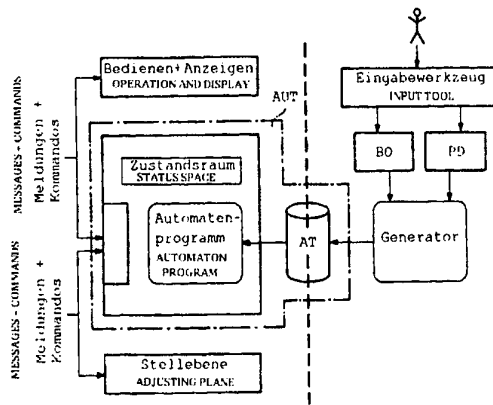


**INTERNATIONALE ANMELDUNG VERÖFFENTLICHT NACH DEM VERTRAG ÜBER DIE
INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS (PCT)**

(51) Internationale Patentklassifikation ⁵ : G05B 19/04	A1	(11) Internationale Veröffentlichungsnummer: WO 94/12914 (43) Internationales Veröffentlichungsdatum: 9. Juni 1994 (09.06.94)
(21) Internationales Aktenzeichen: PCT/DE93/01023 (22) Internationales Anmeldedatum: 22. Oktober 1993 (22.10.93) (30) Prioritätsdaten: 92250346.1 25. November 1992 (25.11.92) EP (34) Länder für die die regionale oder internationale Anmeldung eingereicht worden ist: DE usw. (71) Anmelder (für alle Bestimmungsstaaten ausser US): SIEMENS AKTIENGESELLSCHAFT [DE/DE]; Wittelsbacherplatz 2, D-80333 München (DE). (72) Erfinder; und (75) Erfinder/Anmelder (nur für US): FILKORN, Thomas [DE/DE]; Waltherstrasse 27, D-80337 München (DE). NÖKEL, Klaus [DE/DE]; Zunftstrasse 38a, D-85540 Haar (DE). PETERSEN, Hans-Joachim [DE/DE]; Leipziger Strasse 92, D-38350 Helmstedt (DE). PROTZNER, Sven [DE/DE]; Glogauer Weg 6, D-38302 Wolfenbüttel (DE).		(81) Bestimmungsstaaten: CA, FI, JP, NO, RU, US, europäisches Patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE). Veröffentlicht <i>Mit internationalem Recherchenbericht.</i>
(54) Title: PROCESS FOR DEFINING THE APPLICATION-DEPENDENT LOGIC OF A FREELY PROGRAMMABLE SEQUENTIAL LOGIC SYSTEM, DEVICE FOR IMPLEMENTING THIS PROCESS AND DEVICE FOR OPERATING A CONTROL SYSTEM USING A PROGRAM THUS DEFINED		
(54) Bezeichnung: VERFAHREN ZUM ERSTELLEN DER ANWENDUNGSABHÄNGIGEN LOGIK EINES FREIPROGRAMMIERBAREN SCHALTWERKES, EINRICHTUNG ZUR DURCHFÜHRUNG DIESES VERFAHRES UND EINRICHTUNG ZUM BETRIEB EINES STEUERUNGSSYSTEMS UNTER VERWENDUNG EINES SO ERSTELLTEN PROGRAMMS		
(57) Abstract The sequential logic system used to control the running of a process is modelled as a finite logic controller (AUT). The logic controller table (AT), however, does not contain sequence statuses and results arising through the use of an input symbol on the momentary status of an object, but rather status transition functions with which the actual sequence statuses and results can be calculated. In order to determine these functions, the problems to be dealt with by the individual objects are expressed in technical language in terms of object types, which makes it possible to express logic conditions of elementary statuses of the objects. This language is then applied to the concrete objects and the language elements used are replaced by the logical links of the elementary object statuses contained therein. This is done using a computer. The computer also determines the object-related transient functions from all the conditions laid down for the objects and the individual input symbols and enters them in the logic controller table according to the rules of the binary decision diagrams. 		

(57) Zusammenfassung

Das für die Steuerung eines Prozeßgeschehens verwendete Schaltwerk ist als endlicher Automat (AUT) modelliert. In der Automatentafel (AT) sind aber nicht Folgezustände und Ergebnisse niedergelegt, die sich durch Anwendung eines Eingabesymbols auf den Augenblickzustand eines Objektes ergeben, sondern Zustandsübergangsfunktionen, über die die jeweiligen Folgezustände und Ergebnisse ausgerechnet werden können. Für die Bestimmung dieser Funktionen werden zunächst die von den einzelnen Objekten zu behandelnden Probleme auf Objekttypenebene fachsprachlich erfaßt, wobei die Sprachmittel dieser Fachsprache logische Bedingungen elementarer Zustände der Objekte auszudrücken gestattet. Sodann werden diese Sprachmittel auf die konkreten Objekte angewendet und die verwendeten Sprachbegriffe werden dann ersetzt durch die logischen Verknüpfungen der in ihnen enthaltenen elementaren Objektzustände. Dies geschieht durch Anwendung eines Rechners. Dieser Rechner bestimmt auch die objektbezogenen Übergangsfunktionen aus der Gesamtheit der für die Objekte und die einzelnen Eingabesymbole hinterlegten Bedingungen und legt diese nach den Regeln der Binary Decision Diagramms in der Automatentafel ab.

LEDIGLICH ZUR INFORMATION

Codes zur Identifizierung von PCT-Vertragsstaaten auf den Kopfbögen der Schriften, die internationale Anmeldungen gemäss dem PCT veröffentlichen.

AT	Österreich	GA	Gabon	MR	Mauretanien
AU	Australien	GB	Vereinigtes Königreich	MW	Malawi
BB	Barbados	GE	Georgien	NE	Niger
BE	Belgien	GN	Guinea	NL	Niederlande
BF	Burkina Faso	GR	Griechenland	NO	Norwegen
BG	Bulgarien	HU	Ungarn	NZ	Neuseeland
BJ	Benin	IE	Irland	PL	Polen
BR	Brasilien	IT	Italien	PT	Portugal
BY	Belarus	JP	Japan	RO	Rumänien
CA	Kanada	KE	Kenya	RU	Russische Föderation
CF	Zentrale Afrikanische Republik	KG	Kirgisistan	SD	Sudan
CG	Kongo	KP	Demokratische Volksrepublik Korea	SE	Schweden
CH	Schweiz	KR	Republik Korea	SI	Slowenien
CI	Côte d'Ivoire	KZ	Kasachstan	SK	Slowakei
CM	Kamerun	LI	Liechtenstein	SN	Senegal
CN	China	LK	Sri Lanka	TD	Tschad
CS	Tschechoslowakei	LU	Luxemburg	TG	Togo
CZ	Tschechische Republik	LV	Lettland	TJ	Tadschikistan
DE	Deutschland	MC	Monaco	TT	Trinidad und Tobago
DK	Dänemark	MD	Republik Moldau	UA	Ukraine
ES	Spanien	MG	Madagaskar	US	Vereinigte Staaten von Amerika
FI	Finnland	ML	Mali	UZ	Usbekistan
FR	Frankreich	MN	Mongolei	VN	Vietnam

Verfahren zum Erstellen der anwendungsabhängigen Logik eines freiprogrammierbaren Schaltwerkes, Einrichtung zur Durchführung dieses Verfahrens und Einrichtung zum Betrieb eines Steuerungssystems unter Verwendung eines so
5 erstellten Programms.

Zur Steuerung und Überwachung auch umfangreicher Prozeßgeschehen werden heute nahezu ausschließlich Rechner oder Rechnersysteme verwendet. So werden z. B. zum Sichern des
10 Bahnverkehrs elektronische Stellwerke eingesetzt, die aus einer Vielzahl von in sich sicheren Rechnersystemen bestehen. Diese Rechner behandeln einzelne Komponenten des Prozeßgeschehens "Bahnanlage" entweder zentral wie z. B. bei der Eingabe von Kommandos und der Ausgabe von Meldungen
15 oder aber dezentral wie z. B. bei der Steuerung des eigentlichen Stellwerksgeschehens. Die letztgenannten Rechner, die sogenannten Bereichsrechner oder Bereichs-Stellrechner, sind bestimmten Teilen der zu steuernden Anlage zugeordnet. Sie beinhalten eine anwenderunabhängige Systemsoftware,
20 d. h. diejenigen Programme, welche die Rechner in die Lage versetzen, als Stellwerk zu arbeiten, und eine bahnverwaltungsspezifische Software, welche die logischen Stellwerks- und Blockbedingungen der jeweiligen Bahnverwaltung beinhaltet; die Systemsoftware und die bahnverwaltungsspezifische Software sind bei allen Bereichsrechnern ein und
25 desselben Stellwerkes gleich. Um die ihnen zugewiesenen Stellwerksfunktionen erfüllen zu können, müssen die Bereichsrechner auch Kenntnis haben von den anlagenspezifischen Daten mindestens derjenigen Fahrwegelemente, die zu dem von ihnen gesteuerten Teilbereich der Anlage gehören.
30 Zu diesen anlagenspezifischen Daten gehören Angaben über die jeweils vorhandenen Fahrwegelemente und ihre Anordnung im Gleis in bezug auf die Nachbarfahrwegelemente. Diese Daten sind bei jedem Bereichsrechner unterschiedlich; sie

werden von den Bereichsrechnern in einer Aufrüstphase aus Speichern eines zentralen Rechners individuell abgerufen. Zum Abarbeiten von Stellwerksfunktionen ruft der Bereichsrechner die entsprechenden, in seinem Speicher hinterlegten Unterprogramme z. B. für die Weichenumstellung oder die Flankenschutzsuche ab und verknüpft sie mit den jeweils infrage kommenden anlagenspezifischen Daten, daß z. B. mit den Daten einer bestimmten Weiche oder den Daten der in die Flankenschutzsuche einzubeziehenden konkreten Fahrweg-
elementen. Die anlagenspezifischen Daten werden bei der Projektierung eines Stellwerkes vom Projektierer aufgenommen, wobei dessen Arbeit unterstützt wird durch speziell entwickelte elektronische Hilfsmittel. Diese Hilfsmittel ihrerseits bestehen aus einer Eingabe- und Datensichtstation und sind so konzipiert, daß der Projektierer lediglich die jeweils relevanten Daten an vorgegebenen Stellen in auf einem Sichtgerät formatiert erscheinenden Listen einzutragen hat. Dabei findet gleichzeitig eine Vollständigkeitskontrolle statt. Weitere Projektierungswerkzeuge gestatten eine quantitative Auflistung der für eine Stellwerksanlage benötigten Hardware, die Erstellung der Belegungspläne für Schränke und Rechnerrahmen, die Pläne zur Innenanlageverkabelung und die Erstellung der Pläne der Außenanlage und deren Verkabelung (Signal + Draht, 77(1985)12, Seiten 259 bis 265).

Während es für die eigentlichen Projektierungsarbeiten, also die Auflistung der Element- und Anlagendaten, sowie die Erstellung von Fertigungsunterlagen gut funktionierende Hilfsmittel zu einer mindestens teilweise automatisierten Erstellung dieser Unterlagen gibt, trifft dies für die Erstellung der Systemsoftware und der bahnverwaltungsspezifischen Software und ihre Verknüpfung miteinander nicht zu. Die Erstellung und vor allem die Prüfung dieser Soft-

ware ist außerordentlich zeit- und kostenaufwendig, weil sie regelmäßig in einer an die Betriebsweise von Rechnern angepaßten Programmiersprache und nicht in einer problemorientierten Sprache erfolgt, welche das eigentliche Stellwerksgeschehen in einer für den Entwickler verständlichen Form beschreibt. Durch die Notwendigkeit, für diese Software einen Sicherheitsnachweis zu führen, vergrößern sich die Kosten und die Zeitspannen bis zum Inbetriebgehen derartiger Stellwerke erheblich.

10

Ändern sich die bislang berücksichtigten anwendungsspezifischen Forderungen an eine Steuerungsanlage, weil andere Bedingungen an das Prozeßgeschehen gestellt werden, so sind umfangreiche Programmierarbeiten zur Erstellung einer entsprechend abgewandelten Anwendersoftware nötig; mindestens bei sicherheitsrelevanten Prozeßsteuerungen ist dann auch ein gesonderter Sicherheitsnachweis erforderlich. Auch bei Änderungen der zu steuernden Anlage selbst müssen umfangreiche Neuprojektierungsarbeiten bewerkstelligt werden, auch wenn diese gegebenenfalls durch den Einsatz von speziellen Hilfsmitteln mindestens zum Teil zu automatisieren sind.

15

20

Ansätze in Richtung auf eine für den Entwickler verständliche und übersichtliche problemangepaßte Programmierung im Gegensatz zu einer ausschließlich rechnerorientierten Programmierung gibt es bei der sogenannten objektorientierten Programmierung (Computer Magazin 7-8/91, Seiten 34 bis 40). Bei der objektorientierten Programmierung wird zunächst das anwendungsorientierte Objekt "Steuerungssystem", z. B. "Stellwerk", in mehrere Ebenen von hierarchisch gegliederten Objektklassen unterteilt, von denen die höheren Klassen, z. B. die Signale, die allgemeineren und die nach unten hin gelegenen Klassen die spezielleren

25

30

- Objekte, z. B. die Vor- und Hauptsignale, beschreiben. Die einzelnen Objektklassen fassen jeweils Individualobjekte mit gleichen Eigenschaften zusammen. Die Beschreibung der Objekte und Objekttypen erfolgt innerhalb der einzelnen
- 5 Klassen, wobei in Richtung auf die unteren Hierarchieebene immer speziellere Eigenschaften, in denen sich die Objekte voneinander unterscheiden, beschrieben sind. Innerhalb der einzelnen Ebenen sind auch die Verarbeitungsprogramme für die Objekte, die sogenannten Methoden, definiert. Sie be-
- 10 stehen wie herkömmliche Rechnerprogramme aus einer Abfolge von Befehlen zur Lösung einzelner Teilprobleme und müssen durch einen Programmierer nach den Anweisungen des Entwicklers erstellt werden. Es ist nicht vorherzusagen, wie mehrere solcher Teilprogramme wirken, wenn sie miteinander
- 15 kombiniert werden. Insbesondere ist nicht auszuschließen, daß Programmfehler, die sich in ein Teilprogramm eingeschlichen haben, Auswirkungen auf das kombinierte Programm haben können, die nicht vorhersehbar und nur schwer auszumachen sind. Ein schwerwiegendes Problem objektorien-
- 20 tierter Programmierung ist der Umstand, daß es keine Gewähr für Echtzeitverhalten gibt, weil nicht von vornherein erkennbar ist, wieviele Objekte in einer Bearbeitungsfolge miteinander zu verknüpfen sind.
- 25 Aufgabe der Erfindung ist es, ein Verfahren nach dem Oberbegriff des Patentanspruches 1 anzugeben, mit deren Hilfe es möglich ist, auf übersichtliche und verständliche Weise eine Automatentafel zur vollständigen, Beschreibung des Zustandsübergangs- und Ergebnisverhaltens eines komplexen
- 30 Steuerungssystems zu generieren sowie eine Einrichtung zur Durchführung dieses Verfahrens anzugeben, die es gestattet, die Automatentafel weitgehend automatisch und damit fehlerfrei zu erstellen. Ferner ist es Aufgabe der

Erfindung, eine so erstellte Automatentafel für die Steuerung eines Prozeßgeschehens zu verwenden.

Die Erfindung löst diese Aufgabe durch die kennzeichnenden
5 Merkmale der Ansprüche 1, 6 bzw. 9. Vorteilhafte Ausgestaltungen der Erfindung sind in den auf diese Ansprüche direkt oder indirekt rückbezogenen Unteransprüchen angegeben.

10 Die Merkmale des Anspruches 1 beschreiben die Verfahrensschritte zum Generieren der Automatentafel, wie sie später von einem Rechner oder Rechnersystem zur Behandlung einer bestimmten Steuerungsaufgabe abzuarbeiten ist. Ein wesentlicher Vorteil dieses Verfahrens besteht darin, daß das je-
15 weils zu lösende Problem zunächst in einer rechnerunabhängigen Fachsprache definiert und später durch einen schematischen Vorgang in ein für einen Rechner lesbares Programm umgesetzt wird, wobei durch eine spezielle Darstellung der Umfang für die bei der Problemlösung von einem
20 Rechner abzuarbeitenden Bedingungen handhabbar gehalten werden kann. Das Zustandsübergangs- und Ergebnisverhalten der Objekte wird nicht durch die Ausgangs- und Folgezustände angegeben, sondern durch logische Funktionen, nach denen sich die Folgezustände und Ergebnisse aus den
25 Ausgangszuständen errechnen lassen.

Anspruch 2 benennt nähere Ausgestaltungen zu den für die einzelnen Objekte in der Automatentafel abzuspeichernden Zustandswerten und zu den Gesetzmäßigkeiten für den
30 Zustandswechsel.

Die nach dem Anspruch 3 vorgesehene Aufteilung von Zustandsübergangsfunktionen in durch Eingabesymbole ausgelöste Funktionen und in Folgefunktionen, die vom jeweils

erreichten Zustandswert des Steuerungsprozesses abhängig sind, macht es möglich, die einzelnen Zustandsübergangsbedingungen von ihrem Umfang her überschaubar zu machen und die beabsichtigte Zustandsänderung über eine Folge von Zwischenzuständen zu erreichen, deren Zustandsänderungsfunktionen so häufig angewendet werden, bis sich keine neuen Folgezustände oder Ergebnisse mehr einstellen.

Als besonders vorteilhaft wird die Darstellung der logischen Gleichungen in Form von Binary Decision Diagrams gemäß Anspruch 4 angesehen, weil diese Darstellung bei Abspeicherung der Gleichungen mit noch vertretbarem Aufwand an Speichervolumen gestattet und weil es genügend Werkzeuge für die BDD-Darstellung von Funktionen gibt.

Durch Umsetzen der in der Automatentafel in kompakter Form niedergelegten Funktionen zum Berechnen von Folgezuständen und Ergebnissen in eine an die hardware des oder der Rechner, die die Automatentafel abarbeiten sollen, geeignete Form gemäß Anspruch 5 läßt sich eine besonders effiziente Prozeßsteuerung erreichen.

Anspruch 6 sieht die Verwendung eines Rechners oder Rechnersystems für die automatische Bestimmung der elementaren Zustandsübergangs- und Ergebnisfunktionen der Steuerungsobjekte einer Steuerungsanlage vor. Diese Zustandsübergangs- und Ergebnisfunktionen werden damit durch einen Rechner unabhängig von weiteren Mitwirkungen eines Menschen generiert, was zur Folge hat, daß die fertige Automatentafel, wenn dieser Generierungsprozeß abgeschlossen ist, auch tatsächlich fehlerfrei vorliegt. Der Sicherheitsnachweis für die Anwendung einer Automatentafel erstreckt sich damit im wesentlichen nur auf die Überprüfung der logischen Gesetzmäßigkeiten der in der Fachsprache verwendeten Sprach-

mittel, nicht aber auf die konkreten Einzelobjekte und auf den Generierungsprozeß.

Die automatische Generierung der elementaren Zustandsübergangs- und Ergebnisfunktionen läßt gemäß Anspruch 7 bestimmte Fehler, die sich bei der Formulierung der Fachsprachenausdrücke eingeschlichen haben könnten, bereits in der Erstellungsphase der Automatentafel dadurch erkennbar werden, daß der jeweilige Folgezustand bzw. die Ausgabe nicht ermittelbar ist.

Gemäß Anspruch 8 ist es von Vorteil, die Variablen, die bei einem solchen Fehler mitgewirkt haben, zu einer späteren Fehleranalyse abzuspeichern.

Anspruch 9 beinhaltet die Verwendung der nach den Merkmalen der vorangehenden Ansprüche erstellten Automatentafel für die Steuerung einer ganz bestimmten Anlage, für die diese Automatentafel zuvor erstellt wurde.

Anspruch 10 nennt zwei Alternativen für die Abarbeitung der Automatentafel in einem Rechner oder Rechnersystem.

Der Rechner oder das Rechnersystem, über das das Steuerungssystem betrieben werden soll, ist gemäß Anspruch 11 mit Speichern zum Hinterlegen der Automatentafel und der gültigen Zustandswerte der einzelnen Objekte zu versehen, wobei die Anordnung so getroffen ist, daß gemäß Anspruch 12 die jeweils berechneten Folgezustandswerte die davor aktuellen Zustandswerte in den Speichern überschreiben.

Gemäß Anspruch 13 ist vorgesehen, daß im Rechner bzw. Rechnersystem aus der Automatentafel die einzelnen Zustandsübergangsfunktionen abgelegt sind und daß von einem Eingangs-

besymbol jeweils auf diejenigen Objektzustände hingewiesen wird, die sich bei diesem Eingabesymbol für das Objekt ändern können; nur diese Zustände müssen neu berechnet werden, alle übrigen bleiben bestehen.

5

Entsprechend der bei der Generierung der Automatentafel gewählten Unterteilung in Zustandsübergangs- und Ergebnisfunktionen, die durch Eingaben oder durch den jeweils erreichten Anlagenzustand angereizt werden, werden gemäß

10 Anspruch 14 auch in dem Rechner bzw. Rechnersystem zur Anwendung der Automatentafel entsprechende Bearbeitungsaufteilungen vorgenommen.

Zum Berechnen zeitabhängiger Zustandsänderungsfunktionen
15 sollen die Rechner bzw. Rechnersysteme gemäß Anspruch 15 über Timer verfügen, die nach Abarbeiten der Automatentafel gestartet werden und die bei ihrem Ablauf wieder als Eingaben auf den Automaten selbst einwirken. Dies ist deshalb erforderlich, weil die Automatentafel nur die
20 Generierung von Folgezuständen ermöglicht, nicht jedoch Zeitbedingungen dabei berücksichtigt, wie sie für die Steuerung von Prozeßgeschehen erforderlich sein können.

Die Erfindung ist nachstehend anhand von in der Zeichnung
25 dargestellten Ausführungsbeispielen näher erläutert.

Figur 1 zeigt schematisch den Weg von der Erstellung einer Automatentafel bis zu ihrer Verwendung für die Steuerung eines bestimmten Prozeßgeschehens,
30 Figur 2 die Bildung der Automatentafel,
Figur 3 ein Schema zum Bestimmen der Zustandsübergangs- und
Ergebnisfunktionen eines Objektes und
Figur 4 ein Beispiel für die nach der Erfindung vorgesehene

Darstellung logischer Funktionen in Form von Binary Decision Diagramms.

Figur 1 zeigt in einer schematischen Darstellung den Weg
5 zum Herstellen eines Programms und die Anwendung dieses
Programms zur Lösung der Steuerungsaufgaben eines belie-
bigen Prozeßgeschehens. Die Beschreibung der Steuerungs-
aufgabe und damit die Programmgestaltung basieren auf einer
Modellbetrachtung des ausführenden Schaltwerks als endli-
10 cher Automat AUT mit Ein- und Ausgaben . Dieser Automat
kommuniziert mit seiner Umgebung über zwei Kanäle: Aus
einer Bedienebene empfängt der Automat Informationen über
Bedienungshandlungen, die z. B. durch Eingaben eines Bedie-
ners von Hand an einer Tastatur ausgelöst oder durch ein
15 bestimmtes zuvor festgelegtes Ereignis angereizt werden
können. In umgekehrter Richtung sendet der Automat Informa-
tionen über Systemzustände an die Bedienebene, beispiels-
weise zum Zwecke der Anzeige oder zum Zwecke der Proto-
kollierung. An die Stellebene sendet der Automat Befehle
20 für die Elemente (Objekte) des jeweils betrachteten
Steuerungssystems und er empfängt von dort Meldungen über
die Zustände der einzelnen Prozeßelemente; jede derartige
Meldung verändert seinen internen Zustand, der durch die
Summe aller Einzelzustände seiner Objekte definiert ist.
25
Der Automat AUT verfügt über ein logisches Abbild des Be-
triebszustandes des Steuerungssystems (Zustandsraum), das
zum Teil eine direkte physische Entsprechung besitzt in
Form von Einzelzuständen der Prozeßelemente, zum Teil aber
30 auch ausschließlich logische Bedeutung hat in Form von
Zustandswerten von in bestimmter Form logisch verknüpften
Elementen. Zum besseren Verständnis dieser und der
folgenden Ausführung sei dem Automaten eine bestimmte
Steuerungsaufgabe zugewiesen, nämlich die Steuerung eines

Stellwerkes für den Eisenbahnbetrieb. Die dem Automat bekannten und von ihm verwalteten Zustände beziehen sich dann auf die gegenständlich vorhandenen Prozeßelemente und betreffen z. B. die Lage von Weichen, den Frei- und
5 Besetztzustand von Gleisabschnitten und die jeweils angeschalteten Lichtsignalbegriffe sowie die Topographie der zu steuernden Anlage. Zu den Betriebszuständen mit ausschließlich logischer Bedeutung gehören z. B. die Bildungszustände der einzelnen Fahrstraßen während ihres
10 Aufbaus und Auflösens.

Um seine Aufgabe als freiprogrammierbares Schaltwerk zur Steuerung des Prozeßgeschehens "Stellwerk" ausführen zu können, muß der Automat neben der Kenntnis der aktuellen
15 Zustände der zu steuernden Objekte auch Bescheid wissen darüber, wie aus den gegenwärtigen Objektzuständen auf Folgezustände überzugehen ist. Ein neuer Folgezustand eines Objektes kann beispielsweise in einer neuen Weichenlage, in einer Änderung des Freimeldezustandes oder in einer
20 Änderung eines angeschalteten Signalbegriffes bestehen; eine Ausgabe an den Prozeß als Ergebnis eines bestimmten Eingabesymbols und eines Objektzustandes kann z. B. in einem Stellimpuls zum Einleiten eines Weichenumlaufs oder im Anschalten eines Lichtsignalstellers bestehen. Üblicher-
25 weise greift der Automat AUT hierzu auf eine Automatentafel AT zu, in der die jeweiligen Folgezustände eines Objektes und ggf. zu veranlassende Ausgaben (Ergebnisse) in Abhängigkeit von den gegenwärtigen Objektzuständen und den jeweiligen Eingabesymbolen (Eingaben und Meldungen) auf-
30 listet sind. Diese Automatentafeln sind, abhängig von der Komplexität des jeweiligen Steuerungssystems, meist recht aufwendig.

Bei umfangreichen Steuerungssystemen wie beispielsweise bei Stellwerken ist die Anzahl der Objekte des Steuerungssystems und die Komplexität der logischen Verknüpfungen, also die Anzahl der gegenwärtigen und der zukünftigen Objektzustände so groß, daß diese nicht mehr explizit, d. h. durch vollständige Aufzählung, vom Menschen angegeben werden können; die Wahrscheinlichkeit, daß bei der Erstellung der Automatentafel durch den Menschen fehlerhafte Zustände formuliert werden, ist außerordentlich groß. Die Erfindung, soweit sie sich auf die Erstellung der Automatentafel bezieht, geht von der Erkenntnis aus, daß die Zustandsmenge und die Zustandsübergänge der Objekte eines Steuerungssystems wegen der Gleichheit und der Ähnlichkeit der physikalischen und logischen Objekte ein hohes Maß an Regelmäßigkeit aufweisen und daß auch bei großen Systemen die Zustandsübergänge meist nur von einem sehr kleinen Ausschnitt des Zustandsraumes, d. h. einzelnen Zuständen dieser Objekte, abhängen und daß diese mit ihrem Folgezustand auch nur einen kleinen Teil der Zustandsmenge verändern. Sie geht deshalb von der Überlegung aus, daß es möglich sein müßte, eine Automatentafel zu erstellen, in der das Zustandsübergangs- und Ergebnisverhalten der Objekte durch logische Funktionen beschreibbar ist, die auf bestimmte augenblickliche Zustände der Objekte anzuwenden sind. d. h. die Folgezustände und Ergebnisse liegen nicht vorab vorausbestimmt in der Automatentafel vor, sondern werden aus den gegenwärtigen Zuständen unter Verwendung von in der Automatentafel hinterlegten Funktionen fallweise berechnet. Die Erfindung sieht vor, daß das komplexe Zustandsübergangs- und Ergebnisverhalten der Objekte in mehreren aufeinander aufbauenden Schritten analysiert und in Funktionen umgesetzt wird, wobei die Bedingungen zum Beginn der Generierungsphase noch recht allgemein gehalten und in ihrer Formulierung möglichst prozeßnah ausgeführt

sein sollen, um dann schrittweise immer komplexer zu werden, bis sie schließlich ihren Endzustand in den elementaren objektbezogenen Zustandsübergangs- und Ergebnisfunktionen der Automatentafel des Steuerungssystems gefunden haben. Nach einer besonders vorteilhaften Ausbildung des erfindungsgemäßen Verfahrens sollen die komplexen Zustandsübergangs- und Ergebnisfunktionen innerhalb der Automatentafel in einer komprimierten Form dargestellt werden, die einerseits den Speicheraufwand für die Automatentafel begrenzt und andererseits, was mindestens ebenso wichtig ist, die Bearbeitungszeiten für die Übergangsfunktionen kurz macht. Ein wesentlicher Vorteil der vorausbezeichneten Automatentafel ist der, daß das Funktionsverhalten des Gesamtsystems vor Einschalten des Systems bekannt ist.

Für die Erstellung der Automatentafel, die nachfolgend anhand der Figur 2 erläutert ist, benötigt der Entwickler Eingabewerkzeuge, mit denen er die Betriebsordnung BO, also die anwenderorientierten Regeln für die Bearbeitung des Betriebsgeschehens und die Projektierungsdaten PD der Anlage in Form der Topographie der konkreten Anlageobjekte, der erfindungsgemäßen Generierung zuführt. Wie das im einzelnen geschieht, ist aus Figur 2 ersichtlich, die hierzu wiederum Bezug nimmt auf die übrigen Figuren der Zeichnung.

In der Betriebsordnung BO ist festgelegt, unter welchen Bedingungen auf ein Objekt z. B. des Typs Weiche eingewirkt werden darf. So darf eine Weiche nur dann umgestellt werden, wenn sie frei und nicht verschlossen ist. Diese Bedingungen sind für eine entsprechende Umstellerlaubnis aufzuprüfen und müssen in geeigneter Form abgespeichert vorliegen. Im ersten Schritt, der sogenannten Zustandsat-

tributkodierung, erfolgt die Bestimmung des Speicheraufwandes, der zum Beschreiben des Zustandsraumes der einzelnen Objekttypen als Teil des Zustandsraumes des Gesamtsystems benötigt wird. Unter Objekttypen werden hier
5 die unterschiedlichen Arten von Objekten verstanden bezogen auf das Steuersystem "Stellwerk", also Weichen, Signale, Gleise und Fahrstraßen. Jedes Zustandsattribut, d. h. jede Zustandsart, beinhaltet eine von zwei möglichen Alternativen. So kann z. B. bezüglich des Freimeldezustandes einer
10 Weiche gesagt werden, daß sie entweder freigemeldet oder besetztgemeldet ist. Für die Darstellung dieses Zustandsattributes wird ein Bit benötigt, das abhängig vom jeweiligen Wert, durch eine Meldung entweder auf high oder low gesetzt wird. Eine Weiche kann sich, was ihre Lage angeht,
15 üblicherweise nur entweder in der Pluslage oder in der Minuslage befinden, oder aber sie läuft gerade in eine dieser Endlagen und befindet sich zwischen diesen. Für die Darstellung dieser beiden Alternativpaare werden zwei Bits zur Zustandsbeschreibung benötigt. Insgesamt gibt es für
20 die Zustandsattributbeschreibung von Weichen noch sehr viel mehr Parameter wie z. B. den Fahrstraßenverschluß, die Auffahrmeldung oder die Störungsmeldung. Alle diese Zustandsattribute werden für jeden Objekttyp der Steuerungsanlage bestimmt und in ihrer Zuordnung zueinander
25 festgelegt, d. h. es wird genau gesagt, an welcher Stelle innerhalb der Zustandsattributbeschreibung z. B. der Freimeldezustand und wo der Lagezustand kodiert hinterlegt ist. Ferner werden in der Phase der Zustandsattributkodierung die Anzahl und die Lage der Bits im Zustandsvektor für
30 die Beschreibung der unveränderlichen Merkmale der verschiedenen Objekttypen ermittelt und als Platzhalter für die konkreten Objektdaten typbezogen hinterlegt. Am Ende der Zustandsattributkodierungsphase liegen für alle Objekttypen eines Steuerungssystems Schablonen vor, die

besagen, welche Zustände und Merkmale für ein Objekt des betreffenden Typs beschrieben werden müssen, damit der Automat später ein umfassendes Bild über den Zustandsraum des betreffenden Objektes gewinnen kann.

5

Der zweite Schritt, die sogenannte Zustandsinstantiierung, beinhaltet die Anwendung der zuvor gefundenen Schablonen auf die in der Anlagenbeschreibung, den Projektierungsdaten PD der Anlage, enthaltenen Steuerungsobjekte. Hierdurch entsteht ein Vektor von Aussagenvariablen, die den Zustandsraum des Gesamtsystems repräsentiert. Würde man z. B. zum Beschreiben der möglichen Zustände des Elementtyps Weiche drei Bits benötigen und würde die Anlage hundert Weichen enthalten, so würde der Bitvektor für die Beschreibung aller Weichen des Systems 300 Bits umfassen. Tatsächlich ist der Speicherbedarf allein für die Beschreibung der Weichen sehr viel größer, weil für die Weichen neben den veränderbaren Zustandsangaben - wie bereits ausgeführt - auch unveränderbare Merkmale zu hinterlegen sind. Dies geschieht dadurch, daß für jede Weiche ein zugehöriger Name festgelegt und die Lage dieser Weiche im Gleisplan exakt angegeben wird. Hierzu wird die Gleisanlage z. B. in Segmente unterteilt, die jeweils der Position eines der gegenständlichen Objekte entsprechen. Die einzelnen Segmente sind an den Endpunkten mit Nachbarsegmenten verbunden. Ein Segment kann dabei einem Objekt mit einer realen Längenausdehnung, z. B. einem Gleis, entsprechen, es kann aber auch lediglich als Markierung für den Standort eines punktförmigen Objektes, z. B. eines Signals, dienen. Auch einem solchen punktförmigen Objekt ist ein Segment mit zwei unterscheidbaren Endpunkten zugeordnet, wodurch sich für dieses Objekt eine fahrtrichtungsabhängige Kennzeichnung ergibt. Die Punkte, an denen die Segmente miteinander verbunden sind, können willkürlich numeriert werden und dienen

30

- zur Beschreibung von Pfaden innerhalb der zu steuernden Anlage, die zu verschiedenen Beschreibungszwecken wie z. B. für einen Fahrweg, einen Durchrutschweg oder für die Flankenschutzsuche benötigt werden. Die Topographie einer Gleisanlage wird damit z. B. durch einen beschrifteten Graphen repräsentiert, aus dem die Verbindungsstruktur der Objekte hervorgeht. Für die Beschreibung von Fahrstraßen und Fahrwegen werden die jeweils zu beteiligenden Fahrwegelemente aufgelistet und für die einzelnen Fahrstraßen oder Fahrwege objektbezogen abgelegt. Eine andere Möglichkeit zur Topographiebeschreibung besteht z. B. in der listenmäßigen Erfassung von Nachbarschaftsbeziehungen.
- 15 Neben der Kenntnis der Zustände und Merkmale der einzelnen Objekte benötigt der Automat über die Automatentafel Kenntnis über das Zustandsübergangs- und Ergebnisverhalten der Objekte, um hieraus Folgezustände der Objekte und Ausgaben ermitteln zu können. Nach der Lehre der vorliegenden
- 20 Erfindung soll das Zustandsübergangs- und Ergebnisverhalten der einzelnen Objekte durch Angabe von Funktionen beschrieben werden, nach denen sich die Folgezustände und die Ergebnisse jeweils aus den augenblicklichen Zuständen der betreffenden Objekte und den auf sie wirkenden Eingabesymbolen errechnen lassen. Für die Erstellung der Automatentafel ist es daher erforderlich, die entsprechenden Übergangsbedingungen zu ermitteln und festzulegen. Dies geschieht zunächst auf Elementtypebene, d. h. ohne konkreten Bezug zu einem bestimmten Objekt. Die Beschreibung der
- 25 logischen Gesetzmäßigkeiten des Zustandsübergangsverhaltens der Objekte beim Übergang aus einem gegenwärtigen Zustand in einen direkten Folgezustand sowie die Beschreibung des Ergebnisverhaltens der Steuerungsobjekttypen ergibt sich aus der Betriebsordnung BO, in der anzugeben ist, unter
- 30

welchen Bedingungen sich derartige Änderungen bzw. Ausgaben ergeben sollen. Für die spätere Generierung der Automaten-tafel werden diese logischen Gesetzmäßigkeiten in der Betriebsordnung in einer anwendungsorientierten Fachsprache beschrieben, deren Sprachmittel die Bedingungen für das Erreichen des jeweiligen Folgezustandes oder Ergebnisses als logische Verknüpfung von elementaren Zustandswerten angeben. Konkret für den Anwendungsfall Weiche heißt dies z. B., daß eine Weiche nach Eingabe des entsprechenden Steuerkommandos (Eingabesymbol) nur dann umlaufen darf, wenn sie auch umstellbar ist. "Umstellbar" als Sprachbegriff einer anwendungsbezogenen Fachsprache beinhaltet begrifflich, daß die Weiche frei ist und sich in einer Endlage befindet. Dabei kann sie nur dann z. B. nach Minus laufen, wenn sie zuvor die Pluslage eingenommen hatte. Die Transitionsbedingung (Zustandsübergangsbedingung) einer Weiche für den Lauf nach Minus auf Elementtypebene lautet dann z. B.

umstellbar(Weiche) $\wedge$ plus(Weiche) ** laufNachMinus(Weiche)

Der Folgezustand, gekennzeichnet durch **, "Lauf nach Minus(Weiche)" wird fachsprachlich durch die Bedingungen "umstellbar(Weiche) $\wedge$ plus (Weiche)" beschrieben. Die Fachsprache und deren Sprachmittel die vom Entwickler gewählt, optimiert und festgelegt werden, sind immer problembezogen und unterscheiden sich in Abhängigkeit vom jeweiligen Anwendungsfall. Für den Anwendungsfall Stellwerk sind die Sprachmittel unabhängig von der Topographie der zu steuernden Anlage und den Funktionsbedingungen der jeweiligen Betriebsordnung, die von Anwender zu Anwender verschieden sein kann, gleich, wobei allerdings bei unterschiedlichen Betriebsordnungen verschiedener Betreiber die Sprachmittel durchaus unterschiedliche logische Ver-

knüpfungen von Zustandswerten beinhalten können, so beispielsweise, wenn eine Weiche nur umstellbar sein darf, wenn sie zuvor auch verschlossen war. In diesem Falle muß entweder in den Ausgangs-Zustandswert der Weichenlage der

5 Verschluß impliziert sein oder aber es muß ein zusätzliches Zustandsattribut bereitgestellt werden, das in die Transitionsbedingungen aufzunehmen ist.

In einem folgenden Verfahrensschritt erfolgt die Instanti-

10 ierung der Transitionsbedingungen auf Objektebene. Ähnlich wie bei der Zustandsinstantiierung werden hier die auf den jeweiligen Objekttyp bezogenen Transitionsbedingungen durch Anwendung der zuvor gefundenen Schablone auf die einzelnen Objekte bestimmt und damit vervielfältigt nach der Anzahl

15 der vorhandenen Objekte. Bezogen auf das vorstehend angezogene Beispiel des Umlaufs einer Weiche nach Minus ergibt sich z. B. für die Weiche W001 eines konkreten Gleisplans die Transitionsbedingung

20 umstellbar(W001) $\wedge$ (W001)** laufNachMinus(W001).

Auch nach der Transitioneninstantiierung liegen die Änderungsbedingungen noch in fachsprachlicher Ausführung vor, die vom Anwender noch ohne weiteres lesbar sind. Dies die

25 vom Anwender ohne weiteres lesbar ist. Dies ändert sich in den folgenden Verfahrensschritten, von denen der erste die sogenannte Expansion- und Transitionkodierung beinhaltet. Hierunter wird verstanden die Ermittlung der elementaren Zustandsübergangs- und Ergebnisfunktionen aller

30 Steuerungsobjekte und ihre Kodierung in binärer Form. Beim Expandieren werden zunächst nur die zuvor verwendeten Sprachmittel der Fachsprache durch die entsprechenden elementaren Zustandswerte ersetzt. Für das angenommene Beispiel heißt dies

(plus(W001) V minus(W001)) ^ frei (W001) ^ plus(W001)**
laufNachMinus(W001)).

- 5 Die so gefundenen elementaren Zustandsübergangsfunktionen für die einzelnen Objekte werden nun kodiert unter Verwendung der zuvor für die Objekte festgelegten Kodierungsparameter. Dabei gibt es nicht nur die erläuterten Zustands-
10 entsprechende Zustandswerte für die Folgezustände; für beide wurden bei der Zustandsinstantiierung entsprechende Speicherbereiche reserviert.

- 15 In entsprechender Weise wird auch das Zustandsübergangs- sowie Ergebnisverhalten aller anderen Objekte durch fachsprachliche Umschreibungen von Änderungs- bzw. Ergebnisbedingungen bestimmt und durch Instantiierung, Expansion und Kodierung objektbedingt beschrieben.

- 20 Am Ende der Expansions- und Transitionskodierungsphase sind für alle Objekte alle möglichen gegenwärtigen Zustände in Zuordnung zu den zukünftigen Zuständen bekannt, wobei immer nur jeweils einer der gegenwärtigen Zustandswerte eines Objektes den tatsächlichen Zustandswert trifft. Um nun von den möglichen gegenwärtigen Zustandswerten zu dem jeweils
25 zugehörigen zukünftigen Zustandswert zu gelangen, wird aus den zuvor bestimmten und für ein bestimmtes Objekt bei einem bestimmten Eingabesymbol geltenden Transitionsbedingungen ein Boole'sches Gleichungssystem gewonnen und mit der Methode der Boole'schen Unifikation gelöst. Die Lösung
30 eines Boole'schen Gleichungssystems geschieht analog zur Lösung eines linearen Gleichungssystems. Die Boole'schen Gleichungen werden so aufgestellt, daß sie eine Formalisierung der Aussage "f_{inp} (z. B. umstellen) respektiert alle Tansitionen zu inp (z. B. umstellen)" bilden. Jede

Gleichung des Systems ist von der Form

$f(...C_i...Z_j...) = 1$, d. h. wahr, wobei der Term auf der linken Seite aus einer kodierten, expandierten Transition her vorgeht und sowohl Aussagevariablen enthält, die sich

5 auf den aktuellen Zustand des Objektes beziehen ($...C_i...$) als auch auf seinen Folgezustand ($...Z_j...$). Für die Lösung des Gleichungssystems werden die Aussagevariablen C_i als Konstanten betrachtet und das Gleichungssystem nach Z_j aufgelöst. Dadurch ergibt sich jedes Z_j als Funktion $Z_j =$
10 $f_j (...C_i...)$. Die Funktionen f_j bilden gemeinsam die Funktion f_{inp} . Jedes f_j bestimmt den Wert des j -en bits im Folgezustand in Abhängigkeit von den Bits des gegenwärtigen Zustands.

Diese Zusammenhänge sind nachfolgend anhand der Figur 3
15 graphisch verdeutlicht. Das Beispiel bezieht sich auf das Umstellen einer Weiche. Liegt die Weiche in der Minuslage und ist sie frei, so kann sie auf einen entsprechenden Umsteuerbefehl (Eingabesymbol) nach Plus umlaufen und sie bleibt dabei frei. Liegt die Weiche in Pluslage und ist sie
20 frei, so kann sie auf einen entsprechenden Umsteuerbefehl nach Minus umlaufen, wobei sie dann ebenfalls frei bleibt. Unterhalb dieser beiden in Figur 3 angegebenen Transitionsbedingungen sind jeweils drei einen möglichen Gesamtzustand einer Weiche im Zustandsvektor beschreibende elementare
25 Zustandswerte aufgeführt. Die ersten beiden Zustandsbits beziehen sich entsprechend der bei der Zustandsinstantiierung festgelegten Zuordnung auf die Weichenlage, das dritte Zustandsbit bezeichnet den augenblicklichen Frei/Besetztzustand einer Weiche. Die jeweils links
30 stehende Bitkombination kennzeichnet den augenblicklichen Zustand einer Weiche, die rechte Bitkombination den zugehörigen Folgezustand wie er durch das Eingabesymbol "umstellen" angereizt wird. Im ersten Fall wird der gegenwärtige Zustandswert durch drei Bits der Wertigkeit 1

- dargestellt. Der neue Zustand der Weiche nach Beginn des Umlaufs beinhaltet Bitkombinationen, die dem Lauf nach plus und dem Freizustand der Weiche entsprechen; die Bitkombinationen für den Folgezustand entsprechen
- 5 inhaltlich den Bitkombinationen für die Kennzeichnung der entsprechenden augenblicklichen Zustandswerte. Für den Lauf nach plus war bei der Zustandsinstantiierung die Bitfolge 00 festgelegt worden. Durch Anwendung bekannter mathematischer Algorithmen werden nun durch Dreiecke
- 10 verdeutlichten Funktionen bestimmt, welche die Eingangsvariablen 11 in die dargestellten Ausgangsvariablen 00 umsetzen. In entsprechender Weise verhält es sich mit den Funktionen beim Umsetzen der Eingangsvariablen 10 für die aus der Pluslage umlaufende Weiche beim Umsetzen in die
- 15 Ausgangsvariablen 01. Durch Anwendung der Methode der Boole'schen Unifikation gelangt man in bekannter Weise zu Funktionen f1 und f2, die jeweils für die im Beispiel zwei Eingangsvariablen und eine Ausgangsvariable Gültigkeit haben, unabhängig von deren jeweiligen binären Werten.
- 20 Würde man dem Automaten die so gefundenen Funktionen über die Automatentafel mitteilen, so wäre er in der Lage, die Steuerung des Prozeßgeschehens zu übernehmen. Um die nach der Erfindung vorgesehenen Beschreibung des Zustandsübergangs- und Ergebnisverhaltens der Objekte durch
- 25 Zustandsübergangs- und Ergebnisfunktionen in einer einzigen Automatentafel für das gesamte System handhabbar zu machen, sieht das erfindungsgemäße Verfahren vor, die Boole'schen Gleichungen für die Beschreibung des Zustandsübergangs- und Ergebnisverhaltens in kompakter Weise darzustellen, beispielsweise in Form von Binary Decision Diagramms (BDD).
- 30 Die Darstellung logischer Funktionen als Binary Decision Diagramms als solche ist an sich bekannt (IEEE Transaction on Computers, Vol. C35 Nr. 8, August 1986, Seiten 677 bis 691); ihr Einsatz zu einer effizienten Speicherung von

Steuerungsinformationen ist jedoch neu. Die Ermittlung der Automatentafel durch Lösen der zuvor ermittelten Gleichungen für das Zustandsübergangs- und Ergebnisverhaltens geschieht unter Anwendung der kompakten Darstellungsform deren Gleichungen mit dem Ergebnis, daß auch die Zustandsübergangs- und Ergebnisfunktionen dieser kompakten Darstellungsform genügen. Die entstehende Automatentafel kann für ein konkretes Zielsystem in eine andere, für dieses Zielsystem optimierte Darstellung überführt werden.

Die Wirkungsweise der Binary Decision Diagramms (BDD) für die effiziente Abspeicherung der Funktionen zum Ermitteln von Folgezuständen und Ergebnissen aus den augenblicklichen Zustandswerten von Objekten ist nachfolgend anhand der Figur 4 näher erläutert. Es ist angenommen, daß die Funktion $f(A, B, C)$ zum Umsetzen von Eingangsvariablen in Ausgangsvariable der Bedingung $A \wedge B \vee C$ genügt. Für diese Funktionen ergibt sich der in Figur 4 dargestellte Graph, dessen Schichten den Variablen der Boole'schen Funktion entsprechen. Dieser Graph läßt sich als graphische Darstellung einer Entscheidungstabelle auffassen, weil das Auswerten der Funktion geschieht, indem ein Pfad vom Stamm bis zu einer Wurzel verfolgt wird, wobei die Entscheidung über den jeweiligen Weg an den Verzweigungspunkten durch die Verknüpfungsregel der Boole'schen Funktion vorgegeben ist. Falls die Variable, die dem jeweiligen Niveau entspricht, den Wahrheitswert 0 besitzt, verweist diese Variable auf den jeweils linken Suchweg, weist sie den Wahrheitswert 1 auf, dann verweist sie auf den rechten Suchweg. Die Wurzelnenden repräsentieren die Wahrheitswerte 0 oder 1: Endet ein Suchlauf bei einer 0-Wurzel, ist der Wert der Funktion für die konkreten Variablen gleich 0, endet er bei einer 1-Wurzel, so ist der Wert gleich 1. Für eine Funktion $f = A \wedge B \vee C$, bei der die Variable A den Werte 0, B den

Wert 1 und C den Wert 0 haben, ergibt sich der im oberen Graph der Figur 4 gestrichelt angedeutete Suchweg, der als Ergebnis den Wert 0 angibt.

Die durch die Anwendung der Binary Decision Diagramms bewirkte Verkürzung des Laufweges beim Bestimmen einer Funktion beruht darauf, daß jeweils mehrfach vorkommende Teilwurzeln jeweils nur einmal abgespeichert sind, um so zu einer möglichst kompakten Darstellung zu gelangen. Das dem Graph im oberen Teil der Figur 4 entsprechende Diagramm in BDD-Darstellung ist im unteren Teil der Figur 4 angegeben. Ganz offensichtlich ist es so, daß die Funktion nur dann den Wert 1 annimmt, wenn sowohl A den Wert 1 als auch B den Wert 1 haben, egal welchen Wert C dabei einnimmt, oder wenn C den Wert 1 einnimmt, egal welchen Wert A oder B aufweisen. Bezüglich des ersten Terms $A = 1, B = 1$ endet die Wurzel am Knoten U, bezüglich der Variablen C beginnt der eigentliche Entscheidungsvorgang erst am Knoten V. Zu diesem Knoten gelangt man auf dem rechten Strang, wenn sowohl A als auch B jeweils den Wert 1 einnehmen und auf den linken Strang, wenn diese Variablen jeweils den Wert 0 einnehmen. Auf diesem linken Strang verfährt man auch wenn nur eine der Variablen A oder B den Wert 0 einnimmt, weil die logische Verknüpfung dieser Werte ebenfalls zum Wert 0 führt. Unabhängig von den logischen Werten der Variablen A und B endet der Suchvorgang für die Variable $C = 0$ am Knoten W und bei der Variablen $C = 1$ am Knoten X. Die an den Endknoten U, W und X angegebenen Werte stellen das Ergebnis der angenommenen Funktion dar.

In ein und demselben Binary Decision Diagramm lassen sich mehrere Boole'sche Funktionen darstellen, indem man für jede Funktion einen Einstiegs-knoten markiert und von dort zu den Wurzelspitzen durchlaufend die entsprechende Funktion in der geschilderten Weise darstellt. Solche Diagramme

für mehrere Boole'sche Funktionen werden als BDD-Geflecht bezeichnet.

Das Auffinden der Ergebniswerte von Funktionen auf kürzesten Weg nach dem Prinzip von BDD's ist nicht auf die graphische Darstellung in einem sich verzweigenden Graph beschränkt, sondern läßt sich durch eine entsprechende Programmstruktur auch in einem Rechner realisieren, wobei es Rechner Routinen für die Bestimmung von BDD-Darstellungen aus beliebigen Funktionen gibt. Die Erfindung nutzt die Existenz derartiger BDD's, um die Zustandsübergangs- und Ergebnisfunktionen, die sie zum Bestimmen der jeweiligen Folgezustände und Ergebnisse bei der Steuerung eines umfangreichen Prozeßgeschehens verwendet, auf besonders effiziente Weise darzustellen und in einem Speicher zu hinterlegen. Der Vorteil der Hinterlegung der Übergangsfunktionen in BDD-Darstellung ist neben der Begrenzung des Speicherbedarfes die Verkürzung der Bearbeitungszeit zum Auffinden der Folgezustandswerte, was sich an dem sehr einfachen Ausführungsbeispiel der Figur 4 daran zeigt, daß zur Lösung der angenommenen Funktion auf herkömmliche Art und Weise stets vier Knoten durchlaufen werden müssen, während dies in BDD-Darstellung nur drei Knoten sind. Dieser Einsparungseffekt ist bei komplizierteren Funktionen wie sie sich real bei der Bestimmung der Übergangsfunktionen großer Steuerungssysteme darstellen, noch beträchtlich höher als in dem simplen Ausführungsbeispiel der Figur 4.

Die Sammlung der objektbezogenen Funktionen zusammen mit Angaben, welches Bit eines Nachfolgezustandes unter welchen Bedingungen (Eingabesymbol) jeweils ausgerechnet werden soll, bildet die Automatentafel, wie sie durch die Anwendung des erfindungsgemäßen Verfahrens dargestellt ist. Auf die Anwendung der Automatentafel in einem Rechner oder

Rechnersystem zur Steuerung eines Prozeßgeschehens wird später näher eingegangen.

Die Beschreibung der Steuerungsfunktionen eines Stellwerkes oder eines anderen umfangreichen Prozeßgeschehens jeweils
5 als Abfolge eines gegenwärtigen Zustandes und eines Folgezustandes (Fig. 3) ist außerordentlich aufwendig. Oftmals ist es einfacher, einen Zustandsübergang nicht in Abhängigkeit von einem bestimmten Eingabesymbol zu beschreiben, sondern in Abhängigkeit von einem schon existierenden
10 internen Zwischenzustand des Steuerungssystems. Nach einer besonders vorteilhaften Ausgestaltung des erfindungsgemäßen Verfahrens werden deshalb einzelne Übergangsfunktionen nicht unmittelbar durch ein Eingabesymbol angereizt, sondern durch den jeweils erreichten Zwischenzustand der
15 Anlage. Für jedes Objekt sind die zugehörigen Zustandsübergangsfunktionen, deren Ausführung vom jeweils erreichten Zwischenzustand des Objektes abhängig sind, festzulegen und diese Funktionen kommen bei der Abarbeitung der Automatentafel zur Anwendung, sobald das Eingabesymbol
20 auf das betreffende Objekt verwiesen hat und die dadurch angereizte direkte Transitionbedingung abgearbeitet wurde. Der erreichte Folgezustand veranlaßt dann die Abarbeitung der Transitionsbedingungen, die durch den Folgezustand angereizt werden. So werden jeweils nach Abarbeitung eines
25 Eingabesymbols die zustandsorientierten Zustandsübergangsbedingungen solange aufeinanderfolgend abgearbeitet, bis sich kein neuer Folgezustand des Gesamtsystems und keine Ausgabe mehr einstellt. Danach kann sich ein folgendes Eingabesymbol durchsetzen und eine weitere
30 Behandlung dieses oder eines anderen Objektes veranlassen.

Ein besonderer Vorteil des erfindungsgemäßen Verfahrens besteht darin, daß dann, wenn die Gesamtheit der Transitionen eines Eingabesymbols versehentlich so spezifiziert

wurde, daß zu einem gegenwärtigen Zustand sich widersprechende Bedingungen für den Folgezustand angegeben sind, schon in der Entwicklungsphase, also bei der Erstellung der Automatentafel, dieser Fehler dadurch erkannt wird, daß es
5 keine alle Übergangsfunktionen eines Objektes beschreibende Boole'sche Gleichung gibt, d. h. das Gleichungssystem keine Lösung besitzt. Die weitere Generierung der Automatentafel wird dann unterbrochen und aus der bis dahin anfallenden Information kann eine Beschreibung der Zustände erzeugt
10 werden, für die das Automatenverhalten undefiniert ist. Alle Gegenwartszustände, zu denen Folgezustände widersprüchlich spezifiziert wurden, können zur Fehleranalyse abgespeichert und ausgegeben werden.

Nach dem Abspeichern der Zustandsübergangs- und Ergebnisfunktionen in der Automatentafel läßt sich eine Aussage
15 darüber gewinnen, wie lange ein Rechner, der die Automatentafel zur Lösung eines bestimmten Problems abarbeiten soll, für die Bearbeitung der einzelnen Vorgänge benötigen wird. Es läßt sich so abschätzen, ob der Automat Echtzeitverhalten hat.
20

Die Erfindung sieht in vorteilhafter Weise den Einsatz eines Rechners zur Durchführung des erfindungsgemäßen Verfahrens der Erstellung einer Automatentafel vor; es kann sowohl ein Einzelrechner als auch ein Rechnersystem zur
25 Anwendung kommen. Seine Aufgabe ist es u. a., die elementaren Zustandsübergangs- und Ergebnisfunktionen sämtlicher Steuerungsobjekte der Steuerungsanlage durch Anwendung der definierten logischen Gesetzmäßigkeiten des Zustandsübergangs- und Ergebnisverhaltens der Steuerungsobjekttypen auf
30 die Steuerungsobjekte der konkreten Steuerungsanlage zu ermitteln, d. h. er hat mindestens die Expansion und Transitionen-Codierung, ggf. auch die Zustands- und Transitionen-Instantiierung durchzuführen. Diese Vorgänge laufen

dann unabhängig von der Mitwirkung eines Bedieners ab und sind daher weitestgehend fehlerunempfindlich. Eine weitere Aufgabe dieses Rechners ist die Ermittlung der elementaren Übergangs- und Ergebnisfunktionen und die Darstellung
5 dieser Funktionen z. B. in Form von Binary Decision Diagrams.

Die Arbeit dieses Rechners nach der Erstellung der Automatentafel ist beendet. Die Automatentafel kann in Gestalt einer Diskette, eines EPROMS oder eines sonstigen Speicher-
10 mittels vom Entwicklungsrechner getrennt und einem oder mehreren Rechnern gegenständlich oder inhaltlich zugeführt werden, die für die Steuerung des eigentlichen Prozeßgeschehens vorgesehen sind. Dieser Rechner oder dieses Rechnersystem dient dann zum Abarbeiten der nach dem erfindungsgemäßen Verfahren und der Einrichtung zur Durchführung
15 des Verfahrens erstellten speziellen Automatentafel. Dabei kann die Anordnung so getroffen sein, daß auf dem Rechner oder Rechnersystem ein an dessen spezielle Hardware angepaßtes Automatenprogramm abläuft, das mit den in der Automatentafel abgelegten elementaren Zustandsübergangs- und
20 Ergebnisfunktionen arbeitet oder aber dem Rechner/Rechnersystem kann die Automatentafel in kodierter Form über einen Compiler in für ihn abarbeitbarer Form zugeführt werden. Das Automatenprogramm erhält aus dem zu steuernden Prozeß
25 Zustandmeldungen der Steuerungsobjekte sowie Eingaben als Eingabesymbole und es veranlaßt Ausgaben an die Steuerungsobjekte sowie Meldungen als Ausgabesymbole (Fig.1). Die jeweiligen Ausgaben und Meldungen werden durch die dem Automaten zugeführten Eingabesymbole sowie die
30 jeweils erreichten internen Zwischenzuständen der Objekte nach Maßgabe der aus der Automatentafel aufgerufenen Zustandsübergangs- und Ergebnisfunktionen errechnet und überschreiben dann entweder bestimmte Zustandswerte im

Zustandsvektor oder sie veranlassen die Ausgabe von Stellimpulsen für die Prozeßelemente.

Der Rechner bzw. das Rechnersystem des Automaten verfügt über Speicher zum Ablegen der Automatentafel und zum Verwal-
5 ten der jeweils gültigen Zustandswerte für alle vorkom-
menden Steuerungsobjekte. Die Zustandswerte der einzelnen
Objekte werden in Form eines Datenvektors festgehalten und
der Rechner bzw. das Rechnersystem überschreibt die jeweils
aktuellen Zustandswerte beim Berechnen von Zustandsänderun-
10 gen durch die jeweils sich ergebenden neuen Werte und ver-
ändert damit den Zustandsvektor.

Für jeden einem Objekt zugeordneten Folgezustandswert, der
verschieden sein kann vom jeweiligen Ausgangszustandswert,
ist eine eigene Zustandsübergangsfunktion zum Auffinden des
15 jeweiligen Folgezustandes hinterlegt (Fig. 3). Jedes eine
Zustandsänderung veranlassende Eingabesymbol verweist min-
destens mittelbar auf das oder die jeweils zu ändernden Zu-
stände eines Steuerungsobjektes sowie die in Frage kommen-
den Zustandsänderungsfunktionen. Aus diesem Grunde brauchen
20 diese Änderungsfunktionen nur auf diejenigen Variablen des
Datenvektor angewendet werden, die sich tatsächlich ändern;
alle sich nicht ändernden Zustandswerte bleiben unverändert
erhalten. Durch diese Maßnahme wird der zeitliche Aufwand
für die Bestimmung der Folgezustände und Ergebnisse und
25 auch der Aufwand für die Speicherung der Änderungsfunktio-
nen gegenüber einer Ausbildung verringert, bei der sich die
Änderungsfunktionen auf alle Variablen des betreffenden
Objektes beziehen.

Entsprechend der Generierung der Automatentafel gibt es
30 neben Zustandsübergangs- und Ergebnisfunktionen, auf die
Eingabesymbole direkt hinweisen, weitere Zustandsübergangs-
und Ergebnisfunktionen in der Automatentafel, deren Behand-

- lungen allein vom Erreichen eines bestimmten inneren Zustandes der Steuerungsanlage als Ergebnis der Ausführung einer vorangegangenen Zustandsänderungsfunktion abhängen. Das Abarbeiten dieser vom inneren Zustand der Steuerungs-
- 5 anlage abhängigen Zustandsänderungsfunktionen erspart die vollständige eingabesymbolbezogene Beschreibung dieser Funktionen und die mehrfache Hinterlegung solcher Zustandsübergangsfunktionen, die im Rahmen einer Objektbehandlung mehrfach abgearbeitet werden können.
- 10 Bei der Steuerung eines Prozeßgeschehens kommt es häufig darauf an, daß bestimmte Zeitparameter eingehalten werden. So wird z. B. ein eingeleiteter Stellvorgang abgebrochen, wenn der Stellvorgang nicht innerhalb einer bestimmten Zeit abgeschlossen werden kann. Derartige zeitabhängige Schalt-
- 15 vorgänge sind über die Automatentafel nicht ohne weiteres anreizbar, weil die Automatentafel nur die Berechnung der jeweiligen Folgezustände eines Objektes beschreibt als Reaktion auf bestimmte Eingabesymbole oder innere Zustände. Um dennoch derartige zeitabhängige Vorgänge realisieren zu
- 20 können, soll der Rechner bzw. das Rechnersystem oder eine besondere Hardware zum Berechnen zeitabhängiger Zustandsänderungen über Timer verfügen, die vom Automaten durch Ausgabe spezieller Ausgabesymbole individuell zu starten sind und deren Ablauf bewirkt, daß dem Automaten
- 25 entsprechende Eingabesymbole zugeführt werden. Diese Maßnahme macht es möglich, auch zeitabhängige Steuerungsvorgänge auszuführen, obgleich diese eigentlich im Programm der Automatentafel nicht vorgesehen sind.

- Die Erfindung wurde vorstehend anhand eines sehr einfachen
- 30 Ausführungsbeispiels der Eisenbahnsignaltechnik, nämlich dem Einzelumstellen einer Weiche, erläutert. Das Verfahren zum Erstellen der Änderungsfunktionen zum Abarbeiten von Fahrstraßenaufträgen geschieht in genau der gleichen Weise

- wie bei der Einzelumstellung einer Weiche, nur daß das Objekt, nämlich die jeweilige Fahrstraße, mehrere Einzelobjekte aufweist, die in der Beschreibung des Fahrstraßenobjektes hinterlegt sind und den Aufruf der jeweils zugehörigen Zustandsvariablen im Datenvektor bei der Behandlung der Fahrstraße ermöglichen. Für die Behandlung einer Fahrstraße gelten wiederum an das jeweilige Problem, nämlich die Fahrstraßenbehandlung angepaßte Sprachmittel der Fachsprache für das Stellwerksgeschehen.
- 10 Das relevante Sprachmittel "Zulassungsprüfung" für den Vorgang der Zulassungsprüfung im Rahmen der Fahrstraßenbildung würde als Transitionsbedingung auf Elementtypebene beinhalten: Für einen Pfad im Gleisplan von Start nach Ziel soll gelten: Alle Elemente der Fahrstraße
- 15 sollen frei und nicht gesperrt sein (oder eine oder mehrere andere Bedingungen erfüllen, die durch die Betriebsordnung vorgegeben sind).
- Für AllePfadObjekte (Objekt, Fahrweg, frei(Objekt) $\wedge$ nicht gesperrt(Objekt)).
- 20 Der Ausdruck "für AllePfadObjekte" bindet die Variable "Objekt" = Fahrwegelement an jedes Objekt auf einem konkreten ahrweg. Er wird als Quantor bezeichnet, weil er Bedingungen (hier: frei $\wedge$ nicht gesperrt) an eine Gruppe von Objekten (Fahrwegelemente) bindet und damit diese
- 25 Bedingungen quantifiziert. Quantoren stellen nichts anderes dar als kompakte Schreibweisen für immer wiederkehrende Ausdruckweisen, die leichter zu überblicken sind als die unabgekürzte Form ihrer elementaren Zustandswerte; sie bilden die Sprachmittel der Fachsprache, wobei ihr
- 30 begifflicher Inhalt vorgegeben ist durch das zu beschreibende Prozeßgeschehen. Die expandierte Transitionenbedingung der Zulassungsprüfung einer konkreten

Fahrstraße zwischen einem Startelement S21 und einem Zielelement S22 über die Fahrweegelemente WO4 und WO5 würde lauten:

frei(S21) $\wedge$ nicht gesperrt(S21) $\wedge$ frei(WO4) $\wedge$ nicht
 5 gesperrt(WO4) $\wedge$ frei(WO5) $\wedge$ nicht gesperrt(WO5) $\wedge$ frei(S22) $\wedge$
 nicht gesperrt(WO5)** Folgezustand.

Die an der Fahrstraßenbildung konkret zu beteiligenden, über Quantoren anzusprechenden Fahrweegelemente S21, WO4, WO5, S22 ergeben sich aus der Anforderung einer ganz
 10 bestimmten Fahrstraße aus der Vielzahl der möglichen Fahrstraßen.

Es gibt auch Quantoren für die Bindung von Bedingungen an ganz bestimmte Objekte einer Vielzahl von Objekten, z. B. bei der Flankenschutzsuche. Dort soll das letzte Pfadobjekt
 15 auf einem Flankenschutzpfad Flankenschutz bieten. Der Quantor "Flankenschutz" beinhaltet

für LetztesPfadObjekt(Objekt, fls_Pfad, fls_bieten(Objekt))

Der Quantor bindet hier die Variable Objekt, für das die
 20 Bedingung fls_bieten(Objekt) = Flankenschutz bieten gilt, an das letzte Objekt auf dem Flankenschutzpfad fls_Pfad.

Das Expandieren und Codieren der konkreten Transitionsbedingungen beschiebt wieder automatisch im
 25 Rechner, der die Automatentabelle erstellt, durch Einsetzen der jeweils relevanten elementaren Zustandswerte in die durch die Quantoren und die zugehörigen Eingabesymbole (Anfordern einer ganz bestimmten Fahrstraße) bezeichneten Transitionsbedingungen. Die zunächst verwendeten
 30 Sprachmittel sind damit ersetzt durch eine Folge von Zustanswerten, die vorhanden sein müssen, damit ein Folgezustand einer Fahrstraße erreicht werden kann. Hieraus

werden Boole'sche Gleichungen gebildet und die sich daraus ergebenden Zustandsänderungsfunktionen werden nach Umsetzung in BDD's in der Automatentafel abgelegt.

Die Erfindung ist, obgleich beschrieben anhand von Ausführungsbeispielen der Eisenbahnsignaltechnik, mit Vorteil
5 auch bei der Steuerung anderer vorzugsweise komplexer Prozeßgeschehen zu verwenden, beispielsweise zur Steuerung von Energieversorgungsanlagen oder zum Betrieb von Verkehrssleitsystemen. Dabei kommt es nicht darauf an, ob das
10 Prozeßgeschehen durch nur einen oder durch mehrere Rechner gesteuert wird, die gemeinsam Zugriff haben auf eine gemeinsame Automatentafel oder getrennt zugreifen auf entsprechend ihrem Prozeßgeschehen angepaßte verteilte
Automatentafelteile oder ob die steuernden Rechner einfach,
15 redundant oder signaltechnisch sicher und ggf. redundant ausgeführt sind.

Ein ganz entscheidender Vorteil der Erfindung ist darin zu sehen, daß der Sicherheitsnachweis für die Richtigkeit der in der Automatentafel hinterlegten Zustandsübergangsfunktionen im wesentlichen nur in der Bewahrheitung der logischen Verknüpfung elementarer Zustandswerte in den Sprachmitteln der gewählten Fachsprache besteht. Wenn diese Bedingung erfüllt ist und die Projektierungsdaten fehlerfrei sind, ist auch die erstellte Automatentafel mit sehr großer
20 Wahrscheinlichkeit fehlerfrei, weil die Umsetzung in die elementaren Zustandswerte der einzelnen Objekte der Steuerungsanlage und die Bestimmung der Änderungsfunktionen durch einen Rechner unabhängig von der Sorgfalt eines Menschen geschieht. Bei etwaigen Änderungen der Außenanlage
25 bedarf es keines neuen Sicherheitsnachweises, weil sich an den dem Sicherheitsnachweis zu unterziehenden Gesetzmäßigkeiten nichts geändert hat. Es muß lediglich eine neue
30 Automatentafel erstellt werden, in der das erfindungsgemäße

Verfahren unter Verwendung geänderter Projektierungsdaten nochmals angewendet wird.

Unterschiedliche Anwender, z. B. verschiedene Bahnverwaltungen, stellen ggf. unterschiedliche Anforderungen an die Prozeßsteuerung. Das führt zu unterschiedlichen Sprachmitteln für unterschiedliche Anwender oder aber dazu, daß für unterschiedliche Anwender diese Sprachmittel zwar gleich sind, aber unterschiedliche logische Gesetzmäßigkeiten beinhalten. Sind Sicherheitsnachweise zu erbringen, so sind sie auf alle Sprachmittel der verwendeten Fachsprache anzuwenden.

Die in den ausführenden Rechnern zur Anwendung kommenden Automatentafeln sind verschieden in Abhängigkeit von der jeweils zur Anwendung kommenden Betriebsordnung und den Projektierungsdaten der jeweils zu steuernden Anlage; sie sind prinzipiell unabhängig von der hardware der verwendeten Rechner, können aber zurecht als Optimierung an diese angepaßt werden. Das Automatenprogramm zur Abarbeitung der Automatentafel ist von der zur Ausführung verwendeten hardware abhängig, aber für alle Anwender und alle Anwendungen gleich. Es muß nur dann geändert werden, wenn sich die Technologie der ausführenden Rechner ändert, weil z. B. andere Prozessoren zur Realisierung des Schaltwerkes verwendet werden.

Patentansprüche

1. Verfahren zum Erstellen der anwendungsabhängigen, hard- und software-unabhängigen Logik eines freiprogrammierbaren
- 5 Schaltwerkes zur Abarbeitung eines Prozeßgeschehens nach dem Prinzip des endlichen Automaten, insbesondere für große Steuerungssysteme, für die bislang aus Aufwandsgründen keine Automatentafel explizit aufgestellt werden konnte,
- g e k e n n z e i c h n e t d u r c h
- 10 die folgenden Verfahrensschritte
- a) Beschreiben aller möglichen (nicht veränderbaren) Merkmals- und (veränderbaren) Zustandswerte der physikalisch (gegenständlich) oder logisch (virtuell) existierenden Steuerungsobjekttypen des Steuerungssystems;
- 15 b) Beschreiben der konkreten Steuerungsanlage des Systems durch Benennen ihrer Steuerungsobjekte, durch Zuordnung der Objekte zu jeweils einem der beschriebenen Steuerungsobjekttypen und durch Benennen der durch Projektierungsangaben vorgegebenen konkreten Merkmalswerte
- 20 sowie Ermitteln des Zustandsraumes des Automaten aus der vorliegenden Beschreibung;
- c) Beschreiben des Prozeßgeschehens durch eine Vielzahl von Einzelvorgängen des Zustandsübergangs- und Ergebnisverhaltens auf Objekttypebene in einer anwendungsorientierten Fachsprache, deren Sprachmittel die
- 25 Bedingungen für das Erreichen eines jeweiligen Folgezustandes oder eines Ergebnisses als logische Verknüpfungen elementarer Zustandswerte und/oder Projektierungsdaten beinhalten;
- 30 d) Anwenden der den Objekttypen zugeordneten Beschreibungen gemäß c) auf die Objekte der Steuerungsanlage durch Ersetzen der auf die Objekttypen bezogenen Sprachmittel durch entsprechende objektbezogene Sprachmittel und durch Ersetzen der logischen Verknüpfungs-Merkmale der

objektbezogenen Sprachmittel durch die tatsächlichen
Zustandswerte und die Projektierungsdaten;

- 5 e) Aufstellen eines logischen Gleichungssystems für das
Übergangs- und Ergebnisverhaltens der einzelnen Objekte
und Darstellen dieser Gleichungen in einer kompakten
Form;
- 10 f) Ermitteln der anwendungsabhängigen Automatentafel in
Form der elementaren Zustandsübergangs- und Ergeb-
nisfunktionen aller Steuerungsobjekte aus den jeweiligen
Eingabesymbolen, den gegenwärtigen und den
Folgezustandswerten bzw. Ergebnissen der Objekte in der
gewählten kompakten Darstellungsform.

2. Verfahren nach Anspruch 1,

- 15 g e k e n n z e i c h n e t d u r c h die folgenden
Verfahrensschritte

- a) Beschreiben der möglichen Zustandswerte eines Steuerungs-
objekttyps als eine Aufzählung von Alternativen, von
denen zu jedem Zeitpunkt für jedes Objekt dieses Typs
20 genau eine zutrifft;
- b) Beschreiben der Gesetzmäßigkeiten für die Zustandswechsel
eines Steuerungsobjekttyps und für das Ergebnisverhalten
durch eine Aufzählung von Regeln, in denen jeweils fest-
gelegt ist, unter welchen Bedingungen (Meldungen, Ein-
25 gaben und definierten Zustandswerten) konkrete
Steuerungsobjekte dieses Typs ihren Zustand wechseln und
welche Steuerinformationen in diesem Fall dem Prozeß
zuzuführen sind.

30 3. Verfahren nach Anspruch 1 oder 2,

d a d u r c h g e k e n n z e i c h n e t ,
daß erste logische Gesetzmäßigkeiten für das
Zustandsübergangs- und Ergebnisverhalten der Objekttypen
eingabe- und meldungsorientiert formuliert werden und daß

weitere logische Gesetzmäßigkeiten für das Zustandsübergangs- und Ergebnisverhalten der Objekttypen formuliert werden, deren Ausführung allein abhängig ist vom jeweils erreichten internen Zustand der Steuerungsanlage
5 als Ergebnis der Ausführung einer vorangegangenen Zustandsübergangsfunktion.

4. Verfahren nach Anspruch 1,
d a d u r c h g e k e n n z e i c h n e t ,
10 daß als kompakte Darstellungsform der logischen Gleichungen die Darstellung in Binary Decision Diagrams gewählt ist.

5. Verfahren nach Anspruch 1 oder 4,
d a d u r c h g e k e n n z e i c h n e t ,
15 daß die Automatentafel aus der für die Generierung gewählten logischen Darstellungsform in eine für die Abarbeitung der Automatentafel in einen Rechner oder Rechnersystem optimierten Form umgesetzt wird.

20 6. Einrichtung zur Durchführung des Verfahrens nach einem der Ansprüche 1 bis 5,
g e k e n n z e i c h n e t d u r c h die Anwendung eines Rechners oder Rechnersystems, welcher die elementaren Zustandsübergangs- und Ergebnisfunktionen sämtlicher Steuerungsobjekte der Steuerungsanlage, also die Automatentafel,
25 durch Anwendung der definierten logischen Gesetzmäßigkeiten des Zustandsübergangs- und Ergebnisverhaltens der Steuerungsobjekttypen auf die Steuerungsobjekte der konkreten Steuerungsanlage ermittelt.

30

7. Einrichtung nach Anspruch 6,
d a d u r c h g e k e n n z e i c h n e t ,
daß der/die Rechner für den Fall, daß das Gleichungssystem für die Bestimmung der elementaren Zustandsübergangs- und

Ergebnisfunktionen (fehlerbedingt) sich widersprechende Bedingungen beinhaltet und daher nicht lösbar ist, die weitere Bearbeitung abbricht.

- 5 8. Einrichtung nach Anspruch 7,
dadurch gekennzeichnet,
daß der/die Rechner die Bedingungen, die zum Abbruch der
Ermittlung der elementaren Zustandsübergangs- und Ergebnis-
funktionen geführt haben, zur Fehleranalyse speichert.
- 10 9. Einrichtung zum Betrieb eines Steuerungssystems nach dem
Prinzip des endlichen Automaten,
gekennzeichnet durch die Verwendung
einer nach den Verfahrensschritten des Anspruches 1, 2, 3,
15 4 oder 5 unter Verwendung einer Einrichtung nach Anspruch 6
erstellten Automatentafel zur Abarbeitung in einem Rechner
oder einem Rechnersystem.
10. Einrichtung nach Anspruch 9,
20 gekennzeichnet durch folgendes:
a) Auf einem Rechner oder Rechnersystem läuft ein an seine
spezielle Hardware angepaßtes Automatenprogramm, das mit
den in der Automatentafel abgelegten Zustandsübergangs-
und Ergebnisfunktionen arbeitet oder dem Rechner/Rech-
25 nersystem wird die Automatentafel in kodierter Form über
einen Compiler in für ihn abarbeitbarer Form zugeführt.
b) Das Automatenprogramm erhält aus dem zu steuernden Prozeß
Zustandsmeldungen der Steuerungsobjekte sowie Eingaben
als Eingabesymbole und veranlaßt Ausgaben an die Steue-
30 rungsobjekte sowie Meldungen als Ausgabesymbole.
11. Einrichtung nach Anspruch 9 oder 10,
dadurch gekennzeichnet,

daß der Rechner/das Rechnersystem über Speicher zum Ablegen der Automatentafel und zum Verwalten der jeweils gültigen Zustandswerte für alle physikalisch und logisch existierenden Steuerungsobjekte verfügt.

5

12. Einrichtung nach Anspruch 9 oder 10,
d a d u r c h g e k e n n z e i c h n e t ,
daß im Rechner/Rechnersystem die von den einzelnen Objekten tatsächlich eingenommenen Zustandswerte in Form eines
10 Datenvektors festgehalten sind und daß der Rechner/das Rechnersystem nach Berechnen einer Zustandsänderung den oder die sich nach Maßgabe der jeweils zur Anwendung kommenden Zustandsänderungsfunktion ändernden Zustandswerte im Datenvektor mit dem oder den neuen Werten überschreibt.

15

13. Einrichtung nach Anspruch 12,
d a d u r c h g e k e n n z e i c h n e t ,
daß für jeden einem Objekt zugeordneten Folgezustandswert, der verschieden sein kann vom jeweiligen Ausgangszustands-
20 wert, eine eigene Zustandsübergangsfunktion zum Auffinden des jeweiligen Folgezustandes hinterlegt ist und daß das eine Zustandsänderung veranlassende Eingabesymbol mindestens mittelbar auf das oder die jeweils zu verändernden Zustände eines Steuerungsobjektes und die in Frage kommenden Zustandsänderungsfunktion hinweist.
25

14. Einrichtung nach Anspruch 13,
d a d u r c h g e k e n n z e i c h n e t ,
daß in der Automatentafel neben den Zustandsübergangsfunk-
30 tionen, auf die Eingabesymbole hinweisen, weitere Zustandsübergangs- und Eingabefunktionen hinterlegt sind, deren Behandlung allein vom Erreichen eines bestimmten internen Zustandes des Steuerungsobjektes als Ergebnis der Ausführung einer vorangegangenen Zustandsänderungsfunktion

abhängen und daß diese Funktionen so lange zur Anwendung kommen bis sich keine neuer Folgezustand bzw. keine Ausgabe des Gesamtsystems einstellt.

- 5 15. Einrichtung nach Anspruch 9,
d a d u r c h g e k e n n z e i c h n e t ,
daß der Rechner/das Rechnersystem oder eine gesonderte
Hardware zum Berechnen zeitabhängiger Zustandsänderungen
über Timer verfügt, die vom Automaten durch Ausgabe
10 spezieller Ausgabesymbole individuell zu starten sind und
deren Ablauf bewirkt, daß dem Automaten entsprechende
Eingabesymbole zugeführt werden.

1/2

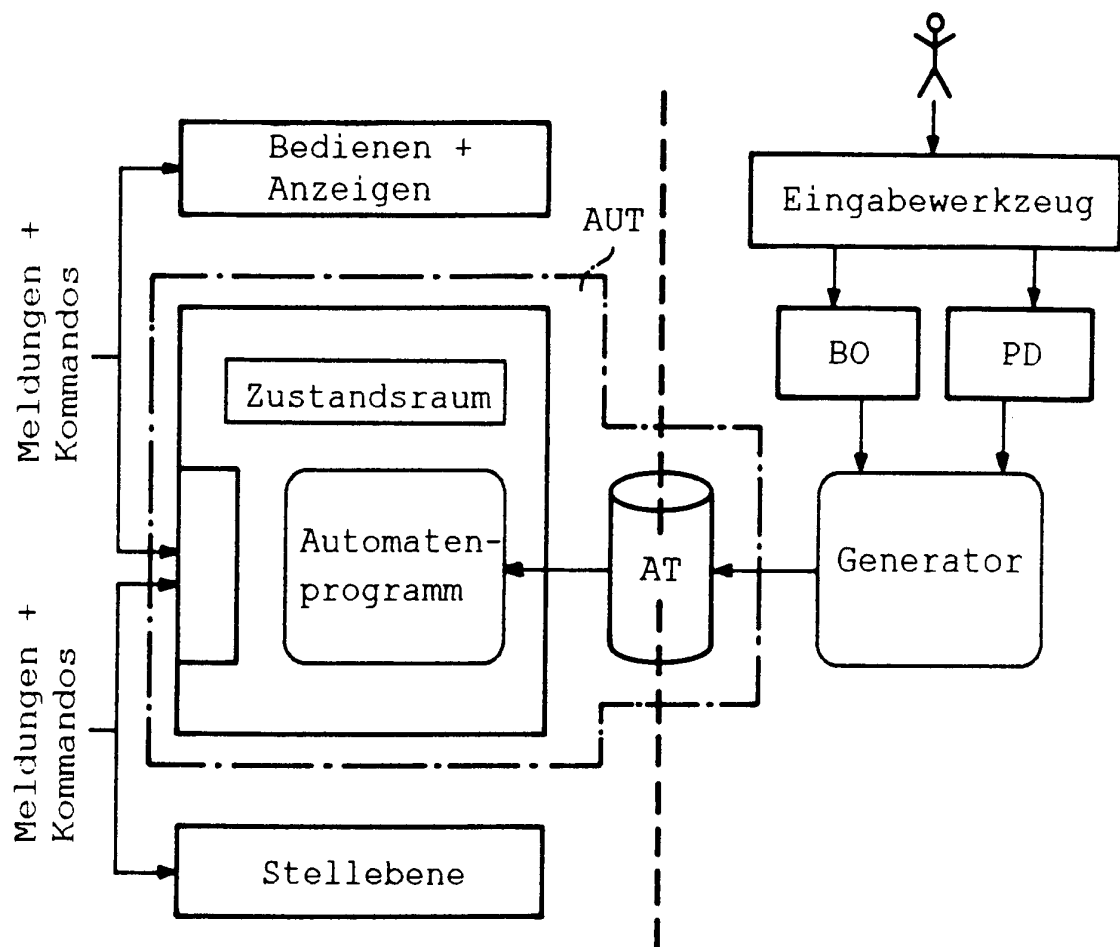


FIG 1

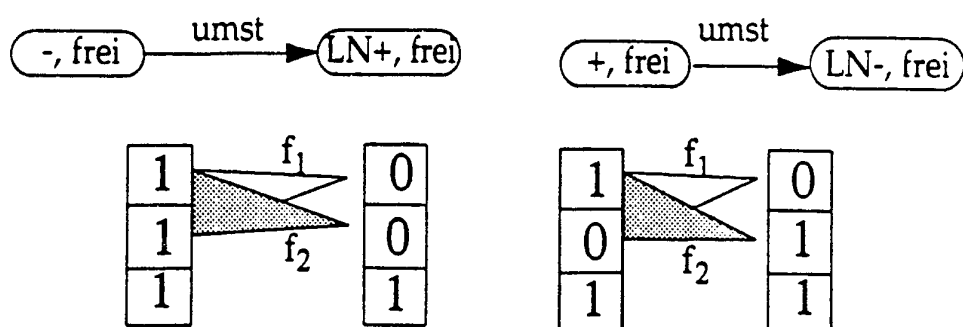


FIG 3

2/2

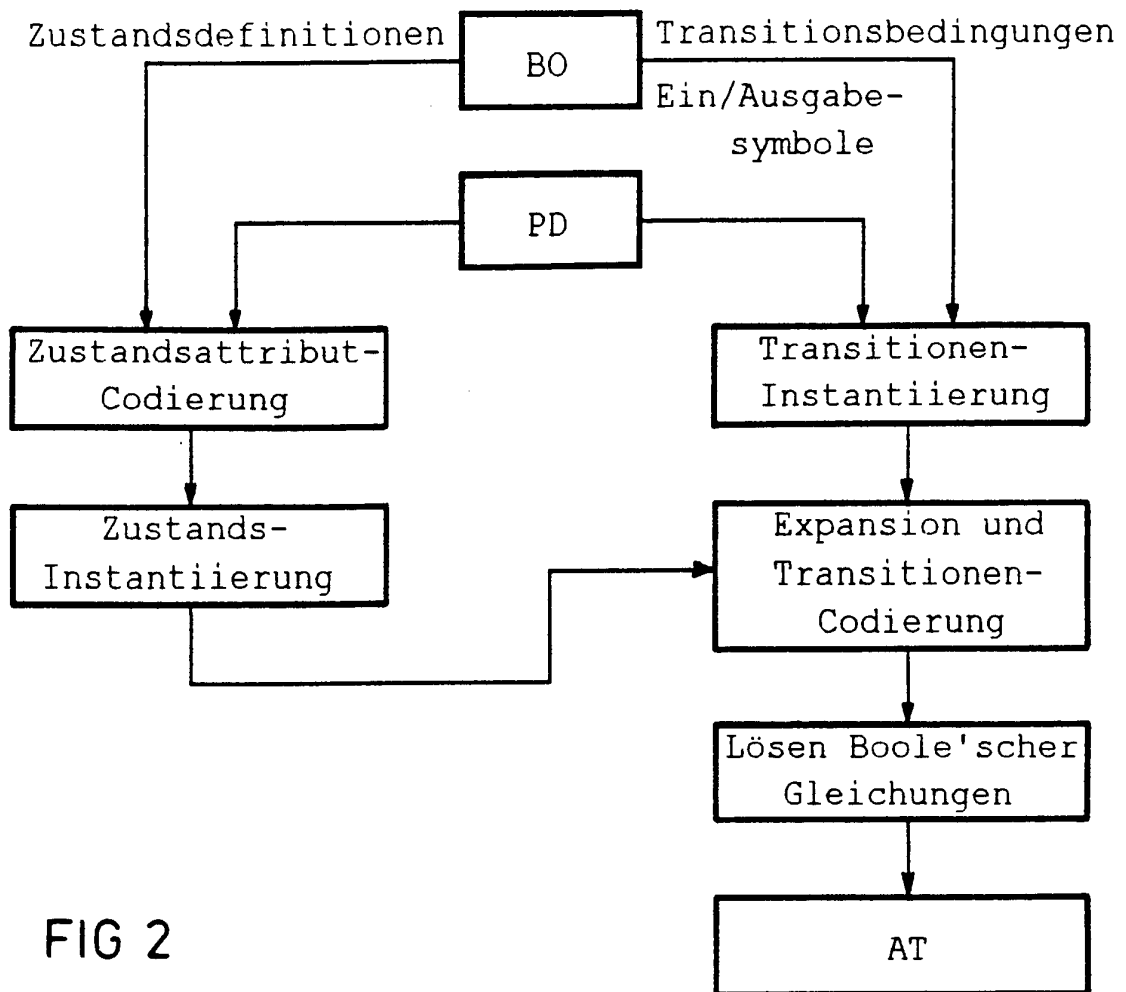


FIG 2

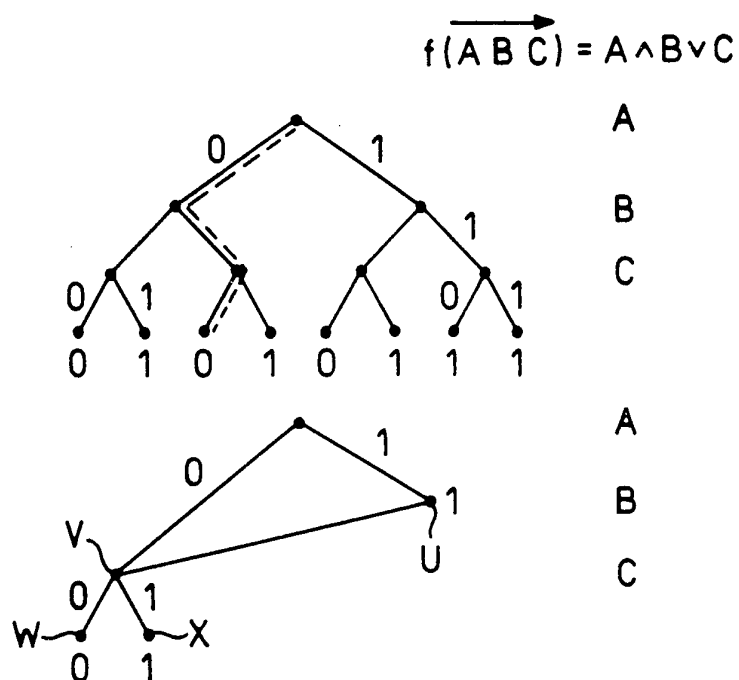


FIG 4

INTERNATIONAL SEARCH REPORT

International Application No

PCT/DE 93/01023

A. CLASSIFICATION OF SUBJECT MATTER
IPC 5 G05B19/04

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 5 G05B

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	ATP-SOFTWAREPRAXIS vol. 32, no. 9 , September 1990 , MÜNCHEN, DE pages 455 - 459 MICHAEL LASKOWSKI & HEINZ UNBEHAUEN 'Problembezogene Synthese von sequentiellen Steuerungsprogrammen' see paragraph 2 -paragraph 3.1A ---	1,9
A	ATP AUTOMATISIERUNGSTECHNISCHE PRAXIS vol. 32, no. 10 , October 1990 , MÜNCHEN, DE pages 509 - 514 MICHAEL LASKOWSKI & HEINZ UNBEHAUEN 'Problembezogene Synthese von sequentiellen Steuerungsprogrammen' see the whole document --- -/--	1,9

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents :

- "A" document defining the general state of the art which is not considered to be of particular relevance
- "E" earlier document but published on or after the international filing date
- "I" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- "O" document referring to an oral disclosure, use, exhibition or other means
- "P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

4 January 1994

Date of mailing of the international search report

1 1. 02. 94

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+ 31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+ 31-70) 340-3016

Authorized officer

Salvador, D

INTERNATIONAL SEARCH REPORT

Int. Patent Application No
PCT/DE 93/01023

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP,A,0 435 188 (SUMITOMO CHEMICAL COMPANY) 3 July 1991 see abstract see column 2, line 1 - line 41 see column 5, line 35 - line 57 see column 7, line 26 - line 12 see claim 6; figures 4,9 ---	1,9
A	VDI-ZEITSCHRIFT vol. 133, no. 12 , December 1991 , DÜSSELDORF, DE pages 60, 63-65 JÜRGEN SCHMIDT & HANS-JOACHIM SCHELBERG 'Objektorientierte Projektierung von Steuerungssoftware' paragraph " TOPAS " -----	1,9

Information on patent family members

PCT/DE 93/01023

Form PCT/ISA, 210 (patent family annex) (July 1992)

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
IPK 5 G05B19/04

Nach der Internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

B. RECHERCHIERTE GEBIETE

Recherchierter Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)
IPK 5 G05B

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie ^o	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	ATP-SOFTWAREPRAXIS Bd. 32, Nr. 9, September 1990, MÜNCHEN, DE Seiten 455 - 459 MICHAEL LASKOWSKI & HEINZ UNBEHAUEN 'Problembezogene Synthese von sequentiellen Steuerungsprogrammen' siehe Absatz 2 - Absatz 3.1A ---	1,9
A	ATP AUTOMATISIERUNGSTECHNISCHE PRAXIS Bd. 32, Nr. 10, Oktober 1990, MÜNCHEN, DE Seiten 509 - 514 MICHAEL LASKOWSKI & HEINZ UNBEHAUEN 'Problembezogene Synthese von sequentiellen Steuerungsprogrammen' siehe das ganze Dokument --- -/--	1,9

☒ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

☒ Siehe Anhang Patentfamilie

* Besondere Kategorien von angegebenen Veröffentlichungen :

A Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist

E älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist

L Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)

O Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht

P Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

T Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist

X Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden

Y Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist

Z Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

4. Januar 1994

Absenddatum des internationalen Recherchenberichts

11.02.94

Name und Postanschrift der Internationalen Recherchenbehörde
Europäisches Patentamt, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+ 31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+ 31-70) 340-3016

Bevollmächtigter Bediensteter

Salvador, D

C.(Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN		
Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	EP,A,0 435 188 (SUMITOMO CHEMICAL COMPANY) 3. Juli 1991 siehe Zusammenfassung siehe Spalte 2, Zeile 1 - Zeile 41 siehe Spalte 5, Zeile 35 - Zeile 57 siehe Spalte 7, Zeile 26 - Zeile 12 siehe Anspruch 6; Abbildungen 4,9 ---	1,9
A	VDI-ZEITSCHRIFT Bd. 133, Nr. 12 , Dezember 1991 , DÜSSELDORF, DE Seiten 60, 63-65 JÜRGEN SCHMIDT & HANS-JOACHIM SCHELBERG 'Objektorientierte Projektierung von Steuerungssoftware' Absatz " TOPAS" -----	1,9

Angaben zu Veröffentlichungen, die zur selben Patentfamilie gehören

PCT/DE 93/01023

Formblatt PCT/ISA/210 (Anhang Patentfamilie)(Juli 1992)