

tanulmányok 74/1978

MTA Számítástechnikai és Automatizálási Kutató Intézet Budapest



MAGYAR TUDOMÁNYOS AKADÉMIA
SZÁMITÁSTECHNIKAI ÉS AUTOMATIZÁLÁSI KUTATÓ INTÉZETE

VORTRÄGE
ÜBER DAS GRAPHISCHE DISPLAY GD`71

Tanulmányok 74/1978

Felelős kiadó:
Dr. Vámos Tibor

ISBN 963 311 057 2
ISSN 0324-2951

Die Vorträge wurden in der DDR anlässlich Ausstellungen
in Verbindung mit dem GD`71 abgehalten.

I N H A L T

Dipl.-Ing. Wolfgang FRANKE ZUR KOPPLUNG GD`71 - KRS 4201.....	1
Dr.rer.nat. R. ORTLEB SOFTWARE ZUM BETRIEB DES GRAFISCHEN BILDSCHIRM- GERÄTES GD`71.....	13
W. FRANKE, M. LUDWIG, D. MONJAU, R. ORTLEB GERÄTE ^Ä TECHNISCHE VORAUSSETZUNGEN UND PROGRAM- MIERUNGSMETHODEN FÜR EINEN DIGITALGEOMETRISCHEN ARBEITSPLATZ.....	16
Dipl.-Math. G. KRAMMER METHODEN DER ANWENDUNG DER INTERAKTIVEN DIGITAL- GRAPHIK IM RECHNERGESTÜTZTEN KONSTRUIEREN.....	31
T. TOLNAY-KNEFÉLY, D. KERESTÉLY EIN GRAPHISCHES PROGRAMM MIT FRAGE- UND ANTWORT- VERFAHREN FÜR DIE MODIFIKATION UND KORREKTUR DER LEITERABBILDUNG GEDRUCKTER LEITERPLATTEN.....	46
Dipl.-Ing. G. DÉRI INTERAKTIVE KONSTRUKTION MIT RECHENTECHNIK IM MASCHINENBAU.....	63
Dipl.-Ing. P. VEREBÉLY INTERAKTIVE KONSTRUKTION MIT RECHENTECHNIK IM MASCHINENBAU.....	82

Dipl.-Ing. Wolfgang FRANKE
Technische Universität Dresden
Sektion Mathematik,
Wissenschaftsbereich Mathematische
Kybernetik und Rechentechnik

ZUR KOPPLUNG GD`71 - KRS 4201

Vorspann

Es wird eine Übersicht über Aufbau und Wirkungsweise einer Gerätesteuereinheit /GSE/ gegeben, die den Anschluss des ungarischen Bildschirmgerätes GD`71 an den DDR-Kleinrechner ROBOTRON 4201 ermöglicht. Dabei werden die für einen Datenaustausch zwischen Display und Rechner notwendigen Befehls- und Signalfolgen beschrieben. Der Datenaustausch erfolgt zwischen Rechner und GSE über das rechner-spezifische Interface /Programmierter Kanal und Externer Speicherkanal/ und zwischen GSE und Display über ein Spezialinterface.

1. Einführung

Im folgenden Beitrag wird eine Übersicht über die hardwaremässige Kopplung des ungarischen grafischen Displays GD`71 mit dem DDR-Rechner ROBOTRON 4201 gegeben. Die Kopplung beider Einheiten ist Resultat einer engen Zusammenarbeit zwischen dem Forschungsinstitut für Rechentechnik und Automatisierung der Ungarischen Akademie der Wissenschaften und der Technischen Universität Dresden, Sektion Mathematik, Wissenschaftsbereich Mathematische Kybernetik und Rechentechnik. Diese arbeitsteilige Zusammenarbeit umfasst die Modifizierung der displayeigenen Steuerung durch den ungarischen Partner und

die Entwicklung einer Gerätesteuereinheit /GSE/ durch Mitarbeiter der TU.

Die GSE hat folgende Aufgaben zu erfüllen:

- a/ Als Bindeglied zwischen Rechner und Display passt sie das GD-spezifische Anschlussbild an die Datenwege des Rechners an.
- b/ Sie übernimmt die Zwischenspeicherung von Daten. Das Fassungsvermögen des Datenregisters der GSE beträgt ein Rechnerwort /16 Bit/.

Die Verbindung zum Rechner folgt über den Programmierten Kanal /PK/, den Externen Speicherkanal /ESK/ und einen der 16 Zusatzunterbrechungskanäle /Bild 1/. Aufgrund der begrenzten Leitungslänge des Rechnerinterface macht sich eine Unterbringung der GSE im Gefäßsystem des KRS 4201 erforderlich.

Die Verbindung zum Display wird durch ein Spezialinterface hergestellt.

2. Informationsaustausch über den Programmierten Kanal

Der Informationsaustausch erfolgt hierbei zwischen dem Akkumulator der ZVE und einem der insgesamt 20 Register der displayinternen Steuerung. Er ermöglicht den Start bzw. den Stop der Bildwiederholung und das Setzen dieser Register in einen gewünschten Zustand /Ausgabevorgang/ bzw. das Auslesen der GD-Registerinhalte /Eingabevorgang/ auch bei laufender Bildwiederholung, sofern nicht gerade eine Übertragung über den ESK stattfindet.

Da die Übertragungszeit über den ESK wesentlich geringer ist /1,25..3 μ s/ als die Zeit, die die Generatoren für die Ausführung des übermittelten Befehls benötigen, verbleibt zwischen 2 ESK-Übertragungen genügend Spielraum für einen Informationsaustausch über den PK.

2.1. Ausgabevorgang

Die Signalfolge bei einem Ausgabevorgang ist in Bild 2 gezeigt. Der Verkehr zwischen ZVE und GSE beginnt stets mit einem Anwahlsteuerbefehl /Anwahl-ASI/. Dadurch wird die Bereitschaft der GSE zum Informationsaustausch mit der ZVE erzwungen. Dieser Anwahl-ASI braucht nur einmal am Beginn jeglichen Datenaustausches gegeben zu werden, d.h., er bleibt solange gültig, bis die GSE und damit auch das Display durch einen Rücksteuerbefehl /Rücksetz-ASI/ bzw. durch Generallöschen in den Grundzustand gesetzt werden.

Vor der eigentlichen Ausgabe muss die Auswahl /Adressierung/ eines der 20 GD-Register erfolgen. Die Adressinformation muss hierfür als Oktalzahl /zwischen 0 und 27/ im Akkumulator des Rechners bereitgestellt werden. Mit Hilfe des Adressausgabebefehls /Adress-AVA/ wird diese Information in das Adressregister der GSE übernommen, dessen Ausgänge mit einem Adressdecoder im Display verbunden sind.

Im Adress-AVA wird durch ein spezielles Bit kenntlich gemacht, dass ein Ausgabevorgang folgen soll.

Alle Ausgabebefehle /AVA-Befehle/ und auch alle Eingabebefehle /ENA-Befehle/ führen selbständig einen Test durch, ob die angesprochene GSE zu einem Datenaustausch mit der ZVE bereit ist. Eine solche Bereitschaft liegt stets dann vor, wenn nicht gerade ein Datenaustausch zwischen GSE und GD`71 erfolgt, bzw. ein vorangegangener Datenaustausch ordnungsgemäss abgeschlossen wurde. In diesem Fall wird der ENA- bzw. AVA-Befehl ausgeführt und der folgende Befehl automatisch übersprungen. Liegt keine Bereitschaft vor, so wird der dem Ein- bzw. Ausgabebefehl folgende Befehl abgearbeitet, der i.a. ein Sprungbefehl auf eben diesen ENA- bzw. AVA-Befehl ist.

Auf diese Art realisiert man eine Warteschleife, die erst mit dem Vorliegen der Bereitschaft der GSE verlassen werden kann.

Die Übernahme der Adresse in das Adressregister der GSE führt zur Abgabe eines Anmeldesignals für Ein- bzw. Ausgabe an das GD`71 /ANMEA/, wodurch für die Dauer des Anliegens dieses Signals Zugriffsforderungen zum ESK blockiert werden.

Anschliessend wird mit Hilfe eines Daten-Ausgabebefehls die im Akkumulator bereitgestellte Information in das Datenregister der GSE übernommen. Bei Ausführung des Daten-AVA wird automatisch der Bereitschaftstest, wie oben beschrieben, durchgeführt.

Mit der Zwischenspeicherung der Daten im Datenregister der GSE ist der Informationsaustausch zwischen ZVE und GSE beendet.

Die GSE steuert selbständig auf ähnliche Weise wie die Anschluss-steuereinheit 1 des R 4201 die Weiterleitung der Daten zum adressierten GD-Register.

Mit Hilfe einer Zählschaltung wird verzögert ein Signal RUFA an das GD`71 abgegeben, das als Taktsignal zur Übernahme der auf den Datenleitungen bereits anliegenden Daten in das ausgewählte GD-Register dient.

Als Quittungssignal läuft ENDA zur GSE zurück, wodurch, wiederum verzögert, RUFA abgeschaltet wird. Dadurch werden schliesslich das ENDA-Signal und auch das obenerwähnte Anmeldesignal für Ein-Ausgabe rückgesetzt, wodurch ein weiterer Ausgabe- oder ein Eingabevorgang bzw. eine Datenübertragung über den ESK eingeleitet werden darf.

2.2. Eingabevorgang

Das Signalspiel bei einem Eingabevorgang zeigt Bild 3. Am Anfang muss auch hier die Auswahl des GD-Registers stehen, dessen Inhalt in das A-Register des Rechners übernommen werden soll. Dies erfolgt analog zum Ausgabevorgang durch einen Adress-AVA, der durch ein spezielles Bit auf einen nachfolgenden Eingabevorgang hinweist. Bereitschaft der GSE führt zum Übernahme der Adresse in ihr Adress-register. Ausserdem blockiert das Anmeldesignal für Ein-Ausgabe /ANMEA/ eine Übertragungsforderung über den ESK.

Anschliessend wird die im vorigen Abschnitt bereits erwähnte Zählschaltung tätig, die die verzögerte Abgabe des RUFÉ-Signals steuert. Mit dem Erhalt von RUFÉ legt das adressierte GD-Register die Daten auf die Datenleitungen zur GSE und das Quittungssignal ENDE, wodurch die Daten in das Datenregister der GSE übernommen werden.

Schliesslich wird RUFÉ verzögert abgeschaltet und dadurch auch ENDE rückgesetzt.

Mit Hilfe eines Eingabebefehls /ENA-Befehl/ werden die zwischengespeicherten Daten in den Akkumulator übernommen. Dies führt zum Rücksetzen des Anmeldesignals für Ein-Ausgabe und damit zur Freigabe für weitere Übertragungsvorgänge.

3. Die Datenübertragung über den ESK

Zur Erzeugung "stehender" Bilder müssen die im Bildwiederholungsspeicher, einem Teil des Hauptspeichers /Display-File/, abgespeicherten Bildinformationen fortlaufend ausgelesen werden. Ausserdem werden bei Anwendung

der Unterprogramm-Technik gewisse Hilfsinformationen in den Kellerspeicher /Stack/, der ebenfalls im Hauptspeicher realisiert ist, eingeschrieben. Die Datenübertragung erfolgt hierbei unter Umgehung der ZVE des Rechners im direkten Speicherzugriff über die Leitungen des ESK. Sie wird stets vom Display durch Abgabe einer Zugriffsforderung ZUFORD initiiert unter der Bedingung, dass nicht gerade eine Übertragung über den PK stattfindet.

Mit ZUFORD gibt das Display auch die 14 Adress-Bits zur Auswahl der Hauptspeicheradresse und ein Steuersignal, das einen geforderten Lese- bzw. Schreibzyklus kennzeichnet, ab.

3.1. Lesezyklus

Die GSE leitet die Steuerinformationen zum richtigen Zeitpunkt zum Rechner weiter, der die Zugriffsforderung i.a. nach einer Wartezeit $\leq 10 \mu\text{s}$ akzeptiert. Eine rechner eigene Prioritäts- und Verteilersteuerung /PuV/ zeichnet für die getaktete Übernahme der Lesedaten in das Datenregister der GSE verantwortlich.

Die anschliessende Weiterleitung dieser Daten zum Display erfolgt in analoger Weise wie im Ausgabevorgang bei PK-Übertragung erläutert. Am Ende des Übertragungsvorganges schaltet das Display schliesslich die Steuer- und Adress-signale ab.

Das Signalspiel für Lese- und Schreibzyklus ist in Bild 4 gezeigt.

3.2. Schreibzyklus

Neben den Steuer- und Adressinformationen legt das Display auch die in die Hauptspeicherzelle einzuschreibenden Daten auf die Interfaceleitungen zur GSE. Nachdem die Zugriffsforderung vom Rechner durch Abgabe des Bestätigungssignals ESKMFF akzep-

tiert worden ist, erfolgt rechnergesteuert die Durchschaltung dieser Daten zum Hauptspeicher und das Eintragen in die Hauptspeicherzelle. Eine Zwischenspeicherung der Daten im Datenregister der GSE erfolgt nicht.

Der Schreibzyklus wird mit dem Abschalten des Bestätigungssignals für ESK-Übertragung beendet, was das Display veranlasst, alle noch anliegenden Signale rückzusetzen.

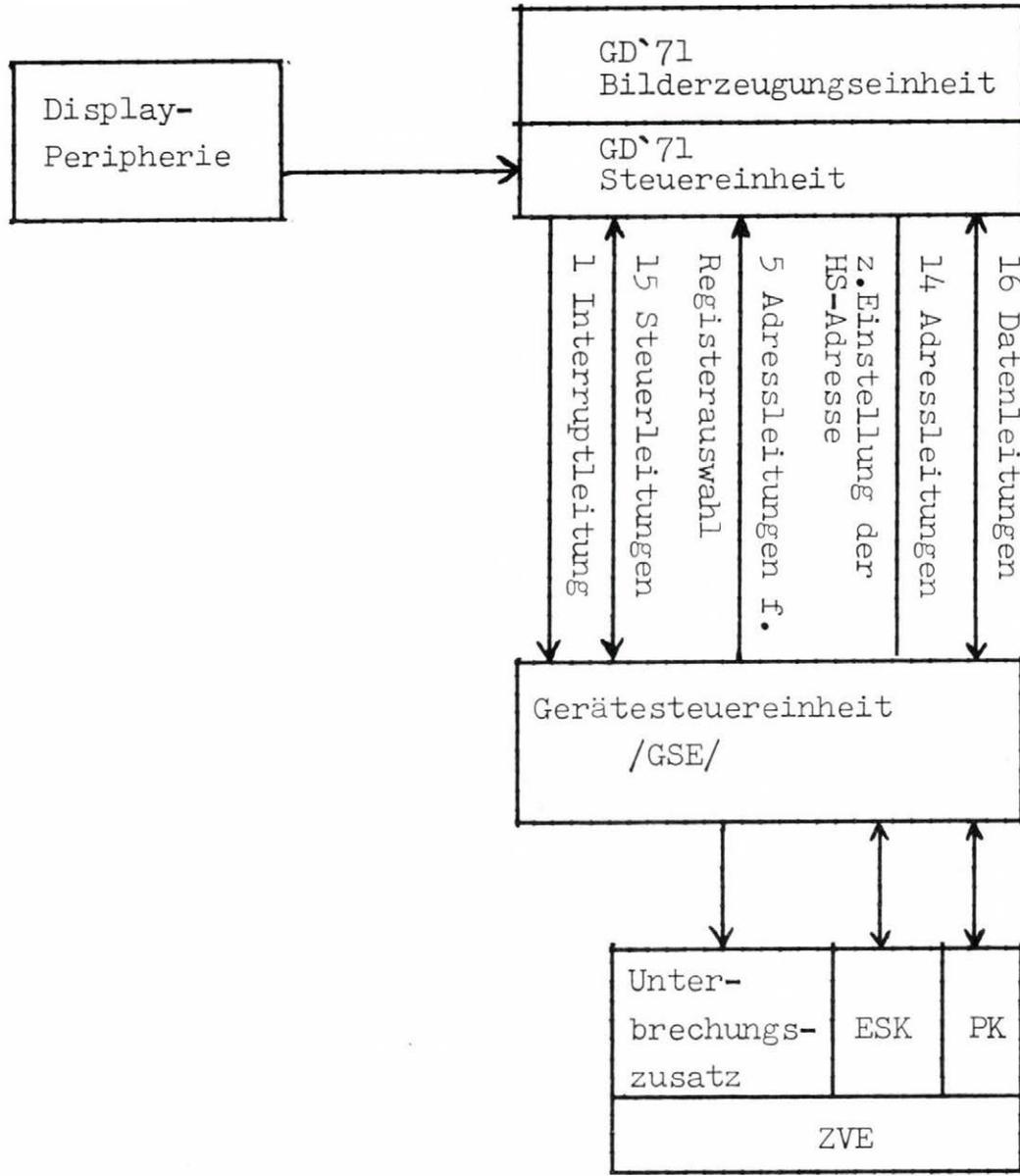
4. Unterbrechungskanal

Wie bereits in /5/ erläutert, hat der Nutzer die Möglichkeit, über die Displayperipherie einen Dialog mit der Bildschirmeinheit zu führen. Das Drücken einer Taste der Funktions- oder der alphanumerischen Tastatur bzw. das Aufsetzen des Lichtstiftes auf einen hellen Punkt des Bildschirms führt zur Abgabe eines Interrupt-Signals, falls die peripheren Einrichtungen vorher durch Programm aktiviert worden waren. Neben diesen programmierbaren Interrupts, denen unterschiedliche Prioritätsränge zugewiesen werden können, gibt die Displaysteuerung bei verschiedenen Fehlerzuständen auch Hardwareinterruptsignale ab, die stets vorrangig behandelt werden. Die prioritätshöchste Unterbrechungsursache wird im Interrupt-Register des GD⁷¹ eingetragen. Gleichzeitig wird durch Abgabe eines Unterbrechungssignals auf der Interruptleitung zur GSE der GSE-Interrupt-Flip-Flop gesetzt /INTFF/. Das Setzen dieses Flip-Flop führt zur Erregung des Zusatzunterbrechungskanals 1 des Rechners und zum Sperren des Display-Interruptregisters. Somit können später eintreffende Interruptursachen erst wirksam werden, nachdem das Interruptregister am Ende des Interruptbehandlungsprogramms durch Rücksetzen des INTFF wieder freigegeben wurde. Das Rücksetzen des INTFF erfolgt durch einen Steuerbefehl /ASI¹¹²⁴/. Die Erregung des Zusatzunterbrechungskanals 1 des KRS 4201 führt unter als bekannt vorausgesetzten Bedingungen zu einer Sprung-

unterbrechung. Es wird ein Interruptbehandlungsprogramm angesprochen, dessen Anfangsadresse in der dem Unterbrechungskanal zugeordneten Verbindungszelle abgespeichert sein muss. Im Rahmen dieses Programms wird die Unterbrechungsursache ermittelt. Dies erfolgt durch Einlesen des Interrupt-Registerinhalts gemäss Abschnitt 2.2.

LITERATURVERZEICHNIS

- 1./ Technische Beschreibung EC-8404; VEB Kombinat Robotron, 1976
- 2./ Kundendokumentation EC-8404; VEB Kombinat Robotron, 1975
- 3./ GD 71/T Hardware Manual; Institut für Rechentechnik und Automatisierung der Ungarischen Akademie der Wissenschaften
- 4./ GD 71/T Graphical Display and Input Implements. Hardware Programming Manual; Institut für Rechentechnik und Automatisierung der Ungarischen Akademie der Wissenschaften, Budapest, Mai 1974
- 5./ Franke, W., Ludwig, M., Monjau, D., Ortleb, R.:
Gerätetechnische Voraussetzungen und Programmierungsverfahren für einen digitalgeometrischen Arbeitsplatz;
Wiss. Z. d. TU Dresden



KRS 4201

Bild 1 Übersicht

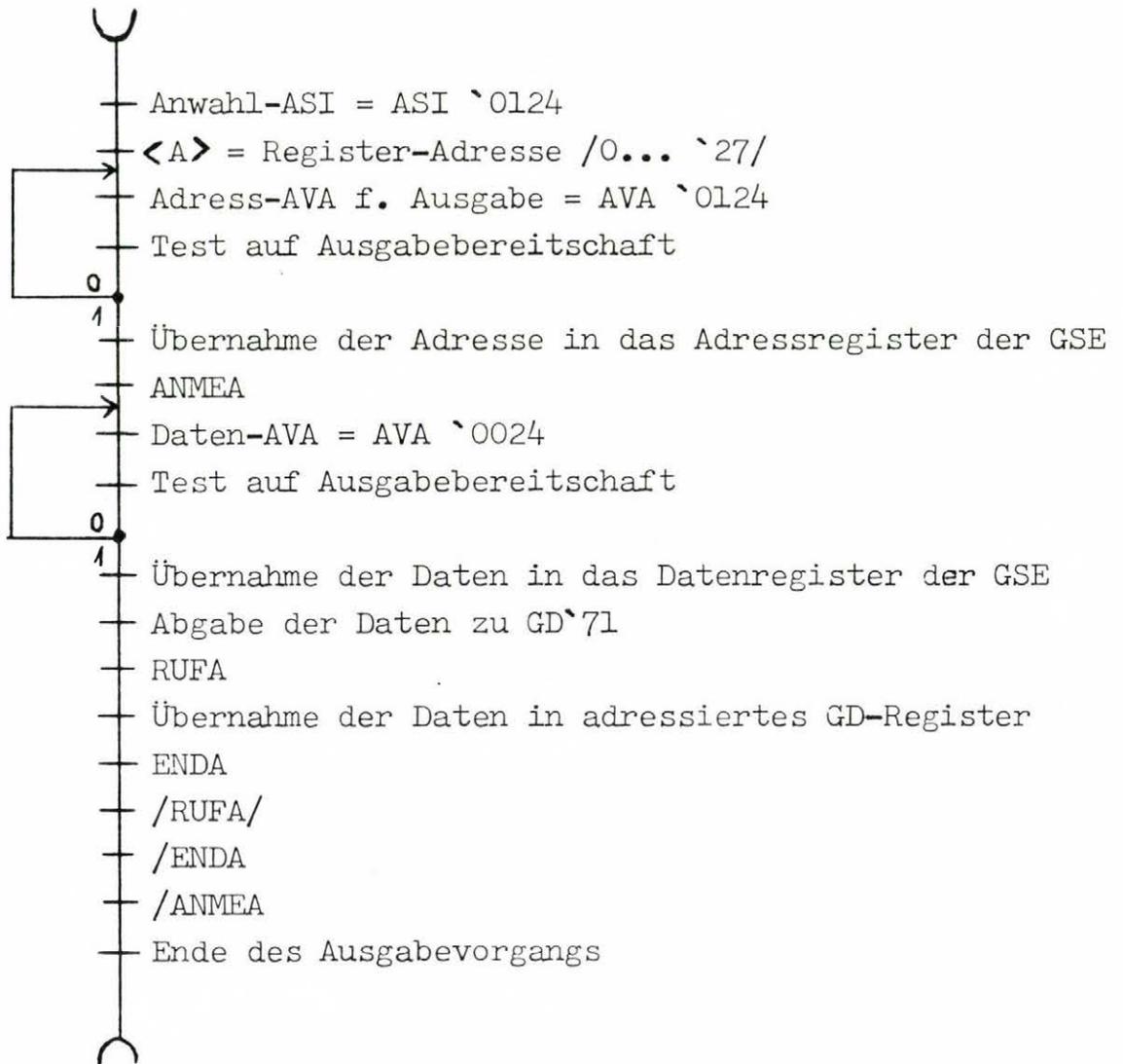


Bild 2 Signalfolge beim Ausgabevorgang

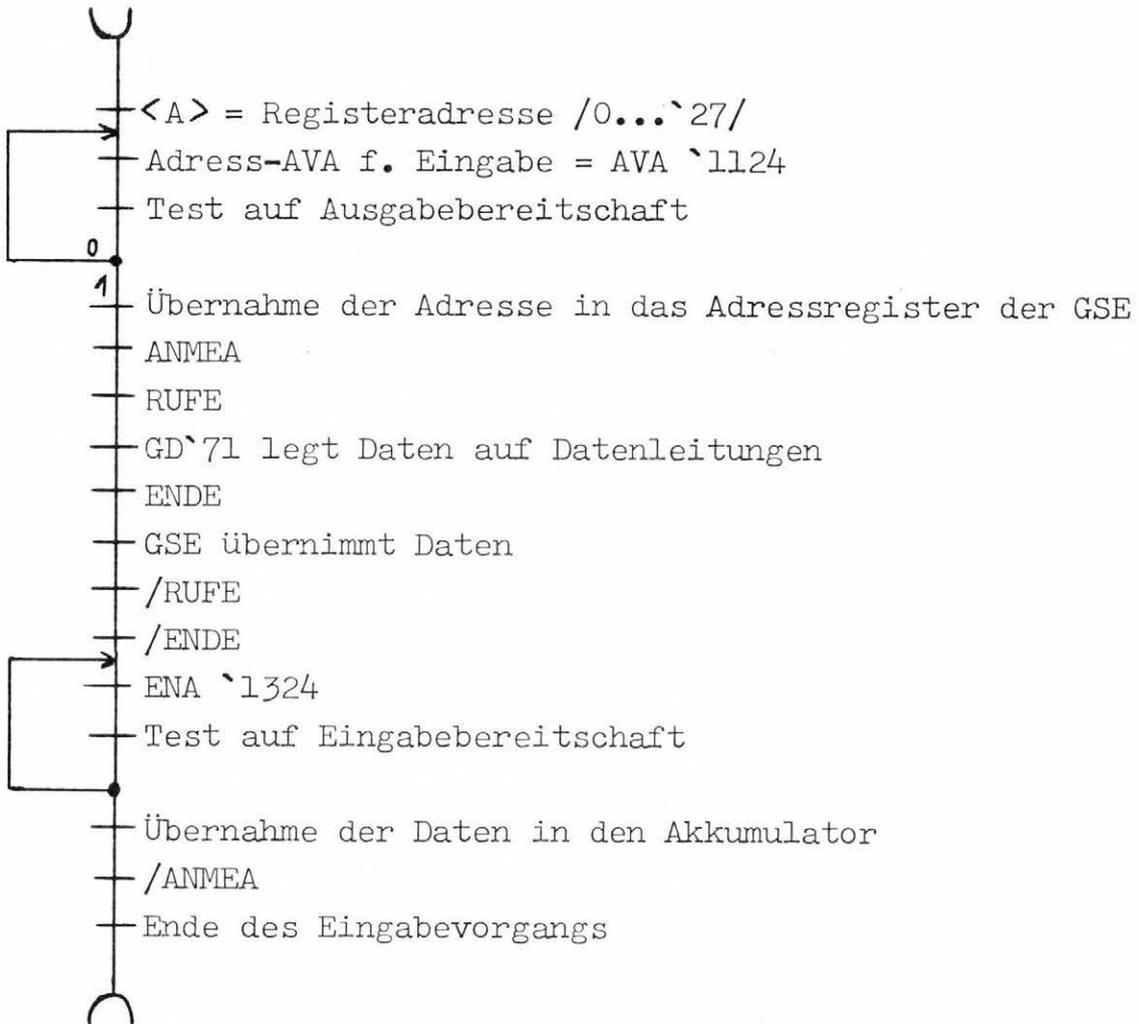


Bild 3 Signalfolge beim Eingabevorgang

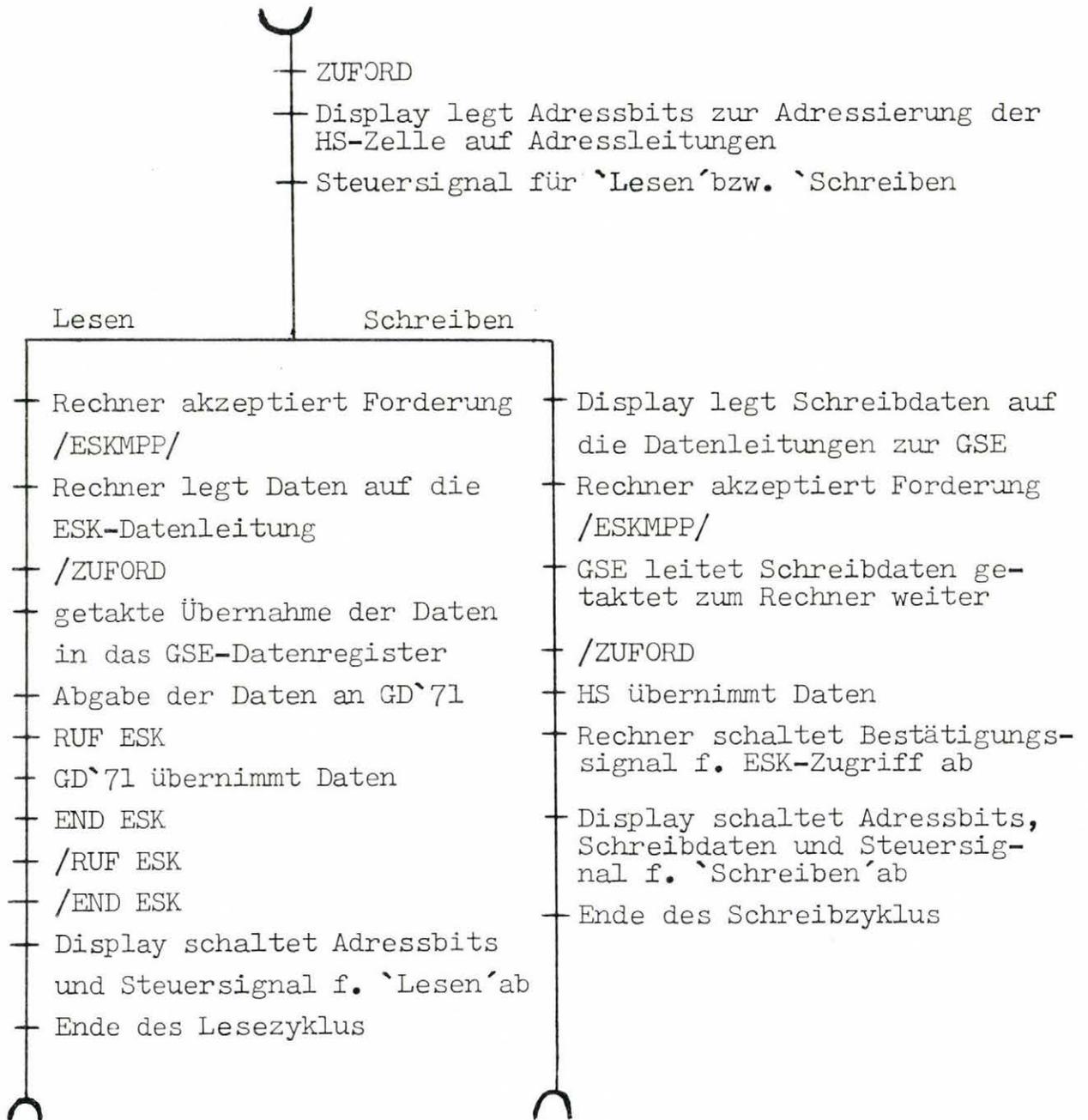


Bild 4 Signalfolge bei Datenübertragung über den ESK

Dr.rer.nat. R. ORTLEB
Technische Universität Dresden
Sektion Mathematik,
Wissenschaftsbereich Mathematische
Kybernetik und Rechentechnik

SOFTWARE ZUM BETRIEB DES GRAFISCHEN

BILDSCHIRMGERÄTES GD`71

Zur Gewährleistung und Stützung des Betriebes des speziellen Ein- und Ausgabegerätes aktiver grafischer Bildschirm notwendige Software wird hinsichtlich des Sprachgebrauchs in Basissoftware, Grundsoftware, problemorientierte Software und Anwendersoftware klassifiziert. Diese Begriffsbildungen werden nicht einheitlich angewendet, sind im allgemeinen nicht unabhängig von Zwängen der konkreten Implementation und bei der maschinennahen Software von den Parametern der Konfiguration beeinflusst. Der Frage nach dem Zweck der jeweiligen Softwareklasse folgend, ist zu unterscheiden in Software /also Systeme, Programme, Moduln, Makros usw./,

- 1./ die die von der Bildschirmsteuereinheit zyklisch lesbare, strukturierte Speicherbelegung /Display-file/ erzeugt und die zu dieser Generierung benötigten darstellungsgerechten Daten aus höherliegenden Programmen /möglicherweise des Hintergrundrechners/, aus internen oder externen Speicherbereichen oder von Datenträgern übernimmt /Basissoftware/.
- 2./ die die /nicht maschinenbezogenen/ darstellungsgerechten Datenstrukturen erzeugt und dazu die Beschreibung geometrischer Objekte als Eingangsdaten benutzt /Grundsoftware/.

- 3./ die die Beschreibung geometrischer Objekte, ihrer Beziehungen zueinander und ihrer Abbildungen gestattet /Geometrie-problemorientierte Software/.
- 4./ die die geometrischen Objekte als Objekte der Modellierung realer Zusammenhänge verwendet /Anwendersoftware/.

Bekannte rechentechnische Realisierungen wie GIPS /Institut für Schiffbau, Rostock/ und DIGRA /Institut für Schiffbau und Wilhelm-Pieck-Universität, Rostock/ lassen sich in /2/ und teilweise /1/ bzw. /3/ einordnen.

Die an der Technischen Universität Dresden betriebenen Entwicklungsarbeiten konzentrieren sich auf Stufe /1/ und Anpassungen zu Stufe /2/. Die hardwaremässige Kopplung zum Grossrechner /EC 1022/ ist in der Installationsphase, zugleich wird die softwaremässige Sicherung des Zugriffs vorbereitet. Durch angebahnte Kooperation wird von VEB Robotron ZFT, Institut für Schiffbau, WPU Rostock und TU Dresden angestrebt, die realisierten Implementierungen auf Stufe /2/ bzw. /3/ modifiziert zu nutzen.

Im gegenwärtigen Stadium ist ein autonomer Einsatz als Bildschirm-Kleinrechner-Konfiguration GD`71/KRS 4201 möglich. Die Programmierleistungen des Anwenders bestehen dabei in:

- der Programmierung seines Hintergrundprogrammes in Assemblersprache oder FORTRAN,
- der Planung des Dialogs durch Einbindung von Unterprogramm- bzw. Subroutinenrufen der Makros der grafischen Programmierung aktiv /GPA/,
- der problemorientierten Benutzung der Standard-Displayfilestruktur, die durch die bereitgestellten Makros der grafischen Programmierung in Standard-Struktur /GPS/ gegeben ist, und in

- der bildinhaltlich orientierten Anwendung der Makros der grafischen Programmierung passiv /GPP/, die grafische Elementfolgen, gegebenenfalls modifiziert bzgl. der Darstellungsart, koordinaten- und bildausschnittgerecht generiert.

Wolfgang Franke
Manfred Ludwig
Dieter Monjau
Rainer Ortleb
Technische Universität Dresden
Sektion Mathematik,
Wissenschaftsbereich Math.
Kybernetik und Rechentechnik

GERÄTETECHNISCHE VORAUSSETZUNGEN UND PROGRAM-
MIERUNGSMETHODEN FÜR EINEN DIGITALGEOMETRISCHEN ARBEITSPLATZ¹

Hauptkommunikationsebene Mensch-Maschine eines Digitalgeometrischen Arbeitsplatzes /DGA/ ist das Ein- und Ausgabe-medium Bildschirm. Im nachfolgenden Aufsatz werden unter besonderer Konzentration auf die Konfigurationselemente interaktives Bildschirmgerät und zugehöriger Steuerrechner geräte- und programmierungstechnische Spezifika solcher Systeme umrissen sowie ein Beispiel einer Anwendungskonzeption dargestellt.

¹ Die Entwicklung der speziellen Gerätetechnik für die Kopplung des ungarischen grafischen Displays GD`71 mit dem Rechner 4201 erfolgte in sozialistischer Gemeinschaftsarbeit zwischen der Ungarischen Akademie der Wissenschaften, Institut für Rechentechnik und Automatisierung, Budapest, und der Technischen Universität Dresden, Sektion Mathematik, Wissenschaftsbereich Math. Kybernetik und Rechentechnik, mit Unterstützung durch den VEB Robotron ZFT Dresden.

Die Nutzung von Methoden und Verfahren der Konstruktionswissenschaften für die Praxis erfordert in immer stärkerem Masse den Einsatz geeigneter rechentechnischer Geräte und Programmsysteme. Eine Analyse und Abstraktion der dabei auftretenden Probleme führt nicht selten zu Fragen der konstruktiven Geometrie, der Verwaltung grosser Datenbestände, der Bereitstellung problemangepasster Programmiersprachen, des Mensch-Maschine-Dialogs und der Nutzung dafür geeigneter rechentechnischer Anlagen.

1. Gerätetechnische Konfiguration und Arbeitsweise

Die wesentlichsten Bestandteile des ungarischen grafischen Displays GD`71 sind der Bildschirm, der Bilderzeugungsteil /Generatoren und Ablenkverstärker/, die Displaysteuereinheit und die peripheren Einrichtungen Lichtstift, Positionierkugel, Funktions- und alphanumerische Tastatur /1/, /2/, /Bild 1/.

Der Rechner 4201 dient als Steuerrechner für das grafische Display. Er enthält im Hauptspeicher die Beschreibung der auf dem Bildschirm darzustellenden grafischen Objekte als Folge von 16-bit-Bildschirmworten und bearbeitet bei der Aktivierung von peripheren Einrichtungen des Displays spezielle Dialogroutinen. Durch diese Funktionen wird die Verarbeitungsleistung des Rechners nicht ausgeschöpft, so dass gleichzeitig Programme zur Manipulation, Veränderung oder Berechnung der dargestellten Objekte bzw. bei der Kopplung mit einem ESER-Rechner zur Organisation des Informationsaustausches ablaufen können. Der Rechner wird dabei mit üblicher peripherie /im Prinzip beliebiger Ausbaustufe/ ausgerüstet, und ist über eine spezielle Gerätesteuereinheit /GSE/ mit dem Display verbunden /5/.

Während der Bilderzeugung übernimmt die Display-Steuereinheit über den externen Speicherkanal /ESK/ selbständig die aufeinanderfolgenden Befehls- und Datenworte aus dem Hauptspeicher

des Rechners. Dazu meldet die Steuereinheit nach Abarbeitung eines Befehls im GD`71 einen Zugriff zum Hauptspeicher an, der Rechner 4201 unterbricht nach Ausführung des in Bearbeitung befindlichen Befehls die weitere Befehlsausführung, die Steuereinheit entnimmt das gewünschte Befehls- bzw. Datenwort aus dem Hauptspeicher, die Programmabarbeitung im Rechner wird fortgesetzt und im GD`71 der übernommene Befehl ausgeführt bzw. weitere Informationen aus dem Hauptspeicher angefordert.

Von der Display-Steuereinheit ausgehende Unterbrechungsanforderungen, die durch Fehlerzustände des GD`71 bzw. Betätigung des Lichtstiftes oder der Tastatur ausgelöst werden, unterbrochen den beschriebenen Ablauf und führen zu Programmunterbrechungen im Rechner 4201. Ein entsprechendes Unterbrechungsbehandlungsprogramm übernimmt bei aufgetretenen Fehlern eine Protokollausgabe bzw. ruft die entsprechenden Dialogroutinen auf.

Der Lichtstift dient zum Identifizieren von auf dem Bildschirm erzeugten grafischen Elementen, indem ein im Blickfeld des Stiftes befindliches helles Element über eine lichtempfindliche Zelle die Abarbeitung eines speziellen Programmes im Rechner 4201 veranlasst, welches den Namen bzw. die Lage des "getroffenen" Elementes bzw. Objektes im Displayfile ermittelt. Das Objekt kann durch Angabe seines Namens oder seiner Position auf dem Bildschirm dem Nutzer kenntlich gemacht werden. Die Lichtstift-Identifikation ist auch eine Grundlage für die Arbeit mit Menüs. Dabei werden über spezielle Objekte auf dem Bildschirm /z.B. Schriftzeichen, Symbole/ Programme zur Veränderung oder Berechnung von Objekten aufgerufen.

Die Positionierkugel ist in beliebiger Richtung drehbar und gestattet die Einstellung eines beliebigen 2-Tupels $/x,y/$ im Bereich $0 \leq x,y \leq 1023$ ganz. Sie findet Anwendung zur Verschiebung von Objekten auf dem Bildschirm in dem vorgegebenen Rasterfeld oder zur Veränderung von Parameterwerten.

Durch Betätigung einer der 32 Tasten der Funktionstastatur lässt sich ein der Taste zugeordnetes Programm aktivieren.

Die alphanumerische Tastatur dient zur Kommandoeingabe und Parameterfestlegung bei der Dialogprogrammierung.

Das grafische Terminal ist als autonomes System oder in direkter Kopplung mit einem ESER-Rechner einsetzbar. Die Kopplung kann sowohl über kurze als auch grosse Entfernungen e /ggf. unter Benutzung des Telefonnetzes/ erfolgen /Bild 2/. An der Sektion Mathematik wird in Abstimmung mit dem Rechenzentrum der TU Dresden eine Kopplungsvariante realisiert, bei der das grafische Terminal über eine Entfernung von ca. 85 m an den Multiplex-Kanal eines EC 1022 angeschlossen ist. Diese Variante gestattet Übertragungsraten von 15...20 KByte/s und damit Reaktionszeiten bei einem über das Terminal durchgeführten Nutzerdialog mit dem ESER-Rechner im Sekundenbereich. Speziell für diese Kopplung wurde eine MPX-Kanal-Verstärkerstation entwickelt, die sowohl zum MPX-Kanal als auch zum grafischen Terminal das Anschlussbild 'SIF ESER' besitzt und an beliebiger Stelle des Kanals angeschlossen werden kann. Die Station ist die Quelle eines zweiten MPX-Kanalzweiges, der zum entfernt aufgestellten Terminal verläuft.

2. Programmierung und Bildschirmbetriebssystem

Um auf dem Bildschirm ein Bild darzustellen, muss im Hauptspeicher des Kleinrechners eine im Sinne der Interpretation durch die Steuereinheit des Bildschirmgerätes zusammenhängende Folge von 16-bit-Bildschirmworten erzeugt werden. Die Folge besteht aus Befehls- und Datenworten für u.a. das Positionieren sowie das Darstellen von Strecken, Kreisbögen /geviertelt/ und Schriftzeichen. Die Helligkeit kann in vier verschiedenen Intensitätsniveaus /einschliesslich "dunkel"/ gesteuert werden. Vier Linienarten stehen zur Verfügung. Zusammenhängende Folge heisst dabei, dass das nächste Wort

- das im Sinne der Adressenzählung nachfolgende oder
- das durch Bildschirm-Sprungbefehl und Sprungadresse bestimmte oder
- das bei Rückkehr aus einem Bild-Unterprogramm durch die hardwaremässig gekhellerte Rückkehradresse /maximale Tiefe 64/ festgelegte

Wort ist. Soll das durch die Bildschirmwortfolge beschriebene Bild sichtbar sein, muss innerhalb dieser Folge /durch z.B. einen Sprung zum "Anfang" am "Ende"/ für deren zyklischen Durchlauf gesorgt werden. Die Menge aller darstellungsfähigen Speicherbelegungsabschnitte, gleich ob lokal zusammenhängend oder nicht und ob derzeit in die Darstellung einbezogen oder nicht, wird Displayfile genannt. Im Displayfile können durch einen speziellen Bildschirmladebefehl und nachfolgendes Identifikationswort über die Bildschirmsteuereinheit abfragbare Identifikatoren gesetzt werden /Laden des Name-Registers/.

Ein Bild durch Programm erzeugen heisst also, Bildschirmwortfolgen /gegebenenfalls strukturiert/ zu generieren und/oder zu Zyklen zu verketteten. Im Zyklus kann bei jedem Befehlsword, das nicht zum Bestand eines Bild-Unterprogrammes gehört, zur Darstellung der Bildstart ausgelöst werden. Wird bei Verkettung mit dem gesamten Bestand des Displayfiles gearbeitet, indem verschiedene Abschnitte einbezogen oder wieder herausgelöst werden, entstehen sich verändernde Darstellungen, wobei zugleich neue Abschnitte generiert oder alte getilgt werden können. Die Generierung der Bildschirmwortfolgen kann programmtechnisch auf verschiedenen Niveaus vollzogen werden:

- Primitivstes Verfahren ist der unmittelbare /bitweise/ Aufbau des Bildschirmcodes im vorgesehenen Speicherbereich. Der Programmierer benötigt genaue Kenntnisse über das gesamte Gerätesystem.

- Bei Benutzung von /Bildschirmcode-/Generatoren wird dem Programmierer die Bildwort-Adressenzählung, die Codeerzeugung und insbesondere die Datenkonvertierung abgenommen. Es genügt, die Startwerte der Zählung einzustellen, die den Befehlen bzw. Daten entsprechenden Generatoren aufzurufen und dabei die Parameter /im Kleinrechnerformat/ zu übergeben. Der Programmierer muss die Syntax der Bildschirm- und Kleinrechnerwortfolgen, aber nicht mehr ihre Codierung benutzen. Die Generatoren sind im Sinne der Kleinrechnerprogrammierung durch Sprung mit Rückkehrabsicht zu erreichende Unterprogramme und Bestandteil des Bildschirmbetriebssystems. Neben dem manuell programmierten Aufruf ist die Benutzung der Generatoren durch Erweiterung der Assemblersprache des Kleinrechners /über Einbeziehung eines Vorübersetzers/ möglich. Prinzipiell können die Generatoren auch aus dem Betriebssystem herausgelöst verwendet werden.

- Höchste /kleinrechnerbezogene/ Stufe stellt die Generierung durch bildinhaltlich orientierte Unterprogramme dar, die in Assemblersprache programmiert, in dieser oder in höheren Sprachen /eventuell auch über Grossrechner/ aufgerufen werden. U.a. masstabgerechte Eintragung in das Koordinatensystem, Befehls- und Datenwortfolgenaufbau entsprechend der anzureihenden grafischen Elemente Punkt, Gerade, Kreis und Textzeichenfolge sowie Bildausschnittbegrenzungen werden automatisch vollzogen. Die Koordinatenparameter können Gleitkommazahlen sein.

Allen drei Stufen gemeinsam ist, dass die Bereitstellung der grafisch darzustellenden Daten mit der Generierung der Bildschirmwortfolgen zunächst nichts zu tun hat. Die Daten, die über das spezielle Ausgabegerät Bildschirm ausgegeben werden, werden hier lediglich maschinenorientiert für den Bildschirm aufbereitet. Die Daten selbst entstehen als Ergebnis herkömmlicher Programmierung, wobei diese durch grafisch orientierte Programmierhilfen gestützt sein kann.

Die Generierung von Bildschirmwortfolgen ermöglicht die Darstellung passiver Bildschirmbilder und werde daher grafische Programmierung passiv /GPP/ genannt.

Auch auf der höchsten Stufe von GPP wird der Programmierer mit der Verwaltung und gegebenenfalls notwendigen Verkettung der einzelnen im Displayfile generierten Folgen belastet. Sofern eine gewisse Standard-Strukturierung akzeptiert wird, stehen Unterprogramme der grafischen Programmierung in Standardstruktur /GPS/ zur Verfügung. Dabei wurden Ergebnisse bekannter Arbeiten /3, 10/ berücksichtigt. Die Standard-Displayfilestruktur sieht als kleinste, für den Nutzer ansprechbare grafische Einheit das Item vor. Die Unterprogramme der Strukturprogrammierung gestatten in ihrem Kern:

- die Itemdefinition /Itemöffnung, wahlweise Namensvergabe, wahlweise Informationszuweisung, Itemabschluss/,
- das Streichen, Verkürzen oder Erweitern eines Items,
- das Auffinden der Anfangsadresse des Items /:Itemadresse/ bei gegebenen Namen /:Itemname/, das Auffinden der Anfangsadresse des zugehörigen Namens- und Informationsblockes /Item-NI-Adresse/ bei gegebener Itemadresse,
- das Einschalten eines Items in die Darstellung und das Löschen eines Items aus der Darstellung,
- das Ausgeben der Liste aller Namens- und Informationsblöcke,
- das Löschen der gesamten aktuellen Darstellung und
- das Streichen aller Items.

Auf diesen Kern aufbauend sind wahlweise einsetzbare Unterprogramme abgeleitet, die z.B.

- Items innerhalb und ausserhalb der aktuellen Darstellung vertauschen oder ersetzen,
- das Blinken eines Items veranlassen,
- sofern das Item mit einem Positionierbefehl beginnt, diese Positionierung verändern oder dieses Item /mit veränderter Position/ kopieren,
- die Einbindung kompletter Items als Bild-Unterprogramme gleichberechtigt mit der Darstellung grafischer Elemente

bei der Itemgenerierung erlauben.

Mit der Einbeziehung der Bedienelemente Lichtstift, Funktionstastatur, alphanumerische Tastatur und Positionierkugel wird der Bildschirmdialog mit dem Rechner im Rahmen der durch das arbeitende Nutzer-Hintergrundprogramm gegebenen Voraussetzungen möglich. Die grafische Programmierung aktiv /GPA/ beinhaltet das Aktivieren und Reaktivieren der Bedienelemente, die Zuweisung von Interruptbehandlungsunterprogrammen und die Interrupterkennung. Für die Programmierung durch Nutzer wird wieder zweckmässig ein dem Bildschirmbetriebssystem zugeordneter Standardmodul bereitgestellt, der im wesentlichen nur die dialog-inhaltliche Seite offenlässt:

Reaktivierung und Aktivierung erfolgt über Unterprogramme, im Interruptfall werden definierte Informationen /Namen der identifizierten Elemente, Tastennummer usw./ angeboten, wobei die eigentliche Reaktion auf den Interrupt durch eigene Definition von Behandlungsunterprogrammen festgelegt werden kann. Typische Behandlungen gehören zum Standard.

Folgende Teile sind damit im Bestand des Kerns des /Kleinrechner-/Bildschirmbetriebssystems: der Displayfilebereich /4K Worte/, die Konvertierung der KRS-Formate und konventionelle Ein- und Ausgabe /2K Worte/, ein Kommando- und Steuer- teil /0,5K Worte/ und wahlweise die Minimalversionen von GPP /1,5K Worte zuzüglich 1K Worte benötigte Gleitkommaroutinen, die allerdings auch für die Nutzerprogrammierung zur Verfügung stehen/, GPS /0,5K Worte/ bzw. GPA /ca. 1K Worte/. Es bleiben /bei einschliesslicher Berücksichtigung des Grundsektors, der im Verhältnis der Gesamtbelegung besetzt ist/ noch 5,5K Worte des 16K-KRS zur Aufstockung des Betriebskomforts bzw. für Nutzerprogramme verfügbar.

3. Entwicklungstendenzen und Anwendung

Eine Weiterentwicklung dialogfähiger grafischer Terminals wird sich sowohl seitens der Programmierung als auch seitens der gerätetechnischen Ausstattung vollziehen. In erster Linie sollen dabei Spezialrecheneinheiten die Funktionsmöglichkeiten des Kleinrechners vergrößern, so dass neben den üblichen arithmetischen und logischen Operationen auch spezifisch geometrische Operationen, wie z.B. Verbinden, Schneiden, Transformation, Projektion, Rahmen und Ausschnittsbildung gerätetechnisch realisiert werden können /8/, /9/. Damit wäre eine schnellere Verarbeitung geometrischer Objekte auf dem Bildschirm gegenüber der bisherigen Verfahrensweise der Unterprogrammrufe oder des Zuschaltens eines Hintergrundrechners möglich. Mit der weiteren Entwicklung der Mikroelektronik könnten derartige Funktionsbausteine direkt in das Displaygerät eingebaut werden, um den Steuerrechner für solche Aufgaben, wie Displayfileverwaltung und Displayfilegenerierung aus Datenbeständen, die Objekte des zwei- und des dreidimensionalen Raumes beinhalten, besser zu nutzen. Die Arbeiten zur Weiterentwicklung der Programmierung digital-grafisch-geometrischer Systeme stehen in enger Wechselbeziehung mit den gerätetechnischen Entwicklungen. Besonderer Wert sollte darauf gelegt werden, dass der Nutzer für die Beschreibung grafisch- oder geometrieorientierter Probleme nicht unnötig mit Problemen der Rechentechnik belastet wird, sondern dass ihm geeignete problemangepasste Programmiersprachen zur Verfügung gestellt werden. Derartige Programmiersprachen sind so aufzubauen, dass sie sowohl für möglichst viele Einsatzgebiete die entsprechenden Operatoren und Operanden zur Beschreibung und Manipulation geometrischer Objekte bereitstellen als auch der spezielle Anwender g_e_n_a_u_s_e_i_n_e Operatoren und Operanden in ihnen vorfindet.

Eine derartige Forderung wird durch e_i_n_e grafisch/geometrieorientierte Sprache kaum zu realisieren sein, vor allem

wenn ausserdem spezifische nichtgeometrische aber mit den geometrischen Objekten in Zusammenhang stehende Parameter für die Modellierung hinzugezogen werden sollen.

Als Ausweg aus dieser Situation wird die Konzeption einer allgemeinen geometrie-orientierten Programmiersprache gesehen, die bausteinartig implimentiert wird. Sie sollte die grundlegenden, für die meisten Anwendungen immer wiederkehrenden geometrischen Operatoren und Operanden enthalten. Spezifische geometrische Operatoren und Operanden sollten Bestandteile problemorientierter Sprachen sein, die ausserdem die typischen nicht notwendig geometrischen Zusammenhänge des entsprechenden Einsatzgebietes widerspiegeln.

Fachsprachen können so aufgebaut werden, dass sie aus einzelnen Bausteinen der geometrie-orientierten Programmiersprache zusammengesetzt werden. Ihre Implementierung erfolgt dann analog der benutzten Basissprache. Sollte die Fachsprache dagegen aus eigenständigen Operatoren und Operanden aufgebaut sein, so werden mittels eines Sprachtransformationssystems Programme der Fachsprache auf Programme der allgemeinen geometrie-orientierten Programmiersprache zurückgeführt. In beiden Fällen können die in der allgemeinen Programmiersprache enthaltenen Algorithmen zur Verknüpfung und Darstellung geometrischer Elemente genutzt werden.

Neben diesen Arbeiten, bei denen die Erfahrungen mit DIGOS /6/ und DEPOT /4/ einfliessen, wird der Entwicklung geeigneter Datenstrukturen zur Verwaltung geometrischer Objekte und der damit in Wechselbeziehung stehenden nichtgeometrischen Parametern grosse Beachtung geschenkt. Dazu sind Untersuchungen zu folgenden Problemen notwendig:

- Entwurf speicherplatzsparender interner Strukturen zur Beschreibung geometrischer Objekte und zugehöriger nicht-geometrischer Parameter

- Zurückführung mehrerer Eingabevarianten für geometrische Objekte auf die einheitliche interne Beschreibungsform
- Wahrung des Zusammenhanges zwischen der internen und der für die Darstellung angepassten Struktur
- Berücksichtigung von im Dialog aktivierten Eingriffen in der Datenstrukturhierarchie
- Rückwirkung von Strukturänderungen auf die Programmiersprache.

Die Lösung der aufgezeigten Probleme führt zu einer effektiveren rechnerinternen Behandlung der geometrischen Objekte im Zusammenwirken mit grossen Datenbeständen, wie sie bei Normteilkatalogen u.ä. auftreten können. Ausserdem wird dem Wunsch entsprochen, einzelne Dialogschritte in ihrer Reihenfolge der Abarbeitung festzuhalten, um den Ablauf eines Konstruktionsprozesses zu ermitteln und nicht nur dessen statischen Endzustand.

Einsatzgebiete für den digitalgeometrischen Arbeitsplatz eröffnen sich überall dort, wo sich zur Lösung von Aufgabenstellungen eine konstruktive geometrische Herangehensweise anbietet und die zur Verfügung stehenden Algorithmen nicht oder nur mit erheblichem Aufwand unter Beachtung aller möglichen Sonderfälle zum Ziel führen. Besonders bieten sich natürlich für die Bearbeitung Aufgaben aus den Konstruktionswissenschaften an, ganz gleich ob es sich um Probleme aus Maschinenbau, Elektrotechnik oder anderen Fachgebieten handelt. Bei der Bearbeitung von Anwendungen wird besonderer Wert darauf gelegt, die Wechselbeziehungen zwischen Geometrie, dem zu lösenden Problem und der Informationsverarbeitung zu erkennen und daraus Bezugspunkte für die Lösung abzuleiten und zu entwickeln. Diese Vorgehensweise soll an einem Beispiel der Vorarbeitung des Einsatzes der Bildschirmtechnik für die Konstruktion von Werkzeugmaschinenstellen demonstriert werden /7/.

Ausgehend von der Analyse der zu bearbeitenden Aufgabenstellung, die in Zusammenarbeit mit dem späteren Nutzer erfolgte, wurde das Problem in prinzipiell folgende Teilaufgaben zerlegt:

- Beschreibung der geometrischen Gestalt des dreidimensionalen Werkzeugmaschinengestells
- Aufbau eines Modells zur Berechnung der statischen, dynamischen und technischen Verformungen
- Algorithmen zur Durchführung obiger Berechnungen
- Rückwirkung der Ergebnisse auf die Gestalt des Werkzeugmaschinengestells.

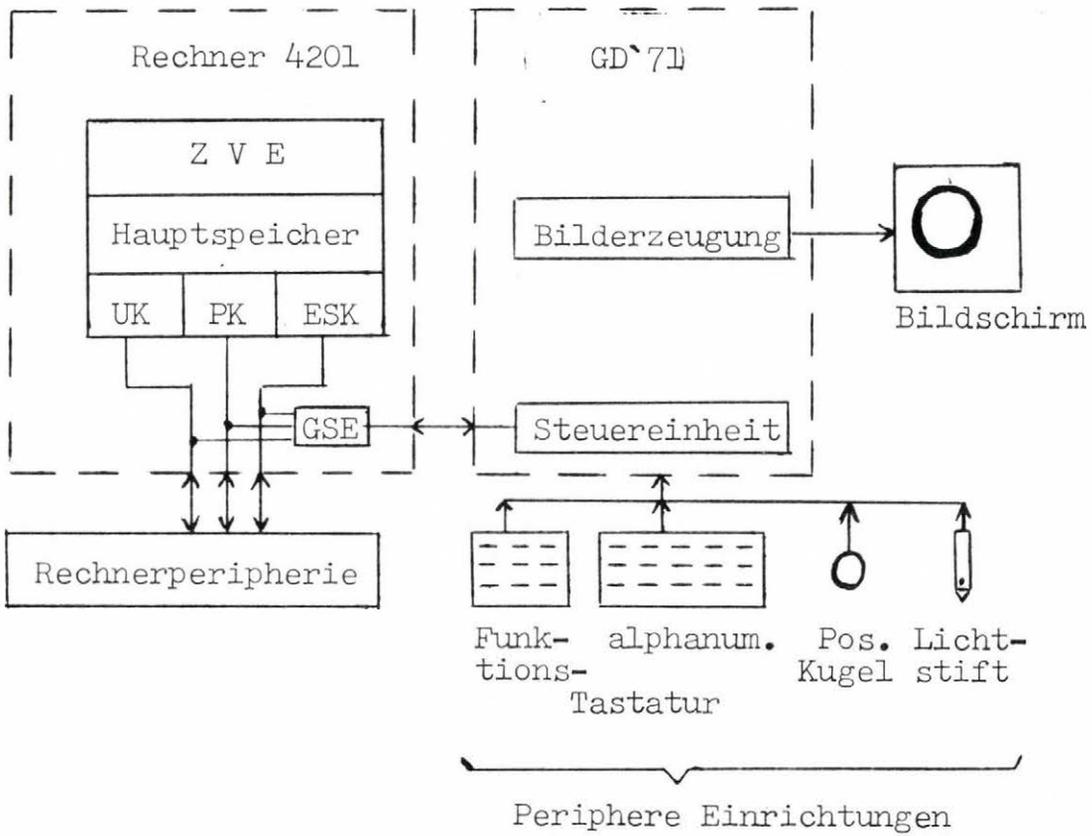
Dabei ist das interaktive Terminal vor allem bei Bewältigung der ersten beiden Problemstellungen einzusetzen. Voraussetzung ist die Entwicklung geeigneter Arbeitsmittel zur programmtechnischen Beschreibung des Gestells /Fachsprache/ einschliesslich notwendiger Veränderungen am Bildschirm im Dialog /ausgelöst durch die letzte Problemstellung/ und geeigneter Datenstrukturen, die den Zusammenhang zwischen der geometrischen Gestalt des Objektes und den Aufbau des zugehörigen Modells ermöglichen. Die Berechnungsalgorithmen, Optimierungsverfahren u.ä. laufen i.a. automatisch ohne Dialogeingriff ab.

Dieses Beispiel kann für weitere Probleme im Maschinenbau, z.B. in der Getriebetechnik, in der Elektrotechnik, in der Mikroelektronik und in anderen Fachdisziplinen stehen, die mit vertretbarem Aufwand nicht mehr automatisch gelöst werden können und wobei stets ein Zusammenhang zwischen der Gestaltung, Dimensionierung oder Anordnung geometrischer Objekte und technischen oder technologischen Prozessen besteht.

LITERATURVERZEICHNIS

- 1./ GD 71/T Hardware Manual; Forschungsinstitut für Rechentechnik und Automatisierung der Ung. Akademie d. Wissenschaften, Budapest

- 2./ Forgács, T., u. Verebély, P.: GD 71/T Graphical Display and Input devices. Hardware/Programming Manual. Editor: Krammer, G., Second Edition, Dec. 1974, Budapest
- 3./ GIPS - Grafisches Interaktives Programm System /Programmierhandbuch/; Institut für Schiffbau, Rechenzentrum, Rostock, 1974
- 4./ Bormann, J., u. Löttsch, J.: Definition und Realisierung von Fachsprachen mit DEPOT. Diss., TU Dresden 1974
- 5./ Franke, W.: Zur Kopplung GD`71 - KRS 4201; Wiss. Z.d. TU Dresden
- 6./ Ludwig, M., Richter, Chr., u. Weber, R.: Sprache und Datenstrukturen des digitalgeometrie-orientierten Systems DIGOS; Wiss. Z.d. TU Dresden 25 /1976/ H.3
- 7./ Ludwig, M.: Eine Fachsprache für die Konstruktion von Werkzeugmaschinenstellen; erscheint in `Berichte des Rostocker Mathematischen Kolloquiums`1977
- 8./ Monjau, D.: Zur gerätetechnischen Realisierung geometrischer Operationen; erscheint in `Berichte des Rostocker Mathematischen Kolloquiums`1977
- 9./ Ortleb, R.: Rechentechnisch-orientierte Darstellung einfacher geometrischer Grundaufgaben; RT/DV 13 /1976/ 1, 46-47
- 10./ Darvas, P., Hosszu, P., u. Krammer, G.: The Design of Another Graphic Subroutine Package; Computer and Automation Institute, Hungarian Academy of Sciences, Report, Budapest 1976



UK Unterbrechungskanal
 PK Programmierter Kanal
 ESK Externer Speicherkanal

Bild 1 Bestandteile des grafischen Terminals

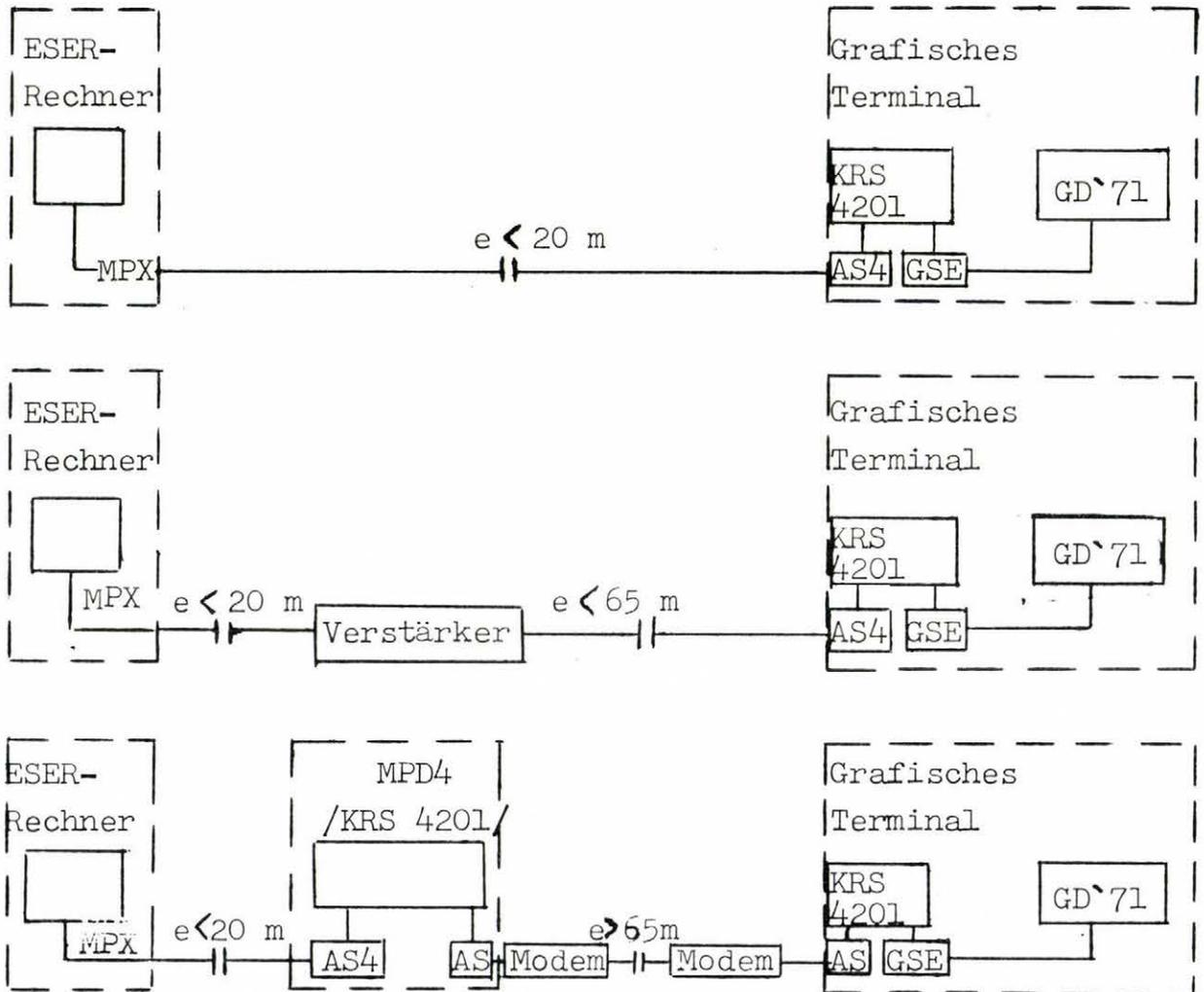


Bild 2 Kopplungsvarianten des grafischen Terminals mit ESER-Rechnern

G. Kramer

Forschungsinstitut für Rechen-
technik und Automatisierung der
Ungarischen Akademie der Wissen-
schaften

METHODEN DER ANWENDUNG DER INTERAKTIVEN DIGITALGRAPHIK
IM RECHNERGESTÜTZTEN KONSTRUIEREN

1. Einleitung

Über die Geräte der Digitalgraphik und die graphischen Software-Probleme könnte man einzeln zwei Vorträge halten. Die Anwendung der Digitalgraphik für verschiedene Applikationen könnte auch Gegenstand interessanter Vorträge sein.

Hier haben wir unternommen, die Anwendung verschiedener graphischen Hardware/Software-Mittel in den Applikationssystemen bekanntzumachen.

Diese Forschungen helfen uns die Rolle, den Aufbau des graphischen Softwares im allgemeinen, sowie die zweckmässige Organisierung der Anwendungsprogrammsysteme zu verstehen.

2. Modell der graphischen Ausgabe/Eingabe

Die grundlegenden Begriffe unseres Themas - der rechnergestützten Graphik - und unsere fundamentale Anschauung bezüglich des Themas werden hier durch sein Ausgabe/Eingabe Modell klargelegt /Abbildung 1./.

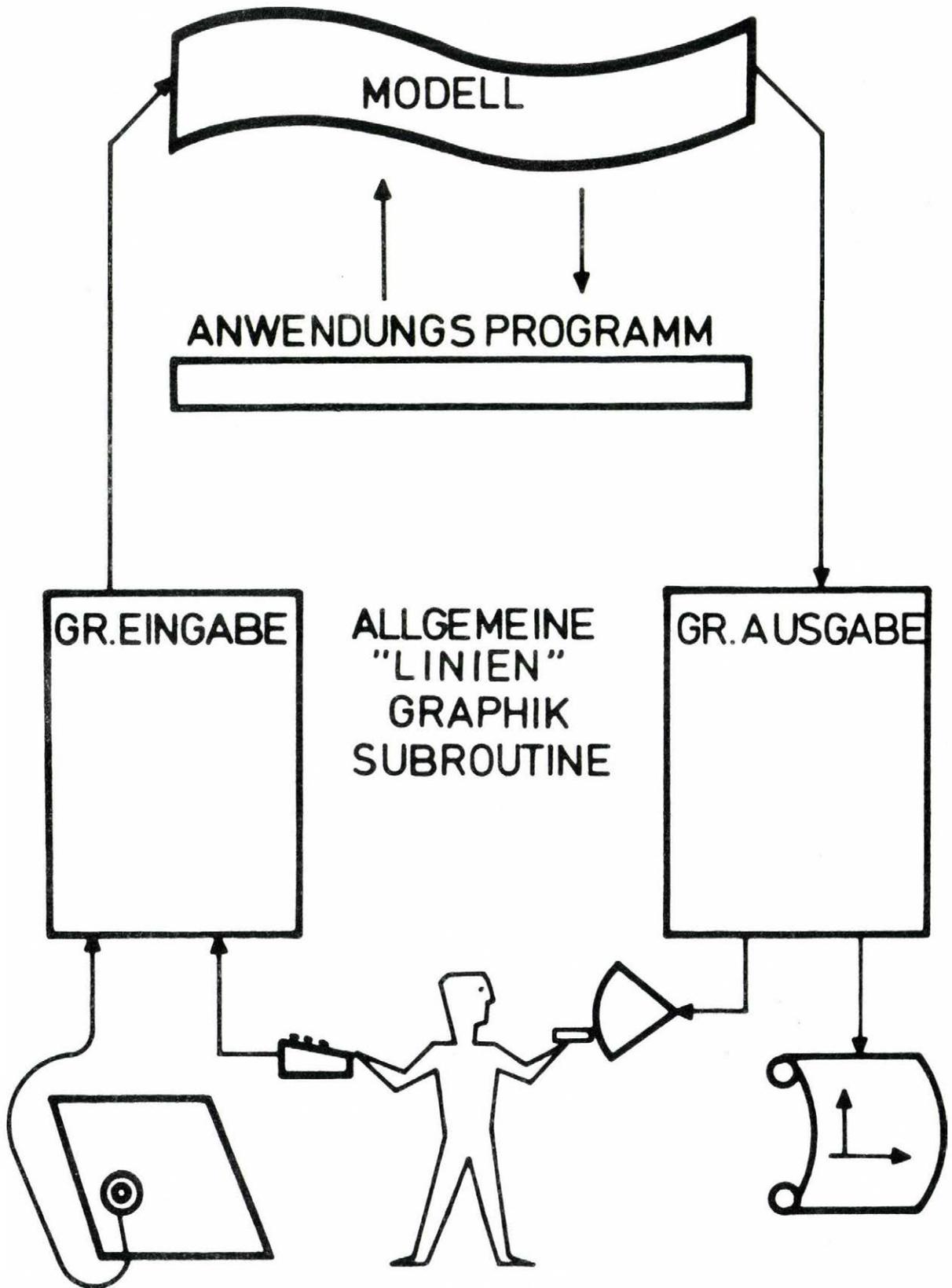


Abb. 1.

Die Aufgabe des Anwendungsprogrammes - im allgemeinen konzipiert - ist Analyse und Synthese bezüglich irgendeines Objektes, kurz gesagt: Modellierung.

Das Programm speichert die den Objekt beschreibenden Daten in einer Datenstruktur; diese Beschreibung des Objektes kann das Modell des Objektes genannt werden. Das Modell kann geometrische und andere - z.B. technologische - Information enthalten.

Von der Modellierung können die Eingabe- und Ausgabe-Operationen gut getrennt werden; für uns ist hier in erster Linie das graphische Input/Output interessant. Das Wesen des graphischen Ausgabe ist, dass ein ausgewählter Teil des Modelles dargestellt werden muss; von der Datenstruktur muss man die relevanten Daten auswählen und aufgrund dieser die gewünschte Abbildung ausarbeiten. Der graphische Eingabe-Prozess ist ähnlich, aber umgekehrt.

Das Wesen der mit Abbildung dargestellten Auffassung ist die Trennung einerseits der graphischen Eingabe/Ausgabe und andererseits der geometrischen Modellierung.

In der Modell Datenstruktur sind die Daten der Ansprüchen der Analyse und Synthese gemäss - die den grösseren Teil der Arbeit bedeuten - organisiert. Im Falle der fallweisen graphischen Aufgabe wählt das Anwendungsprogramm von den geometrischen Daten des Modelles die zur gewünschten Abbildung benötigten Daten aus. Umgekehrt: das Programm legt die graphischen Eingabedaten entsprechenderweise ausgelegt ins Modell ein.

4. Struktur von Anwendungssystemen

Zu den zweckmässigen Methoden der Erstellung von Anwendungssystemen ist zweckmässig, zuerst ihren Aufbau zu analysieren.

Wie bekannt, ein jedes Programm ist grundlegend mit der Beschreibung von Input, Output und Algorithmus zu bezeichnen.
/Abbildung 2./

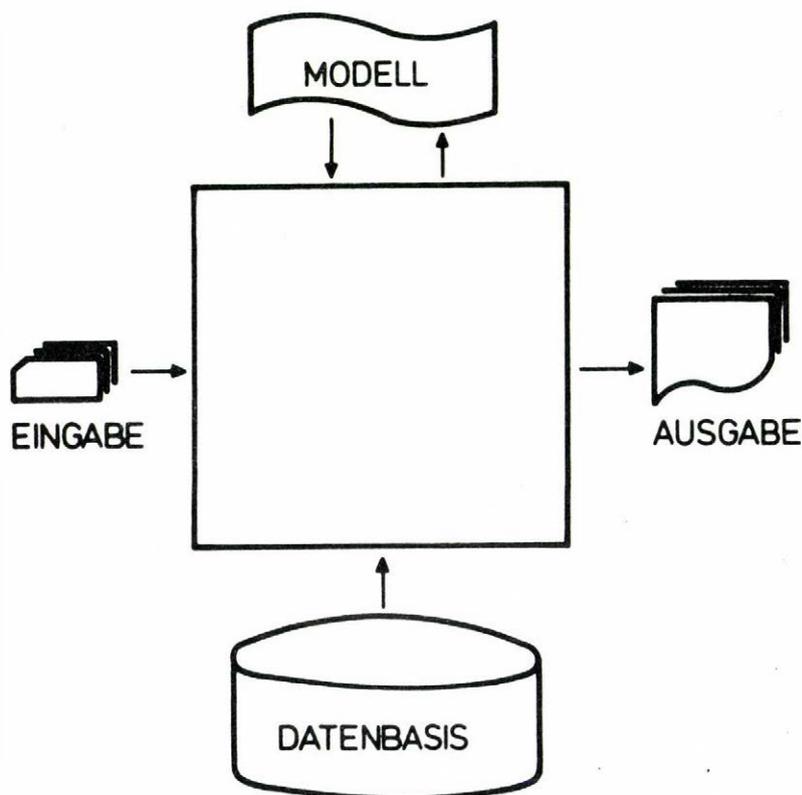


Abb.2.

Es ist zweckmässig, die Charakterisierung grösserer Programme
- mit der Charakterisierung des angewandten Daten-
basis und der Modell Datenstruktur zu erweitern.

Die Datenbasis ist im Laufe der Funktion des Programmes der Speicher von unveränderten Daten, während die Modell Datenstruktur den geordneten Speicher der im Laufe des Programmes entstehenden Daten bildet. Zwischen den letzten sind die Ergebnisse des Programmes, die "Ziel-Daten" zu finden, und es gibt auch provisorisch benötigte Daten.

Auch die Daten der Datenbasis können sich mit der Zeit verändern, die Datenbasis ist unabhängig von dem geprüften Anwendungsprogramm gewartet.

Die Eingabe-/Ausgabedaten sind auf der Abbildung durch Lochkarte und gedruckte Blätter veranschaulicht. Diese sind "ovrher gemachte" Daten und stellen "endgültig ausgeschriebene" Ergebnisse dar.

Im Falle von Dialogprogrammen /Abbildung 3/ hat die Eingabe/Ausgabe einen schnelleren, flexibilen Weg: der Mann /der Konstrukteur/ erhält Ergebnisse durch eine Dialogperipherie /Schreibmaschine, Display/ und nach ihrer Auswertung kann er dem Programm Daten angeben.

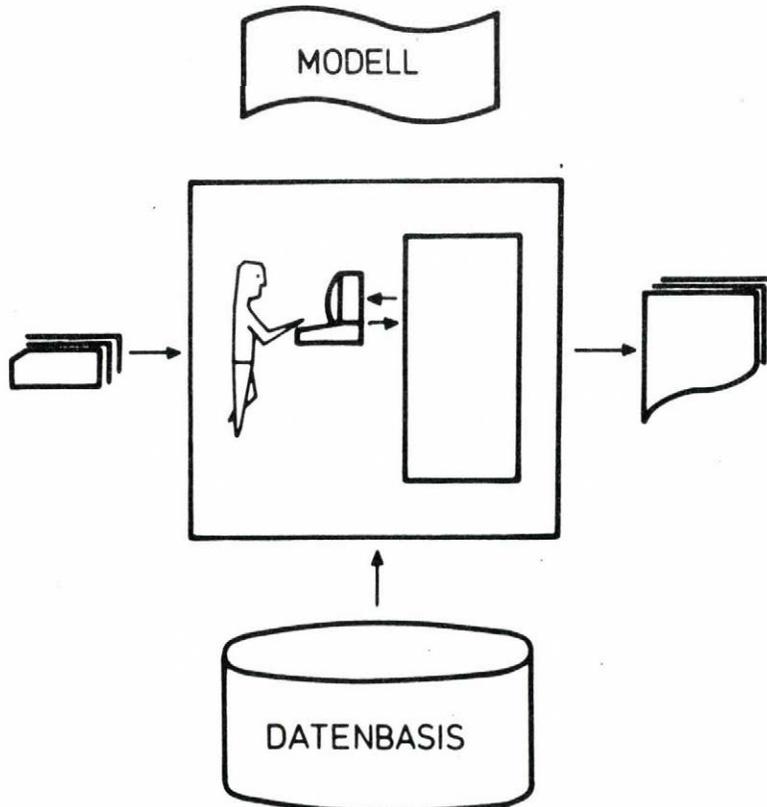


Abb.3.

Das Dialogprogramm besteht aus den aufeinanderfolgenden Schritten des Dialogs, ein jeder von diesen kann mit den in der betreffenden Dialogsprache vorkommenden Eingabe-, Ausgabedaten, mit dem angewandten Modell/-detail/ und Datenbasis/-detail/ charakterisiert werden.

Grössere Programmsysteme bestehen auch über dies aus gut trennbaren Programmphasen. In grossen Entwurfsprogrammen sind oft zu trennen:

- die Phasen der Datenaufbereitung, der Datenbasis-Wartung, des die Bestimmung der Ergebnisse ausführenden Entwurfsprogrammes und der Ergebnisanalyse /Abbildung 4/.

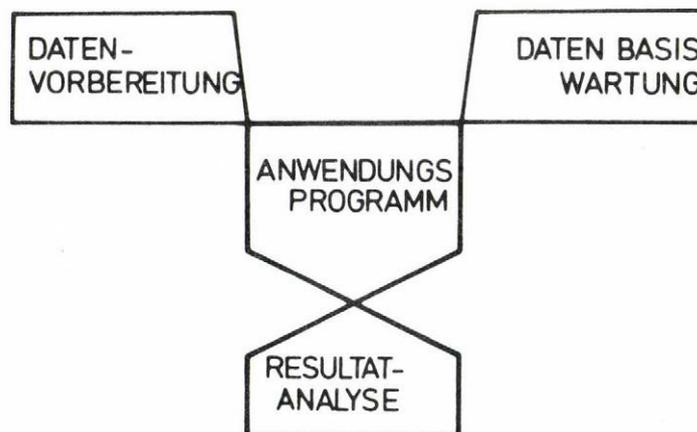


Abb.4.

Von den Anwendungen abhängig ist es zweckmässig, das Programmsystem auf Phasen zu zerlegen; unten den Phasen sind Dialogprogrammphasen und algorithmische Programmphasen zu finden, es gibt welche, die graphische Eingabe/Ausgabe verwenden oder nicht verwenden /Abbildung 5/.

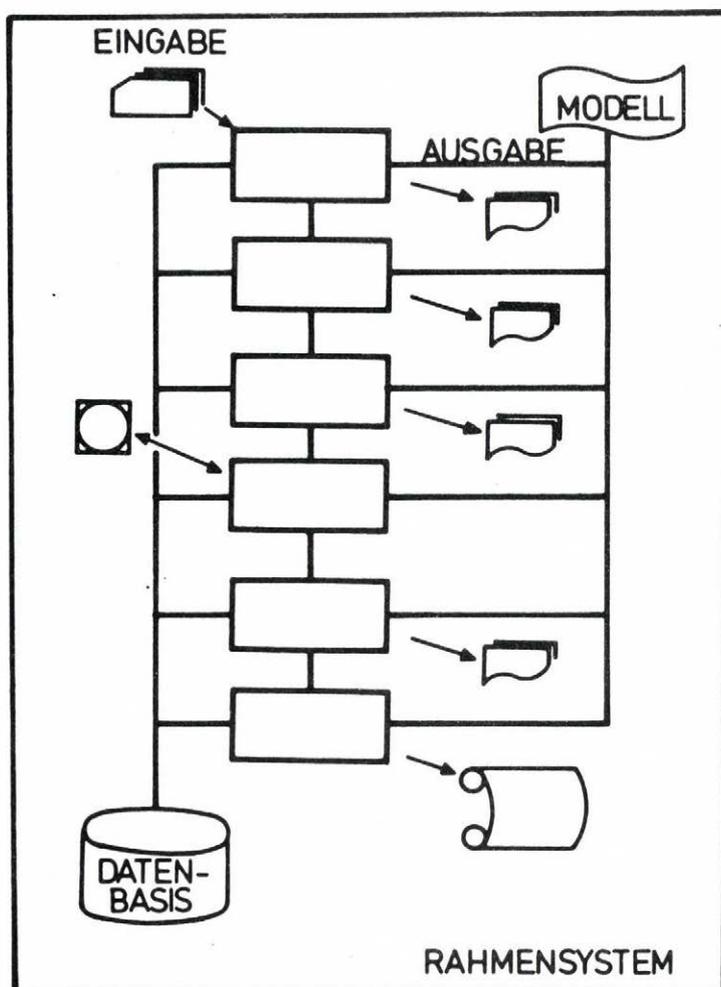


Abb.5.

Die Zerlegung des Programmes auf Phasen, die Aufgliederung des Dialogprogrammes in Schritten, die Eingabe, Ausgabe, die Trennung des Modelles und der Datenbasis /in jeder Phase und Schritt/ hilft beim Verstehen und Konstruieren des Systemes. Diese Teilen bilden aber schliesslich ein einheitliches Ganzes.

Von einem Standpunkt aus bedeutet den das Programmsystem in eine Einheit fassenden Rahmen das Operationssystem, das die Behandlung der Programme, die Speicherung und den Schutz der Daten sichert. Für ein Entwurfssystem wird oft ein "Rahmenprogramm" zustande gebracht.

Die gemeinsam angewandte Modell Datenstruktur und Datenbasis sichern den Zusammenhang zwischen den Programmphasen und Dialogschritten.

Die Einheit der Dialogschritten ist durch das Dialogsystem gesichert, das einerseits eine "Sprache" für die vorherige Bestimmung des Dialoges bildet, andererseits auch während des Dialoges sichert, dass Mann und Maschine in Richtung des festgelegten Ziels halten.

4. Linienzeichende graphische Programmiersprache allgemeinen Zweckes

Die erste, das Modell der das graphische Input/Output vorführenden Abbildung kann auf den folgenden, ausführlichen Weisen zerlegt werden: /Abbildung 6./

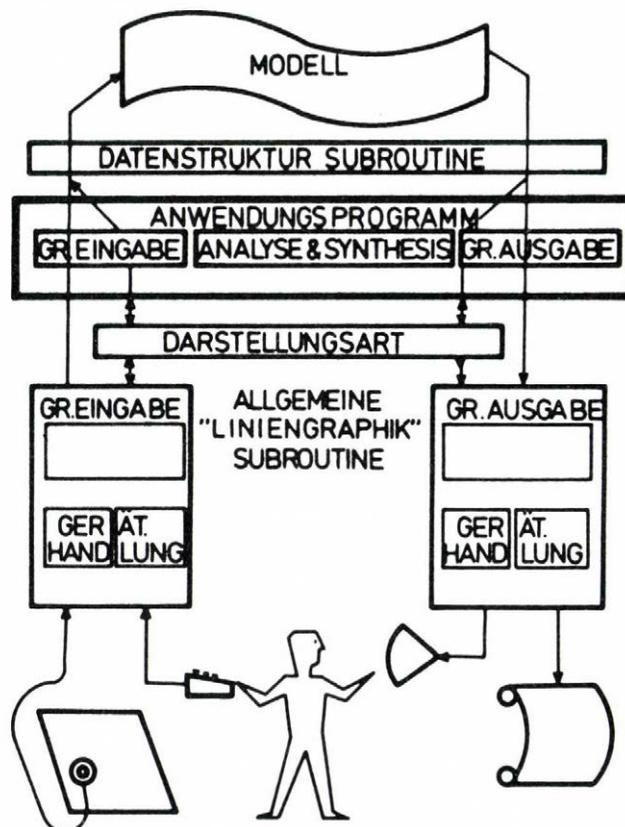


Abb. 6.

Die zwei - für uns - wichtigsten Funktionen des Anwendungsprogrammes sind: die Modellierung und das graphische Input/Output.

Wie die Anwendungsprogramme heute meistens auf Programm-sprachen von hohem Niveau geschrieben sind, ist es auch im Falle des graphischen Input/Output zweckmässig; in idealem Fall gebraucht man zu diesem Zweck das graphische Subsystem einer noch erweiterbaren Programm-sprache von hohem Niveau.

In den wohlbekanntesten Sprachen von hohem Niveau - FORTRAN, PL/I - kann es nur so durchgeführt werden, dass ein graphisches Subroutine-Package allgemeinen Zweckes zustande gebracht wird. /Die graphische Erweiterung einer Sprache kann durch Makros vorgenommen werden, die von einem Preprozessor in Subroutine-rufen umgewandelt werden./

Die wichtigsten Funktionen von ~~so~~ einem linienzeichnenden graphischen Subroutine-Package allgemeinen Zweckes sind:

- Eingabe/Ausgabe der Bildelemente
- Auslegung einer Art Bildstruktur
- Speicherung einer /mehr beschränkten/ Bildstruktur und Bildmanipulation
- Identifizierung von Bildelementen
- Übertragung des Anwendungskoordinatensystemes zum Koordinatensystem des Input/Output-Gerätes

Das Subroutine-System ist auf zwei Teile zu zerlegen: der erste Teil beschäftigt sich mit den allgemeinen geometrischen und manipulatorischen Aufgaben, während zum anderen Teil die Kodegeneratoren der verschiedenen graphischen Geräte gehören.

Das Subroutine-System ist für Aufgaben geeignet, wo wir mit Abbildungen aus Linien /Punkten, Geraden, Kreisen/ arbeiten.

Die Anwendungen können innerhalb der obigen verschiedene "Darstellungsarten" gebrauchen /Fertigung von dreidimensionalen Projektionsabbildungen und von axonometrischen Abbildungen usw./ zu der Auslegung derer logischen Elemente die sich auf den vorherigen bildenden Subroutinen fertiggemacht werden können. Diese können als ein Teil der "angewandten" Graphik betrachtet werden.

Die Untersuchung von Anwendungen weist darauf hin, dass diese vom Standpunkt der graphischen Eingabe/Ausgabe zweckmässig in drei Gruppen einzuteilen sind.

In der ersten Gruppe kann die obenerwähnte graphische Eingabe/Ausgabe von hohem Niveau zweckmässig gebraucht werden. In diesem Fall ist die Distanz der dargestellten Bildelemente, der identifizierten Bildelemente und des Niveaus der manipulierten Bildteile charakteristisch.

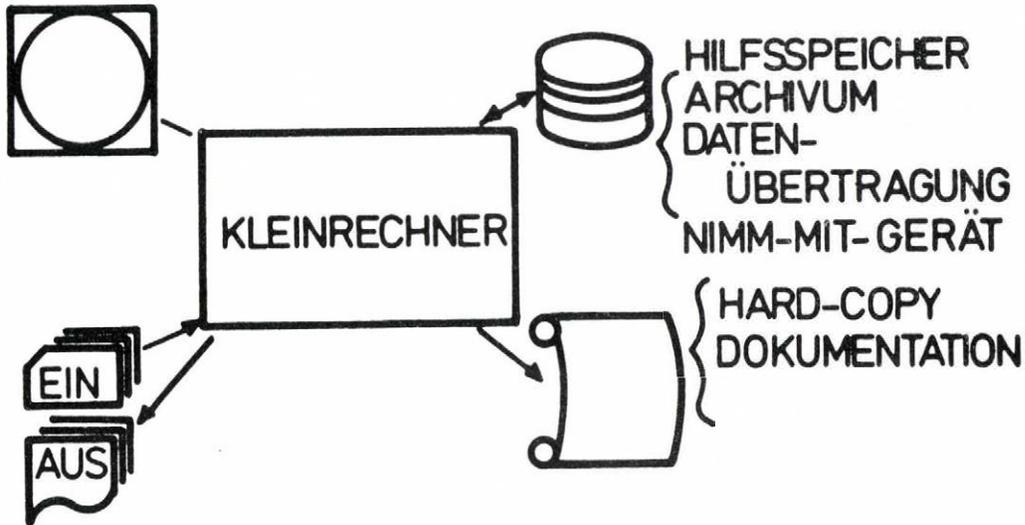
In der zweiten Gruppe verschwindet sich die Distanz, und selbst das Anwendungsmodell speichert die Abbildung /oder Information von linearen Abbildungsart/. Die Applikationsgruppe oft verwendet integer Raster-Koordinaten.

In dieser Gruppe haben die direkten Abbildungsmanipulationen einen Sinn, während im Falle eines Höchststandes nur die Modellmanipulationen vom Sinne sind, die Abbildung wird zweckmässig durch Darstellung des veränderten Modells modifiziert.

5. "Arbeitsteilung" graphischer Konfigurationen

Unten den interaktiven graphischen Konfigurationen sind drei Haupttype zu unterscheiden /Abbildung 7/:

- selbständige Kleinrechner-Konfiguration
- einfaches graphisches Terminal
- graphische Satellit-Konfiguration.



SELBSTÄNDIGE GRAPHISCHE KONFIGURATION



EINFACHES TERMINAL

Abb.7

Die selbständigen Kleinrechner-Konfigurationen sind in Fällen gut zu gebrauchen, wo das Entwurfsprogramm gut segmentierbar ist und die Grösse des angewandten Modelles und der Datenbasis es erlaubt. Die Änderung des Begriffes "Kleinrechner" muss natürlich in Betracht genommen werden. Eine wichtige Aufgabe bedeutet die Sicherung des Datenflusses zwischen der selbständigen Konfiguration und anderer Rechenmaschinensystemen - das kann z.B. durch Magnetband gelöst werden.

Das einfache graphische Terminal ist ein Eingabe-Ausgabe Gerät eines Grossrechners. Dies wird in allgemeinen Fällen das Kode interpretieren, das von dem Kodegeneratoren erzeugt ist. Das Kodegenerator ist meistens ein Unterprogramm im Grossrechner.

Neue, programmierbare Steuereinheiten ermöglichen die Kodegenerierung im Terminal durchzuführen.

Schliesslich kann die Intelligenz des Satellites durch Steigerung der Kapazität /Speicher und Peripherien/ des als Terminal angewandten Kleinrechners in zwei Richtungen erhöht werden:

- der Stand des ausgelegten Codes kann erhoben werden,
- bzw. ganze Phasen des Anwendungsprogrammsystemes können für den Satellit überlassen werden /Abbildung 8./.

ZYKLISCHES BATCH

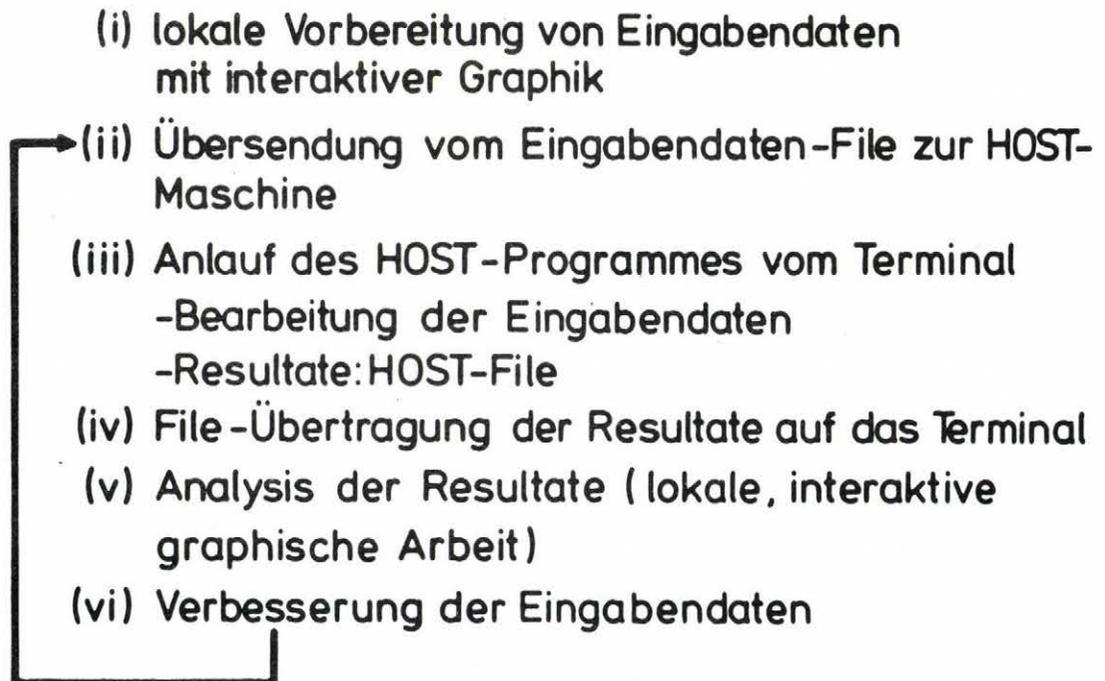


Abb.8.

Im "zyklischen Batch"-Betriebsart vorbereitet der Satellit die Daten durch interaktive, graphische Arbeit. Diese werden in Form eines Files, als Batch-Arbeit zum Grossrechner übertragen werden, woher später die Ergebnisse auch in File-Form können erhalten werden. Die Auswertung der Ergebnisse kann wieder am Satellit durchgeführt werden.

Die Arbeit des Satellites und des "Grossrechners" ist vollkommen asynchron, können unabhängig von einander für andere Arbeiten verwendet werden. Das zyklische Batch ist besonders in Umgebungen wichtig, wo nur ein Mittelrechner zur Verfügung steht /es gibt Grossrechner nur über 1 MByte, so eine Maschine ist zur Zeit in Ungarn noch nicht zu finden./

Ein interessantes Forschungsthema für die Zukunft bedeutet die dynamische Teilung der Aufgaben /Tasks/ zwischen Grossrechner und Satellit - von der Momentanbelastung abhängig gemacht.

6. Das graphische System GD`71/T

Die oben erwähnten werden am graphischen System GD`71/T illustriert. Der allgemeinen Auslegung der rechnergestützten Graphik gemäss versuchten wir das Hardware/Software GD`71/T zu verwirklichen /soweit es uns die Entwicklung der Technik, die Parallelität der Forschung, der Weiterentwicklung und der Anwendungen erlauben/.

Den Kern des GD`71/T bildet das graphische Display GD`71 /das ist eine Anzeigeeinheit von Mittelkategorie/ zusammen mit dem TPA-70 Kleinrechner. /Das GD`71/T Software wurde grundsätzlich am TPA-70 ausgearbeitet, dieses gutbewährte Software wollen wir im Falle auch von anderen Maschinen anwenden./

Das Operationssystem DOST für TPA-70 geschrieben bildet den Rahmen der rechnergestützten Systeme /und es leistet auch ein Programmsystem/. Wegen der engen Verbindung zur Anwendungen können wir das DOST-System nicht beenden; erfahrungsgemäss wird es manchmal mit einem File-Behandlung-System, manchmal mit einem Editorprogramm erweitert. In der letzten Zeiten wurden der "im Hintergrund" parallel mit der interaktiven Arbeit funktionierende File-Printer, File-Punch und File-Plott Programme fertiggemacht.

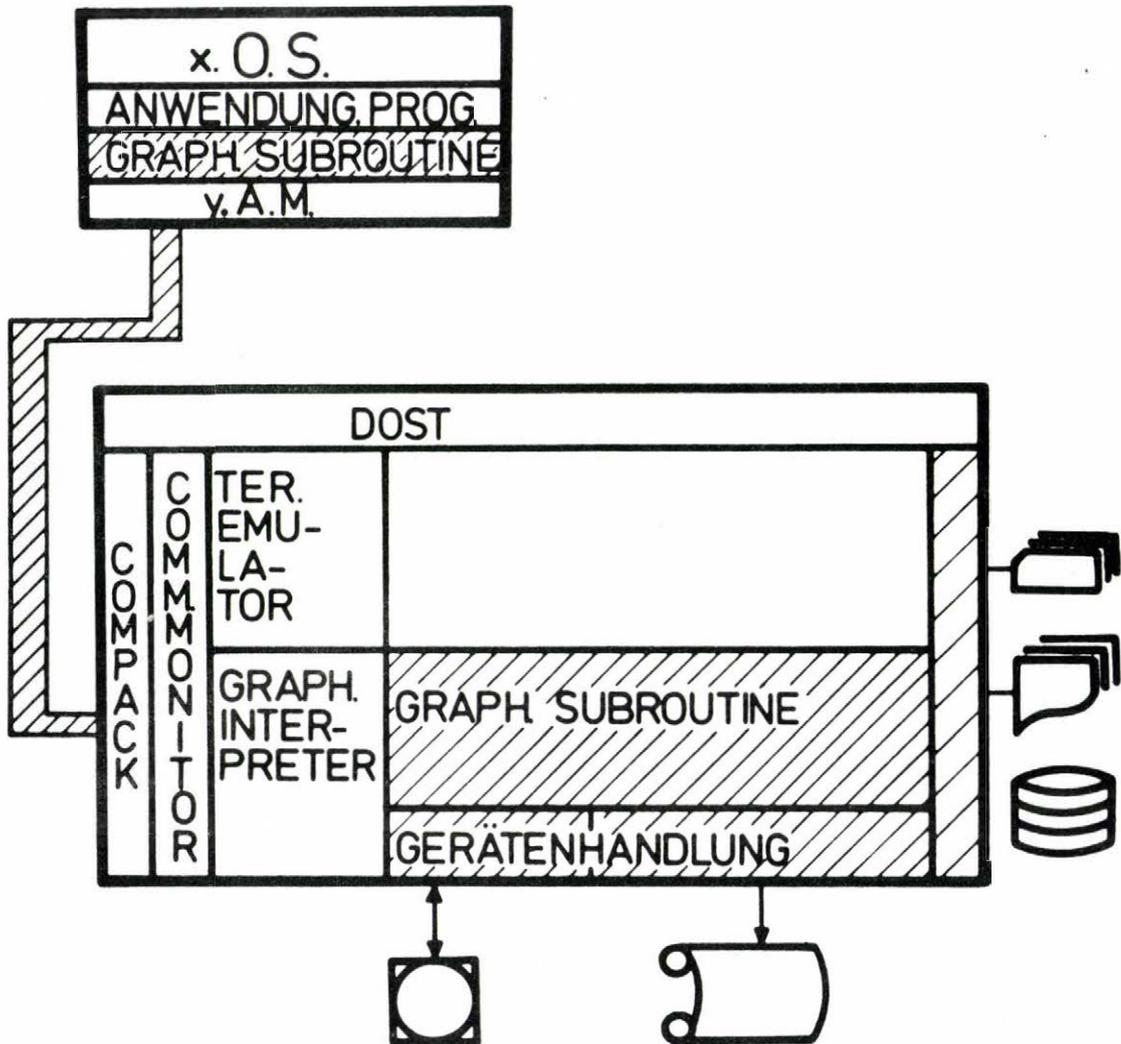


Abb.9.

Von den früher bekanntgemachten Software-Komponenten möchte ich hier einige aufführen:

- GDIO: grundlegendes graphisches Input/Output;
- GTU: Allgemeines linienzeichendes graphisches Input/Output;
- FORTRAN Kompiler, Assembler;
- Overlay-Organisierung;
- Kommunikationsroutine für CDC und IBM Rechenmaschinen;
- CDC und IBM Batch Terminal Emulator.

Der moderne Aufbau des ungarischen Kleinrechners TPA-70 ermöglicht, verschiedene Peripherien zum Input/Output Bus anzuschliessen. Von den bisher angewandten sind die wichtigsten wie folgt:

- Speicher /800 KByte fix, oder 2.5 MByte fix + 2.5 MByte austauschbar/;
- Lochstreifen I/O, Zeilendrucker, Kartenleser;
- CALCOMP und DIGIGRAF Zeichengeräte;
- Floppy Disk Kopplung /das ist noch in Vorbereitungsphase/.

T. Tolnay-Knefély-D. Kerestély
Forschungsinstitut für Rechen-
technik und Automatisierung
der Ungarischen Akademie der
Wissenschaften

EIN GRAPHISCHES PROGRAMM MIT FRAGE- UND ANTWORT-
VERFAHREN FÜR DIE MODIFIKATION UND KORREKTUR DER
LEITERABBILDUNG GEDRUCKTER LEITERPLATTEN

1. Einleitung

Dank der schnellen Entwicklung der Digitaltechnik werden immer mehrere gedruckte Leiterplatten entworfen und hergestellt. Im Folge des scharfen Konkurrenzkampfes besitzt die kurze Durchlaufzeit eine immer grössere Bedeutung. Das Programm, das ich Ihnen hier bekanntmachen möchte, ermöglicht die schnelle, wirkungsvolle und fehlerlose Modifikation und Korrektur des entworfenen Druckbildes.

Die grundlegende Aufgabe des Programmes besteht darin, eine Möglichkeit für Streichung von überflüssigen Leitungen, für Einschaltung neuer Leitungen und für geometrische Transformation der von irgendeiner Rücksicht zusammengehörenden Teile des Druckbildes zu sichern. Ausserdem soll es noch wirksame Leistungen zur Durchführung der obigen bieten und bestimmte Kontrollfunktionen verrichten!

Das Programm fügt sich eng in das in unserem Institut entwickelte, schon lange als Leistung verwandte, gedruckte Leiterplatten entwerfende Programmsystem. Als Input kann das Verdrahtungsbild der gedruckten Leiterplatte, mit Hilfe von Grossrechnermaschine oder manuell entworfen, funktionieren; als Output spielen die Eingabedaten des Postprozess-Programmes, das das Steuerloch der numerisch-gesteuerten Bearbei-

tungsmaschine erstellen, eine Rolle.

Obwohl wir unser für einen Kleinrechner geschriebenes Programm nicht als Entwurfsprogramm zustande gebracht haben, ist diese Aufgabe so vielseitig und kompliziert, dass es ist der Grossrechner, der uns eine wirksame Lösung ermöglicht - ihr Dasein ist wie folgt begründet.

Unser Entwurfsprogramm ist nur zur Behandlung integrierter Stromkreismodule fähig, deshalb müssen die gewohnten Elemente - Widerstand, Kondensator, Transistor unter anderem - und ihre Verknüpfungen nachträglich eingestellt werden. Ein Charakterzug des Algorithmus für Verdrahtung ist, dass das Programm die Leiterabbildung auf jeden Fall entwirft, aber im Falle eines komplizierten Stromkreises kommt es vor, dass es einen grösseren Teil in Anspruch nimmt, als was vorhanden ist. In beiden Fällen ist eine nachträgliche Modifikation nötig.

Laut des Lee-Algorithmus oder seiner veränderten Varianten können Verdrahtungsprogramme manchmal nicht alle Verknüpfungen an der Platte einlegen, man muss das nachträglich tun.

Entwurfsprogramme können nicht alle spezielle Anforderungen zum Beispiel beschränkte Drahtlänge im Interesse der Reduktion der Laufzeit in Betracht nehmen.

Auch heute wird noch die Leiterabbildung von vielen Platten manuell entworfen, derer Korrektur auch eine wirkungsvolle Methode verlangt.

In vielen Fällen stellt es sich nur bei Messung der Platte oder der Einrichtung heraus, dass der Logikentwurf des Stromkreises modifiziert werden muss. Damit wird natürlich auch die Verdrahtung verändert werden. Wenn es sich nur um kleinere Modifikationen handelt, lohnt es sich nicht,

mit der ganzen Entwurfsperiode neu anzufangen, die Korrektur des Druckbildes reicht.

Die bereitstehenden interaktiven graphischen Programme sind im allgemeinen universal, während unser Programm "nur" einer speziellen Aufgabe durchzuführen dient. Aber eben damit wird es möglich, auch Kontrollfunktionen verrichten zu können. Unser Programm nämlich - im Interesse der wirksamen und fehlerlosen Arbeit - kontrolliert ständig, ob - im Falle einer Streichung - irgendein Potential eine Menge von galvanisch zusammenhängenden Punkten nicht unterbrochen wurde, beziehungsweise ob - im Falle einer Einschaltung - die neu eingelegten Drähte, Bohrlöcher einen Kurzschluss nicht zustande gebracht haben, und in beiden Fällen zeigt es dem Operator an. Wir sind der Meinung, dass diese Kontrollfunktion die wichtige und wesentliche Neuheit des Programmes bedeutet.

2. Datenbasis und Programmorganisation

Als Input des Programmes dient die geometrisch-technologische Beschreibung des Druckbildes. Diese enthält die Linienstärke, die Seitenmarkierung, die absoluten Koordinaten der einzelnen Punkten, die in einzelnen Punkten auszubildende graphische Form, durchzuführende technologische Operation. Das Programm verändert die Eingabedaten seiner inneren Darstellungsart entsprechend soweit, dass das neue Modell des Verdrahtungszeichens die obenerwähnten Informationen vollkommen enthält, während es den schnellen und wirksamen Betrieb sichert. Eine Möglichkeit ist vorhanden, eine bestimmte Beschreibung den verschiedenen technologischen Parameterwerten gemäss auszulegen. Dazu ist nur erforderlich, dass der Verwender mit Hilfe eines - zur Verfügung stehenden - Programmes die ihm entsprechenden tech-

nologischen Parameter definiert, und die Datenumwandlung mit Benützung einer diese Weise generierten Tabelle erledigt.

Die sich logisch differenzierenden Teile des Programmes bilden selbständige Programmsegmente /Abbildung 1./

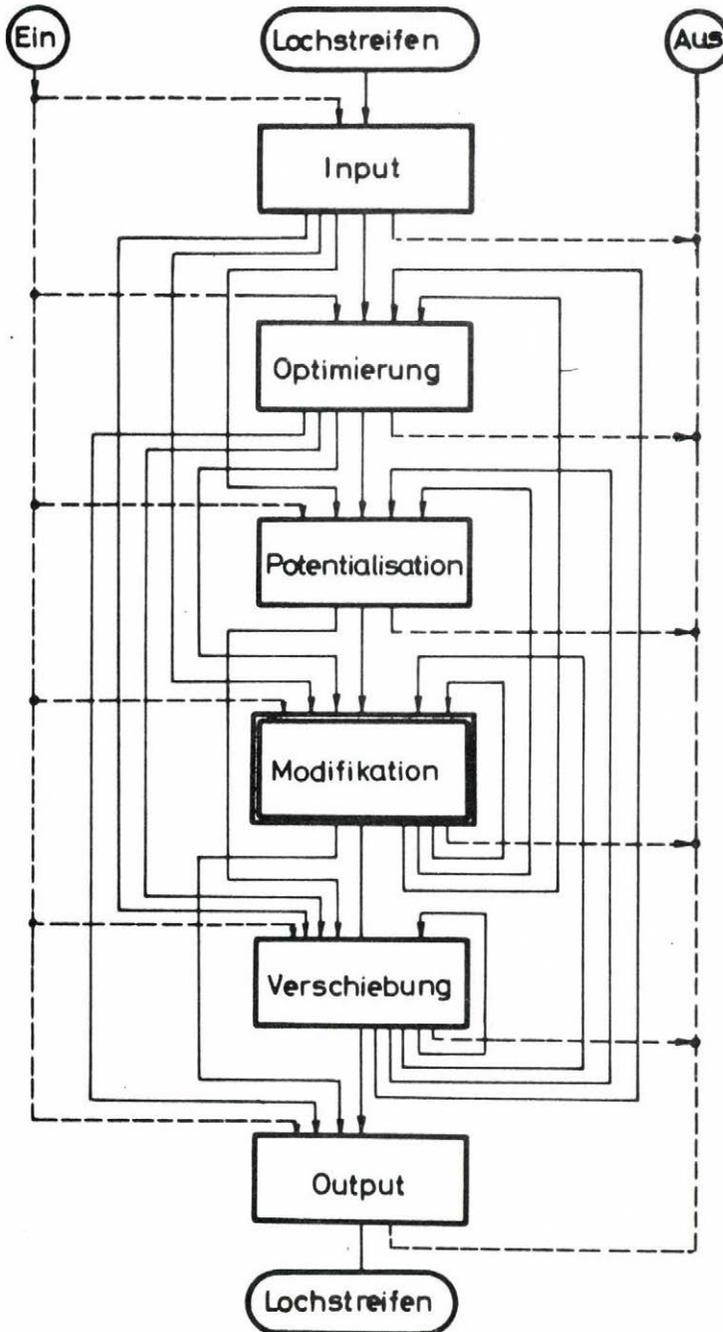


ABBILDUNG 1.

Die Programmsegmente sind in zwei Gruppen einzuteilen. Die zur ersten Gruppe gehörenden modifizieren die Verdrahtung der Karte der am graphischen Display verrichteten interaktiven Arbeit gemäss, während die anderen Segmente nehmen nur eine Modifikation am Modell vor, die die Modifikation des Druckbildes nicht ergibt.

Wenn wir am graphischen Display-Gerät arbeiten, ist das Verdrahtungsbild im Speicher des Rechners sogar in zwei Formen zu finden. Eine davon ist das obenerwähnte Modell, mit den absoluten Koordinaten, während die andere das sogenannte Bildprogramm bedeutet. Das Dasein der vorigen ist wegen der schnellen und wirkungsvollen Potential-Kontrolle nötig, während die letzte zur Darstellung des Verdrahtungsbildes beansprucht wird. Die gut angebrachten Pointer sichern die eindeutige Verbindung zwischen den gleichen Elementen beider Modelle.

Die logisch vertrennte Segmente sind gleichzeitig auch Overlay-Segmente, und auf diese Weise kann der mit verhältnismässig kleinem operativen Speicher versehene Rechner wirkungsvoll verwendet werden.

Da ein jeder Programmmodul seine Daten von dem Speicher bekommt und die Ergebnisse dort festlegt, wird die Kompliziertheit des Verdrahtungszeichnungs nicht durch die Speichergrösse, sondern durch die Kapazität des Magnetscheibenspeichers eingeschränkt. Auf diese Weise kann das Programm eine jede der in der Praxis vorkommenden Karten korrigieren.

3. Aufgabe der einzelnen Programmsegmente

Der Input-Segment erhält die Eingabedaten entweder von einem Lochstreifen oder einem Speicher. Er führt die Syntaxkontrolle aus /schreibt die fehlerhaften Daten, sowie den Typ des Fehlers heraus/, ruft das zur weiteren Arbeit erforderliche Modell ins Leben, und - falls es sich kein Fehler ergab - legt das Ergebnis am Speicher fest. Die durch das Programm behandelbare grösste Karte hat eine Grösse von 1 x 1 Meter, die Zeichenbeschreibung kann max. 16 verschiedene technologische Befehle enthalten und eine Linienbreite von 8 Arten ist erlaubt /der Wert dieser letzten ist nur vom Zeichengerät beschränkt.

Der Optimierungs-Segment stellt die einzelnen Linien in eine Reihenfolge, wobei der Elektronen-Strahl des graphischen Display-Gerätes während der Darstellung einen Minimalweg durchlaufen muss. Das ist wegen der Verringerung der Rezirkulationszeit - das heisst, wegen der Vermeidung des Bildgeflimmers - notwendig.

Der Potentialisations-Segment definiert die Teilmenge der galvanisch zusammenhängenden Punkte /einer jeden ist eine Zahl zugeordnet/. So wird es während der Modifikationen möglich sein, die Kurzschlüsse, bzw. die Potentialunterbrechungen anzeigen zu können.

Der Verschiebungs-Segment ist für den Zweck geeignet, die in dem bestimmten Rechteck fallenden Punkte der Leiterabbildung wunschgemäss zu verschieben.

Mit Hilfe des Modifikations-Segmentes können wir Leitungen streichen und einschalten, bzw. dieser Segment bietet uns Leistungen, die diese Tätigkeiten durchzuführen helfen.

Der Output-Segment gewährleistet uns eine Liste, bzw. einen Lochstreifen über die geometrisch-technologische Beschreibung der fertiggemachten Leiterabbildung, sowie legt das Ergebnis auch am Speicher fest. Das letzte bildet das Input des sogenannten Postprozess-Programmes. Das macht das Steuerloch der numerisch-gesteuerten Bearbeitungsmaschine fertig.

4. Modifikationsegment

4.1. Das graphische Eingabe-Ausgabe-System

Ein interaktives Programm soll es sichern, dass der Operator den aktuellen Zustand des Programmes und der Datenbasis kennt, und es soll eine Möglichkeit für die Veränderung aller beiden geben. Die Wirksamkeit der Arbeit ist grundlegend durch die Geschwindigkeit des Informationsaustausches zwischen Mann und Maschine bestimmt. Sowohl die Daten als auch der Zustand des Programmes können durch Befehle verändert werden. Ein Befehl besteht aus dem Identifizierungscode und den Daten /z.B. eine Phase wird durch den Code der Linienbreite/, des Linientypes - welche Seite ist das -, durch den Typ und Koordinaten der Endpunkte definiert. Zu den, den Zustand des Programmes ändernden Steuerung-Befehlen gehören keine Daten.

Voraussetzungen den Befehlen gegenüber sind wie folgt:

- eine Korrektur soll durch Befehlen von einer Minimalzahl durchgeführt werden,
- die Befehle sollen leicht zu merken sein,
- und von ergonomischem Standpunkt aus sollen sie einfach sein.

Als Input ist es weder von Lernzwecken, noch vom ergonomischen Standpunkt richtig, verschiedene Mittel zu gebrauchen, aber mit einem Mittel ist die Angabe von Daten verschiedener Typen schwerfällig. Wir haben als Haupt-Eingabemittel den Lichtgriffel gewählt. Die Befehle sind durch Hinweisung auf die Elemente eines Menüs auszuwählen. Der Vorteil dieser Methode besteht darin, dass es für den Operator genügt, die Befehlscode zu verstehen, und auf diesem Grund kann eine Wahl getroffen werden, und nur so ein Befehl gegeben werden, der im aktuellen Zustand des Systemes gültig ist. Auch ein Teil der Daten kann durch den Lichtgriffel angegeben werden, und zwar durch Identifizierung der Elemente der dargestellten Verdrahtungszeichnung. Die Angabe bestimmter Daten benötigt die Bezeichnung eines Punktes oder Feldes der Zeichnung. Zu diesem Zweck kann man das durch den Rollkugel bewegte Markierungszeichen /Fadenkreuz oder Rahmen/ gebrauchen.

Das Output ist graphisch und alfanumerisch. Das Programm - nebst Darstellung der Verdrahtungszeichnung - berichtet uns auch über seinen eigenen Zustand. Darauf weist sonst auch das aktuelle Menü an. Das Programm erteilt also drei verschiedene Informationen, und dementsprechend ist auch der Bildschirm auf drei Teile geteilt /Abbildung 2/.

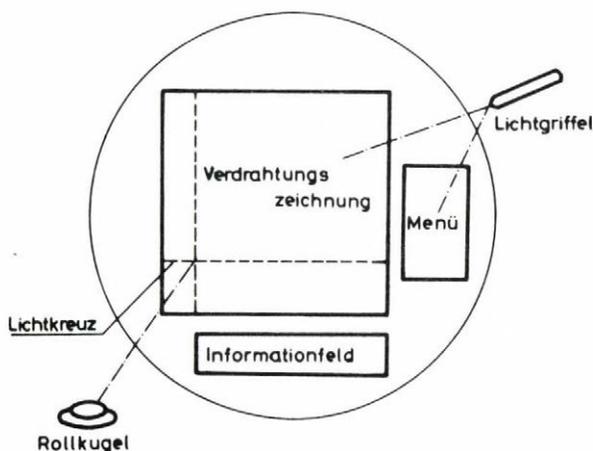


Abb.2.

Diese sind wie folgt:

- Bildfeld /Verdrahtungszeichnung/
- Menügebiet /Menü/
- Informationsfeld /Daten, Zustandsanzeiger/.

Der Modifikationsegment - ebenfalls wie das Programm und wegen der gleichen Gründe - besteht aus Overlay-Segmenten.

4.2. Teile des Modifikationsegmentes

Die Modifikation der Leiterabbildung ist funktionell in vier Teile zu zerlegen:

- Darstellung
- Vergrößerung
- Streichung
- Einschaltung.

Neben diesen, die grundlegenden Funktionen auszuführenden Segmenten enthält das Programm weitere, die Arbeit des Operators helfenden Module, so wie:

- Gehäuse-Einschaltung
- Linienbreite-Modifikation
- Aushub - Löschen
- Informationsabfrage.

Von den einzelnen Funktionen kann man die eben benötigte von dem sogenannten Hauptmenü auswählen. Wenn es gemacht ist, verschwindet das Hauptmenü, und an seiner Stelle wird das Menü der gegebenen Funktion sichtbar. Falls wir eine andere Arbeit ausführen wollen, rufen wir das Hauptmenü auf, und davon können wir wieder wählen /Abbildung 3./.

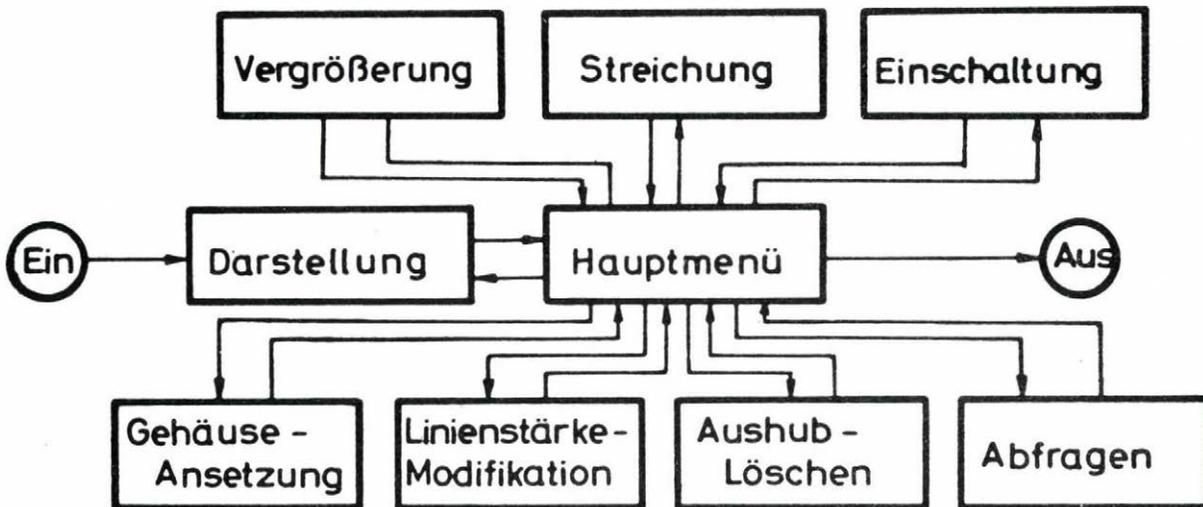


Abb.3.

Der Darstellung-Segment läuft immer als erster in einem Korrekturzyklus. Wegen der Bildschirmgröße und den Speichergrenzen können wir in einem Zyklus bis zum nächsten Ruf des Darstellung-Segmentes auf irgendeinem, 15 x 15 cm grossen Teilgebiet der Platte Korrekturen durchführen. Nach Bestimmung dieses Gebietes wird das Modell der im gegebenen Feld befindlichen Verdrahtungszeichnung von der Magnetplatte in den Speicher gelangen, ihr Bild dagegen - nach Darstellung des Bildprogrammes - erscheint am graphischen Display in einem Masse 1:1. Zur gleichen Zeit wird auch das Hauptmenü sichtbar.

Der Vergrößerung-Segment dient einer Vergrößerung im Verhältnis 2:1 oder 4:1 von irgendwelchem Teil des am graphischen Display dargestellten Druckbildes, der sich in dem, mit dem Rollkugel beweglichen symmetrischen Rahmen befindet. Es besteht die Möglichkeit, das vergrößerte Bild zu "schwemmen", da ist die Zeichnung unter dem festgesetzten Fenster mit dem Rollkugel zu bewegen.

Der Verschiebung-Segment ermöglicht uns, die Leitungen streichen zu können. Der Operator wählt zuerst vom Menü den gewünschten Typ /Punkt oder Linie an Stückseite oder an der Lötung-Seite/, dann zeigt mit dem Lichtgriffel auf das zu streichende Bildelement, das danach vom Bild gestrichen wird, aber es bleibt noch eine Weile in dem Modell, das im Speicher gespeichert ist /das heisst provisorische Streichung/. Nach einer Leitung können weitere gestrichen werden. Diese können entweder wieder eingeschaltet werden oder die Streichung der letzten Leitung kann definitiv gemacht werden. In diesem letzten Fall wird die Leitung auch vom im Speicher befindlichen Modell gestrichen, zur gleichen Zeit erscheint das diese Leitung enthaltende Potential mit einer immer grösseren Intensität /Aushub/, und die anderen, provisorisch gestrichenen Elemente werden wieder sichtbar. Diese Durchführung der Streichung macht es uns möglich, dass sogar im Falle einer dichten Zeichnung oder wenn es sich um sich deckenden Punkten oder Linien handelt, das gewünschte Element - und nur dies - gestrichen wird.

Der Einschaltung-Segment dient zur Einlegung neuer Leitungen. Auch die Einschaltung wird vom Bildelement zu Bildelement ausgeführt, der Typ muss ebenso definiert werden wie bei Streichung. Die Stelle der Punkten kann mit dem, durch den Rollkugel beweglichen Lichtkreuz eingestellt werden. /Die aktuellen Koordinaten sind im Informationsfeld zu sehen./

Bei Erzielung der gewünschten Position kann der entsprechende punkt-technologische Befehl vom Menü ausgewählt werden. Die Linienbreite kann auf zwei Arten angegeben werden:

- mit dem Lichtgriffel bezeichnen wir eine Linie, die die gewünschte Breite besitzt;
- den neuen Wert schreiben wir mit der alfanumerischen Tastatur ab.

Während der Einschaltung wird sowohl das Bildprogramm, als auch das im Speicher befindliche Modell erweitert.

Ein jedes neues Bildelement erscheint ausgehoben. Wenn eine neue Linie oder ein neuer Punkt zum früheren näher kommt, als es das von der Technologie bestimmte Minimum erlaubt, erscheint eine Mitteilung im Informationsfeld, und das betroffene Potential wird hell. Der Vorteil der veränderten Potentiale ist, dass bei Darstellung eines neuen Zeichenteiles sehbar wird, welche Potentiale früher korrigiert geworden sind.

Der Gehäuse-Ansetzungssegment ist der Einschaltung ähnlich, mit Hilfe der obigen können alle IC-Pins in einem Schritt generiert werden.

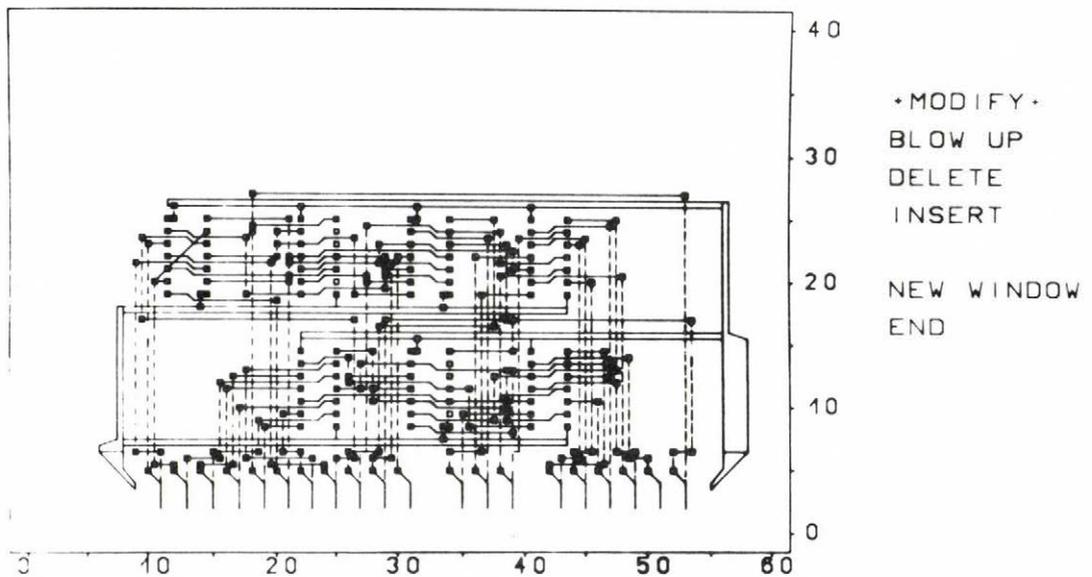
Mit dem linienbreite-modifizierenden Segment kann der mit Lichtgriffel festgesetzten Linie ein neuer Linienbreitenwert zugeordnet werden.

Der Aushub - Löschen - Segment gewährleistet eine Hilfe bei der Kontrolle der Verdrahtungszeichnung. Die Elemente eines, zu einem - mit Lichtgriffel festgesetzten - Bildelement gehörenden Potential oder Linien, die eine gleiche Linienbreite haben, wie eine - ebenso mit Lichtgriffel bestimmte - Linie, können ausgehoben oder zurückgeschaltet werden.

Irgendwelche Seite der Platte kann ausgeschaltet und wieder dargestellt werden, die Teilstückseite kann von der Lötung-Seite fortgerückt werden.

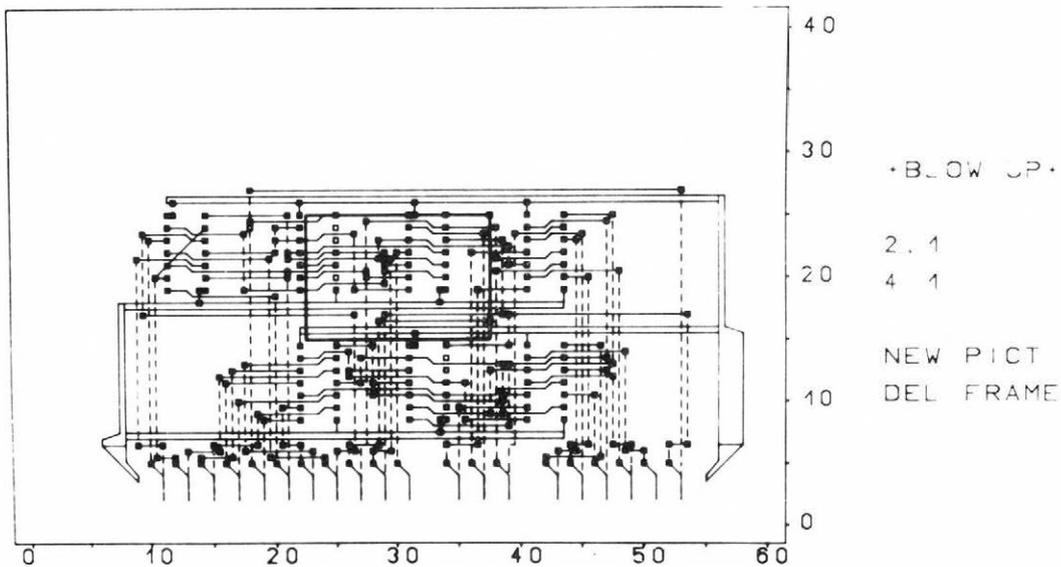
Der Abfragensegment schreibt alle Charakterzeichen irgendwelcher, mit einem Lichtgriffel bezeichneten Leitung im Informationsfeld heraus.

Und jetzt sehen Sie sich bitte an, wie man mit dem Programm Korrekturen durchführen kann!

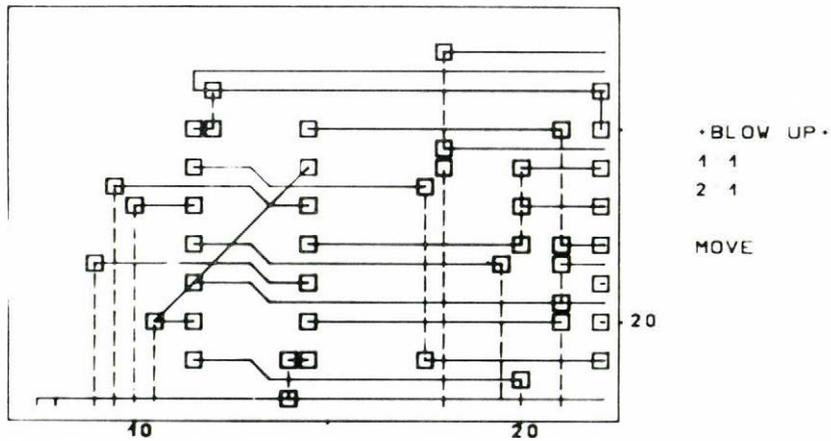


Am Bild sehen Sie die Verdrahtungszeichnung einer Platte im Mass 1:1. Auf der einen Seite sind die Drähte mit vollen, auf der anderen mit unterbrochenen Linien bezeichnet. Die Linie in dem Ecke links oben ist fehlerhaft, die wir korrigieren wollen.

Zuerst vergrössern wir den auszubessernden Teil. In dem Hauptmenü weisen wir auf den Befehl "Vergrößerung" hin, demnach der entsprechende Segment im Rechner belastet wird, und gleichzeitig erscheint sein eigenes Menü.

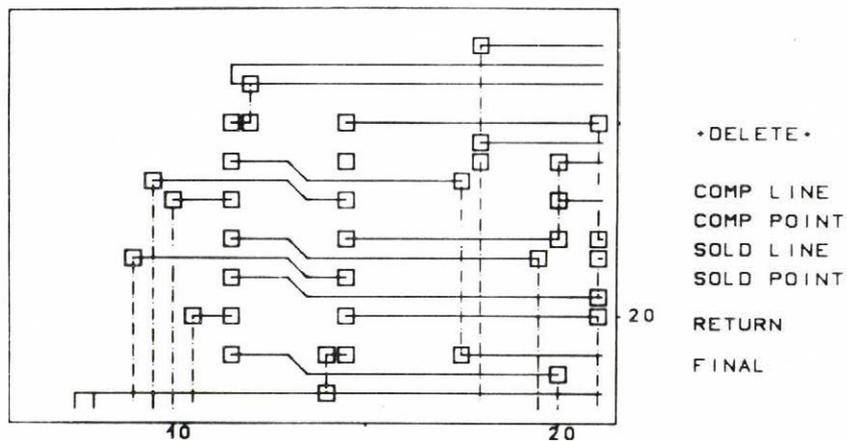


Hier zeigen wir auf die gewünschte Vergrößerung hin, da erscheint der mit dem Rollkugel bewegliche Rahmen, und damit können wir den zu vergrößernden Bildteil bezeichnen. Auf einen neuen Befehl erscheint das vergrösserte Bild.



4:1

Jetzt ist schon die fehlerhafte Linie gut zu sehen. Diese muss gestrichen werden. Wir kehren zum Hauptmenü zurück, dann rufen wir den Streichungsegment auf. Nachdem wir da den Typ des zu streichenden Elementes bestimmt haben, streichen wir die fehlerhafte Linie mit Hilfe des Lichtgriffels.

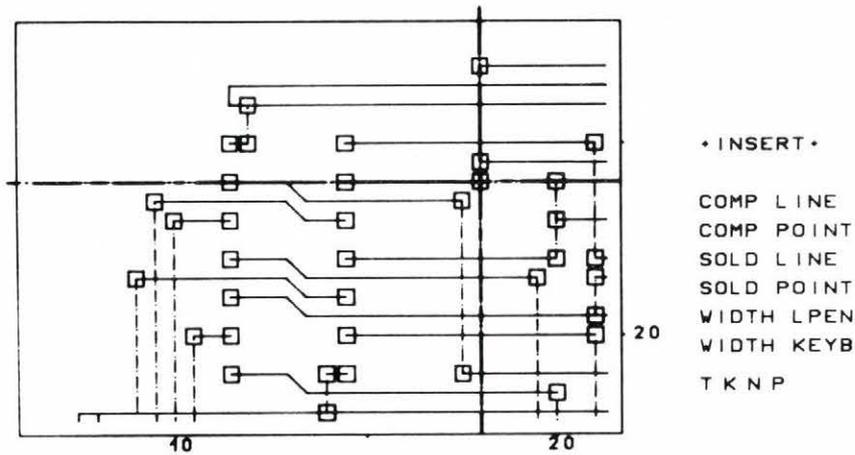
4:1
0.45

SOLD LINE

T: 14.5, 24.0.
T: 10.5, 20.0.

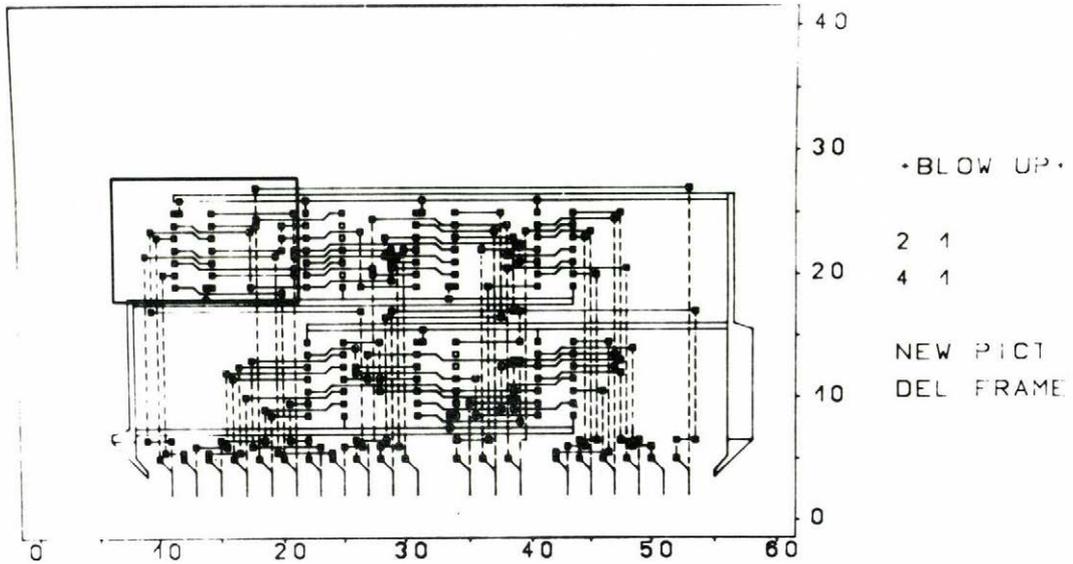
Im Informationsfeld erscheint die geometrisch-technologische Beschreibung des gestrichenen Bildelementes.

Die nächste Aufgabe ist die richtige Einlegung des Drahtes. Wir kehren wieder zum Hauptmenü zurück, dann rufen wir den Einschaltungsegment ein.



4 : 1	18 . 0	24 . 0	SOLD LINE
0 . 45	T : 14 . 5	24 . 0	SHORT CIR ?
	T : 18 . 0	24 . 0	

Zuerst bestimmen wir den Typ der einzuschaltenden Leitung, dann legen wir die Stelle der Punkte mit dem durch den Rollkugel beweglichen Lichtkreuz fest. Von der im unteren Teil des Menüs befindlichen Tabelle wählen wir den Typ des Punktes aus. Dann erscheint die eingeschaltete Linie am Bildschirm, während im Informationsfeld die punkt-technologische Beschreibung zu sehen ist.



Nach Beendigung der Korrektur kehren wir zu der Zeichnung im Mass 1:1 zurück, die ganze Platte ist sichtbar, jetzt schon korrigiert.

Das Programm ist auf Assembly-Sprache des ungarischen Kleinrechners TPA-70 geschrieben. Es verwendet das graphische Grundsoftware-Standardprogramm, das das Generieren von bildlichen Programmen und die Behandlung der Peripherien des graphischen Displays im grossen Masse vereinfacht und dies auf logischem Niveau ermöglicht.

Unser Programm ist seit August 1976 verwandt. Erfahrungsgemäss beansprucht die Verwendung dieses Programmes ungefähr eine achteil der Zeit der mit der früheren Methode /Editierung der geometrisch-technologischen Beschreibung/ durchgeführten Korrektur.

G. Déri
Forschungsinstitut für Rechen-
technik und Automatisierung
der Ungarischen Akademie der
Wissenschaften

INTERAKTIVE KONSTRUKTION MIT RECHENTECHNIK IM MASCHINENBAU

1. Einleitung

Die Konstruktionsarbeit im Maschinenbau besteht aus vielen Rutineaufgaben. Auf vielen Gebieten wurden schon Schritte gemacht, um diese Aufgaben mit Hilfe einer Rechenmaschine zu lösen, und damit die schnelle Durchführung und die Erleichterung der Konstruktion erreichen zu können. Von den verschiedenen Systemen haben sich in den einzelnen Phasen die interaktiven Systeme bewährt, da mittels der obigen die Rutineaufgaben in der Konstruktion, die komplizierten Rechnungen durch einen Rechner ausgeführt werden konnten, während die Kontrolle und die intuitiven Entscheidungen auch weiterhin in Händen des Konstrukteurs blieben.

So ein interaktives System wurde in unserem Institut ins Leben gerufen, um die Konstruktionsarbeit im Maschinenbau erleichtern zu können. Das System arbeitet mit Hilfe eines graphischen Displays. Die angewandte Hardware-Konfiguration ist an der ersten Abbildung zu sehen.

Das ist eine Kleinrechner-Konfiguration, die sich auf den ungarischen Kleinrechner TPA-70 gründet. Von den Peripherien des Rechners möchte ich das graphische Display GD`71 hervorheben, da es auch die Rolle eines interaktiven Anschlussgerätes spielt.

Zu diesem Zweck wird das Gerät durch die folgenden Mittel geeignet:

- Rollkugel
- Lichtgriffel
- alfanumerische Tastatur
- funktionelle Tastatur.

Dieses Konstruktionssystem wird Ihnen jetzt in erster Linie als eine Möglichkeit für Anwendung der digitalen Graphik vorgestellt. Deshalb werde ich im weiteren zuerst über den allgemeinen Aufbau des Konstruktionssystemes im Maschinenbau, dann über das angewandte geometrische Modell und am Ende über das Dialogsystem sprechen.

2. Allgemeine Auseinandersetzung des Konstruktionssystemes

Wie es an der Abbildung 2.1. zu sehen ist, besteht das Konstruktionssystem aus drei Hauptteilen. Diese sind die folgenden:

- geometrisches System
- technologisches System
- Dokumentationssystem

Das geometrische System ist ein allgemeines System, das für Konstruktionsaufgaben von zwei Dimensionen zu gebrauchen ist. Das technologische und Dokumentationssystem enthält allgemeine Grundprinzipien, aber beide sind für eine früher geplante Konstruktionsaufgabe bestimmt. In diesem Fall bedeutet es die Konstruktion von Presswerkzeugen.

Prüfen wir jetzt die einzelnen Teilen:

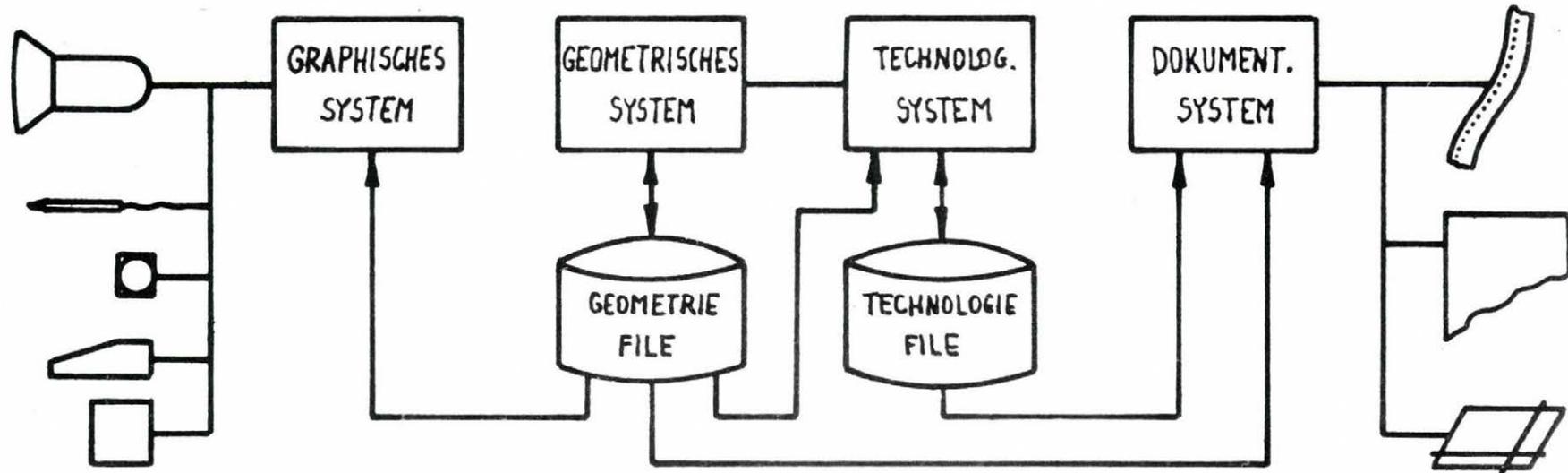


ABB. 2.1.

Die Aufgabe des geometrischen Systemes ist die Eingabe von den zur Konstruktion benötigten geometrischen Informationen in das Konstruktionssystem und ihre Darstellung.

Es ermöglicht uns, Transformationen und auch Manipulationen auf den Abbildungen durchführen zu können.

Die Darstellung erfolgt durch das graphische System aufgrund eines geometrischen Modelles.

Die Zeichnung einer endgültigen Abbildung wird in zwei Schritten gemacht.

Im ersten Schritt entwerfen wir mit Hilfe der sogenannten endlosen geometrischen Elemente /Punkt, Gerade, Kreis/ die gewünschte Zeichnung.

In der zweiten Phase wird die Abbildung aufgrund der Konstruktionslinien durch Nachziehen der zu bestimmenden Kontur eindeutig gemacht. Auf diesen nachgezogenen Konturen können verschiedene geometrische Manipulationen durchgeführt, transformiert, zusammengezogen, weggelassen werden.

Die Lösung dieser Aufgabe erfolgt auch durch das geometrische System.

Das geometrische Modell der fertigmachten Zeichnungen gelangt ins Geometrie-File. Von hier kann es später wieder aufgerufen werden, um weitere Bau- oder andere Konstruktionsaufgaben vornehmen zu können.

Zusammengefasst, das geometrische System erfüllt die folgenden Aufgaben:

- Entwurfstätigkeit
- Darstellung einfacher Konturen

- Transformation der Konturen. Das kann
 1. Verschiebung
 2. Drehung
 3. Spiegelung sein.
- Zusammenziehen der Konturen
- Bemessung der Zeichnungen
- Korrektur und Modifikation der bemessenen Zeichnungen.

Das technologische System führt alle zur Konstruktions-tätigkeit nötigen Rechnungen, Kontrolle, Datenbehandlung, eventuelle Konstruktionsentscheidungen usw. durch.

Vom Funktionsstandpunkt kann es auf drei Teile aufgelöst werden:

In dem ersten Teil - von den Daten des herzustellenden Arbeitstückes ausgegangen - legt es die folgenden fest:

- die Erzeugungstätigkeit
- die wichtigsten, zu den einzelnen Grundrechnungsarten gehörenden Parameter
- die zur Herstellung brauchbaren Normalwerkzeugen
- den Typ der Bearbeitungsmaschine.

In dem zweiten Teil werden die Werkzeuge, bzw. ihre Ersatzteile konstruiert. In dieser Phase werden auch die Werkzeichnungen der Ersatzteile fertiggemacht.

In dem dritten Teil geschieht die produktionstechnologische Konstruktion der herzustellenden Ersatzteile.

Das technologische System verwendet die Daten der Geometrie- und Technologie-Dateien für seine Funktion.

Von den entworfenen Werkzeugen müssen Zeichnungen, Produktionsbefehlen, usw., also Konstruktionsdokumentation gemacht werden. Diese Tätigkeit wird durch das Dokumentationssystem vollendet. Das Dokumentationssystem arbeitet aufgrund der an den Technologie- und Geometrie-Files befindlichen Daten. Seine Hauptaufgaben sind wie folgt:

1. Die Ausarbeitung der im Laufe der Konstruktion entstandenen Zeichnungen. Das wird mit Hilfe eines Zeichengerätes gemacht, das mit dem Rechner im On-line Betriebsart arbeitet.
2. Ausfertigung der die Rechnungsergebnisse enthaltenden Listen, Produktions- und Montierbefehle, unter anderem.
3. Falls die Verfertigung der entworfenen Ersatzteile an einer NC-Maschine verrichtet wird, die Ausführung der Steuerlochstreifen.

3. Das geometrische Modell

Das grundlegende Mittel der Konstruktion im Maschinenbau ist die Zeichnung. Um diese Zeichnungen am graphischen Display darstellen zu können, ist ein zweckmässiges, leicht gebrauchsfähiges geometrisches Modell erwünscht.

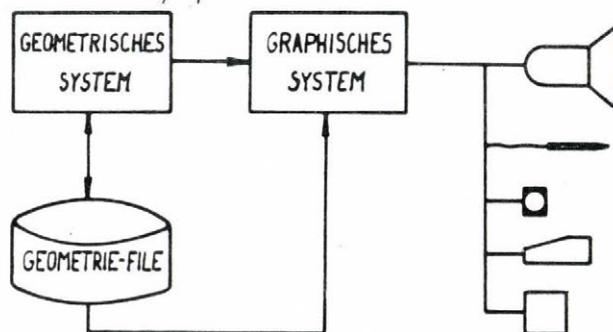


ABB. 3.1.

Auf der Abbildung 3.1. sind von dem allgemeinen Schema des Konstruktionssystemes die auf die Geometrie bezügliche Teile sichtbar. Durch Vereinfachung der Abbildung kann so eine Skizze entworfen werden, auf der die Funktion des Systemes gut betrachtet werden kann /3.2./.

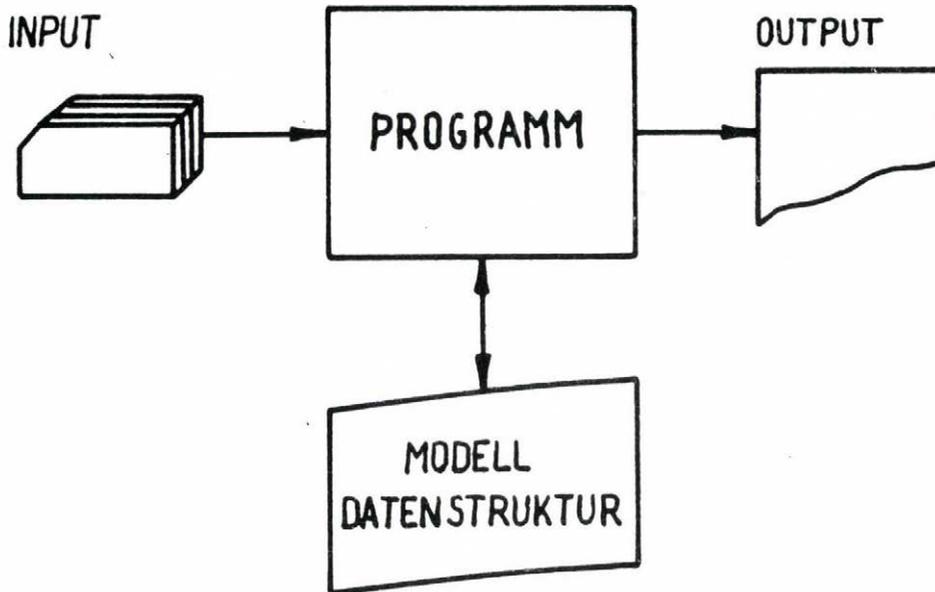


ABB. 3.2.

Mit dem Wort "Programm" bezeichnen wir das geometrische und graphische System. Dieses Programm /wie ein jedes rechnergesteuerte Programm/ hat Input und Output. Diesen entspricht in unserem Fall das graphische Display. Das Programm arbeitet auf eine Modell Datenstruktur. Das ist am vorigen Bild mit Geometrie-File bezeichnet.

Ich möchte jetzt diese Modell Datenstruktur, das geometrische Modell eingehend bekanntmachen.

Die Voraussetzungen dem geometrischen Modell gegenüber sind wie folgt:

1. Der Typ und Zustand der geometrischen Elemente müssen eindeutig festgelegt werden.
2. Das Modell soll die zur Darstellung, bzw. zum Zusammenziehen, zur Transformation der Bildelemente benötigten Daten enthalten.

Auf Grund der obenerwähnten unterscheidet man die folgenden Bildelemente:

- Punkt
- Gerade
- Kreis
- Einfache Kontur
- Komplex-Bildelement
- Transformiertes Bildelement

Zu jedem Bildelement gehört ein sogenanntes Statuswort /ST/. Das ist für das Bildelement und seinen Zustand eindeutig charakteristisch.

Auf das Statuswort folgen die zum Bildelement gehörenden Daten. So bringen wir die an der Abbildung 3.3. sichtbaren Datenfelde zustande.

PUNKT
 $P(X,Y)$

ST
X
Y

GERADE
 $AX+BY+C=0$

ST
A
B
C

KREIS
 $(X-U)^2+(Y-V)^2=R^2$

ST
U
V
R

EINFACHE
KONTUR

ST
ST1
X
Y
ST1
X
Y
⋮
END-SIGNAL

KOMPLEX
BILDELEMENT

ST
POINTER
POINTER
POINTER
⋮
END-SIGNAL

TRANSFORM.
BILDELEMENT

ST
POINTER
X
Y
Z

ABB. 3.3

Es lohnt sich, über die einfache Kontur, bzw. das transformierte sowie das Komplex-Bildelement separat zu sprechen.

Die einfache Kontur besteht aus Konturelementen. Ein jedes Konturelement wird mit einem Statuswort identifiziert. /Auf der Abbildung ist es mit ST1 bezeichnet./ Das ist auch für den Zustand des Konturelementes charakteristisch.

Alle Konturen beginnen mit einem Beginnpunkt, den die untenaufgeführten Konturelemente folgen:

- Gerade
- Kreis
- Konturelement, das kann auch
 - Punkt
 - Gerade
 - Kreis sein.

Nach den entsprechenden Statuswörtern kommen die Koordinaten des Endpunktes des aktuellen Konturelementes ins Modell.

Ein transformiertes Bildelement entsteht durch die Transformation einer vorher definierten Kontur. Sein Statuswort enthält den Charakter der Transformation /Verschiebung, Drehung, Spiegelung/. Auf das folgt ein Pointer, der auf das Statuswort des zu transformierenden Bildelementes weist. Auf den Pointer folgen die Daten der Transformation.

Das zu transformierende Bildelement kann ausser der einfachen Kontur noch transformiert oder komplex sein.

Ein Komplex-Bildelement kommt zustande, wenn wir uns auf mehrere Bildelemente gleichzeitig beziehen wollen. Auf das Statuswort des Komplex-Bildelementes folgen Pointer, die auf die Statuswörter der zusammenzuziehenden Bildelemente zeigen.

Die Behandlung eines geometrischen Modelles erfolgt durch ein Subroutine-Package. Mit Hilfe dieses können neue Bildelemente ins Leben gerufen werden und verschiedene Manipulationen am Bildelement durchgeführt werden.

- Zum Beispiel: - Ein-/Ausschaltung } der Bildelemente
 - Geblimmer
 - Empfindlichkeit oder Unempfindlichkeit für die Wirkung des Lichtgriffels der Bildelement
 - Verlassung der Bildelement, usw.

Es ist zu sehen, dass es uns gelungen ist, eine ständige und eindeutige Identität zwischen Bild und Modell zu verwirklichen, und dass dieses Modell für Beschreibung verschiedener Zeichnungen geeignet ist.

4. Das Dialog-System

Das Dialogsystem ist von der gleichen Bedeutung wie das Konstruktionssystem im Maschinenbau. Dies sichert die Verbindung zwischen Mann und Maschine, das heisst dem Konstrukteur und dem Konstruktionssystem.

Dem Dialog-System ist ein eigener Abschnitt gewidmet, weil es nicht direkt ins Konstruktionssystem des Maschinenbaues gehört.

Hier handelt es sich um ein allgemeines System. Mit Hilfe dieses Systemes können nicht nur die Konstruktion im Maschinenbau, sondern auch andere interaktiven Aufgaben gelöst werden. Der Aufbau ist an der Abbildung 4.1. zu sehen.

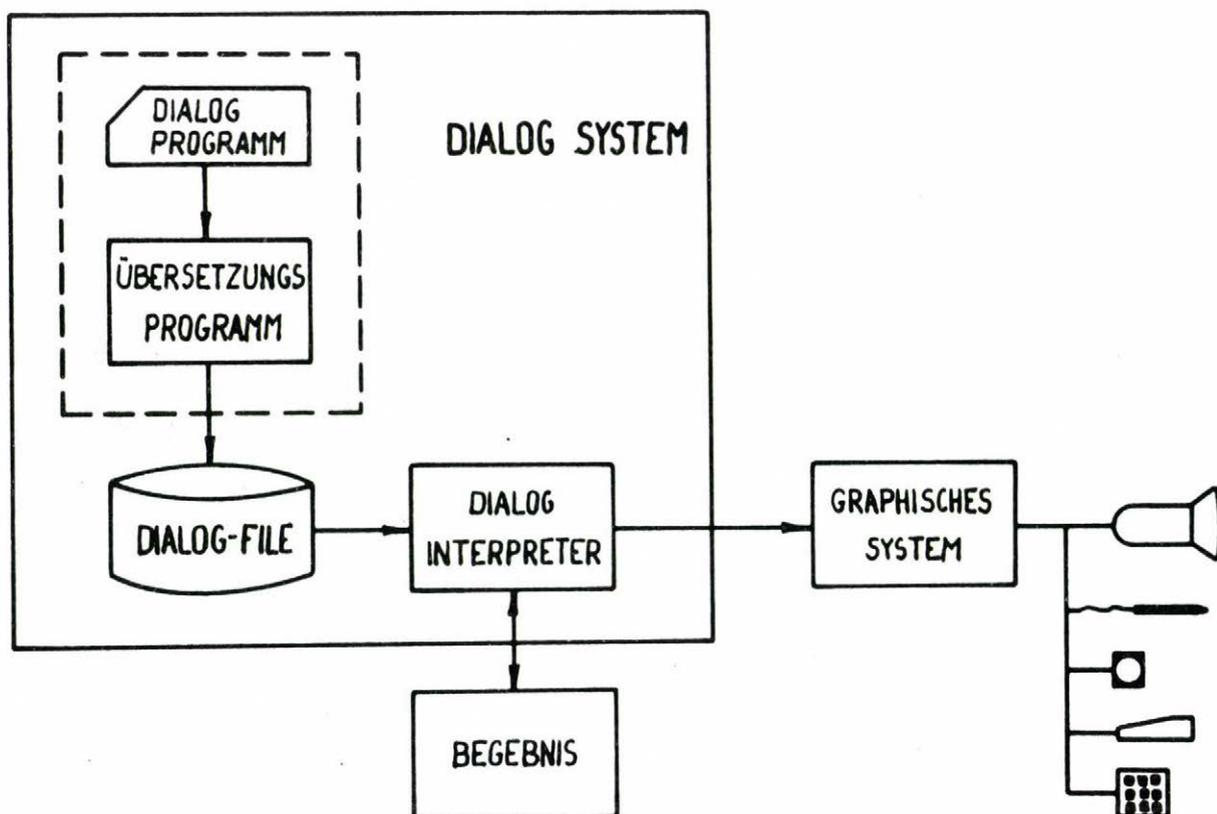


ABB. 4.1.

Zur Bekanntmachung seiner Funktion müssen wir vorerst über einige Begriffe klar sein.

Um eine Aufgabe erfüllen zu können, benötigen wir verschiedene Daten. Das Dialogsystem ruft diese Daten vom Verwender mit einem Frage- und Antwortverfahren auf. Im Falle von fehlerfreien Antworten leitet es die Daten zu dem die Aufgabe /das Begebnis/ auszuführenden System weiter.

Im Falle des Konstruktionssystemes im Maschinenbau können diese ausübende Systeme die geometrische, technologische und Dokumentationssysteme sein.

Zum Aufruf der entsprechenden Informationen müssen also Dialoge herausgestaltet werden. Davon kommt, dass sich diese Dialoge eng der zu lösenden Aufgaben, Begebnisse anschliessen sollen. Deshalb, zusammen mit den die Aufgaben lösenden Prog-

rammteilen müssen wir auch die einzelnen Dialogbilder konstruieren.

Der Dialog enthält also alle Informationen, die zum Dialog zwischen Verwendungssystem und dem Operator benötigt sind.

Der Dialog besteht aus Teildialogen, die in Aktionen geteilt werden können.

Die Aktion ist die kleinste selbständige Einheit des Dialogs, die den Aufruf eines Datenteiles festlegt.

Der Aufruf aller zum Ergebnis benötigten Informationen wird Teildialog genannt.

Der Teildialog besteht also aus Serien von Aktionen. Grundslegend unterscheiden wir zwischen zwei Typen des Teildialoges: der eine ist von Menü-Art, der andere von Informationsaufruf-Typ.

Um diese verstehen zu können, prüfen wir die folgende Abbildung 4.2.

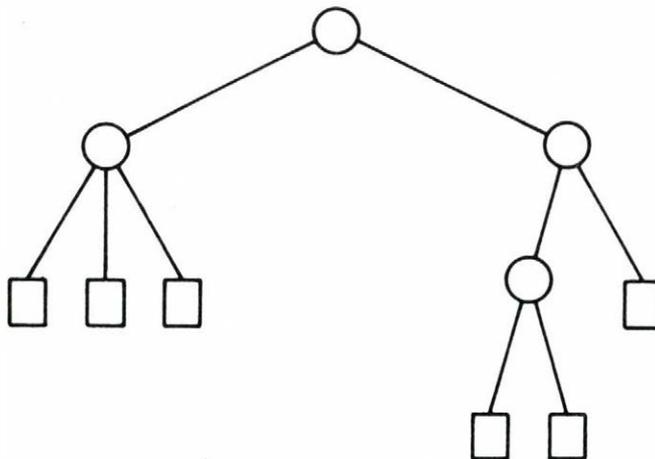


ABB. 4. 2.

Auf dieser Abbildung bezeichnen die Kreise Teildialoge, die von Menü-Art sind, während die Quadrate für Teildialoge von Antwort-Art sind. Es ist also zu sehen, dass die Teildialoge von Menü-Art die entsprechende Translationsrichtung zum Datenaufruf bestimmen, während die Teildialoge von Antwort-Art den effektiven Datenaufruf ausführen.

Kehren wir jetzt zur Bekanntmachung des Aufbaues des Dialog-Systemes zurück /Abbildung 4.1./.

Zur Erleichterung der Beschreibung der Dialoge haben wir eine Dialogsprache entworfen.

Auf der Dialogsprache geschriebene Dialoge werden Dialogprogramme genannt.

Vom Dialogprogramm wird das Dialog-File durch ein Übersetzungsprogramm entwickelt.

Aufgrund der Daten des Dialog-Files arbeitet der Dialog-Interpreter. Dieser führt die Ausgabe von Fragen, Informationen, Fehlertexten usw. und die Eingabe der Antworten des Operators durch. Er analysiert, bewertet die eingegangenen Antworten. Falls er alle zu einem Ergebnis benötigten Daten schon gespeichert hat, geht der Lauf des Programmes am Ergebnis weiter. Nach Durchführung des Ergebnisses wird die Steuerung wieder zum Dialog-Interpreter geführt.

Der Dialog-Interpreter verwendet das graphische Display als interaktives Anschlussgerät, dem er durch ein graphisches System angeschlossen ist. Der Aufruf von Antworten kann in alfanumerischer oder graphischer Form erfolgen.

Sehen Sie sich jetzt ein einfaches Beispiel für Beschreibung eines Dialoges an. /Abbildung 4.3./

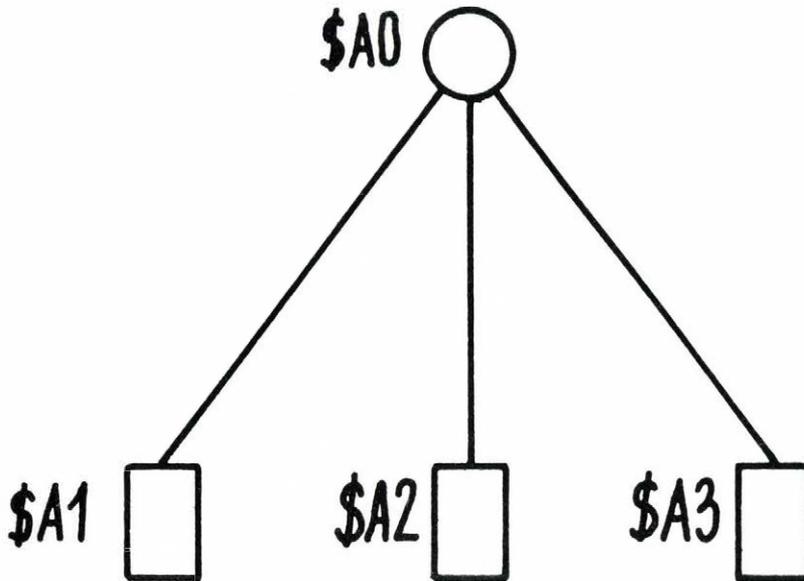


ABB. 4.3.

Auf der Abbildung ist zu sehen, dass ein jeder Teildialog durch eine Marke identifiziert ist. Die Identifizierungen und Referenzen erfolgen auch in der Sprache aufgrund dieser Marken.

Nehmen wir als Aufgabe den Aufruf der Koordinaten eines Punktes.

Kommen wir zu diesem Punkt durch ein Menü hin.

In unserem Beispiel erfolgt also der Aufruf der Koordinaten des Punktes in zwei Schritten, aufgrund der Teildialoge \$A0 und \$A1. Sehen wir uns jetzt die zu diesen gehörenden Dialogbilder an /Abbildung 4.4./.

\$A0	PUNKT	<	>	\$A1
	GERADE	<	>	\$A2
	KREIS	<	>	\$A3

\$A1	COORDINATEN	X:	<	>
		Y:	<	>

ABB. 4.4

Es ist sichtbar, dass wir im Teildialog \$A0 von drei Fällen /Punkt, Gerade, Kreis/ wählen können, und allen diesen gehört eine Teildialogmarke. Das ist ein Dialog von Menü-Art.

Der Teildialog \$A1 liest schon die Koordinaten des Punktes ein. Dieser ist also von Antwort-Art.

Das zu den Bildern gehörende Dialogprogramm ist an der nächsten Abbildung /4.5./ zu sehen.

\$A0	V	\$A0
	QAW1 1	2 PUNKT
	QAW1 2	2 GERADE
	QAW1 3	2 KREIS
	J CAW1	1 \$A1
	J CAW1	2 \$A2
	J CAW1	3 \$A3
\$A1	V	\$A0
	QAW1 1	1 KOORDINATEN X:
	A NAW1	1
K1	V	\$A1
	QAW1 2	14 Y:
	A NAW1	2

ABB. 4.5.

Gehen wir vom Schritt zu Schritt durch das Programm! Die erste Linie identifiziert den Teildialog \$A0. Die zweite, dritte, vierte Linien beginnen mit dem Buchstabe Q, der den Typ des Befehles identifiziert. Auf Einfluss des Befehles vom Q-Typ wird der im letzten Feld, im Textfeld befindliche Text auf die von den anderen Parametern des Befehles bestimmte Stelle am Bildschirm herausgeschrieben.

Auf dem graphischen Display können gleichzeitig alfanumerische Segmente von einer beliebigen Zahl stattfinden. Das zweite Feld teilt uns mit, in welches alfanumerische Fenster die Ausgabe durchgeführt werden muss. Die nächsten zwei Felder geben die Position des Textes /Linie, Kolonne/ innerhalb des alfanumerischen Fensters an.

Die Befehle vom J-Typ weisen darauf hin, ob es sich um einen Teildialog von Menü-Art handelt. Mittels dieser Befehle kann geprüft werden, von welcher Linie wir die Antwort erhalten haben und dementsprechend kann der Dialog auf verschiedene Marken gerichtet werden.

Der Dialog \$A1 ist von anderen Charakterzügen. Hier folgt auf den Q-Befehl ein A. Mit Hilfe von Befehlen vom A-Typ können Antworten vom Bildschirm abgelesen werden.

Die auf den Buchstabe A folgende N Korrektur teilt uns mit, dass wir mit einer numerischen Antwort zählen können. Die nächsten zwei Felder dagegen, woher.

Im Programm befindet sich auch eine Marke Kl. Die innerhalb eines Teildialoges vorkommenden Aktionen bezeichnen wir mit Marken Kl, K2...usw.

Die Punktkoordinaten bedeuten zwei Daten, ihr Aufruf erfolgt also in zwei Schritten, im Laufe zweier Aktionen.

Mit der Bekanntmachung und ausführlicher Syntax der anderen Befehle der Sprache beschäftigen wir uns jetzt nicht.

Ich hoffe, dass dieses Beispiel Ihnen den einfachen Charakter und die Anwendbarkeit der Sprache gut vorgestellt hat.

Zusammenfassung

Schliesslich werfen wir einen Blick auf die letzte Abbildung /5.1./.

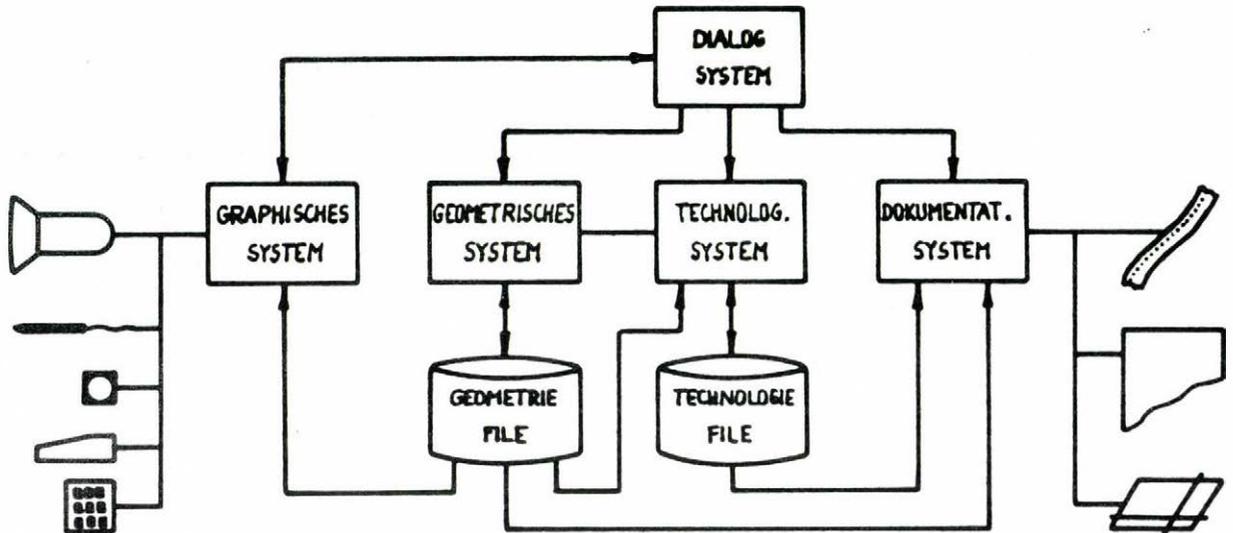


ABB. 5.4.

Auf dieser Abbildung ist neben dem Konstruktionssystem im Maschinenbau auch das Dialog-System dargestellt.

Es ist gut sichtbar, dass den interaktiven Prozess das Dialog-System steuert.

Es übergibt die gesammelten Daten den die Begebnisse vollziehenden Systemen, die ihre Aufgabe selbständige erfüllen. Das geometrische System arbeitet und zeichnet auf den

Bildschirm aufgrund des geometrischen Modelles. Das technologische System führt die technologischen Rechnungen durch, während das Dokumentationssystem das Output des Konstruktionssystemes im Maschinenbau, Zeichnungen, Listen, Lochstreifen zustande bringt.

Das Dialogsystem und das Konstruktionssystem sind mit dem graphischen Display durch das graphische System in Verbindung gestellt.

P. Verebely
Forschungsinstitut für Rechen-
technik und Automatisierung
der Ungarischen Akademie der
Wissenschaften

INTERAKTIVE KONSTRUKTION MIT RECHENTECHNIK
IM MASCHINENBAU
/Kurzreferat/

Die Konstruktionsarbeit im Maschinenbau besteht aus vielen Routineaufgaben. Auf vielen Gebieten wurden schon Schritte unternommen, um diese Aufgaben mit Hilfe einer Rechenmaschine zu lösen. Damit soll die schnelle Durchführung und die Erleichterung der Konstruktion erreicht werden. Von den verschiedenen Systemen haben sich in den einzelnen Phasen die interaktiven Systeme bewährt, mit denen die Routineaufgaben in der Konstruktion und die komplizierten Berechnungen ausgeführt werden können, während die Kontrolle und die intuitiven Entscheidungen auch weiterhin in Händen des Konstrukteurs bleiben.

So ein interaktives System wurde in unserem Institut entwickelt, um die Konstruktionsarbeit im Maschinenbau erleichtern zu können. Das System arbeitet mit Hilfe des graphischen Displays, das in der DDR mit dem KRS 4201 des VEB Kombinat Robotron als Steuerrechner gekoppelt eingesetzt wird.

Das graphische Display GD⁷¹ besteht aus folgenden Funktionsgruppen:

- Bildschirm
- Positionierkugel
- Lichtstift
- alphanumerische Tastatur
- funktionelle Tastatur.

Im weiteren werde ich zuerst über den allgemeinen Aufbau des Konstruktionssystems im Maschinenbau, dann über das angewandte geometrische Modell und am Ende über das Dialogsystem sprechen.

Das Konstruktionssystem besteht aus 4 Hauptteilen, dem

- geometrischen System
- technologischen System
- Dokumentationssystem
- Dialogsystem.

Das geometrische System ist ein allgemeines System, das für zwei-dimensionale Konstruktionsaufgaben geeignet ist. Das technologische und das Dokumentationssystem enthalten allgemeine Grundprinzipien.

Die Aufgabe des geometrischen Systems ist die Eingabe von den zur Konstruktion benötigten geometrischen Informationen in das Konstruktionssystem und ihre Darstellung.

Es ermöglicht uns, Transformationen und auch Manipulationen auf den Abbildungen durchzuführen.

Die Darstellung erfolgt durch das graphische System auf Grund eines geometrischen Modelles.

Die Zeichnung einer endgültigen Abbildung wird in zwei Schritten durchgeführt.

In ersten Schritt entwerfen wir mit Hilfe der sogenannten endlosen geometrischen Elemente - Punkt, Gerade, Kreis - die gewünschte Zeichnung.

In der zweiten Phase wird die Abbildung auf Grund der Konstruktionslinien durch Nachziehen der zu bestimmenden

Kontur eindeutig gemacht. Auf diese nachgezogenen Konturen können verschiedene geometrische Manipulationen durchgeführt, transformiert, zusammengezogen, weggelassen werden.

Vom Standpunkt eines Konstrukteurs ist es eine sehr wichtige Aufgabe, die Zeichnungen zu bemessen, zu beschriften und mit verschiedenen Ergänzungsinformationen zu versehen.

Die Lösung dieser Aufgabe erfolgt auch durch das geometrische System.

Das geometrische Modell der fertigen Zeichnungen gelangt ins Geometrie-File. Von hier kann es später wieder aufgerufen werden, um weitere Konstruktionsaufgaben vornehmen zu können.

Zusammengefasst erfüllt das geometrische System die folgenden Aufgaben:

- Entwurfstätigkeit
- Darstellung einfacher Konturen
- Transformation der Konturen. Das können Verschiebungen, Drehungen und Spiegelungen sein
- Zusammenziehen der Konturen
- Bemessung der Zeichnungen
- Korrektur und Modifikation der bemessenen Zeichnungen.

Das technologische System führt alle zur Konstruktions-tätigkeit nötigen Berechnungen, Kontrolle, Datenbehandlung, eventuelle Konstruktionsentscheidungen usw. durch.

Vom Funktionsstandpunkt kann es in drei Teile aufgelöst werden:

In dem ersten Teil - von den Daten des herzustellenden Arbeitsstückes ausgegangen- legt es die folgenden fest:

- die Erzeugnistätigkeit
- die wichtigsten, zu den einzelnen Grundrechnungsarten gehörenden Parameter
- die zur Herstellung brauchbaren Normalwerkzeuge
- den Typ der Bearbeitungsmaschine.

In dem zweiten Teil werden die Werkzeuge, beziehungsweise ihre Ersatzteile konstruiert. In dieser Phase werden auch die Werkzeichnungen der Ersatzteile fertiggestellt.

In dem dritten Teil geschieht die produktionstechnologische Konstruktion der herzustellenden Ersatzteile.

Das technologische System verwendet die Daten der Geometrie-Technologie-Files für seine Funktion.

Von den entworfenen Werkzeugen müssen Zeichnungen, Produktionsbefehle, usw. also Konstruktionsdokumentationen, hergestellt werden. Diese Tätigkeit wird durch das Dokumentationssystem vollendet. Das Dokumentationssystem arbeitet auf Grund der an den Technologie- und Geometrie-Files befindlichen Daten. Seine Hauptaufgaben sind folgende:

- 1./ Die Ausarbeitung der im Laufe der Konstruktion entstandenen Zeichnungen. Das wird mit Hilfe eines Zeichengerätes gemacht, das mit dem Rechner im On-line- oder Off-line-Betrieb arbeitet.
- 2./ Ausfertigung der die Rechenergebnisse enthaltenden Listen, Produktions- und Montagebefehle.
- 3./ Falls die entworfenen Ersatzteile an einer NC-Maschine hergestellt werden, die Ausführung der Steuerlochstreifen.

Das Dialogsystem ist von der gleichen Bedeutung wie das Konstruktionssystem im Maschinenbau. Dies sichert die Verbindung zwischen Mensch und Maschine, das heisst dem Konstrukteur und dem Konstruktionssystem.

Hier handelt es sich um ein allgemeines System. Mit Hilfe dieses Systemes können nicht nur die Konstruktion im Maschinenbau, sondern auch andere interaktive Aufgaben gelöst werden.

Um eine Aufgabe erfüllen zu können, benötigen wir verschiedene Daten. Das Dialogsystem ruft diese Daten mit einem Frage- und Antwortverfahren auf. Im Falle von fehlerfreien Antworten leitet es die Daten zu dem die Aufgabe auszuführenden System weiter.

Im Falle des Konstruktionssystemes im Maschinenbau können diese ausübenden Systeme die geometrischen, technologischen und Dokumentationssysteme sein.



Készült a SZÁMOK Reprográfiai Üzemében

400 példányban

Munkaszám: 8015

