

**KERNFORSCHUNGSZENTRUM  
KARLSRUHE**

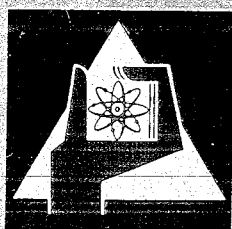
März 1970

KFK 1185

Labor für Elektronik und Meßtechnik

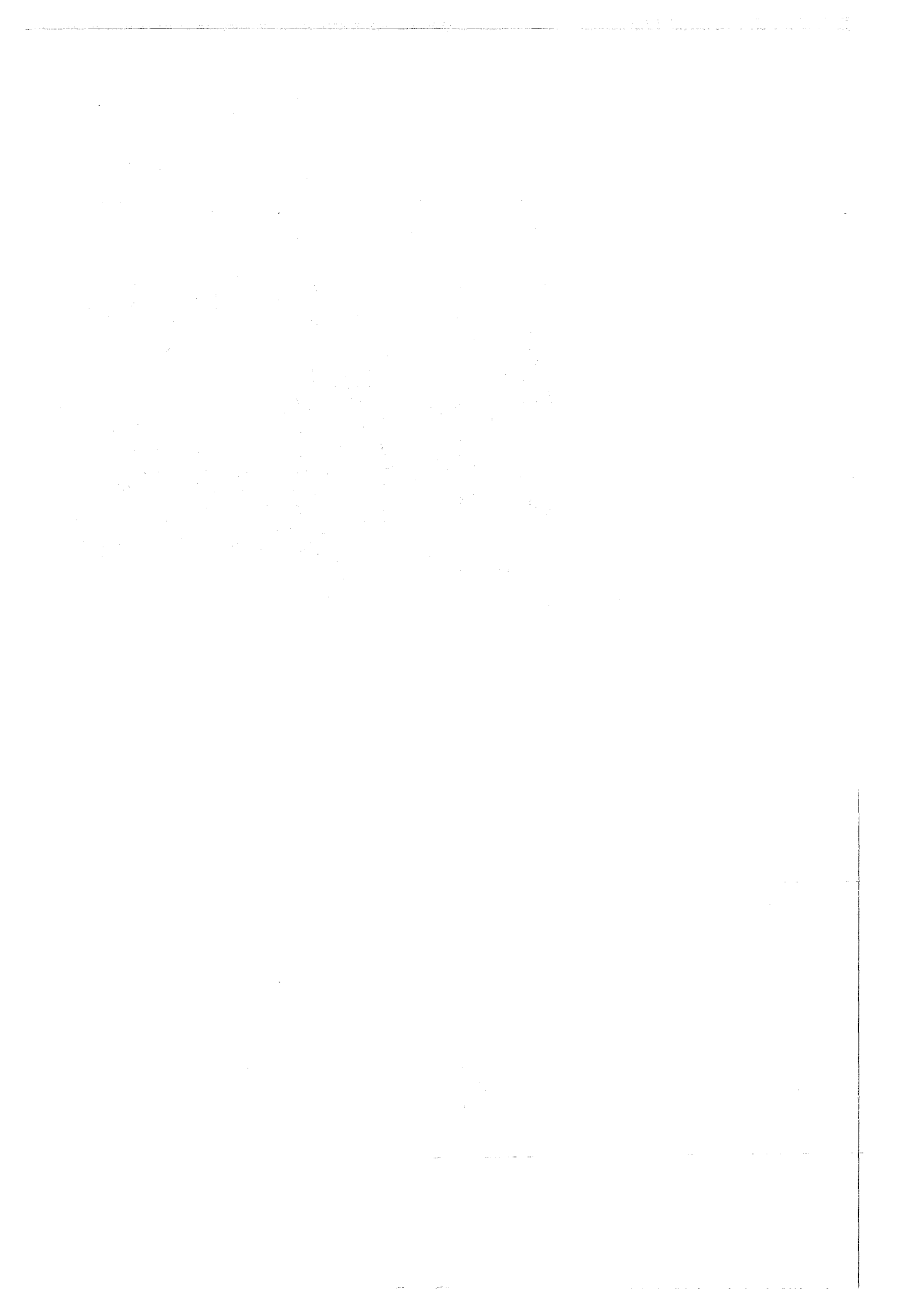
Code-Umsetzer

J. Ottens



GESELLSCHAFT FÜR KERNFORSCHUNG M. B. H.

KARLSRUHE



Dr.-Ing. J. Ottes

## Code-Umsetzer

DK 681.3.053

### 1 Einleitung

In der Digitaltechnik entsteht häufig die Notwendigkeit der Umwandlung einer Information von einer Darstellungsart in eine andere. So zum Beispiel bei der Eingabe von Daten über Lochstreifen mit einem tetradischen Code in einen Rechner, der intern rein nach dem Dualsystem arbeitet. Auch bei der Anzeige der Position von Werkzeugmaschinen mit binär absolut codierten linearen oder rotatorischen Positionsgebern [4] sind Umcodierungen notwendig, um bei Handeinstellung dem Bedienenden eine verständliche Anzeige in dekadischer Schreibweise zu liefern. Bei weitem am häufigsten werden Umrechnungen aus dem Dualsystem in das dezimale benötigt, gelegentlich auch die umgekehrten [3]. In diesem Aufsatz wird der Versuch gemacht, einige der bekannten Methoden und Verfahren zur Durchführung solcher Umsetzungen zu beschreiben, wobei das Hauptgewicht auf solchen Verfahren liegt, die sich nicht auf die Verwendung eines Rechners stützen.

### 2 Mathematische Grundlagen

In sogenannten polyadischen Zahlensystemen läßt sich eine Zahl in der folgenden Form ausdrücken:

$$N = a_n B^n + a_{n-1} B^{n-1} + \dots + a_1 B + a_0 + a_{-1} B^{-1} + \dots \quad (1)$$

Darin bedeutet  $B$  die Basis des Zahlensystems (für das Dezimalsystem ist die Zahl 10 die Basis und für das Dualsystem die Zahl 2). Die positiven Koeffizienten  $a_i$  haben einen Wertevorrat von 0 bis  $B-1$ , beim Dezimalsystem also von 0...9. Die Zahl 1970 würde sich demzufolge so darstellen:

$$N = 1 \cdot 10^3 + 9 \cdot 10^2 + 7 \cdot 10^1 + 0 \cdot 10^0 \quad (1a)$$

Die Gleichung (1) läßt sich in eine Form bringen, die im weiteren Verlaufe der Untersuchung ständig gebraucht wird:

$$N = \{[(a_n \cdot B + a_{n-1}) \cdot B + a_{n-2}] B + \dots + a_1\} B + a_0 \quad (2)$$

Es werden nur ganze Zahlen betrachtet.

Der Autor ist Mitarbeiter der Gesellschaft für Kernforschung mbH in Karlsruhe.

Die Code-Umsetzung ist eine in der Technik der Zähl- und Rechengerte wie auch der Datenverarbeitung immer wiederkehrende Aufgabe. Streng genommen muß man dabei unterscheiden zwischen der Umsetzung von einem Zahlensystem in ein anderes, wobei immer die Zahl als Ganzes betrachtet werden muß, und der einfachen Umcodierung innerhalb eines Zahlensystems, bei der jede Stelle für sich getrennt behandelt werden darf. Von den mathematischen Grundlagen der Zahlensysteme ausgehend, gibt der Autor für diese beiden Aufgaben Gruppen aus der Fülle der Möglichkeiten eine Reihe gängiger Lösungen an und schafft damit einen Überblick über dieses selten zusammenhängend dargestellte Gebiet.

### 3 Umsetzungsmethoden- und -Verfahren

#### 3.1 Umsetzung des Dual- in den 8421-BCD-Code

Zunächst setzt man in Gleichung (2) zur Anpassung an die Aufgabe  $B = 2$ . Die Koeffizienten  $a_i$  haben somit die Werte 0 und 1. Die Eins wird im folgenden immer als L geschrieben, um Verwechslungen mit den entsprechenden Koeffizienten anderer Zahlensysteme zu vermeiden. Die Gleichung (2) lautet dann:

$$N = \{[(a_n \cdot 2 + a_{n-1}) \cdot 2 + a_{n-2}] \cdot 2 + \dots + a_1\} \cdot 2 + a_0 \quad (2a)$$

An einem Zahlenbeispiel kann man dann sofort den Wert der einzelnen Koeffizienten  $a_i$  ablesen, zum Beispiel:

$$N = LLOLOL \quad (3)$$

Es ist:

$$a_0 = 1, a_1 = 0, a_2 = 1, a_3 = 0, a_4 = 1, a_5 = 1$$

In der Form der Gleichung (1a) geschrieben lautet die Gleichung (3):

$$N = 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \quad (3a)$$

Damit ist auch klar, wie man den dezimalen Wert eines Ausdruckes von der Form nach Gleichung (3) bzw. (3a) bekommt: Man hat lediglich die 2er-Potenzen mit den von Null verschiedenen Koeffizienten zu addieren. Mit Papier und Bleistift würde man bei solchen Umwandlungen folgendermaßen vorgehen:

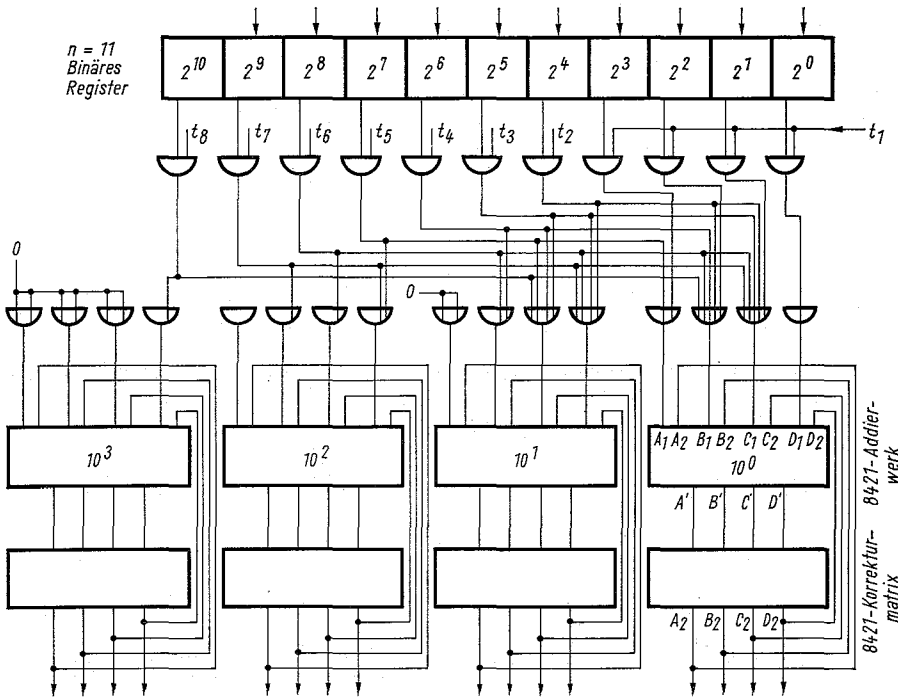
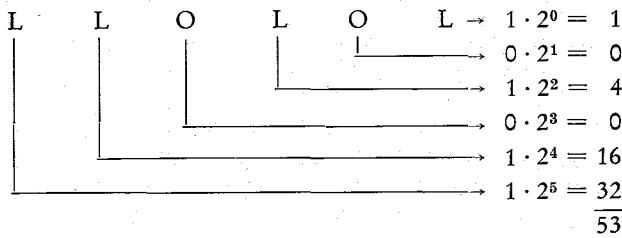


Bild 1. Blockschaltbild für die Umsetzung des Dual-Codes in den 8421-BCD-Code mit Hilfe eines BCD-Addierwerkes



In Anlehnung an Gleichung (2a) läßt sich noch eine andere Methode angeben, bei der man nicht erst immer die Potenzen von 2 auszurechnen braucht. Sie wird auch mit „Doubling and Dabbling“ bezeichnet.

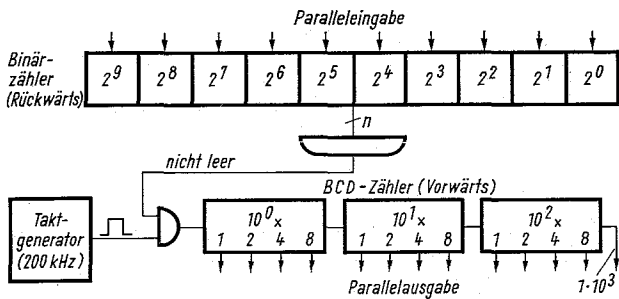
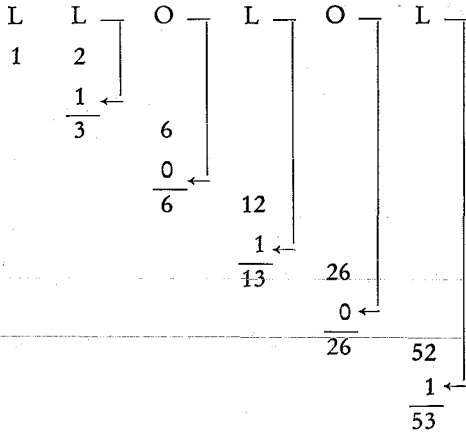


Bild 2. Blockschaltbild für die Umsetzung des Dual-Codes in den 8421-BCD-Code mit zwei Zählern

### 3.1.1 Umsetzung mit einem Rechner

Hier wird die Umsetzung mit einem Unterprogramm [1] erledigt. Das Programm könnte z. B. so aufgebaut sein, daß zuerst geprüft wird, wie oft eine bestimmte Zehnerpotenz — beginnend mit der höchstmöglichen — in der umzusetzenden Zahl enthalten ist. Der Rest wird dann mit der nächst niedrigeren Potenz von Zehn genauso weiterbehandelt. Dabei entstehen für jede Potenz BCD-Darstellungen der Koeffizienten. Die Potenzen müssen natürlich in binärer Codierung vorliegen. Bei dem schon mehrfach benutzten Zahlenbeispiel würde das so aussehen:

$$\begin{array}{r}
 LLOLOL : LOLO = LOL \\
 \text{(Koeffizient von } 10^1) \\
 \hline
 LOLO \\
 OOLL \\
 \hline
 LOLO \\
 OOLL \\
 \hline
 \text{Rest: LL} \\
 \text{(Koeffizient von } 10^0)
 \end{array}$$

Man erhält also in BCD-Darstellung:

$$\begin{array}{r}
 OLOL / OOLL \\
 5 \quad 3
 \end{array}$$

### 3.1.2 Umsetzung mit einem festverdrahteten Addierwerk

In einem BCD-Addierwerk könnte die Umwandlung nach Bild 1 in  $n$  Schritten stattfinden, wobei  $n$  die Anzahl der Bits der binären Zahl ist. Man wird nacheinander die Partialsummen der einzelnen Bits in dem Addierwerk bilden, beginnend mit dem höchstwertigen Bit  $2^n$ .

Bei Verwendung eines BCD-Volladdierers sind zur Beseitigung der sogenannten Pseudo-Tetraden Korrekturglieder nötig. Hierauf wird später noch sehr ausführlich eingegangen.

### 3.1.3 Umsetzung mit zwei Zählern

Bei dieser einfachen, aber sehr zeitraubenden Methode [2] benutzt man eine Anordnung entsprechend Bild 2. Die umzusetzende Binärzahl wird in einen binären Rückwärts-

Tabelle 1. Wahrheitstabelle für Korrekturmatrik

A	B	C	D	E	Ü	A'	B'	C'	D'	Dez.
16	8	4	2	1		8	4	2	1	0
0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	L	0	0	0	0	L	2
0	0	0	L	0	0	0	0	0	L	3
0	0	L	0	0	0	0	L	0	0	4
0	0	L	0	L	0	0	L	0	L	5
0	0	L	L	0	0	0	L	L	0	6
0	0	L	L	L	0	0	L	L	L	7
0	L	0	0	0	0	L	0	0	0	8
0	L	0	0	L	0	L	0	0	0	9
0	L	0	L	0	L	0	0	0	0	10
0	L	L	0	0	L	0	0	0	0	11
0	L	L	0	L	0	L	0	0	L	12
0	L	L	L	0	L	0	L	0	0	13
0	L	L	L	L	0	L	0	L	0	14
L	0	0	0	0	L	0	L	0	L	15
L	0	0	0	L	L	0	L	L	L	16
L	0	0	L	0	L	L	0	0	0	17
L	0	0	L	L	0	L	L	0	0	18
L	0	0	L	L	L	L	L	0	L	19

zähler gegeben. Dieser wird anschließend durch Impulse von einem Taktgenerator leergezählt. Dieselben Impulse werden gleichzeitig in einen BCD-Vorwärtszähler gegeben. Nach Räumung des Rückwärtszählers steht die Zahl im BCD-Code im Vorwärtszähler. Weitere Impulse des freilaufenden Generators werden in dem UND-Gatter vor dem Aufwärtszähler abgesperrt. Der größte Nachteil dieser Anordnung ist der hohe Zeitaufwand. Bei einer Zahl mit einer Kapazität von 24 bit und einer Taktfrequenz von 200 kHz dauert die Umcodierung länger als 83 s. Das ist für viele Anwendungen unzulässig.

### 3.1.4 Umsetzung mit einer Kombination von Schieberegister und Korrektur-Matrizen (Patent angemeldet)

Bei dieser Methode der Umcodierung wird der in Gleichung (2a) festgelegte Algorithmus (siehe auch das Beispiel für Doubling and Dabbling) direkt verwirklicht. Wesentliche Bestandteile der Einrichtung sind ein Schieberegister und Additions-Korrektur-Matrizen (Bild 3). Wie schon im Abschnitt 3.1.2 erwähnt, werden diese Matrizen bei BCD-Addierwerken benötigt, um Pseudo-Tetraden zu beseitigen.

Bei der Herleitung der Korrektur-Matrix kann man von der Wahrheitstabelle ausgehen (Tabelle 1). Daraus lassen sich die logischen Gleichungen (4 bis 8) für die Ausgänge  $D'$ ,  $C'$ ,  $B'$ ,  $A'$  und  $U'$  der Matrix ablesen. Es gilt:

$$D' = E \quad (4)$$

$$C' = \bar{A}\bar{B}C\bar{D}\bar{E} + \bar{A}\bar{B}CDE + \bar{A}\bar{B}C\bar{D}E + \bar{A}\bar{B}CDE + \bar{A}BC\bar{D}\bar{E} + \bar{A}BCDE + \bar{A}BC\bar{D}E + \bar{A}BCDE \quad (5)$$

$$B' = \bar{A}\bar{B}C\bar{D}\bar{E} + \bar{A}\bar{B}CDE + \bar{A}\bar{B}C\bar{D}E + \bar{A}\bar{B}CDE + \bar{A}BC\bar{D}\bar{E} + \bar{A}BCDE + \bar{A}BC\bar{D}E + \bar{A}BCDE \quad (6)$$

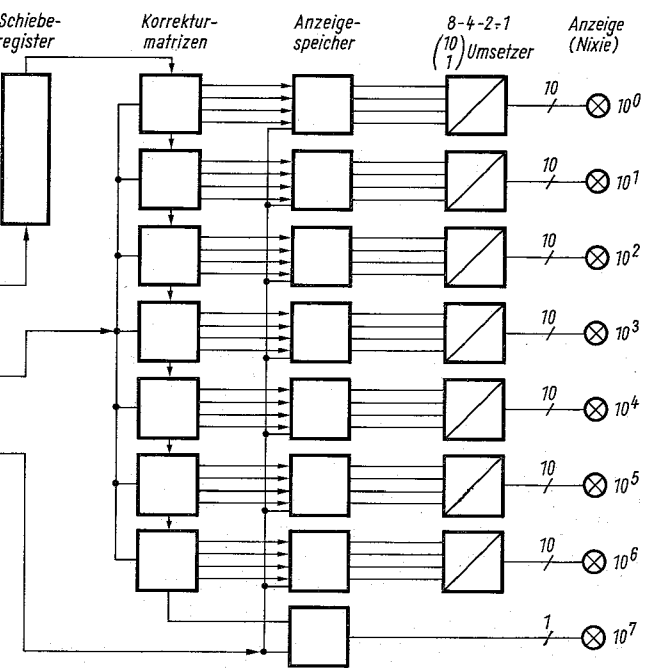


Bild 3. Blockschaltbild für die Umsetzung des Dual-Codes in den 8421-BCD-Code mit einem Schieberegister und Korrekturmatrizen

$$A' = \bar{A}\bar{B}\bar{C}\bar{D}\bar{E} + \bar{A}\bar{B}\bar{C}\bar{D}E + \bar{A}\bar{B}\bar{C}DE + \bar{A}\bar{B}\bar{C}DE \quad (7)$$

$$U' = \bar{A}\bar{B}\bar{C}\bar{D}\bar{E} + \bar{A}\bar{B}\bar{C}\bar{D}E + \bar{A}\bar{B}\bar{C}DE + \bar{A}\bar{B}\bar{C}DE + \bar{A}BC\bar{D}\bar{E} + \bar{A}BCDE + \bar{A}BC\bar{D}E + \bar{A}BCDE + \bar{A}\bar{B}C\bar{D}\bar{E} + \bar{A}\bar{B}CDE \quad (8)$$

Diese Gleichungen lassen sich nach verschiedenen Methoden (McCLUSKEY, KARNAUGH), stark vereinfachen und man erhält dann die Ausdrücke (4a) bis (8a):

$$D' = E \quad (4a)$$

$$C' = \bar{A}\bar{B}D + \bar{A}BC\bar{D} + \bar{A}\bar{B}C\bar{D} \quad (5a)$$

$$B' = \bar{A}\bar{B}C + \bar{A}BCD + \bar{A}\bar{B}C\bar{D} \quad (6a)$$

$$A' = \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} \quad (7a)$$

$$U' = \bar{A}\bar{B}D + \bar{A}\bar{B}C + \bar{A}\bar{B}C \quad (8a)$$

Auch diese Ausdrücke sind noch nicht die Minimalform. Diese ist gegeben durch:

$$D' = E \quad (4b)$$

$$C' = \bar{A}\bar{B}D + BC\bar{D} + A\bar{D} \quad (5b)$$

$$B' = A\bar{D} + \bar{A}CD + \bar{A}\bar{B}C \quad (6b)$$

$$A' = B\bar{C}\bar{D} + AD \quad (7b)$$

$$U' = A + BC + BD \quad (8b)$$

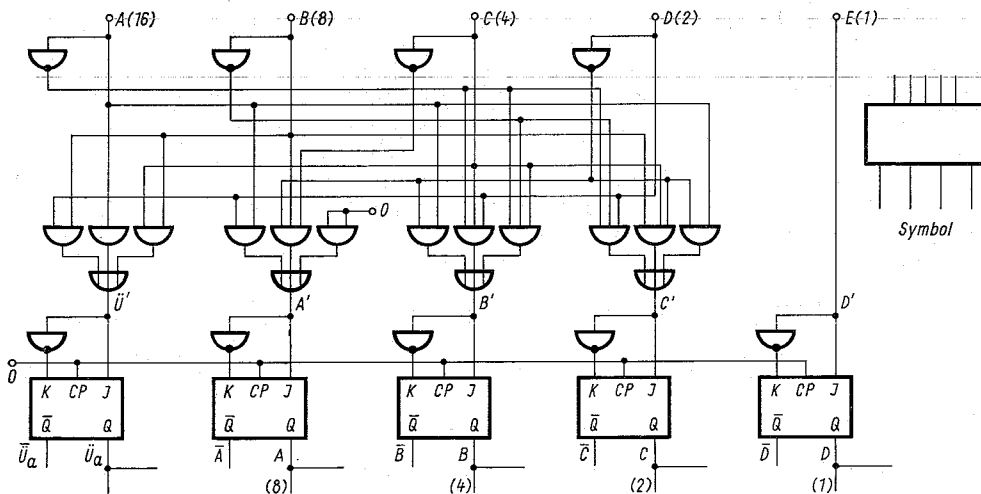


Bild 4. Aufbau und Symbol einer Korrekturmatrix

Der Algorithmus nach Gleichung (2a) verlangt abwechselnd die Multiplikation des schon umgesetzten Anteiles der binären Zahl mit dem Faktor 2 und dann die Addition

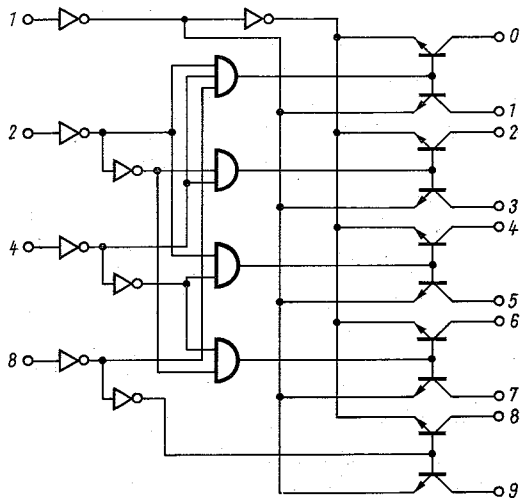


Bild 9. Schaltung zur Umcodierung des 8421-BCD-Codes in den 1-aus-10-Code

Damit sind aber die Unterschiede zwischen binär codierten Oktalzahlen und rein binären Zahlen verschwunden und die weiteren Darlegungen unter Punkt 3.1 sind vollinhaltlich hier anwendbar.

### 3.3 Dezimal-Dual-Umsetzung

In einer reinen  $\binom{10}{1}$ -Darstellung wird man Dezimalzahlen in der Digitaltechnik selten finden, meistens werden sie in einer binär verschlüsselten Form vorliegen (BCD). Jedenfalls sind solche Formen sehr leicht durch Code-Umsetzer zu erzeugen. Von einer solchen Darstellung soll ausgegangen werden: Es ist:

$$N = \{ [(a_n \cdot 10 + a_{n-1}) \cdot 10 + a_{n-2}] \cdot 10 + \dots + a_1 \} \cdot 10 + a_0 \quad (2c)$$

#### 3.3.1 Umsetzung mit einem Rechner

Die Dezimalzahl soll bereits im 8-4-2-1-Code vorliegen. Die weiteren Schritte werden nach Anweisung von Gleichung (2c) von einem Unterprogramm erledigt, wobei natürlich die Multiplikationen mit der Basis  $B = 10$  in der Darstellung LOLO erfolgen müssen. Der Koeffizient der höchsten Stelle wird mit LOLO multipliziert und dann der Koeffizient der nächsten Stelle addiert usw. Nach Addition von  $a_0$  hat man die Zahl in binärer Form.

#### 3.3.2 Umsetzung mit einem Addierwerk

Nach Bild 8 werden hierzu ein rein binäres Addierwerk, ein BCD-Speicher sowie eine entsprechende Anzahl von

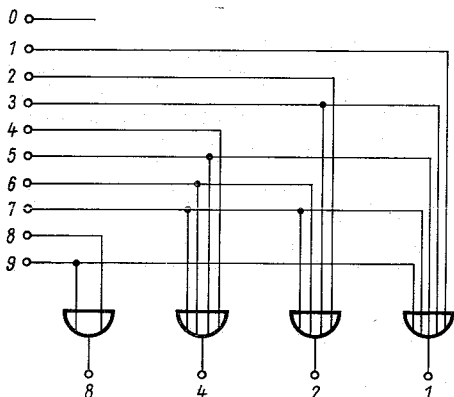


Bild 10. Schaltung zur Umcodierung des 1-aus-10-Codes in den 8421-BCD-Code

Steuerimpulsen benötigt. Alle Stellen liefern nacheinander Bit für Bit ihre Beiträge zu den im Addierwerk auflaufenden binären Partialsummen. Bei dieser Anordnung wächst der Bedarf an Zeit nur proportional mit der Stellenzahl. Der Schaltungsaufwand wächst zwar überproportional, da es sich aber im wesentlichen um Dioden handelt, erscheint er noch gut tragbar. In den niedrigsten drei Dekaden kommt man mit denselben Steuerimpulsen  $t_1$  bis  $t_4$  aus, weil es nicht möglich ist, daß zwei Dioden gleichzeitig auf denselben Eingang des binären Addierwerkes arbeiten, was natürlich zu Fehlern Anlaß geben würde. Dieses Verfahren wird z. B. bei AEG-Werkzeugmaschinen-Steuerungen mit Programmeingabe über Lochstreifen angewendet.

## 4 Umcodierung im gleichen Zahlensystem

Die bisher gegebene Zusammenstellung ist nicht annähernd vollständig, zu jeder Fragestellung wurden nur einige Lösungen gegeben, obwohl noch sehr viele andere denkbar sind. Dem Verfasser kam es darauf an zu zeigen, wie man solche Umsetzungen durchführen kann, denen ein Wechsel des Zahlensystems zugrunde liegt. Findet kein solcher Wechsel statt, sondern wird nur eine andere Codierung innerhalb ein- und desselben Zahlensystems benötigt, dann ist die Aufgabe leichter. Hierzu einige Beispiele:

### 4.1 Umcodierung von 8421-BCD- in den 1-aus-10-Code

Diese Aufgabe kann mit dem Baustein SN 5441/7441 von Texas Instruments durchgeführt werden. Bild 9 zeigt die entsprechende Schaltung; sie erfüllt die Bedingungen der nachstehenden Wahrheitstabelle.

D	C	B	A	0	1	2	3	4	5	6	7	8	9
0	0	0	0	X	0	0	0	0	0	0	0	0	0
0	0	0	L		X								
0	0	L	0			X							
0	0	L	L				X						
0	L	0	0					X					
0	L	0	L						X				
0	L	L	0							X			
0	L	L	L								X		
L	0	0	0									X	
L	0	0	L										X

### 4.2 Umcodierung vom 1-aus-10- in den 8421-BCD-Code

Man verwendet dazu die im vorigen Abschnitt gezeigte Wahrheitstabelle. Es läßt sich daraus entnehmen, daß insgesamt nur vier Gatter benötigt werden, wie in Bild 10 dargestellt.

In ganz ähnlicher Weise werden andere Codes behandelt, wie z. B. der Aiken-Code und der 3-Exzeß-Code. Diese Umcodierungen sind so einfach, daß man in fast jeder auf dem Markt erhältlichen Schaltungsfamilie die gängigsten Codierer und Decodierer als fertige Bausteine beziehen kann.

## Literatur

- [1] WEHRIG, H.: Informationsdarstellung und -verarbeitung in EDV-Anlagen. ELEKTRONIK 1968, H. 6, S. 165...168.
- [2] LANGE, W. R.: Entwurf eines seriellen BCD-Binär-Codierers unter Anwendung des Karnaugh-Diagrammes. Frequenz 23 (1969), H. 2, S. 50...u. f.
- [3] KLEIN, P. E.: Zahlensysteme und Codierung. ELEKTRONIK 1967, H. 3, S. 65...70.
- [4] RAIBLE, B.: Binär codierte Maßstäbe. ELEKTRONIK 1967, H. 3, S. 71...74.