K o m m u n i k a t i o n
m i t
A u t o m a t e n

Von der Fakultät für Mathematik und Physik
der Technischen Hochschule Darmstadt

zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer.nat.)

genehmigte
Dissertation

vorgelegt von

C a r l A d a m P e t r i

aus Leipzig

Referent: Prof.Dr.rer.techn.A.Walther
Korreferent: Prof.Dr.Ing.H.Unger

Tag der Einreichung: 27.7.1961
Tag der mündlichen Prüfung: 20.6.1962

D 17

Bonn 1962

Eingegangen am 27. 7. 1961

Anschrift des Verfassers: Institut für Angewandte Mathematik
der Universität Bonn, Wegelerstr. 10

Als Manuskript gedruckt im Mathematischen Institut der Universität
Bonn, Wegelerstr. 10

Inhaltsverzeichnis

In Klammern eingeschlossene Zahlenangaben verweisen auf eine laufende Nummer im Literaturverzeichnis sowie auf die Nummer der ersten Seite einer Literaturstelle.

L i t e r a t u r :

1. Automata Studies.
   Ed.: C.E. Shannon und I. McCarthy.
   Princeton University Press 1956.

2. The Bell System Technical Journal 1960.

3. Journal of the Association for Computing Machinery, Apr. 1958 Vol 5, Nr. 2.

4. Bulletin of the Research Council of Israel, 9F, 1960.

5. Computability and Unsolvability.
   Martin Davis. McGraw - Hill 1958.

6. Finite combinatory Processes.
   Emil L. Post 1936 , Journal of Symbolic Logic 1, 103.

7. Journal of the Association for Computing Machinery, 1957, Vol. 4.

8. Proceedings of IRE, 1953, 41

9. The Bell System Technical Journal, 1955, 34.

10. IRE Transactions on Electronic Computers 1960, Vol. EC-9, Nr. 1.

11. Communications of the ACM, Vol. 3, Nr. 4, Apr. 1960.

12. Combinatory Logic. H.B. Curry, R. Feys, W. Craig. 1958.
    North Holland P.C.

13. Self - Organizing Systems. Yovits u. Cameron. Pergamon 1960.

14. Proceedings of an International Symposium on the Theory of Switching. Harvard University Press. 1959. Part I.

15. Übersetzung algorithmischer Formelsprachen in die Programmsprache für Turingmaschinen. H. Bottenbruch. Diss. Darmstadt 1957.

16. Symbolische Logik. R. Carnap. Wien 1960 .

17. Operational Philosophy. A. Rapoport, Harter, New York 1953.

18. Logical Foundations of Probability. R. Carnap, Chicago 1950.

19. IBM Journal of Research and Development. Vol. 4, No. 2, 1960.

§1.

Diese Arbeit befaßt sich mit den begrifflichen Grundlagen einer Theorie der Kommunikation. Die Aufgabe dieser Theorie soll es sein, möglichst viele Erscheinungen bei der Informationsübertragung und Informationswandlung in einheitlicher und exakter Weise zu beschreiben.

Dem Begriff der Kommunikation kann eine Schlüsselstellung im Gebäude des wissenschaftlichen Denkens zugeschrieben werden. Daher ist es erforderlich, an die Darstellungsmittel und auch an die Beweismittel die strengsten Forderungen der tatsächlichen Konstruierbarkeit zu stellen.

Andererseits bestehen zwei unmittelbare Anwendungen einer solchen Theorie in Methoden zum Entwurf und zur Programmierung von Informationsmaschinen. Deshalb sollten sich die Begriffsbildungen der Theorie nicht zu weit von den auf diesen Gebieten bereits eingeführten Begriffen entfernen, die ja ihren Nutzen bereits unter Beweis gestellt haben.

Die hier vorgelegte Darstellung unterscheidet sich jeweils in mindestens einem aus folgenden wesentlichen Punkten von den zur Zeit bekannten Theorien, die die Kommunikation betreffen:

1.) Das Vorhandensein einer Metrik wird weder für Zeit, noch Raum, noch für physikalische Zustandsgrößen vorausgesetzt.

2.) Die Zeit wird als nur lokale Relation zwischen Zuständen eingeführt.

3.) Die Objekte der Theorie sind diskret und werden nur mit streng finiten Mitteln verknüpft und erzeugt.

Nun mag die grundsätzliche methodische Bedeutung dieser Vorgehensweise den Praktiker wenig interessieren; man muß aber darauf hinweisen, daß diese Arbeit mit dem ausdrücklichen Ziel der Beseitigung praktischer Schwierigkeiten begonnen wurde, und daß dieses Ziel in einem noch zu beschreibenden Umfang erreicht wird.

Der Schwerpunkt der Arbeit soll im Auffinden zweckmäßiger Begriffsbildungen für eine exakte Theorie der Kommunikation liegen, nicht so sehr im mathematischen Ausbau dieser Theorie, wie er sich in einer Aufzählung von Theoremen ausdrücken würde. Wegen der axiomatischen Methode, mit der die Begriffe in §4 eingeführt werden, ist nämlich der Übergang zur formalen Logik, Algebra und Topologie sofort möglich, und dort vorliegende Theoreme lassen sich anwenden.

Von praktischem Interesse können folgende naheliegenden Schlüsse aus den Ergebnissen der Arbeit werden:

1.) Die Toleranzansprüche an die metrischen Eigenschaften von Bauelementen für Informationsmaschinen lassen sich bei geeigneter Struktur der Maschinen wesentlich abschwächen.

2.) Es ist möglich, Informationsmaschinen strukturell so anzulegen, daß sie asynchron, in allen Teilen parallel arbeitend, beliebig erweiterungsfähig sind.

3.) Die Theorie liefert ein Darstellungsmittel für komplizierte organisatorische Vorgänge beliebiger Art, das bei gleicher Strenge und Einfachheit mehr leistet als die vorhandene Theorie der synchronen Automaten.

Nun wird man nicht erwarten, daß solche Vorzüge ganz ohne Kosten geliefert werden. Diese Kosten bestehen aber hier nicht in einer zu hohen Abstraktion, die die Anwendbarkeit beeinträchtigen würde, sondern darin, daß vom Benutzer der Theorie eine gewisse Umwertung der Begriffe bezüglich ihrer Einfachheit verlangt wird. Begriffe wie "Zeitintervall", "Messung", "Zustand eines Systems" bilden keineswegs die Grundbausteine der Theorie, sondern es soll ihre wahre Komplexität gerade erst beschrieben werden; "wahr" in bezug auf die exakten, der Beobachtung und der Steuerung zugänglichen Kommunikationsverhältnisse, die z. B. bei dem Vorgang "Messung eines Zeitintervalls" bestehen.

Wir stellen nicht die Frage, welche Begriffe in Wahrheit die einfachsten sind. Nur die erfolgreiche Anwendung der auf dem vorgelegten mathematischen Modell aufzubauenden Theorie wird als

Rechtfertigung der Begriffseinführungen beansprucht werden.

Es wird nicht behauptet, daß es notwendig sei, genau die Begriffe von §4 zu bilden, um zum Ziel zu gelangen; es wird nicht behauptet, daß der Inhalt dieser Arbeit eine formal abgeschlossene Theorie darstellt.

Im Rahmen des Bekannten läßt sich die Vorgehensweise etwa so darstellen: Die Ausbreitung physikalischer Wirkungen wird vom Standpunkt der kombinatorischen Topologie aus untersucht und als Schaltlogik von extrem asynchronen Automaten interpretiert.

§2.

In der Theorie der Automaten sind eine Anzahl von Modellen informationsverarbeitender Maschinen vorgelegt worden. Diesen abstrakten Modellen ist folgendes gemeinsam:

Man geht aus von der Fiktion, daß es stets sinnvoll sei, vom Gesamtzustand Z eines Systems S zu einem Zeitpunkt t zu reden. Ein Vorgang in S wird dann durch eine Funktion f beschrieben: $Z = f(t)$. Der Definitionsbereich von f ist die Menge der reellen Zahlen oder eine Untermenge von dieser.

Auch die klassische Physik nimmt eindeutig diesen Standpunkt ein. Die Zweckmäßigkeit dieser Betrachtungsweise wurde erstmals durch die Relativitätstheorie in Frage gestellt.

Wenn wir diese Fiktion also nicht zur Grundlage unserer Betrachtung machen, entfernen wir uns sehr weit von der gewohnten Denkweise. Daher erscheint es nötig, die Beschränkungen zu schildern, die diese Fiktion den abstrakten Modellen auferlegt; weiterhin ist es nötig, nachzuweisen, daß sich alle tatsächlich verwendbaren Ergebnisse der Automatentheorie zwanglos durch die neu einzuführenden Begriffe ausdrücken lassen und somit verwendbar bleiben; und schließlich ist zu zeigen, daß die neue Vorgehensweise greifbare Vorteile hat.

Wir beginnen mit der Schilderung der Schwierigkeiten, die sich ergeben haben, wenn die abstrakten Modelle der Automatentheorie mit der Wirklichkeit konfrontiert werden.

Wir werden folgendermaßen argumentieren:

1.) Wir erkennen folgende Postulate an:
Es gibt eine obere Grenze für die Geschwindigkeit von Signalen.
Es gibt eine obere Grenze für die räumliche Dichte von Information.

2.) Automaten von fester endlicher Größe können höchstens iterativ erklärte Klassen von Eingangsfolgen wahrnehmen. (1,3; 3,181)

3.) Rekursiv erklärte Klassen von Eingangsfolgen, die nicht höchstens iterativ dargestellt werden können, können nur von Automaten unbeschränkter Größe wahrgenommen werden.

4.) Damit ein Automat eine lösbare rekursive Aufgabe lösen kann, muß die Möglichkeit gegeben sein, daß er "auf Vorrat" oder "bei Bedarf" unbeschränkt erweitert werden kann.

5.) Automaten, die im Sinne der Automatentheorie konzipiert werden, geraten nach endlich vielen Erweiterungsschritten in Widerspruch zu mindestens einem der genannten Postulate.

Zusammengefaßt: Die Automatentheorie kann den physikalisch-reellen Informationsfluß bei der Lösung einer rekursiven Aufgabe nicht darstellen.

Bevor wir die Argumente ausführen, wollen wir ihre Zusammenfassung in der Sprache der technischen Vorgänge beim Bau einer Rechenanlage interpretieren, um das Ziel der Überlegungen zu verdeutlichen.

Der logische Plan einer Anlage schreibt eine endliche Anzahl von Zuständen vor, die zu bestimmten Zeitpunkten oder innerhalb bestimmter Zeitintervalle angenommen werden müssen.

Den logischen Zuständen müssen physikalisch unterscheidbare Zustände der Maschine zugeordnet werden, die als getrennt liegende Gebiete eines Phasenraumes angesehen werden können.

Wir versuchen nun, auf dieser Maschine eine rekursive Aufgabe zu lösen, wir wissen also nichts über den Speicherbedarf der Maschine. Wir gelangen nun entweder zur Lösung oder aber in ein Stadium des Lösungsvorganges, in dem die Maschine weiteren Speicherbedarf anmeldet. Wenn wir nun nicht den Lösungsversuch ohne jeden Nutzen abbrechen wollen, haben wir zwei Möglichkeiten:

Entweder veranlassen wir die Maschine zur Ausgabe von Information auf materiellen Datenträgern, mit der Möglichkeit der Wiedereingabe, und wir manipulieren diese Datenträger so, daß sie in bezug auf die Maschine einen externen Speicher darstellen. Dann aber löst die Maschine selbst nur einen trivialen, nämlich höchstens iterativen Teil der rekursiven Aufgabe;

oder aber, wir bauen eine weitere Maschine, die die Manipulation der Datenträger ersetzt (vergrößerter Speicher) oder für uns durchführt. Wenn diese Maschine wiederum durch einen logischen Plan beschrieben wird, werden im Phasenraum bei diesem Erweiterungsschritt endlich viele Gebiete zugefügt. Es entsteht nun das Problem, beide Maschinen in ihrer Wirkungsweise zu koordinieren. Das aber ist im allgemeinen nicht möglich, ohne die ursprüngliche Maschine technisch zu verändern, d.h. praktisch: neu zu bauen. Denn nehmen wir an, daß die zeitliche Trennung der Zustände von einem Taktgeber mit bestimmter Grundfrequenz definiert wird, so können die nach der Erweiterung möglichen Signallaufzeiten die Taktzeit überschreiten; jedenfalls tun sie dies nach endlich vielen Erweiterungsschritten. Wir müssen also die Grundfrequenz herabsetzen, und nach endlich vielen Herabsetzungen müssen wir die metrischen Eigenschaften der Bauelemente ändern und an die neue Grundfrequenz anpassen. (Für diejenigen Kopplungsarten, die keine obere Schranke für die Taktzeit vorschreiben, ist die Anpassung nötig, um die Zahl der Fehlfunktionen pro Takt beschränkt zu halten.)

Ein Analogon zu diesen Verhältnissen stellt die Speicheradressierung bei fester Wortlänge dar: Ein Speicher ist bezüglich der Adressierung nicht unbeschränkt erweiterbar; Erweitern wir einen Speicher, so kommen wir nach endlich vielen Schritten an einen Punkt, wo das Wort die zur Kennzeichnung der Adressen nötige Information nicht mehr aufnehmen kann; wir müssen dann zu anderen Zugriffsprinzipien übergehen oder aber die Wortlänge erhöhen, und damit die Struktur des Speichers von Grund auf ändern.

Wir behandeln diese Schwierigkeiten als logisch-strukturelles Problem der Kommunikation, mit dem Ziel, die Kommunikationsform so zu verallgemeinern, daß die Schwierigkeiten entfallen.

Wir folgen nun den einzelnen Punkten der oben skizzierten Argumentation.

1.) Wir verzichten auf den Versuch, die Existenz von Schranken für Signalgeschwindigkeit und Informationsdichte aus einfacheren Annahmen abzuleiten, und führen also Postulate ein. Es wird eingeräumt, daß die Argumentation hinfällig wird, wenn ein Verfahren gefunden wird, Signalgeschwindigkeiten oder Informationsdichten beliebig zu erhöhen. Als Erläuterung sei angefügt, daß wir in der Lichtgeschwindigkeit eine obere Schranke für alle Signalgeschwindigkeiten sehen. Das zweite Postulat soll eine grundsätzliche Unmöglichkeit ausdrücken, ein physikalisches Objekt von endlichem Volumen herzustellen, welches wir als Speicher für beliebige Informationsmengen benutzen können. Vielleicht gewinnt das Postulat an Plausibilität, wenn wir es mit dem Quantenpostulat in Verbindung bringen.
Kennzeichnen wir z.B. die gespeicherte Information durch die Impulskoordinaten einer gegebenen Menge von materiellen Teilchen, so

schreibt die zu speichernde Informationsmenge eine bestimmte Meßgenauigkeit für die Impulskoordinaten vor;
Bei steigender Informationsmenge müssen die Impulskoordinaten immer schärfer festgelegt werden, und nach dem Quantenprinzip übersteigt dann die Unschärfe der Lagekoordinaten jede Schranke. Daher müssen wir das Speichervolumen, nämlich dasjenige räumliche Gebiet, in dem wir die Information vor Störung zu schützen haben, ebenfalls unbeschränkt vergrößern.

2.) Wir sprechen zunächst nur von digitalen, deterministischen, synchronen Automaten. Für diese geht man von der Vorstellung aus, daß sie zu bestimmten Zeiten t_i, $i \varepsilon \{\ldots -2, -1, 0, 1, 2 \ldots\}$ wohldefinierte Zustände annehmen; der Zustand zur Zeit t_{i+1} und das Ausgabesignal zur Zeit t_i seien eindeutige Funktionen von dem Zustand zur Zeit t_i und dem Eingabesignal zur Zeit t_i. Der vorgegebene Wertebereich der Signale sei endlich.

Wenn die Menge der möglichen Zustände vorgegeben und endlich ist, sprechen wir von einem endlichen Automaten.

Automaten lassen sich nun in verschiedener Weise charakterisieren (2, 1267):

a) durch die Angabe der Verknüpfung zwischen Signalen und Zuständen zu den Zeiten t_i und t_{i+1} für alle i;
b) durch ihre Programmierung;
c) durch die Struktur ihrer Zusammensetzung aus idealisierten Bauteilen;
d) durch die Abbildung der Eingabefolgen auf die Ausgabefolgen.

Zu unserem Zweck ist besonders die letzte Art der Charakterisierung wesentlich. Wir können uns fragen, welche Eigenschaften von Eingangssignalfolgen von einem gegebenen Automaten überhaupt wahrgenommen werden können. Als wahrnehmbar gilt eine Eigenschaft, wenn sie eindeutig einer Eigenschaft eines einzelnen Ausgabesignals zugeordnet werden kann.

Sei $X = \{a, b, c, \ldots\}$ die endliche Menge der Eingabesignale

$Y = \{\alpha, \beta, \gamma, \ldots\}$ die endliche Menge der Ausgabesignale,

PY die Menge der Untermengen von Y.

Die Untermengen von Y, also die Elemente von PY, charakterisieren alle möglichen Eigenschaften eines einzelnen Ausgabesignals. Z.B. bezeichnet die Untermenge $\{\alpha, \beta\}$ die Eigenschaft des Signals, entweder α oder auch β zu sein. Die Eigenschaften bilden einen Booleschen Verband.

Nun wollen wir gewisse Eigenschaften von Eingangssignalfolgen charakterisieren. Gemäß unserer Aufgabe, die Kommunikationsverhältnisse zu untersuchen, fassen wir die Eingangsfolgen als Mitteilungen an den Automaten auf. Da PY nur endlich viele Elemente hat, kann der Automat höchstens die gleiche Anzahl von Eigenschaften an Mitteilungen wahrnehmen.
Die Menge M der Mitteilungen bildet bezüglich der Verkettung eine freie Halbgruppe mit Einselement und freiem Erzeugendensystem X , welches wir als Alphabet einer (formalen) Sprache ansehen wollen. Da wir natürlich nur formale Eigenschaften von Mitteilungen betrachten können, definieren wir die Ausdrücke dieser Sprache rein syntaktisch. Zur Darstellung der Systax bedienen wir uns einer formalen Metasprache, deren Alphabet alle Elemente von X , also a, b, c... enthält und außerdem die Zeichen
∇ / $*$ $\emptyset$

Ein Ausdruck der Metasprache bezeichnet eine Menge E von Eingangsfolgen. Wir wählen die Metasprache so, daß ihre Ausdrücke selbst einer möglichst einfachen Syntax (Simple Phrase Structure Grammar) angehören (Chomsky; Bar-Hillel: 4, 1) oder allgemeiner, daß sie Formeln eines

möglichst einfachen kombinatorischen Systems sind (Post , 6).
In der Definition kombinatorischer Systeme folgen wir Davis (5 , 81): Ein Wort ist eine endliche Folge von Symbolen. Auch die leere Folge ist ein Wort. Eine Produktion ist ein zweistelliges Prädikat zwischen Worten und ist gekennzeichnet durch ein geordnetes Sextupel von Worten g, h, k, $\bar{g}$, $\bar{h}$, $\bar{k}$.
Für ein gegebenes geordnetes Paar von Worten F, G ist das Prädikat wahr dann und nur dann, wenn es zwei Worte P, Q gibt sodaß

$$F = g P h Q k \quad \text{und} \quad G = \bar{g} P \bar{h} Q \bar{k} \text{ ist.}$$

G heißt dann ein Nachfolger von F in bezug auf die Produktion. Ein kombinatorisches System besteht aus genau einem nichtleeren Wort, genannt das Axiom des Systems, und einer endlichen Menge von Produktionen.
Das Alphabet eines kombinatorischen Systems besteht aus allen Symbolformen, die im Axiom oder in den Worten auftreten, die die Produktionen bestimmen.
Eine Ableitung in einem kombinatorischen System ist eine endliche Folge von Worten $F_1, F_2, \ldots F_m$ derart, daß F_1 das Axiom des Systems ist und für alle i mit $1 < i \leq m$ gilt: F_i ist ein Nachfolger von F_{i-1} in bezug auf eine der Produktionen.

Ein Wort ist ein Satz des kombinatorischen Systems, wenn es das letzte Wort einer Ableitung ist.

Eine Formel ist ein Satz, der keinen Nachfolger hat (in bezug auf die Produktionen eines vorgegebenen kombinatorischen Systems).
Speziell sollen unsere Ausdrücke zur Beschreibung von Eingangsfolgen Formeln des folgenden kombinatorischen Systems KS1 sein:

Axiom: Σ

Produktionen: $g, k, \bar{g}, \bar{k}$ sind leer,

$h = \Sigma$

Die Produktionen sind dann nur in $\bar{h}$ verschieden, und das Auffinden eines Nachfolgers besteht in einer Substitution

$$\Sigma \leftarrow \bar{h}$$

Demgemäß beschreiben wir die Produktionen:

$$\Sigma \leftarrow \nabla \Sigma \Sigma$$

$$\Sigma \leftarrow / \Sigma \Sigma$$

$$\Sigma \leftarrow * \Sigma$$

$$\Sigma \leftarrow \emptyset$$

$$\left.\begin{array}{l} \Sigma \leftarrow a \\ \Sigma \leftarrow b \end{array}\right\} \text{Aufzählung der Elemente von X}$$

....

Jeder Formel dieses Systems ordnen wir eine Menge von Folgen von X - Elementen folgendermaßen zu:

$\emptyset$ bezeichnet die leere Menge

a bezeichnet die Menge, die die Folge "a" der Länge 1 enthält

b entsprechend ....

$\vee\Sigma\Sigma$ bezeichnet die Vereinigung derjenigen beiden Mengen, die den Teilformeln zugeordnet sind, die durch Substitution aus den beiden Σ - Symbolen hervorgegangen sind;

$/\Sigma\Sigma$ bezeichnet die Menge, die genau diejenigen Folgen der Länge l enthält, für die gilt:

$l = l_1 + l_2$; $l_1 > 0$; $l_2 > 0$;

die ersten l_1 Glieder der Folge bilden eine Folge, die der Menge angehört, die der Teilformel zugeordnet ist, die durch Substitution aus dem ersten Σ - Symbol hervorgegangen ist; die letzten l_2 Glieder der Folge bilden eine Folge, die der Menge angehört, die der Teilformel zugeordnet ist, die durch Substitution aus dem zweiten Σ - Symbol hervorgegangen ist.

$*\Sigma$ bezeichnet die Menge, der genau diejenigen Folgen angehören, für die gilt: Es gibt eine natürliche Zahl $n > 0$, sodaß die Folge in n Teilfolgen zerlegt werden kann, deren jede der Menge angehört, die der Teilformel zugeordnet ist, die durch Substitution aus dem Σ - Symbol hervorgegangen ist.

Wir bezeichnen demnach

∇ als Auswahloperator

/ als Folgeoperator

$\divideontimes$ als Iterationsoperator .

Wir behaupten nun:

T1: Alle Eigenschaften von Eingabesignal-Folgen, die von einem endlichen, digitalen, deterministischen, synchronen Automaten wahrgenommen werden können, werden durch eine endliche Menge von endlichen Formeln des erklärten kombinatorischen Systems beschrieben.

Und umgekehrt:

T2: Für jede endliche Menge von endlichen Formeln des erklärten kombinatorischen Systems läßt sich ein endlicher, digitaler, deterministischer, synchroner Automat angeben, derart, daß jeder Formel eine Eigenschaft von Eingabesignal-Folgen entspricht, die vom Automaten wahrgenommen wird.

Die Theoreme erhalten ihre Bedeutung vorwiegend durch die Definition der Wahrnehmbarkeit. Wir wiederholen und diskutieren daher zunächst.

Def: Eine Eigenschaft E einer Eingangssignal-Folge wird von einem Automaten wahrgenommen, wenn es eine Eigenschaft F einzelner Ausgabesignale gibt derart, daß das Ausgabesignal die Eigenschaft F zur Zeit t dann und nur dann besitzt, wenn die Eingabefolge, deren letztes Symbol zur Zeit t auftritt, die wahrzunehmende Eigenschaft E besitzt.

Die Zweckmäßigkeit dieser Definition soll kurz erläutert werden: Die Eigenschaften einzelner Eingangssignale müssen wir jedenfalls als unterscheidbar und damit im umgangssprachlichen Sinne als wahrnehmbar bezeichnen; das liegt in der Natur von Signalen digitaler Art. Darüberhinaus interessieren die Eigenschaften von Folgen solcher Eingangssignale, z. B. die Eigenschaft, nur die Signale a und b zu enthalten, die wir mit $\ast \nabla$ a b bezeichnen. Diese könnte man nun auf Eigenschaften von Folgen von Ausgabesignalen abbilden (Burks, Wang, Wright, Moore, Mealy: 7,193; 7,279; 8,1357; 1,129; 9,1045); demgegenüber betrachten wir gerade diejenigen Eigenschaften, die sich im einzelnen Ausgabesignal ausdrücken (verallgemeinert nach Kleene, Copi, Elgot, Wright: 1,3; 3,181). Selbstverständlich lassen sich bei deterministischen Automaten daraus die Eigenscha[ften der ausgegebenen Folgen wieder aufbau]en; dieser Aufbau interessiert hier aber nicht, da er für den Automaten nicht charakteristisch ist.

Die Menge der möglichen Ausgabesignale haben wir als vorgegeben betrachtet. Man könnte fragen, ob es vielleicht interne Signale geben kann, die nicht zu den Ausgabesignalen gehören, und die auch solche Eingabeeigenschaften zu unterscheiden vermögen, die sich durch Ausgabesignale nicht unterscheiden lassen. Es wäre dann fraglich, ob man die betreffenden Eigenschaften als wahrgenommen bezeichnen sollte. Von diesem Dilemma wollen wir uns völlig befreien, indem wir festlegen: das Ausgabesignal zur Zeit t ist das geordnete p-tupel der Werte von allen p der Feststellung zugänglichen Elementarsignalen im Automaten zur Zeit t. Wir müssen uns also im Einzelfall vorher festlegen, welche Elementarsignale wir als zugänglich betrachten. Dann sind die Theoreme sowohl bei der "black box" - Auffassung als auch bei physikalisch-struktureller Kennzeichnung des Automaten anwendbar.

Für den Beweis von T2 können wir auf Copi, Elgot und Wright (3, 181) verweisen, wo ein Algorithmus zur Synthese von Schaltwerken (logical nets) angegeben ist, welche die Forderungen von T2 erfüllen. (Dort werden völlig gleichwertig die Produktionen

$\Sigma \leftarrow (\Sigma \vee \Sigma)$ statt $\Sigma \leftarrow \nabla \Sigma \Sigma$

$\Sigma \leftarrow (\Sigma \cdot \Sigma)$ " $\Sigma \leftarrow / \Sigma \Sigma$

$\Sigma \leftarrow \Sigma^*$ " $\Sigma \leftarrow * \Sigma$

$\Sigma \leftarrow \Lambda$ " $\Sigma \leftarrow \emptyset$

benutzt, was von Vorteil ist, wenn die Formeln selbst nur zur Erklärung dienen) . Die Autoren benutzen Konjunktion, Disjunktion, Negation und Verzögerung um einen Zeitschritt; als Eigenschaften der Ausgabesignale werden nur die Werte einzelner Elementarsignale benötigt. Von dieser Charakterisierung des Automaten kann man bekanntlich ohne weiteres zu den anderen Darstellungen, wie Zustands-Übergangsgraph usw. übergehen.

Der Beweis von T1 wird im wesentlichen ebenfalls von Copi, Elgot und Wright angegeben (3, 187) , und es bedarf nur noch einiger ergänzender Bemerkungen, um die hier gebrauchte Form des Theorems zu rechtfertigen.

Es ist keine Beschränkung der Allgemeinheit, den Wertebereich der Elementarsignale als $\{0, 1\}$ anzunehmen. Für die Entscheidung, ob ein Signal eine bestimmte Eigenschaft besitzt oder nicht, genügt ein Signal von diesem Wertebereich; andererseits läßt sich ein Signal von größerem Wertebereich $\{a_1, a_2, \ldots a_n\}$ in bekannter Weise durch ein k - tupel binärer Elementarsignale ausdrücken, wenn

$2^k \geq n$ gewählt wird. Wir denken uns $k = n$ gesetzt und die entstehende binäre Verschlüsselung der Eingangssignale durch den ganzen Automaten hin fortgesetzt. Weiterhin ist es keine Einschränkung der Allgemeinheit, wenn die Autoren nur Automaten betrachten, die durch die Kombinationsweise von Schaltelementen charakterisiert sind. Ist statt dessen der endliche Zustandsgraph des Automaten vorgegeben, so läßt sich nach den bekannten Methoden der Schaltwerkstheorie stets ein Schaltwerk angeben, welches dem Graphen genügt. Darüber hinaus geben Mc Naughton und Yamada Algorithmen an (10, 39), welche direkt gestatten, Formeln der Ausgabesignale aus dem Zustandsgraphen zu finden und umgekehrt.

Schließlich sprechen wir allgemeiner von Eigenschaften eines Ausgabesignals, während Copi, Elgot und Wright nur die binären Eigenschaften , d.h. die Werte zur Zeit t , der einzelnen ausgabeseitigen Elementarsignale betrachten. Wir verallgemeinern dies mit der Absicht, das Theorem, welches ja mit spezieller Codierung nichts zu tun hat, auch in seinem Wortlaut von der Verschlüsselung von Signalen unabhängig zu machen. Dadurch wird zunächst unser Theorem 1 stärker, unser Theorem 2 schwächer als die entsprechenden Theoreme bei Copi, Elgot, Wright. Wir müssen daher zu T1 einen Beweisschritt ergänzen, während bei T2 der

Hinweis genügt, daß durch den angegebenen Synthesealgorithmus die wahrzunehmenden Eigenschaften tatsächlich bereits durch binäre Elementarsignale ausgedrückt werden.

Der zu ergänzende Beweisschritt lautet:

Es ist zu beweisen: Jede Untermenge der Menge der Ausgabesignale wird durch eine KS1 - Formel (d. h. durch eine Formel des oben angegebenen kombinatorischen Systems) dargestellt, wenn jedes Ausgabesignal durch eine solche Formel dargestellt wird.

Beweis: Die Untermengen bilden, wie erwähnt, einen Booleschen Verband von 2^k Elementen, mit den k Erzeugenden α, β, ... Die Darstellung der leeren Untermenge ist die Formel "$\emptyset$", die Darstellungen von α, β ..... seien F_α, F_β ..... Ist nun H eine Untermenge mit h Elementen und F_H ihre Darstellung, so ist die Darstellung F_J einer Untermenge $J \supset H$ von $h + 1$ Elementen gegeben durch

$F_J = \nabla F_H F_\nu$, wobei F_ν die Darstellung des hinzugekommenen Elements ist. Dies ist aber wieder eine Formel des kombinatorischen Systems; und da jede Untermenge höchstens k Elemente enthält, ist die sie darstellende Formel endlich.

Wir können nun sagen, daß ein Automat der betrachteten Art zwar beliebige Eingangsfolgen aufnimmt, welche Mitteilungen in einer bestimmten formalen Sprache enthalten; daß er aber die Mitteilungen nur insoweit verwerten kann, d.h. eine nach Art und Zeitpunkt wohldefinierte Wirkung auslösen kann, als die Syntax der formalen Sprache durch Formeln des Systems KS1 ausgedrückt werden kann.

Eine solche Syntax nennen wir im folgenden iterativ, weil ihre Reichweite im wesentlichen durch den Iterationsoperator beschrieben wird. Besitzt nämlich ein Automat nur einen Zustand, ist er also einem Schaltkreis äquivalent, so kann er nur einzelne Eingangssignale wahrnehmen: die Syntax der möglichen eingangsseitigen Sprachen ist in KS1 enthalten und nur der Operator ∇ wird benutzt. Werden dann Verzögerungsglieder hinzugenommen, so wird der Operator $/$ gebraucht, und schließlich wird die Rückkopplung von Signalen durch den Operator $\ast$ ausgedrückt.

Die Bezeichnung "iterativ" für eine Syntax dehnen wir der Kürze halber auch auf die durch eine solche Syntax definierten formalen Sprachen aus, und auch auf Mitteilungen in diesen Sprachen.

Ferner wollen wir unter einer iterativen Aufgabe eine solche verstehen, deren Behandlung darin besteht, daß eine beliebige Symbolfolge (die Aufgabenstellung), als Eingangssignalfolge eines Automaten betrachtet, gemäß einer iterativen Syntax wahrgenommen wird, wobei die Folge der Wahrnehmungen (Ausgangssignalfolge) die Lösung darstellt.

Eine Syntax, die nicht vollständig durch endlich viele Formeln aus KS1 beschrieben werden kann, nennen wir "irregulär". Irregulär ist z.B. die Syntax jeder Sprache, in welcher Klammern in der üblichen Bedeutung unbeschränkt zugelassen sind, wie in der Mehrzahl der mathematischen Formelsprachen. Ferner ist die Syntax von Algol sowie die der natürlichen Sprachen irregulär. Das Interesse an irregulären Sprachen im Zusammenhang mit Automaten stammt vor allem von der Notwendigkeit her, rekursive Aufgaben zu lösen, soweit sie mit Hilfe einer Turingmaschine gelöst werden können. Es sei aber angemerkt, daß der Konfliktbegriff in §4 die Behandlung von noch allgemeineren irregulären Aufgaben zuläßt.

Der zweite Punkt der Argumentation, welcher lautete "Automaten von fester endlicher Größe können höchstens iterativ erklärte Klassen von Eingangsfolgen wahrnehmen" möge damit als ausreichend erläutert gelten.

3.) "Rekursive Mitteilungen können nur von Automaten unbeschränkter Größe wahrgenommen werden."

Wir beweisen diese Aussage, indem wir zei-gen, daß schon bei einer geringen Erweiterung des kombinatorischen Systems, welche dazu diente, mit endlichen Mitteln eine im allgemeinen unendliche Menge von Zeichenfolgen zu charakterisieren, ein Automat fester endlicher Größe nicht ausreicht, um die Zugehörigkeit einer Mitteilung zu einer solchen Menge zu entscheiden, wohl aber eine Turingmaschine unter Einschluß ihres Bandes. Dieses Band stellen wir uns zunächst als aus einem einzigen Feld bestehend vor, und bestimmen, daß jede Bandverschiebung, die den Lesekopf nicht auf ein vorhandenes Feld führt, unter dem Lesekopf ein neues Feld mit stets demselben Symbol erzeugt. Die Größe dieser Maschine einschließlich Band ist dann nicht beschränkt.

Zunächst wollen wir den Ausdruck "rekursive Mitteilung" erläutern. Gemeint ist eine Mitteilung, also eine Zeichenfolge, die einer Sprache mit rekursiver Syntax angehört.

Eine Syntax nennen wir rekursiv, wenn sie einen Rekursionsoperator R enthält, der sogleich definiert werden soll. Sie möge erzeugt werden von einem kombinatorischen System KS2 :

Axiom: Σ

Produktionen:

$$\Sigma \leftarrow \nabla \Sigma \Sigma$$

$$\Sigma \leftarrow / \Sigma \Sigma$$

$$\Sigma \leftarrow R$$

$$\left.\begin{array}{l} \Sigma \leftarrow a \\ \Sigma \leftarrow b \end{array}\right\} \text{Aufzählung der Elemente von } X$$

$$\begin{array}{c} \cdot \\ \cdot \end{array}$$

Interpretation der Formeln von KS2 :

Jeder Formel des Systems ordnen wir eine Menge von Folgen von X - Elementen folgendermaßen zu:

$$\left.\begin{array}{l} a \\ b \\ \cdot \\ \cdot \\ \cdot \\ \nabla \Sigma \Sigma \\ / \Sigma \Sigma \end{array}\right\} \text{ebenso wie bei KS1}$$

R interpretieren wir konstruktiv, indem wir eine Turingmaschine mit Ein- und Ausgabemöglichkeit betrachten, welche jedes Element der Folgenmenge auf ihrem Band zu erzeugen vermag, indem sie die anfangs auf ihr Band gegebene Formel durcharbeitet und die zum Hinweis auf ein bestimmtes Element erforderliche Information in Form einer 01 - Folge ihrer Umgebung abverlangt. (Dasselbe könnte man in einfacherer Art auf KS1 - Formeln anwenden).

Auf diese Weise ergibt sich zugleich eine Zuordnung zwischen 01 - Folgen und den Sätzen einer rekursiven Sprache, deren Bedeutung natürlich tiefer liegt als die Möglichkeit, die Einzelzeichen der Sätze in 01 umzuschreiben.

Die Maschine liest die Formel von links nach rechts und handelt folgendermaßen: Am Anfang enthalte das Band die Formel und dahinter das Klammernpaar [ ] zur Aufnahme der generierten Folge. Die Maschine ist bereit, das erste Zeichen der Formel zu interpretieren.

Interpretiertes Zeichen:	Operation:
/	keine (nächstes Zeichen rechts interpretieren)
a b . .	das betreffende Zeichen ersetzt] und dahinter wird] gesetzt. Das interpretierte Exemplar bleibt unverändert.
∇	Anfrage der Maschine an die Umgebung, ob ein Rekursionsschritt ausgeführt werden soll.

(Antwort 0 = nein, 1 = ja)
Wenn nein:
Behandeln der nächsten Teilformel, Überspringen der übernächsten Teilformel.
Wenn ja :
Überspringen der nächsten Teilformel (und damit: Behandeln der übernächsten Teilformel.)

R Eine Kopie der kürzesten ∇ - Teilformel, in deren zweiter Unter-Teilformel das R - Exemplar auftritt, ersetzt dieses Exemplar. Die Interpretation wird mit dem ersten Zeichen der eingesetzten Unter-Teilformel fortgesetzt.

Der Versuch, die [- Klammer zu interpretieren, beendigt den Vorgang bzw. kann die "Ausgabe" des [ ] - Inhalts auslösen. Wenn bei der Interpretation von R keine kürzeste

∇ - Teilformel gefunden wird oder wenn ein bestimmter Rekursionsschritt immer wieder gefordert wird, kommt der Interpretationsvorgang nicht zum Ende. Es wird dann also keine Folge ausgegeben; d.h. die Interpretation von R ist eine zum selben Effekt führende Verallgemeinerung von derjenigen von $\emptyset$ (in KS1).

Weiterhin lassen sich alle mit KS1 - Formeln darstellbaren Folgenmengen auch in KS2 darstellen, denn die von $\nabla \Sigma \Sigma$ in KS1 erzeugte Menge ist ja dieselbe wie die von $\nabla \Sigma \Sigma$ in KS2 erzeugte Menge, wenn bei der Auswertung kein R bzw. $\emptyset$ angetroffen wird.
Ferner können wir $\ast \Sigma$ in KS1 durch $\nabla \Sigma / \Sigma$ R in KS2 ausdrücken.

Die programmiertechnischen Einzelheiten des Interpretationsvorgangs seien hier übergangen; es sei nur erwähnt, daß der Maschine zusätzlich eine beschränkte Menge von Zeichenformen oder Bändern zur Verfügung stehen sollen, womit die Markierung von betrachteten Zeichenstellen, von zu überspringenden und von zu erkennenden Teilformeln möglich ist. Das Erkennen von Teilformeln kann in einfacher Weise durch Auszählen geschehen: man setzt einen Zähler auf "1" und eine Marke vor die zu prüfende Zeichenkette. Man zählt für ∇ oder / "+1", für R , diesem Zeitpunkt steht fest, welches Element der Menge am Anfang der Eingangsfolge stehen kann. Die nun folgende Tätigkeit des Schaltwerks muß für jedes Element der Menge eine andere sein, also muß sich das Schaltwerk für jedes Element der Menge beim Erscheinen der ersten 0 in einem anderen Zustand befinden. Da es aber nur einer bestimmten endlichen Anzahl von Zuständen fähig ist, ist dies unmöglich.

Dagegen läßt sich die Menge durch die KS2 - Formel

F1: $\nabla\, 0 \,/\!/\, 1\, R\, 1$

erzeugen, und es ist klar, daß es eine Turingmaschine gibt, die diese Menge wahrnimmt (5).
Betrachten wir die Formel aus KS1

$/\!/ \ast\, 1\, 0 \ast 1$

so erzeugt sie zwar eine echte Obermenge von $\nabla\, 0 \,/\!/\, 1\, R\, 1$, aber eben nicht genau die Menge; es gibt keine Möglichkeit, die gleiche Anzahl der Iterationsschritte für die beiden $\ast$ - Symbole in KS1 auszudrükken (12).
In der Theorie der endlichen Automaten können wir also nicht einmal die Gleichheit zwischen natürlichen Zahlen ausdrücken.

4.) Dieser Sachverhalt erscheint wohl den meisten Praktikern trivial. Da man ohnehin und endgültig keine Möglichkeit hat, etwas wie einen "beliebig großen" oder gar "unendlichen" Automaten zu bauen, findet man sich hiermit ab. Ein endliches physikalisches Gebilde kann eben nur endlich viel "Information speichern", es ist nur endlich vieler unterscheidbarer Zustände fähig. Daraus folgt, daß es irreguläre Mitteilungen grundsätzlich nicht wahrnehmen kann - wenn wir es als synchronen Automaten betrachten müssen. Nur diese letzte, gewöhnlich unausgesprochene Voraussetzung soll hier widerlegt werden.

Es ist nämlich keineswegs gleichgültig, an welcher Stelle wir die Endlichkeit der Schranken für die Leistungsfähigkeit eines Automaten ins Spiel bringen. Das zeigt sich stets dann, wenn die Struktur eines Automaten oder eines Programms selbst Gegenstand einer Datenverarbeitung ist und wir diese Datenverarbeitung wiederum einem Automaten übertragen wollen. Dieser Schritt ist nicht allgemein ausführbar.

Betrachten wir als Beispiel die Übersetzung von Algol - Texten in eine Maschinensprache. Algol ist eine irreguläre Sprache: man beachte die induktiven Erklärungen für "arithmetischer Ausdruck", "Block" usw. Einen Algol - Übersetzer konstruieren, heißt: ein Programm herstellen zur Übersetzung einer Ersatzsprache (iterative Pseudo - Algol - Syntax).

Der Benutzer des Übersetzers muß dann die Ersatzsprache kennen, um die Tätigkeit des Übersetzers voraussehen zu können.

Das scheint nicht schwerwiegend zu sein: Der Benutzer glaubt, die Notwendigkeit von Beschränkungen einzusehen, wie:"höchstens 500 Klammerpaare dürfen ineinandergeschachtelt werden" , "höchstens 10 000 Namen dürfen auftreten" oder implizit: "höchstens 50 000 Befehle dürfen entstehen" .

Die Schwierigkeit ist aber folgende: Bei zunehmender Komplexität der Maschinenbenutzung entsteht ein nicht lösbares Verständigungsproblem zwischen Maschine und Benutzer, denn der Benutzer braucht notwendig eine rekursive Beschreibung der Sprache, die er zum Ausdruck seines Problems nehmen darf, oder in der ein anderer Benutzer ein Problem formuliert hat, wenn er das Problem "verstehen" will. Im einfachsten Fall muß er wissen können, daß in einem Bereich ein Exemplar eines Namens auf dasselbe Objekt hinweist wie ein formgleiches zweites Exemplar, und zwar allgemein, für beliebig viele verschiedene Namen. Von einem solchen "Verständnis" kann bei einem Automaten (Schaltwerk) keine Rede sein, wie das Beispiel ∇ 0 / / 1 R 1 beweist. Es ist also unmöglich, daß Benutzer und Maschine dieselbe vollständige Beschreibung einer Sprache als gemeinsame Grundlage der Kommunikation annehmen!

Es ist daher nicht möglich, eine formale Sprache wie Algol einschließlich ihrer Semantik (mit Hilfe von Äquivalenzrelationen) vollständig zu formalisieren, ohne den Boden der Automatentheorie zu verlassen.
(Es gelingt in einer endlichen Zahl von KS2-Schritten).
Wenn es also gelingt, die Begriffsbildungen der Automatentheorie so zu verallgemeinern, daß die Theorie

a) den physikalischen Postulaten gleich gut oder besser angepaßt ist, und
b) eine Kommunikation in KS2 - Sprachen zuläßt,

so ist auch die Situation der praktischen Datenverarbeitung grundlegend geändert.

Entscheidend ist dabei die Tatsache, daß die Menge der KS2 - Formeln (also die "Menge der rekursiven Sprachen") auf dieselbe Weise aus einer einzigen KS2 - Formel (für ein festes Alphabet X) erzeugt werden kann, wie die Menge der Zeichenfolgen einer Sprache aus den KS2 - Formeln.
Dies gilt nicht für KS1!
Denn die Menge der KS1 - Formeln gehört, ebenso wie die der KS2 - Formeln, als Zeichenfolgen betrachtet, keiner regulären Metasprache an. Dagegen läßt sich leicht zeigen, daß es finite Interpretationsvorschriften für Zeichenfolgen gibt, die einen Isomorphismus zwischen KS2 und der Syntax der

beschreibenden Sprache (Meta - KS2) zulassen.
Im interpretierten KS2 haben wir also die Möglichkeit, die Syntax einer Metasprache mit denselben Ausdrucksmitteln zu beschreiben, wie die Sprache selbst.
Bezeichnen wir die durch den Isomorphismus bedingte Zuordnung der KS2 - Zeichenformen zu den Meta - KS2 - Zeichenformen durch Einrahmung der KS2 - Zeichenformen, so haben wir zunächst für ein Alphabet

$\{ (0), (1), (\nabla), (/), (R) \}$:

Aus der Formel

F2: $\nabla \nabla \nabla (0) (1) (R) \; / \; / \nabla (\nabla) (/) R \; R$

ist jede KS2 - Formel für das Alphabet $\{ (0), (1) \}$ in endlich vielen Schritten ableitbar.

Beweis: Wir numerieren die Zeichen der Formel so:

$\nabla_1 \nabla_2 \nabla_3 (0) (1) (R) \; /_1 \; /_2 \; \nabla_4 (\nabla) (/) \; R_1 \; R_2$

und erklären :

∇_3 beschreibt die Auswahl eines Alphabetzeichens;

∇_2 " " " zwischen (R) einerseits und Alphabetzeichen andererseits

∇_4 beschreibt die Auswahl eines Operators
(∇) oder (/) ;

$/_2$ " das Anhängen einer Formel an einen Operator, dessen vordere Teilformel sie bilden soll;

$/_1$ " das Anhängen einer Formel an die erste Teilformel eines Operators.

R_1 und R_2 hängen nur von ∇_1 ab und bezeichnen also zu interpretierende Kopien der obigen Formel;

∇_1 beschreibt die rekurrierende Anwendung der Produktionen von KS2:

$$\Sigma \leftarrow / \Sigma \Sigma \quad \text{bzw.} \quad \Sigma \leftarrow \nabla \Sigma \Sigma$$

Nun müssen noch die Zeichenformen

(0) (1) (∇) (/) (R) durch passend gewählte Folgen der beiden noch nicht benutzten Zeichen 0 und 1 der Metasprache ausgedrückt werden. Hierbei haben wir einige Freiheit. Wir wählen eine Zuordnung, deren Auswirkungen leicht überschaubar sind, und denken uns folgende Substitutionen in F2 ausgeführt:

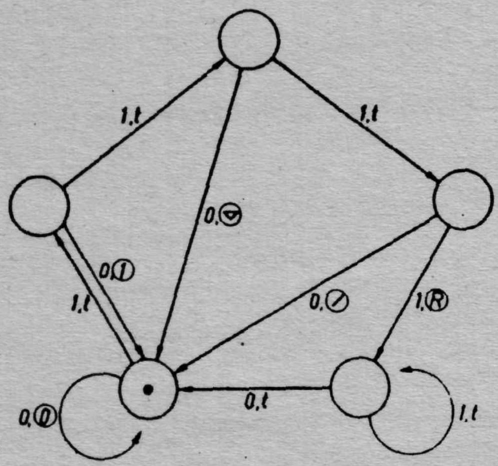
(0) ← 0

(1) ← /10

(∇) ← //110

(/) ← ///1110

Dann gibt es ein endliches Schaltwerk mit zwei Eingängen $\{0, 1\}$ und sechs Ausgängen $\{0, 1, \nabla, /, R, t\}$, welches jeder Folge von Meta - Zeichenformen $\{0, 1\}$ eine (durch t lückenhafte) Folge von Objektzeichen und somit eine Folge von KS2 - Formeln zuordnet:



Das Ende der ersten Formel der Folge kann, falls es existiert, durch die oben beschriebene Zählung erkannt werden.
Dann ist die Beschreibung der Arbeitsweise der Maschine, welche die Formel interpretieren soll, nur eine genauere konstruktive Formulierung dessen, was wir unter einem interpretierten kombinatorischen System mit entsprechend einfachen Produktionen verstanden haben.

Wir wollen eine Maschine zur Interpretation von KS2 - Ausdrücken eine R - Maschine nennen. Diese nimmt zunächst eine KS2 - Formel auf und sodann eine weitere, den Ablauf der Rekursionen steuernde 01 - Folge; diese 01-Folge denken wir uns als von einer anderen R - Maschine stammend. Die Anwendung der Formel auf die Folge nennen wir wie in der kombinatorischen Logik "Applikation" (12). Auf diese zweistellige Operation reduziert sich hier der Übergang von metasprachlichem zu sprachlichem Ausdruck. Da die R - Maschinen sich nur durch ihre KS2 - Formeln unterscheiden, wird es möglich, sie ausschließlich durch solche Formeln zu beschreiben, und zwar auch unbeschränkte Maschinen durch endliche Formeln.

In dieser Beschreibung entfällt jeder strukturelle Unterschied zwischen Mitteilungen und Automaten. Wir könnten die Programmierung der R - Maschinen so ergänzen, daß die R - Maschine nach einer Ausgabe nicht stehenbleibt, sondern ihr Band löscht (was ja in endlich vielen Schritten möglich ist, da sonst keine Ausgabe stattgefunden haben kann) und sich zur Aufnahme einer neuen KS2 - Formel bereitmacht, indem sie ein Signal abgibt.

Wir könnten dann den Informationsfluß in einem applikativen System (12) verfolgen : dieses besteht aus einer Menge von Objekten (nämlich endlichen Formeln), über der es eine dreistellige Relation gibt (nämlich die Applikation: Formel 1 angewandt auf Formel 2 gibt Formel 3).

Da, wie oben erwähnt, die Interpretation von KS2 diejenige von KS1 mit umfaßt, kann eine R - Maschine, der ein Exemplar von F2 vorliegt, die Beschreibung jedes endlichen Automaten (Schaltwerks) erzeugen, ferner jede endliche 01 - Folge (unter Anhängen einer 0).
Insbesondere kann die Beschreibung einer universalen Turingmaschine erzeugt werden und jedes Programm für diese Maschine, damit also jede Turingmaschine (unter passenden zusätzlichen Verabredungen über Codierung).
Nun weiß man aus der Theorie der Turingmaschinen (5) , daß es keinen Algorithmus (wieder durch eine Turingmaschine darzustellen) geben kann, der allgemein aus einer vorgelegten Formel 1 ("Programm") und Formel 2 ("Ausgangsdaten") zu entscheiden gestattet, ob Formel 3 (Applikationsergebnis) existiert, d.h. ob die sie darstellende Folge endlich ist (z.B. 5, 70). Für den Versuch, eine Applikation auszuführen, gibt es also im allgemeinen keine Garantie des Erfolges, und erst recht keine Garantie dafür, daß ein Erfolg durch eine vorher festgelegte Zahl von Rechenschritten erreicht wird.

Wenn wir am Ergebnis einer Applikation interessiert sind, so bleibt uns nur übrig, den Prozeß einzuleiten und auf ein Fertigsignal zu warten, über dessen Auftreten innerhalb einer gegebenen Zeitspanne wir im allgemeinen Fall schlechthin nichts aussagen können. Diese wohlbekannte fundamentale Tatsache der irregulären Kommunikation soll uns, neben physikalischen Erwägungen, ein weiterer Anlaß sein, die primitiven "Schaltelemente" gerade so auszuwählen, daß nur diejenige Eigenschaft von ihnen verlangt wird, auf die es im Gesamtsystem allein ankommt und die allein in einem unbeschränkten System effektiv aufrechterhalten werden kann: nämlich die Invarianz gewisser kombinatorisch - topologischer Verknüpfungen zwischen benachbarten Elementen.

5.) Automaten, die im Sinne der Automatentheorie konzipiert werden, geraten nach endlich vielen Erweiterungsschritten in Widerspruch zu mindestens einem der genannten Postulate. J. von Neumann (unveröffentlicht, nach einer Mitteilung in (13), und A. W. Burks (13, 282) haben die Konstruktion eines Speichers aus logischen ($\wedge$, $\vee$, $\ddagger$) und verzögernden Elementen angegeben, der die Arbeitsweise des Bandes einer Turingmaschine besitzt. Hierbei werden "unendlich schnelle" Signale vorausgesetzt, womit der Speicher von vornherein einem der Postulate widerspricht.

Wir können nun die Konstruktion dadurch abändern, daß wir die Verknüpfungsvorschrift der Elemente, die über einem ebenen quadratischen Netz gegeben sei, durch eine dem Postulat der endlichen Signalgeschwindigkeit entsprechende Vorschrift ergänzen, derart, daß jedes Signal, das einen Punkt mit den (ganzzahligen) Netzkoordinaten (x_1, y_1) mit einem Punkt (x_2, y_2) verbindet, mindestens

$$v \cdot (|x_2 - x_1| + |y_2 - y_1|)$$

Verzögerungsglieder zu durchlaufen hat. v sei eine Konstante:

$$0 < v < 1$$

Dann ist die Zahl der Verzögerungsglieder bei wachsendem Speicher nicht beschränkt. Somit ist auch ihr Abstand voneinander nicht beschränkt. Wäre er beschränkt, so könnten wir den "Gleichlauf" der Verzögerungsglieder durch Taktsignale erreichen, sodaß eine Ungenauigkeit der Verzögerungszeit $\Delta \Delta t$ keinesfalls dazu führen kann, daß ein späteres Objektsignal ein früheres auf einem anderen Wege überholt.

Die Taktsignale müßten nun beliebig weit voneinander entfernte Punkte verknüpfen, sie müssen also selbst der ergänzenden Verknüpfungsvorschrift unterworfen werden.

Damit müssen neue Verzögerungsglieder eingefügt werden, deren Zahl proportional zum betrachteten Abstand ist, und die selbst wieder getaktet und auf freie Gitterpunkte des Netzes gelegt werden müssen.

Selbst wenn für den zu erweiternden Automaten nur eine lineare Möglichkeit der Erweiterung gefordert wird, und wenn wir ein n - dimensionales Netz zulassen, läßt sich zu jedem k und v ein l finden, sodaß gilt

$$N > k \cdot L^n \quad \text{wenn } L > l$$

$$(k > 0, \quad 0 < v < 1)$$

wo L die Länge der zu konstruierenden linearen Anordnung von Schaltwerken ist, festzustellen durch Abzählen von $(n - 1)$ - dimensionalen Ebenen, und N die Gesamtzahl der durch die Verknüpfungsvorschriften geforderten Verzögerungsglieder.
Die Elemente lassen sich also selbst dann nicht im Netz unterbringen, wenn eine bloße lineare Aufreihung von Schaltwerken derselben Form gefordert wird.

Also bleibt uns nur übrig, entweder $v = 0$ zuzulassen (unendliche Signalgeschwindigkeit) oder aber beliebig genaue Verzögerungsglieder zu fordern. Wenn wir uns unter einem Verzögerungsglied ein physikalisches Gebilde mit räumlicher Ausdehnung vorstellen, so müßten wir die Möglichkeit haben, solche Elemente

derart herzustellen, daß sie - als "Uhren" in sich selbst zurückgekoppelt - nach beliebig langer Zeit um nicht mehr als einen Takt verschiedenen Stand haben. Wir verneinen diese Möglichkeit in noch schärferem Maße als die Relativitätstheorie, nämlich auch für relativ zu einander ruhende Uhren.
Wir können dies als Postulat aussprechen:

Pl "Der Gleichlauf von Uhren kann nur durch Kommunikation herbeigeführt werden."

Von erkenntnistheoretischem Interesse ist eine weitergehende Formulierung:
"Der Begriff "Gleichlauf zweier Uhren" erhält seinen Inhalt ausschließlich durch die Beschreibung einer geschlossenen Signalkette, welche beide Uhren enthält."

In dieser Formulierung soll das Postulat hier nicht verteidigt werden. Dagegen akzeptieren wir nunmehr die erste Form und begründen dies mit dem Postulat der endlichen Informationsdichte: Wir können nämlich ein Experiment angeben, in welchem die Taktzeiten zweier Uhren durch Zählung von Takten beliebig genau verglichen werden, in dem also Δt_1 bezüglich einer Einheit Δt_0 gemessen wird. Wir können jede positive rationale Zahl als Maß für Δt_1 erhalten. Wenn wir annehmen, daß die Folge dieser Zahlen beim Fortschreiten des Experiments sich einem Grenzwert nähert, der nur vom physikalischen Zustand der Uhren zu Beginn des Experiments abhängt, so schreiben wir dem Uhrenpaar einen Informationsgehalt zu, der jede Schranke übersteigt.

Da wir diesen Informationsgehalt jedenfalls nicht besitzen, geschweige denn manipulieren können, so folgt, daß es begrifflich einfacher ist, wenn wir auf seine Erwähnung beim Aufbau einer Theorie der Kommunikation ganz verzichten.

Diesen wichtigen Schritt vollziehen wir in allen Fällen, wo reelle Zahlen auftreten : Raumkoordinaten, physikalische Zustandsgrößen (von denen wir nicht explizit sprechen werden), Wahrscheinlichkeit.

§ 3.

Durch die bisherigen Ausführungen ist erklärt, warum wir sowohl Schaltelemente ohne Zeitbedarf ($\wedge$, $\vee$, $\sim$ ....) als auch Schaltelemente mit fest gegebenem Zeitbedarf (Δ t) als Idealisierungen unserer elementaren logischen Bausteine ablehnen müssen, wenn wir die physikalische Realisierung von Automaten veränderlicher Gesamtstruktur im Auge haben. Wir dürfen sie aber auch ablehnen für Automaten mit fester endlicher Struktur, und es ist nunmehr zu zeigen, daß es möglich ist, andere Idealisierungen für logische Bauelemente anzugeben, derart daß gilt:

P2A: Jedes logische Element in seiner idealisierten Form hat ein manipulierbares Modell auf jedem Gebiet, auf dem wirkliche Kommunikation stattfindet.

Wenn dies gelingt, so lassen sich mathematisch - logische Modelle von allgemeinen Kommunikationsvorgängen schaffen, die in einem stärkeren Sinn exakt sind, als wir ohne diese Voraussetzung erreichen könnten, und es ist zu hoffen, daß solche Modelle eine tieferen Einblick in das Wesen der Kommunikation ermöglichen.

Unter Kommunikation verstehen wir hier alle Erscheinungsformen des tatsächlichen Infortionsflusses. Der Informationsfluß beschreibt die strukturellen Eigenschaften des Flusses der in der Physik "Wirkung" genannten Größe Die Forderung P2A soll uns die Universalität unseres Modells der Kommunikation gewährleisten. Wie aber soll man nachweisen, daß sie erfüllt ist? Ein solcher Nachweis wird hier nicht versucht werden. Wir wollen ihn aber erleichtern, indem wir die Forderung P2A aus ihrer indirekten Form in eine direkte bringen und sie dabei etwas verschärfen:

P2: Bei der Interpretation idealisierter logischer Elemente darf man sich nur auf solche finiten Eigenschaften der sie realisierenden physikalischen Objekte berufen, deren Invarianz durch ein allgemeines Prinzip (AP) gesichert ist.

Damit ist das Problem auf den Begriff des AP verschoben. Die philosophischen Implikationen des Begriffs brauchen uns hier nicht zu beschäftigen; dafür wollen wir drei Beispiele für APs angeben, die für die hier einzuführenden Schaltelemente bereits ausreichen (§ 5):

AP1: Die Existenz einer quantisierten Größe (z.B. Wirkung, Ladung)

AP2: Die Gültigkeit eines Erhaltungssatzes (z.B. Ladung)

AP3: Die Gültigkeit eines Reaktionsprinzips (z.B. für Kräfte)

Diese Prinzipien wurden gewählt, weil sie die Form der am besten gesicherten Aussagen der Physik haben. Es erweist sich, daß sie, zusammengenommen, die Freiheit in der Wahl logischer Elemente stark einschränken. Wir müssen dann zeigen, daß aus den noch erlaubten Elementen tatsächlich Gebilde erzeugt werden können, welche den Informationsfluß in jeder gewünschten Weise lenken können.

Ein naheliegendes Bedenken wäre nun, daß der Verzicht auf die klassischen Schaltelemente uns sofort auf das Gebiet der asynchronen Automaten verweist, welches als mathematisch so schwierig angesehen wird (14, 204), daß auf Anschaulichkeit verzichtet werden muß und daß an eine praktische Verwendung nur in den einfachsten Fällen gedacht werden kann.

Es ist nun merkwürdig, daß dieses Bedenken völlig verschwindet, wenn wir die Theorie unter Verwendung von P1 und P2 - also unter Verzicht auf gewisse zusätzliche, gewöhnlich akzeptierte Möglichkeiten, also durch Beschränkung unserer Darstellungsmittel - konsequent neu aufbauen.

Ferner suchen wir nach einem Weg, deterministische Vorgänge mit den gleichen Mitteln zu beschreiben und den Unterschied zwischen beiden durch topologische Eigenschaften von endlichen Strukturen über diskreten Mengen zu erklären.

Ein wesentliches Merkmal unserer Vorgehensweise ist, daß wir auf metrische Eigenschaften aller Art ganz verzichten müssen.

Wir setzen lediglich voraus, daß es Objekte mit irgendwelchen invarianten Eigenschaften gibt (zu interpretieren, je nach Zusammenhang, als Energiemengen, Wirkungsquanten, elektrische Ladungen, materielle Teilchen usw.), zwischen denen es beschränkte Verknüpfungen gibt.

Wir werden diese Objekte zu Netzen verknüpfen und zeigen, daß es Netze gibt, die einer Turingmaschine äquivalent sind, und deren ferne Teile ohne Bezugnahme auf den Ausgangspunkt der Konstruktion, also ohne Kommunikation zwischen beliebig voneinander entfernten Teilen des Netzes, erzeugt werden können. Daher wird sich zeigen, daß ein solches Netz, selbst wenn es erst aus wenigen Objekten besteht, von einem "unendlichen" Netz, welches rekursive Mitteilungen wahrnehmen kann, durch kein Experiment unterscheidbar und ihm insofern äquivalent ist.

Um den Zusammenhang mit den Anwendungen deutlicher hervortreten zu lassen, verzichten wir auf die mathematisch wesentlich elegantere Möglichkeit, die zeitliche Nachbarschaftsrelation und die räumliche Nachbarschaftsrelation zwischen Phasenzellen einheitlich zu behandeln; wir werden daher von Netzen und von Veränderungen auf ihnen sprechen und nicht von Zellenkomplexen.

Wir werden nun folgendermaßen vorgehen:

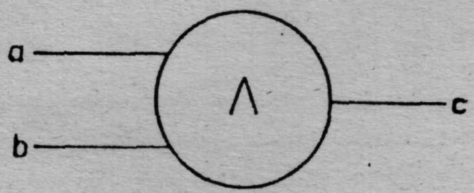
a.) Wir geben eine Methode an, mit Hilfe von intuitiv gewählten, nicht - metrischen Schaltelementen gewisse Netze zu erzeugen, die den physikalischen Informationsfluß, also den strukturellen Zusammenhang von Ereignis-mengen wiedergeben. Dabei erfüllen wir nur AP1, nicht aber AP2 und AP3. (§4).

b.) Wir schränken die Wahl der Schaltelemente ein, indem wir AP1 - 3 fordern, sodaß P2 verifiziert werden kann. Damit ist der Weg für Anwendungen freigegeben, und die Grunderscheinungen der Kommunikation sind mit rein kombinatorisch - topologischen Mitteln darstellbar. (§5)

Beim Schritt a) werden wir so sprechen, als ob die diskreten Objekte der Theorie in eine kontinuierliche Raum - Zeit - Welt eingebettet seien, um das Verständnis zu erleichtern. Es sei aber vermerkt, daß dies keineswegs notwendig ist; die scheinbare Anschaulichkeit des Kontinuums beruht letzten Endes auf der irrigen Annahme, daß das Axiom der Dichtigkeit (16, 200) einen operationalen (z. B. (17)) Sinn hat. Das Axiom ist vielmehr ein sprachliches Instrument zur Hypothesenbildung im Sinne der induktiven Logik (18; 19, 208).

§ 4.

Wir erklären die nun einzuführende Betrachtungsweise am Beispiel der Und - Schaltung.



Ein solches Schaltelement verknüpft drei Dinge (Drähte usw.), welche zu bestimmten Zeiten gewisse Zustände (Potential usw.) annehmen, in folgender Weise: Die Mengen der möglichen Zustände von a, b und c seien Za, Zb, Zc. Die Mengen 0a und 1a seien disjunkte Untermengen von Za, desgleichen für b, c. Die Wirkung der Schaltung wird beschrieben durch eine Abbildung der Menge aller Zustandstripel in sich selbst; die Fixpunkte der Abbildung sind die stabilen Zustandstripel:

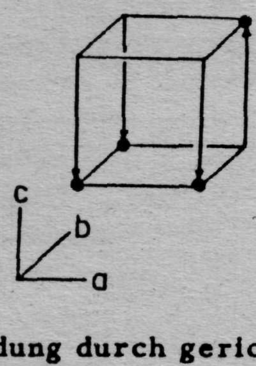
0a 0b 0c → 0a 0b 0c

0a 0b 1c → 0a 0b 0c

.....

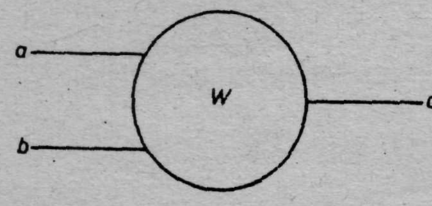
1a 1b 0c → 1a 1b 1c

.....



und wir können die Abbildung durch gerichtete Strecken kennzeichnen, welche die Eckpunkte eines Würfels verbinden. Die schwach bezeichneten Kanten durchläuft der Punkt, der den Zustand der Schaltung kennzeichnet, durch Einwirkung von Einflüssen außerhalb der Schaltung; die stark gezeichneten durchläuft er "von selbst", d.h. aufgrund der physikalischen Eigenschaften des Schaltelements. Dabei ist zu bemerken, daß beim Durchlaufen dieser c - Kanten der Zustand von a und b nicht derart verändert werden darf, daß er die betreffende Untermenge 0a, 1a, 0b oder 1b, der er angehört, wieder verläßt. Das läßt sich aber nur dadurch mit Sicherheit erreichen, daß die c - Kanten parallel zur c - Achse eines Zustands - Koordinatensystems verlaufen, gleichgültig ob wir die Zustandsmengen als dicht oder als diskret annehmen. Das bedeutet aber, daß hier ein physikalischer Vorgang ohne jede Rückwirkung (reactio) auf seine Ursache ablaufen muß. Daher haben wir sogar ohne die Forderung, daß der Vorgang "mit sehr geringem Zeitbedarf" ablaufen soll, kein exaktes physikalisches Modell zur Verfügung, welches sich entsprechend verhält: Die Und - Schaltung genügt nicht dem Postulat P2 .

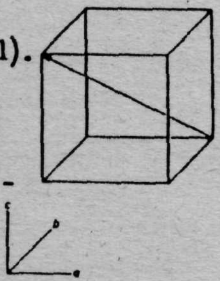
Wir betrachten ein Schaltelement W:



mit den einfachen Eigenschaften:

1a 1b 0c → 0a 0b 1c

(alle übrigen Tripel sind stabil).



Dieses Schaltelement könnten wir als "Und - Schaltung mit Rückwirkung" bezeichnen: "dadurch daß" c in den Zustand 1c übergeht, werden a und b in den Zustand 0a bzw. 0b versetzt. Da sich beim Durchlaufen der "aktiven Kante" des Elements alle Zustände ändern, tritt die genannte Schwierigkeit der Interpretation gar nicht auf. Wir können das Verhalten des Schaltelements so erläutern: W wartet, bis es in seiner Nachbarschaft Objekte 1a, 1b bemerkt, und stellt aus ihnen ein neues Objekt 1c her.
Dieses Warten des Schaltelements W soll aber in keiner Weise als eine Aktivität ausgelegt werden, etwa derart, daß das Element wiederholt die Zustände von a, b und c abfragt und bei einem gewissen Ergebnis der Abfrage die Zustände von a, b, c ändert; sondern es stellt lediglich eine Verknüpfung von a, b, c dar, die nur dann Anlaß zu einer Zustandsänderung geben kann, wenn sowohl 1a als auch 1b und 0c vorhanden ist; und die Zustandsänderung besteht darin und nur darin, daß nach ihrem Eintreten die Objekte 0a, 0b und 1c vorhanden sind.

Den Vorgang der Substitution von 0a, 0b, 1c für 1a, 1b, 0c betrachten wir als einen einzigen, nicht mehr in Teile zerlegbaren Elementarvorgang, und wir werden (in §5) nur physikalisch sinnvolle Vorgänge als elementar zulassen, was die Konstruktionen zunächst sehr erschwert, aber bemerkenswert einfache Resultate liefert.

Andererseits soll von einer Verknüpfung (einem Schaltelement) W weiter nichts als das beschriebene Verhalten gefordert werden: Wir verlangen nur, daß 0a 0b 1c ein (zeitlicher) Nachfolger von 1a 1b 0c sein soll, und werden schreiben:

$$W(a\,b\,c) \quad : \quad 1\,1\,0 \rightarrow 0\,0\,1$$

Über einen "Zeitbedarf" dieses Vorganges brauchen wir nichts auszusagen: Wie aus der Interpretation klar sein müßte, ist das Verschwinden eines Objekts 1a, das Verschwinden eines Objekts 1b und das Erscheinen eines Objekts 1c derselbe Akt.
Um die volle Breite der Interpretationsmöglichkeiten zu kennzeichnen, sind folgende Bemerkungen wesentlich:

1.) Dualität zwischen 0 - und 1 - Objekten:

Wir können dasselbe Schaltelement W auf zwei Arten charakterisieren:
$W \equiv 1a \quad 1b \rightarrow 1c$ und $W \equiv 0c \rightarrow 0a \quad 0b$.

Betrachten wir an Stelle von 1 - Objekten die zu ihnen dualen 0 - Objekte, so besteht die Wirkungsweise von W in einer Aufspaltung eines Objekts in zwei von ihm und untereinander verschiedene Objekte. Ein Netz aus solchen und ähnlichen Elementen läßt sich also in verschiedener Weise auffassen, je nachdem, welche der beiden Objektklassen man auszeichnen will. Wir verabreden daher: Die Zuordnung der Symbole 0 und 1 zu den Objektklassen ist willkürlich, soll aber festgehalten werden. Wenn wir eine Objektklasse auszeichnen wollen, so werden wir dies tun, indem wir ihr das Symbol 1 zuordnen.

Durch Vertauschung der Objektklassen geht ein Schaltelement in das zu ihm duale über:
$\overline{W}$ (a b c) : 0 0 1 $\rightarrow$ 1 1 0.

Für Schaltelemente, welche sämtliche erfaßten Objekte in eine andere Klasse überführen, ist der Übergang zum dualen Element offenbar mit einer Vertauschung des zeitlichen Richtungssinnes äquivalent.

2.) Wirkungsbereiche eines Schaltelements:

Die mit a, b und c bezeichneten "Anschlüsse" des Schaltelements sind seine (drei) Wirkungsbereiche: wenn ein 1 - Objekt (oder im dualen Fall ein 0 - Objekt) in einen Wirkungsbereich eintritt, kann es vom Schaltelement erfaßt werden.

Eine mögliche Interpretation des Wirkungsbereichs ist die, ihm ein räumliches (oder raumzeitliches) Gebiet zuzuordnen. Dabei entsteht die Frage nach der genauen Abgrenzung dieses Gebiets. Wir richten unsere Grundbegriffe auch hier wieder so ein, daß die Frage gegenstandslos wird. Wir vereinbaren nämlich: Wenn in einer Interpretation den Wirkungsbereichen Raum - (bzw. Raumzeit-) Gebiete zugeordnet werden, so sollen solche Gebiete nicht anders rückinterpretiert werden können als auf Wirkungsbereiche von Schaltelementen. Damit benötigen wir nur noch die topologischen Eigenschaften von Gebietsmengen: getrennte Lage, Überlappung, Einschließung.

Wirkungsbereiche ohne Angabe ihrer Interpretation wollen wir der Kürze halber mit einem besonderen Namen versehen: Wir nennen sie <u>Stellen</u>. W ist also "dreistellig".

In den wichtigsten Interpretationen kann jeder Wirkungsbereich nur eine gleichmäßig beschränkte Anzahl von Objekten aufnehmen. Es genügt dann, wie sich erweisen wird, nur Stellen zuzulassen, die stets genau ein Objekt enthalten. Der Übergang zu "Stellen", die stets genau n Objekte enthalten und die wir quasimetrische Stellen nennen können, ist dann stets durch Verknüpfung gewöhnlicher Stellen mit geeigneten Schaltelementen möglich.

Schaltelemente ohne Angabe der Interpretation nennen wir Knoten.

Wir wollen nun folgende Relationen zwischen Stellen zulassen: Identität, Nachbarschaft und getrennte Lage ($\equiv$ Nicht-Identität $\wedge$ Nicht-Nachbarschaft); nicht aber allgemeine Überlappung und Einschließung. Das führen wir durch folgende Definitionen herbei, in Übereinstimmung mit unserer obigen Verabredung über die Handhabung der Interpretation:
Sind a, b, ... die Wirkungsbereiche eines Schaltelements K, so sagen wir, die Stellen a, b, ... gehören dem Knoten K an:

J (a, K), J (b, K), ... (Inzidenz)

Zwei Stellen a, b sind identisch, wenn sie dasselbe Objekt enthalten: I (a, b); d.h. es soll gleichgültig sein, ob wir eine Identität zwischen Stellen oder zwischen Objekten zugrundelegen. I ist reflexiv, symmetrisch und transitiv: die von I erzeugten Äquivalenzklassen von Stellen sollten wir I - Stellen nennen, wollen aber weiterhin nur diese betrachten und das Wort "Stelle" auch für sie gebrauchen. Die Unterscheidung wird erst dann wesentlich, wenn wir die Konstruktionsvorschriften für Netze voll formalisieren.
Zwei verschiedene Stellen heißen benachbart: B (a, b), wenn es einen Knoten gibt, dem beide angehören:

$$B(a, b) \equiv (\exists K)(J(a, K) \cdot J(b, K)) . \sim I(a, b)$$

B ist irreflexiv, symmetrisch und nicht - transitiv.

Wir betrachten nur Stellen, die wenigstens einem Knoten angehören; denn nur durch die Inzidenz mit einem Knoten kann die Stelle an der Kommunikation teilnehmen:

$St(a) \equiv (\exists K)(J(a, K))$ ("a ist eine Stelle")

Zwei verschiedene Stellen heißen verbunden: C (a, b), wenn es eine natürliche Zahl $n \geq 1$ gibt, sodaß eine n - gliedrige B - Kette zwischen ihnen existiert:

$$C(a, b) \equiv (\exists n)(B^n(a, b))$$

C ist symmetrisch und transitiv.

Wir benötigen nun eine Menge von zwei Elementen, die wir mit den Symbolen 0 und 1 bezeichnen und mit denen wir die Stellen bewerten wollen; sie mögen die Binärwerte oder die totalen Bits heißen:

$$Tob(i) = i \varepsilon \{0, 1\}$$

Ein Objekt ist ein geordnetes Paar aus einer Stelle und einem totalen Bit:

$$Ob(a, i) \supset St(a) . Tob(i)$$

Wir gebrauchen das Wort "Objekt" also hier so, daß es sinnlos wäre, zu sagen: "das Objekt bewegt sich von Stelle zu Stelle". Die Genidentität (16, 201) zwischen Informationsmengen (Teilchen) muß vielmehr durch Angabe einer Netzstruktur beschrieben werden; sie ist nicht notwendig vom Beobachter unabhängig.

Die B - Relation erteilt einer Menge von Stellen eine Struktur, die B - Struktur. Eine zusammenhängende B - Struktur heiße ein B - Netz. Durch Bewertung der Stellen eines B - Netzes erhalten wir eine zusammenhängende B - Struktur über einer Menge von Objekten. Diese Struktur stellt den "Zustand eines Systems" in der Automatentheorie dar: Dort wird vorausgesetzt, daß man effektiv eine Nachfolger - Relation über einer Menge solcher Strukturen angeben oder gar erzeugen kann. Von dieser Voraussetzung haben wir uns hier freigemacht, nachdem wir nachgewiesen haben, daß sie nicht immer erfüllbar ist.

Statt dessen lassen wir nur eine Nachfolger-Relation "$\rightarrow$" für geordnete p - tupel von solchen Objekten zu, die sich an Stellen befinden, die einem einzelnen Knoten angehören.

Für unbeschränktes p erhalten wir wieder die Begriffsbildungen der Theorie der synchronen Automaten, welche in dieser Sicht eine Art von Fernwirkungstheorie darstellt. Die wesentliche Abweichung besteht also in der Beschränkung von p (auf kleine natürliche Zahlen); damit wird in der Interpretation nur auf Nahwirkungen Bezug genommen.

Wir wählen zunächst versuchsweise und willkürlich

$$\underline{\underline{P = 4}} \qquad 1 \leq p \leq P$$

und verabreden, daß in den Aktionen eines Knotens, z.B.

$$p = 3 : W(a\,b\,c) : 1\;1\;0 \rightarrow 0\;0\;1$$

stets alle Stellen genannt sind, die dem Knoten angehören. (Dies ist keine Einschränkung der Allgemeinheit).

Damit gehören jedem Knoten höchstens P Stellen an.
Die Inverse einer Aktion entsteht aus dieser durch Vertauschung von Vorder - und Hinterglied der Nachfolger - Relation. Einen Knoten, der zu jeder seiner Aktionen auch deren Inverse besitzt, nennen wir reversibel.
Die Anzahl F der verschiedenen möglichen Schaltfunktionen, d.h. der Belegungen von Knoten mit Aktionen, ist jedenfalls beschränkt, nämlich

$$F \leq \sum_{p=1}^{P} 2^{(2^{2p})} < 2^{(2^{2P+1})}$$

also für $P = 4$: $F < 2^{257}$. Wir wollen versuchen, mit einem geringen Teil von diesen auszukommen, und definieren:

Q (a)	:	$0 \to 1$
T (a, b)	:	$1\ 0 \to 0\ 1$
W (a, b, c)	:	$1\ 1\ 0 \to 0\ 0\ 1$
V (a, b, c)	:	$1\ 0\ 0 \to 0\ 0\ 1$
		$0\ 1\ 0 \to 0\ 0\ 1$
U (a, b, c)	:	$1\ 0\ 0 \to 0\ 1\ 1$
		$1\ 1\ 0 \to 0\ 0\ 1$
S (a, b, c, d)	:	$0\ 1\ 0\ 0 \to 0\ 0\ 0\ 1$
		$1\ 1\ 0\ 0 \to 0\ 0\ 1\ 0$

Inverse:

$\bar{Q}$ (a)	:	$1 \to 0$
$\bar{T}$ (a, b) $\equiv$ T (b, a)	:	$0\ 1 \to 1\ 0$
$\bar{W}$ (a, b, c)	:	$0\ 0\ 1 \to 1\ 1\ 0$

Diese Funktionen sind lediglich wegen ihrer Anschaulichkeit ausgewählt: in §5 werden wir sie durch einfachere ersetzen. Hier haben wir: Quellen (Q) von ausgezeichneten Objekten, Transport (T) von totalen Bits, Warten (W) auf das Erscheinen zweier Objekte, Vereinigung (V) von Bit - Kanälen, Umwandeln

(U) eines Objekts in das duale, und eine Funktion, die das zerstörende Abfragen eines Speichers (S) erlaubt.

Die definierten Relationen können nur zwischen benachbarten Stellen bestehen: z. B.

$$S(a,b,c,d) \Rightarrow B(a,b).B(a,c).B(a,d).B(b,c).$$
$$.B(b,d).B(c,d)$$

Damit ist auch die Nicht - Identität zwischen den Stellen eines Knotens impliziert.
Ein f - Netz definieren wir als Struktur, die von Schaltfunktionen aus einer Menge f von P - beschränkten (d.h. höchstens P Stellen verknüpfenden) Schaltfunktionen über einer Menge von Stellen erzeugt wird, derart, daß die implizierte B - Struktur ein B - Netz ist, also zusammenhängt.

Für die wichtigsten Interpretationen müssen wir die f - Netze einer weiteren Beschränkung unterwerfen, indem wir der Tatsache Rechnung tragen, daß nicht jedes Objekt jederzeit und von überallher unmittelbar zugänglich sein kann. Wir fordern, daß jede Stelle höchstens M verschiedenen Knoten angehört:

$$1 \leq m \leq M$$

und betrachten nun den wichtigen Fall

$$\underline{M = 2}.$$

Unter einem Netz verstehen wir nunmehr ein f - Netz mit P = 4, M = 2.

Dann können wir die B - Struktur eines Netzes in bequemer Weise durch einen Graph darstellen, dessen Kanten den Stellen und dessen "Knoten" den Netzknoten entsprechen. Wir wollen nun gewisse Netze so einrichten, daß für jedes Objekt stets eindeutig be stimmt ist, an welcher Aktion welches Knotens es teilnehmen kann. Das Verhalten des Netzes ist dann offenbar determiniert, wenn wir außerdem fordern, daß verschiedene Aktionen eines Knotens schon in ihren Vordergliedern verschieden sind.

Unter einem Konfliktfall verstehen wir die Situation, daß ein Objekt den Vordergliedern von mehr als einer anwendbaren Aktion angehört. Es ist dann völlig unbestimmt, welche der Aktionen angewandt wird. Für die Erkennung eines Konfliktes gilt:

Wenn eine anwendbare Aktion ihre Anwendbarkeit anders als durch ihre Anwendung verliert, so hat ein Konflikt vorgelegen, und umgekehrt.

Netze, in denen Konflikte unmöglich sind, nennen wir konfliktfrei. Die Konfliktfreiheit muß bewiesen werden durch ihre Erblichkeit gegenüber den Konstruktionsschritten des Netzes.

Was aber soll mit dem kritischen Objekt im Konfliktfall geschehen? Wir setzen mit Rücksicht auf die Interpretationen das

<u>Axiom: Im Konfliktfall kann höchstens eine Aktion eintreten.</u>

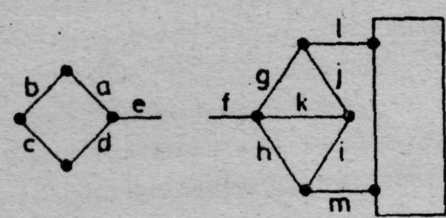
Es bleibt also lediglich unbestimmt, welche der Aktionen in Frage kommt; keinesfalls können mehrere Aktionen "gleichzeitig" angewandt werden.

Die <u>Randstellen</u> eines Netzes sind die Stellen, die genau einem Knoten des Netzes angehören.

Wenn wir Randstellen zweier konfliktfreier Netze paarweise identifizieren, so entsteht nicht notwendig ein konfliktfreies Netz. Das gleiche gilt, wenn wir die Randstellen durch T - Knoten verbinden, statt sie zu identifizieren.

Beispiel: Netz 1 ist eine "Uhr", Netz 2 ein Beobachter mit eigener Uhr

Netz 1:	T(ab)	Netz 2:	S(fkgh)
	T(bc)		$\bar{W}$(ljg)
	T(cd)		$\bar{W}$(mih)
	U(dea)		V(ijk)



Am Anfang gelte 1a, 1k, und der Rest von Netz 2 wirke wie $\bar{Q}$ (l) , $\bar{Q}$ (m) .

Beide Netze sind konfliktfrei, jedoch ist die Messung des Ganges der Uhr a b c d bezüglich des Ganges der Uhr h i k durch Verknüpfung oder Identifikation von e und f stets mit der Möglichkeit eines Konflikts verbunden, dessen kritisches Objekt in e oder f liegen kann.

Dieser Sachverhalt weist uns auf einen engen Zusammenhang zwischen Messung und Konflikt hin, der nicht unbeachtet bleiben sollte.

Ein konfliktfreies Netz mit zwei Randstellen, welches ausschließlich $n \geq 1$ T - Knoten besitzt, nennen wir einen Kanal. Die T - Knoten müssen in gleicher Weise orientiert sein. Damit ist auch die Orientierung des Kanals gegeben.

Eine orientierte Stelle ist eine Stelle folgender Art: Sie gehört genau zwei Knoten an; sämtliche Aktionen des ersten Knotens führen das Objekt der Stelle in das duale über, und sämtliche Aktionen des zweiten Knotens tun dasselbe im umgekehrten Sinne. In der Graphendarstellung bezeichnen wir solche Stellen durch eine gerichtete Strecke.

Ein Netz ist kommunikationsfähig, wenn es Randstellen besitzt. Kommunikation zwischen zwei Netzen wird durch Identifizierung von Randstellen ermöglicht. Es entsteht ein Gesamtnetz, welches mit denselben Mitteln wie die Teilnetze beschrieben werden kann.

Wenn wir von Kommunikation mit einem Netz sprechen, so denken wir uns selbst in die Lage eines Teilnetzes versetzt, haben aber nicht unmittelbar die Möglichkeit, auf dieses Teilnetz dieselbe Form der Beschreibung anzuwenden, wie wir sie für Netze vereinbart haben. Wir müssen vielmehr die Aktionen, die wir auf die gemeinsamen Randstellen ausüben, umgangssprachlich ausdrücken. Dadurch werden Begriffe wie "Signal", "Information", "Bedingung", "solange bis " usw. auftreten.
Auf der anderen Seite können wir häufig ein Netz angeben, welches unser Verhalten, unsere Absichten, unseren Kenntnisstand usw. vollständig oder zu einem genau abgrenzbaren Teil exakt beschreibt. Der Vergleich zwischen umgangssprachlicher Formulierung und Netz liefert somit eine Art von Übersetzungsvorschrift zwischen Umgangssprache und Netzsprache, die bis auf die explizite und zum Teil abweichende Behandlung der zeitlichen Zusammenhänge den Sprachformen der symbolischen Logik (z.B. 16) ähnlich ist.

Es hat sich gezeigt, daß eine große Zahl von Netzen, die auf diese Weise erhalten werden, nicht konfliktfrei sind; oft läßt sich nachweisen, daß ein äquivalentes konfliktfreies Netz nicht existiert.
Beispiel: Wir verfolgen einen Plan von der Form

"Wenn A oder B , dann C "

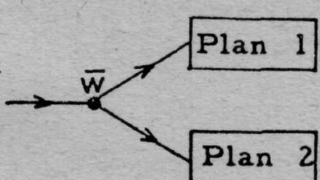
NA, NB, NC seien irgendwelche Netze, mit denen wir kommunizieren können. Es sei betont, daß die Kommunikation nur über die Randstellen stattfinden kann; falls wir die Möglichkeit zulassen wollten, Netze unter unserer Kontrolle durch Eingriff auf innere Stellen (d.h. Nicht - Randstellen) oder Knoten zu verändern, so würden wir zu unseren Definitionen in Widerspruch geraten.

Die Klausel "Wenn A" bedeute bei Interpretation des Planes: Um das Erfülltsein der Klausel zu prüfen, geben wir ein Signal an NA; d.h. wir ändern durch eine Aktion das Objekt einer Randstelle von NA , und merken uns, daß wir es geändert haben. Nun beginnen wir, die Randstellen von NA zu beobachten (möglicherweise die Randstelle mit geändertem Objekt als einzige) und bereiten eine Aktion vor derart, daß sie genau dadurch anwendbar wird, daß NA auf seine Randstellen gewisse Aktionen ausübt, welche das bzw. die Randobjekte verändern. Wir können auch zwei Aktionen vorbereiten, die einer "Ja" - oder "Nein" - Antwort von NA zugeordnet werden. Die Ausführung einer vorbereiteten Aktion stellt die Wahrnehmung der Antwort dar. Selbst wenn wir die Struktur von NA genau kennen, können wir auf keine Weise die Antwort innerhalb einer vorgegebenen "Zeitspanne" erzwingen. Diese Ausdrucksweise würde bedeuten: Wir haben eine zyklische Aktionskette, die nicht mit NA in Verbindung steht;

und wir fragen die Randobjekte von NA nach dem Takt dieser Uhr wiederholt ab. Die Möglichkeit des Abfragens bedeutet aber, daß die durch NA zur Antwort führende Aktionskette ebenfalls geschlossen wird; sie stellt also eine zweite unabhängige Uhr dar. Der Gleichlauf dieser Uhren kann nach P1 nur dadurch herbeigeführt werden, daß die Durchläufe der Zyklen kausal (konfliktfrei) voneinander abhängig sind; je nach Betrachtungsweise verliert also entweder das Wort "Abfragen" oder das Wort "Zeitspanne" hier seinen Sinn, wenn wir P1 anerkennen.

Dasselbe gelte für die Kommunikation mit NB.

"Dann C" bedeute, daß (bei Erfülltsein der Gesamtklausel "Wenn A oder B ") ein Signal an NC abgegeben wird. Eventuelle Rücksignale von NC mögen zu anderen Teilen eines Gesamtplanes gehören, welche mit dem betrachteten Plan durch einen Parallelitätsoperator "&" verknüpft seien, welcher besagen soll, daß sowohl der betrachtete Plan als auch die Gesamtheit der übrigen Teilpläne durchgeführt werden sollen. Die Aktivierung von so zusammengesetzten Plänen erfolgt über einen Knoten W :



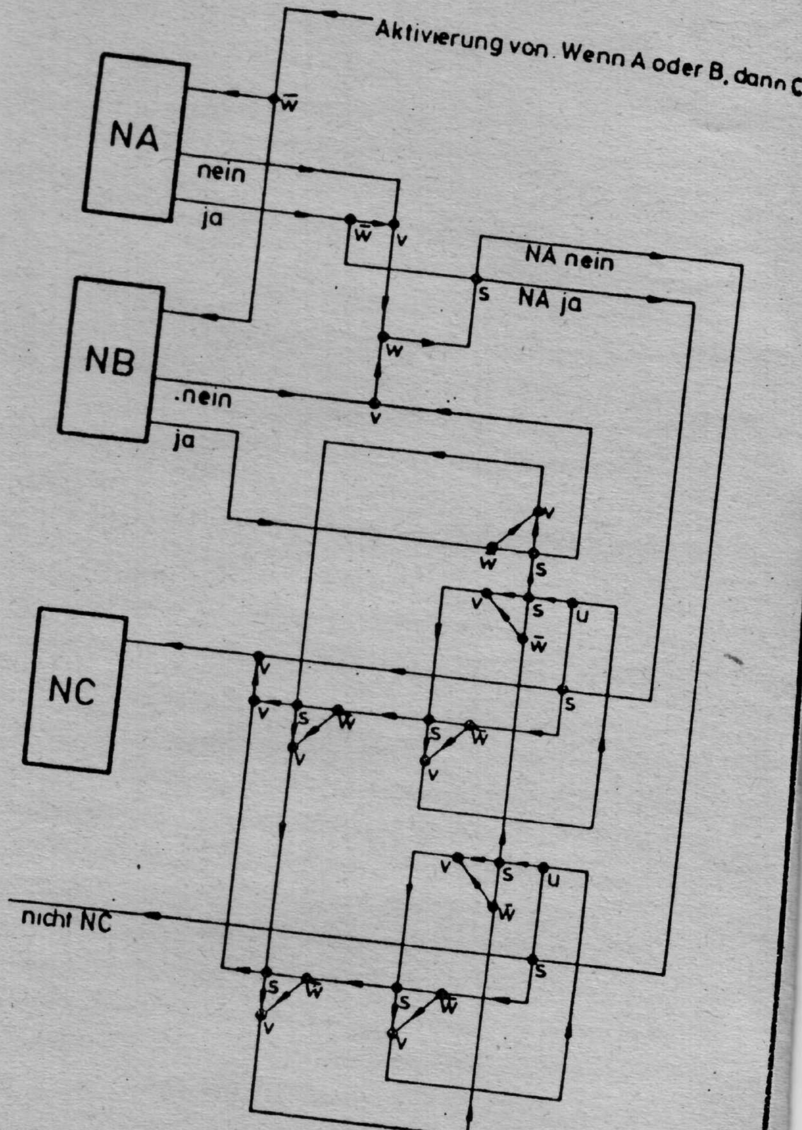
& [Plan 1] [Plan 2]

Nun müssen wir den Plan "Wenn A oder B, dann C " zerlegen. Der Sinn dieser Zerlegung ist hier also, eine Netzdarstellung des (umgangssprachlichen) "Oder" zu finden, womit eine exakte operationale Definition des "Oder" gewonnen wird. Dieser Funktor entspricht im allgemeinen nicht der Disjunktion des zweiwertigen Aussagekalküls, die sich, wie alle Funktionen dieses Kalküls, konfliktfrei darstellen läßt. -

Auf der folgenden Bildseite 68 geben wir eine konfliktfreie Darstellung der Disjunktion. Wir haben nicht die "einfachste" Darstellung (hinsichtlich der Zahl der Knoten, der Aktionen oder der Stellen) gewählt, sondern eine solche, aus der ersichtlich ist, wie ein Netz für jede Aussagenfunktion systematisch aufgebaut werden kann.

Das Netz besagt im wesentlichen folgendes: Nach Aktivierung des Gesamtplanes werden sowohl NA als auch NB aktiviert. Die durch die Abfragen erhaltenen Teilinformationen werden gespeichert und aus dem Vorgang der Speicherung werden Fertigsignale abgeleitet; wenn beide Fertigsignale vorliegen, werden die gespeicherten Informationen gemäß der Wertetafel der Disjunktion verknüpft und abgegeben.

Obwohl also z.B. das Vorliegen einer Ja - Antwort von NA allein schon das Endergebnis definiert und uns zur Entscheidung Ja -NC



Konfliktfreie Darstellung der Disjunktion

berechtigen würde, muß dieses Netz auf die NB - Antwort warten.
Dies entspricht genau den formalen Vorschriften zur Auswertung einer Wertetafel, wenn deren Eingänge unmittelbar den Antwortmöglichkeiten (von NA und NB) zugeordnet sind.

Die auffällige Komplexität des Netzes erklärt sich durch die Auswahl der Schaltelemente, deren jedes höchstens zwei Aktionen trägt. Wir erwähnen hier vorgreifend den Umstand, daß die Komplexität sich durch die Wahl eines einzigen Schaltelements (§5) mit nur einer Aktion wieder beseitigen läßt, und daß die Wahl dieses Schaltelements gerade mit den Erfordernissen von P2 zusammenfällt. Wir betonen damit nochmals die Willkürlichkeit aller Konstruktionen in §4, die nicht etwa Schaltungsvorschläge, sondern vorläufige Beweismittel sind.

Allgemeine Bemerkung
zum Verständnis der Netzkonstruktionen:

Die Darstellung von Netzen wie auf der vorigen Seite und wie im folgenden ist gemeint als abgekürzte Form eines Beweises für die Existenz einer Ableitung gewisser Kommunikationsformen aus einfacheren, durch Axiome und Definitionen vorgelegten Kommunikationsformen, den Aktionen.
Entsprechend dem elementaren Charakter der dargestellten Zusammenhänge ist eine voll-

ständige formale Ausarbeitung eines solchen Beweises äußerst umfangreich. Ebenso würde eine einigermaßen exakte umgangssprachliche Erläuterung aller Einzelheiten zu lang sein, um einleuchtend zu bleiben.
Andererseits wollen wir uns nicht auf Theoreme mit Alloperatoren berufen, damit die Beweise im strengsten konstruktiven Sinn verifizierbar sind.
Wir begnügen uns daher mit der Angabe einer Methode, sowohl den vollen Beweis zu erzeugen, als auch zur Einsicht in die Wirkungsweise eines Netzes zu gelangen:
Wir stellen uns vor, daß die Ausführung der Aktionen eines bestimmten Knotens unter unserer Kontrolle steht; aber nicht dadurch, daß die Aktionen von zusätzlichen Stellen abhängen, sondern so, daß wir die Rolle des Knotens sozusagen selbst übernehmen. Dann zählen wir sämtliche Folgen von möglichen Aktionen des Netzes auf und bestimmen deren Abhängigkeit von den eigenen Aktionen. Dasselbe tun wir für jeden Knoten des endlichen Netzes. -

Die folgende "operationale" Darstellung entspricht gewissen Interpretationen von mehrwertigen Aussagekalkülen, z. B. folgender Wertetafel des "Oder" :

A = F U T

B =	F	U	T
F	F	U	T
U	U	U	T
T	T	T	T

Tafel für

"A oder B"

mit der Interpretation für die Eingänge der Tafel:

F: Ein bestimmtes Experiment hat das Ergebnis a ("nein")

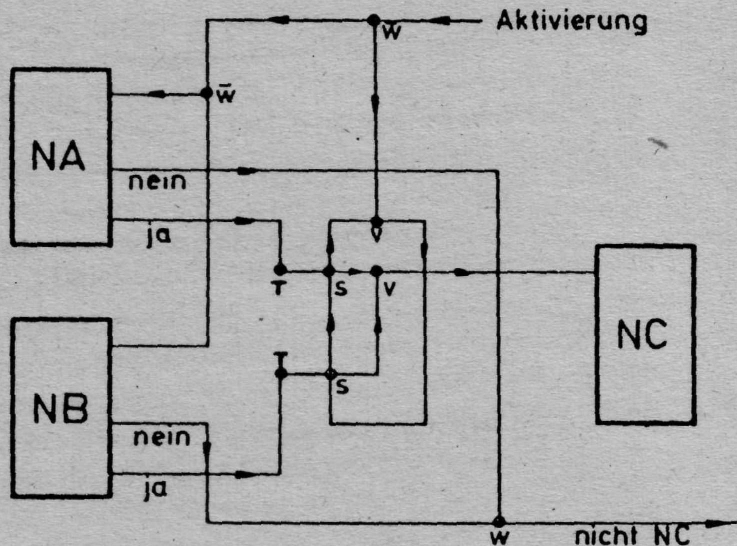
T: Ein bestimmtes Experiment hat das Ergebnis b ("ja")
wobei a und b einander ausschließen;

U: Das Ergebnis eines bestimmten Experimentes ist nicht bekannt.

Wir wollen voraussetzen, daß der Gesamtplan nur eine einmalige Auswertung der Klausel "Wenn A oder B " vorsieht, um Komplikationen zu vermeiden. Trotzdem wird das die Klausel auswertende Netz einen Konflikt enthalten, da ja durch irgendeinen Knoten entschieden werden muß, ob nun A , oder B , oder beide die Ursache zur Aktivierung von C darstellen. Diese Entscheidungsfälle gehören jedenfalls zu verschiedenen Aktionen. Wenn beispielsweise die Aktion, A allein zur Ursache von C zu machen, ermöglicht, aber noch nicht ausgeführt ist, so läßt sich nicht verhindern, daß B eintritt, bevor die Aktion eingetreten ist . Damit

wird eine weitere Aktion auf dieselbe Stelle ermöglicht oder aber die erste Aktion unwirksam gemacht, bevor sie eintrat: in beiden Fällen liegt ein Konflikt vor.
Folgende Netze geben die obige Wertetafel wieder, und zwar besagen sie etwas mehr als die Wertetafel allein, indem sie die Stellen möglichen Konfliktes in jeweils verschiedener Weise näher bezeichnen; d.h. sie geben zwei Verfahren zur Auswertung der Tafel an:

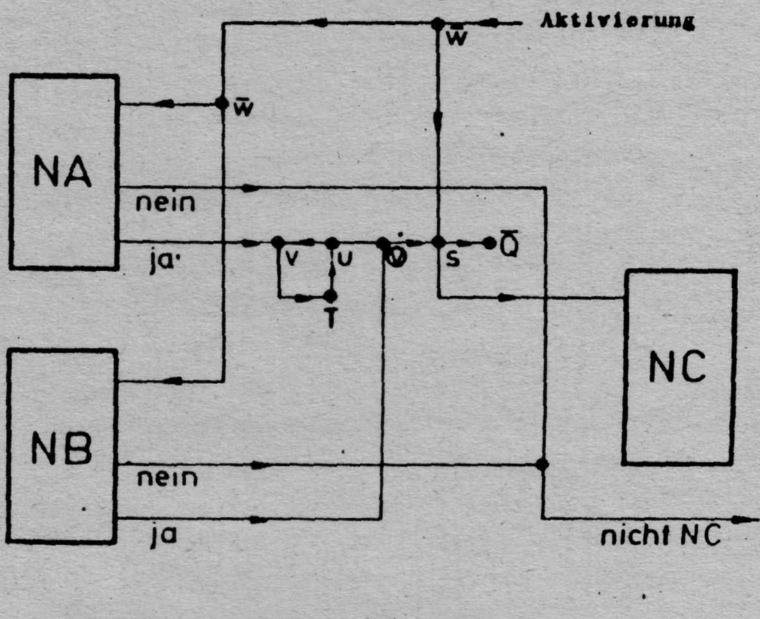
Erster Fall: NA und NB gleichberechtigt:



Die kritischen Objekte der beiden möglichen Konflikte liegen auf den T — S Stellen; die Konflikte beziehen sich auf Aktionen der S - Knoten.

Im zweiten Fall liegen kritische Objekte auf den Eingangsstellen des (V) - Knotens, und die Konflikte beziehen sich auf Aktionen von (V) und U :

Zweiter Fall: Vorrang von NB - Antwort "ja":



Der S - Knoten sorgt dafür, daß NC höchstens einmal aktiviert wird.

Mit diesen Beispielen sollte geschildert werden:

1) die Notwendigkeit der Zulassung von Konflikten bei der exakten Darstellung gewisser Kommunikationsformen;
2) die anfänglichen Schwierigkeiten bei der Konstruktion konfliktfreier Netze.

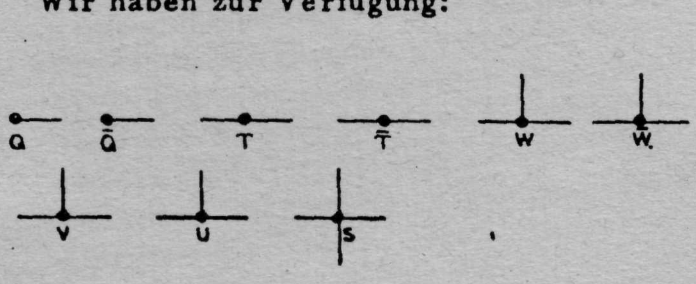
Wir wollen nun ein kommunikationsfähiges konfliktfreies Netz konstruieren, welches einer Turingmaschine entspricht.
Wir legen die Konstruktion so an, daß zugleich ersichtlich wird, wie konfliktfreie Netze für

1) beliebige Schaltwerke endlicher fester Größe
2) beliebig erweiterungsfähige Speicher

angelegt werden können.

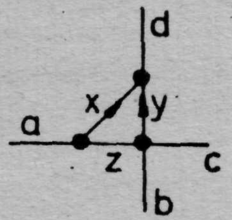
Auch hier wieder soll es nicht auf minimale Knoten- oder Stellenzahl, sondern auf Verallgemeinerungsfähigkeit ankommen.

Wir haben zur Verfügung:



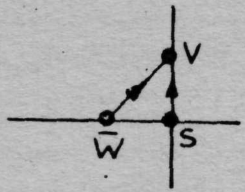
und definieren

N1:



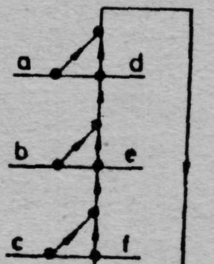
N1 (a b c d) = $\bar{W}$ (x z a)
V (x y d)
S (z b c y)

wofür wir kürzer angeben werden:



N1 wird vorwiegend als Teil von N2 gebraucht:

N2:



N2 kann dazu dienen, mehrere Aktionsketten vorübergehend zu verknüpfen, um beispielsweise ein Netz Ni von verschiedenen Stellen her zu aktivieren. Bedingung ist jedoch, daß jeweils höchstens eine der Aktionsketten aktiv ist; diese Bedingung muß von der Umgebung von N2 erfüllt werden. Wir werden solche Bedingungen durch KS3 - Formeln ausdrücken, wodurch der Begriff der Kommunikationsform einen exakten Sinn erhält.

KS3 habe das Axiom Σ und die Produktionen:

$\Sigma \leftarrow \nabla \Sigma \Sigma$	(Auswahl oder Konflikt)
$\Sigma \leftarrow / \Sigma \Sigma$	(Kausalität)
$\Sigma \leftarrow \& \Sigma \Sigma$	(Parallelität)
$\Sigma \leftarrow * \Sigma$	(Iteration, Signalzyklus; "Uhr")
$\Sigma \leftarrow \emptyset$	(Widerspruch; Nicht-Hinreichen des Netzes)

$$\left.\begin{array}{l} \Sigma \leftarrow a \\ \Sigma \leftarrow b \\ \ldots \end{array}\right\} \text{Aufzählung der Elemente von X}$$

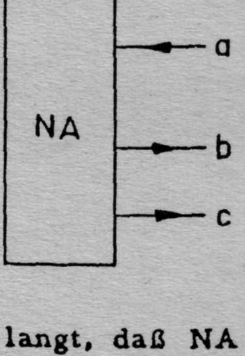
Die KS3-Formeln sollen den zeitlichen und strukturellen Zusammenhang eines Netzes mit seiner Umgebung so weit beschreiben, wie er sich durch Ereignisse in den Randstellen allein ausdrückt. Diesen Zusammenhang nennen wir eine Kommunikationsform für die betreffenden Stellen. Unter einem Ereignis sei das Auftreten eines ausgezeichneten Objektes verstanden.

Wir definieren:
X enthalte genau alle möglichen Randereignisse eines Netzes.

Wir fordern zunächst (ohne wesentliche Einschränkung der Allgemeinheit), daß die betrachteten Randstellen durch die Kommunikation zu orientierten Stellen werden. Die Randereignisse zerfallen dann in zwei Klassen: Eingangs- und Ausgangsereignisse. (Duale Vertauschung möglich).

Wir verzichten auf die weitschweifige umgangssprachliche Erläuterung der Interpretation, weil sie analog zu der von KS1 ist, mit dem Unterschied, daß Kommunikationsformen nicht nur Folgen von Ereignissen, sondern wegen des Parallelitätsoperators auch Verbände von Ereignissen enthalten. Wir erklären durch Beispiele:

$* / a \nabla b c$ gibt die Kommunikationsform (K-Form) einer Informationsquelle an. Sich an diese K-Form zu halten, bedeutet, dem umgebenden Netz eine Struktur zu geben, die es unmöglich macht, nach einer Abfrage (Aktivierung) an NA eine zweite abzugeben, bevor die Antwort auf die erste erschienen ist; ferner wird verlangt, daß NA keine Antwort gibt, ohne gefragt worden zu sein, daß es pro Frage genau eine Antwort gibt und nicht beide Antworten



"zugleich" oder "nacheinander" auf dieselbe Frage.

$\ast$ a ist eine K-Form von Q und $\bar{Q}$,

$\ast$ / a b ist eine K-Form von T und $\bar{\bar{T}}$,

$\ast$ / & a b c ist eine K-Form von W (und von $\bar{\bar{W}}$)

$\ast$ / a & b c ist eine K-Form von $\bar{\bar{W}}$ (und von W),

$\ast$ ∇ / / a d / b c $\ast$ / b d ist eine K-Form von N1

K1: $\ast$ / ∇ ∇ a b c ∇ ∇ d e f ist eine K-Form von (N2 + Ni)

K2: $\ast$ ∇ ∇ / a d / b e / c f ist eine K-Form von (N2 + Ni)

Offensichtlich besagt K2 mehr als K1 ; wenn wir K2 kennen, können wir daraus K1 ableiten; Der Formalismus der Ableitung liegt außerhalb der für diese Arbeit gesteckten Grenzen.

Wir wollen nur ausdrücken, daß über den Formeln von KS3 eine nicht-symmetrische, reflexive, transitive Relation des Enthaltenseins erklärt werden kann, wie hier:

$$K2 \supseteq K1$$

Wir nennen K2 die <u>engere</u>, K1 die <u>weitere</u> K-Form und untersuchen die Algebra der

K-Formen hier nicht näher.

Wir wollen nun ein Netz konstruieren, welches die K-Form

K3 :
/∇/ab//cd/ab∇/ab/*/ce/ab

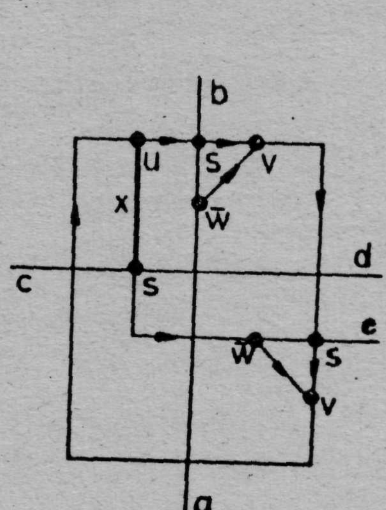
und damit die weitere K-Form:

*∇/ab/c∇de besitzt.

Die engere K-Form besagt in Worten, daß das zu konstruierende Netz den Charakter einer <u>1-Bit-Speicherzelle</u> besitzt, welche durch Aktivierung von c abgefragt werden kann; das Ergebnis einer Abfrage ist d, wenn die Gesamtzahl der /ab -Durchläufe gerade ist, und e, wenn sie ungerade ist.

Folgendes Netz leistet das Verlangte:

N3:



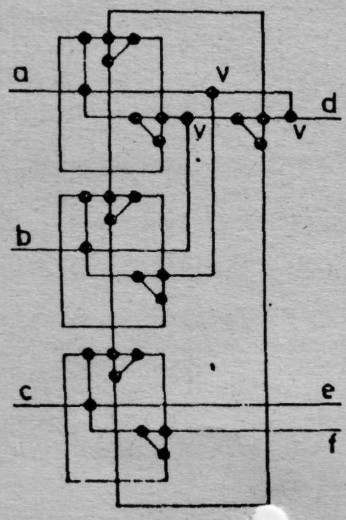
Hier ist deutlich zu erkennen, daß der Begriff "Zustand von N3", welcher durch das Objekt der mit x bezeichneten Stelle gekennzeichnet wird, an die Kommunikationsform gebunden ist, unter der wir das Netz betrachten.

Wir konstruieren eine weitere Art von Speicherzelle N4 , welche ohne Abzählung von / a b - Durchläufen (mod 2) gesetzt werden kann; mit der weiten K-Form

$\ast \nabla / \nabla$ a b d / c ∇ e f

(K-Formen, in denen jedes Randereignis genau einmal genannt ist, mögen normal heißen)

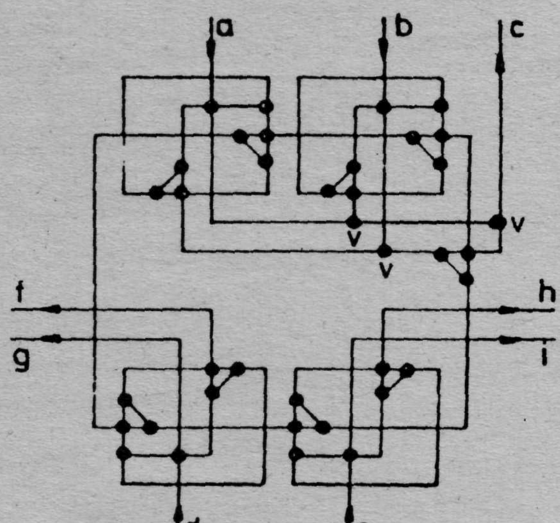
N4:



Da das Netz aus einer Kombination von N1, N2, N3 - Exemplaren besteht, geben wir nur die hinzukommenden Knoten an.
Eine K3 entsprechende K-Form, welche ∇ e f explizit macht, läßt sich mit denselben Überlegungen wie bei K3 finden. Wir benutzen nur noch normale K-Formen.

Weiterhin haben wir Netze N5 und N6, die ein Abfragen bzw. Setzen von Speicherinhalten von mehreren (hier zwei) Signalzyklen her erlauben:

N5:

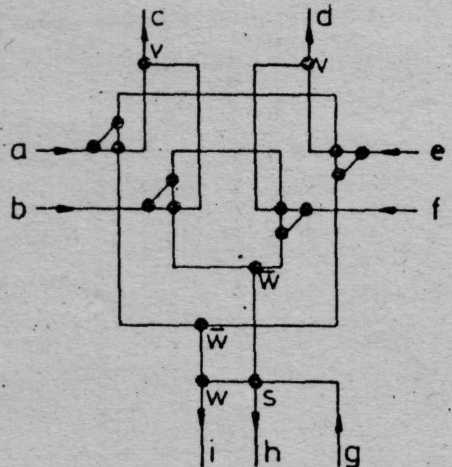


Normale K-Form:

$$\ast\,\underbrace{\nabla\nabla/\nabla\ \ a\,b\,c}_{\text{Setzen}}\underbrace{/\,d\,\nabla\ \ f\,g}_{\text{Abfragen 1}}\underbrace{/\,e\,\nabla\ \ h\,i}_{\text{Abfragen 2}}$$

Wir haben die Randstellen hier bereits orientiert, obwohl sie streng genommen erst durch den Einbau des Netzes in ein größeres die Orientierung erhalten. Dies soll die Anschaulichkeit erhöhen und ausdrücken, daß die K-Form, unter der wir das Netz betrachten, diese Orientierung fordert. -

N6:



Dieser Speicher hat eher den Charakter eines Puffers für 1 Bit, wie aus der K-Form folgt:

$$\ast / \underbrace{\nabla\ / \nabla\, a\, b\, c}_{\text{Setzen 1}} \underbrace{/ \nabla\, e\, f\, d}_{\text{Setzen 2}} \underbrace{/ g\, \nabla\, h\, i}_{\text{Abfragen}}$$

Setzen (entweder von rechts oder von links) und Abfragen müssen abwechselnd erfolgen, beginnend mit einem Setzen. Es ist klar, daß sich nach dem Muster von N4 Speicher bilden lassen, die dieser Einschränkung nicht unterliegen. -

Um ein allgemeines Schaltwerk in Netzform zu bringen, müssen wir zunächst

1) eine passende Kommunikationsform festlegen,
2) eine der verschiedenen Darstellungsformen für Schaltwerke auswählen.

Es ist bekanntlich keine Beschränkung der Allgemeinheit, für Eingang und Ausgang eines Informationswandlers nur Alphabete mit zwei Zeichenformen anstelle von beliebigen endlichen Alphabeten zuzulassen. Wir ordnen den Zeichenformen die vier Randstellen eines Netzes zu: a b c d.
Der Zeitschritt des Schaltwerkes sowie die übrigen Verabredungen über seine Wirkungsweise lassen sich ausdrücken durch die normale K-Form

K7: $\ast / \nabla\ a\, b\ \nabla\ c\, d$

Die <u>Entwicklung der K-Form nach den ∇</u> führt zu einer engeren K-Form, die das Verhalten eines speziellen Schaltwerkes wiedergibt.

Als definierende Darstellung des Schaltwerks wählen wir den Zustands-Übergangsgraphen bzw. die zugehörige Tabelle

q_1	a	i_{11}	q_{11}
q_1	b	i_{12}	q_{12}
q_2	a	i_{21}	q_{21}
.			
.			
.			
q_n	b	i_{n2}	q_{n2}

mit $i_{jk} = \nabla\, c\, d$

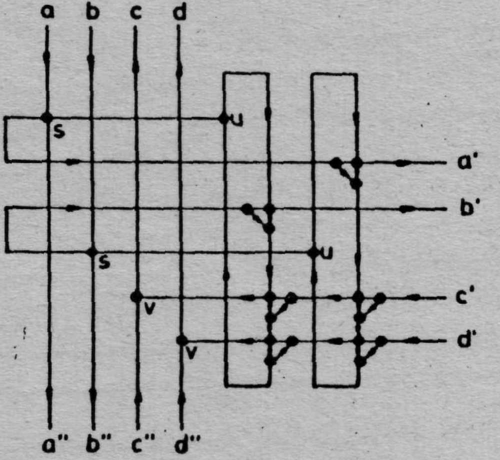
$q_{jk} = q_l,$
$1 \leq l \leq n$

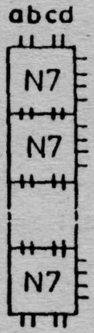
welche unvollständig sein darf.

Interpretation einer Zeile q_m b c q_l :
Wenn das Schaltwerk sich im Zustand q_m befindet und zugleich das Eingangssignal b besitzt, so gibt es das Signal c ab und geht durch den nächsten Zeitschritt in den Zustand q_l über.

Wir führen für jede Zeichenform q_m , die in der Tabelle auftritt (also nicht für jedes Zeichenexemplar q_m) ein Exemplar von N7 ein :

N7:





 sind Exemplare von N1.

Wir verknüpfen diese N7-Exemplare miteinander in der Reihenfolge, in der die q_m beim Lesen der Tabelle erstmalig angetroffen werden, und sobald wir ausdrücken wollen, daß das Netz nicht mehr erweitert werden soll, schließen wir die letzten Randstellen der Art a'', b'', c'', d'' durch $\overline{Q}$.

Die Bedeutung dieser Schließung liegt vor allem im Verzicht auf Zugänglichkeit, d.h. auf Möglichkeiten der Kommunikation. Die Schließung ist nicht lediglich ein formaler Akt mit dem Zweck, willkürliche Verabredungen über unsere Konstruktionen einzuhalten.

Durch die Ausführung der Schließung werden nämlich offenbar Randstellen des konstruierenden Subjektes frei, welches damit neue, nicht-redundante Kommunikationsmöglichkeiten erhält. Derartige Überlegungen erhalten natürlich erst bei voller Formalisierung des Konstruktionsvorganges ihr ganzes Gewicht; wir führen sie nur an, um einen Irrtum über die Bedeutung der Schließung abzuwenden. - Durch die Verknüpfung ist eine N7-Kette entstanden, deren Glieder eindeutig den q_m in der Tabelle zugeordnet sind. Das einem N7-Glied zugeordnete q_m heiße seine Marke.

Weiterhin verknüpfen wir die Kette zum Schaltwerk nach folgender Vorschrift:

Für jede Zeile der Tabelle bringen wir genau einen V-Knoten an mit der Verknüpfung, wenn die Zeile

q_m b c q_l lautet:

Der eine Eingang des V-Knotens wird identifiziert mit dem b′ - Ausgang des N7-Netzes mit der Marke q_m;

der zweite Eingang bleibt frei;
der Ausgang des V-Knotens wird identifiziert mit dem c-Eingang des N7-Netzes mit der Marke q_l, und wenn dieser Eingang nicht mehr frei ist, mit dem freien Eingang desjenigen V-Knotens, der als letzter an den c-Eingang oder dessen V-Knotenkette angefügt wurde.

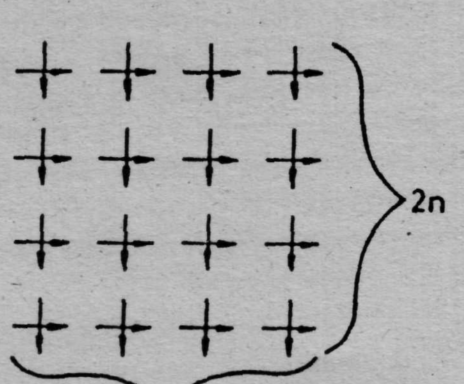
Entsprechend für

$q_m\,a\,c\,q_1$, $q_m\,a\,d\,q_1$, $q_m\,b\,d\,q_1$.

Nach Behandlung der letzten Zeile schließen wir alle Randstellen außer a b c d durch $\bar{Q}$. -

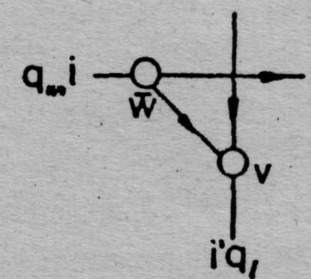
Hierbei war vorausgesetzt, daß jedes vordere Zeichenpaar $q_m\,i$ höchstens einmal in der Tabelle auftritt, wie es gewöhnlich von Schaltwerken gefordert wird. Von dieser Voraussetzung können wir uns befreien, wenn wir die obige Verknüpfungsvorschrift durch eine Matrixvorschrift ersetzen, welche so einfach ist, daß wir sie nur andeuten:

Wir legen neben die N7-Kette, welche n Glieder habe, eine "Matrix" von Stellen mit je 2n Zeilen und Spalten, und gebildet aus $2\cdot(2n)^2$ zu orientierenden Stellen:



Die Zeileneingänge ordnen wir den a', b' Ausgängen der N7-Kette zu, die Spaltenausgänge den c', d' Eingängen.

Nun führen wir für jede Zeile genau je einen $\bar{W}$- und V-Knoten ein, so:



Nach Behandlung der letzten Tabellenzeile schließen wir die Lücken zwischen benachbarten Randstellen im Innern der Matrix durch T-Knoten und die verbleibenden Randstellen mit $\bar{Q}$.

Bei dieser Art der Verknüpfung können Konflikte auftreten, entsprechend den "sneak paths" der Relaistechnik, aber auch solche allgemeinerer Art. Andererseits können auch dann, wenn die Tabellen die oben genannte Voraussetzung nicht erfüllen, unter Umständen Netze entstehen, die unter der K-Form K7 konfliktfrei sind. Damit dieser Sachverhalt richtig beurteilt werden kann, erwähnen wir ohne Beweis:

Zu jedem kommunikationsfähigen Netz gibt es eine K-Form, die bei Kommunikation Konflikte zuläßt.

Die Matrixdarstellung der Schaltwerks - Teilnetze legen wir im weiteren auch bei der Konstruktion eines Netzes für eine universale kommunikationsfähige Turingmaschine zugrunde, mit dem Unterschied, daß nicht alle mit $\bar{Q}$ geschlossenen Randstellen der Matrix so geschlossen werden. Auf diese Weise können wir über die Randstellen der Matrix mit der Maschine kommunizieren, und zwar auch in irregulären Sprachen, und dies nach Belieben konfliktfrei.

Mit dem fest einzubauenden Universalprogramm beschäftigen wir uns nicht; es läßt sich bekanntlich durch eine endliche Tabelle ausdrücken. Wir denken uns diese Tabelle durch feste Standardprogramme in der Programmsprache der Turingmaschinen (15) für Ein- und Ausgabe, Bandlöschen und Zustandsabfrage ergänzt und in Matrixform aufgestellt.
Es verbleibt die Aufgabe, die Funktionsweise des "Lesekopfes" und des Bandes in Netzform auszudrücken.

Hier handelt es sich nun nicht mehr darum, eine vorgegebene endliche Zahl von aktionsfähigen Elementen zur Lösung einer Gesamtaufgabe zu organisieren, sondern wir müssen eine Form der Organisation finden, an der unbeschränkt viele Elemente teilnehmen können, die aber mit endlichen Mitteln derart ausgedrückt werden kann, daß jedes Element nur mit einer gleichmäßig beschränkten Zahl von stets denselben unmittelbar benachbarten Elementen mit gleichmäßig beschränkten Mitteln kommuniziert.

Die Existenz einer solchen Organisationsform können wir nur durch Konstruktion nachweisen.

Da der Rand des Lesekopfes mit Band mit den Stellen a b c d eines Schaltwerks kommunizieren soll, muß der Lesekopf die K-

Form $\ast / \nabla$ c d ∇ a b besitzen, wenn wir die zu identifizierenden Stellen mit dem gleichen Symbol bezeichnen.
Wir fragen uns zunächst, ob diese Form ausreicht.

Für die vom Bande zu lesenden Zeichen, zusammen mit ihrer zeitlichen Abgrenzung, reicht jedenfalls ∇ a b hin; es ist bekannt, daß das Band einer Turingmaschine nur zwei Zeichenformen benötigt ("blank" und "markiert", oder "Null" und "Eins").

Wir müssen noch die Signale

- l : betrachte nächstes Zeichen links
- r : betrachte nächstes Zeichen rechts
- n : ersetze das betrachtete Zeichen durch "Null"
- e : ersetze das betrachtete Zeichen durch "Eins"

auf ∇ c d abbilden. Das ist so möglich:

Da aufgrund der Konstruktion von N7 stets bekannt ist, welches Zeichen zuletzt betrachtet wurde, können wir n und e zusammenfassen zu u und w :

u : das betrachtete Zeichen bleibt unverändert;
w : das betrachtete Zeichen wird gegen das duale ausgewechselt.

Wir definieren:

c ≡ l , d ≡ / w r

und haben damit die Unterprogramme

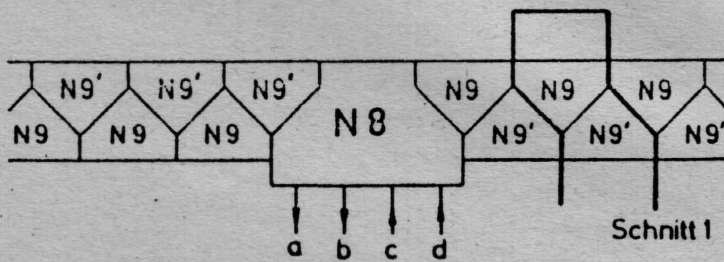
l = c
r = / / d c d
u = / / / d c d c
w = / d c

Damit ist die Programmsprache für Turingmaschinen in die Schaltwerkssprache übersetzt, und die vierstellige normale K-Form

$\ast$ / ∇ c d ∇ a b als hinreichend bewiesen.

Wir haben oben bereits einen von verschiedenen Signalketten her abfragbaren 1-Bit-Speicher (N5) und einen passenden 1-Bit-Puffer (N6) angegeben, die wir nun verwenden wollen. (Aus der Konstruktion ist leicht zu ersehen, wie ein n-Bit-Speicher oder Puffer anzulegen wäre).

Unser Plan ist nun, den Lesekopf mit dem betrachteten Feld des Bandes in ein Netz N8 und die beiden Resthälften des Bandes in je eine Netzkette aus Netzen N9 zu übersetzen:



Denken wir uns N8 entfernt, so verbleiben zwei Speicher von spezieller Art, deren beschränkte Formen in der Technik des Programmierens wohl bekannt sind ("Stapel" oder "Keller"; wir benutzen das Wort LIFO, last-in-first-out, zur Kennzeichnung des Prinzips.) Wie bei jeder Speicherung allgemeiner Art hat man auch beim Lifospeicher den Plan, so zu kommunizieren:

Kl1 : ✳∇ / c ∇ d e / ∇ h i g

mit ∇ h i als zu stapelnde Information, c als Abholsignal für die zuletzt gestapelte Information, ∇ d e als Antwort auf c, und g als Fertigsignal beim Stapeln.

Nach den Ergebnissen von §2 ist es aber unmöglich, daß in allen vollständig trennenden Schnitten wie Schnitt 1 eine so einfache Kommunikationsform herrscht: denn wir könnten sie sonst als KS1-Formel interpretieren und müßten feststellen, daß eine endliche KS1-Formel das offensichtlich rekursive Lifo-Prinzip wiedergibt. Es ist daher notwendig, daß die K-Formen der Schnitte einen Parallelitätsoperator enthalten, der nicht eliminiert werden kann.

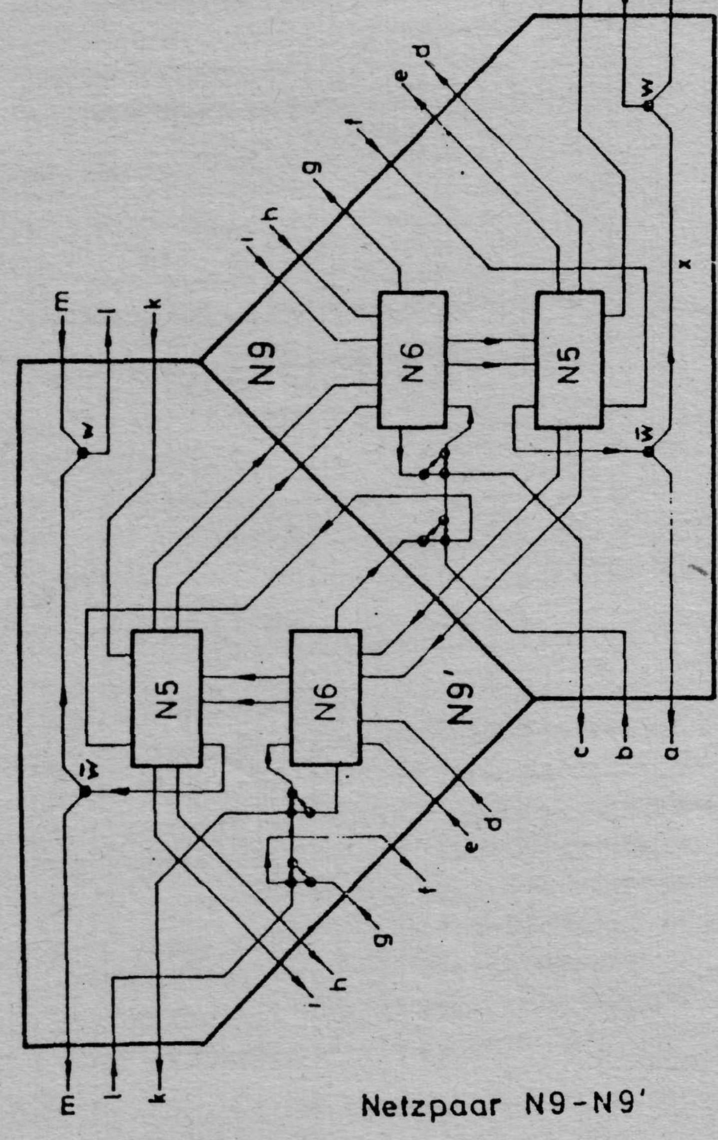
Es gibt viele Möglichkeiten, passende K-Formen für die Schnitte intuitiv zu wählen. Sie sind mit Hilfe eines hier nicht zu beschreibenden Kalküls überführbar. Wir geben sogleich eine K-Form an, die nur ein einziges Exemplar des &-Operators enthält und dadurch die Nicht-Eliminierbarkeit des & besonders deutlich hervortreten läßt.

K9:

$\times$/&∇ ////c∇demlk////∇ higmlfab

Diese K-Form ermöglicht es im Gegensatz zu entsprechenden "synchronen" Konstruktionen über KS1, daß von jeder Aktivierung des Speichers c bzw. ∇ h i eine Aktionskette von gleichmäßig beschränkter Länge zum Netzrand führt, unabhängig von der Länge des Lifospeichers.

Im folgenden Bild geben wir ein Netz N9 an. Das Netz N9' entsteht daraus durch Spiegelung, nicht durch Drehung. Jedes Paar N9-N9' von miteinander kommunizierenden Netzen genügt an beiden 12-tupeln von Randstellen der angegebenen K-Form konfliktfrei.



Netzpaar N9-N9'

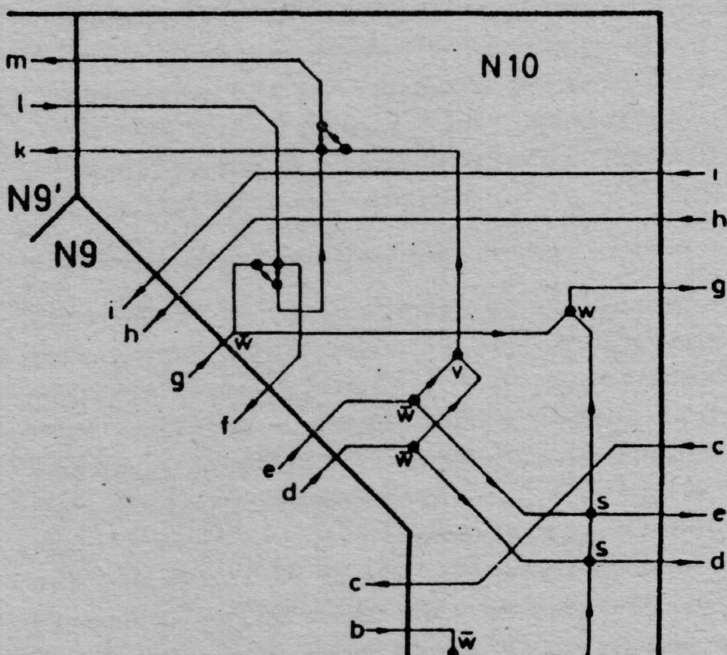
Vom Rand eines solchen Lifo-Speichers au... gehend können wir die K-Form

✱ ∇/ c ∇ d' e' /∇ h i g'

aus der K-Form

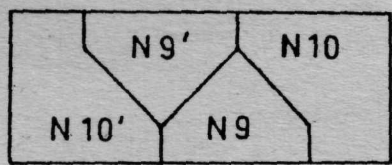
K9:
✱/&∇////c∇demlk////∇higmlfab

abtrennen durch ein Netz:

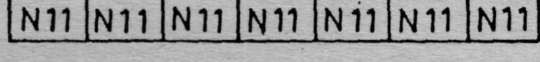


Daher scheint es, als ob man durch eine solche Ergänzung den &-Operator aus dem ganzen Lifospeicher eliminieren könnte, indem man, vom Rande her beginnend und von Netzpaar zu Netzpaar N9 - N9' fortschreitend, Netzpaare N10 - N10' zwischenschaltet. Nehmen wir an, der Lifospeicher habe eine endliche Länge, so kommt dieser Prozeß an ein Ende. Der Speicher besteht dann aus einer Kette von Netzen N11 :

N11 :



(N10' ist ein zu N10 duales Netz)



wobei die Schnitte zwischen den N11 die &-freie Speicher-K-Form

K11 : $\ast\nabla / c\nabla\; d\, e / \nabla\; h\, i\, g$

besitzen.

Der grundsätzliche Unterschied zwischen beiden Arten von Lifospeichern besteht in folgendem: Wenn am Rand einer für iterative Benutzung angelegten N11-Kette eine Anfrage gestellt wird, so ist es unmöglich, daß eine Antwort erscheint, bevor nicht eine Aktionskette sämtliche N11-Exemplare durchlaufen hat; außerdem muß das Ende des Speichers durch einen Kommunikationspartner (Netz) der K-Form K11 besetzt sein. Wenn dieser Kommunikationspartner einen Plan besitzt, der durch ein endliches Netz mit der Randform K11 ausgedrückt werden kann, so muß entweder dieses Netz in getrennte Netze zerfallen oder Konflikte besitzen, entsprechend der "freien Wahl" des Partners am Eingang der Kette, die ∇ zu bewerten.

Beispiel: Der Plan des Endpartners lautet:

> "Empfangene Information (d.h. aus dem Lifospeicher herausgeschobene Information ∇ e d) geht verloren; bei Anfrage vom Speicher ist stets die Antwort h zu geben (wie beim Linksschieben in einem Rechenregister von rechts Nullen nachgezogen werden und beim Rechtsschieben die herausgeschobenen Stellen verloren gehen.)"

In die Sprache der K-Formen übersetzt heißt dies, daß die Bewertung der ∇ -Exemplare für die Kommunikationspartner folgende ist

$$* \nabla / c \nabla d e / \nabla h i g$$

Nummer: 1 2 3

∇ 1 und ∇ 3 stehen unter Kontrolle ("im Belieben") des Eingangspartners; das liegt im Sinn der Speicherbenutzung: man hat die

Wahl, abzufragen oder zu speichern, und falls man sich für Speichern entschieden hat, steht eine Wahl offen, was gespeichert werden soll. ∇ 2 bringt Informationsgewinn und seine Bewertung unterliegt nicht der Kontrolle des Eingangspartners.

Beim Netz, das den obigen Plan des Endpartners darstellen soll, ist es umgekehrt: nur ∇ 2 unterliegt seiner Kontrolle. Der Plan verlangt, daß

/ c ∇ d e   zerfällt in   / c d   und

/ ∅ e ,   d.h. $\bar{Q}$ (e) ;

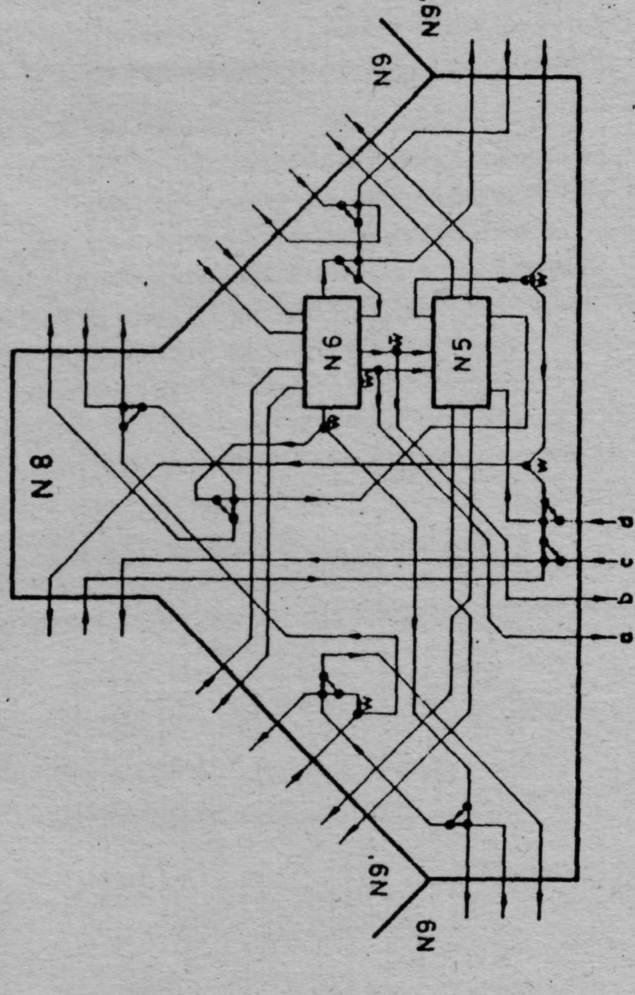
ferner zerfällt   ✱∇/ ... / ...   in

✱/ ...   und   ✱/ ...

Daher kann ein etwa mit c am Eingang beginnender Signalzyklus nur über / c d am Ausgang konfliktfrei geschlossen werden, und eine gedachte "unendliche" N11-Kette wäre überhaupt nicht arbeitsfähig. -

Anders beim ersten Lifospeicher, der nur aus Netzpaaren N9 - N9' besteht: für jede Aktivierung am Eingangsrand pflanzt sich eine "Welle" von Aktionen bis ans Ende hin fort, welche die verschränkte Information aus ∇ 1 und ∇ 3 mit sich führt. Nur durch den &-Operator (d.h. durch Einführung verschränkter W-$\bar{W}$-Zyklen) ist es möglich, die gewünschten Rücksignale unabhängig vom Fortschreiten der Welle und damit unabhängig von der Länge des Speichers zu machen. -

Es fehlt nun noch die Konstruktion des Lesekopfes einer Turingmaschine, zusammen mit dem betrachteten Feld ihres Bandes (N8) :

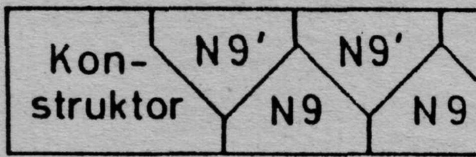


Damit ist die Konstruktion einer Netzdarstellung für den Informationsfluß in einer Turingmaschine beendet. Zu diskutieren sind noch die Bedingungen an den Enden des Bandes. Auch dort muß die K-Form K9 eingehalten werden.
Da das Band aber keine Information verlieren darf und über die Verknüpfung der Randstellen am Bandende nichts ausgesagt wird, muß K9 trivial erfüllt sein, das heißt durch das Fehlen von Signalen.

Diese Notwendigkeit liefert uns nun eine Konstruktionsvorschrift für derartige unbeschränkte Speicher, die sich offenbar dadurch auszeichnet, daß sie einen endlichen Zyklus besitzt und daß das Durchlaufen dieses Zyklus völlig unabhängig von den Arbeitszyklen des konstruierten Netzes, hier des Bandes, ist. In diesem Fall bedeutet das, daß der Konstruktor (13, 299) mit dem Benutzer der Maschine nicht zu kommunizieren braucht.

Bei der Konstruktion einer Turingmaschine hat man also neben einem endlichen Netz zwei gleichartige N9 - N9' - Konstruktoren zu erzeugen; die Aktivierung der Konstruktoren beendet zugleich die Kommunikationsmöglichkeit mit ihnen.

Der Plan eines Konstruktors muß lauten, wenn wir etwa die Situation



der linken Bandhälfte als Anfang nehmen:

Cl: ✱/////b a (N9) l m (N9)

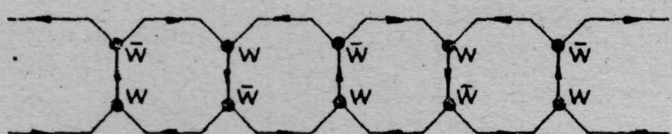
wobei (N9) eine konstruktive K-Form ist, die die Erzeugung eines N9-Netzes bedeutet. (Um sie zu definieren, müßten wir den Bereich der bis hierher benutzten Begriffe erweitern, um die Erzeugung neuer Stellen und den Vorgang der Verknüpfung zu beschreiben. Statt dessen weisen wir nur darauf hin, daß die formale Beschreibung dieses Prozesses jedenfalls endlich ist, und daß sie ohne Angabe einer Interpretation wenig Wert hätte. -)

Nach einem Durchlauf von Cl ist wieder dieselbe Situation entstanden; dies erklärt die Berechtigung des ✱ -Operators.

Wesentlich sind nun die Teile / b a bzw. / l m von Cl . Ihre Interpretation verdeutlicht die Unabhängigkeit des Konstruktors:

Die Netze N9, N9' werden so konstruiert, daß sie nur ein ausgezeichnetes totales Bit enthalten: die "Null"-Information in N6 . Dieses totale Bit ist nicht charakteristisch für das Netz: es kann ja nach der Bestimmung des Netzes verschoben und ersetzt werden. Dagegen wird durch b ein totales Bit in das N9 - bzw. N9' - Netz eingeführt, welches dem Netz immer erhalten bleibt und es von rechts her aktivierungsfähig macht. Dieses charakteristische ausgezeichnete totale Bit (Eigentobit) wird durch das Ereignis b eingeführt, im $\bar{W}$-Knoten von N9 (siehe Bild von N9) durch a dem Konstruktor nach links bestätigt, und befindet sich an der Stelle x zwischen $\bar{W}$ und W, sooft die K-Form n-mal vollständig durchlaufen ist ($n \geq 0$). Seine Lage gibt jeweils die betrachtete Stelle der K-Form an. Der Lauf des Eigentobits durch die K-Form ist analog zu dem Algorithmus zur Auswertung von KS2-Formeln (Seite 26).

Der Beweis für die konfliktfreie Funktionsfähigkeit sowie die Einsicht in Arbeitsweise des Lifospeichers ergibt sich aus der Betrachtung der charakteristischen Wege der Eigentobits:



Diese Figur wird aus der Speicherstruktur durch Kontrahierung der ∇ und / gewonnen.

Auf jedem Zyklus befindet sich stets genau ein Eigentobit. Wird ein Zyklus ohne Eigentobit am Ende angefügt, so beeinflußt er die Aktivität der übrigen Kette überhaupt nicht; daher können an diesem Zyklus Konstruktionen vorgenommen werden, genau entsprechend den Produktionen von KS3, und sobald die Konstruktionen beendet sind, kann dem Zyklus sein Eigentobit erteilt werden wonach er an den Aktivitäten der Kette teilnehmen kann, ohne aber die Konstruktion auf einem nächsten Zyklus zu stören, da ihn sein Eigentobit ja nicht verlassen kann.

Damit ist die Topologisierung des Informationsflusses in einer Turingmaschine unter Berücksichtigung des Postulats P1 beendet. Insbesondere ist gezeigt worden, daß die K-Form des Lesekopfes $\times\!\!\!\!\!\leftarrow / \nabla\ c\ d\ \nabla\ a\ b$ und damit die K-Form des Randes der Universalprogramm-Matrix unabhängig von der Tätigkeit der Konstruktoren und damit unabhängig von der Bandlänge ist, ohne daß das Gesamtnetz [Maschine + Konstruktoren] Konflikte besitzt.

Es gibt daher kein konfliktfreies Experiment (Algorithmus) am Programmrande der Turingmaschine, welches die Bandlänge festzustellen gestattet.

In diesem Sinne ist das angegebene Netz durch den Akt der Ablösung der Konstruktoren einem unendlichen Netz äquivalent; d.h. es ist imstande, irreguläre Mitteilungen wahrzunehmen.

§ 5.

Die Konstruktion des Netzes für eine Turingmaschine hat uns vorwiegend dazu gedient, das Enthaltensein der Theorie der synchronen Automaten in einer allgemeineren Theorie der Kommunikationsformen zu beweisen.

Wir müssen nun noch über die Anwendbarkeit und über den weiteren Ausbau einer solchen Theorie sprechen.

Wie in §1 erwähnt wurde, bestehen diejenigen Anwendungen der Theorie, die den Anlaß zu dieser Arbeit gegeben haben, in Methoden zum Entwurf und zur Programmierung von Informationsmaschinen.

Einige logische Grundlagen zu diesen Anwendungen sind in §4 durch explizite Vorlage geschaffen worden; es fragt sich nun noch, ob es möglich ist, den Formalismus der Netze in geeigneter Weise zu interpretieren; ob es also geeignete materielle Gegenstände gibt, welche genau das Verhalten der intuitiv eingeführten Schaltelemente von §4 besitzen.

Zunächst ist es klar, daß wir jedes der gezeigten konfliktfreien Netze dadurch realisieren können, daß wir die Knoten mit Personen besetzt denken, welche den Auftrag haben, gemäß den Aktionen des Knotens zu handeln. Die B-Struktur des Netzes wird durch eine räumlich-topologische Struktur der Kommunikationsmöglichkeiten zwischen den Personen ausgedrückt. Damit wird gezeigt, daß konfliktfreie Netze jedenfalls eine arbeitsfähige Organisationsform mit vollständiger Angabe des Informationsflusses darstellen.

Wir haben in §4 solche Schaltelemente gewählt, deren Aufgaben unserem Verständnis am nächsten liegen. Um die allgemeine Interpretierbarkeit im physikalischen Bereich zu beweisen, wäre es nötig, sämtliche primitiven Begriffsbildungen der Physik in der Netzsprache auszudrücken. Diese Aufgabe würde jedoch den Rahmen dieser Arbeit bei weitem überschreiten; zudem erweist es sich als zweckmäßig, zu prüfen, ob nicht physikalische Gesichtspunkte viel stärker schon bei der Auswahl der Grundbegriffe und der Schaltelemente berücksichtigt werden sollten, bevor an eine solche Aufgabe gedacht wird.
Wir wollen infolgedessen untersuchen, ob sich nicht die Begriffe von §4 auf einfachere und weniger zahlreiche zurückführen lassen, mit dem dreifachen Ziel:

1) Postulat P2 soll leichter verifiziert werden können (wir haben dies nur für W-Knoten und dort nur unvollständig angedeutet);

2) die vereinfachten Begriffe sollen den physikalischen Begriffen nachgebildet sein;

3) es soll nicht unmöglich sein, im Prinzip das Verhalten jedes materiellen Körpers vollständig durch eine K-Form auszudrücken.

Beim Versuch der Vereinfachung fällt zunächst auf, daß die Knotenarten U, V, S je zwei Aktionen tragen. Wenn es gelingt, mit Ein-Aktions-Knoten auszukommen, so können wir den Begriff des Knotens ganz fallenlassen und die Stellen unmittelbar durch Aktionen verknüpfen.

Mit gleichgestalteten Definitionen und Axiomen wie in §4, mit der zusätzlichen Definition, daß jeder Knoten genau eine Aktion trägt, und mit Elimination des Begriffes "Knoten" erhalten wir statt der dort erklärten Netze eine neue Art von Strukturen, die wir Aktionsnetze (A-Netze) nennen.

Indem wir einen Knoten eines Netzes als Überlagerung von höchstens zwei Aktionen auffassen, erkennen wir sofort:

Jedes Netz mit den Parametern P, M - 2 ist einem Aktionsnetz mit den Parametern P, M isomorph.
Unser Interesse gilt also nun den Aktionsnetzen mit

$$P = 4, \qquad M = 4 \quad :$$

An jeder Aktion nehmen höchstens 4 Stellen teil;
Jede Stelle nimmt an höchstens 4 Aktionen teil.

Nachbarschaft von zwei verschiedenen Stellen heißt Teilnahme an einer gemeinsamen Aktion, Nachbarschaft von zwei verschiedenen Aktionen heißt Existenz einer gemeinsamen Stelle.

Die Bewertung einer Stelle mit einem Tobit (§4) wird nur durch einen Vorgang, die Ausführung einer Aktion, geändert.

Um dem genannten Ziele näherzukommen, überlegen wir, auf welche Weise wir erreichen können, daß eine physikalische Größe und die Gesetze, nach denen sie sich verändert, durch eine Verteilung von Tobits auf A-Netzen ausgedrückt wird. Das könnte dadurch geschehen, daß wir der Netzstruktur gewisse Beschränkungen auferlegen. Viel befriedigender erscheint jedoch der Versuch, allgemeine physikalische Prinzipien wie AP1 - 3 (§3) unmittelbar mit Eigenschaften der Aktionen in Verbindung zu bringen. Diesen Gedanken wollen wir so auslegen, daß nur solche Aktionen zugelassen werden, für die in jeder beliebigen Interpretation AP1 - 3 erfüllt sind.
AP1, die Existenz einer quantisierten Größe, haben wir schon durch die Wahl der Grundbegriffe berücksichtigt. AP2, die Gültigkeit eines Erhaltungssatzes, wollen wir dadurch erreichen, daß wir nur <u>konservative Aktionen</u> zulassen:

Eine Aktion heiße konservativ bezüglich einer Objektklasse i, wenn die Anzahl der i - Objekte ($i \in \{0,1\}$) in Vorder- und Hinterglied der Aktion dieselbe ist.

Beispiel: T ist konservativ bezüglich beider Objektklassen, W ist nicht konservativ.

AP3, die Gültigkeit eines Reaktionsprinzips, bringt es mit sich, daß jeder Beobachtungsvorgang die beobachtete Eigenschaft ändert. Eine Beobachtung ohne Änderung können wir nur dadurch völlig ausschließen, daß wir nur reaktive Aktionen zulassen:

Eine Aktion heiße reaktiv, wenn sämtliche in ihrer Beschreibung genannten Objekte verschieden sind (wenn also der "Zustand" aller teilnehmenden Stellen durch das Eintreten der Aktion geändert wird).

Beispiel: Q, T, W sind reaktiv, die Aktionen von V und S sind nicht reaktiv.

Den Konstruktionsprozeß wollen wir hier nicht formalisieren und behandeln also nur solche Veränderungen von Objektstrukturen, die in finiten Umbewertungen von Stellen durch die Ausführung von Aktionen bestehen. Wir betrachten daher im weiteren nur nichtsinguläre Aktionen:

Eine Aktion heiße singulär, wenn die Menge der im Vorderglied genannten Stellen ver-

schieden ist von der Menge der im Hinterglied genannten Stellen.
Eine konservative, reaktive, nichtsinguläre Aktion heiße eine (reine) Transition.

In einem nur aus Transitionen zusammengesetzten Aktionsnetz sind AP1 - 3 erfüllt für alle physikalischen Größen, die bei der Interpretation durch Mengen von Objekten einer bestimmten Klasse ausgedrückt werden.
Die Gesamtheit der möglichen Transitionen läßt sich leicht abzählen: es sind die Aktionen

01→10, 0011→1100, 000111→111000
usw.

oder in leicht verständlicher Schreibweise

+ -, + + - -, + + + - - - usw.

und jede mögliche Transition ist in dieser Folge enthalten (auf die Reihenfolge der Stellen kommt es nicht an).
Von einem Elementarvorgang wird man erwarten, daß er nur eine kleine Zahl von Stellen erfaßt. Wir suchen nun die kleinste Zahl P - falls sie existiert - derart, daß der Wirkungsfluß in einem entsprechenden Netz eine nichttriviale Struktur hat.
Für P = 2 besitzen wir nur das T - Element von §4. Man überzeugt sich leicht, daß aus diesem Element allein nur triviale Strukturen aufgebaut werden können.
Dagegen erweist es sich, daß P = 4 ausreicht,

und daß sogar allein aus der Transition

$$X : \quad ++--$$

und allein aus Stellen mit denselben formalen Symmetrieeigenschaften Netze für alle Schaltwerke und Turingmaschinen aufgebaut werden können.
Wir kommen nämlich mit Stellen von folgender Eigenschaft aus:

Jede Stelle nimmt an genau zwei Aktionen teil, welche die Umbewertung $0 \rightarrow 1$ auf ihr ermöglichen, und an genau zwei weiteren Aktionen, die die Umbewertung $1 \rightarrow 0$ auf ihr ermöglichen.
Dies hat zur Folge, daß wir uns jedes Ereignis an einer Stelle mit einem eigentlichen Informationsfluß (Transport eines Bits, nicht eines Tobits) verbunden denken können:
bei jeder Bewertung der Stelle gibt es genau zwei Aktionen X_1 und X_2, welche ein Ereignis herbeiführen können, und das Ereignis wird schließlich durch genau eine der Aktionen wirklich eintreten, sodaß der Entscheidungsgehalt des genannten Bits durch $\nabla X_1 X_2$ beschrieben wird. Wir können daher jeder Stelle formal eine K-Form (in KS3)

$$K_S : \quad \ast/ \, \nabla X_1 X_2 \; \nabla X_3 X_4$$

zuordnen, ebenso jeder X. - Transition eine K-Form

K_X : $\divideontimes$ / & a b & c d

Dieser formale Zusammenhang erst verschafft den Anschluß an den üblichen Informationsbegriff und hat das Auffinden der folgenden Netze erleichtert. Vielleicht unterstützt es die Anschauung, wenn wir ihn - etwas oberflächlich - so kommentieren:

Informationsbits (∇) werden stets ($\divideontimes$) getrennt bewegt (/) und sind auf ihre Nachbarn bezogen (&).

Selbstverständlich werden wir aus dieser unscharfen Formulierung keine Folgerungen ziehen.

Wir betrachten nun nochmals den Begriff des Konfliktes.
In einem konfliktfreien Netz wird jede Entscheidung allein auf Grund einer im Netz ausgedrückten Situation gefällt; es kann kein Bit neu entstehen. Wir stellen uns also vor, daß dasjenige Bit, welches einen vorliegenden Zwei-Aktionen-Konflikt entscheidet, einem Netz von außen zugeführt werden muß. Wenn wir annehmen, daß dies nur durch Aktionen geschehen kann, müssen wir ein nicht - konfliktfreies Netz einfach als <u>unvollständig</u> ansehen. Da ein kommunikationsfähiges Netz seiner Natur nach unvollständig ist, sind wir ohnehin vorwiegend an unvollständigen Netzen interessiert. Daher können wir ein konflikt-

freies Netz als ein Netz mit vollständig angegebenem Informations<u>zufluß</u> ansehen.

Nun ist für die abstrakte Beschreibung eines physikalischen Systems der zeitliche Richtungssinn gleichgültig, wenn nur seine Zuordnung zum Richtungssinn des Flusses der beteiligten Größen erhalten bleibt. Wenn wir den Richtungssinn "→" der Aktionen eines Netzes umkehren, so werden wir Informations-Zufluß und Abfluß vertauschen.

Die Umkehrbarkeit des "→" läßt sich für physikalische Elementarvorgänge wohl nicht wegdenken. Wir wollen mit diesem Symbol daher nur einen abstrakten Orientierungssinn ausdrücken, der Zeitsinn und Flußrichtungssinne einander zuordnet. Wenn wir nun in einem konfliktfreien Netz die "→" Richtung umkehren, so bleibt das Netz im allgemeinen nicht konfliktfrei. Die entstehenden Konflikte weisen genau auf die Stellen hin, an denen wir Information zuführen müssen, also vor der Umkehrung des "→" abgeführt haben (z.B. V - Element; Oder-Schaltung) ohne dies zu erwähnen.

Ein Aktionsnetz, welches für beide "→" Bewertungen konfliktfrei ist, wollen wir <u>versibel</u> nennen. In einem versiblen Netz kann also keine Information erzeugt wer -

den oder verloren gehen, und alle ihre Zufluß- und Abflußstellen sind wohldefiniert. Den Zusammenhang von reversiblen physikalischen Prozessen und versiblen Aktionsnetzen wollen wir hier nicht weiter verfolgen.

Derartig einfache Aktionsnetze sind möglicherweise von Wert für die Diskussion unserer Modellvorstellungen von der Feinstruktur der Materie. Denn wenn wir Postulat P1 als allgemeingültiges physikalisches Prinzip zulassen - dies scheint für Grundlagenuntersuchungen recht vorteilhaft zu sein -, so gewinnen wir mit einer Theorie der Aktionsnetze (bzw. des Informationsflusses) sicherlich ein nützliches mathematisches Hilfsmittel zur theoretisch-physikalischen Beschreibung derjenigen Größen, die den bei Postulat P2 erwähnten Gesetzmäßigkeiten genügen. Dies ist besonders deshalb nützlich, weil es die Einbettung der effektiv zu beobachtenden Größen in ein gedachtes Kontinuum überflüssig macht, dessen über das Beobachtbare hinausgehenden Eigenschaften doch recht willkürlich gewählt werden können und im Bereich der Grundlagenfragen jedenfalls eine unnötige Belastung darstellen. -

Wir wollen nun die wichtigsten Netze von §4 zu versiblen Aktionsnetzen erweitern und dabei nur die Transition X und die erwähnte Art von Stellen benutzen. Da die Netze unvollständig sein müssen, können nicht alle Stellen und Aktionen die obigen Definitionen erfüllen, sondern solche, in denen "genau zwei" durch "höchstens zwei" usw. ersetzt ist. Alle in diesem Sinne unvollständig verknüpften Stellen und Aktionen bilden den Rand des Aktionsnetzes.

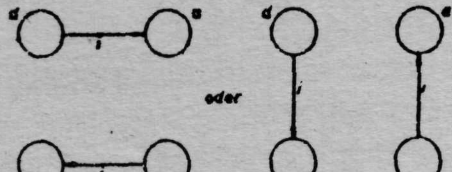
Zur bildlichen Darstellung können wir wegen $M = 4$ nicht mehr die Graphen von §4 verwenden. Wir verabreden:

Stelle mit 0 - Tobit im Augenblick der Konstruktion: ○

Stelle mit 1 - Tobit im Augenblick der Konstruktion: ⊙

Aktion X_i (abcd) : ++-- :

beliebig entweder

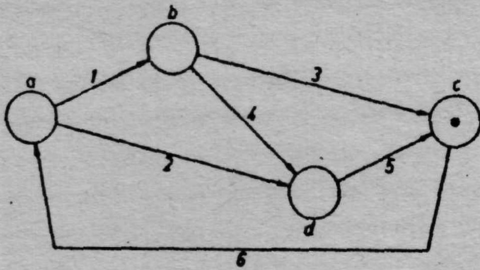


Eine Aktion ist als Paar gerichteter Strecken mit der gleichen natürlichen Zahl als Marke dargestellt. Verschiedene Aktionen haben verschiedene Marken.

Aussagenlogische Funktionen zweier Variablen :

Es genügt, die (bis auf Vertauschung der Benennung der Anschlüsse übereinstimmende)

Darstellung der acht nichtzerfallenden "1:3"-Funktionen anzugeben, da sich die Sheffer-Funktion unter ihnen befindet. Wir wählen die Disjunktion:

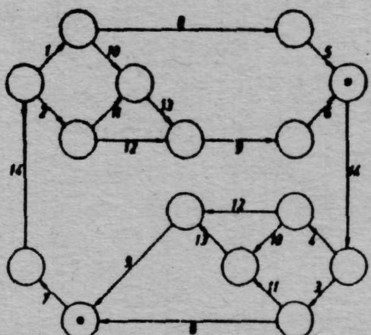


6 ist die Aktivierung;
$\nabla 12 = x$
$\nabla 34 = y$
$\nabla 35 = (x \vee y)$
$\nabla 42$ wird abgeführt

Für die Konjunktion ist also z.B. :

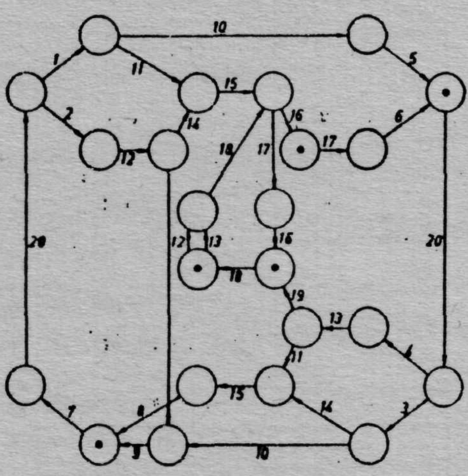
$\nabla 21 = x$, $\nabla 43 = y$, $\nabla 53 = (x \wedge y)$ usw.

Falls x und y stets beide abgefragt werden sollen, wie im Beispiel "Disjunktion" §4, treten einige vollständige Transitionen auf:



7 = Aktivierung
$\nabla 12 = x$
$\nabla 34 = y$
$\nabla 56 = (x \vee y)$

Diese Anordnung bringt uns nur in einem nicht - konfliktfreien Netz Vorteil, wenn nämlich die Entscheidungen x und y eine "größere Zeitspanne" beanspruchen, wenn sie nacheinander aktiviert werden, als wenn dies simultan geschieht. (Der Begriff der Zeitspanne ist nur in nicht - konfliktfreien Netzen sinnvoll). Dann aber können wir den Konflikt in die Funktionsauswertung mit hereinnehmen (wie in §4) und aus dem Zeitgewinn den größtmöglichen Nutzen ziehen. Außerdem gelingt es hier leicht, im Gegensatz zu §4, das Netz für beliebig häufige Benutzung anzulegen:



7 = Aktivierung

$\nabla 12 = x$

$\nabla 34 = y$

$\nabla 56 = (x \vee y)$ (beschleunigt)

$\nabla \widehat{12}\ \widehat{13}$ Konflikt bei $x = y = 1$

$\nabla 89 = (x \equiv y)$ (nicht beschleunigt)

Das Netz liefert zusätzlich die Äquivalenzfunktion, die sich natürlich nicht beschleunigen läßt. -

Schaltwerksfunktionen.

Das effektive Alphabet für Ein- und Ausgabe habe wie in §4 je zwei Elemente a, b bzw.

c, d. Die normale K-Form ist

$$* / \nabla \; a \; b \; \nabla \; c \; d.$$

Wir wollen jetzt Wert auf die Reversibilität (Vertauschbarkeit von Ein- und Ausgabe) aller Schaltwerke legen, die in der üblichen Theorie nur in Ausnahmefällen auftritt. In der folgenden Netzdarstellung ergibt es sich nämlich ganz von selbst, daß alle dem Schaltwerk verlorengehende Information an einer wohldefinierten Stelle abgeführt werden muß. Die Kommunikation über diese latente Randstelle ergänzt also den Informationsfluß derart, daß auch für die Information ein Erhaltungssatz gilt. Dies gilt allgemein für versible Aktionsnetze.

Gewisse Fragen der Minimisierung von Schaltwerken lassen sich aufgrund dieser Überlegungen überraschend einfach beantworten, wie an anderer Stelle geschildert werden soll. Hier wollen wir sie beiseitelassen; in diesem Sinne soll die Verallgemeinerung auf Schaltwerke mit beliebigem endlichen Alphabet so ausgeführt gedacht werden, daß durch zwei besonders einfache, ohnehin reversible Schaltwerke eine Umcodierung (wie in §2, 4.) in und aus $\{0, 1\}$ stattfindet, und daß der Zustandsgraph des "zentralen" Schaltwerks entsprechend modifiziert wird.

Sei nun $Q = \{q_1 \ldots q_n\}$ eine Menge von Zuständen, $B = \{a, b\}$ und $C = \{c, d\}$ die Alphabete für Ein- und Ausgabe, so wird jedes Schaltwerk dargestellt durch eine eindeutige Abbildung A von $Q \times B$ in $Q \times C$. Es gibt dann stets eine Menge

$R = \{r_1 \ldots r_m\}$ mit $m \leq 2n$, Alphabete

$B' = \{e, f\}$, $C' = \{g, h\}$ und eine <u>umkehrbar</u>

eindeutige Abbildung F von

$(Q \times B) \cup (R \times B')$ <u>auf</u> $(Q \times C) \quad (R \times C')$

sodaß gilt:

Wenn $F'(C') = e$ so ist $(FF')^i F = A$ mit

$0 \leq i \leq m$.

Beispiel:

A:

$q_1 a$	$c q_2$
$q_1 b$	$c q_2$
$q_2 a$	$c q_1$
$q_2 b$	$d q_1$

F:

$q_1 a$	$g r_1$
$q_1 b$	$h r_1$
$q_2 a$	$c q_1$
$q_2 b$	$d q_1$
$r_1 e$	$c q_2$
$(r_1 f$	$d q_2)$

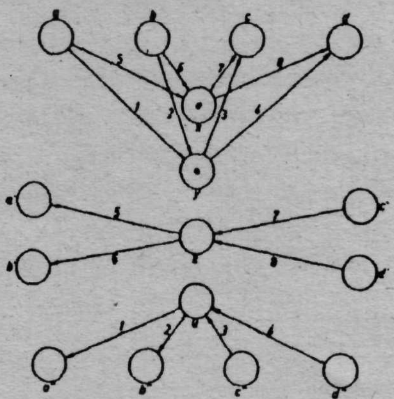
Der Informationsabfluß ist also auf F' konzentriert, und wir haben endlich viele Zwischenzustände r eingeführt. Die Struktur des Schaltwerks ist durch eine

Permutation von $(Q \cup R) \times \{0, 1\}$ ausgedrückt.

Umgekehrt stellt jede solche Permutation ein Schaltwerk dar.

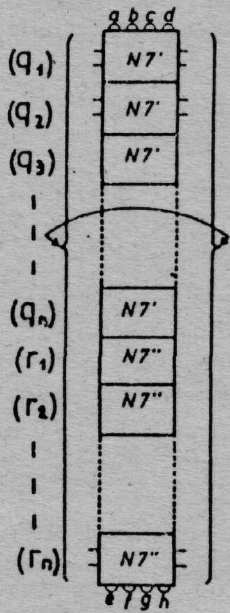
Wir brauchen jetzt nur noch N7 zu beschreiben und die freien Randstellen a'b'c'd' gemäß der genannten Permutation zu identifizieren; dieser Teil der Verknüpfungsvorschrift, der aus einer N7 - Kette ein spezielles Schaltwerk herstellt, ist damit bereits erledigt.

Die direkte Übersetzung von N7 in die Sprache von §5 führt zu einem nicht-versiblen Netz von 36 Transitionen.
Durch Betrachtung des Informationsflusses läßt sich aber erkennen, an welchen Stellen ein überflüssiger Bit - Transport stattfindet, und durch Weglassen dieser Stellen ergibt sich folgendes Bild:



mit nur vier inneren Stellen x y z u .
Durch Spiegelung an einer Horizontalen entstehe N7'' .

Das gesamte Schaltwerk hat dann die Form



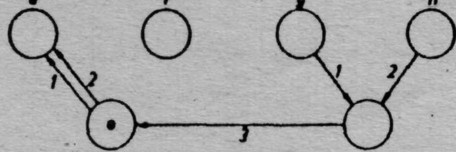
Permutation P

Nun vertauschen wir beim obersten Netz die Bewertungen von x und z (bei der Konstruktion); dann ist das Schaltwerk arbeitsbereit und "befindet sich im Zustand" q_1.

Es würde nun solange arbeiten, wie die benutzten Teile der Schaltwerksfunktion A eine umkehrbar eindeutige Teilfunktion von A bilden; dann aber versucht es, ein Bit ∇ g h abzugeben, und wir müssen vorgesorgt haben, daß am latenten Rand die K-Form

$*/ \nabla$ g h e erfüllt ist;

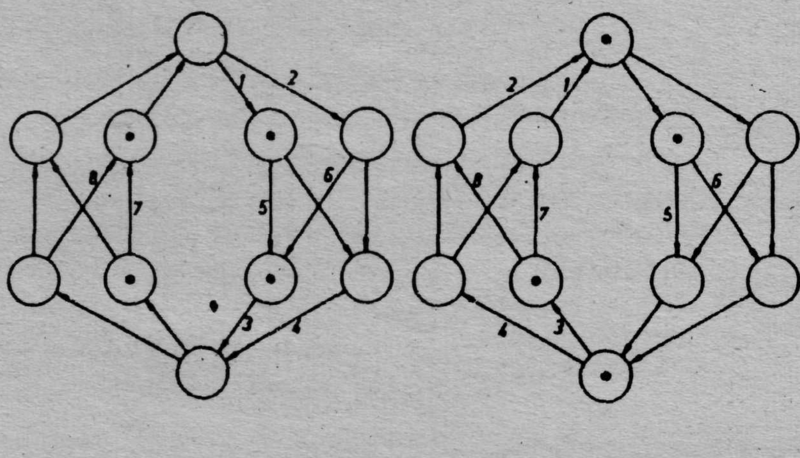
entweder durch ein nicht - versibles Netz



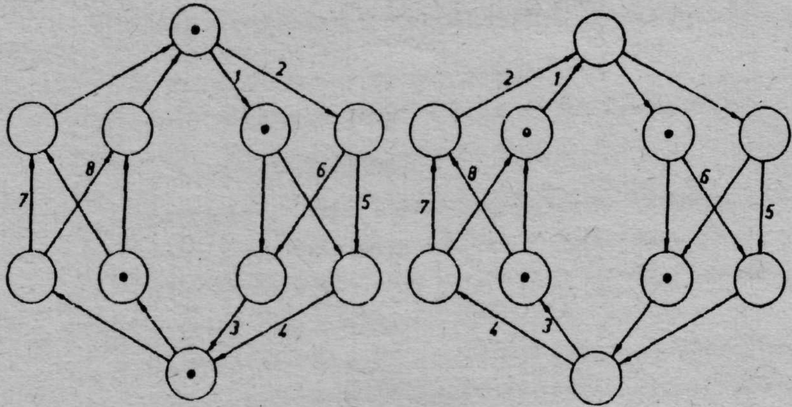
oder durch einen versiblen Lifo - Speicher. Einen solchen Speicher liefert aber gerade das Netzpaar N9 - N9' durch Verkettung, und die Existenz eines versiblen Aktionsnetzes mit dieser Funktion beweist zugleich die Darstellbarkeit der beiden Bandhälften einer Turingmaschine. Wir geben zwei Netzpaare an, um die Art der Verkettung zu zeigen. Jedes Netzpaar trägt zwei Bit der gespeicherten Information.

(Rechtes Halbband): ···E F E F E F···

Wir zeigen die Verkettung E F :



und die Verkettung F E :



Die Frage der technischen Realisierung der betrachteten Schaltelemente soll in dieser Arbeit nicht erörtert werden, und zwar mit folgender Begründung:

1) Die Frage ist in dieser Form nicht relevant. Die Auswahl der Schaltelemente war rein willkürlich (§4). Wesentlich ist allein die Einhaltung der Kommunikationsformen an den Rändern beliebig zusammengefaßter, zusammenhängender B - Strukturen.

2) Daß der Zwang zur Synchronisierung der Aktionen entfällt, ist jedenfalls eine wesentliche Erleichterung für die technische Interpretation. Dafür entsteht die Frage, wie man die Invarianz der Aktionen sicherstellen soll. Im makroskopisch-mechanischen Bereich ist die Frage jedenfalls befriedigend

lösbar (Relaistechnik; Beschriftung, Bewegung und Ablesung materieller Datenträger; Kommunikation zwischen Personen durch Sprache, Schrift, übergebene Gegenstände); desgleichen offenbar im makromolekularen Bereich (Nervensystem) und möglicherweise mit voller Strenge im atomaren Bereich. Auf dem Gebiet der Elektronik ist die Frage noch nicht untersucht; wahrscheinlich deshalb, weil man asynchrone Prozesse vorzugsweise in einem Kontinuum (zumindest der Zeit) zu betrachten pflegt und bei Anlässen zu diskontinuierlicher Betrachtung die Aufteilung der Zeit in eine vollgeordnete Menge von Einzelschritten für ein stets sinnvolles, die Allgemeinheit für alle praktischen Zwecke nicht einschränkendes Vorgehen hält; irrtümlich -, wie wir unter Berufung auf §§ 2 - 4 hinzufügen müssen. Es ist daher erforderlich, die Frage der unmittelbaren technischen Repräsentation von Aktionsnetzen derart durchzudenken, daß der Umweg über den üblichen Zeitbegriff für Schaltwerke ganz vermieden wird.

3) Die Annahme erscheint gerechtfertigt, daß die Wahl des Schaltelements in §5 aufgrund physikalischer Überlegungen die Realisierung nicht erschwert, sondern ihr bei geeigneter Interpretation

sogar entgegenkommt. - Die weitere Erörterung dieser Fragen muß den Fachleuten überlassen bleiben.

Wir stellen nochmals die von dieser Betrachtungsweise zu erhoffenden Vorteile zusammen:

1) Die Möglichkeit der konfliktfreien Kommunikation mit Automaten und zwischen Automaten;

2) Die Unabhängigkeit der Konstruktionsvorgänge von den Arbeitsprozessen von Automaten und damit die unbeschränkte Möglichkeit, Automaten während ihrer Arbeit zu erweitern;

3) Die Vereinfachung der Interpretation in der Hinsicht, daß von jedem Schaltelement nur eine "möglichst kurze Schaltzeit" gefordert wird, ohne daß von der logischen Struktur her Toleranzen in der Schaltzeit einzuhalten sind. (Dabei ist anzumerken, daß der Begriff der Schaltzeit nur in solchen Gesamtnetzen auftreten kann, die wenigstens eine Konfliktstelle besitzen).

4) Die konfliktfreie Verknüpfung simultan ablaufender Prozesse.

5) Die Verlagerung von nicht eliminierbaren Konflikten auf solche Stellen, wo aus ihrer Existenz Nutzen gezogen werden kann (siehe Oder-Schaltung).

Wir kommen nun zurück auf die in §1 erwähnte Anwendung auf Fragen der Programmierung.

Diese Anwendungsmöglichkeit ist so offenkundig, daß wir nur auf die begriffliche Ähnlichkeit der gezeigten Systeme von Netzen mit den üblichen Flußdiagrammen hinzuweisen haben. Wieder ist es die vom Gewohnten abweichende Art der Behandlung von zeitlichen Abhängigkeiten, was uns eine völlig explizite Darstellung des Wirkungsflusses und daher eine streng formale Behandlung der Kommunikation unter Berücksichtigung der physikalischen Möglichkeiten erlaubt.
Die wichtige Rolle des Parallelitätsoperators ist schon in §4 beschrieben worden. Seine Einführung kann gedeutet werden als Verzicht auf einen universellen und daher gewöhnlich nicht erwähnten Parallelitätsoperator, welcher für jeden Zeitschritt obligatorisch ist und sämtliche Vorgänge in je zwei aufeinanderfolgenden Zeitschritten verknüpft -, und seine Ersetzung durch eine ausdrücklich anzugebende gleichmäßig beschränkte Verknüpfung. Es ist klar, daß dies auf die Beschreibung von organisatorischen Zusammenhängen einen tiefgreifenden Einfluß hat.

Ohne einen solchen Operator ist eine sinnvolle interpretierbare Formalisierung von irregulären Kommunikationsvorgängen nicht möglich. Darüberhinaus gestattet der Begriff des Konfliktes eine formal exakte, endliche Darstellung solcher Ausdrücke wie "NA versucht, die Aufgabe B zu lösen" unabhängig von irgendwelchen Angaben über den Inhalt, über die Struktur der Aufgabe B. Der Konfliktbegriff liefert ferner eine Darstellung der Spielsituation und gestattet eine übersichtliche Trennung der verschiedenen Wahrscheinlichkeitsbegriffe durch explizite Angabe von Netzen, welche die Vorschriften (K-Formen) zur Kommunikation zwecks "Feststellung einer Wahrscheinlichkeit" erfüllen. Darunter verstehen wir einen Plan zur Ableitung von Entscheidungen ∇ x y aus einem Wahrscheinlichkeitsurteil; solche Urteile werden als Ausgabe von Netzen geliefert, deren Konfliktfreiheit nicht erwiesen ist.

Auf diese Weise rücken gewisse sehr einfache Prinzipien der Organisation in den Bereich der hier als Programmierung bezeichneten Tätigkeit, die dem Zugriff der gewohnten "Zeitschrittprogrammierung" grundsätzlich entzogen sind. Dies ist von Interesse wegen der wachsenden Kompliziertheit der Aufgaben, die den Informationsmaschinen übertragen werden sollen: bei irregulären Aufgaben entsteht die Gefahr, daß der Benutzer der Maschine die Menge der einzuhaltenden Verabredungen nicht mehr zu übersehen vermag. Da aber sämtliche irregulären Teile der Aufgaben auf solche

Verabredungen abgebildet werden müssen, bedeutet die Übernahme eines Benutzer - Teilplanes durch einen Automaten nur eine vergleichsweise triviale Erleichterung für den Benutzer, wenn der Teilplan in Zeitschritt-Programmierung ausdrückbar ist.

In diesem Sinne sollte das Problem der Automatisierung in stärkerem Maße als ein Kommunikationsproblem gesehen werden.

Die mathematisch-logische Behandlung dieses Problems kann, so ist zu hoffen, durch die Entwicklung eines <u>interpretierten Kalküls der K-Formen</u> erleichtert werden. Durch Ansätze zu dieser Entwicklung sind die in dieser Arbeit gezeigten Netze gefunden worden; weiterhin haben solche Ansätze wertvolle Hinweise für die Übersetzung formaler Sprachen wie Algol geliefert. Daher wollen wir nochmals auf diesen Kalkül zurückkommen.

Analog zu dem Übergang von KS1 zu KS2 verallgemeinern wir KS3 zu KS4 mit den zweistelligen Operatoren ∇, &, /, und dem nullstelligen Operator R.

Für die Bequemlichkeit der Interpretation lassen wir außerdem daraus ableitbare Operatoren zu, wie $\divideontimes$ und einen Zyklus-Operator Z:

$$\divideontimes a \subseteq \nabla\, a/aR \; ; \quad Zab \subseteq \nabla\, a//abR \; .$$

Ein R im Zweitglied einer ∇ - Formel führt wie in KS2 zur Rekursion. Durch R kann die Aktivierung eines Konstruktors ausgedrückt werden. Die Interpretation von R ist noch nicht genügend untersucht. Es wird vermutet, daß mit Hilfe der KS4 - Formeln und einer geeignet definierten und interpretierten $\subseteq$ - Relation zwischen ihnen bereits alle elementaren Aspekte der Kommunikation befriedigend dargestellt werden können. Die Vermutung könnte durch ein zu T1, T2 analoges Paar von Theoremen stark unterstützt werden. Ein entsprechender formaler Beweis ist für gewisse Aktionsnetze möglich; wir verzichten jedoch auf die Wiedergabe wegen einiger Schwierigkeiten der Interpretation, die noch nicht beseitigt sind. Mit anderen Worten: Die Klasse von verallgemeinerten Automaten, für welche ein Paar von Theoremen wie T1, T2 mit KS4 statt KS1 gültig ist, ist noch nicht hinreichend charakterisiert.

Dagegen hofft der Verfasser, mit dieser Arbeit zur Klärung der begrifflichen Grundlagen einer Theorie der Kommunikation beizutragen. Er dankt Herrn Prof. Dr.-Ing. A. Walther in Darmstadt und Herrn Prof. Dr.-Ing. H. Unger in Bonn, durch deren Wirken die Beschäftigung mit Informationsmaschinen in diesem Land einen entscheidenden Auftrieb erhalten hat, und durch deren Anregung auch der Verfasser auf dieses Gebiet geführt wurde.