HONEYWELL

SOFTWARE BULLETIN

SERIES 200

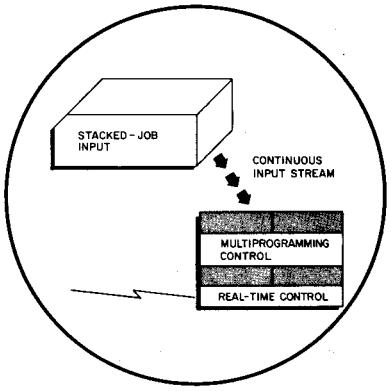
SERIES 200/ OPERATING SYSTEM - MOD 2

The Series 200/Operating System - Mod 2 is an integrated set of interdependent programs providing the most efficient means for program development and operation. The Operating System is the interface between the user and the computing system. By reducing the mutual dependence of the user and the computer, the Mod 2 Operating System increases throughput, decreases turnaround time, makes program development and maintenance simpler and more flexible, and standardizes operating procedures.

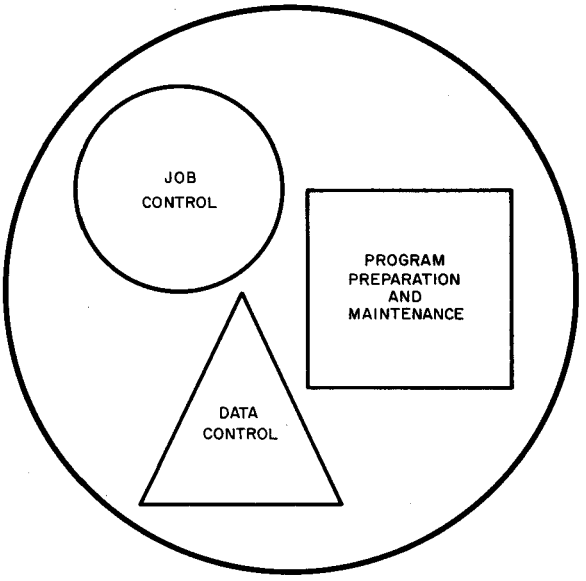
FUNCTIONS OF THE MOD 2 OPERATING SYSTEM

Job Control

Job control under the Mod 2 Operating System is handled by two monitors, whose primary function is to provide automatic job-to-job transition. One monitor remains resident in core storage at all times; the other monitor is loaded periodically to assist in the automatic transition process. Most of the human operations involved in transitions between programs within a job and between jobs in the input stack have been absorbed by the Mod 2 Operating System. These clerical duties include collecting the output produced by the previous program, locating the next program to be executed and loading it into memory, and coordinating peripheral device assignments. Job control functions are initiated by system control cards, on which the user schedules program executions and peripheral device assignments. Reducing the idle time between runs by automatic job-to-job transition is one of the ways in which the Mod 2 Operating System increases system throughput.



JOB CONTROL



Continuous processing of jobs is combined with a method called stacked-job processing, a refinement of the earlier batched-job approach. Under batched-job processing, a single processing function, e.g., compilation, is applied to all jobs in the batch. While a group of jobs may be compiled in succession and then executed in succession, program generation is divorced from the execution of a batch of pregenerated programs. Under stacked-job processing, any number of processing functions such as compilation, maintenance, and execution may be successively applied to the same job. Thus, each job in the input stack is processed to completion before the next job is accepted.

In batched-job processing, the elapsed time between the submission of a job and the receipt of results (turnaround time) is equal to the total processing time for the entire batch which includes the job. In contrast, stacked-job processing dramatically reduces turnaround time for a given job by completing each job before beginning the next.

The job control function performed by the Operating System after loading each program into memory is monitoring, which consists of controlling the internal sequencing of all programs executed under the system. At the proper instant, control is delegated to a program or retrieved from it. Monitoring facilities of the Mod 2 Operating System support operation in communication/real-time and multiprogramming environments. Monitor-

Specifications remain subject to change in order to allow the introduction of design improvements.

122.0005.002J.0-429
10666
Printed in U.S.A.

When ordering this publication please specify Title and Underscored portion of File Number.

Honeywell

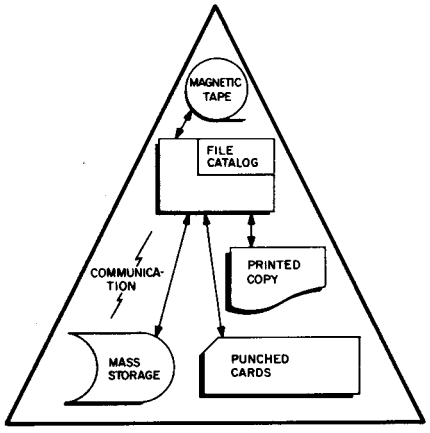
ELECTRONIC DATA PROCESSING

ing in a communication environment involves the control of message flow to and from the computer and message processing within the computer. Monitoring in a multiprogramming environment consists of supervising the concurrent execution of two programs. One program is normally peripherally limited and is executed in upper memory. The second program runs in lower memory during the peripheral cycles of the upper-memory program. This optimal utilization of the central processor and peripheral devices is another source of increased throughput under the Mod 2 Operating System.

Another job control function is communication with the operator, advising him of the status of processing and requesting necessary operator actions.

Data Control

Data control in the Mod 2 Operating System encompasses all functions related to the creation and maintenance of the data base, i.e., the entire collection of information which enters or leaves the computer main memory. The facilities available under data control provide efficient storage, flow, and retrieval of all data in the system. These facilities include two functions: file access and file control.



DATA CONTROL

The principal file access function is the physical exchange of data between main memory and unit record, magnetic tape, mass storage, and communication equipment. Resident routines execute the data transfer functions, using several different access methods. Other file access functions are associated with data transfer. These other functions are automatic error detection and correction, automatic data buffering, automatic data blocking and unblocking, dynamic scheduling of input/output facilities, and overlapping of processing with input/output operations.

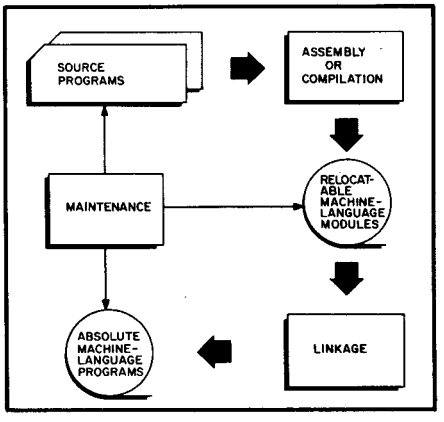
The file control function includes management of logical data files at a level which is independent of the physical characteristics of the files and their storage devices. Files are assigned symbolic names, and the Operating System maintains a symbolic file catalog. The file catalog is constructed like a library catalog, so that each file may be indexed, according to its functions, within a set of symbolic classification levels. Upon receiving a request for a file with a given symbolic classification, the Operating System consults its symbolic catalog, determines the physical identity and location of the file, and retrieves the file.

In conjunction with constructing the file catalog, the Operating System automatically allocates and partitions space for mass storage files. Storage allocation is complemented by automatic file protection.

Program Preparation and Maintenance

The most familiar program preparation function is language processing. Programs written in compiler or assembly language are translated to program modules in relocatable machine language. A program module is the basic program unit in the Operating System. Each module is created independently. Modules are relocatable and can be combined with other modules to fashion a variety of complete programs. The second program preparation function consists of assembling a complete program by selecting specified program modules, providing linkages between the modules, and assigning absolute memory addresses to the relocatable machine code. Complete programs may be built to run anywhere in core storage using any combination of modules. Also, all language processors in the Mod 2 Operating System generate the identical type of relocatable modules. Hence, a complete program may be subdivided into program modules on the basis of physical size, functional breakdown, or the nature of the source language best suited for solving a portion of the total problem.

The program maintenance functions include adding and deleting modules from the system files and correcting lines within specified modules. Maintenance may be carried out at the source-language, relocatable-code, or absolute-code level. The identical maintenance functions are applied to both system and user programs, and may



PROGRAM PREPARATION AND MAINTENANCE

be used to incorporate user-written modules into the Mod 2 Operating System.

Other Functions

The automatic debugging facilities of the Mod 2 Operating System include dynamic core and tape dumps. The data editing and transcription functions include sorting and merging magnetic tape and mass storage files and performing media conversion operations.

BENEFITS OF THE MOD 2 OPERATING SYSTEM

Ease of Programming

The relocatable program module is the common denominator of the Mod 2 Operating System. Because they are relocatable, all modules are essentially library routines which the Operating System can freely combine. Free communication between program modules is maintained through the standard interface of the Operating System. Because all language processors generate the same basic building blocks (program modules), a programmer is not limited to solving an entire problem in a single source language.

Responsibility for tedious and complex input/output programming is transferred from the user to the centralized routines of the Operating System. Programmers need not be conversant with the unique programming characteristics of any peripheral devices. In addition, the Mod 2 Operating System automatically frees the user from allocating buffers, checking file labels, blocking and unblocking records, and error-checking data transfer operations. Also, the Operating System ensures continuous use of the system facilities by maximizing the simultaneity of data flow and internal processing, a capability which is inherent in Series 200 hardware.

Controlling the flow of data to and from peripheral devices is just part of the device independence provided by the Mod 2 Operating System. The Operating System also manages the logical data files themselves. The symbolic file catalog within the Operating System allows programmers to request data files by using only their symbolic names. The mechanics of locating and retrieving data files are the responsibility of the Operating System. The Operating System also controls space allocation and formatting on mass storage devices.

Finally, the standardized and automatic debugging facilities of the Mod 2 Operating System, coupled with the brief turnaround time per job, enhance the ease and efficiency of program checkout.

Ease of Operating

A common set of operating procedures is superimposed on both user-written programs and components of the Mod 2 Operating System. Operators do not have to cope with the peculiarities of every program, a fact which simplifies operator training and increases the reliability of machine room operation. In the same fashion, man/machine communication is reduced to a standard dialogue between the operator and the Operating System. Most functions required for automatic job-to-job transition have been programmed into the Mod 2 Operating System.

The Operating System also administers internal hardware facilities such as the interrupt system and storage protection, further simplifying the role of the operator. Standardized operating procedures make the mode of operation sensitive to the requirements of each application. Stacked-job processing, batched-job processing, and real-time processing are handled with equal facility.

Ease of Maintenance and Expansion

Both user programs and Honeywell-supplied components of the Mod 2 Operating System are easily modified because of their modular structure. A series of complex, time-consuming programs is required initially to generate some operating systems. However, the same single-phase component of the Mod 2 Operating System which is used to update the system files is also used to create a working version of the Operating System itself. System generation is both selective and efficient. A personalized operating system is tailored to each installation by incorporating only those system modules required by the user. A typical business-oriented version of the Mod 2 Operating System is generated in less than 15 minutes. System programs and user programs are easily updated without recompiling. For example, additional modules may be added to user programs to take advantage of newly acquired hardware. Additional modules may be added to the Operating System to provide further processing capabilities for growing applications. Also, the Operating System may be expanded by the inclusion of user-written components.

Over-all Benefits

From the perspective of the data processing manager, the convenience and modularity at each level of the Mod 2 Operating System are reflected and amplified in the over-all efficiency and reliability of the hardware/software complex. Standardized programming and operating procedures provide the most efficient path from initial formulation of a programming problem to final utilization of the solution. Total hardware utilization through multiprogramming and reduced idle time because of automatic job-to-job transition increase throughput under the Mod 2 Operating System. At the same time, the stacked-job capability provides comprehensive software service with minimal turnaround time. The flexible framework of the Mod 2 Operating System supports growth into applications such as total information and real-time systems. The magnitude and complexity of the functions performed by the Mod 2 Operating System simplify the jobs of programmer, operator, and manager. By furthering the independence of these personnel from the computer, the Mod 2 Operating System allows them to use it more effectively.

MINIMUM HARDWARE REQUIREMENTS

1. A Series 200 Model 1200, 2200, or 4200 processor with 49,192 characters of core storage and the Optional Instruction Feature (0191).
2. 5 Type 204B Magnetic Tape Units and tape control equipped with the IBM Format Feature (050) and the IBM Code Compatibility Feature (051), **or** 3 tape units and 1 Mass Storage Transport.
3. 1 Type 223 or 214-2 Card Reader and control, **or** 1 additional tape unit.
4. 1 Type 222 Printer with 132 print positions and control, **or** 1 additional tape unit.
5. 1 Type 220-3 Console.

HONEYWELL
ELECTRONIC
DATA
PROCESSING
WELLESLEY HILLS,
MASSACHUSETTS 02181

CRP