgeschaltet.

Dadurch gelangt das Fehlermuster bitparallel in das Schieberegister 31. Sowie dieser Vorgang abgeschlossen ist, wird von der Zusatzlogik 32 erneut der Verschiebetakt an das Schieberegister 31 und den Zähler angelegt. Der Inhalt des Schieberegisters 31 muss solange verschoben werden, bis der Zähler die Differenz Null anzeigt.

Umschalter V_1 bis V_{12} verbinden die Ausgänge des Schieberegisters 31 entweder mit den Eingängen des Festwertspeichers 34 zur Eingabe der Adresse oder mit den Eingängen von zweimal 12 UND-Gattern 37 bzw. 38 zur Ausgabe des Fehlermusters. Die jeweils zweiten Eingänge der UND-Gatter 37 und 38 sind mit dem Ausgang des RS-Flipflops 36 verbunden. Jetzt werden die Umschalter V_1 bis V_{12} durch ein Steuersignal von der Zusatzlogik 32 umgeschaltet und das Fehlermuster zur Ausgabe aus dem Schieberegister 31 freigegeben. Das 12-Bit-Fehlermuster gelangt nun, gesteuert von dem RS-Flipflop 36 und den UND-Gattern 37 bzw. 38 auf die erste oder zweite Hälfte des 24-Bit-Ausgangs und somit an die Eingänge der modulo-2-Addierer 58, 59 und steht somit zur Korrektur der Nutzinformation wie schon beschrieben zur Verfügung.

An den Ausgang des Festwertspeichers 34 ist ein NAND-Gatter 72 angeschaltet, das bei der Ausgabe des Nullwortes eine logische Eins zum Abbruch der Korrekturschritte an die UND-Gatter 55, 56, 69 und das ODER-Gatter 71 gibt.

Die erwähnte Zusatzlogik 32 ist eine einfache Hilfsschaltung zur Steuerung der geschilderten Vorgänge wie Zählen, Taktdurchschaltung und Ableitung der Steuersignale zur Umschaltung der Umschalter U und V , so dass keine weitere Erläuterung notwendig erscheint.

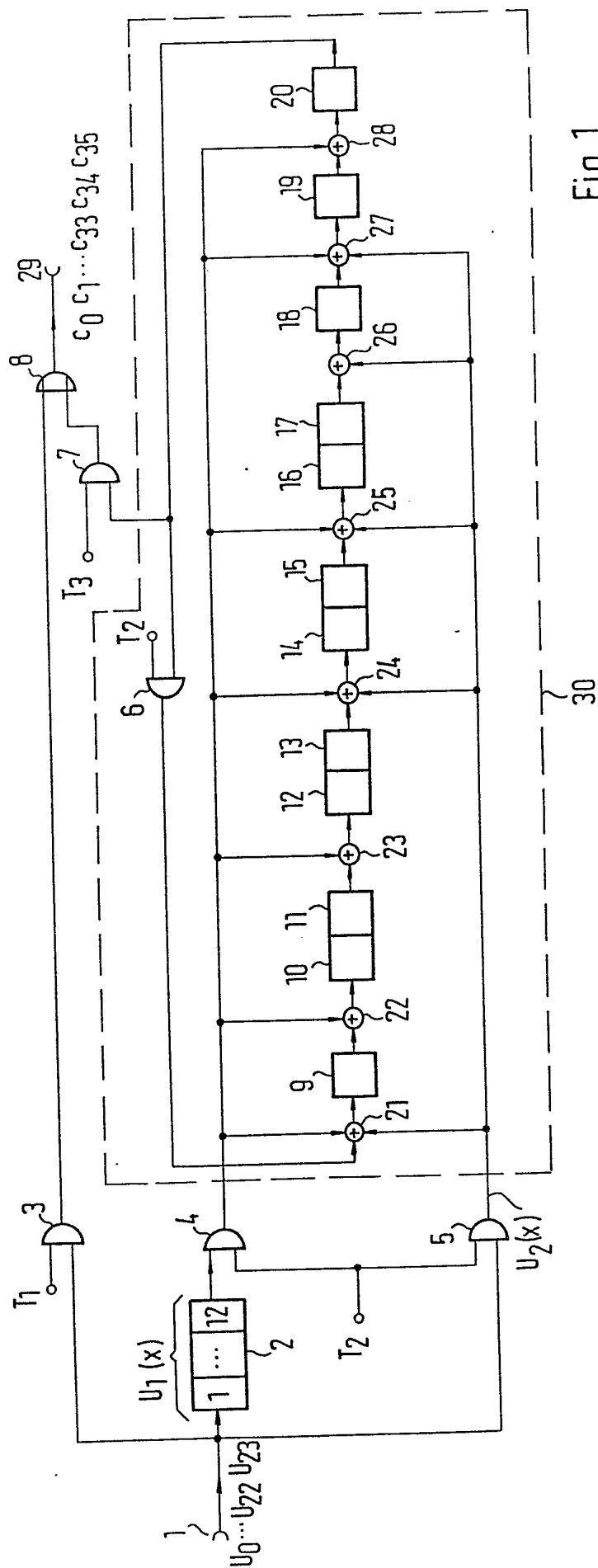


Fig. 1

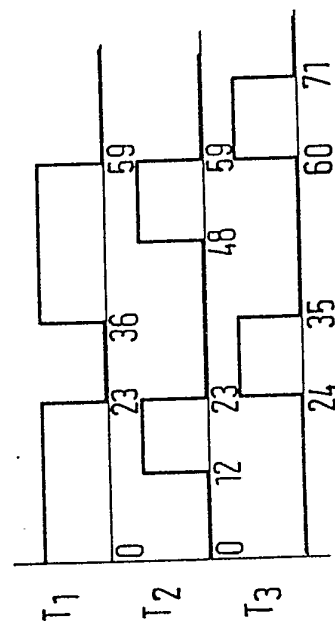
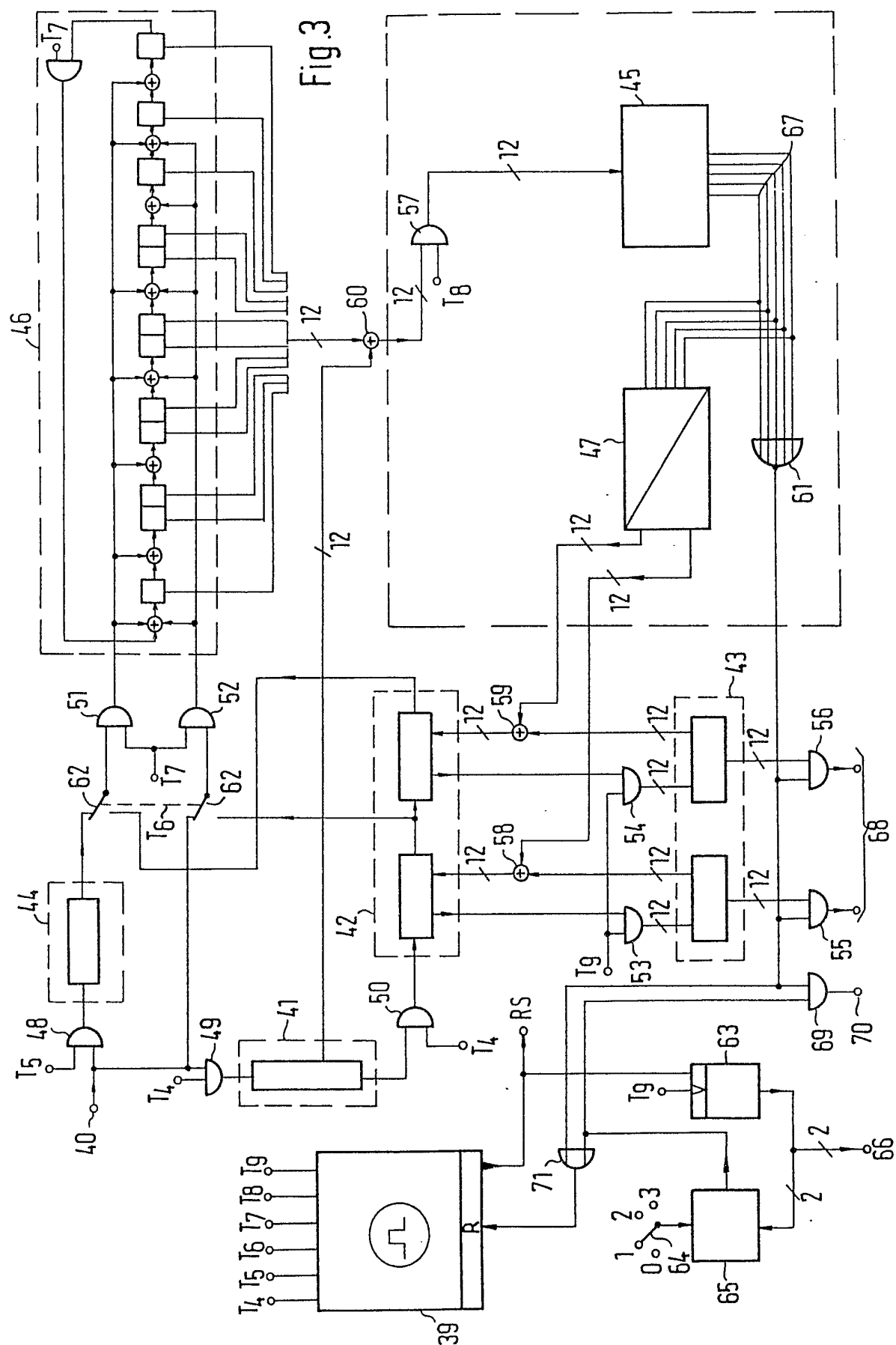


Fig. 2



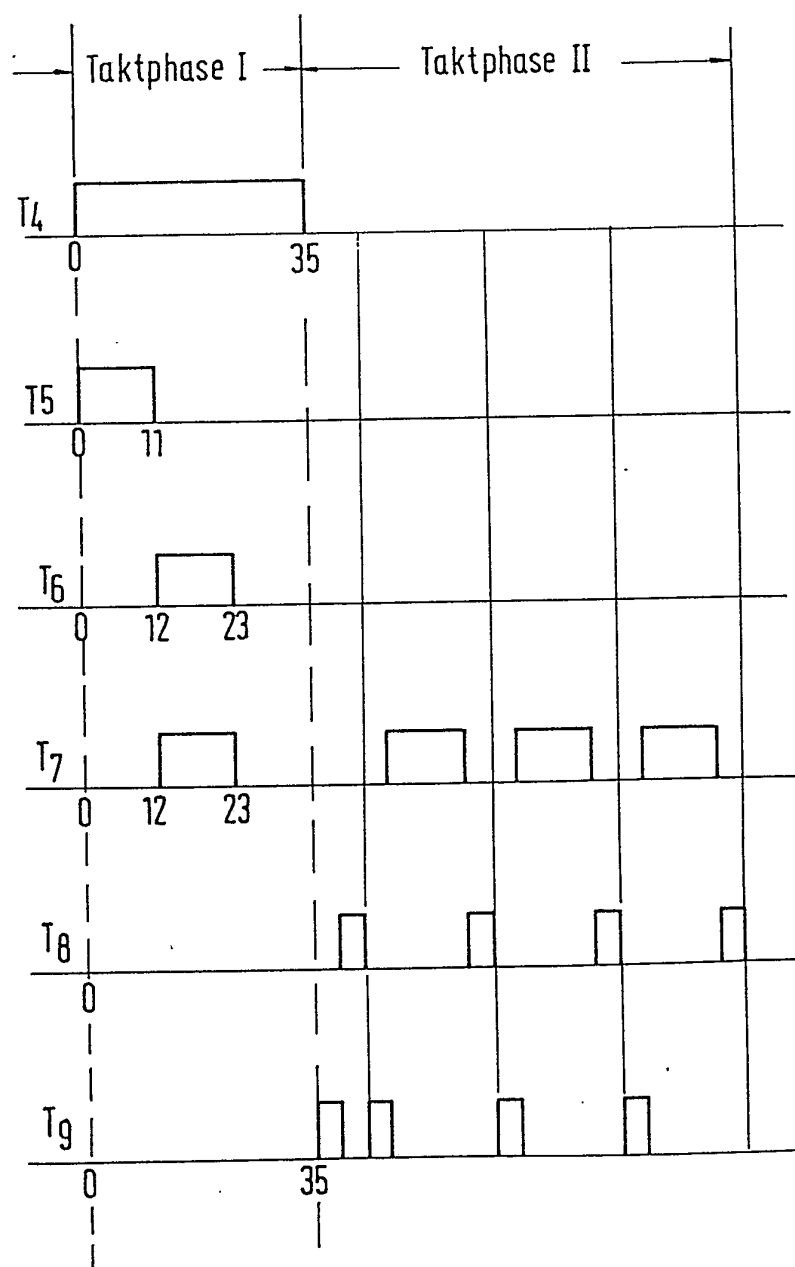


Fig.4

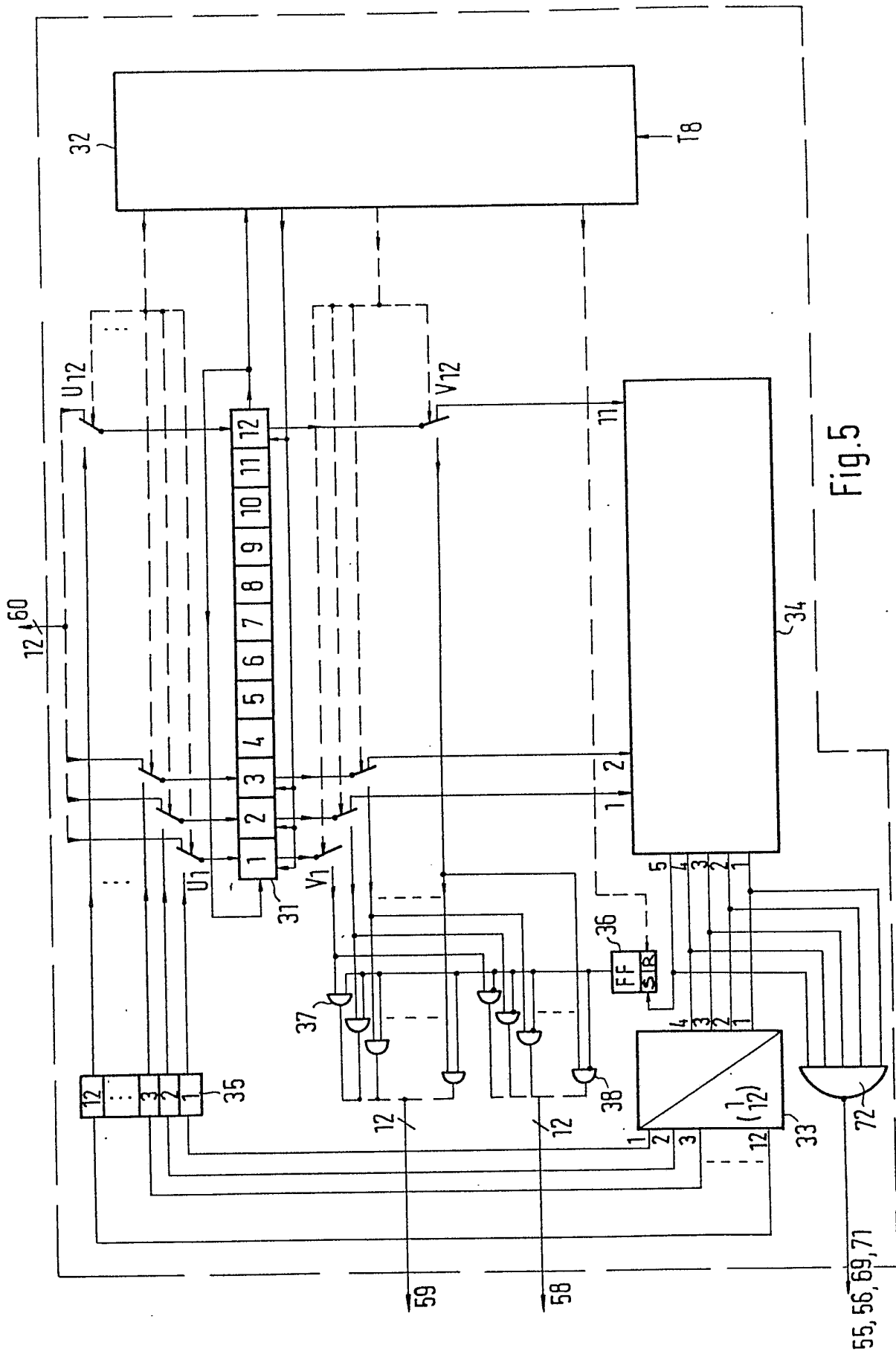


Fig. 5