

# Binäre Gruppencodes zur Sicherung logischer Schaltkreise gegen Fehler

*The use of binary group codes for error protection of logic circuits*

von J. SWOBODA

Institut für Nachrichtenvermittlung  
und Datenverarbeitung der  
Technischen Hochschule Stuttgart

Elektron. Rechenanl. 7 (1965), H. 2, S. 85—90  
Manuskripteingang: 22. 9. 1964

*Durch das angegebene Verfahren lassen sich zweiwertige logische Schaltkreise sowohl ohne Speicher als auch solche mit Speicher gegen Fehler schützen.*

*Der Grundgedanke ist, eine Anzahl von  $m$  Booleschen Variablen durch eine redundante Schaltung um  $k$  Boolesche Variable zu erweitern in der Weise, daß die  $m + k$  Variablen ein Codewort eines linearen Gruppencodes bilden. Die hochentwickelte Theorie der Gruppencodes gestattet es, für vorgegebene Fehler-sicherungseigenschaften einen geeigneten Gruppencode anzugeben, aus dem die redundante Logik zur Sicherung zwangsläufig folgt.*

*The method described in this paper provides error protection for two-value logic circuits with and without storage. The idea is to increase a number  $m$  of Boolean variables by a number  $k$  of such variables by means of redundant circuitry. This is done in such a way that the resulting  $m + k$  variables form a code word of a linear group code.*

*The well developed theory of group codes permits to indicate a suitable group code for predetermined error protection characteristics. From this group code the redundant logic for error protection is determined.*

## 1. Einleitung

In logischen Schaltkreisen können durch Ausfall von Bauelementen oder durch andere Störeinflüsse Fehler auftreten. Es ist bekannt, daß durch zusätzliche Schutzrichtungen fehlerhaftes Arbeiten signalisiert werden kann. Jedoch ist es bei logischen Schaltkreisen nicht immer einfach, eine wirksame Fehlererkennung in das System einzuplanen. Ferner ist das Verfahren bekannt, von einer als

fehlerhaft angezeigten Baugruppe auf eine gleiche Reservebaugruppe automatisch umzuschalten. Diese Dopplung einer Schaltung einschließlich der automatischen Umschaltung zur Fehlerbeseitigung ist nicht immer wirtschaftlich. Im folgenden wird ein allgemeines Verfahren gegeben, von einem ungeschützten System logischer Schaltkreise zu einem geschützten System zu gelangen. Es ist möglich, für die Fehleranzeige mit wachsendem Aufwand beliebig hohe Sicherheit zu erreichen. Ferner läßt sich mit dem Verfahren die Sicherung auch als Fehlerkorrektur planen, bei der zwar die Fehleranzeige für die Wartung vorsorglich gegeben werden kann, jedoch augenblicklich das falsche Ergebnis des Systems korrigiert wird, ohne daß das defekte Bauelement ausgetauscht werden muß.

Es werden zwei Systeme betrachtet:

- a) Ein statischer Zuordner mit  $r$  Booleschen Eingangsvariablen und  $m$  Ausgangsvariablen im ungeschützten Fall. Jede Ausgangsvariable ist eine beliebige Boolesche Funktion der Eingangsvariablen.
- b) Ein logisches Folgeschaltnetzwerk mit  $m$  binären Speicherplätzen im ungeschützten Fall. Es wird angenommen, daß das Netzwerk durch einen Takt weitergeschaltet wird. Jede Datenverarbeitungsanlage mit synchroner Logik oder ein beliebiger Teil derselben läßt sich hier einordnen.

## 2. Sicherung eines binären Zuordners durch Fehleranzeige

### 2.1 Der ungesicherte Zuordner

Der ungesicherte Zuordner besitzt  $r$  Boolesche Eingangsvariable  $x_1, x_2, \dots, x_r$  und  $m$  Ausgangsvariable  $y_1, y_2, \dots, y_m$ , die je eine logische Funktion der Eingangsvariablen sind.

Der Zuordner wird durch diese  $m$ -Funktionen beschrieben:

$$\begin{aligned} y_1 &= f_1(x_1, x_2, \dots, x_r), \\ y_2 &= f_2(x_1, x_2, \dots, x_r), \\ &\dots \\ y_m &= f_m(x_1, x_2, \dots, x_r). \end{aligned} \quad (1)$$

## 2.2 Der gesicherte Zuordner

Das Prinzip der Sicherung eines Zuordners durch Fehleranzeige ist in Bild 1 dargestellt.

Es bedeuten Zuordner I der ungeschützte Zuordner und Zuordner II und III die redundanten Schaltungen zur Fehleranzeige. In Zuordner II werden weitere  $k$ -Ausgangsvariable  $y_{m+1}, y_{m+2}, \dots, y_{m+k}$  erzeugt. Die logischen Funktionen von Zuordner II haben die gleichen Eingangsvariablen  $x_1, x_2, \dots, x_r$  wie der zu schützende Zuordner I. Über die noch offenen logischen Variablen  $y_{m+1}, \dots, y_{m+k}$  von Zuordner II wird derart verfügt, daß  $y_{m+1}, \dots, y_{m+k}$  sich nicht nur aus den gemeinsamen Eingängen  $x_1, \dots, x_r$  bestimmen, sondern daß die Ausgänge von Zuordner II auch aus den Ausgängen von Zuordner I berechnet werden können. Diese zweite Bestimmung der Variablen  $y_{m+1}, \dots, y_{m+k}$  aus den Ausgängen  $y_1, \dots, y_m$  des Zuordners I wird von Zuordner III vorgenommen. Darüber hinaus vergleicht Zuordner III das eigene Ergebnis für  $y_{m+1}, \dots, y_{m+k}$  mit dem Ergebnis von Zuordner II und gibt im Falle einer Unstimmigkeit Fehlersignal ( $s = L$ ). Ein Fehler wird selbstverständlich nicht nur bemerkt, wenn er in dem zu sichernden Zuordner I auftritt, sondern auch in den der Sicherung dienenden Zuordnern II und III.

Als Bildungsvorschrift für die redundanten Ausgänge  $y_{m+1}, \dots, y_{m+k}$  sind viele Möglichkeiten denkbar. In dieser Arbeit wird vorgeschlagen, die logischen Funktionen in Zuordner II so festzulegen, daß die Ausgänge  $y_1, \dots, y_m$  von Zuordner I mit den redundanten Ausgängen  $y_{m+1}, \dots, y_{m+k}$  von Zuordner II zusammen ein Codewort eines binären Gruppencode bilden. Das hat den Vorteil, daß die hochentwickelten und wirksamen Methoden der Fehlersicherung zur binären Datenübertragung [1], [2], [3], [4] unmittelbar auch auf die Sicherung zweiwertiger logischer Schaltkreise angewendet werden können.

Diese Gruppencodes arbeiten derart, daß jede bestimmte redundante Stelle  $y_{m+i}$  eine andere, feste Auswahl aus den ursprünglichen Stellen  $y_1, \dots, y_m$  auf gerade Anzahl von markierten Stellen (Stelle ist L bzw. 1) ergänzt. Die Bildungsvorschrift für Codeworte wird in der folgenden Ziffer 3 noch genauer dargestellt.

Die Sicherungseigenschaften von binären Gruppencodes sind schon weitgehend bekannt [5], [6], [7]. Es gibt Codes, die a) bis zu einer vorgegebenen Anzahl von  $e$  Fehlern sicher, und eine größere Anzahl von Fehlern mit vorgegebener Wahrscheinlichkeit erkennen lassen,

b) bis zu einer vorgebbaren Zahl von  $e'$  Fehlern sicher korrigieren lassen.

In Ziffer 4 werden aus einer nach Fehlererkennungseigenschaften gewählten Codevorschrift und aus den logischen Funktionen des Zuordners I die redundanten Zuordner II und III berechnet.

## 3. Vorschrift für Codeworte eines Gruppencode

Ein Codewort möge bestehen aus  $m$  sog. Nachrichtenstellen  $y_1, y_2, \dots, y_m$ , über deren Wahl frei verfügt werden kann, und aus weiteren  $k$  sog. Prüfstellen  $y_{m+1}, y_{m+2}, \dots, y_{m+k}$ , die aus den  $m$  Nachrichtenstellen berechnet werden. Die Berechnungsvorschrift für die  $k$  Prüfstellen ist ein lineares Gleichungssystem:

$$\begin{bmatrix} y_{m+1} \\ y_{m+2} \\ \dots \\ y_{m+k} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{k1} & a_{k2} & \dots & a_{km} \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \quad (\text{mod. } 2) \quad (2)$$

Alle Terme, die  $y_i$  und auch die  $a_{ij}$ , haben entweder den Wert 0 oder 1. Ferner muß die sogenannte modulo 2 Arithmetik angewendet werden, d.h., es ist definiert:

$$\begin{aligned} 0 + 0 &= 0 & 0 \cdot 0 &= 0 \\ 0 + 1 &= 1 & 0 \cdot 1 &= 0 \\ 1 + 0 &= 1 & 1 \cdot 0 &= 0 \\ 1 + 1 &= 0 & + 1 &= -1 & 1 \cdot 1 &= 1 \end{aligned} \quad (3)$$

Die Matrix mit den Gliedern  $a_{ij}$  kann von der Codetheorie für vorgegebene Fehlersicherungseigenschaften angegeben werden. Dies wird hier vorausgesetzt.

Als ein Beispiel für einen Code ( $m = 4, k = 3$ ), der die Erkennung von bis zu  $e = 2$  Fehlern erlaubt, wird die folgende Bildungsvorschrift angegeben:

$$\begin{bmatrix} y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} \quad (\text{mod. } 2) \quad (4)$$

Die erste Zeile des Gleichungssystems lautet

$$y_5 = y_1 + y_2 + y_3 \quad (\text{mod. } 2) \quad (5)$$

d.h.,  $y_5$  ergänzt  $y_1$  und  $y_2$  und  $y_3$  auf gerade Anzahl von Variablen mit dem Wert 1.

Man bemerke, daß die „Addition modulo 2“ mit der Booleschen Operation der Antivalenz ( $\oplus$ ) übereinstimmt, und die „Multiplikation mod. 2“ mit der Booleschen Operation der Konjunktion ( $\&$ ) übereinstimmt.

Damit läßt sich das System Gl. (4) folgendermaßen schreiben:

$$\begin{aligned} y_5 &= (1 \& y_1 \oplus 1 \& y_2 \oplus 1 \& y_3 \oplus 0 \& y_4) \\ y_6 &= (1 \& y_1 \oplus 1 \& y_2 \oplus 0 \& y_3 \oplus 1 \& y_4) \\ y_7 &= (1 \& y_1 \oplus 0 \& y_2 \oplus 1 \& y_3 \oplus 1 \& y_4) \end{aligned} \quad (6)$$

In die Form von Gl. (6) läßt sich jede Vorschrift für einen Gruppencode bringen. Diese Form wird benutzt, um jene Boolesche Funktionen zu bestimmen, die in Zuordner II und Zuordner III zu verdrahten sind.

## 4. Berechnung der redundanten Zuordner

Im folgenden werden die Schaltfunktionen der redundanten Zuordner II und III von Bild 1 ermittelt.

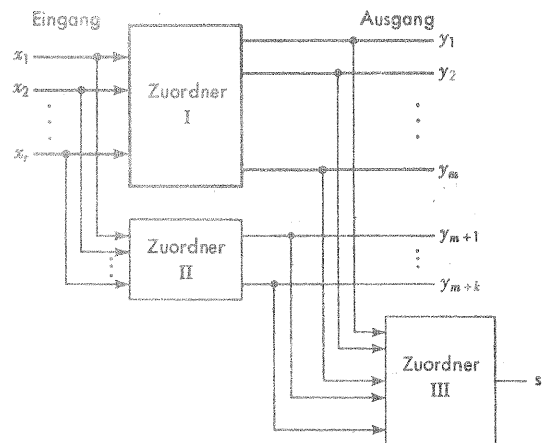


Bild 1. Blockschaubild für die Sicherung eines Zuordners I. Zuordner II und III sind der redundante Teil der Schaltung. Aus den Eingängen des Zuordners I erzeugt Zuordner II  $k$  Prüfvariable. Zuordner III erzeugt ebenfalls  $k$  Prüfvariable, jedoch aus den Ausgängen von Zuordner I, und vergleicht seine Prüfvariablen mit den Prüfvariablen aus Zuordner II. Wenn kein Fehler vorhanden ist, so stimmen die Prüfvariablen, die aus den Eingängen von Zuordner I gewonnen werden, überein mit den Prüfvariablen, die aus den Ausgängen von Zuordner I gewonnen werden.

#### 4.1 Zuordner II

Die gesuchten Funktionen von Zuordner II sind von der Form

$$\begin{aligned} y_{m+1} &= f_{m+1}(x_1, x_2, \dots, x_r), \\ y_{m+2} &= f_{m+2}(x_1, x_2, \dots, x_r), \\ &\dots \\ y_{m+k} &= f_{m+k}(x_1, x_2, \dots, x_r). \end{aligned} \quad (7)$$

Es sind gegeben die logischen Funktionen  $f_1, f_2, \dots, f_m$  des ungeschützten Zuordners gemäß Gl. (1). Ferner sei bekannt die Matrix  $[a_{ij}]$  nach Gl. (2) des Gruppencode  $(m, k)$  aufgrund vorgeschriebener Fehlersicherungseigenschaften. Setzt man Gl. (7) und Gl. (1) in Gl. (2) ein, so ergibt sich in der Schreibweise von Gl. (6) das Ergebnis:

$$\begin{aligned} f_{m+1}(x_1 \dots x_r) &= (a_{11} \& f_1) \bar{\&} (a_{12} \& f_2) \bar{\&} \dots \bar{\&} (a_{1m} \& f_m), \\ &\dots \\ f_{m+k}(x_1 \dots x_r) &= (a_{k1} \& f_1) \bar{\&} (a_{k2} \& f_2) \bar{\&} \dots \bar{\&} (a_{km} \& f_m). \end{aligned} \quad (8)$$

Gl. (8) bedeutet, daß die einzelnen Funktionen des redundanten Zuordners II gebildet werden durch eine Auswahl von antivalent verknüpften Funktionen des zu schützenden Zuordners I. Die Auswahl wird durch die Matrixelemente  $a_{ij}$  des Gruppencode geliefert. Dem Benutzer des Verfahrens bleibt es noch überlassen, die in Gl. (8) erhaltenen Funktionen  $f_{m+1} \dots f_{m+k}$  mit den bekannten Verfahren der Schaltalgebra [8], [9], [12] umzuformen, um je nach Schaltkreistechnik eine billige Schaltung zu erhalten.

#### 4.2 Zuordner III

Die Ausgänge  $y_1 \dots y_m$  von Zuordner I und die Ausgänge  $y_{m+1} \dots y_{m+k}$  von Zuordner II bilden die Eingänge von Zuordner III. Da alle diese Variablen zusammen ein Codewort eines Gruppencode bilden, besteht zwischen ihnen der Zusammenhang nach Gl. (2):

$$y_{m+i} = (a_{i1} \& y_1) \bar{\&} (a_{i2} \& y_2) \bar{\&} \dots \bar{\&} (a_{im} \& y_m), \quad i = 1, 2, \dots, k.$$

Ob der Zusammenhang für den Ausgang  $y_{m+i}$  erfüllt ist, sei durch  $s_i$  ausgedrückt.

Es ist  $s_i = 0$ , wenn für  $y_{m+i}$  der Zusammenhang erfüllt ist, bzw.  $s_i = L$ , wenn dafür der Zusammenhang nicht erfüllt ist. Damit ergibt sich

$$s_i = y_{m+i} \bar{\&} (a_{i1} \& y_1) \bar{\&} (a_{i2} \& y_2) \bar{\&} \dots \bar{\&} (a_{im} \& y_m), \quad i = 1, 2, \dots, k. \quad (9)$$

Wenn mindestens einer von den  $k$  Zusammenhängen nicht erfüllt ist, also mindestens eine der Variablen  $s_i = L$ , so ist ein erkennbarer Fehler aufgetreten. Dies soll durch das Fehlersignal  $s$  angezeigt werden. Es ist  $s = L$  für den Fall irgendeines erkennbaren Fehlers. Das Fehlersignal  $s$  ergibt sich durch disjunktive Verknüpfung.

$$s = s_1 \vee s_2 \vee \dots \vee s_k. \quad (10)$$

Die Gleichungen (9) und (10) geben die Schaltung für Zuordner III an. Die Koeffizienten  $a_{ij}$  in Gl. (9) sind durch den gewählten Gruppencode gegeben.

#### 4.3 Fehlerkorrektur mit Hilfe eines Zuordners IV

Das Prinzip der Fehlerkorrektur soll nur angedeutet werden: Die bisher beschriebene redundante Schaltung bei Fehlererkennung ist auch bei Fehlerkorrektur erforderlich. Darüber hinaus wird ein weiterer Zuordner IV notwendig, der aus den  $k$  Vergleichsergebnissen des Zuordners III ableitet, welche der Ausgänge  $y_1, y_2, \dots, y_m, y_{m+1}, \dots, y_{m+k}$  gefälscht sind. Dieser Zuordner IV läßt sich ebenfalls aus der Codevorschrift ableiten. Seine Ausgangssignale können dann die Korrektur der gestörten Ausgänge mittels Antivalenzverknüpfung vornehmen.

### 5. Sicherung eines Folgeschaltnetzwerks

#### 5.1 Das ungesicherte Folgeschaltnetzwerk

Ein Zuordner von der Art, wie in Ziffer 2.1 beschrieben, läßt sich leicht zu einem Folgeschaltnetzwerk erweitern.

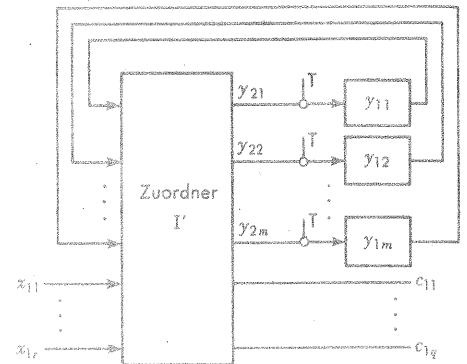


Bild 2a. Das ungeschützte Folgeschaltnetzwerk mit Speichern und Rückkopplungen.

Das Netzwerk wird synchron betrieben, und T ist der Fortschalttakt (gezeichnet für Taktperiode  $t = 1$ ).

Der Zuordner I' in Bild 2a hat  $m + q$  Ausgänge. Die ersten  $m$  Ausgänge führen auf  $m$  binäre Speicherplätze, die  $q$  weiteren Ausgänge geben Ergebnisse des Folgeschaltnetzwerks nach außen ab. Wesentlich ist, daß die  $m$  Ausgänge der Speicherplätze auf den Eingang des Zuordners zurückgekoppelt werden können. Die restlichen  $r$  Eingänge des Zuordners sind die Eingänge des Folgeschaltnetzwerks. Der Zustand des Netzwerks soll sich taktweise ändern. Die doppelt indizierten Booleschen Variablen geben mit dem ersten Index die Taktperiode an, mit dem zweiten Index die Ordnungsnummer der Leitung. Die Speicherinhalte  $y_{11}, y_{12}, \dots, y_{1m}$  und die Eingänge  $x_{11}, x_{12}, \dots, x_{1r}$  zur 1. Taktperiode werden logisch verknüpft, und geben am Ausgang von Zuordner I' die Speicherinhalte  $y_{21}, y_{22}, \dots, y_{2m}$  der 2. Taktperiode an. Durch einen Taktimpuls T werden die alten Speicherinhalte durch die neuen Inhalte  $y_{21} \dots y_{2m}$  ersetzt. Der Übergang zur 3. und jeder folgenden Taktperiode geschieht in gleicher Weise. Von den  $x$ -Eingängen wird vorausgesetzt, daß sie sich ebenfalls nur zu den Taktzeitpunkten ändern. Die Funktion eines beliebigen datenverarbeitenden Systems mit synchroner Logik läßt sich nach Bild 2a darstellen [10].

Das System von Bild 2a wird beschrieben durch  $m + q$  logische Funktionen. Die ersten  $m$  Funktionen geben die Speicherzustände für die  $(t + 1)$ -te Taktperiode abhängig von den Größen zur Taktperiode  $t$  an, die weiteren  $q$  Funktionen beschreiben die Ausgänge des Systems.

$$\begin{aligned} y_{t+1,1} &= F_1(y_{t,1}, y_{t,2}, \dots, y_{t,m}; x_{t,1}, x_{t,2}, \dots, x_{t,r}), \\ y_{t+1,2} &= F_2(y_{t,1}, y_{t,2}, \dots, y_{t,m}; x_{t,1}, x_{t,2}, \dots, x_{t,r}), \\ y_{t+1,m} &= F_m(y_{t,1}, y_{t,2}, \dots, y_{t,m}; x_{t,1}, x_{t,2}, \dots, x_{t,r}). \end{aligned} \quad (11)$$

$$\begin{aligned} c_{t,1} &= c_1(y_{t,1}, y_{t,2}, \dots, y_{t,m}; x_{t,1}, x_{t,2}, \dots, x_{t,r}), \\ c_{t,q} &= c_q(y_{t,1}, y_{t,2}, \dots, y_{t,m}; x_{t,1}, x_{t,2}, \dots, x_{t,r}). \end{aligned} \quad (12)$$

#### 5.2 Das gesicherte Folgeschaltnetzwerk

Es erweist sich als zweckmäßig, das ungesicherte Netzwerk nach Bild 2a umzuzeichnen in die Form von Bild 2b.

Die  $q$  Ausgänge  $c_{11} \dots c_{1q}$  werden von dem rückgekoppelten Netzwerk abgespalten. Der Zuordner 0, der die  $q$  Ausgänge aus den Speicherinhalten  $y_{11} \dots y_{1m}$  und den Eingängen  $x_{11} \dots x_{1r}$  ableitet, kann nach dem Verfahren von Ziffer 2 und 4 geschützt werden. Die folgende Sicherung des Folgeschaltnetzwerks beschränkt sich auf den rückgekoppelten Teil mit den Speichern und dem Zuordner I, also auf den oberen Teil in Bild 2b.

Die Sicherung des Folgeschaltnetzwerks ist der Sicherung des Zuordners von Ziffer 2.2 analog.

Die  $m$  Ausgänge des Zuordners I (Bild 3) für das ungeschützte Folgeschaltnetzwerk werden wieder um  $k$  Ausgänge vermehrt mittels eines redundanten Zuordners II, und zwar in der Weise, daß alle  $m + k$  Ausgänge ein Code-

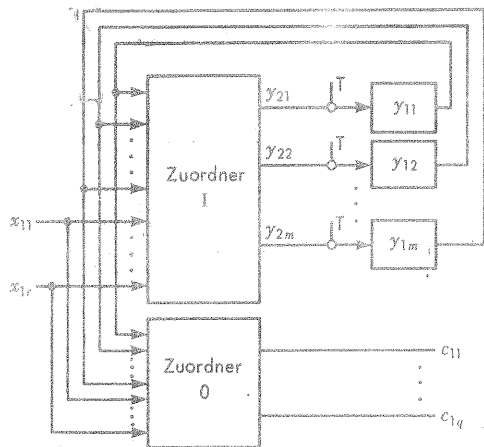


Bild 2b. Das Folgeschaltnetzwerk von Bild 2a in umgezeichneter Form.

Die Ausgänge \$c\_{11} \dots c\_{1q}\$ wurden über einen getrennten Zuordner 0 herausgeführt.

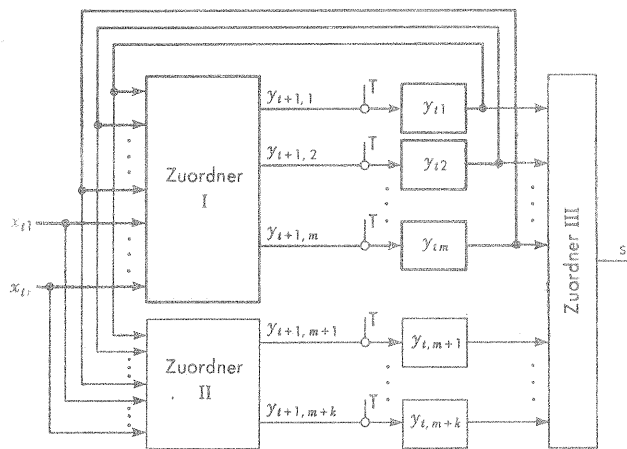


Bild 3. Blockschaltbild des gesicherten Folgeschaltnetzwerks. Der obere, linke Teil der Abbildung entspricht dem Zuordner I einschließlich Speicher in Bild 2b, der untere und rechte Teil des Bildes zeigt den redundanten Teil der Schaltung (Zuordner II mit zugehörigen Speicherplätzen und Zuordner III).

ort eines vorgegebenen Gruppencode bilden. Um auch die Taktschaltungen und die Speicherplätze mit in die Sicherung einzubeziehen, erhalten die Ausgänge von Zuordner II auch Speicherplätze. Der Zuordner III überwacht dann, ob alle Speicherinhalte zusammen ein Codewort im Sinne von Ziffer 3 bilden. Dazu müssen die Speicher zerstörungsfrei abfragbar sein.

Die Speicherinhalte \$y\_{t1} \dots y\_{tm}\$ des gesicherten Netzwerks stimmen mit jenen des ungesicherten Netzwerks von Bild 2a, b unter gleichen Nebenbedingungen zu jeder Taktperiode \$t\$ überein. Die redundanten Speicherinhalte \$y\_{t,m+1} \dots y\_{t,m+k}\$ werden nämlich nur dem Zuordner III zugeführt, nicht aber dem Zuordner I.

### 5.3 Berechnung des gesicherten Folgeschaltnetzwerks

Die Schaltung des Zuordners III, welche nachprüft, ob \$y\_{t1} \dots y\_{t,m+k}\$ das Gleichungssystem des gewählten Gruppencode erfüllt, ergibt sich wie in Ziffer 4 mit Hilfe von Gl. (9) und (10). Der Zuordner III hängt nur ab von den Matrixelementen \$a\_{ij}\$, welche den gewählten Gruppencode definieren.

Für die Berechnung von Zuordner II kann ebenfalls wie in Ziffer 4 verfahren werden: Schreibt man die Schaltfunktionen von Zuordner II in der Form

$$y_{t+1,m+1} = F_{m+1}(y_{t1}, y_{t2}, \dots, y_{tm}; x_{t1}, \dots, x_{tr}),$$

$$y_{t+1,m+k} = F_{m+k}(y_{t1}, y_{t2}, \dots, y_{tm}; x_{t1}, \dots, x_{tr}), \quad (13)$$

so erhält man diese Schaltfunktionen \$F\_{m+1} \dots F\_{m+k}\$, indem man Gl. (13) und Gl. (11) in Gl. (2) einsetzt. Es ergibt sich in der Schreibweise von Gl. (6) das Gleichungssystem (14). Es stimmt mit Gl. (8) überein, nur steht jetzt sinngemäß statt klein \$f\_i\$ groß \$F\_i\$.

$$F_{m+1}(y_{t1} \dots x_{tr}) = (a_{11} \& F_1) \& (a_{12} \& F_2) \& \dots \& (a_{1m} \& F_m), \quad (14)$$

$$\dots$$

$$F_{m+k}(y_{t1} \dots x_{tr}) = (a_{k1} \& F_1) \& (a_{k2} \& F_2) \& \dots \& (a_{km} \& F_m).$$

Es sind \$a\_{ij}\$ die Matrixelemente des gewählten Gruppencode und \$F\_1 \dots F\_m\$ die Schaltfunktionen der zu schützenden Schaltung. \$F\_{m+1} \dots F\_{m+k}\$ beschreiben den Zuordner II. Wenn die gesamte Schaltung in Bild 3 ohne Fehler arbeitet, so bilden die \$m+k\$ Speicherinhalte ein Codewort des gewählten Gruppencode. Die Überwachung erfolgt durch das Ausgangssignal \$s\$ des bereits berechneten Zuordners III.

Die Darstellungen des Folgeschaltnetzwerks setzen bisher voraus, daß der Fortschalttakt \$T\$ immerwährend am System anliegt und nicht abgeschaltet werden kann. Üblicherweise wird man fordern, daß das Netzwerk auch einen Zustand „Stop“ einnehmen kann, währenddessen die Speicherinhalte zeitinvariant sind. Diese Bedingung läßt sich mit immerwährendem Takt nur dadurch erreichen, daß die Ausgänge der Zuordner im Zustand „Stop“ mit den Speicherinhalten übereinstimmen.

$$y_{t+1,i} = y_{t,i}, \quad i = 1, 2, \dots, m+k.$$

Die Realisierung dieser Bedingung kann u. U. eine unnötig komplizierte Schaltung erfordern.

In ausgeführten Schaltungen verursacht eine Stopbedingung üblicherweise, daß der Takt an zentraler Stelle gesperrt wird. Eine folgende umgeformte Darstellung für die Berechnung von Zuordner II ist in solchen Fällen leichter zu handhaben.

Der Zustand „Betrieb“ oder „Stop“ des Netzwerks werde durch eine Boolesche Variable \$B\$ ausgedrückt.

Es sei \$B = L\$ für den Zustand Betrieb, und \$B = 0\$ für den Zustand Stop.

Selbstverständlich ist die Variable \$B\$ abhängig von den Speicherinhalten \$y\_{t1} \dots y\_{tm}\$ und von den Eingängen \$x\_{t1} \dots x\_{tr}\$ des Netzwerks. (Eine Start-Stop-Taste z. B. gehört auch zu diesen Eingängen.)

Damit lassen sich die Schaltfunktionen \$F\_1 \dots F\_m, F\_{m+1} \dots F\_{m+k}\$ von Zuordner I und II wie folgt umschreiben.

$$y_{t+1,i} = F_i(y_{t1} \dots x_{tr}) = (B \& y_{ti}) \vee (B \& G_i), \quad i = 1, 2, \dots, m+k. \quad (15)$$

Hierbei sind \$F\_i, B\$ und \$G\_i\$ logische Funktionen der Variablen \$y\_{t1}, y\_{t2} \dots y\_{tm}; x\_{t1} \dots x\_{tr}\$. Es stimmen \$G\_i\$ und \$F\_i\$ in ihren Werten überein, solange der Zustand „Betrieb“ gilt. Die Funktionen \$B\$ sowie \$G\_1 \dots G\_m\$ des ungeschützten Netzwerks sind vorgegeben. Zur Beschreibung des Zuordners II genügt die weitere Kenntnis der Funktionen \$G\_{m+1} \dots G\_{m+k}\$. Man erhält diese durch Einsetzen von Gl. (15) in Gl. (14), indem für \$B = L\$ (Betrieb) gesetzt wird. Das Ergebnis stimmt formal mit Gl. (14) bzw. mit Gl. (8) überein.

$$G_{m+1}(y_{t1} \dots x_{tr}) = (a_{11} \& G_1) \& (a_{12} \& G_2) \& \dots \& (a_{1m} \& G_m), \quad (16)$$

$$\dots$$

$$G_{m+k}(y_{t1} \dots x_{tr}) = (a_{k1} \& G_1) \& (a_{k2} \& G_2) \& \dots \& (a_{km} \& G_m).$$

Es ist wichtig festzustellen, daß bei der angeführten Schaltung die Speicherausgänge \$y\_{t1} \dots y\_{tm}\$ und die Eingänge \$x\_{t1} \dots x\_{tr}\$ unmittelbar an den Eingang von Zuordner II führen. Es ist verboten, logische Teilfunktionen in Zuordner I, die in Zuordner II ebenfalls vorkommen, in Zuordner II zu sparen und sie von Zuordner I nach Zuordner II zu führen (Verlust an redundanter Logik!). Wie Bild 3

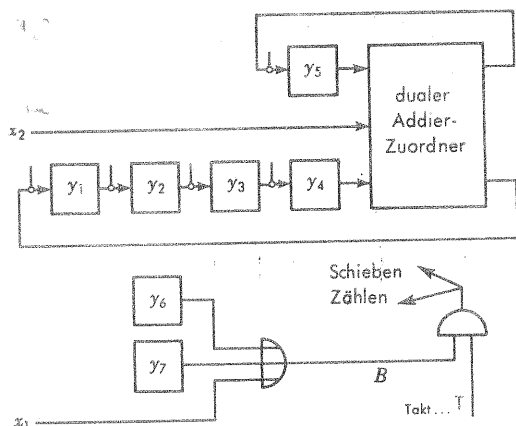


Bild 4. Blockschaltbild eines dualen Serienaddierwerkes einschließlich Steuerung.

- $x_2$  Eingang für den 1. Operand,
- $y_1 \dots y_4$  Speicher für 2. Operand und Ergebnis,
- $y_5$  Übertragungsspeicher,
- $y_6, y_7$  Zähler für Arbeitsschritte des Rechenablaufs,
- $x_1$  Auslösung der Addition,
- T Takt.

Der Index  $t$  ist bei den Variablen  $x_{ti}, y_{ti}$  vereinfachend weggelassen.

zeigt, haben Zuordner I und II nur gemeinsame Eingänge, sonst aber keinerlei Verbindungen. Die Betriebsfunktion  $B = B(y_{t1} \dots y_{tm}; x_{t1} \dots x_{tr})$ , welche über Betrieb bzw. Stop entscheidet, muß daher auch in Zuordner II gesondert verdrahtet werden.

## 6. Beispiel für Folgeschaltnetzwerk: Einfaches Addierwerk

### 6.1 Die ungeschützte Schaltung

Bild 4 zeigt das betrachtete Dual-Addierwerk als Blockschaltbild. Die Kästchen  $y_1 \dots y_7$  stellen Speicherplätze (z.B. bistabile Kippstufen) dar. Der Index  $t$  wird hier vereinfachend weggelassen. Die Funktion ist folgende: Der erste Operand (4 Binärstellen) stehe bereits im Akkumulatorregister, das von  $y_1, y_2, y_3, y_4$  gebildet wird. Der zweite 4stellig gedachte Operand steht in Serie am Eingang  $x_2$  an. Nach Additionsbeginn wird bei jedem Takt die duale Summe von  $x_2$  und  $y_4$  und  $y_5$  nach  $y_1$  eingeschrieben, während gleichzeitig der Operand in  $y_1 \dots y_4$  um eine Stelle nach rechts geschoben wird. In  $y_5$  speichert man den Übertrag, der dann beim nächsten Takt zur nächsten Stelle hinzuaddiert wird. Nach 4 Arbeitstakten steht die Summe im Akkumulatorregister  $y_1 \dots y_4$ . Die Speicherstellen  $y_6$  und  $y_7$  stellen einen Dualzähler dar, der die 4 Arbeitsschritte einer Addition abzählt.

Durch das Signal  $x_1 = L$ , das einen Takt lang anstehen möge, wird der Additionsbeginn ausgelöst. Solange  $y_6$  und  $y_7$  und  $x_1$  je den Wert „Null“ haben, ist die Schaltung im Zustand Stop ( $B = 0$ ). Sobald  $x_1 = L$  ist, wird  $B = L$ . Solange  $B = L$  gilt, werden die Taktimpulse zum Schieben von  $y_1 \dots y_4$  und  $y_5$  verwendet, und von dem Dualzähler  $y_6, y_7$  abgezählt. Währenddessen ist  $B = y_6 \vee y_7 \vee x_1 = L$ . Nach 4 Taktimpulsen stehen  $y_6$  und  $y_7$  wieder beide in „Null“, wodurch  $B = 0$  wird. Weitere Taktimpulse haben keine Wirkung mehr, und der Zustand Stop ist erreicht. Unter Voraussetzung von Gl. (11) und Gl. (15) ist obige Schaltung in ihrer Funktion vollkommen beschrieben durch folgende Gleichungen.

(Es wird hier für die Antivalenz ( $\oplus$ ) Addition modulo 2 „ $\oplus$ “ geschrieben, und für die Konjunktion (&) das Multiplikationszeichen „ $\cdot$ “ verwendet. Hausenblas [11] hat gezeigt, daß sich jede logische Funktion mit diesen beiden Verknüpfungen darstellen läßt. Diese Darstellung ist hier sehr günstig.)

$$\begin{aligned}
 B &= x_1 \vee y_6 \vee y_7, \\
 G_1 &= x_2 \oplus y_4 \oplus y_5, \\
 G_2 &= y_1, \\
 G_3 &= y_2, \\
 G_4 &= y_3, \\
 G_5 &= y_4 \cdot y_5 \oplus y_4 \cdot x_2 \oplus y_5 \cdot x_2, \\
 G_6 &= y_6 \oplus y_7 \oplus 1, \\
 G_7 &= y_6 \oplus y_7.
 \end{aligned} \tag{17}$$

### 6.2 Erste geschützte Schaltung

Als einfachster Gruppencode für den Schutz obiger Schaltung soll die bekannte Quersummenprüfung auf gerade Anzahl von markierten Stellen durch ein Parity-Bit gewählt werden. Der Code erlaubt die Erkennung ungeradzahlig vieler Fehler.

Die Codematrix ist für  $m = 7, (k = 1)$

$$[a_{11} \ a_{12} \ \dots \ a_{17}] = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]. \tag{18}$$

Damit erhält man aus Gl. (16) unmittelbar die Schaltfunktion  $G_8$  für den Zuordner II, der die redundante Stelle  $y_8$  liefert.

$$\begin{aligned}
 G_8 &= G_1 \oplus G_2 \oplus G_3 \oplus G_4 \oplus G_5 \oplus G_6 \oplus G_7, \\
 &= x_2 \oplus y_4 \oplus y_5 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_4 \cdot y_5 \oplus y_4 \cdot x_2 \\
 &\quad \oplus y_5 \cdot x_2 \oplus y_6 \oplus 1 \oplus y_6 \oplus y_7, \\
 &= x_2 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus \bar{y}_7 \oplus x_2 \cdot y_4 \oplus \\
 &\quad \oplus x_2 \cdot y_5 \oplus y_4 \cdot y_5.
 \end{aligned} \tag{19}$$

Die Schaltfunktion für Zuordner III ergibt sich aus Gl. (9) und (10). Zuordner III prüft hier nach, ob die Anzahl aller markierten Speicherplätze (Speicherinhalt ist L) geradzahlig ist.

$$\begin{aligned}
 s_1 &= y_8 \oplus y_1 \oplus y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7, \\
 s &= s_1.
 \end{aligned} \tag{20}$$

Der Mehraufwand der geschützten Schaltung gegenüber der ungeschützten Schaltung ist

- a) eine bistabile Kippstufe mehr gegenüber sieben vorhandenen
- b) zusätzliche logische Verknüpfungen, die durch Gl. (19) und (20) ausgedrückt sind, gegenüber den bereits vorhandenen nach Gl. (17). Ferner muß  $B = x_1 \vee y_6 \vee y_7$  für Zuordner II noch einmal getrennt verdrahtet werden.

Der Mehraufwand hängt wesentlich davon ab, ob die Antivalenzverknüpfung in der gewählten Schaltungstechnik billig ist.

Die Sicherung des beschriebenen Addierwerkes reicht weiter als der übliche Schutz der Operanden je durch ein Parity-Bit, da auch die Steuerung der Operation mit in der Prüfung erfaßt wird.

### 6.3 Zweite geschützte Schaltung

Für die gleiche ungeschützte Schaltung wird ein zweiter Gruppencode zur Berechnung der redundanten Schaltung als Beispiel angegeben, der die sichere Erkennung von 2 Fehlern gestattet und für mehr als 2 Fehler nur etwa  $1/16$  aller Fehlermöglichkeiten nicht erkennt. Die Codematrix ist für  $m = 7, (k = 4)$

$$\begin{bmatrix} a_{11} & \dots & a_{17} \\ \dots & & \\ a_{41} & \dots & a_{47} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \tag{21}$$

Im folgenden wird nur das leicht berechenbare Ergebnis für die redundante Schaltung niedergeschrieben. Funktionen von Zuordner II aus Gl. (16)

$$\begin{aligned}
 G_8 &= x_2 \oplus y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7, \\
 G_9 &= x_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7 \oplus x_2 \cdot y_4 \oplus \\
 &\quad \oplus x_2 \cdot y_5 \oplus y_4 \cdot y_5, \\
 G_{10} &= y_1 \oplus y_3 \oplus y_7, \\
 G_{11} &= y_2 \oplus y_6 \oplus x_2 \cdot y_4 \oplus x_2 \cdot y_5 \oplus y_4 \cdot y_5.
 \end{aligned} \tag{22}$$

Funktionen von Zuordner III aus Gl. (9) und (10)

$$\begin{aligned} s_1 &= y_1 \oplus y_2 \oplus y_3 \oplus y_7 \oplus y_8, \\ s_2 &= y_1 \oplus y_4 \oplus y_5 \oplus y_7 \oplus y_9, \\ s_3 &= y_2 \oplus y_4 \oplus y_6 \oplus y_7 \oplus y_{10}, \\ s_4 &= y_3 \oplus y_5 \oplus y_8 \oplus y_{11}, \\ s &= s_1 \vee s_2 \vee s_3 \vee s_4. \end{aligned} \quad (23)$$

Der Mehraufwand dieser zweiten geschützten Schaltung ist a) vier weitere bistabile Kippschaltungen gegenüber sieben bereits vorhandenen,

b) die logische Verknüpfung  $B = x_1 \vee y_6 \vee y_7$ , sowie die logischen Verknüpfungen von Gl. (22) und (23), die den redundanten Zuordnern II und III entsprechen.

#### 6.4 Sicherheit der geschützten Schaltungen

Es wird angenommen, daß in einem Speicherplatz mit einer Wahrscheinlichkeit  $p$  pro Taktperiode ein Fehler entsteht oder der Speicherplatz falsch gesetzt wird.

Es interessiert die Fehlerwahrscheinlichkeit je Takt  $p_F$  der ungeschützten Schaltung und die Restfehlerwahrscheinlichkeit  $p'_F$  für unbemerkte Fehler in der geschützten Schaltung.

Für das behandelte Beispiel ist

Zahl der Speicherstellen der ungeschützten Schaltung  $m=7$ ,

Zahl der zusätzlichen Speicher in Ziffer 6.2  $k_1 = 1$ ,

1 der zusätzlichen Speicher in Ziffer 6.3  $k_2 = 4$ .

Angenommene (hohe!) Fehlerwahrscheinlichkeit pro Stelle und Taktperiode (bei einem Takt von 100 KHz entspricht dies einem Fehler pro Speicherstelle in jeder Sekunde)  $p = 10^{-5}$ .

Es wird Unabhängigkeit der Fehler in den Speicherstellen angenommen. Es gelten dann die aus der Wahrscheinlichkeitsrechnung bekannten Formeln:

Fehlerwahrscheinlichkeit je Takt für die ungeschützte Schaltung

$$p_F = \sum_{i=1}^m \binom{m}{i} \cdot p^i \cdot (1-p)^{m-i} \approx m \cdot p = 7 \cdot 10^{-5}.$$

Restfehlerwahrscheinlichkeit je Takt der nach Ziffer 6.2 geschützten Schaltung:

$$\begin{aligned} p'_{F1} &= \sum_{i=1}^{1/2 \cdot (m+k_1)} \binom{m+k_1}{2i} \cdot p^{2i} \cdot (1-p)^{m+k_1-2i} \approx \\ &\approx \binom{m+k_1}{2} \cdot p^2 = 2,8 \cdot 10^{-9}. \end{aligned}$$

Restfehlerwahrscheinlichkeit je Takt der nach Ziffer 6.3 geschützten Schaltung:

$$\begin{aligned} p'_{F2} &\approx 2^{-k_2} \cdot \sum_{i=3}^{m+k_2} \binom{m+k_2}{i} \cdot p^i \cdot (1-p)^{m+k_2-i} \approx \\ &\approx \binom{m+k_2}{3} \cdot p^3 \cdot 2^{-k_2} = 1,0 \cdot 10^{-14}. \end{aligned}$$

In diesem Beispiel werden demnach die Fehlerwahrscheinlichkeiten für unbemerkte Fehler um etwa  $10^4$  bei der nach 6.2 gesicherten Schaltung, und um etwa  $10^9$  bei der nach 6.3 gesicherten Schaltung gegenüber der ungesicherten Schaltung verbessert. Die Rechnung mag wegen der als unabhängig angenommenen Fehlerstellen optimistisch sein. Andererseits ist die Annahme einer Fehlerwahrscheinlichkeit von  $p = 10^{-5}$  pro Takt und Stelle sehr pessimistisch zu nennen.

Man sieht, daß stationäre Zuordner wie auch Folgeschaltungen mit logischer Schaltungsredundanz bei erträglichem Aufwand sehr wirksam gegen Fehler gesichert werden können. Die Berechnung der zusätzlichen Zuordner, die als redundante Schaltungsteile die Sicherung leisten, erfolgt mit Hilfe der Theorie der binären Gruppencodes. Damit können die aus der binären Datenübertragung bekannten und bewährten Sicherungsverfahren für die Sicherung von Schaltkreisen übernommen werden.

#### Literatur

- [1] *Hamming, R. W.*, Error detecting and error correcting codes. Bell Syst. Techn. J., April 1950, S. 147—160.
- [2] *Peterson, W. W.*, Error-correcting codes. John Wiley & Sons, Inc., 1962.
- [3] *Peterson, W. W.*, and *Brown, D. T.*, Cyclic codes for error detection. Proceedings IRE, Januar 1961, S. 228—235.
- [4] *Berger, E. R.*, Nachrichtentheorie und Codierung. Taschenbuch der Nachrichtenverarbeitung, herausgegeben von *K. Steinbuch*. Springer 1962, S. 59—100.
- [5] *Marko, H.*, Systemtechnik der Datenübertragung. Nachrichtentechn. Fachberichte 19. Vieweg & Sohn 1961.
- [6] *Berger, E. R.*, Die Wirksamkeit von Blocksicherungsverfahren gegenüber gebündelten Störungen bei der Datenübertragung. Archiv der Elektr. Übertragung 16 (1962), S. 51—55.
- [7] *Nili, H.*, Fehlerwahrscheinlichkeit und Geschwindigkeit bei der Übertragung digitaler Informationen durch Gruppencodes. Archiv der Elektr. Übertragung 18 (1964), 5, S. 282—292.
- [8] *McCluskey jr., E. J.*, Minimization of Boolean functions. Bell Syst. Techn. J., 35 (1956), S. 1417—1444.
- [9] *Weinmiller, J.*, Hilfsmittel zur Vereinfachung von Schaltfunktionen. Elektron. Rechenanl. 3 (1961), 3, S. 123—129.
- [10] *Zemanek, H.*, Logische Algebra und Theorie der Schaltnetze. Taschenbuch der Nachrichtenverarbeitung, herausgegeben von *K. Steinbuch*. Springer 1962, S. 100—160.
- [11] *Hausenblas, A.*, Schaltungssynthese mit Koinzidenz- und Antivalenzgattern. Elektron. Rechenanl. 4 (1962), 5, S. 217—221.
- [12] *Bartee, T. C.*, Computer design of multiple-output logical networks. IRE Transactions on Electronic Computers, Vol. EC-10, 1961, 1, S. 21—30.