

RECHENZENTRUM TH MÜNCHEN
ARBEITSGRUPPE BETRIEBSYSTEME

INTERNSCHRIFT Nr. 3

THEMA:

Deutsche Normen für Begriffe der Informationsverarbeitung
DIN 44300

VERFASSER:

DATUM:

31.8.1969

FORM DER ABFASSUNG

SACHLICHE VERBINDLICHKEIT

ENTWURF

ALLGEMEINE INFORMATION

AUSARBEITUNG

DISKUSSIONSGRUNDLAGE

ENDFORM

ERARBEITETER VORSCHLAG

VERBINDLICHE MITTEILUNG

VERALTET

ÄNDERUNGZUSTAND

BEZUG AUF BISHERIGE INTERNSCHRIFTEN

Vorkenntnisse aus:

Erweiterung von:

Ersatz für:

BEZUG AUF KÜNTIGE INTERNSCHRIFTEN

Vorkenntnisse zu:

Erweiterung in:

Ersetzt durch:

ANDERWEITIGE LITERATUR

Informationsverarbeitung

Begriffe

DIN

44 300

Information processing, vocabulary

Einsprüche bis 31. Januar 1969

Dieser Norm-Entwurf wird der Öffentlichkeit zur Stellungnahme vorgelegt. Er enthält die vorgesehene Neufassung der Norm DIN 44 300, Ausgabe April 1965. Die genannte Ausgabe wird hierdurch nicht ungültig.

Zurückgezogen wird der Norm-Entwurf DIN 44 300 Blatt 10 vom Juni 1965, da sein Inhalt in den vorliegenden Norm-Entwurf eingearbeitet ist.

Da sich der Inhalt noch in wesentlichen Teilen ändern kann, bitten wir, sich noch nicht auf die Arbeit nach diesem Norm-Entwurf einzustellen, sondern die endgültige Fassung des Normblattes abzuwarten. Einsprüche und Änderungsvorschläge (möglichst zweifach) zu diesem Norm-Entwurf werden erbeten an den Fachnormenausschuß Informationsverarbeitung (FNI) im DNA, 1 Berlin 30, Burggrafenstraße 4-7.

Deutscher Normenausschuß

Inhalt

	Seite		Seite
Nr 1 bis 23 Datendarstellung	1	Nr 119 bis 129 Konstruktiver Aufbau digitaler Rechenanlagen	12
Nr 24 bis 32 Zahlendarstellung	4	Nr 130 bis 147 Speicher	13
Nr 33 bis 43 Programmiersprache	5	Nr 148 bis 161 Betriebsarten	15
Nr 44 bis 51 Befehlsstruktur	6	Tabelle 1 Binärcodes für Dezimalziffern	16
Nr 52 bis 61 Programmiersysteme	6	Tabelle 2 Einstellige boolesche Verknüpfungen	16
Nr 62 bis 75 Programmstruktur	7	Tabelle 3 Zweistellige boolesche Verknüpfungen	16
Nr 76 bis 88 Schaltwerke	8	Stichwortverzeichnis	17
Nr 89 bis 118 Funktioneller Aufbau digitaler Rechensysteme	10		

Die Benennungen sind dort fett gedruckt, wo die zugehörigen Begriffe, zum Teil implizit, definiert sind. Das Stichwortverzeichnis verweist auf diese Stellen. Für die mit einem * versehenen Begriffe ist an anderer Stelle der Norm eine Begriffsbestimmung angegeben. Die kursiv gesetzten englischen Benennungen sind nicht Bestandteil dieser Norm; sie sollen nur das Übersetzen erleichtern, jedoch ist nicht sichergestellt, daß sich die deutschen Begriffsbestimmungen in allen Einzelheiten mit den englischen und den amerikanischen decken.

An der Aufstellung dieser Norm waren auch Fachleute aus Österreich und der Schweiz beteiligt.

Begriffe der Analogrechentechnik sind in dieser Norm nur dort berücksichtigt, wo Gemeinsamkeiten mit der Digitalrechentechnik bestehen.

Nr	Benennung	Bestimmung
Datendarstellung		
1	Zeichen character	<p>Ein Element aus einer vereinbarten endlichen Menge von (verschiedenen) Elementen. Die Menge wird Zeichenvorrat (character set) genannt.</p> <p><i>Anmerkungen: Beispiele für Zeichen sind die abstrakten Inhalte von Buchstaben des gewöhnlichen Alphabets, Ziffern *, Interpunktionszeichen, Steuerzeichen (z. B. für Wagenrücklauf) und andere Ideogramme. Zeichen werden üblicherweise durch Schrift (Schriftzeichen) wiedergegeben oder technisch verwirklicht durch Lochkombinationen, Impulsfolgen und dergleichen. „Zeichen“ ist nicht gleichbedeutend mit „Symbol“, siehe Nr 15.</i></p>
2	Alphabet alphabet	<p>Ein (in vereinbarter Reihenfolge) geordneter Zeichenvorrat *</p> <p><i>Anmerkung: Die Begriffsbestimmung enthält als Sonderfall das gewöhnliche, aus Buchstaben bestehende Alphabet.</i></p>

Fortsetzung Seite 2 bis 18
Erläuterungen Seite 19 und 20

Fachnormenausschuß Informationsverarbeitung (FNI) im Deutschen Normenausschuß (DNA)
Fachnormenausschuß Elektrotechnik (FNE) im DNA

Nr	Benennung	Bestimmung
3	alphanumerisch alphanumeric	sich auf einen Zeichenvorrat * beziehend, der mindestens aus den Dezimalziffern * und den Buchstaben des gewöhnlichen Alphabets * besteht
4	binär binary	zweier Werte fähig; die Eigenschaft bezeichnend, eines von zwei Binärzeichen * als Wert anzunehmen <i>Anmerkungen: Der Ausdruck „logisch“ an Stelle von „binär“ ist als mißverständlich zu vermeiden¹⁾.</i> „binär“ ist nicht gleichbedeutend mit „dual“, siehe Nr 21.
5	Binärzeichen binary character	Jedes der Zeichen * aus einem Zeichenvorrat * von zwei Zeichen <i>Anmerkung: Als Binärzeichen können beliebige Zeichen benutzt werden, insbesondere 0 und 1; wenn keine Verwechslung mit Ziffern * zu befürchten ist, auch 0 und 1. Auch Ja und Nein, Wahr und Falsch, 12 V und 2 V sind Paare von Binärzeichen.</i>
6	Bit bit	1. Kurzform für Binärzeichen *; auch für Dualziffer *, wenn es auf den Unterschied nicht ankommt (das Bit, die Bits) 2. Sondereinheit für die Anzahl der Binärentscheidungen (Kurzzeichen bit) <i>Anmerkung: Alle logarithmisch definierten Größen der Informationstheorie, wie Entscheidungsgehalt, Informationsgehalt, Redundanz usw., erhält man in bit, wenn der Logarithmus zur Basis Zwei genommen wird (1 bit, 2 bit, ...).</i>
7	Wort word	Eine Folge von Zeichen *, die in einem bestimmten Zusammenhang als eine Einheit betrachtet wird <i>Anmerkung: Im Grenzfall kann ein Wort aus einem einzigen Zeichen bestehen.</i>
8	Stelle position	Die einem Zeichen * innerhalb einer Folge von Zeichen zukommende Lage <i>Anmerkung: Insbesondere die örtliche oder zeitliche Lage einer Ziffer * innerhalb einer Zahl.</i>
9	Code code	1. Eine Vorschrift für die eindeutige Zuordnung (Codierung) der Zeichen * eines Zeichenvorrats * zu denjenigen eines anderen Zeichenvorrats (Bildmenge) <i>Anmerkung: Die Abbildung braucht nicht umkehrbar eindeutig zu sein.</i> 2. Der bei der Codierung als Bildmenge auftretende Zeichenvorrat <i>Anmerkung: Die Zeichen der Bildmenge können selbst Wörter * aus Elementen eines anderen Zeichenvorrats sein.</i>
10	Binärcode binary code	Ein Code *, bei dem jedes Zeichen * der Bildmenge ein Wort * aus Binärzeichen * ist (Binärwort). Sofern jedes Wort aus n Binärzeichen besteht, heißt es auch n-Bit-Zeichen .
11	Fehlererkennungscode error detecting code	Ein Code *, dessen Zeichen * nach solchen Gesetzen gebildet werden, die es ermöglichen, durch Störungen verursachte Abweichungen von diesen Gesetzen zu erkennen <i>Anmerkungen: Solche Codes gehören zu den redundanten Codes. Es verbleibt die Möglichkeit, daß durch Störungen aus einem Zeichen ein anderes entsteht, das diesen Gesetzen genügt und deshalb nicht als fehlerhaft erkannt wird.</i>
12	Fehlerkorrekturcode error correcting code	Ein Fehlererkennungscode *, bei dem eine Teilmenge der gestörten Zeichen * auf Grund des Bildungsgesetzes (ohne Rückfrage) korrigiert werden kann

¹⁾ Statt „logischer Entwurf“ sage „funktioneller Entwurf“.

Nr	Benennung	Bestimmung
13	binärer Fehlererkennungscode <i>binary error detecting code</i>	Ein Binärkode*, dessen Zeichen* voneinander mindestens die Hamming-Distanz 2 haben ²⁾
14	binärer Fehlerkorrekturcode <i>binary error correcting code</i>	Ein Binärkode*, dessen Zeichen* voneinander mindestens die Hamming-Distanz 3 haben ²⁾
15	Symbol <i>symbol</i>	Ein Zeichen* oder Wort*, dem eine Bedeutung beigemessen wird
16	Daten <i>data</i>	Durch Zeichen* oder kontinuierliche Funktionen auf Grund bekannter oder unterstellter Abmachungen zum Zwecke der Verarbeitung dargestellte Informationen. Sind Daten im betrachteten Zusammenhang zu einem bestimmten Zeitpunkt Objekt der Verarbeitung oder Übertragung, werden sie Nutzdaten genannt. Daten, welche zum selben Zeitpunkt die Verarbeitung oder Übertragung gemäß gegebener Arbeitsbedingungen beeinflussen, heißen Leitdaten .
17	digitale Daten <i>digital data</i>	Auf Grund bekannter oder unterstellter Abmachungen zum Zwecke der Verarbeitung codiert* dargestellte Informationen <i>Anmerkung:</i> Digitale Daten sind also aus Zeichen* aufgebaut.
18	Datenträger <i>input/output medium, data carrier</i>	Ein Mittel, auf dem Daten* aufbewahrt werden können <i>Anmerkung:</i> Beispiele für Datenträger sind Lochkarten, Lochstreifen, Magnetbänder, Magnetplatten, Magnettrommeln und maschinell lesbare Schriftstücke.
19	Signal <i>signal</i>	Die Darstellung von Informationen durch den Wert oder Werteverlauf einer physikalischen Größe <i>Anmerkungen:</i> Die Darstellung kann sich auf Übertragung, Verarbeitung und Speicherung von Informationen beziehen. Bei abstrakten Betrachtungen kann die Bezugnahme auf eine bestimmte physikalische Größe entfallen, falls die physikalische Verwirklichung nicht interessiert oder noch nicht festgelegt ist. In diesen Fällen können auch Werte und Werteverläufe einer mathematischen Größe, die der abstrakten Betrachtung zugrunde liegt, Signale genannt werden.
20	Signalparameter	Diejenige Kenngröße des Signals*, welche die Informationen trägt <i>Anmerkung:</i> Zum Beispiel ist bei einer amplitudenmodulierten Wechselspannung als Signal die Amplitude der Signalparameter.
21	analoges Signal <i>analog signal</i>	Ein Signal*, bei dem einem kontinuierlichen Wertebereich des Signalparameters* Punkt für Punkt unterschiedliche Informationen zugeordnet sind
22	digitales Signal <i>digital signal</i>	Ein Signal* mit einer endlichen Anzahl von Wertebereichen des Signalparameters*, wobei jedem Wertebereich als Ganzem eine bestimmte Information zugeordnet ist <i>Anmerkung:</i> Eine solche bestimmte Information ist ein Zeichen*.
23	Binärsignal <i>(Zweipunktsignal)</i> <i>binary signal</i>	Ein digitales Signal* mit nur zwei Wertebereichen des Signalparameters*

²⁾ Zur Hamming-Distanz siehe Hamming, R. W.: Error detecting and error correcting codes. Bell Syst. Techn. Journal Vol 29 (1950) No. 2, p. 147–160.

Nr	Benennung	Bestimmung
Zahlendarstellung		
24	Ziffer digit	<p>Ein Zeichen * aus einem Zeichenvorrat * von N Zeichen, denen als Zahlenwerte die ganzen Zahlen $0, 1, 2, \dots, N-1$ umkehrbar eindeutig zugeordnet sind.</p> <p>Je nach der Anzahl N nennt man die zugrunde liegenden Ziffern Dualziffern (<i>binary digits</i>, $N = 2$), Oktalziffern ($N = 8$), Dezimalziffern (<i>decimal digits</i>, $N = 10$), Dodezimalziffern ($N = 12$), Sedezimalziffern ($N = 16$).</p> <p><i>Anmerkung:</i> „dual“ ist nicht gleichbedeutend mit „binär“, siehe Nr 4.</p>
25	Binärkode für Dezimalziffern <i>binary coded decimal (BCD-) code</i>	<p>Darstellung von Dezimalziffern * durch Binärwörter *.</p> <p>In Tabelle 1 sind einige Binärcodes für Dezimalziffern angegeben.</p>
26	Pseudodezimale <i>pseudo-decimal digit</i>	<p>Bei einem Binärkode für Dezimalziffern * jedes Binärwort *, das keiner der Dezimalziffern * 0 bis 9 zugeordnet ist</p> <p><i>Anmerkungen:</i> Die Pseudodezimale gehören nicht zum Binärkode für Dezimalziffern.</p> <p>Zum Beispiel ist beim 8-4-2-1-Code 1011 eine Pseudodezimale.</p>
27	Stellenschreibweise <i>positional notation</i>	<p>Eine Darstellungsart für Zahlen, bei welcher der Beitrag jeder Ziffer * von ihrer Stelle * und ihrem Zahlenwert * abhängt</p> <p><i>Anmerkung:</i> Die häufigste Stellenschreibweise ist die Radixschreibweise *. Ein Beispiel für eine andere Stellenschreibweise ist die Angabe einer Zeitspanne in Tagen, Stunden und Minuten.</p>
28	Radixschreibweise <i>radix notation</i>	<p>Eine Stellenschreibweise *, bei welcher der Beitrag jeder Ziffer * als Produkt aus dem Zahlenwert * der Ziffer und der der Stelle * entsprechenden Potenz einer bestimmten Grundzahl B, der Basis der Zahlendarstellung (<i>radix or base of the number representation</i>), gebildet wird</p> <p><i>Anmerkung:</i> Je nach dem Zahlenwert der Basis B spricht man von einem Dualsystem ($B = 2$), Oktalsystem ($B = 8$), Dezimalsystem ($B = 10$), Dodezimalsystem ($B = 12$), Sedezimalsystem ($B = 16$).</p>
29	Radixpunkt <i>radix point</i>	<p>Bei der Radixschreibweise * die Grenze zwischen dem ganzzahligen und dem gebrochenen Teil der Zahl</p> <p><i>Anmerkung:</i> Die Grenze wird häufig durch ein bestimmtes Trennzeichen, wie Komma oder Punkt, markiert. In Programmiersprachen * wird überwiegend der Punkt benutzt.</p>
30	halblogarithmische Schreibweise <i>floating-point representation</i>	<p>Die Schreibweise von Zahlen Z durch Zahlenpaare x und y mit der Bedeutung $Z = x \cdot b^y$, wobei b eine natürliche Zahl, die Basis der halblogarithmischen Schreibweise (<i>floating-point radix</i>), ist. Die Zahl x wird Mantisse (<i>fixed-point part</i>), die Zahl y Exponent (<i>exponent</i>) genannt. Beide werden üblicherweise in Radixschreibweise * geschrieben. Die hierbei benutzten Basen brauchen weder miteinander noch mit b übereinzustimmen.</p>
31	Festpunkt-Rechnung <i>fixed-point computation</i>	<p>Die Verknüpfung von Zahlen in Radixschreibweise * durch ein Verfahren, das den Radixpunkt * an einem festen Platz relativ zum Zahlenanfang oder Zahlenende unterstellt</p>
32	Gleitpunkt-Rechnung <i>floating-point computation</i>	<p>Die Verknüpfung von Zahlen in halblogarithmischer Schreibweise *</p>

Nr	Benennung	Bestimmung
Programmiersprache		
33	Anweisung statement	<p>Eine in einer beliebigen Sprache abgefaßte Arbeitsvorschrift, die im gegebenen Zusammenhang im Sinne der benutzten Sprache abgeschlossen ist.</p> <p>Eine Anweisung heißt bedingte Anweisung, wenn sie eine Vorschrift zur Prüfung einer Bedingung enthält.</p> <p><i>Anmerkungen:</i> Anweisungen können nach Art der Arbeitsvorschriften klassifiziert werden. Wichtige Klassen sind:</p> <ul style="list-style-type: none"> Arithmetische Anweisung Boolesche Anweisung Verzweigungsanweisung Sprunganweisung Transportanweisung <p>Eine Anweisung kann Teile enthalten, die wiederum Anweisungen sind. Siehe aber Befehl.</p>
34	Befehl instruction	<p>Eine Anweisung *, die in der benutzten Sprache keinen Teil mehr enthält, der selbst Anweisung ist</p>
35	Vereinbarung declaration	<p>Ein Teil einer Anweisung *, der definitorischen Charakter hat und/oder eine ergänzende Erläuterung zu Teilen der Arbeitsvorschrift darstellt</p> <p><i>Anmerkungen:</i> Vereinbarungen sind also keine Anweisungen, können jedoch ihrerseits Anweisungen enthalten.</p> <p>Beispiele für Vereinbarungen sind:</p> <ul style="list-style-type: none"> Prozedur-Vereinbarung *, Format-Vereinbarung, Dimensions-Vereinbarung.
36	Prozedur-Anweisung procedure statement	<p>Eine Anweisung *, deren Bedeutung als Arbeitsvorschrift nur zusammen mit einer Vereinbarung * im Programm * (Prozedur-Vereinbarung) definiert ist. Bestandteile der Prozedur-Anweisung, welche die in dieser Vereinbarung enthaltene Arbeitsvorschrift vervollständigen, heißen aktuelle Prozedur-Parameter (Kurzwort: Aktualparameter). Ihnen entsprechen in der Vereinbarung formale Prozedur-Parameter (Kurzwort: Formalparameter).</p>
37	Prozedur procedure	<p>Die durch eine Prozedur-Vereinbarung * definierte Arbeitsvorschrift</p> <p><i>Anmerkungen:</i> Diese Arbeitsvorschrift ist unvollständig, soweit sie formale Prozedur-Parameter * enthält.</p> <p>Eine Prozedur kann innerhalb des Gültigkeitsbereiches der Prozedur-Vereinbarung an beliebiger Stelle und beliebig oft durch Prozedur-Anweisungen * aufgerufen werden.</p> <p>In maschinenorientierten Programmiersprachen * werden Prozeduren auch Unterprogramme genannt.</p>
38	Programm program	<p>Eine vollständige Anweisung * zur Lösung einer Aufgabe mittels einer digitalen Rechenanlage *</p>
39	Programmiersprache programming language	<p>Eine zum Abfassen von Programmen * geschaffene Sprache</p>
40	problemorientierte Programmiersprache problem oriented language	<p>Eine Programmiersprache *, die dazu dient, Programme * aus einem bestimmten Anwendungsbereich unabhängig von einer bestimmten digitalen Rechenanlage * abzufassen und die sich an eine in dem betreffenden Bereich übliche Schreib- oder Sprechweise anlehnt</p> <p><i>Anmerkungen:</i> Solche Sprachen sind z. B. ALGOL, COBOL, FORTRAN, EXAPT.</p> <p>Benennungen wie anwendungsorientiert, benutzerorientiert, verfahrensorientiert anstelle von problemorientiert sind aus Gründen der Einheitlichkeit zu vermeiden.</p>

Nr	Benennung	Bestimmung
41	maschinenorientierte Programmiersprache <i>computer oriented language</i>	Eine Programmiersprache*, deren Anweisungen* die gleiche oder eine ähnliche Struktur wie die Befehle* einer bestimmten digitalen Rechenanlage* haben
42	Maschinensprache <i>computer language</i>	Eine maschinenorientierte Programmiersprache*, die zur Abfassung von Arbeitsvorschriften nur Befehle* zuläßt, und zwar solche, die Befehlswörter* einer bestimmten digitalen Rechenanlage* sind
43	Befehlsvorrat <i>instruction set</i>	Die Menge der zulässigen Befehle* einer bestimmten maschinenorientierten Programmiersprache*
Befehlsstruktur		
44	Adresse <i>address</i>	Ein bestimmtes Wort* zur Kennzeichnung eines Speicherplatzes, eines zusammenhängenden Speicherbereiches oder einer Funktionseinheit* <i>Anmerkung: Der Zusammenhang muß nicht durch den technischen Aufbau des Speichers* gegeben sein, sondern kann durch irgendwelche, auch wechselnde, Vorschriften definiert sein.</i>
45	Maschinenadresse <i>machine address</i>	Eine Adresse* zur Kennzeichnung einer Speicherzelle*
46	Befehlswort <i>instruction word</i>	Ein Wort*, das von einer digitalen Rechenanlage* als ein Befehl* interpretiert wird <i>Anmerkung: Ein Befehlswort kann mehr als ein Maschinenwort* umfassen, oder es können mehrere Befehlswörter in einem Maschinenwort enthalten sein.</i>
47	Operationsteil <i>operation part</i>	Der Teil eines Befehlswortes*, der die auszuführende Operation angibt <i>Anmerkung: Die Zeichen* dieses Teils müssen nicht in aufeinanderfolgenden Stellen* stehen.</i>
48	Operationscode <i>operation code</i>	Ein Code* zur Darstellung des Operationsteils* von Befehlswörtern*
49	Operandenteil <i>operand part</i>	Der Teil eines Befehlswortes*, der Operanden oder Angaben zum Auffinden von Operanden oder Befehlswörtern enthält <i>Anmerkung: Die Zeichen* dieses Teils müssen nicht in aufeinanderfolgenden Stellen* stehen.</i>
50	Adressenteil <i>address part</i>	Ein Operandenteil* oder ein Bestandteil eines Operandenteils, der Adressen* von Operanden oder Befehlswörtern* enthält <i>Anmerkung: Die Zeichen* dieses Teils müssen nicht in aufeinanderfolgenden Stellen* stehen.</i>
51	Befehlsliste <i>instruction list</i>	Der Operationscode* einer bestimmten maschinenorientierten Programmiersprache* mit Beschreibung der zugehörigen Funktionen und mit Angaben über die Operandenteile*
Programmiersysteme		
52	Betriebssystem <i>operating system</i>	Die Programme* eines digitalen Rechensystems*, die zusammen mit den Eigenschaften der Rechenanlage* die Basis der möglichen Betriebsarten des digitalen Rechensystems bilden und insbesondere die Abwicklung von Programmen steuern und überwachen. Eine Sprache, der ein Betriebssystem gehorcht, heißt Betriebssprache (<i>operating language</i>).

Nr	Benennung	Bestimmung
53	Übersetzer <i>translator</i>	Noch dieser Norm ein Programm*, das in einer Programmiersprache* A (Ursprungssprache) abgefaßte Anweisungen* ohne Veränderung der Arbeitsvorschriften in Anweisungen einer Programmiersprache B (Zielsprache) umwandelt (übersetzt). Die in der Ursprungssprache abgefaßte Anweisung wird Ursprungsanweisung oder Ursprungssprogramm , die in der Zielsprache entstandene Anweisung wird Zielanweisung bzw. Zielprogramm genannt.
54	Kompilierer <i>compiler</i>	Ein Übersetzer*, der in einer problemorientierten Programmiersprache* abgefaßte Ursprungsanweisungen* in Zielanweisungen* einer maschinenorientierten Programmiersprache* umwandelt (kompiliert)
55	Assemblierer <i>assembler</i>	Ein Übersetzer*, der in einer maschinenorientierten Programmiersprache* abgefaßte Ursprungsanweisungen* in Zielanweisungen* der zugehörigen Maschinensprache* umwandelt (assembliert)
56	Interpretierer <i>interpreter</i>	Ein Programm*, das es gestaltet, auf einer bestimmten digitalen Rechenanlage* Anweisungen*, die in einer von der Maschinensprache* dieser Anlage verschiedenen Sprache abgefaßt sind, ausführen (interpretieren) zu lassen
57	Simulierer <i>simulator</i>	Ein Interpretierer*, bei dem das zu interpretierende Programm* in einer maschinenorientierten Programmiersprache* abgefaßt ist
58	Simulator <i>simulator</i>	Nach dieser Norm ein Programm*, das die Gesetze eines Prozesses auf einer Rechenanlage* nachbildet und damit die Anlage als Modell auffassen läßt
59	Programmiersystem <i>programming system</i>	Eine Programmiersprache* und alle Programme*, die dazu dienen, in dieser Programmiersprache abgefaßte Programme für eine bestimmte digitale Rechenanlage* ausführbar zu machen
60	Betriebsanweisung <i>directive</i>	Eine Anweisung* an das Betriebssystem*
61	Übersetzungsanweisung <i>directive</i>	Eine Anweisung* an einen Übersetzer*

Programmstruktur

62	Datenfluß <i>data flow</i>	Die zeitlichen Beziehungen im Ablauf einer Menge zusammengehöriger Vorgänge an Daten* und Datenträgern* <i>Anmerkung:</i> Die zeitlichen Beziehungen enthalten keine Aussage über die Ablaufgeschwindigkeit.
63	Datenflußplan <i>data flowchart</i>	Die Darstellung eines Datenflusses*, die im wesentlichen aus Sinnbildern mit zugehörigem Text und orientierten Verbindungslien besteht <i>Anmerkung:</i> Sinnbilder für Datenflußpläne siehe DIN 66 001.
64	Programmablauf <i>program flow</i>	Die zeitlichen Beziehungen zwischen den Teilvergängen, aus denen sich die folgerichtige Ausführung eines Programms* zusammensetzt
65	Programmablaufplan <i>program flowchart</i>	Die Darstellung der Gesamtheit aller beim Programmablauf* möglichen Wege* <i>Anmerkung:</i> Sinnbilder für Programmablaufpläne siehe DIN 66 001.

Nr	Benennung	Bestimmung
66	Strecke	<p>Eine Folge von Anweisungen*, die dann und nur dann in der Reihenfolge der Niederschrift ausgeführt werden, wenn die erste Anweisung angesprochen wird</p> <p><i>Anmerkung: Zur Darstellung von Strecken werden Ablauflinien und/oder Sinnbilder für Operationen benutzt, siehe DIN 66 001.</i></p>
67	Weg	<p>Eine (zusammenhängende) Folge von Strecken*</p>
68	Zweig	<p>Ein Teil eines Programmablaufplans*, bestehend aus allen Wegen*, die mit genau einer Strecke* beginnen, oder aus allen Wegen, die in genau einer Strecke enden</p>
69	abgeschlossener Zweig	<p>Ein Zweig*, der mit einer Strecke* beginnt und in einer Strecke endet</p> <p><i>Anmerkung: Ein abgeschlossener Zweig kann in einer größeren Darstellung als Strecke erscheinen.</i></p>
70	Verzweigung <i>branch</i>	<p>Eine Stelle im Programmablaufplan*, an der im Programmablauf* einer und nur einer von mehreren möglichen Zweigen* eingeschlagen wird.</p> <p>Die Benennung wird bei der Beschreibung des Programmablaufes auch für den Vorgang des Verzweigens, d. h. die Auswahl eines der möglichen Zweige, benutzt.</p> <p><i>Anmerkung: Sinnbild siehe DIN 66 001.</i></p>
71	Zusammenführung <i>junction</i>	<p>Eine Stelle im Programmablaufplan*, an der mehrere Zweige* zusammenlaufen.</p> <p>Die Benennung wird bei der Beschreibung des Programmablaufes* auch für den Vorgang des Zusammenführens benutzt.</p> <p><i>Anmerkung: Sinnbild siehe DIN 66 001.</i></p>
72	Schleife <i>loop</i>	<p>Ein Paar von Zusammenführung* und Verzweigung* und ein Paar von abgeschlossenen Zweigen*, von denen der eine von der Zusammenführung zur Verzweigung und der andere von der Verzweigung zurück zur Zusammenführung führt</p>
73	Aufspaltung	<p>Eine Stelle im Programmablaufplan*, von der aus im Programmablauf* mehrere Zweige* parallel verfolgt werden können.</p> <p>Die Benennung wird bei der Beschreibung des Programmablaufes auch für den Vorgang des Aufspaltens benutzt.</p>
74	Sammlung	<p>Eine Stelle im Programmablaufplan*, an der im Programmablauf* alle in den zusammenlaufenden Zweigen* parallel ablaufenden Tätigkeiten zu Ende gebracht sein müssen, ehe der weiterführende Zweig verfolgt wird.</p> <p>Die Benennung wird bei der Beschreibung des Programmablaufes auch für den Vorgang des Sammelns benutzt.</p>
75	Synchronisations-schnitt	<p>Eine Stelle im Programmablaufplan*, an der unabhängige Wege* im Programmablauf* zeitlich aufeinander abgestimmt werden</p>
Schaltwerke		
76	Schaltvariable <i>logic variable</i>	<p>Eine Variable, die nur endlich viele Werte annehmen kann</p> <p><i>Anmerkung: Die Menge dieser Werte bildet einen Zeichenvorrat*. Am häufigsten sind binäre* Schaltvariablen.</i></p>

Nr	Benennung	Bestimmung
77	Schaltfunktion <i>logic function</i>	Eine Funktion, bei der jede Argument-Variable und die Funktion selbst nur endlich viele Werte annehmen können. Wird eine Schaltfunktion mit Hilfe eines Operators dargestellt, spricht man von Verknüpfung (<i>logic operation</i>). Boolesche Verknüpfungen siehe Tabellen 2 und 3.
78	boolesch <i>Boolean</i>	binär*, überdies darauf hinweisend, daß über den binären Schaltvariablen* die Schaltfunktionen* der Booleschen Algebra ³⁾ ausgeführt werden (siehe Tabellen 2 und 3). <i>Anmerkungen:</i> In diesem Zusammenhang heißt es also „boolesche Schaltvariable“ ⁴⁾ und „boolesche Verknüpfungen“. Der Ausdruck „logisch“ an Stelle von „boolesch“ ist als mißverständlich zu vermeiden ¹⁾ .
79	Schaltwerk <i>sequential circuit</i>	Eine Funktionseinheit* zum Verarbeiten von Schaltvariablen*, wobei der Wert am Ausgang zu einem bestimmten Zeitpunkt abhängt von den Werten am Eingang zu diesem und endlich vielen vorangegangenen Zeitpunkten. <i>Anmerkung:</i> Ein Schaltwerk kann somit eine endliche Anzahl von inneren Zuständen annehmen und ist, abstrakt gesehen, ein endlicher Automat ⁵⁾ . Man kann also auch sagen: Der Zustand am Ausgang zu einem bestimmten Zeitpunkt hängt ab vom inneren Zustand und dem Wert am Eingang.
80	Schaltglied	Ein Bestandteil des Schaltwerks*. Schaltglieder werden eingeteilt in verknüpfende und speichernde Schaltglieder: Verknüpfungsglieder* und Speicherglieder*.
81	Speicherglied	Ein Schaltglied*, das Schaltvariable* aufnimmt, aufbewahrt und abgibt. <i>Anmerkung:</i> Die Abgabe kann je nach dem technischen Aufbau ständig oder zu bestimmten Zeitpunkten oder auf Anforderung hin erfolgen.
82	Flipflop <i>flipflop</i>	Ein Speicherglied* mit zwei stabilen Zuständen, das aus jedem der beiden Zustände durch eine geeignete Ansteuerung in den anderen Zustand übergeht (bistabiles Kippglied)
83	Verknüpfungsglied <i>logic element</i>	Ein Schaltglied*, das eine Verknüpfung* von Schaltvariablen* bewirkt. Spezielle Verknüpfungsglieder sind: NICHT-Glied (NOT element), UND-Glied (AND element), ODER-Glied (OR element), NAND-Glied (NAND element), NOR-Glied (NOR element) <i>Anmerkung:</i> Der Ausdruck „Gatter“ ist als mißverständlich zu vermeiden.
84	Schaltnetz <i>combinatorial circuit</i>	Ein Schaltwerk*, dessen Wert am Ausgang zu irgendeinem Zeitpunkt nur vom Wert am Eingang zu diesem Zeitpunkt abhängt. <i>Anmerkungen:</i> Bei einem Schaltnetz können innere Zustände von außen nicht unterschieden werden. Die für die prinzipielle Funktion unwesentlichen Übergangs- und Verzögerungszeiten bleiben hier außer Betracht. Ein Schaltnetz läßt sich aus Verknüpfungsgliedern* aufbauen. Ein Code-Zuordner ist ein Beispiel für ein Schaltnetz. Der Ausdruck „Schaltkreis“ an Stelle von „Schaltnetz“ ist als mißverständlich zu vermeiden.

¹⁾ Siehe Seite 2³⁾ Nach George Boole, englischer Mathematiker, 1815—1864⁴⁾ Im Sinne der Mathematik gleich „boolesche Variable“⁵⁾ Siehe z. B. Gluschkow, W. M.: Theorie der abstrakten Automaten, VEB Verlag der Wissenschaften, Berlin 1963, Seite 9

Nr	Benennung	Bestimmung
85	Addierglied <i>full adder</i>	<p>Ein Schaltnetz* mit zwei Eingängen für Ziffern*, mit einem Eingang für den ankommenen Übertrag und mit je einem Ausgang für die Stellensumme und für den abgehenden Übertrag</p> <p><i>Anmerkung: Für den Fall, daß es sich um Ziffern von 0 bis N-1 handelt, ist die Stellensumme die arithmetische Summe modulo N.</i></p>
86	Zähler <i>counter</i>	<p>Ein Schaltwerk*, in dem eine Zahl gespeichert ist, zu der durch ein Signal* jeweils eine konstante Zahl, die Zähleinheit, addiert wird</p> <p><i>Anmerkungen: Die Zähleinheit ist eine positive oder negative ganze Zahl, meist 1.</i></p> <p><i>Bei den meisten Zählern besteht die Möglichkeit, den Nullzustand durch ein besonderes Signal direkt einzustellen.</i></p>
87	Zweirichtungszähler <i>reversible counter</i>	<p>Ein Zähler*, bei dem man die Zähleinheit* sowohl addieren als auch subtrahieren kann</p>
88	Modulo-N-Zähler	<p>Ein Zähler*, bei dem die Addition modulo N verstanden wird</p> <p><i>Anmerkung: Ein Modulo-N-Zähler kehrt nach N Schritten mit der Zahl einheit* 1 in seinen ursprünglichen Zustand zurück.</i></p>

Funktioneller Aufbau digitaler Rechensysteme

89	Funktionseinheit <i>functional unit</i>	<p>Ein funktionelles Gebilde, bestimmt durch die von ihm zu bewältigenden Aufgaben</p> <p><i>Anmerkungen: Ein System von Funktionseinheiten kann in einem gegebenen Zusammenhang wieder als eine Funktionseinheit aufgefaßt werden. Bei Benennung bestimmter Funktionseinheiten werden in Zusammensetzungen gebraucht (vorzugsweise in absteigender Rangfolge): -system, -werk, -glied, -element.</i></p> <p><i>Der Funktionseinheit können auf der konstruktiven Seite eine einzige Baueinheit* oder mehrere gleiche oder verschiedene Baueinheiten entsprechen.</i></p>
90	Rechensystem, Datenverarbeitungssystem <i>data processing system</i>	<p>Eine Funktionseinheit* zur Verarbeitung von Daten*, nämlich zur Durchführung boolescher*, arithmetischer, vergleichender, umformender, übertragender und speichernder Operationen</p> <p><i>Anmerkung: Siehe auch Rechenanlage (Nr 120).</i></p>
91	digitales Rechensystem, digitales Datenverarbeitungssystem <i>digital data processing system</i>	<p>Ein Rechensystem*, das, als Funktionseinheit* gesehen, ein Schaltwerk* ist</p> <p><i>Anmerkung: Ein digitales Rechensystem kann also nur digitale Daten* verarbeiten.</i></p>
92	Speicher <i>storage</i>	<p>Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, die digitale Daten* aufnimmt, aufbewahrt und abgibt</p>
93	Rechenwerk <i>arithmetic unit</i>	<p>Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, die Rechenoperationen ausführt</p> <p><i>Anmerkung: Zu den Rechenoperationen können neben den arithmetischen Operationen auch boolesche* Verknüpfungen*, Vergleichen, Umformen, Verschieben, Runden usw. gehören.</i></p>

Nr	Benennung	Bestimmung
94	Leitwerk <i>control unit</i>	Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, die die Reihenfolge steuert, in der die Befehle* eines Programms* ausgeführt werden, die diese Befehle entschlüsselt und dabei gegebenenfalls modifiziert und die für ihre Ausführung erforderlichen digitalen Signale* abgibt <i>Anmerkung: Die Befehle können in einem gesonderten Befehlsrechenwerk oder Adressenrechenwerk oder auch im allgemeinen Rechenwerk* modifiziert werden.</i>
95	Prozessor <i>processor</i>	Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, die Rechenwerk* und Leitwerk* umfaßt
96	Eingabewerk	Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, die das Übertragen von Daten* von Eingabeeinheiten* oder peripheren Speichern* in den Zentralspeicher* steuert
97	Ausgabewerk	Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, die das Übertragen von Daten* vom Zentralspeicher* in Ausgabeeinheiten* oder peripheren Speicher* steuert
98	EA-Werk	Eine Funktionseinheit*, welche die Funktionen von Eingabewerk* und Ausgabewerk* in sich vereinigt. Ein Prozessor*, der als EA-Werk dient, wird EA-Prozessor genannt.
99	Zentralspeicher	Ein Speicher*, zu dem Prozessoren* sowie Eingabe- und Ausgabewerke* unmittelbar Zugang haben
100	Zentraleinheit, Rechner	Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, welche die Prozessoren*, die Eingabe- und Ausgabewerke* und die Zentralspeicher* umfaßt
101	periphere Einheit <i>peripheral unit</i>	Jede Funktionseinheit* innerhalb eines digitalen Rechensystems*, die nicht zur Zentraleinheit* gehört
102	Eingabeeinheit <i>input unit</i>	Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, mit der das System Daten* von außen her aufnimmt
103	Ausgabeeinheit <i>output unit</i>	Eine Funktionseinheit* innerhalb eines digitalen Rechensystems*, mit der das System Daten*, z. B. Rechenergebnisse, nach außen hin abgibt
104	peripherer Speicher	Jeder Speicher*, der nicht Zentralspeicher* ist
105	Benutzerstation <i>user terminal</i>	Eine Funktionseinheit* innerhalb eines Rechensystems*, mit deren Hilfe einem Benutzer der direkte Informationsaustausch mit dem Rechensystem möglich ist <i>Anmerkung: Benutzerstationen werden oft über größere Entfernungen (Platzbuchungsanlagen, Bankbetrieb) mit dem zugehörigen Rechensystem verbunden.</i>
106	Mehrprozessorsystem <i>multiprocessor</i>	Ein digitales Rechensystem*, bei dem mindestens zwei unabhängige Prozessoren* einen Zentralspeicher* oder Teile davon gemeinsam benutzen
107	Mehrrechnersystem <i>multicomputer system</i>	Ein digitales Rechensystem*, bei dem mindestens zwei unabhängige Zentraleinheiten* periphere Einheiten* gemeinsam benutzen
108	Teilnehmer-Rechensystem	Ein digitales Rechensystem* mit mehreren angeschlossenen Benutzerstationen*, von denen aus unabhängig Aufgaben abgewickelt werden können

Nr	Benennung	Bestimmung
109	Umsetzer <i>converter</i>	Eine Funktionseinheit * zum Ändern der Darstellung von Daten * <i>Anmerkung: Die Änderung kann sich sowohl auf die Codierung * als auch auf den Datenträger * oder die zur Darstellung benutzte physikalische Größe beziehen.</i>
110	Code-Umsetzer <i>code translator</i>	Ein Umsetzer *, in dem den Zeichen * eines Code * A Zeichen eines Code B zugeordnet werden <i>Anmerkung: Als Code-Umsetzer kann ein Schaltwerk * dienen.</i>
111	Codierer <i>coder</i>	Ein Code-Umsetzer * mit mehreren Eingängen und Ausgängen, bei dem immer nur einem Eingang ein Signal * zugeführt wird und bei dem jedes dieser Eingangssignale eine spezifische Kombination von Ausgangssignalen zur Folge hat
112	Decodierer <i>decoder</i>	Ein Code-Umsetzer * mit mehreren Eingängen und Ausgängen, bei dem für jede spezifische Kombination von Eingangssignalen immer nur je ein bestimmter Ausgang ein Signal * abgibt
113	Serie-Parallel-Umsetzer <i>serial-parallel converter</i>	Ein Umsetzer *, in dem zeitlich sequentiell dargestellte digitale Daten * in parallel dargestellte digitale Daten umgewandelt werden
114	Parallel-Serie-Umsetzer <i>parallel-serial converter</i>	Ein Umsetzer *, in dem parallel dargestellte digitale Daten * in zeitlich sequentiell dargestellte digitale Daten umgewandelt werden
115	Multiplexer <i>multiplexer</i>	Eine Funktionseinheit *, die von mehreren Nachrichtenkanälen ⁶⁾ auf eine andere Anzahl von Nachrichtenkanälen umsetzt <i>Anmerkung: Zum Beispiel kann von vielen „langsam“ Nachrichtenkanälen auf einen „schnellen“ durch zeitliche Verschachtelung umgesetzt werden.</i>
116	Uhr <i>clock</i>	Nach dieser Norm eine vom Programm * her abfragbare Funktionseinheit * zur Angabe der Zeit
117	Zykluszeit <i>cycle time</i>	Bei einer Funktionseinheit * die kleinstmögliche Zeitspanne zwischen dem Beginn zweier aufeinanderfolgender gleichartiger zyklisch wiederkehrender Vorgänge
118	Zugriffszeit <i>access time</i>	Bei einer Funktionseinheit * die Zeitspanne zwischen dem Zeitpunkt, zu dem von einem Leitwerk * die Übertragung bestimmter Daten * nach oder von der Funktionseinheit gefordert wird, und dem Zeitpunkt, zu dem die Übertragung beendet ist

Konstruktiver Aufbau digitaler Rechenanlagen

119	Baueinheit	Ein konstruktives Gebilde, bestimmt durch technische Erfordernisse und Gesichtspunkte <i>Anmerkung: Bei Benennung bestimmter Baueinheiten werden in Zusammensetzungen gebraucht (vorzugsweise in absteigender Rangfolge): -anlage, -gerät, -teil.</i>
120	Rechenanlage, Datenverarbeitungsanlage <i>computer</i>	Die Gesamtheit der Baueinheiten *, aus denen ein Rechensystem * aufgebaut ist

⁶⁾ Begriff siehe DIN 44 301

Nr	Benennung	Bestimmung
121	digitale Rechenanlage, digitale Datenverarbeitungsanlage digital computer	Die Gesamtheit der Baueinheiten*, aus denen ein digitales Rechensystem* aufgebaut ist
122	Taktgeber clock	Ein Pulsgenerator zur Synchronisierung von Operationen in digitalen Rechenanlagen*
123	Bedienungsfeld operator control panel	In einer Rechenanlage* eine Baueinheit*, die dem Bedienungspersonal Eingriffe in den Betrieb erlaubt <i>Anmerkungen: Beispiele für solche Eingriffe sind Ureingabe von Programmen, Überwachung von Alarmaneinrichtungen, Ein-, Aus- und Umschalten der Rechenanlage oder ihrer Teile.</i> <i>Für das technische Personal gibt es gelegentlich ein eigenes „Wartungsfeld“.</i>
124	Eingabegerät input device	In einer Eingabeeinheit* eine Baueinheit*, durch die Daten* mit Hilfe von Datenträgern* oder auf eine andere Weise in eine Rechenanlage* eingegeben werden können
125	Kurvenleser curve follower	Ein Eingabegerät* zum Umsetzen gezeichneter Kurven in Signale*
126	manuelle Eingabe manual input device	Ein Eingabegerät* in der Funktion, den Benutzer Daten* manuell eingeben zu lassen <i>Anmerkung: Beispiele sind Tastaturen und Lichtgriffel.</i>
127	Ausgabegerät output device	In einer Ausgabeeinheit* eine Baueinheit*, durch die Daten* mit Hilfe von Datenträgern* oder auf eine andere Weise von einer Rechenanlage* ausgegeben werden können
128	Kurvenschreiber plotter	Ein Ausgabegerät* zum Zeichnen von Kurven oder Einzelpunkten
129	Sichtanzeigegerät display device	Ein Ausgabegerät* in der Funktion, dem Benutzer Daten* sichtbar zu machen <i>Anmerkung: Beispiele sind Bildschirmgeräte und Ziffernanzeiger.</i>

Speicher

130	Speicherelement	Ein in einem gegebenen Zusammenhang nicht weiter zerlegbarer Teil eines Speichers*
131	Speicherstelle	Ein Teil eines Speichers* zur Aufnahme eines Zeichens*
132	Speicher mit Wortstruktur word organized storage	Ein Speicher*, dessen Speicherelemente* nur in Gruppen zugänglich sind, deren Länge und Einteilung durch den technischen Aufbau bestimmt sind. Die in einer solchen Gruppe gespeicherten Zeichen* bilden ein Maschinenwort (machine word).
133	Speicherzelle storage location	Bei einem Speicher mit Wortstruktur* eine Gruppe von Speicherelementen*, die ein Maschinenwort* aufnimmt
134	Hauptspeicher	Der Teil des Zentralspeichers*, dessen einzelne Speicherzellen* durch Maschinenadressen* aufgerufen werden können
135	Ergänzungsspeicher	Jeder Teil des Zentralspeichers*, der nicht Hauptspeicher* ist

Nr	Benennung	Bestimmung
136	geschützter Speicherbereich <i>protected storage area</i>	Ein Bereich eines Speichers *, der gegen unerwünschtes Lesen und/oder Überschreiben geschützt ist
137	Festspeicher <i>read-only store</i>	Ein Speicher *, dessen Inhalt betriebsmäßig nur gelesen werden kann <i>Anmerkung: Zum Ändern seines Inhalts sind besondere Maßnahmen, wie Auswechseln des Datenträgers * oder Verdrahtungsänderungen, notwendig.</i>
138	Puffer <i>buffer</i>	Ein Speicher *, der Daten * vorübergehend aufnimmt, die von einer Funktionseinheit * zu einer anderen übertragen werden
139	Merker	Ein Speicherglied *, das durch seinen Zustand den späteren Programmablauf * an Verzweigungen * und Aufspaltungen * zu beeinflussen ermöglicht
140	Register <i>register</i>	Ein Speicher *, der anders als durch Maschinenadressen * aufrufbar ist. Er nimmt gewöhnlich ein Maschinenwort * oder eine Maschinenadresse auf <i>Anmerkungen: Kennzeichnend ist nicht, ob ein Register auch durch Maschinenadressen aufgerufen werden kann.</i> <i>Beispiele für Register sind Akkumulator *, Indexregister *, Funktionell gesehen, werden auch Speicherzellen * des Hauptspeichers * als Register benutzt.</i>
141	Akkumulator <i>accumulator</i>	In einem Rechenwerk * ein Register *, das für Rechenoperationen benutzt wird, wobei es ursprünglich einen Operanden und nach durchgeföhrter Operation das Ergebnis enthält <i>Anmerkung: Insbesondere dient ein Akkumulator zum Bilden von Zwischensummen.</i>
142	Befehlszähler <i>instruction counter</i>	In einem Leitwerk * ein Register *, aus dem die Adresse * des nächsten auszuführenden Befehls * gewonnen wird
143	Befehlsregister <i>instruction register</i>	In einem Leitwerk * ein Register *, aus dem der gerade auszuführende Befehl * gewonnen wird <i>Anmerkung: Die verschiedenen Teile eines Befehlswortes * (z. B. Operationsteil *, Operandenteil *) werden oft auch in getrennten Registern gespeichert.</i>
144	Indexregister <i>index register</i>	Ein Register *, das vorwiegend zum schnellen Modifizieren von Adressen *, zum schnellen Durchführen von Zähloperationen an Adressen und zum schnellen Einleiten einer Verzweigung * dient
145	Kellerspeicher <i>stack</i>	Eine Folge gleichartiger Register * oder Speicherzellen *, von denen nur das bzw. die erste aufgerufen wird. Bei der Eingabe in den Kellerspeicher wird der Inhalt jedes Registers in das nächstfolgende übertragen und das zu speichernde Wort * in das erste Register geschrieben. Bei der Ausgabe wird der Inhalt des ersten Registers gelesen und der Inhalt jedes übrigen Registers an das vorhergehende übertragen. Dasselbe gilt sinngemäß für Speicherzellen. <i>Anmerkung: Kellerspeicher werden oft auch im Hauptspeicher * imitiert. Dann wird nicht der Inhalt der Speicherzellen übertragen, sondern die Adresse * des ersten „Registers“ bei jedem Zugriff verändert. Ein Adreßregister, das diese Adresse enthält, wird auch „Stapelzeiger“ genannt.</i>
146	Assoziativspeicher <i>associative storage</i>	Ein Speicher *, dessen Speicherzellen * durch Angabe eines Teils ihres Inhalts aufrufbar sind
147	Randomspeicher <i>random access storage</i>	Ein Speicher *, bei dem die statistische Erwartung für die Zugriffszeit * eines bestimmten Zugriffs unabhängig vom jeweils vorhergehenden Zugriff ist

Nr	Benennung	Bestimmung
Betriebsarten		
148	Multiplexbetrieb	<p>Eine Funktionseinheit * bearbeitet mehrere Aufgaben, abwechselnd in Zeitabschnitten verzahnt</p> <p><i>Anmerkungen: Die Bearbeitung begonnener Aufgaben wird zugunsten anderer, auch neu zu beginnender, Aufgaben unterbrochen. Die Zeitabschnitte können von unterschiedlicher Länge sein.</i></p>
149	sequentieller Betrieb	<p>Eine Funktionseinheit * bearbeitet mehrere Aufgaben, eine nach der anderen</p>
150	Parallelbetrieb	<p>Mehrere Funktionseinheiten * eines Rechensystems * arbeiten gleichzeitig an Teilaufgaben derselben Aufgabe</p> <p><i>Anmerkung: Die einzelne Funktionseinheit arbeitet dabei entweder im Multiplexbetrieb * oder im sequentiellen Betrieb *.</i></p>
151	Simultanbetrieb	<p>Ein Rechensystem * arbeitet tatsächlich gleichzeitig (Parallelbetrieb *) und/oder scheinbar gleichzeitig (Multiplexbetrieb *) an mehreren Aufgaben</p>
152	Mehrprogrammbetrieb multiprogramming mode	<p>Ein Betrieb eines Rechensystems *, bei dem das Betriebssystem * für den Multiplexbetrieb * der Zentraleinheit(en) * sorgt</p>
153	Abrufbetrieb	<p>Ein Betrieb eines Rechensystems *, bei dem eine Zentraleinheit * nach einer festgelegten Vorschrift von Benutzerstationen * Daten * abruft</p>
154	Anforderungsbetrieb	<p>Ein Betrieb eines Rechensystems *, bei dem eine Zentraleinheit * von einer Benutzerstation * zur Übernahme angebotener Daten * veranlaßt wird</p>
155	Stapelbetrieb batch processing	<p>Ein Betrieb eines Rechensystems *, bei dem eine Aufgabe vollständig gestellt sein muß, bevor mit ihrer Abwicklung begonnen werden kann, und vollständig abgewickelt werden muß, bevor eine neue Aufgabe gestellt werden kann</p>
156	Realzeitbetrieb real time processing	<p>Ein Betrieb eines Rechensystems *, bei dem Programme * zur Verarbeitung anfallender Daten * betriebsbereit sind derart, daß die Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeit verfügbar sind. Die Daten können je nach Anwendungsfall nach einer zeitlich zufälligen Verteilung oder zu vorbestimmten Zeitpunkten anfallen.</p>
157	Dialogbetrieb conversational mode	<p>Ein Betrieb eines Rechensystems *, bei dem zur Abwicklung einer Aufgabe Wechsel zwischen dem Stellen von Teilaufgaben und den Antworten darauf stattfinden können</p>
158	Reaktionszeit	<p>In einer Zentraleinheit * die Zeit zwischen dem Ende des Eintreffens einer Aufgabenstellung und dem Beginn der Bearbeitung</p>
159	Bearbeitungszeit	<p>In einer Zentraleinheit * die Zeit zwischen Beginn und Ende der Bearbeitung einer Aufgabe</p>
160	Anlieferungszeit	<p>An einer Benutzerstation * die Zeit zwischen dem Ende einer Aufgabenstellung und dem Ende der Übertragung der vollständigen Antwort darauf von der Zentraleinheit * her</p>
161	Beantwortungszeit	<p>An einer Benutzerstation * die Zeit zwischen dem Ende einer Aufgabenstellung und dem Vorliegen der vollständigen Antwort darauf</p>

Tabelle 1. Binärcodes für Dezimalziffern

Dezimalziffer decimal digit	8 4 2-1-Code 8-4-2-1 code	5-4-2-1-Code ⁷⁾ biquinary code	2-4-2-1-Code ⁷⁾ 2-4-2-1 code	Exzess-3-Code excess-three code	2-aus-5-Code ⁷⁾ two-out-of-five code	1-aus-10-Code one-out-of-ten code
0	0000	0000	0000	00LL	000LL	000000000L
1	000L	000L	000L	0LOO	00L0L	000000000L0
2	00L0	00L0	00L0	0L0L	00L0L0	000000000L00
3	00LL	00LL	00LL	0LLO	0L00L	000000000L000
4	0LOO	0LOO	0LOO	0LLL	0LOLO	00000L0000
5	0LOL	LO00	LO00	LO00	OL00	0000L00000
6	OL00	LOOL	LOOL	LOOL	LO00L	000L000000
7	OLLL	LOLO	LL0L	LOLO	LO0LO	00L0000000
8	LO00	LOLL	LLLO	LOLL	LOLO0	0L00000000
9	LOOL	LLOO	LLLL	LL00	LL000	L000000000

⁷⁾ Der hier wiedergegebene Code ist nur ein Vertreter von Codes gleicher Benennung.

Tabelle 2. Einstellige boolesche Verknüpfungen

Benennung der Verknüpfung	Definition durch Funktionswert $y = f(a)$		Schreibweise nach DIN 66 000
	für $a =$	0	
Negation <i>negation</i>		L	$\neg a$ oder \bar{a}

Tabelle 3. Zweistellige boolesche Verknüpfungen

Benennung der Verknüpfung	Definition durch Funktionswert $y = f(a, b)$		Schreibweise nach DIN 66 000	Schreibweise mit den Zeichen $\wedge \vee \neg$	Bemerkungen
	für $a =$	$b =$			
UND-Verknüpfung, <i>Konjunktion</i> <i>AND operation</i>	O O O L	O O L L	$a \wedge b$	$a \wedge b$	
ODER-Verknüpfung, <i>Disjunktion</i> <i>OR operation</i>	O L L L	O L L L	$a \vee b$	$a \vee b$	Auch als „inklusives Oder“ bekannt
NAND-Verknüpfung <i>NAND operation</i>	L L L O	L L L O	$\neg a \wedge \neg b$	$\neg a \wedge \neg b$	Auch als „Sheffer-Funktion“ bekannt
NOR-Verknüpfung <i>NOR operation</i>	L O O O	L O O O	$\neg a \vee \neg b$	$\neg a \vee \neg b$	Auch als „Peirce-Funktion“ bekannt
Inhibition <i>exclusion</i>	O O L O	O L O O	—	$a \wedge b$ $a \wedge \bar{b}$	
Implikation <i>inclusion</i>	L O L L	L L O L	$a \supset b$ $b \supset a$	$\bar{a} \vee b$ $a \vee \bar{b}$	
Äquivalenz <i>equivalence</i>	L O O L	L O O L	$a \equiv b$	$(a \wedge b) \vee (\bar{a} \wedge \bar{b})$	
Antivalenz <i>non-equivalence</i>	O L L O	O L L O	$a \neq b$	$(a \wedge \bar{b}) \vee (\bar{a} \wedge b)$	Auch als „exklusives Oder“ bekannt

Stichwortverzeichnis

Stichwort	Nr
abgeschlossener Zweig	69
Abrufbetrieb	153
Addierglied	85
Adresse	44
Adressenteil	50
Äquivalenz	(Tabelle 3)
Akkumulator	141
aktueller Prozedur-Parameter	36
Alphabet	2
alphanumerisch	3
analoges Signal	21
Anforderungsbetrieb	154
Antivalenz	(Tabelle 3)
Antwortzeit	161
Anweisung	33
Anweisung, bedingte	33
Assemblierer	55
assemblieren	55
Assoziativspeicher	146
Aufspaltung	73
Ausgabeeinheit	103
Ausgabegerät	127
Ausgabewerk	97
Auslieferungszeit	160
Basis der halblogarithmischen Schreibweise	30
Basis der Zahlendarstellung	28
Baueinheit	119
Bearbeitungszeit	159
Bedienungsfeld	123
bedingte Anweisung	33
Befehl	34
Befehlsliste	51
Befehlsregister	143
Befehlsvorrat	43
Befehlswort	46
Befehlszähler	142
Benutzerstation	105
Betrieb, sequentieller	150
Betriebsanweisung	60
Betriebssprache	52
Betriebssystem	52
binär	4
Binärkode	10
Binärkode für Dezimalziffern	25
binärer Fehlererkennungscode	13
binärer Fehlerkorrekturcode	14
Binärsignal	23
Binarwort	10
Binärzeichen	5
Bit	6
boolesch	78
Code	9
Code Umsetzer	110
Codierer	111
Codierung	9
Daten	16
Daten, digitale	17
Datenfluß	62
Datenflußplan	63
Datenträger	18
Datenverarbeitungsanlage	120
Datenverarbeitungsanlage, digitale	121
Datenverarbeitungssystem	90
Datenverarbeitungssystem, digitales	91
Decodierer	112
Dezimalziffer	24
Dezimalziffern, Binärkode für	25
Dialogbetrieb	157
digitale Daten	17
digitale Datenverarbeitungsanlage	121
digitales Datenverarbeitungssystem	91
digitale Rechenanlage	121
digitales Rechensystem	91
digitales Signal	22
Disjunktion	(Tabelle 3)
Dualziffer	24
Duodezimalziffer	24
EA-Prozessor	98
EA-Werk	98
Eingabe, manuelle	126
Eingabeeinheit	102
Eingabegerät	124
Eingabewerk	96
Einheit, periphere	101
Ergänzungsspeicher	135
Exponent	30
Fehlererkennungscode	11
Fehlererkennungscode, binärer	13
Fehlerkorrekturcode	12
Fehlerkorrekturcode, binärer	14
Festpunkt-Rechnung	31
Festspeicher	137
Flipflop	82
formaler Prozedur-Parameter	36
Funktionseinheit	89
Gatter siehe Verknüpfungsglied	136
geschützter Speicherbereich	136
Gleitpunkt-Rechnung	32
halbilogarithmische Schreibweise	30
Haupspeicher	134
Implikation	(Tabelle 3)
Indexregister	144
Inhibition	(Tabelle 3)
interpretieren	56
Interpretierer	56
Kellerspeicher	145
Komma siehe Radixpunkt	54
Kompilierer	54
komplizieren	54
Konjunktion	(Tabelle 3)
Kurvenleser	125
Kurvenschreiber	128
Leitdaten	16
Leitwerk	94
logisch siehe binär und boolesch	1
Mantisse	30
manuelle Eingabe	126
Maschinendresse	45
maschinenorientierte Programmiersprache	41
Maschinensprache	42
Maschinenswort	132
Mehrprogrammbetrieb	152
Mehrprozessorsystem	106
Mehrrechnersystem	107
Merker	139
Modulo-N-Zähler	88
Multiplexbetrieb	148
Multiplexer	115

	Nr
NAND-Glied	83
NAND-Verknüpfung	(Tabelle 3)
n-Bit-Zeichen	10
Negation	(Tabelle 2)
NICHT-Glied	83
NOR-Glied	83
NOR-Verknüpfung	(Tabelle 3)
Nutzdaten	16
ODER-Glied	83
ODER-Verknüpfung	(Tabelle 3)
Oktalziffer	24
Operandenteil	49
Operationscode	43
Operationsteil	47
Parallel-Serie-Umsetzer	114
Parallelbetrieb	150
periphere Einheit	101
peripherer Speicher	104
problemorientierte Programmiersprache	40
Programm	38
Programmablauf	64
Programmablaufplan	65
Programmiersprache	38
Programmiersprache, maschinenorientierte	41
Programmiersprache, problemorientierte	40
Programmiersystem	59
Prozedur	37
Prozedur-Anweisung	37
Prozedur-Parameter	36
Prozedur-Parameter, aktueller	36
Prozedur-Parameter, formaler	36
Prozedur-Vereinbarung	36
Prozessor	95
Pseudodezimale	26
Puffer	138
Radixpunkt	29
Radixschreibweise	28
Randomspeicher	147
Reaktionszeit	158
Realzeitbetrieb	156
Rechenanlage	120
Rechenanlage, digitale	121
Rechensystem	90
Rechensystem, digitales	91
Rechenwerk	93
Rechner	100
Register	140
Sammlung	74
Schaltfunktion	77
Schaltglied	80
Schaltkreis siehe Schaltnetz	
Schaltnetz	84
Schaltvariable	76
Schaltwerk	79
Schleife	72
Schreibweise, halblogarithmische	136
Sedezimalziffer	24
sequentieller Betrieb	150
Serie-Parallel-Umsetzer	113
Sichtanzeigegerät	129
Signal	19
Signal, analoges	21
Signal, digitales	22
Signalparameter	20
Simulator	58
Simulierer	57
Simultanbetrieb	151
Speicher	92
Speicher, peripherer	104
Speicher mit Wortstruktur	132
Speicherbereich, geschützter	136
Speicherlement	130
Speicherglied	81
Speicherstelle	131
Speicherzelle	133
Stackbetrieb	155
Stelle	8
Stellenschreibweise	27
Strecke	66
Symbol	15
Synchronisationsschnitt	75
Taktgeber	122
Teilnehmer Rechensystem	108
Übersetzer	53
übersetzen	53
Übersetzungsanweisung	61
Uhr	116
Umsetzer	109
UND-Glied	83
UND-Verknüpfung	(Tabelle 3)
Ursprungsanweisung	53
Ursprungsprogramm	53
Ursprungssprache	53
Vereinbarung	35
Verknüpfung	77
Verknüpfungsglied	83
Verzweigung	70
Weg	67
Wort	7
Zähleinheit	86
Zähler	86
Zahlenwert	24
Zeichen	1
Zeichenvorrat	1
Zentraleinheit	100
Zentralspeicher	99
Zielanweisung	53
Zielprogramm	53
Zielsprache	53
Ziffer	24
Zugriffszeit	118
Zusammenführung	71
Zweig	68
Zweig, abgeschlossener	69
Zweirichtungszähler	87
Zykluszeit	117

Erläuterungen

1. Entstehung und Zielsetzung

Im Jahre 1957 veröffentlichte die Fachgruppe 6 „Informationsverarbeitung“ der Nachrichtentechnischen Gesellschaft im VDE als NTG-Empfehlung 0601 einige wichtige Begriffe aus dem Gebiet der Informationsverarbeitung. Sie wurden im Mai 1958 als Vornorm DIN 44 300 „Programmgesteuerte elektronische Rechenanlagen, Begriffe“ in das deutsche Normenwerk übernommen.

Nach der Gründung des Fachnormenausschusses Informationsverarbeitung (FNI) nahm dessen Arbeitsausschuß 1 „Begriffe“ die Arbeit an der grundlegenden, für die Ausschüsse des FNI verbindlichen Terminologie auf. Insbesondere zur Verständigung der anderen Ausschüsse untereinander mußte rasch eine Auswahl wichtiger Begriffe behandelt werden. Neben der alten Vornorm konnten die Ergebnisse der derzeitigen internationalen Bemühungen um die Terminologie der Informationsverarbeitung in der International Federation for Information Processing (IFIP Technical Committee on Terminology) herangezogen werden. Das Ergebnis mehrjähriger Arbeit wurde im April 1965 als Norm DIN 44 300 der Öffentlichkeit übergeben.

In der Zwischenzeit wurde das Ziel weiterverfolgt, eine Grundterminologie zu erarbeiten, die den einzelnen Ausschüssen einerseits genügend Anhaltspunkte für die Ausarbeitung ihrer spezifischen Terminologie gibt, andererseits aber auch hinreichend Freiheit läßt. So konnte darauf verzichtet werden, eine Vollständigkeit anzustreben, wie sie für Wörterbücher angezeigt sein mag; jedoch sollten Lücken ausgefüllt werden, wo die Weiterentwicklung der Informationsverarbeitung zur Klärung noch offenstehender Fragen geführt hatte. Aus der Fülle der inzwischen entstandenen Begriffe konnte eine größere Anzahl, die von allgemeiner Bedeutung und klar abgegrenzt waren, übernommen werden. Da die vorliegende Neuauflage von DIN 44 300 als eine erweiterte Fassung der Ausgabe von 1965 zu verstehen ist, wurden die dort aufgeführten Benennungen und Begriffe, soweit es irgendmöglich war, unverändert gelassen. Die damit verbundene Erschwerung seiner Arbeit hat der Ausschuß im Interesse der Kontinuität auf sich genommen.

2. Zum Aufbau der Norm

Für die terminologische Arbeit auf dem Gebiet der Informationsverarbeitung stellten sich ein paar bemerkenswerte Grundsätze heraus, die der Einfachheit und Klarheit des Aufbaus dienen. Hier sind zu nennen der Grundsatz, Grenzfälle mit einzubeziehen (z. B. Nr 7 **Wort**) und der Grundsatz, den rekursiven Gebrauch von Begriffen nirgends zu verhindern (z. B. Nr 33 **Anweisung**: Eine Anweisung kann Teile enthalten, die wiederum Anweisungen sind). Mit dem rekursiven Gebrauch von Begriffen geht einher eine grundsätzliche Relativierung aller Begriffe, z. B. kann ein Rechenwerk, Teil eines Rechensystems, seinerseits einen Teil enthalten, der selbst ein (verkümmertes) Rechensystem ist, z. B. eine Multipliziereinheit.

Bemerk soll auch noch werden, daß aus Gründen der Textvereinfachung der Plural stets nicht nur eine aus zwei oder mehreren Elementen bestehende Menge, sondern auch eine einelementige Menge bezeichnen kann.

Bei der Abgrenzung der Begriffsbestimmungen und der Wahl der Benennungen wurden die sich auf internationaler Ebene abzeichnenden Klärungen berücksichtigt. Bei den Benennungen wurde Wortstämme finden, die sich in üblicher Form auch in anderen Sprachen finden, der Vorzug gegeben, wenn sich auch die damit verbundenen Begriffe decken; andernfalls wurden, um Schwierigkeiten bei der Übersetzung zu verhüten, solche Wortstämme nicht verwendet.

Für einige sehr allgemeine Begriffe, wie Information, Nachricht, digital, analog, werden hier keine Bestimmungen gegeben. Die Ausgabe der Norm konnte nicht so lange verzögert werden, bis ein Abgleich des verschiedenartigen Gebrauchs in verwandten Gebieten und in anderen Sprachen erreicht ist. „Information“ wird im semantischen Sinn der Umgangssprache gebraucht als „Kenntnis über Tatsachen, Ereignisse, Abläufe usw.“. „Nachricht“ ist gegen „Information“ (noch) nicht abgegrenzt und wird in dieser Norm nicht benutzt, siehe jedoch DIN 44 301, Begriffe der Informationstheorie (Shannonsche Nachricht). „Digital“ tritt hier nur im Zusammenhang mit Nr 17 **digitale Daten**, Nr 22 **digitales Signal**, Nr 91 **digitales Rechensystem** und Nr 121 **digitale Rechenanlage** auf und ist dabei auf Nr 9 **Code**, Nr 76 **Schaltvariable** und Nr 79 **Schaltwerk** abgestützt. Eine Abgrenzung „analog“ gegen „digital“ erfolgte nur im Zusammenhang mit „Signal“ (Nr 21, 22).

Zusammengehörige Begriffe dieser Norm unterliegen meist einer hierarchischen Ordnung. Das Muster dieser Ordnung ist nicht absolut zu verstehen, sondern relativ zu einer Betrachtungsebene (im Sinne von Auflösungsvermögen), worauf verschiedentlich mit „in einem gegebenen Zusammenhang“ o. ä. hingewiesen wird.

Bei der Auswahl der Begriffe und ihrer Bestimmungen stand die funktionelle Betrachtungsweise (im Unterschied zur konstruktiven, geräteorientierten) im Vordergrund (vergleiche Nr 89 **Funktionseinheit** gegen Nr 119 **Baueinheit**), da sie vom mehr oder weniger zufälligen, technisch-physikalischen Aufbau unabhängig ist. Gleichwohl wurde nicht auf gerätebedingte Begriffe verzichtet (Nr 119 bis 129), jedoch entspricht in dieser Norm nicht jedem funktionell definierten Begriff ein gerätemäßig definierter.

Der funktionelle Standpunkt hat ferner den Vorteil, daß durch ihn die Entscheidung, ob etwas durch „hardware“ oder durch „software“ realisiert wird, offen bleibt. Benennungen für Funktionseinheiten können selbstverständlich auch für genau korrespondierende Baueinheiten verwendet werden.

3. Zu den einzelnen Abschnitten

Datendarstellung

Das **Zeichen** (Nr 1) soll nicht nur als konkretes Ideogramm verstanden werden, sondern auch als abstrakter Inhalt eines solchen Ideogramms oder einer technischen Realisierung, wobei es lediglich auf Unterscheidungsmerkmale ankommt. Somit sind auch die Werte, die eine **Schaltvariable** (Nr 76) annimmt, im abstrakten Sinne lediglich Zeichen.

Auf den Unterschied zu Nr 15 **Symbol** wird hingewiesen. Zum Beispiel wird das Zeichen „Liegendes Kreuz“ üblicherweise als Symbol für Multiplikation verstanden, aber auch das Zeichen „Stern“ oder das Zeichen „Punkt auf halber Höhe“. Oft werden Zeichen gebraucht, ohne daß sie Symbolcharakter haben; manchmal dient dasselbe Zeichen unter verschiedenen Gesichtspunkten zur Darstellung verschiedener Symbole.

Für Nr 11 **Fehlererkennungscode** und Nr 12 **Fehlerkorrekturcode** sind auch noch die Benennungen „prüfbarer Code“ bzw. „korrigierbarer Code“ im Gespräch. Zu den neuen Begriffen dieses Kapitels gehören auch Nr 19 **Signal** und Nr 20 **Signalparameter**.

Zahlendarstellung

Bei den **Binärcodes für Dezimalziffern** (Nr 25 und Tabelle 1) wurden die Benennungen, wo es angängig war,

nach den Wertigkeiten der einzelnen Stellen gewählt, z. B. ist im 2·4·2·1-Code LLO der Ziffer 8 zugeordnet, die sich als $2 \cdot 1 + 4 \cdot 1 + 2 \cdot 1 + 1 \cdot 0$ ergibt. Man beachte jedoch, daß sich aus den Wertigkeiten die Codierung nicht immer eindeutig ergibt.

Programmiersprache

In Nr 33 Anweisung und später steht Arbeitsvorschrift kurz für Vorschrift zu einer bestimmten Informationsverarbeitung.

Programmstruktur

Um unnötige Begriffe zu vermeiden, wurde über die Art der Darstellung in der Bestimmung zu Nr 65 Programmablaufplan nichts gesagt, insbesondere wurde die Darstellung nicht auf Begriffe der Graphentheorie abgestützt. Auch ein gut gegliedertes Programm in einer geeigneten Programmiersprache kann als Programmablaufplan angesehen werden.

Die Begriffe Nr 70 Verzweigung und Nr 71 Zusammenführung beziehen sich auf eine solche Programmstruktur, bei der eine echte Alternative vorliegt, d. h. wo fallweise einer von mehreren möglichen Zweigen wirklich durchlaufen wird. Die Begriffe Nr 73 Aufspaltung und Nr 74 Sammlung hingegen beziehen sich auf eine Programmstruktur, bei der eine Anzahl parallellaufender Zweige vorliegen, von denen alle oder einige gleichzeitig (im Sinne einer Parallelverarbeitung) durchlaufen werden.

Nr 75 Synchronisationsschnitt dient ebenfalls der Parallelverarbeitung, wobei die Abstimmung z. B. darin bestehen kann, daß Weg A und Weg B aufeinander warten oder daß Weg A auf Weg B wartet, nicht aber umgekehrt.

Schaltwerke

Das Wort „logisch“ wird wegen seiner breiten Verwendung in der Umgangssprache und in anderen Fachgebieten ausdrücklich vermieden (siehe Nr 4 und 78). Im technischen Bereich kennzeichnet der Hinweis auf die Schalteigenschaft die Beschränkung auf endlich viele Werte: Nr 76 Schaltvariable, Nr 77 Schaltfunktion, Nr 80 Schaltglied usw., entsprechend dem Gebrauch des englischen „switch“, z. B. in switching theory. Das allgemeinste, auch Speicherglieder umfassende Schaltgebilde heißt Nr 79 Schaltwerk, wobei „Werk“ das Durchlaufen verschiedener innerer Zustände andeuten soll. So sind bei den digitalen Rechensystemen auch die Benennungen Nr 93 Rechenwerk, Nr 94 Leitwerk, Nr 96 Eingabewerk, Nr 97 Ausgabewerk gebildet.

Die Benennungen für boolesche Verknüpfungen in den Tabellen 2 und 3 sollen einer künftigen Normung von Verknüpfungen der mathematischen Logik nicht vorgreifen.

Funktioneller Aufbau digitaler Rechensysteme

Zum Bilden von Benennungen für Funktionseinheiten sei auf die Möglichkeit von Zusammensetzungen mit den in Nr 89 gegebenen „-system“, „-werk“, „-glied“, „-element“ hingewiesen. Die für Baueinheiten vorgesehenen „-anlage“, „-gerät“, „-teil“ sollen für Funktionseinheiten nicht benutzt werden.

Die Benennungen Nr 90 Rechensystem und Datenverarbeitungssystem stehen nebeneinander. Der Wortstamm „Rechen“ wird zwar vielfach im verallgemeinerten Sinne gebraucht, indem unter Rechenoperationen auch boolesche Verknüpfungen, Vergleichen usw. verstanden werden, vergleiche bei Nr 93 Rechenwerk. Jedoch ist die Verallgemeinerung dort nicht sehr eingängig, wo die arithmetischen Operationen gegenüber anderen stark zurücktreten oder ganz entfallen, wie in großen Bereichen der kaufmännischen Datenverarbeitung oder bei der Dokumentation.

Konstruktiver Aufbau digitaler Rechenanlagen

Zum Bilden von Benennungen für Baueinheiten sei auf die Möglichkeit von Zusammensetzungen mit den in Nr 119 gegebenen „-anlage“, „-gerät“, „-teil“ hingewiesen. Die für Funktionseinheiten vorgesehenen „-system“, „-werk“, „-glied“, „-element“ können zum Bilden von Benennungen von Baueinheiten benutzt werden, wenn diese mit Funktionseinheiten korrespondieren.

Speicher

Nach der funktionellen Definition unter Nr 92 wurde hier die Einteilung sowohl nach der Datendarstellung als auch nach der technischen Zugangsmöglichkeit getroffen. Die Speicherstelle (Nr 131) nimmt ein Zeichen auf, was bei Zahlen vielfach auch der Ziffernstelle entsprechen wird. Wenn von der Unterteilung eines Speichers gesprochen wird, heißt der im gerade vorliegenden Zusammenhang kleinste Teil Speicherelement (Nr 130).

Die technisch bedingte Unterteilung bei Nr 132 Speicher mit Wortstruktur gestaltet, Nr 132 Maschinenwort, Nr 133 Speicherzelle und Nr 45 Maschinenadresse ohne Abstützung auf den Befehlsaufbau einzuführen.

Nr 140 Register wird in seiner Doppelfunktion als Speicher und als integraler Bestandteil eines Werkes gesehen.

Betriebsarten

„Aufgabe“ ist in den Begriffsbestimmungen dieses Kapitels in derselben Bedeutung gebraucht wie in Nr 38.

F. L. Bauer