

Das Betriebssystem der EDVA EC 1055: OS/ES

*Dr. Sylvia Lampenscherf
VEB Kombinat Robotron*

1. Übersicht

Parallel zur Entwicklung der Modelle des ESER II wurden auch die Betriebssysteme weiterentwickelt. Das der Leistungsfähigkeit der EC 1055 angemessene Betriebssystem ist das OS/ES, das beginnend mit der Ausgabe 6.1 auch den Funktionsumfang der Modelle des ESER II unterstützt. Zu den wesentlichsten Aspekten der Erweiterung des Betriebssystems OS/ES gehören neben der Unterstützung des virtuellen Speicherkonzepts eine Verbesserung der Fehlerdiagnose und damit der Zuverlässigkeit, die Unterstützung neuer peripherer Geräte und die Aufnahme neuer Funktionskomplexe die ein breiteres Anwendungsspektrum zulassen.

1.1. Funktionen

Das OS/ES bearbeitet Programme, die in einer der problemorientierten Programmiersprachen PL/1, FORTRAN, RPG, COBOL oder ALGOL oder in der maschinenorientierten Assemblersprache geschrieben sind. Damit das Betriebssystem das formulierte Problem bzw. den Auftrag übernehmen und bearbeiten kann, ist eine genaue Auftragsbeschreibung erforderlich, in der alle für den Auftrag benötigten Ressourcen, Hauptspeicherbedarf, Dateien oder Programme definiert sind. Die Aufträge werden in einer speziellen Sprache, der Jobsteuersprache, beschrieben. Das Betriebssystem nimmt den Auftrag, der aus einzelnen Unteraufträgen (Jobschritten) bestehen kann, entgegen und teilt ihn in verschiedene auszuführende Aufgaben auf. Die Aufgabe ist aus der Sicht des Betriebssystems die kleinste zu bearbeitende Einheit. Das Betriebssystem selbst funktioniert ebenfalls auf der Basis der Ausführung von Aufgaben, die als Systemaufgaben bezeichnet werden.

Mit der gleichzeitigen Bearbeitung mehrerer Aufträge wird eine möglichst bedarfsgerechte Aufteilung der Ressourcen zwischen den einzelnen Aufträgen und deren Aufgaben und außerdem eine effektive Auslastung des Gesamtsystems erzielt. Diese Arbeitsweise, bei der mehrere Aufträge parallel ausgeführt werden, wird Mehrauftragsbetrieb genannt. Es ist eine Variante des Multiprogrammbetriebs, zu dem außerdem der Mehraufgabenbetrieb zu rechnen ist. Beim Mehraufgabenbetrieb handelt es sich um eine Arbeitsweise, die durch das Bestehen von Unteraufgaben innerhalb der Ausführung eines Auftrages gekennzeichnet ist und vom Nutzer selbst organisiert wird. Die Auftragsannahme des Betriebssystems interpretiert die i

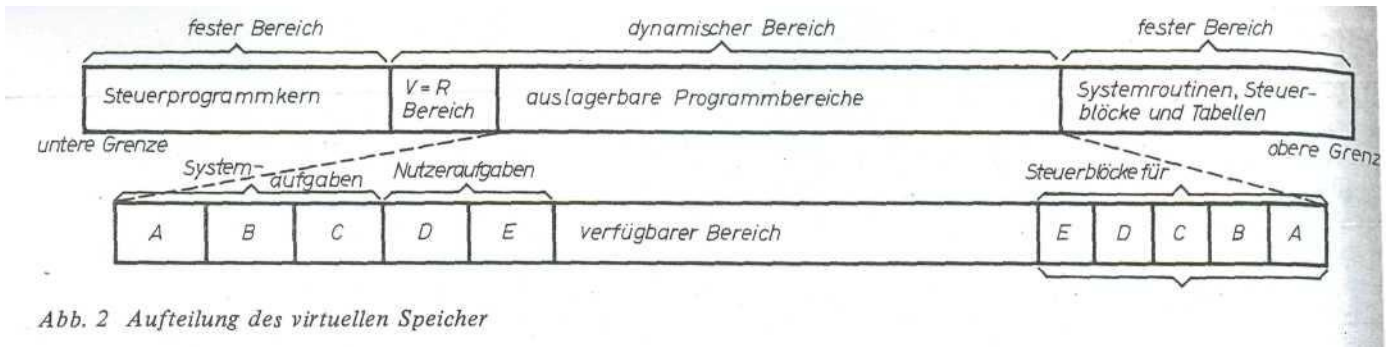


Abb. 2 Aufteilung des virtuellen Speicher

Verteilung von Ressourcen und der Behandlung von Fehler-situationen. Es besteht aus den Teilen Jobverwaltung, Daten-verwaltung, Aufgabenverwaltung sowie Teilen, die zur Unter-stützung von Programmvorbereitung und Test sowie der Diagnose dienen.

1.3. Betriebsarten

Die Betriebsart legt den grundsätzlichen organisatorischen Ablauf der Arbeit mit der DV-Anlage fest. Das Betriebssystem OS/ES kann in zwei Betriebsarten, im Stapelbetrieb und im Teilnehmerbetrieb, genutzt werden. Im Stapelbetrieb werden Aufträge ausgeführt, für die vor Bearbeitungsbeginn die Auftragsbeschreibung und alle Eingabedaten bereitstehen. Während der Verarbeitung kann der Ablauf nicht mit zusätz-lichen Eingriffen gesteuert werden. Damit werden eine kon-tinuierliche und effektive Auslastung der Anlage und geringe Verarbeitungskosten gewährleistet. Die Durchlaufzeit eines einzelnen Auftrages ist, bedingt durch den organisatorischen Ablauf, relativ hoch.

Beim Teilnehmerbetrieb ist der Nutzer direkt mit der DV-Anlage verbunden und steuert unmittelbar die Verarbeitung seines Auftrages. Eingabedaten werden an der Datenstation erzeugt und Verarbeitungsergebnisse dort empfangen. Damit kann die Durchlaufzeit eines Auftrages stark reduziert werden. Das OS/ES bietet mit den Funktionen *Dialogeingabe-system (CRJE)*¹ und *Teilnehmerunterstützung (TSO)* zwei Varianten des Teilnehmerbetriebes mit unterschiedlicher Leistungsfähigkeit an, wobei TSO höheren Ansprüchen an den Teilnehmerbetrieb gerecht wird. CRJE steht mit der Ausgabe 6.1 auch im SVS zur Verfügung; es ist für Entwicklung, Test und Wartung von Programmen geeignet. Der Dialog mit dem System wird mit einer eigenen Kommandosprache geführt. Jobs, die im Dialpg unter CRJE aufgebaut werden, werden in die Jobwarteschlange des normalen Stapelbetriebes eingeordnet und von der zentralen Job Verwaltung bearbeitet, ohne daß der Nutzer weitere Eingriffe vornehmen kann. Die Ergebnisse des Jobs werden nicht über die Systemausgabe, sondern direkt an den Nutzer der Datenstation übermittelt, damit wird die Jobdurchlaufzeit erheblich verkürzt. Mit TSO wird dem Nutzer ein gegenüber CRJE funktionell erweiterter Dialogbetrieb ermöglicht. In der Ausgabe 6.1 des OS/ES steht TSO nur im Zusammenhang mit der Steuer-programmkonfiguration MVT zur Verfügung, kann aber mit EC 1055 genutzt werden.

Der wesentliche Unterschied zwischen CRJE und TSO besteht darin, daß die Jobs unter TSO nicht in die normale Jobwarte-

schlange eingeordnet werden, sondern unmittelbar unter Kon-trolle des Nutzers ausgeführt werden. Im Gegensatz zu CRJE, das auf dem Niveau einer Systemaus-gabe arbeitet, beeinflusst TSO den Ablauf des gesamten Steuer Programms. In Abb. 3 ist der Ablauf der Steuerung mit TSO skizziert.

Die TSO-Steueraufgabe enthält systemerweiternde Funktion zur Initialisierung des TSO. Das Nachrichten-Steuerprogramm behandelt alle E/A-Operationen der unter TSO arbeitenden Datenstationen. Für jeden Bereich, in dem TSO arbeitet, gibt es eine Bereichs-Steueraufgabe, die alle zugeordneten Jobs überwacht. Mit der LOGON-LOGOFF-Verarbeitung werden Eröffnung und Abschluß einer Sitzung an einer Datenstation gesteuert. Die Steuerung wird zwischen der LOGON-Prozedur und dem Problemprogramm über ein Monitor-Programm zur Verarbeitung der TSO-Kommandos übergeben.

1.4. Mehrrechnersysteme

Das Modell EC 1055 bietet alle Voraussetzungen für Mehr-rechnersysteme verschiedenster Varianten. Die Kopplung und Informationsübertragung zwischen EC 1055 und anderen ZE des ESER, Reihen I und II, kann über die Direktsignalsteue-rung, den Kanal-Adapter oder die gemeinsame Nutzung von Direktzugriffsspeichern, Magnetbandspeichern und Datenfern Verarbeitungseinrichtungen erfolgen. Das OS/ES unterstützt in der Ausgabe 6.1 die Informationsübertragung zwischen

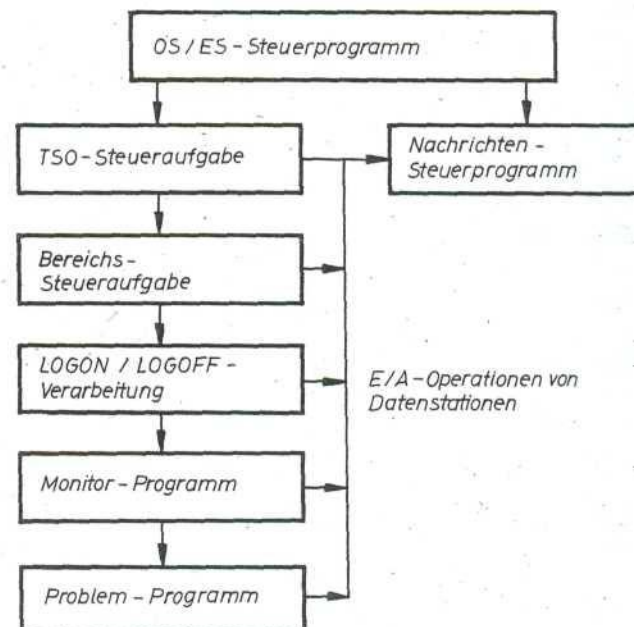


Abb. 3 Ablauf der Steuerung mit TSO

1 Vgl. auch rechen-technik/datenverarbeitung, beiheft 2/1978, S. 34: Das Jobferneingabesystem CRJE.

r

pelten Rechnern auf dem Niveau des Problemprogrammes; Mit der Ausgabe 6.2 werden Mehrrechnersysteme gesteuert- mit denen eine Verteilung der zu bearbeitenden Problemprogramme zwischen den gekoppelten DV-Anlagen optimiert werden kann. Das Mehrrechnersystem wird von einem Leitrechner zentral gesteuert, der die Verteilung von Jobs an einzelne Arbeitsrechner, die Zuordnung von E/A-reräten zu Problemprogrammen und die Steuerung der gesamten Systemeingabe und Systemausgabe übernimmt, nie angeschlossenen Arbeitsrechner sind für das Ausführen der Problemprogramme vorgesehen. Abb. 4 zeigt den prinzipiellen Aufbau eines Mehrrechnersystems, Ein Mehrrechnersystem vereinfacht für Großanwender, die einen hohen täglichen Jobdurchsatz mit unterschiedlichsten Anforderungen an Datenträgern zu bewältigen haben, die Arbeitsvorbereitung und Organisation und erhöht die Sicherheit, da der Ausfall von Ressourcen im Mehrrechnersystem leichter überbrückt werden kann, weil sie teilweise mehrfach vorhanden sind.

2. Verwaltungsfunktionen

Der zentrale Teil des Betriebssystems OS/ES, das Steuerprogramm, führt alle Verwaltungsfunktionen aus, dazu gehören [je Jobverwaltung, Datenverwaltung, Aufgabenverwaltung und die Fehlerverwaltung. Die Verbindung und der Steuerablauf zwischen den Teilen des Steuerprogramms und dem Problemprogramm sind in Abb. 5 schematisch dargestellt.

.1. Jobverwaltung

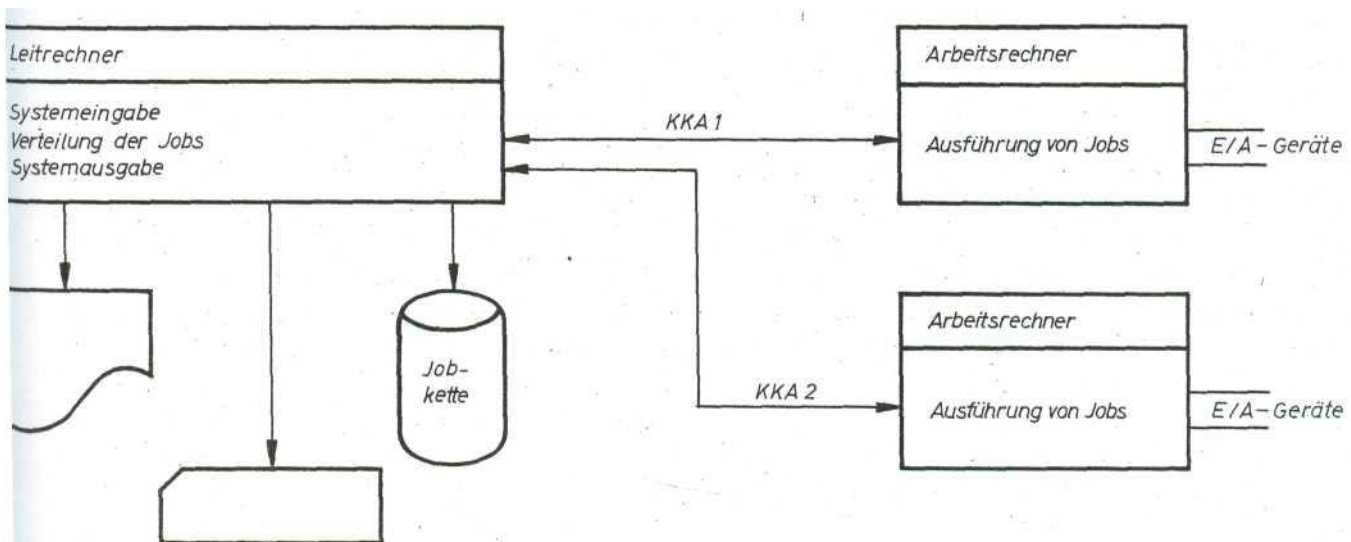
Von der Jobverwaltung werden die Aufträge entgegengenommen, interpretiert und mit dem Zuordnen von Ressourcen für die Ausführung vorbereitet. Nach der Jobausführung werden die Ergebnisse ausgegeben oder zwischengespeichert. alle dafür erforderlichen Informationen erhält das System in den vom Nutzer formulierten Jobsteueranweisungen. die wichtigsten und für jede Auftragsbeschreibung erforderlichen Anweisungen sind die JOB-, die EXEC- und die DD-Anweisungen. Mit der JOB-Anweisung werden Name, Jobklasse und Priorität für einen Job definiert. Der Name dient der Identifikation

des Jobs. Mit dem Zuordnen von Jobklassen kann der Mehrauftragsbetrieb effektiv organisiert werden. Wenn Jobs mit gleichem Ressourcenbedarf die gleiche Jobklasse zugeteilt wird, so werden diese Jobs nacheinander verarbeitet und behindern sich nicht gegenseitig. Über weitere Parameter sind die Angabe einer Priorität für vorrangige Verarbeitung, die Begrenzung der Joblaufzeit, die Ausgabe von Nachrichten über den Job und der Wiederanlauf des Jobs bei anormalem Ende auswählbar. Außerdem wird in der JOB-Anweisung die Größe des für die Jobausführung benötigten Speicherbereiches angegeben, und es wird festgelegt, ob das auszuführende Programm dem Seitenwechsel unterliegen darf. Ein Jobschritt wird mit einer EXEC-Anweisung beschrieben, mit der das auszuführende Programm oder eine Prozedur angegeben wird. Unter einer Prozedur ist eine definierte Folge von Jobsteueranweisungen zu verstehen, die vor der Ausführung für spezielle Zwecke modifiziert werden können. Mit den Parametern der EXEC-Anweisung können für den Jobschritt ähnliche Festlegungen getroffen werden wie in der JOB-Anweisung. Außerdem ist eine bedingte Verarbeitung eines Jobschrittes in Abhängigkeit vom erfolgreichen Beenden des vorhergehenden Jobschrittes möglich. Für jede Datei, die während des Ausführens eines Jobschrittes verarbeitet oder aufgebaut werden soll, muß eine DD-Anweisung codiert werden. Mit der DD-Anweisung können noch zur Ausführzeit des Jobschrittes die Attribute der Datei definiert und Festlegungen zur Dateiverarbeitung und zum verwenden des Gerätetyp getroffen werden. Damit kann der Nutzer weitestgehend geräteunabhängig programmieren.

Die Verarbeitung eines vom Nutzer definierten Jobs läuft in drei Phasen ab:

- Das Systemeingabeprogramm steuert die Jobeingabe.
- Der Programmlauforganisator steuert die Vor- und Nachbereitung der Job- und Jobschrittausführung.
- Das Systemausgabeprogramm gibt die Ergebnisse aus.

Die im Jobeingabestrom zusammengefaßten Jobbeschreibungen werden vom Systemeingabeprogramm gelesen, interpretiert und in Steuertabellen zusammengestellt. Entsprechend ihrer Priorität werden die Informationen der Jobs in die Kette der Jobs gleicher Klasse in der Jobkettendatei SYS1.SYS JOBQE eingeordnet. Sind im Eingabestrom neben den Steuer-



b. 4 Prinzip des Mehrrechnersystems

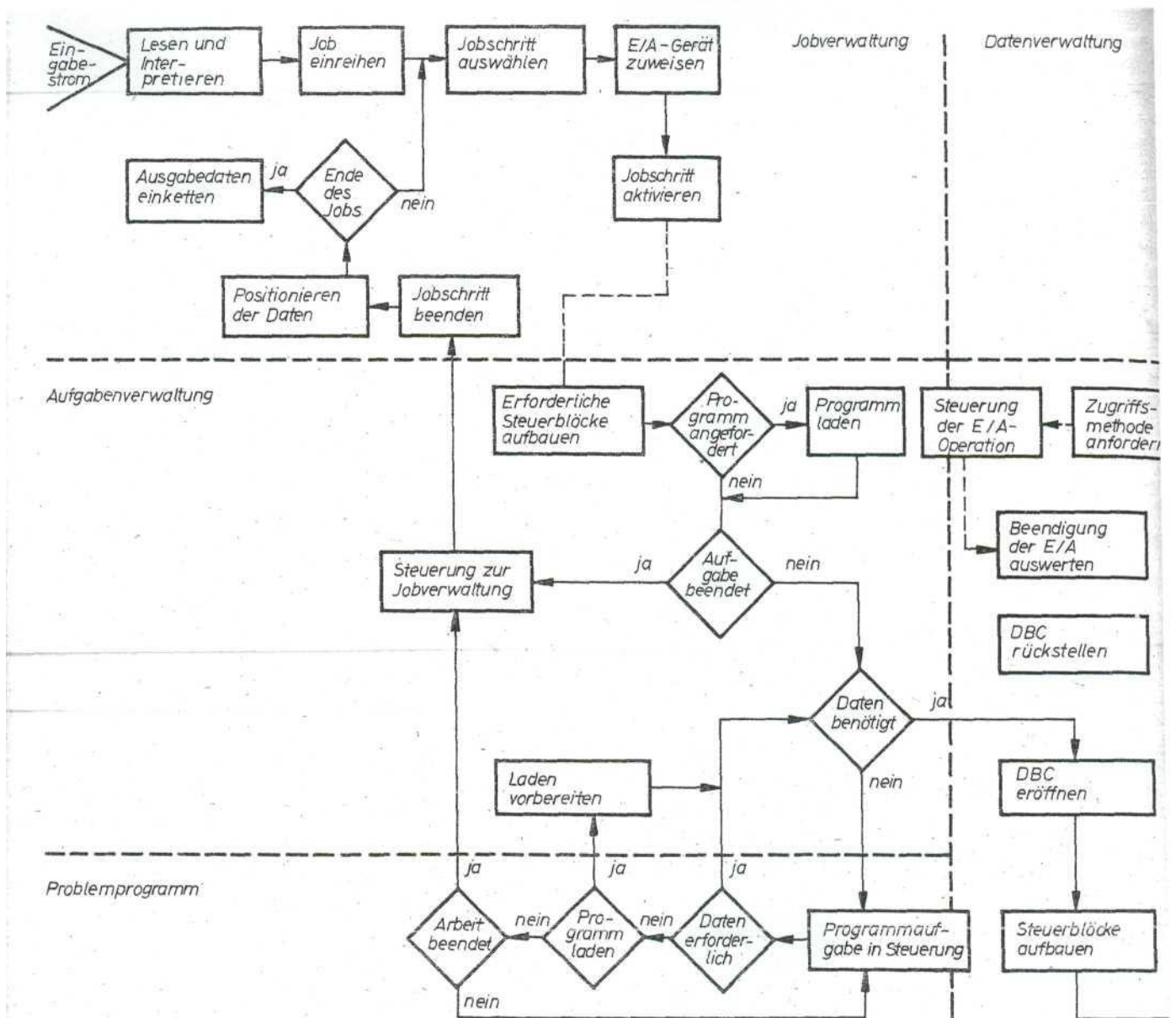


Abb. 5 Zusammenwirken der Steuerprogrammteile

anweisungen auch Eingabedaten enthalten, so werden diese auf Direktzugriffsspeichern zwischengespeichert. Die Jobs werden durch Programmlauf Organisatoren, die vom Bediener unter Zuordnung bestimmter Jobklassen gestartet wurden, aus den entsprechenden Jobketten zur Verarbeitung ausgewählt. Der Programmlauf Organisator erhält jeweils vor und nach Ausführung eines Jobschrittes die Steuerung. Bei der Vorbereitung werden die benötigten E/A-Geräte und Datenträger und Speicherplatz zugeordnet, das Problemprogramm wird geladen und erhält die Steuerung. Bei der Nachbereitung werden die zugeordneten Ressourcen unter Sicherung der erzeugten Daten freigegeben. Ausgabedaten für langsam arbeitende Geräte werden unter Berücksichtigung bestimmter Ausgabeklassen auf Direktzugriffsspeichern zwischengespeichert. Mit der Nachbereitung werden außerdem alle für Abrechnungszwecke erforderlichen Informationen über den Jobschritt zur Verfügung gestellt.

Vom Bediener werden zum geeigneten Zeitpunkt Systemausgabeprogramme unter Zuweisung bestimmter Ausgabeklassen und E/A-Geräte gestartet, mit denen die bei der Programmausführung entstandenen Daten ausgegeben werden.

Mit der Ausgabe 6.2 des OS/ES wird ein Jobeingabe-Subsystem (JES) zur Verfügung stehen, mit dem eine Leistungssteigerung gegenüber der bisher beschriebenen Jobverwaltung erzielt werden kann. Abb. 6 zeigt die Arbeitsphasen und die benötigten Arbeitsdateien unter Steuerung von JES. Mit JES werden die Informationen der eingelesenen Jobs in internen Jobketten nach Klassen und Prioritäten eingeordnet. Erst nach Auswahl durch den Programmlauforganisator werden die Steueranweisungen interpretiert und als Steuertabellen in Arbeitsbereichen des jeweiligen Programmlauforganisators gespeichert. Während bei der Jobverwaltung ohne JES die Jobketten (Eingabeketten und Ausgabeketten) und die aus den Jobsteueranweisungen aufgebauten Steuertabellen alle in der einen Systemdatei SYS 1 .SYS JOBQE gespeichert werden, wird diese Datei bei JES durch getrennte Arbeitsbereiche der Programmlauforganisatoren und die zwischengespeicherten Informationen ersetzt. Dadurch erfolgen die Zugriffe zu den Jobketten und Steuertabellen nicht als Zugriffe zur gleichen Datei, was die Gefahr der gegenseitigen Blockierung beseitigt, und die Arbeit der Programmlauforganisatoren wird schneller.

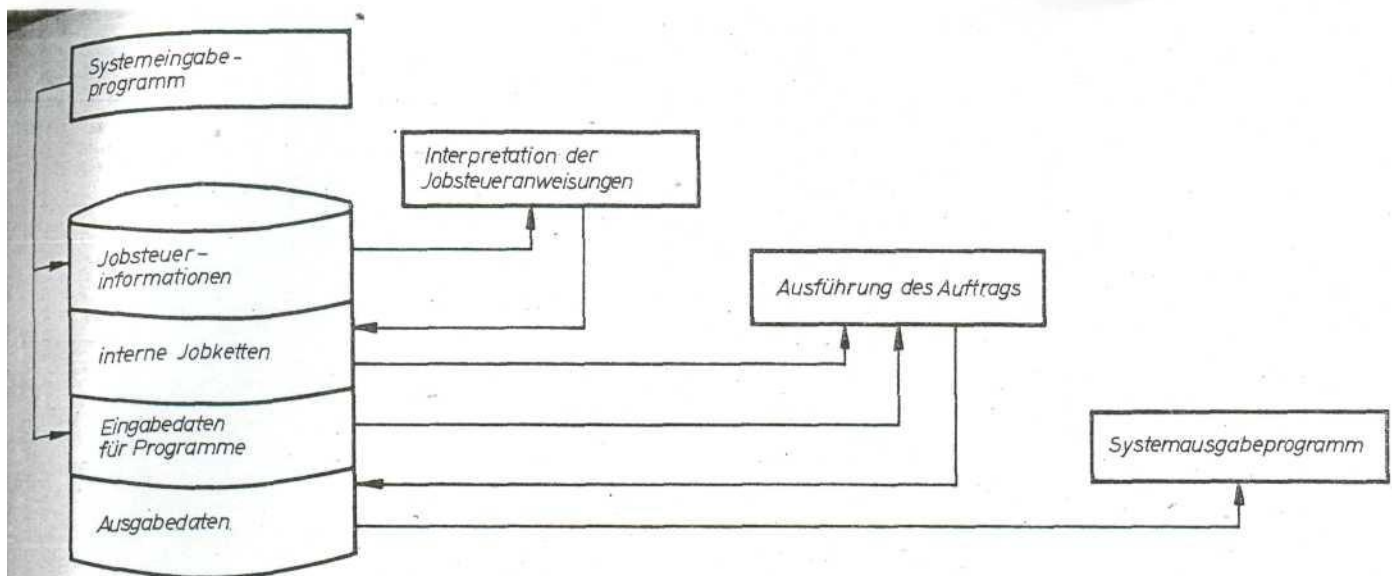


Abb. 6 JES: Verarbeitungsphasen und Pufferung der Daten

Von der Jobverwaltung wird außerdem die Kommunikation zwischen System und Bediener organisiert. Vom Betriebssystem werden bereits in der Initialisierungsphase Aktionen des Bedieners angefordert, es sind Nachrichten einzugeben, und während der Verarbeitungsphase erhält der Bediener z. B. Anforderungen über zu wechselnde Datenträger oder zu beseitigende Fehler an E/A-Geräten. Außerdem hat der Bediener die Möglichkeit, die Systemarbeit zu überwachen, indem er spezielle Kommandos von der Bedieneinheit aus an das System übergibt.

Es gibt drei Gruppen von Kommandos, die der Funktion nach einzuordnen sind in Kommandos zur Informationsgewinnung, zur Systemsteuerung und zur E/A-Steuerung. Für Anlagen mit einer großen **Anzahl** von E/A-Geräten ist es vorteilhaft, mehrere Bediener mit unterschiedlichem Verantwortungsbereich einsetzen zu können. Die wahlweise generierbare Funktion der Mehrbedieneinheiten-Unterstützung gewährleistet, daß Bediener von getrennten Bedieneinheiten aus mit dem System korrespondieren können. Jeder Bedieneinheit werden Nachrichtenleitcodes und ein Kommandoautorisierungscode zugeordnet, über die die Kommunikation mit dem einzelnen Bediener erfolgen kann. Eine der Bedieneinheiten arbeitet als Hauptbedieneinheit, die für bestimmte Kommandos allein autorisiert werden kann. Das System führt in Form einer Dauerkopie ein Protokoll über den gesamten Informationsaustausch zwischen allen Bedienern und dem System.

2.2. Datenverwaltung

Mit den Routinen der Datenverwaltung wird die Verarbeitung aller Daten, die als Dateien auf Datenträgern aufgebaut werden, organisiert. Dazu gehören das Steuern aller E/A-Operationen, das Aufbereiten der Daten und das Überlappen und Synchronisieren der Arbeit der ZE mit den E/A-Operationen sowie die Datensicherung und Katalogisierung der Dateien.

Unter Ausnutzung des vollen Funktionsumfangs der E/A-Geräte wird eine weitgehende Unabhängigkeit des Problemprogramms von den Geräten gesichert. Grundlage dafür ist

ein Konzept, nach dem erst zur Ausführung des Programms während der Eröffnungsphase für eine Datei die E/A-Bereiche im Hauptspeicher (Puffer) zur Aufnahme der Daten angelegt, die zur Verarbeitung benötigten Systemprogramme (Zugriffsroutinen) geladen und die erforderlichen Kanalprogramme für das E/A-Gerät gebildet werden.

4,

• Dateistruktur und Datenträger

Entsprechend dem zu lösenden Problem wählt der Nutzer für seine zu bearbeitenden Dateien eine bestimmte Organisationsform und Art der Datenträger aus. So werden z. B. größere Datenmengen zum Aufbau einer Datei über Lochkarten oder Datenfernverarbeitungsgeräte eingegeben. Eine zu verwaltende Datei wird je nach Häufigkeit der Verwendung und Notwendigkeit einer Aktualisierung entweder auf Magnetband (große Bestände mit mittlerer Zugriffsrate) oder auf Direktzugriffsspeichern (Bestände mit hoher Zugriffs- und Aktualisierungsrate) aufgebaut.

Dateien können in vom Anwender lesbarer Form über Bildschirmgeräte, Drucker oder Mikrofilmabgabegeräte erzeugt werden. Die kleinste Informationseinheit einer Datei ist der Satz. Auf dem Datenträger werden Blöcke physisch getrennt gespeichert, die aus einem oder mehreren Sätzen bestehen. Die Sätze können ein festes, variables oder auch undefiniertes Satzformat haben. Sind alle Sätze einer Datei gleich lang, so werden sie als Sätze fester Länge bezeichnet. Sätze variabler Länge enthalten ein zusätzliches Feld, in dem die eigene Satzlänge eingetragen ist, ebenso hat jeder Block, der Sätze variabler Länge enthält, ein Feld, das die Blocklänge definiert. Bei Sätzen fester und variabler Länge besteht die Möglichkeit, einen Satz auf mehrere Blöcke aufzuteilen. Damit wird die Begrenzung der Satzlänge von der Blockgröße aufgehoben. Der Teil eines Satzes, der im Block enthalten ist, wird als Segment bezeichnet.

Sätze undefinierter Länge enthalten keine Angabe zu ihrer Länge. Sie dürfen nur eine vom Problemprogramm festgelegte Länge nicht überschreiten, da es sonst Probleme mit dem bereitgestellten E/A-Puffer gibt.

Alle Datenträger, die dem Betriebssystem zur Verarbeitung angeboten werden, sollten eine visuell erkennbare Kennzeich-

nung tragen, über die eine Zuordnung zum Inhalt des Datenträgers möglich ist. Darüber hinaus schützt sich das Betriebssystem gegen Fehlbedienungen, indem zur Dateizuordnung und Datensicherung der Inhalt und Aufbau der Datenträger für Magnetbandspeicher und Direktzugriffsspeicher mit maschinenlesbaren standardisierten Kennsätzen beschrieben werden. Ausgehend von ihrer Funktion werden Datenträgerkennsätze und Dateikennsätze unterschieden. Der Datenträgerkennsatz enthält eine Datenträgerarchivnummer (Abb. 7 und Abb. 8) und für Direktzugriffsspeicher auch die Adresse des Datenträgerinhaltsverzeichnis (VTOC). Das Datenträgerinhaltsverzeichnis enthält Dateikennsätze, die die gespeicherten Dateien nach Größe, Lage und Organisationsform beschreiben und Informationen zur Dateiidentifikation, zum Kennwortschutz, Herstellungs- und Verfallsdatum und Dateicharakteristika enthalten. Unter Verwendung dieser Informationen können Dateien auf Wunsch des Nutzers katalogisiert werden. Eine Datei gilt als katalogisiert, wenn Name, Datenträger- und Gerätetypinformationen im Systemkatalog eingetragen sind. Der Systemkatalog ist selbst eine Datei (SYS 1, SYSCATLG), die auf mehreren Direktzugriffsspeichern aufgebaut sein kann. Die zur Datenverwaltung gehörende Katalogverwaltung gestattet dem Anwender, eine katalogisierte Datei nur über Angabe ihres Namens wieder aufzufinden und zu verarbeiten. Um eine logisch sinnvolle Namensgebung für Dateien zu unterstützen, bietet das System die Möglichkeit, einen Namen zu wählen, der sich aus einzelnen bis zu acht Zeichen langen und mit Punkten voneinander getrennten Unterbegriffen zusammensetzt. Dadurch wird die Gefahr der Auswahl doppelter Namen wesentlich herabgesetzt. Derartige Namen werden als qualifizierte Dateinamen bezeichnet. Der Aufbau der Katalogdatei in Indexstufen ist diesem Unterteilungsprinzip angepaßt. Wenn es erforderlich ist, Dateien häufig zu aktualisieren und die ursprünglichen Daten noch aufzubewahren, so bietet die Katalogverwaltung mit den Generationsdatengruppen eine gute Unterstützung. Jede aufzubewahrende Neufassung einer

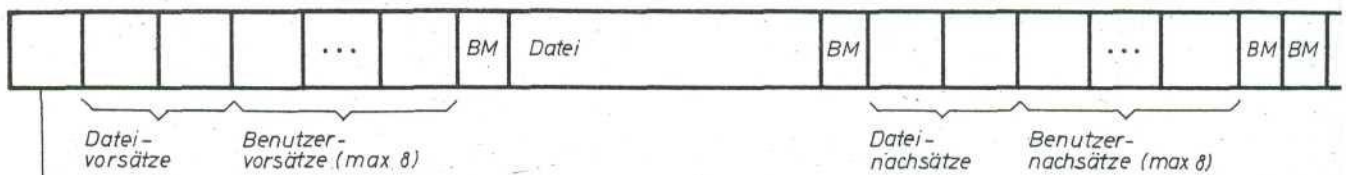
solchen Datei wird als Generation bezeichnet. Durch Verwenden eines Dienstprogramms kann der Nutzer eine spezielle Indexstufe im Systemkatalog anlegen lassen und dabei angeben, wieviel Generationen einer Datei gleichzeitig katalogisiert sein dürfen. Bei Überschreiten dieser Zahl wird die älteste Generation automatisch gelöscht.

• *Dateiorganisation*

Der interne Aufbau einer Datei wird außer vom Satzform; vor allem von der Dateiorganisationsform bestimmt. Mit dem Betriebssystem OS/ES werden folgende Dateiorganisationsformen unterstützt: sequentielle, indexsequentielle, untergliederte und gesteuerte Dateien sowie Bildschirmdateien, Mikrofilmdateien, Dateien der Datenfernverarbeitung und eine verallgemeinerte Dateiorganisationsform für Direktzugriffsspeicher. Letztere wird besonders effektiv mit der Nutzung aller Funktionen der 100- bzw. 200-M-Byte-Wechselplattenspeicher und der virtuellen Speicherkonzeption angepaßt. Die sequentielle Dateiorganisation erfordert eine starr fortlaufende Verarbeitung, die nicht auf einen logischen Zusammenhang zwischen einzelnen Sätzen der Datei, sondern auch deren Reihenfolge orientiert ist. Sie ist für alle Datenträger und Gerätetypen einsetzbar.

Indexsequentielle Dateien werden auf Direktzugriffsspeichern aufgebaut und können sequentiell und wahlfrei verarbeitet werden. Einzelne Sätze können aktualisiert, gelöscht oder auch hinzugefügt werden. Beim Aufbau einer indexsequentiellen Datei werden die Sätze in aufsteigender Folge aufgezeichnet, jedem Satz muß ein Ordnungsbegriff als Schlüssel vorgestelt werden (Abb. 9).

Jede indexsequentielle Datei besteht aus einem Hauptdatenbereich für die Datensätze, einem Indexbereich, der vom Steuerprogramm zum schnellen Auffinden der Sätze aufgebaut wird und einem Überlaufbereich, in den bei Neuzugängen Sätze ausgelagert werden können. Untergliederte Dateien bestehen aus mehreren sequentiell organisierten Beständen und können nur auf Direktzugriff



Datenträgerkennsatz

Abb. 7 Aufbau eines Magnetbandes mit Standardkennsätzen und einer Datei

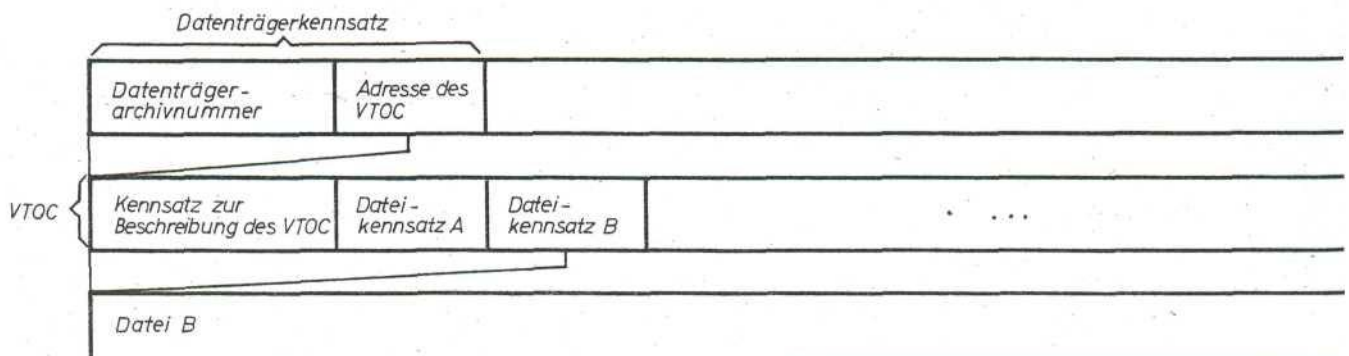


Abb. 8 Kennsätze eines Datenträgers für direkten Zugriff

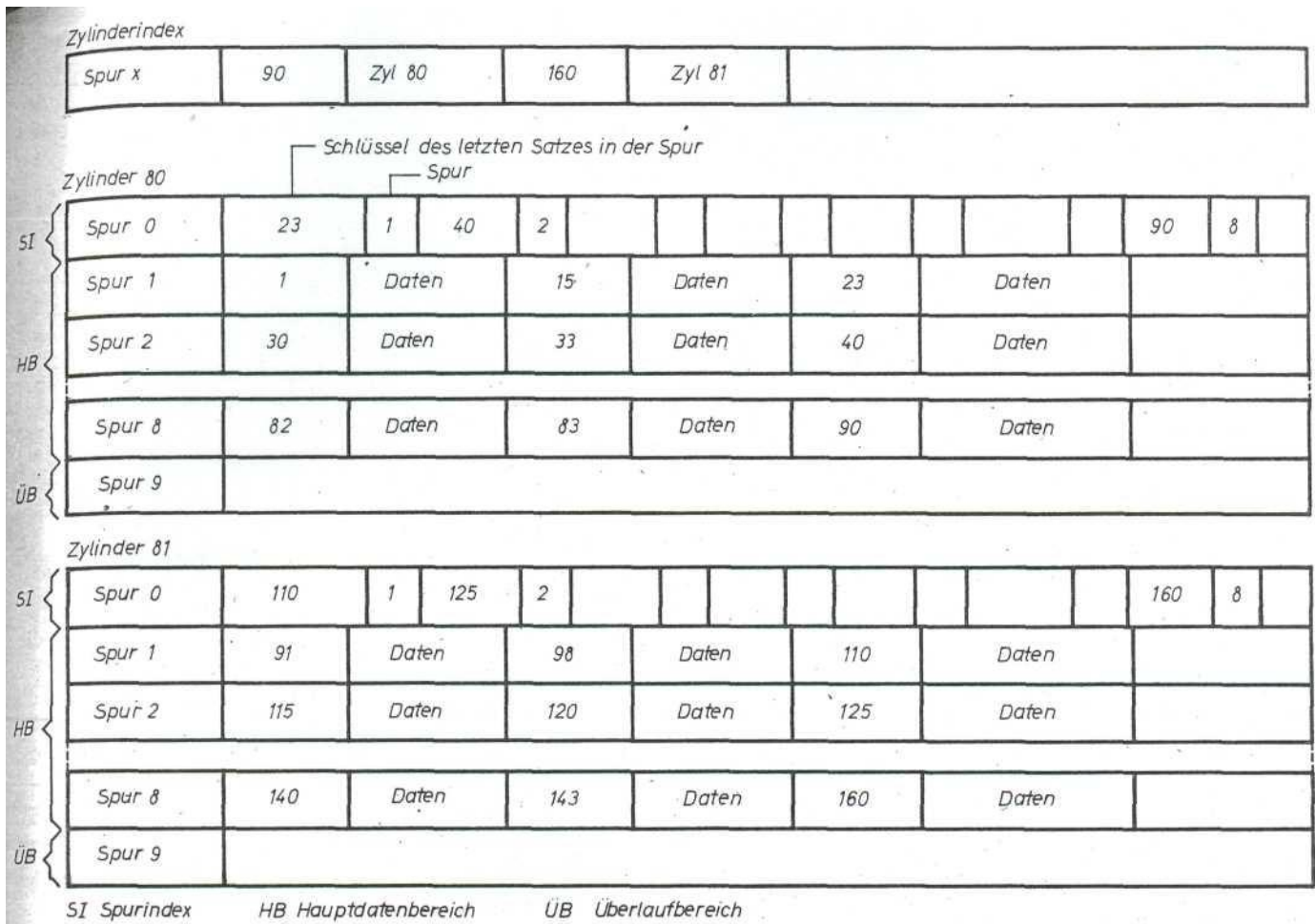


Abb. 9 Aufbau einer indexsequentiellen Datei

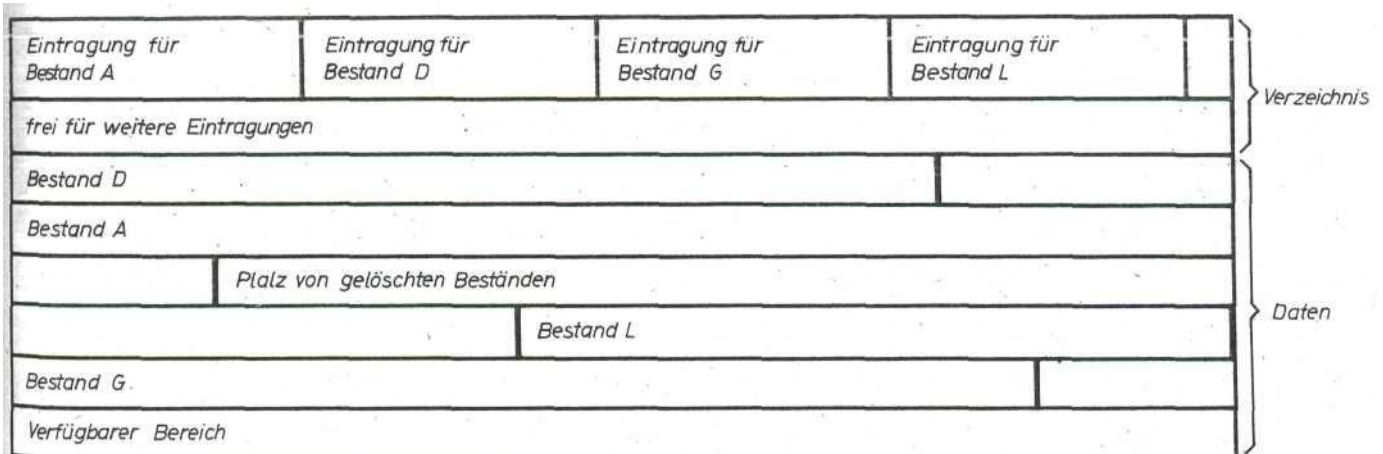


Abb. 10 Aufbau einer untergliederten Datei

speichern aufgebaut werden. Am Anfang jeder untergliederten Datei steht ein Verzeichnis, in dem ein bis zu acht Zeichen langer Name für jeden Bestand und eine relative Spuradresse, bezogen auf den Dateianfang, eingetragen sind (Abb. 10). In einer untergliederten Datei können Bestände gelöscht oder hinzugefügt werden, Voraussetzung sind gleiche Eigenschaften der Bestände bezüglich Satzformat und Blocklänge. Diese Organisationsform ist für den Aufbau von Programm Bibliotheken geeignet und wird auch vom Betriebssystem vielfach selbst verwendet. Bei Dateien mit gesteuert Organisationsform besteht eine

feste Beziehung zwischen dem Ordnungsbegriff (Schlüssel) der Sätze und der Adresse des Speicherplatzes auf dem Direktzugriffsspeicher. Diese Beziehung stellt das Problemprogramm her. Diese Organisationsform sollte verwendet werden, wenn die Zugriffszeit zu einer bestimmten Information minimal gehalten werden muß.

Für die Arbeit mit dem virtuellen Speicherkonzept steht als effektivste Form der Datenspeicherung auf Direktzugriffsspeichern die verallgemeinerte Organisationsform zur Verfügung, die für den Nutzer den Funktionsumfang der sequentiellen, indexsequentiellen und gestreuten Organisation in sich

vereinigt. Die Datensätze werden in Zugangsfolge oder in Schlüsselfolge gespeichert, die Adressierung der Sätze erfolgt in beiden Fällen über die relative Byteadresse, bezogen auf den Dateianfang. Damit wird die Datei unabhängig vom Typ des Direktzugriffsspeichers. Die Struktur der Datei wird durch formatierte Steuerbereiche fester Länge bestimmt, die eine gleichgroße Anzahl von Steuerintervallen fester Länge und den Steuerbereich beschreibende Definitionsfelder enthalten (Abb. 11).

Wird die Datei in Schlüsselfolge aufgebaut, so wird durch das Steuerprogramm eine Hierarchie von Indexstufen geschaffen. Die unterste Indexstufe enthält einen Indexsatz für jeden Steuerbereich. Dieser Indexsatz enthält eine Eintragung für jedes Steuerintervall des Steuerbereiches, bestehend aus dem Schlüssel des letzten Satzes und der relativen Byteadresse des ersten Satzes im Steuerintervall oder der Anzeige für ein freies Intervall. Die nächsthöhere Indexstufe enthält eine Eintragung je Indexsatz der darunterliegenden Indexstufe, bis die höchste Indexstufe aus nur einem Indexsatz besteht. Die Dateiorganisationsform bei Datenfernverarbeitungsgeräten hat einen sequentiellen Charakter, ist aber dadurch gekennzeichnet, daß mit diesen Geräten Daten und Nachrichten in beliebiger Reihenfolge übertragen werden, die nicht der Reihenfolge der Verarbeitung im Programm entsprechen muß. Dateien, die über Bildschirmgeräte verarbeitet werden, lassen eine Adressierung der einzelnen Zeichen oder Bildpunkte innerhalb einer Zeile zu.

Eine visuell lesbare Aufzeichnung von Daten kann mit Mikrofilmausgabegeräten erfolgen, was sich besonders für die Archivierung umfangreicher Datenmengen eignet. Die Daten werden in Bildern aufgezeichnet, die der Druckerausgabe im Format A3 mit 132 Zeichen pro Zeile oder im Format A4 mit 2 x 62 Zeichen pro Zeile und 64 Zeilen pro Bild entsprechen. Die Bilder werden sequentiell aufgezeichnet, wobei aber an beliebigen Stellen zwischen den Bildern zusätzliche Bilder mit Kennzeichnungsdaten zur visuellen Wiederauffindbarkeit der Mikrofilme eingefügt werden können.

• Zugriffsmethoden

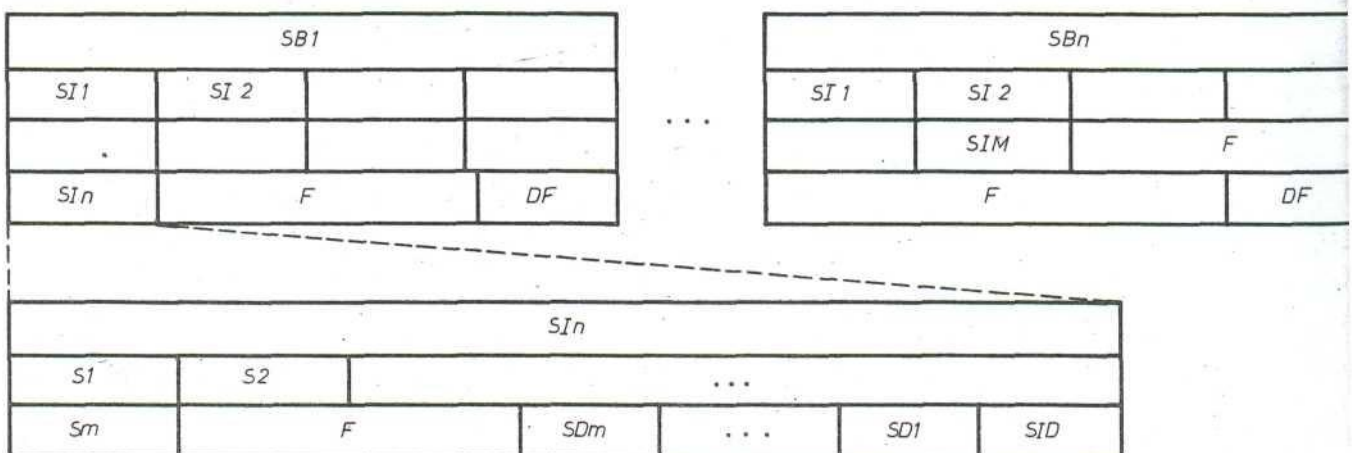
Zur Behandlung der einzelnen Datenstrukturen stellt das Betriebssystem dem Nutzer eine Reihe von Zugriffsmethoden

zur Verfügung. Unter einer Zugriffsmethode ist das Verfahren zu verstehen, mit dem unter Verwenden einer speziell, Zugriffstechnik eine nach bestimmten Prinzipien aufgebaute Datei verarbeitet wird. Somit ist eine Zugriffsmethode eine Kombination von Dateioorganisation und einfacher oder erweiterter Zugriffstechnik. Die erweiterte Zugriffstechnik unterstützt das Blocken und Entblocken logischer Sätze und die Synchronisation der Programmverarbeitung mit den E/A-Operationen. Bei der einfachen Zugriffstechnik ist der Anwender für das Blocken oder Entblocken der Daten und für das Pflegen der Beendigung von E/A-Operationen verantwortlich. Mit der Auswahl von Dateiorganisationsform und Zugriffstechnik legt das Problemprogramm die Zugriffsmethode fest die als Komplex spezieller Routinen die Steuerung der Datenverarbeitung übernimmt. Die vom Betriebssystem OS/ES unterstützten Zugriffsmethoden sind in der Tab. 1 zusammen gefaßt (s. Seite 53).

2.3. Aufgabenverwaltung

Die Aufgabenverwaltung ist der zentrale Teil zur Organisation des Multiprogrammbetriebes und überwacht alle Aufgaben Problemprogramme und des Steuerprogramms. Jeder Aufgabe wird ein Komplex von Ressourcen zugewiesen dazu gehören geladene Programme, Arbeitsbereiche und Daten auf E/A-Geräten. Die Aufgabe bewirbt sich konkurrierend mit anderen Aufgaben um die Zuteilung von ZE-Zeit, damit das ihr zugeordnete Programm aktiv werden kann. Die Auswahl der zu aktivierenden Aufgabe erfolgt nach der Priorität. Verliert eine Aufgabe ihre Arbeitsfähigkeit, z. B. wegen des Starts einer E/A-Operation, oder wird eine Aufgabe höherer Priorität arbeitsfähig, z. B. wird eine von ihr eingeleitete E/A-Operation beendet, so tritt ein Aufgabenwechsel mit erneuter Aufgabenauswahl ein.

Um bei der Bearbeitung von Aufgaben, die vorwiegend die ZE beanspruchen, die Auswahl anderer Aufgaben nicht zu behindern, gibt es weitere Auswahlkriterien, die der Prioritätsauswahl untergeordnet sind. So werden z. B. bei der dynamischen Aufgabenauswahl die Prioritäten innerhalb eines fest zulegenden Prioritätsintervalls vom Steuerprogramm verändert indem die E/A-Aktivitäten der einzelnen Aufgaben ständig



SB Steuerbereich
 SI Steuerintervall
 S Satz
 F freier Bereich
 DF Definitionsfeld
 SD Satzdefinitionsfeld
 SID Steuerintervall - Definitionsfeld

Abb. 11 Strukturelemente einer Datei für verallgemeinerten Direktzugriff

Tabelle 1 Zugriffsmethoden

Dateiorganisation	Zugriffstechnik	
	einfach	erweitert
sequentiell	einfache sequentielle Zugriffsmethode BSAM	erweiterte sequentielle Zugriffsmethode QSAM
indexsequentiell	einfache indexsequentielle Zugriffsmethode BISAM	erweiterte indexsequentielle Zugriffsmethode QISAM
untergliedert	einfache Zugriffsmethode für untergliederte Dateien BPAM	
direkt	einfache Direktzugriffsmethode BDAM	
Bildschirmdatei	grafische Zugriffsmethode GAM	
Mikrofilmdatei		erweiterte Mikrofilm-Zugriffsmethode QFAM
Datenfernverarbeitung	Basis-Zugriffsmethode für Datenfernverarbeitung BTAM	erweiterte Zugriffsmethode für Datenfernverarbeitung mit Nachrichtensteuerprogramm TCAM virtuelle Zugriffsmethode für Datenfernverarbeitungsnetze VTAM
verallgemeinerter Direktzugriff (Schlüsselfolge)	VSAM	VSAM
	direkte Verarbeitung	sprungsequentielle und sequentielle Verarbeitung
verallgemeinerter Direktzugriff (Zugangsfolge)	direkte Verarbeitung	sequentielle Verarbeitung

ausgewertet und Aufgaben mit hoher E/A-Aktivität hohe Prioritäten zugeordnet werden.

Alle Aktivitäten der Aufgabenverwaltung beginnen mit einer Unterbrechung der Ausführung der normalen Anweisungsfolge eines Programms. Ursache der Unterbrechung kann entweder das Eintreten eines Ereignisses, das durch das Steuerprogramm zu überwachen ist oder die Anforderung der Dienst der Aufgabenverwaltung vom Programm sein. Die Unterbrechung wird analysiert, und alle Informationen, die zur Fortsetzung des unterbrochenen Programms erforderlich sind, werden gerettet.

Zu den vom Programm anforderbaren Diensten der Aufgabenverwaltung gehören:

- Ändern der Auswahlpriorität
- Synchronisation von parallel ablaufenden und voneinander abhängigen Vorgängen r Anforderung und Freigabe von Bereichen im virtuellen Speicher
- r Fixieren und Freigeben von Seiten im Realspeicher -
- Laden und Aktivieren von Programmen

- Setzen von Prüfpunkten für Wiederanlauf bei Fehlern
- Anwenden der Zeitgebereinrichtung für zyklische Unterbrechungen.

Diese Funktionen sind von Problemprogrammen vollständig nutzbar, wenn sie in der Assemblersprache codiert sind. Jede vom Nutzer definierte Aufgabe kann weitere Aufgaben — Unteraufgaben — erzeugen, die aber parallel zur Ursprungsaufgabe verarbeitbar sein müssen. Eine Aufgabe wird unter Freigabe aller zugeordneten Ressourcen beendet, wenn alle zur Aufgabe gehörenden Unteraufgaben verarbeitet wurden. Tritt bei der Verarbeitung ein Fehler auf, so erfolgt eine anormale Beendigung, wobei nutzereigene Routinen die Steuerung erhalten, um geeignete Maßnahmen für das abgebrochene Programm einzuleiten.

2.4. Verwaltung des virtuellen Speichers

Dieser Abschnitt setzt die Kenntnis von Begriffen und Darstellungen zum virtuellen Speicher voraus, die aus gerätetechnischer Sicht für den Hauptspeicher der EC 1055 einschließlich der dynamischen Adreßumsetzung erläutert wurden. Das Steuerprogramm, alle gleichzeitig laufenden Aufträge und die von ihnen gemeinsam genutzten Routinen teilen sich in den virtuellen Speicher von 16 M Byte. Die Zuordnung des virtuellen Speichers an die einzelnen Aufträge erfolgt in Vielfachen von Segmenten. Damit wird die Segmenttabelle zum wesentlichsten Verwaltungsinstrument, das auch zum Schutz des virtuellen Speicherbereiches eines Auftrages vor dem Zerstören oder Benutzen durch einen anderen Auftrag dient.

Zur Verwaltung des Realspeichers führt das Steuerprogramm eine Seitenrahmentabelle. Sie hat für jeden Seitenrahmen eine Eintragung, die angibt, ob der Seitenrahmen frei ist oder welche Seite er enthält. Seitenrahmen werden einzeln nach Bedarf zugeordnet. Dem ständig im Realspeicher festgehaltenen Teil des Steuerprogramms ist Realspeicherplatz in Vielfachen von 2 K Byte zugeteilt.

Der Seitenspeicher ist in 2 K Byte große Blöcke, die Seitenablagefächer, eingeteilt, die bei Bedarf vom Steuerprogramm mit Seiten belegt werden. Die Belegung des Seitenspeichers geschieht unabhängig von der Reihenfolge, in der die Seiten im virtuellen Adreßraum angeordnet sind. Das Steuerprogramm führt eine Übersicht über die Belegung der Seitenablagefächer. Sie wird benutzt, um ein freies Fach auszuwählen, wenn eine Seite aus dem Realspeicher ausgelagert werden muß. Außerdem verwendet das Steuerprogramm externe Seitentabellen, die angeben, wo sich ausgelagerte Seiten befinden, und die benutzt werden, wenn Seiten in den Realspeicher gebracht werden müssen. Zu jeder Seitentabelle gehört jeweils eine externe Seitentabelle, die für jede Seite, des betreffenden Segments eine Eintragung enthält. Diese Eintragung gibt die Nummer des Faches an, auf dem sich die Seite befindet, nachdem sie ausgelagert wurde. An der gleichen Stelle der Seitentabelle ist die Eintragung als ungültig markiert. Die Tab. 2 faßt die Hauptfunktion jeder zur Verwaltung des virtuellen Speichers benutzten Tabelle zusammen. Mit dem Konzept des virtuellen Speichers wird der für die " ablaufenden Programme und deren Daten verfügbare Speicherraum erheblich vergrößert. Da der Realspeicher nur mit aktiven Programmteilen belegt wird, ist es möglich, Problemlösun-

Tabelle 2 Übersicht der Tabellen zur Verwaltung des virtuellen Speichers

Verwaltungsmittel	Verwaltungsbereich	Funktion
Segmenttabelle	virtueller Speicher	Abbildung des virtuellen Speichers auf den Realspeicher
Seitentabelle	Segment	Abbildung des virtuellen Speichers auf den Seitenspeicher
externe Seitentabelle	Segment	
Seitenrahmentabelle	Realspeicher	Belegungsübersicht
Fachbelegungsübersicht	Seitenspeicher	Belegungsübersicht

gen zu realisieren, deren benötigter Adreßraum die Realspeichergroße weit überschreitet.

2.5. Fehlerverwaltung und Diagnose

Mit dem Betriebssystem OS/ES werden Diagnoseprogramme bereitgestellt, um die Auswirkungen gerätetechnisch bedingter Fehlfunktionen möglichst gering zu halten und die Zuverlässigkeit des Gesamtsystems zu erhöhen. Diese Diagnoseprogramme werden aktiviert, wenn das Betriebssystem über eine Maschinenfehlerunterbrechung einen Maschinenfehler oder über eine E/A-Unterbrechung einen Kanal- oder Gerätefehler erkennt. Der Fehlerzustand wird analysiert und aufgezeichnet, wobei Informationen ausgewertet werden, die die Gerätetechnik zur Verfügung stellt. Die Aufzeichnung aller Fehlerarten erfolgt in Form von Fehlersätzen in der Datei SYS1.LOGREC von der sie zu jedem beliebigen Zeitpunkt in aufbereiteter Form ausgedruckt werden können.

Unter Steuerung des Betriebssystems arbeiten folgende Routinen zur Fehlerverwaltung:

- Geräteabhängige Fehlerrouninen analysieren Fehlerzustände von E/A-Geräten und wiederholen fehlerhafte E/A-Operationen.
- Die Maschinenfehlerroutine (MCM) entscheidet nach der Analyse des Fehlerzustandes, ob eine Wiederherstellung der Systemfunktion möglich ist. Ist das System noch arbeitsfähig, so wird die betroffene Aufgabe normal beendet und der Fehler aufgezeichnet. Andernfalls wird das System in den Wartezustand versetzt.

- Die Kanalfehlerroutine (CCH) analysiert Fehlerzustände der Kanäle und leitet Wiederholungen der betroffenen Operationen ein oder versetzt das System in den Wartezustand.
- Mit der dynamischen Rekonfiguration der Geräte (DE) wird bei nichtkorrigierbaren Gerätefehlern ein Umsetzen des zugeordneten Datenträgers auf ein nicht belegtes **Gerät** organisiert und die Weiterarbeit des betroffenen Jobs gewährleistet.
- Das Aufbereiten der aufgezeichneten Fehlersätze erfolgt mit dem Programm EREP unter Steuerung des Systems

Wenn so schwerwiegende Fehler auftreten, daß eine Aufnahme durch das Betriebssystem nicht mehr möglich ist, können durch das systemunabhängige Programm SEREP die von den technischen Diagnoseeinrichtungen bereitgestellten Informationen sofort in aufbereiteter Form ausgedruckt werden.

Die Zusammenarbeit der Diagnoseprogramme zeigt Abb. Neben den Routinen zur Analyse von Fehlerzuständen steht ein Steuerprogramm für den on-line-Gerätetest parallel zur Systemarbeit zur Verfügung. Das Steuerprogramm stellt die Verbindung zwischen dem Betriebssystem und den Gerätetestsektionen für das spezielle Gerät her. Damit ist das Wartungspersonal in der Lage, ohne Unterbrechung der Systemarbeit eine prophylaktische Überprüfung der E/A-Geräte zu führen.

3. Verarbeitungsfunktionen Zum Umfang des Betriebssystems OS/ES gehört eine Reihe problemunabhängiger Programme, mit deren Anwenden < Codieraufwand für häufig auszuführende Arbeiten stark herab gesetzt werden kann. Dazu gehören die Dienstprogramme sind Programmverbinder und Lader und ein Sortier-/Mischprogramm

3.1. Dienstprogramme

Dienstprogramme können über Jobsteueranweisungen aufgerufen werden, ihre Ausführung wird mit Angabe spezielle Steueranweisungen modifiziert. Die Dienstprogramme sind funktionell in drei Gruppen einteilbar. Die Dateidienstprogramme sind zum Ändern, Reorganisieren oder Vergleichen von beliebigen Dateien vorgesehen. Die Gruppe der Systemdienstprogramme dient der Verwaltung von Datenträgern und Systemsteuerdaten. Sie sind zur

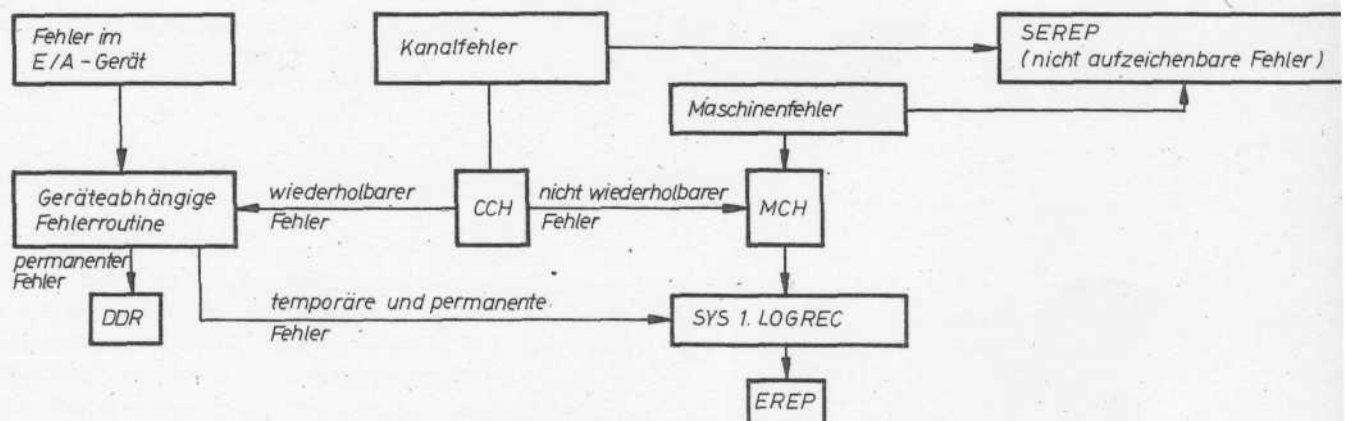


Abb. 12 Zusammenwirken der Komponenten der Fehlerverwaltung

Fehlersuche sowie zur Systemwartung einsetzbar. Mit Hilfe der n Gruppe, der systemunabhängigen Dienstprogramme, können-Direktzugriffsspeicher für die Arbeit mit dem Betriebssystem bereitet und Informationen zur Fehlersuche durch Ausdrucken von Speicherinhalten gewonnen werden. In Tab. 3 sind die Dienstprogramme mit ihren Hauptfunktionen zusammengefaßt.

3.2. Programmverbinder und Lader

Jedes Programm, das vom Betriebssystem verarbeitet werden soll, muß in einer geeigneten, ausführbaren Form vorliegen. Diese Form besitzen die vom Assembler und von Sprachübersetzern erzeugten Programme noch nicht. Erst nach der Bearbeitung durch den Programmverbinder oder Lader erreichen sie eine ausführbare Form, wobei mehrere Programmteile, die von verschiedenen Sprachübersetzern erzeugt worden sind, zu einem einzigen Programm verbunden werden können. Programmverbinder und Lader unterscheiden sich darin, wie das verbundene Programm weiter behandelt wird. Der Programmverbinder speichert das erzeugte Programm in einer externen Bibliothek, aus der es zu einem späteren Zeitpunkt aufgerufen werden kann, während der Lader das Programm im Hauptspeicher placiert und sofort aktiviert. Es bleibt keine wiederverwendbare Kopie auf den Externspeichern erhalten. Das Verwenden des Laders ist daher vorwiegend für die Phase des Programmtests geeignet.

Beim Verbinden der verschiedenen Programmteile können mit entsprechenden Steueranweisungen beliebige Programmabschnitte gelöscht, ersetzt, umgeordnet oder auch umbenannt werden.

Tabelle 3 Übersicht der Dienstprogramme

Datei-Dienstprogramm	Hauptfunktionen
IEBCOMPR	Vergleichen sequentieller Dateien, untergliederter Dateien und von Satzteilen sequentieller und untergliederter Dateien
IEBCOPY	Kopieren untergliederter Dateien und ausgewählter Bestände, Ersetzen und Umbenennen von Beständen, Kopieren und Ausschluß bestimmter Bestände, Veränderung des Speicherplatzes einer Datei, Umblocken, Verdichten
IEBDG	Erzeugen oder Umblocken von Dateien, Ändern der Satzlänge, Umordnen, Indizieren oder Überschriften von Satzfeldern, Ändern der Satzanzahl, Ändern der Organisationsform einer Datei
IEBEDIT	Zusammenstellen eines Jobstromes
IEBGENER	Erzeugen von Beständen, Kopieren oder Aufbereiten einer sequentiellen Datei oder eines Bestandes, Verarbeiten von Nutzerkennsätzen
IEBISAM	Kopieren, Laden, Entladen oder Drucken einer indexsequentiellen Datei
IEBPTPCH	Drucken oder Stanzen einer Datei, ausgewählter Bestände, ausgewählter Sätze, des Verzeichnisses oder Drucken und Stanzen mit Aufbereiten einer Datei
IEBUPDTE	Erzeugen und Aktualisieren symbolischer Bibliotheken, Aktualisieren von Beständen der Bibliotheken, Ändern der Organisationsform von sequentiellen oder untergliederter Dateien, Eintragen von Nutzerkennsätzen

System-Dienstprogramm	Hauptfunktionen
IEHATLAS	Wiederherstellen von Daten auf Direktzugriffsspeichern
IEHCHCAT	Ändern der Eintragungen im Katalog
IEHDASDR	Direktzugriffsspeicher Initialisieren, Formatieren, Abziehen, Ausdrucken, Rückspeichern, Kennsatz ändern, Ersatzspuren zuordnen
IEHINITT	Initialisieren von Magnetbändern
IEHIOSUP	Änderung von relativen Adresseintragungen
IEHLIST	Ausdrucken des Datenträger-Inhaltsverzeichnisses, des Katalogs oder des Verzeichnisses einer untergliederten Datei
IEHMOVE	Umspeichern oder Kopieren einer Datei, einer Gruppe katalogisierter Dateien, des Katalogs oder eines Datenträgers
IEHPROGM	Eintragen und Löschen im Katalog, Löschen von Dateien oder Beständen, Umbenennen von Dateien oder Beständen, Verbinden und Trennen von Katalogdatenträgern, Aufbau eines Gruppenindex von Dateigenerationen, Eintragen, Löschen und Drucken der Kennwortdatei
IFCDIPOO	Initialisieren und Formatieren der Datei SYS1.LOGREC
IFCEREPO	Erzeugen und Fortschreiben einer Fehlerstammdatei, Drucken von Fehlersätzen
AMDPRDMP (SVS)	Ausdrucken wahlweiser Bereiche eines Speicherabzuges in aufbereiteter Form, Kopieren von Speicherabzügen
IMDPRDMP (MFT/MVT)	
IMBLIST	Ausdrucken von Übersichten des Steuerprogrammkerne, Ausdrucken von Übersichten der Lademoduln in untergliederten Dateien oder im Bereich für residente Lademoduln
IMASPZAP	Ersetzen von Befehlen oder Konstanten in ausführbaren Programmen, Ausdrucken ausführbarer Programme (geeignet für Fehlersuche und Korrektur)
Systemunabhängige Dienstprogramme	Hauptfunktionen
AMDSADMP (SVS)	Erzeugen eines Hauptspeicherabzuges
IMDSADMP (MFT/MVT)	
IMCJQDMP	Ausdrucken des Inhalts der Jobkettendatei
IBCDASDI	Initialisieren von Direktzugriffsspeichern, Zuweisen von Ersatzspuren
IBCDMPRS	Abziehen des Inhalts eines Direktzugriffsspeichers auf Magnetband, Rückspeichern vom Magnetband auf Direktzugriffsspeicher

3.3. Sortieren / Mischen

Mit dem Sortier-/Mischprogramm können Dateien sortiert werden, die aus geblockten oder ungeblockten Sätzen von fester oder variabler Länge bestehen. Die Ein- und Ausgabe kann über alle E/A-Geräte erfolgen, die mit der erweiterten sequentiellen Zugriffsmethode angesprochen werden können. Die beim Sortieren aufgebauten

können auf Magnetbandspeichern oder Direktzugriffsspeichern angelegt

werden. Darüber hinaus können mit dem Sortier-/Mischprogramm bis zu 16 bereits sortierte Dateien, die sich auf Datenträgern gleichen Typs befinden, gemischt, das heißt zu einer sortierten Datei vereinigt werden. Das Programm ist als Unterprogramm auch von problemorientierten Programmiersprachen aufrufbar.

4. Programmiersprachen

Das Betriebssystem OS/ES stellt dem Anwender eine Reihe von Sprachübersetzern zur Verfügung, einen Assembler für die Übersetzung von in maschinenorientierter Sprache geschriebenen Programmen sowie Compiler für PL/1, FORTRAN IV, RPG, COBOL und ALGOL 60. Beim Programmieren in Assemblersprache können alle Dienste des Steuerprogramms in Anspruch genommen und alle speziellen Funktionen der E/A-Geräte genutzt werden. Mit Hilfe der problemorientierten Programmiersprachen wird das Formulieren eines Problems unabhängig vom Befehlssatz einer ZE, das Programm kann in einer kurzen und übersichtlichen Form geschrieben werden. Damit verringert sich der Programmieraufwand. Jedoch unterliegen die problemorientierten Programmiersprachen gewissen Einschränkungen hinsichtlich der Nutzung der E/A-Geräte und der speziellen Funktionen des Steuerprogramms. Da beim Aufbau verarbeitbarer Programme die Verbindung von Programmteilen, die von verschiedenen Sprachübersetzern erzeugt wurden, möglich ist, können entsprechende Unterprogramme in Assemblersprache zur Ausnutzung der Steuerprogrammendienste hinzugefügt werden. In Tab. 4 ist eine Übersicht der Programmiersprachen zusammengestellt. Auf eine, ausführlichere Beschreibung der problemorientierten Programmiersprachen wird hier verzichtet, da in der DDR bereits umfangreiche Sprachbeschreibungen, z. B. in der Reihe Automatisierungstechnik des VEB Verlag Technik Berlin, erschienen sind.

Die Assemblersprache ist eine symbolische Maschinensprache. Für jeden Maschinenbefehl eines Programms muß ein symbolischer Befehl codiert werden. Adressen können in symbolischer Form angegeben werden. Außerdem enthält die Assemblersprache Anweisungen für Bereichs-, Konstanten- und Definitionen sowie verschiedene Steueranweisungen für den Übersetzungsvorgang.

Der Assembler erlaubt das Verwenden von Makroanweisungen. Damit werden die in einer Bibliothek gespeicherten Makrodefinitionen aufgerufen, unter denen häufig verwendete, nur wenig zu modifizierende Anweisungsfolgen zusammen gefaßt sind. Die Auswahl der aus einem Makro erzeugten Anweisungsfolge wird anhand der ihm beim Codieren im Assemblerprogramm mitgeteilten Parameter getroffen. In mit dem Betriebssystem gelieferten Makrobibliotheken sind Systemmakrodefinitionen enthalten, deren Verwendung sinnvoll ist, wenn verschiedene Steuerprogrammendienste in Anspruch genommen werden sollen.

5. Systemgenerierung

Der Gesamtumfang des Betriebssystems OS/ES wird einschließlich einer speziellen Systemkonfiguration, dem Startersystem und der systemunabhängigen Dienstprogramme auf Verteilungsdaträgern (Magnetbändern) ausgeliefert. Aus diesem Verteilungssystem kann der Nutzer ein für seine spezielle Anlagenkonfiguration und sein Anwendungsprofil effektives Betriebssystem durch Auswahl verschiedenster Funktionen generieren. Das Betriebssystem stellt für den Vorgang der Systemgenerierung spezielle Makrobibliotheken zur Verfügung. Der Nutzer formuliert für die Generierung ein Programm unter Verwendung dieser Makrodefinitionen, wobei mit Angabe spezieller Parameter die zu generierenden Funktionen ausgewählt werden. Bei der Systemgenerierung wird eine bestimmte Steuerpro

Sprache	Anwendungsbereich	Dateiorganisationsform	Sprachübersetzer	Speicherbedarf
ASSEMBLER	leistungssteigernde Unterprogramme	alle Dateiorganisationsformen	ASSEMBLER	44 K Byte
PL/1	kommerzielle, wissenschaftliche und technische Probleme	BSAM QSAM ISAM VSAM BDAM QFAM	PL/1-ST PL/1-OP PL/1-Test	64 K Byte 128 K Byte
FORTRAN	wissenschaftliche und technische Probleme	BSAM BDAM (QFAM)	FORTRAN-ST FORTRAN-OP	128 K Byte 192 K Byte
RPG	kommerzielle Probleme	BSAM QSAM BDAM ISAM QFAM	RPG	15 K Byte
COBOL	kommerzielle Probleme	BSAM QSAM BDAM ISAM QFAM	COBOL	80 K Byte
ALGOL	wissenschaftliche und technische Probleme	BSAM	ALGOL-60	44 K Byte

Tabelle 4
Programmiersprachen

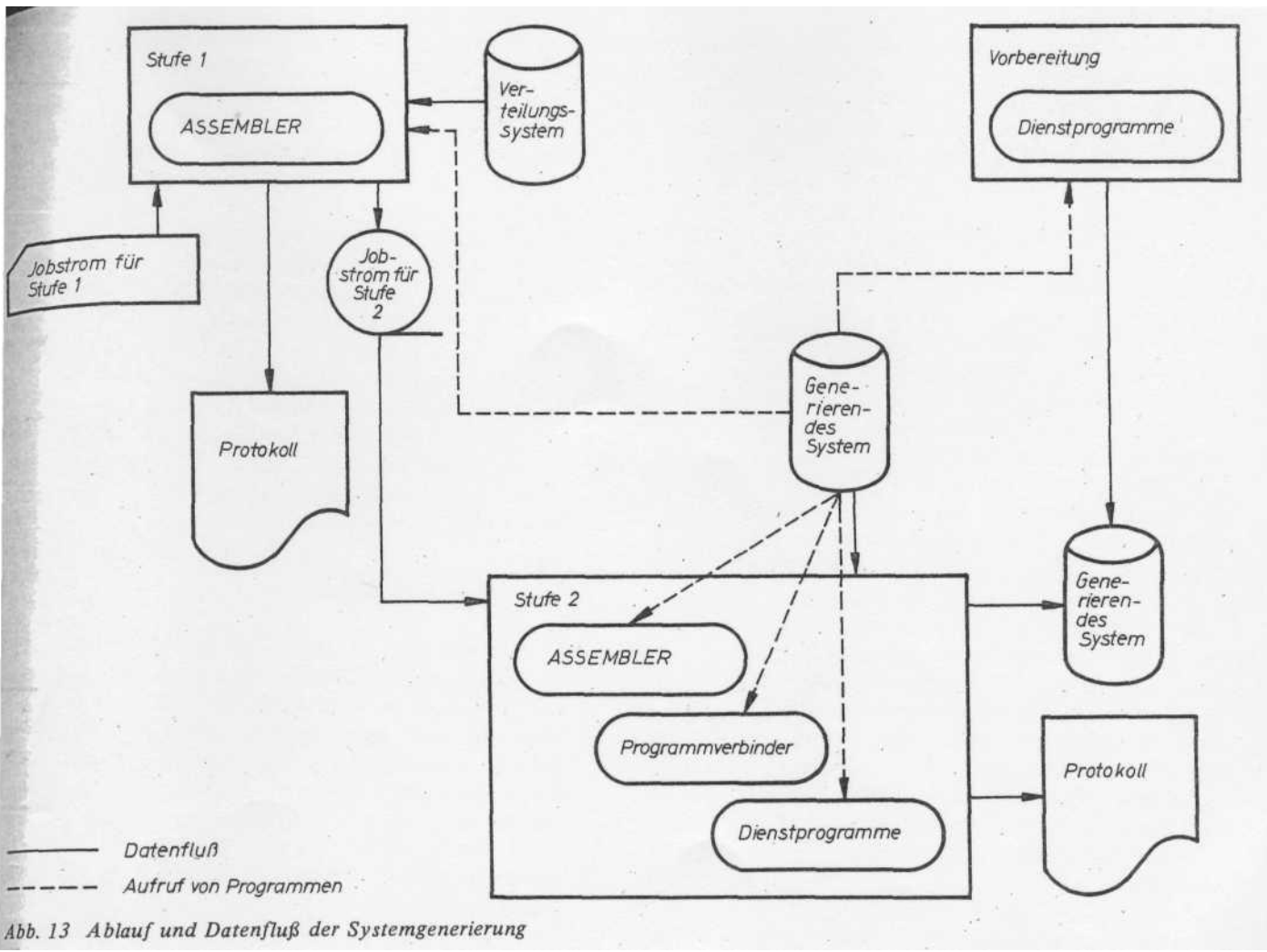


Abb. 13 Ablauf und Datenfluß der Systemgenerierung

grammkonfiguration - für EC 1055 in der Regel das SVS - ausgewählt. Darin sind eine Reihe von Standardfunktionen enthalten und andere können wahlweise eingeschlossen werden. Außerdem werden die vom Nutzer benötigten Verarbeitungsprogramme einschließlich zugehöriger Unterprogramm- und Makrobibliotheken aufgebaut.

Die Systemgenerierung erfolgt unter Steuerung eines arbeitsfähigen Systems in drei Etappen (Abb. 13):

- Vorbereiten der Datenträger, Bibliotheken und Arbeitsdateien
- Übersetzen der Makroanweisungen, mit denen der Nutzer das gewünschte System beschrieben hat; der entstehende Objektmodul ist ein Jobstrom.
- Verarbeiten dieses erzeugten Jobstroms.

Mit der Systemgenerierung kann entweder ein neues Betriebssystem erzeugt oder ein bereits vorhandenes erweitert werden. Nach dem Ziel unterscheidet man vier Arten der Systemgenerierung:

- Die vollständige Systemgenerierung baut ein neues Betriebssystem auf.
- Mit einer Steuerprogrammkernel-Generierung wird in ein bestehendes System ein neuer Steuerprogrammkernel eingefügt, wobei die Konfiguration aber erhalten bleiben muß. Bei einer Verarbeitungsprogramm-Generierung werden zum bestehenden System Verarbeitungsprogramme und damit erforderliche Bibliotheken hinzugefügt.

Die E/A-Generierung ändert ein bestehendes Betriebssystem für die Unterstützung einer neuen E/A-Konfiguration ab. Bei der Systemgenerierung sollte vom Nutzer auch beachtet werden, daß neben der Auswahl der für ihn geeigneten Funktionen auch die Verteilung der Systemdateien auf den Datenträgern der Systemresidenz die Effektivität des zu erzeugenden Systems beeinflusst.

6. Emulation DOS/ES

Das Modell EC 1055 verfügt über eine Verträglichkeitseinrichtung zur Emulation des DOS/ES unter Steuerung des OS/ES.

Mit dem Betriebssystem OS/ES wird die Emulation des DOS/ES mit einem speziellen Emulatorprogramm unterstützt, das den Status eines Verarbeitungsprogramms besitzt. Die Emulation steht in allen Steuerprogrammkonfigurationen des OS/ES zur Verfügung.

Das Emulatorprogramm belegt einen Programmbereich des OS/ES und ermöglicht die Emulation des DOS/ES-Steuerprogramms und von bis zu drei DOS-Programmbereichen. Da sowohl die DOS-Programme als auch das DOS/ES-Steuerprogramm emuliert werden, sind keine Änderungen des DOS-Steuerprogramms oder der Problemprogramme erforderlich. Für die Arbeit mit E/A-Geräten unter den Bedingungen der Emulation muß vorausgesetzt werden, daß die Geräte sowohl

vom DOS/ES als auch vom OS/ES unterstützt und diese Geräte im OS/ES ausschließlich dem Emulatorprogramm zugeordnet werden. Anstelle der Systemeingabe- und Systemausgabegeräte (Kartenleser, Drucker, Stanzer) kann die gepufferte Systemeingabe-/Ausgabe des OS/ES genutzt werden, um ein Blockieren dieser meist nur einmal vorhandenen Geräte zu vermeiden. E/A-Anforderungen an die DOS-Bedieneinheit werden auf der OS/ES-Bedieneinheit simuliert. Das Emulatorprogramm wird über eine EXEC-Anweisung mit speziellen Parametern angefordert und durch DD-Anweisungen zur Zuweisung der vom Emulator benötigten Geräte ergänzt. Es können im OS/ES mehrere Emulatorprogramme parallel zueinander laufen.

Der Speicherbedarf des Emulators liegt zwischen 24 und 34 K Byte und ist von der OS/ES-Steuerprogrammkonfiguration, den Zugriffsmethoden und der Anzahl der E/A-Geräte im DOS abhängig. Es können alle DOS/ES-Versionen zwischen VM 1.3 und VM 1.7 sowie VM 2.1 emuliert werden.

7. Autonome Testprogramme

Zum Überprüfen der technischen Bestandteile des Modells EC 1055 stehen neben den Geräte- und Mikroprogramm-diagnosemitteln, die unter der gerätetechnischen Beschreibung der EC 1055 erläutert werden, dem Wartungstechniker autonome Testprogramme zur Verfügung. Dazu gehören der Diagnosemonitor, ein Steuerprogramm für Gerätetests und das Konfigurationstestprogramm.

• *Diagnosemonitor*

Im Komplex des Diagnosemonitors sind Basistestprogramme, der Diagnosemonitor des ESER (DMES), Diagnosesektionen für ZE und E/A-Geräte und Serviceprogramme enthalten. Die Basistestprogramme prüfen ausgewählte Befehle, die Unterbrechungsbearbeitung sowie E/A-Geräte, die vom Diagnosemonitor als Ladegerät für die Testsektionen und als Kommunikationsgerät vom Techniker verwendet werden. Der Diagnosemonitor ist ein Steuerprogramm zur Verarbeitung der Diagnosesektionen. Er realisiert die Kommunikation mit dem Nutzer und enthält Standardfunktionen für die Diagnosesektionen, z. B. die Aufbereitung und Ausgabe von Fehlermeldungen, die Unterbrechungsbehandlung und die Bereitstellung von Informationen über die ZE oder E/A-Geräte.

Die Verbindung zwischen DMES und den Diagnosesektionen erfolgt durch standardisierte Anschlußbedingungen. Die Diagnosesektionen enthalten spezielle Tests für die ZE-Kanäle und E/A-Geräte. Die ZE-Diagnosesektionen prüfen den vollständigen Befehlsvorrat, die Unterbrechungsbearbeitung, die Programmereignisregistrierung, die dynamische Adreßumsetzung und die Emulationseinrichtung. Die Sektionen für E/A-Geräte testen in serieller Folge alle angeschlossenen E/A-Geräte entsprechend ihrer Funktion. Im Ergebnis der Tests werden Fehlermeldungen an den Nutzer ausgegeben. Dem Nutzer wird das DMES-Testprogrammsystem mit dem Diagnoseband übergeben. Mit Hilfe der Serviceprogramme kann der Nutzer für die Konfiguration seiner DV-Anlage entweder auf Magnetband oder Direktzugriffsspeicher einen Testprogrammumfang zusammenstellen, der von diesem Datenträger aus verarbeitbar ist.

• *Steuerprogramm für Gerätetest*

Um die E/A-Geräte zu prüfen, stehen außer den DMES Sektionen noch Sektionen zur Verfügung, die vom systemunabhängigen Steuerprogramm für Gerätetests oder dem entsprechenden Steuerprogramm unter Betriebssystem verarbeitet werden können. Das Steuerprogramm übernimmt folgende Funktionen:

- Auswahl der Geräte, Sektionen und Testbedingungen
Angaben des Nutzers
- Steuerung der Kommunikation mit dem Nutzer
- Schutz der Daten auf den Datenträgern.

Die Gerätetestsektionen führen folgende Funktionen aus

- Verarbeiten gerätespezifischer Kanalprogramme
- Test der Zustandsmeldungen
- Test der Gerätesteuereinheit.

Die Arbeit mit diesem Programmpaket ist sowohl für die phylaktische Überprüfen der Geräte als auch für die Suche komplizierter Fehler vorgesehen.

• *Konfigurationstestprogramm*

Dieses Testprogrammsystem ist für den Nachweis der Funktionsfähigkeit der ZE und der fehlerfreien Zusammenarbeit aller Bestandteile der DV-Anlage verwendbar. Da die Zeit für die programmtechnische Wartung bis zum Nachweis der Funktionsfähigkeit aller Bestandteile einer DV-Anlage mit seriell arbeitenden Programmen, wie DMES, relativ hoch ist, wird im Konfigurationstestprogramm das Verarbeiten der einzelnen Tests so organisiert, daß der Nutzer in kurzer Zeit einen Überblick über die Verfügbarkeit der Gesamtanlage erhält. Dieser Test ist durch eine möglichst hohe Parallelarbeit verschiedener E/A-Geräte charakterisiert.

Zum Konfigurationstestprogramm gehören ein Steuerprogramm, das zentrale Funktionen und die Kommunikation übernimmt, Segmente, die Testalgorithmen für die Prüfung der ZE und E/A-Geräte enthalten und Serviceprogramm die Anpassung des Programmsystems an die konkrete DV-Anlage gewährleisten.

Literatur aus dem Verlag Die Wirtschaft

Funktionsprinzipien des ESER, Reihe 2

Schriftenreihe Informationsverarbeitung.
Berlin 1979, DDR 14,50 M.