# DISC FILE
## APPLICATIONS

*Reports Presented at the*
*Nation's First Disc File Symposium*

Sponsored by Informatics Inc.,
Culver City, California

Data Processing
Library Series

Library of Congress Catalog Card Number: 64-23108


22nd Floor, Book Tower, Detroit 26, Michigan

# CONTENTS

**by W. F. Bauer**
**Informatics Inc.**
**Culver City, California**

# Disc File Symposium Objectives

IT IS NOT OFTEN in data processing that a development occurs as important as the disc file. In the 1950's the introduction of magnetic core memories made truly high speed internal storage a sudden reality, allowing a manyfold increase in speed of calculation. Also, the high capability magnetic tapes with removable reels had an important impact in allowing flexibility through extensive auxiliary storage and convenient off line operations. Many are of the opinion that the modern disc file will have an equal effect in providing a high capacity extension of internal random access memory, and in implementing large scale, on line data gathering systems. It is the purpose of this symposium to study and discuss this new instrument to gauge better its impact on applications and to understand better the economic and technical factors involved in its use.

Disc files are making possible large randomly accessible files such as those in advanced business systems. They are making possible scientific computer systems where nearly all needed data are immediately accessible to the machine, offering the opportunity to develop systems with great operational flexibility. They are making possible extensive on line systems such as reservation systems and store-and-forward communications systems. They are seeing increasing use as connection and buffering units in multi-computer systems. Perhaps most significantly it appears that disc files will seriously challenge tapes as the basic auxiliary storage for many hundreds of medium and large scale systems, a point of very great economic significance.

This symposium is concerned principally with disc files with movable heads. It is concerned with disc files of the so-called new generation files with capacities in excess of 20 million characters and average access times as low as 100 milliseconds or less. As part of our program, however, we expect lively discussions of disc files compared with drums, tapes, and fixed-head files.

To those who have been only briefly introduced to the subject of disc files, this storage device may appear either as simply an extension of magnetic tape or as a device strictly limited to applications such as inventory control, where random accessing plays an important role. However, respect for the complexity and significance of disc files grows rapidly with increased exposure to the subject. The obvious characteristics of disc files – random access and high transfer rates – tell only part of the story.

The configuration of data within the file is flexible and has important interplay with programming and application. Random access capability frequently requires new data handling design and opens up new applications. Arm motion, latency, verification, and read/write phases are of importance to the designer and user. These are but a few of the considerations of this new device. The number of parameters available for adjustment or required for definition by the programmer and the designer is probably an order of magnitude greater than with magnetic tapes.

## MAJOR AREAS OF DISCUSSION

It is the ambitious objective of the symposium to cover nearly all aspects of disc files during the two-day period in some 15 papers and a panel discussion. The presentations can be regarded as being in four major areas: applications, programming, data configuration and storage, and file characteristics.

Since the symposium is designed to disseminate information it seems appropriate to introduce the technical subject matter by presenting some questions to which the speakers on our program address themselves.

Concerning *applications*, typical questions which arise are:

- What kinds of business systems require or benefit from disc file usage?
- What are the principal advantages of random access memories in large scale scientific systems?
- Why do store-and-forward and message

switching systems in communications work need disc files?

- What new application areas are likely to emerge?
- What is the relative importance of the disc file application areas of auxiliary storage, on line data storage, and buffering-interface?

Disc files raise important questions regarding *programming:*

- What are the characteristic programming problems and techniques in disc file input/output control systems?
- Will modern operating systems employing monitors and compilers work more efficiently with disc files than with tapes?
- How can compiler and translator algorithms be designed to take advantage of random access auxiliary storage?
- What techniques are possible for look-ahead and storage overlay using disc files?

There are also important questions on data *configuration* and *storage:*

- What are the addressing techniques used in a given file and what is their relation to programming techniques such as "address randomizing"?
- What is the relative importance and use of such items as "cylinder" recording, parallel transfer, and fixed heads?
- Are there techniques or principles of data configuration and data file organization which reduce access time and what are they?
- What are the hardware and software techniques for error detection and recovery?

Underlying all of the technical aspects are the *file characteristics*:

- What is the head-arm configuration and how does it affect the user?
- How is the file controlled and how much flexibility is available to the user?
- What is the precise definition of "access time" and "average access time"?
- What are the design factors involved in the selection of such items as track width, head-arm configuration, and number and characteristics of recording zones?
- What are the physical-environmental requirements and characteristics?

It is the hope of the symposium organizers that questions such as those above will be answered or at least appropriately illuminated. The speakers were carefully chosen on a nationwide basis as leaders in the field who have been engaged in disc file work. We believe that their lucid presentations and well-organized papers represent an important contribution to computer technology.

by A. J. Perlis
Carnegie Institute of Technology
Pittsburgh, Pennsylvania

# A Disc File Oriented Time Sharing System

THE DIGITAL COMPUTER is now more than a computing device. It is a system — and all computer people recognize that. Indeed we all recognize that a "computer" will ultimately be the center of an automatic information transmission system, one of the least critical of whose parameters is the distance between some of its parts, e.g., input/output stations. Another characteristic of such systems is increasing heterogeneity: of performance requirements (speed, memory, input/output), of programming languages (ALGOL, IPL, etc.), of duration and frequency of computer contact per program submitted.

In short, as the demands on, and of, the sensory units of the system become more heterogeneous and, by units, more specialized, the central processor (including its programs) develops through a process of abstraction. By this I mean sets of previously central tasks are combined, parameterized, and made purely syntactic to a new level of central task which is both more specialized and more generally applicable. However its usefulness is only meaningful in an environment requiring greatly extended maintenance of both hardware (which however is becoming more reliable) and programs. The development of such a system seems to make necessary large random access files, particularly because of non-scheduled computer use controlled from a distance over communication lines of low capacity. Actually the capacity of the communication lines is not terribly critical; consider the two extreme cases:

1. Low capacity (200 bits/second). Then storage requirements of a problem could be so great that low channel capacity prevents transmission of required data.

2. High capacity ($10^6$ bits/second). Here it is a question of storage economics: $n$ distributed storage units of $m$ characters versus one unit of $\geqq nm$ characters.

Consequently the two main conditions leading to large random access files are:

A requirement for fast response time;

Random arrival times for requests for system functions.

The preceding remarks are, by now, recognized platitudes. We are interested, in the discussion that follows, in considering a particular system, the mode of its operation, the parameters that determine its effectiveness, and some value assignments to these parameters.

## SPECIFICATION OF THE SYSTEM

The system consists of:

1. A computer with internal storage of 32 + K words, each of four characters, six microsecond memory access time.

2. A certain number $n \leqq 6$ magnetic tape units, read-write transfer rate 120K characters/second, and search rate 220K characters/second (at normal recording density).

3. A disc file having a total store of 30 x $10^6$ characters, an average access time of 200 milliseconds, a restricted average access[1] time of 32.5 milliseconds, and a transfer rate of 167K characters/second to and from the computer.

4. A buffered in-out card/printer system with read speed 48K characters/second and print speed 120K characters/second.

5. A collection of $N$ remote input stations each having a 96-character buffer and an input/output speed of 10 characters/second and a transfer rate of 20K characters per second to the computer.

That describes the major hardware units. Of equal importance is the specification of the major programming units in the system. The operating environment is that of the now classical open programming and closed center operation shop. The problem distribution environment is that of a university. The programming languages environment is that of a language of the ALGOL class with the addition of some machine coding and symbol manipulation being provided via additional languages. Actually, work is in progress

[1] Average without head movement.

at Carnegie to imbed both a LISP-like language and machine assembly code into ALGOL so that a majority of the computing functions can be processed within one language framework. In this way some part of the debugging function might be the systematic symbolic replacement of pure ALGOL by assembly language ALGOL.

The major control programming units are the monitor and the executor systems. It is important to distinguish between them. The monitor system controls the computer and programming operations required in processing a program admitted to execution; the executor selects which jobs to process and how to decide their continuation and termination conditions, and accordingly bookkeeps their history. The various language translators are treated as adjuncts to the hardware and so are controlled by the monitor. No programmer has access directly to any of the input/output.

One of the major functions of the monitor is its interrupt service routine. This routine is executed whenever any of the hardware or program functions shift into an interrupting (interrupting the normal computer fetch-execute operation) state. The number of such situations that can arise is of course a function of the hardware and the state of the environment. In the case of the G-20, interrupts arise both from executed program requests, from a real time clock, from abnormal arithmetic and control states, and from conversations being carried out on the communication lines which connect the G-20 to a variety of peripheral units.

The executor routine is responsible for scheduling. Various models have been proposed. One such that appears to be ideal for the environment involved is described by Dr. Corbato and associates.[2] It is possible to go overboard on the scheduling problem but a simple algorithm like the one suggested is probably adequate.

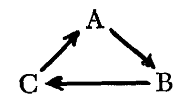
## SOLVING A PROBLEM ON THE COMPUTER

Under ideal circumstances solving a problem on a computer is a sequence of conversations between the interior systems and the external programmer concerning successive modifications of a program. We might list some standard communications:

1. A program is initially being entered for execution and/or internal storage.

2. Data is being entered to be used in the ultimate execution of an already existent internally stored program.

3. A program is being altered for running. Fundamentally all are the same control problem. They really differ only in how much previously entered information is to be used in subsequent stages of problem solving. To see how this may be done we might consider the initial input of a program into the system. Let a program be entered from a remote station. Then the program may be considered part of a trinity:




*A* defines the executive statements specifying the disposition of the operational program text *B*. *C* defines the control actions to be taken on *A* (and *B*) as a consequence of the execution of *B*.

The execution statements may be part of a quite extensive language structure. A simple one is the so-called AND (Alpha-Numeric Director) System used at Carnegie Tech.[3]

It is a routine connected with Carnegie Tech's operating system for the G-20. It permits the semi-permanent storage on tape or disc of symbolic records of users' programs, and the changing of these records. The system permits a user to request an "assembly", in any available language of an arbitrary mixture of symbolic information previously stored and entering information. The merged information thus created may be written back into storage, either replacing the original information or being appended as a new record. Then, the newly created program may be assembled by a language processor and executed.

Let us consider briefly a simple example of the use of AND. A programmer writes a program in ALGOL. He precedes it by a job statement specifying AND as processor and code interpreted by AND. AND reads the input of the user's program and stores it, assigning space for this particular user and giving the space a name assigned on the basis of the programmer's usage number. If at the end of the message there is an instruction directing AND to execute the resulting program, ALGOL will be loaded into core by the monitor and the resulting program will be translated and run as usual.

Let us assume that, as a result of this run, the programmer detects an error. He finds that the error can be corrected by inserting two lines and

[2] By Corbato, Fernando J., Marjorie Merwin-Daggett and Robert C. Daley, "An Experimental Time Sharing System", *AFIPS Proceedings of 1962 Spring Joint Computer Conference*, pp. 335-344.

[3] AND was designed by the author and A. Evans, Jr. The coding is the work of Alex Nesgoda.

deleting another one. He then prepares a code specifying the name of the location in storage where his previous program was stored, and containing suitable instructions directing AND to load in this previously read-in program, make the insertions and deletions indicated, and write the program back out onto storage. The resulting program is then executed. The programmer may continue this operation as many times as he desires. The advantages are obvious: for each subsequent assembly after the first, only correction lines need be provided.

AND keeps on file a directory, referred to hereafter as AND-DIR. Programs kept in the AND system are referred to by the usage number of the programmer who created the program. Each usage number may have as many programs associated with it as desired, up to 127. Thus a programmer refers to a given program on tape by specifying the usage number used when the program was stored and specifying the program number. AND-DIR has an entry for each currently used usage number. This entry leads to a list of programs currently in operation under that usage number. This list of programs then leads to the appropriate place in the permanent AND records.

An AND run, in general, consists of three phases of operation. In the first phase, a record of the program to be run is created on a work area of disc. This record is an appropriate merge of previously created AND records and line images currently read in. Thus, most of the input to AND will consist of information used to create in the work area a program to be run.

In the second phase of operation (which may be absent) the contents of the work area may be written back onto disc in the AND area as a standard AND record. The created program may either replace an already existing program or it may be assigned a new number and appended to the AND file. Finally, in the third phase the program in the work area may be compiled by any translator currently available at Tech.

Thus, we have the creation of a program about to be run and its storage in the work area, the ability to store this program back into AND records if desired, and finally the ability to compile and execute the resulting program.

The operation of AND is controlled by a very simple but adequate coding language. There are two instructions for accessing the directory for this purpose: USER and PROGRAM. USER has one parameter: a usage number. PROGRAM has, as its parameter, an unsigned integer, the program number.

## PROGRAM INSTRUCTIONS

There is a set of instructions for the creation of a program in the disc work area. It should be remembered that there are two sources of information in creating such programs: previously stored AND records and new input of symbolic information. The pseudo TEXT directs AND to take information from input, and to store it, in the order input, into the work area. It is terminated by a line marked AND in the language field. A pointer, delta, refers to sequence numbers labelling the lines in the AND record. There are several instructions which change the value of delta. The simplest of these are: 1) SET, with one parameter: an unsigned integer. Its effect is to set delta to the value of the integer; and 2) GET, also having a single parameter which is an unsigned integer. Its effect is to get, from the currently specified AND record and starting at line delta, that number of lines specified by the parameter. Delta, of course, is incremented by the appropriate value.

It happens that the instructions given already are adequate for the purposes of AND. However a few additional instructions are also available. For example, note that it takes a SET and a GET to fetch information from storage. The same effect is achieved by using LOAD which has two parameters, both of which are integers, separated by the word TO. For example, LOAD 103 to 157 is equivalent to the two instructions SET 103; GET 55. Delta will be left at 158. Another is INSERT AFTER, having as its parameter an integer. The notation INSERT AFTER 87 (which has meaning only if delta is less than 87) has the effect of GET-ing records starting at the current value of delta up to and including record 87. Delta is left at 88. INSERT AFTER will normally be followed by TEXT.

Another convenient instruction is DELETE, which may have either of two types of parameters: integer, or integer TO integer. The notation DELETE 13 to 17 has the following meaning (again it is assumed that delta is less than 13):

Records from the current value of delta up to and including record 13 are added to the current scratch area. Delta is then set to 18. The operation of DELETE when the parameter is a single integer should now be clear: only one record is deleted. The instruction CLEAR initializes the pointer in the work area.

A very important instruction is DUMP, which writes the work record just created back into the permanent AND records. The name subsequently to be used to refer to the record just dumped is the same name as given by the last USER and PROGRAM instruction. For protection of records, a restriction is introduced. When DUMP occurs, the usage number specified on the last USER instruction must agree with the usage number which appeared on the job statement for the program. The effect, then, is that only a given programmer may write over his own program on AND records. (Of course, any programmer may *use* any other programmer's AND records in creating his own program.) A final instruction is RUN. The parameter to RUN is the name of a system, such as ALGOL or IPL, etc. The effect is to load the specified system which will then take as input the work record just created.

It is usually the case that a program fails to run to correct completion the first time entered. The portion C gives the control needed for execution of error checking and error correcting runs. While several approaches are possible, the one principal approach which seems to the author to be necessary regarding the specification of C is:

The program language C must be at least as powerful as the combined languages A and B.

Indeed, the obvious language C for ALGOL input is a modification of ALGOL combined with the language A, an example of which is AND. Depending on the sophistication of the programmer, debugging a program combines the operations of monitoring and modifying. By monitoring is meant the checking of in-run conditions and (usually) the selective printing of intermediate results at the remote station.

The format for intermediate output should be as flexible as for problem output and can be easily arranged through print statements imbedded in the program text B. Similarly the internal control in B during intermediate output can be a combination of: continued computation with buffered output or stop and go operation under remote station control. The former is a special case of the latter.

## PROGRAMMER ACTION

Upon the completion of a run the programmer is in possession of some output, a translated program and a symbolic program within the computer. What can he do? It is proper to assume that the programmer has the option of distinguishing the termination of his run from the termination of his conversation with the computer system. Consequently on completion of the program (i.e., it executed a "HALT" statement) control is returned to the programmer but the terminal status of his program is maintained (at least for some time) for post-mortem information. Actually at this stage he can but inquire about the ultimate contents of his storage dynamically existent at HALT time. This he does by a program in the source language of B which operates on the data storage extant. He now has more printed information. While certain errors have now been identified and located, others are known only by their presence; however, their exact nature is not yet known. A subsequent run is now to be made with additional control being provided. The programmer does not necessarily have before him a complete *bona fide* listing of his program and, in any event, his subsequent actions are likely to invalidate previous *bona fide* listings. He may, of course, now modify his source program (by AND) and request a new listing. This is a time consuming operation for him and the queue at his station but not necessarily onerous for the central control since one remote station is now locked in a routine task. In any event his modifications, intended to isolate some and remove other of his errors, are entered and the program re-initiated.

## DEBUGGING INFORMATION

There are certain kinds of debugging information awkward to obtain by mere modifications of the text because they require distributed lexicographic inserts to describe single supervisory control functions over the state of storage. Examples of these are:

if a $\geqq$ 25 print (x, y, z, the line number), mark to the remote station.

Here the sense of the inequality is perform these actions *whenever* $a \geqq 25$. Now the ALGOL block structure gives a rational interpretation to the nature of these statements and hence how to specify them. They are precisely block declarations and their range of (temporal) validity is that time spent in executing the block in which they are declared. Thus, for example,

*state if* $a \geqq 25 \wedge A[i, j] = 0$ *then begin* print (x, y, z); print *line*; *mark* remote [k] *end*;

causes the stores into *a* and A [i, j] to be interpreted during running and the actions indicated taken. Print *line* refers to a line number inserted by AND. *Mark* remote [k] causes the program to suspend and return control to the typewriter whose label is the current value of *k.*

More precisely the syntax of *state* is:

<conditional element> :: = <if clause> *then* (<state list>) *else*
(<state list>)

<*state* list element> :: = <variable>|<conditional element>

<*state* list> :: = <state list element>|<state list element>, <state list>

<*state* statement> :: = <state list>|<state list> → <statement>

<*state* statement list> :: = <state statement>|
<state statement list>; <state statement>

<*state* declaration> :: = <state statement list>;

The semantics of *state* is: Each variable listed in a state list will, whenever stored in an assignment statement in the enclosed block, be followed by execution of either an automatic printing of its value and name if it occurred only in a state statement of the form "<state list>"; or by execution of each of the accompanying statements, in order listed, in which the variable occurs in a statement of the form

"<state list> → <statement>".

Naturally output to a remote station will be buffered and only a transfer of control to the remote will terminate the execution of the given program.

By means of the *state* declaration the programmer is now able to debug in source language (A U B).

**by Lowell N. McClung**
**The Johns Hopkins University**
**Applied Physics Laboratory**
**Silver Spring, Maryland**

# A Disc Oriented IBM 7094 System — A Summary

THE COMPUTING MACHINERY CONFIGURATION described in this report is the instrumentation of a research program, the purpose of which is the implementation of an information processing system that puts staff members who require computing support, and the machinery that is required, in direct communication. While carrying out this purpose several problems arising out of current hardware and programming systems are being solved. The Johns Hopkins University Computing Center is a technical group whose purpose is to use the computing workload originating at the Applied Physics Laboratory, the Homewood Campus, and the Medical School as data for developing information systems that are effective research tools.

## DESCRIPTION OF PRESENT SYSTEM

The computing complex currently installed within the University Computing Center consists of an IBM 7094 and two IBM 1401 computers located in Howard County, Maryland. The Homewood Campus in Baltimore and the Medical School in East Baltimore both have IBM 1401 computers. The 1401 computers at the Medical School and the Homewood Campus are connected to one 1401 computer at Howard County through IBM 1945 data transmission units by a 105KC (15,000 characters/second) data link, 18 miles of which is microwave. The 1401 computers at the remote locations are used as input/output processors that accumulate and pre-process computer programs that are transmitted to the Applied Physics Laboratory in Howard County for 7094 processing. The 1401 pre-processing programs are really editing programs that check the grammar of FORTRAN statements while going card to tape. Since so many of the programs originating at the campus are written by students and inexperienced people, grammatical errors are of some concern. This mode of operation allows a rapid response during checkout and doesn't unnecessarily clutter up the 7094 operation. One of the peripheral support 1401 computers is connected to the 7094 by a tape switching device. The other one is used for transmission, overflow from the other, and for running some programs written specifically for the 1401 of a non-support nature.

The principal compilers for the system are the SMASHT version of the Share Operating System and FORTRAN II Version 2.

## PROBLEMS WITH PRESENT SYSTEM

1. The University Computing Center processes about 500 jobs per day. These programs are generated by about that many technical staff members scattered about the various campuses and buildings of the Applied Physics Laboratory. Accumulating the walking time to and from the computing center of staff members just within the Laboratory places about 50 technical staff members in the halls for their entire working day carrying some one-half ton of cards. This is, at least, an unscientific use of technical talent.

2. Of these 500 jobs per day, by far the largest percentage are FORTRAN-type jobs. Anyone who has gone through a 7090 to 7094 conversion is well aware of the obsolescence of the FORTRAN compiler for this series of equipment. For the kind of computing done at APL there has been a gain of a factor of two in the execution time of programs written for the 7090 and run on the 7094. There is no corresponding gain during compiling or execution of jobs that have much input or output. The higher speed 729 model 6 tape units help some but not enough. Many of these FORTRAN jobs are so short that printing the on line commentary exceeds the execution time of these jobs. In some cases the printing of the I.D. card exceeds the running time. Removing the on line commentary from the system helps; however, some on line information is required to maintain control of the system in its present form.

The situation with regard to the FORTRAN compiler would not be intolerable if the compiler were modifiable. It's not that it can't be done, it's just impossible. The compiler is written in such a way that the few modifications it needs to correct its major ailments cannot be incorporated in any way short of a complete rewrite of the compiler. The compiler is at least three years behind the hardware developments.

The hardware has some noticeable deficiencies. The most conspicuous is the on line 716 printer. The only justification for its retention in the system, as I understand, is that it is required for the engineering diagnostics.

Magnetic tape units are the weakest part of the system. Some 90 percent of the number of unscheduled maintenance calls on the APL 7094 are for magnetic tape unit failures. A corresponding percent of the total maintenance time is devoted to repairing magnetic tape units. The largest percent of the "failure to run" situations are traceable to bad physical tape.

3. Computing at the University, as well as elsewhere, has increased the complement of supporting staff personnel. This is a situation that should prevail only so long as the state of the art leaves no alternatives. The construction of an information system that makes the hardware available to staff members directly has an early effect of replacing keypunch operators, clerks, and reducing the number of machine operators. It will have a long range effect of de-emphasizing programming as a profession.

## DESCRIPTION OF PROPOSED SYSTEM

Since the early days of computing we have had computers with fair to excellent main frames with dangling input/output gear. This system proposes to change the conception of central processors with input/output gear to a concept of central memory with peripheral processors where a peripheral processor can be, for example, one or more 7090, 7094, 7044, 7040, 1410, 7750, and other computers and communication equipment. The first model to be installed at the University Computing Center will consist of four 1301 disk files for central memory. One 7094 main frame will be connected to central memory by two 7909-7631 channels. One 1410 input/output processor will be attached to central memory on one channel. The 1410 will have one 1402 card read punch unit, two 1403 printers and six 729 model six tape units. Some tape units are kept in the system to communicate with the outside world and to allow processing of test data available on tape. At present, for a lack of no better way to do it, the video communication equipment will be connected to the 1410 tape channel.

## DESCRIPTION OF COMMUNICATION EQUIPMENT

Communication terminals will consist of commercial portable television receivers with an auxiliary keyboard. System output will be displayed on the scope at a rate of 500 characters per frame and the frame speed is up to the viewer. Input to the system will be by keyboard displayed on the scope. Handwritten character input readers are in development. Properly constructed messages are transmitted to the buffer in the gray box that sits on the 1410 tape channel. Each of the 50 communication terminals planned for the first try are individually addressable for message traffic.

## OPERATING SYSTEM PHILOSOPHY

The operating system for this configuration of hardware is being constructed at APL. The interim system, operating by June of 1963, consists of an integration of FORTRAN II Version 2 and a SMASHT version of the Share Operating System, an execution program, task scheduler, and a storage allocation program.

Each processor runs independently of all other processors and is not aware, nor needs to be aware, that there are any other processors in the system. All communication and assignment of tasks take place in a message center called the task scheduler. The program library, all active task programs and the system reside in central memory. Active programs are called up by order from either the 1402 card reader or communication terminals. Output is sent to the requested output medium. Changing the number of peripheral processors does not require modification of the operating system.

by William F. Cervenka
Socony Mobil Research Laboratories
Princeton, New Jersey

# A Computer System to Minimize Expected Turnaround Time

It has long been recognized that computers can be extremely effective research tools; however, much of their potential usefulness is not achieved due in large degree to the lack of rapid, concise communication between the user and the computer. That most current computer systems fail to provide adequate communication facilities can be seen by the delays, typically hours, in obtaining a few minutes of computer time, and also by the dependence on human operators to prepare and initiate each computer run.

The computer system described in this paper provides rapid turnaround for short computations and provides concise communications by having the computer perform almost all of the program handling formerly carried out by humans. This system, which does not include tape units, is practical because of recent hardware developments in random access disk units and, of course, automatic program interrupt capabilities of computers.

## RAPID TURNAROUND

The need for reducing the turnaround time of jobs is an old one. Various treatments of this problem range from breakpoint insertion and multi-programming to presorting of jobs, and express runs. However, these proposals, while adequate for many kinds of computing (especially those that can be scheduled in advance) are grossly inadequate in operating computers as a research tool. The research scientist typically operates serially, i.e., he follows a train of thought in some logical sequence. It is often inefficient, or even impossible, for him to schedule his future activities. Thus, waiting for the results of a computation may force him to divert his attention to other, less fruitful areas. Through the use of new hardware, notably random access disc units, our system has the computer itself perform the interrupting and rescheduling necessary to turnaround short jobs quickly, while not seriously affecting longer computations.

This system has three advantages over previous solutions to the turnaround problem. These are: 1) The computer is always working on the job expected to be completed earliest, thus minimizing the average turnaround time; 2) Many human operations are eliminated; 3) Although the system is designed mainly to reduce the turnaround time of short jobs, the delay for long jobs may also be reduced because of a reluctance to begin a long computation which would later have to be interrupted.

The reduction of turnaround time is accomplished by a scheduling program that controls the flow of jobs through the computer. When a new job arrives, it is stored on the random access disc units. The computation in progress is then interrupted and the scheduling program determines which job, awaiting execution, could be expected to be completed earliest. If the interrupted job is still at the top of the schedule, its execution is resumed and it continues as if it had never been interrupted. If the new job is now at the top of the schedule, the interrupted job is stored on the disc units and the new job is started. This job is continued until it is interrupted by another new job, an error condition is encountered, or the job is finished. When a job is finally completed, the scheduling program removes the job from its queue of jobs, and initiates the next job on its list. In particular, the last computation, if any, that was interrupted is reloaded from the disc units and continued as if it had never been interrupted.

In this computer system the disc is used as an intermediate storage for both input and output information. The job input, which has just been described, includes data input that is to be used by the object program. The printer and punch output is stored on the disc in output queues as it is generated by the object program. These output queues are scheduled independently of, and simultaneously with, computing by a similar scheduling program.

## CONCISE COMMUNICATIONS

Many monitor systems and problem-oriented translators have been developed to ease communications between the programmer and the computer. However, program debugging still requires that the program be repeatedly communicated to the computer as the programmer attempts to locate and correct errors. In addition, between runs a human effort is required to make changes in the program. These changes, whether made by the programmer, or by the computer operator under instructions from the programmer, easily lead to errors and, thus, wasted computer runs.

Our operating system contains a Job Editor Control (JEC) program which provides the programmer with an automated program library. Through the JEC all programs are kept in a library on the disc units, and the programmer is provided with facilities whereby he can store, retrieve, modify, and execute programs in this library. Three special advantages of the JEC are: it maintains both symbolic and binary programs in the library; it eliminates much redundant communication required to describe and execute a job; and it performs almost all of the human operations previously required, and, in fact, makes possible some operations that cannot be performed by humans with a reasonable degree of reliability.

The programmer communicates with the JEC by the use of control cards. He must first have the JEC place his program in the library. The programmer can then retrieve the program for modification, execution, or both. Modifications can be either temporary or permanent. Temporary modifications are effective only for the job during which they are submitted and are used to great advantage during debugging. Permanent modifications are effective for the current and all future uses of the program. They are most often used to correct errors that have been located. When, for any reason, a program requiring no temporary modifications is compiled, the JEC places a copy of the binary program in the library. If a job does not require modification of a particular program, the JEC will use the binary copy of the program. When permanent changes are made to a symbolic program the binary copy of the program also reflects these changes.

The use of random access disc units to store programs is required for the efficiency of the JEC library operations. By this use of disc units, the computer time which the JEC requires per job is very small. In fact, the JEC may actually save computer time if wasted runs due to human errors are considered.

## SUMMARY

The use of discs instead of tapes for auxiliary storage is essential to the solution of two critical problems: 1) the time taken to interrupt a job, which must be kept as small as possible; and, 2) maintaining a library of programs, which is a random process.

by B. G. Carlson and E. A. Voorhees
University of California
Los Alamos Scientific Laboratory
Los Alamos, New Mexico

# Use of the Disc File on STRETCH*

In order to appreciate the environment in which the Los Alamos disc file system operates, it seems well to review the general characteristics of the STRETCH (IBM 7030) computer as installed at Los Alamos. The computer consists of a central processing unit connected to 98,304 words of 2.18 microsecond core storage, each word consisting of 64 information bits plus eight check bits. An instruction may be either a full word or a half word. The floating point add time (48 bit mantissa) is 1.5 microseconds and the floating point multiply time is 2.5 microseconds.

All input/output, other than disc operation, is controlled by the *exchange*. Our exchange consists of eight channels to which are attached eight control units to which, in turn, are attached 12 tape units, one printer, one console and typewriter, two card readers, and one card punch. All eight channels may be operating concurrently with the central processing unit and the disc file.

The two disc files (IBM 353) cannot, because of their greater speed, be connected to the exchange. They are attached to a higher speed exchange referred to as the disc synchronizer (IBM 7612). Each of the disc files is connected to the disc synchronizer on a separate channel, but only one channel can be transmitting information at a given time. The disc file on the other channel, however, during such time, may be positioning its arms in anticipation of its next data transmission activity.

Both the exchange and the disc synchronizer transmit information directly to and from the core storage. Input/output requests are issued from the central processing unit and the exchange or disc synchronizer from this point continues the input/output operation until completed or terminated. The requesting instruction (read or write) contains, along with other information, the location of a control word. The control word is fetched from core storage by the exchange or disc synchronizer and contains the initial data word address and a count of the number of words to be transmitted. The input/output unit will signal, unless such action is suppressed, the central processing unit when the transmission has been successfully completed. If the transmission cannot be executed, due to program error, machine malfunction, or not-ready conditions, an appropriate signal is sent to the central processing unit and will cause an interrupt to occur if the computer is in an "Enabled" state.

Data transmitted between the exchange or disc synchronizer and core storage are checked using the check bits associated with each word. Single bit errors are automatically corrected and no indication is given. Errors involving two bits are detected but not corrected and an indicator is turned on which normally will cause an interrupt to occur. Most errors involving more than two bits are similarly detected.

The disc file system on the Los Alamos STRETCH computer consists of one disc synchronizer, two disc control units (IBM 354) and two disc storage units (IBM 353). Only one disc synchronizer may be connected to the computer, but any number, up to a maximum of 32, of disc storage units may be connected to the disc synchronizer via disc control units.

## DESCRIPTION OF THE DISC STORAGE UNIT

One disc storage unit (IBM 353) contains 2,097,152 ($2^{21}$) words of 72 bits. Data may be transmitted between the disc and core storage at a rate of one word every eight microseconds.

The disc array is mounted on a vertical rotating shaft. There is a recording surface on each side of the disc. A "comb" of arms is mounted vertically alongside the edges of the discs. Each arm contains two sets of read-write heads, one for the surface above and one for the surface below. The arms move in a simultaneous manner in a horizontal direction and are thereby "located" on the

* Work performed under the auspices of the U.S. Atomic Energy Commission.

desired track. One half-word consisting of 32 information bits plus seven check bits is read (or written) in parallel. Two such half words are assembled (or disassembled) into a full word by the disc synchronizer.

The disc consists of 256 tracks or circles of information. Each track is divided into eight arcs and each arc consists of 1,024 bits or two bits from each of 512 words. There are two such sets of disc surfaces called modules, but this is of little concern to the user. An arc (512 word block) is the smallest addressable record of data.

The time required to move the arms from one track to another track ranges from 51 milliseconds (within the same group of 16 tracks) to 265 milliseconds (when the arms are moved from the outermost track to the innermost track). No movement of the arms is required when reading or writing another arc on the same track.

The discs make a complete revolution in 34.4 milliseconds. Hence, after the arms have been positioned on the desired track, it will be necessary to wait an average of 17.2 milliseconds for the beginning of the desired arc to come under the heads. The data are then transmitted at the rate of one full word every eight microseconds.

If, however, one can accommodate blocks of 4,096 words, the capacity of a full track (eight arcs), the transmission can be initiated as soon as the beginning of any of the eight arcs comes under the heads. This reduces the average rotational delay from 17.2 milliseconds to 2.15 milliseconds. For such operation, 4,096 words of core storage must be available and must have a first word address of 4,096 or some multiple of 4,096.

There are five basic input/output instructions used in the operation of the disc system. These are LOCATE, READ, WRITE, RELEASE, and COPY CONTROL WORD. Each of the first four of these instructions has a variant form which can be used when no signal of end of operation is desired.

Unless the transmission of data is to begin with the next arc following the last arc referenced, it is necessary to execute a LOCATE instruction prior to the READ or WRITE instruction. The LOCATE instruction specifies the disc storage unit to be used and the arc address. The arc addresses range from 0000 to 4,095 and implicitly specify the proper track. After the successful completion of the LOCATE instruction, the READ or WRITE instruction may be given.

The READ or WRITE instruction initiates the data transmission. These instructions contain the channel address of the desired disc unit and the location of the control word in core storage. The control word contains the first data word location and the count of the number of words to be transmitted. After the control word has been decoded in the disc synchronizer, the actual transmission commences. As each word is transmitted, the data word address is increased by one and the count is decreased by one. When the count reaches zero, the transmission is terminated.

The RELEASE instruction immediately terminates any data transmission activity in progress by the disc unit specified. The COPY CONTROL WORD instruction sends the data word address field and the count (as modified) from the disc synchronizer to the specified location in core storage. This instruction will not be executed while data transmission is occurring.

## SYSTEMS APPLICATIONS

With few exceptions, all problems on STRETCH are operated under a monitor program called the Master Control Program (MCP). In the near future, there will be four programming languages available, all of which compile under control of MCP. The STRETCH Assembly Program (STRAP II) is used for longhand symbolic codes. COLASL and IVY are automatic coding languages which were developed locally. A STRETCH version of FORTRAN IV is expected to become operational in the near future. All of these systems, plus our subroutine library, are stored on the disc. Approximately 260,000 words of disc storage are currently required for all of these programs.

Because of the possible high speed, random access (blocks of 512 words) operational characteristics of the disc, the compilers are able to operate faster than would be possible using STRETCH'S 729 IV tapes. The data transmission rate of the disc is in excess of 20 times the maximum transmission rate of these tapes (ignoring all access and startup times). Because of the large capacity of the disc system, the compilers can have access to a large number of diagnostic-type comments, quickly available. Random "ping-ponging" of sections of the compiler and of the code being compiled have become more feasible than was before possible using tapes. This means that compilers can devote more of the core storage to tables of instantly needed information, such as symbol tables, logic flow information, etc.

The systems programs which use disc storage usually have at least two "flip-flop" buffers or working areas for each type of activity. Such a buffer will usually consist of 512 words. When several such sets are used, the disc may consist of a random mixture of arcs with differing types of information. Here, a common practice is to keep a "label" in each arc which specifies the arc addresses of the previous and following arcs which contain information of the same type.

Normally, all input/output is done under control of MCP. This is done in a "flip-flop" buffered fashion, but not using the disc. Here, it appears, there is little to be gained by using the disc for accumulation purposes since the time required for editing and conversion is roughly equal to the time required to read or write the information using tape. If this associated computing time could be reduced or eliminated by some means, then the disc might efficiently be used as an accumulation area to "average out" peak loads.

The disc system has no built-in protection facilities to prevent accidental writing over the library. Since, however, most disc operation is performed via pseudo operations through MCP, such protection has not been needed. The extra time required by these pseudo operations to initiate disc actions appears to be negligible.

The systems furnished by IBM (e.g., STRAP II and FORTRAN) were built on the assumption that a minimum STRETCH computer consisted of one disc unit and a 48K core storage. Some thought has been given to modifying these codes to enable them to take greater advantage of our larger configuration. Hopefully, through the use of more and/or larger buffers and the elimination of disc conflicts, more speed could be attained.

Some thought has also been given to the storage of frequently used problem codes on the disc. There does not appear to be sufficient benefits attainable at the present time with the present problem "mix". Also, the use of the disc for such activity is not attractive from a "computer administrative" standpoint due to the addition and deletion of problems, nor from the fact that less disc storage would be available for problem usage.

## PROBLEM APPLICATIONS

There are primarily two functions for which the disc system has been put to use by the problem programs. These functions are dumping and the "ping-ponging" of code and/or data.

Problem dumps are often taken for two reasons: recovery in the event of computer malfunction; and backup when special mathematical or physical phenomena develop. Often a backlog of five or more dumps are kept available. Disc dumps on STRETCH are usually taken every few minutes. As a protection against a major machine failure, tape dumps may be taken every 30 to 60 minutes by the longer running problems.

The ability to make rapid disc dumps assists in "backing-up" a problem when it becomes necessary to change the method of calculation for a portion of the problem after the need for such a change has become apparent. In certain types of problems, e.g., problems which are time-dependent, the cycle increment would be increased or decreased and the calculation would be resumed from some previous point. In other types of problems, a change to alternate numerical methods might be made. Later in the calculation, a change to the previously used method or some still different method could be made. For example, suppose one is computing the pressure inside a rubber ball (as a function of time) as it is dropped and allowed to bounce. Using this technique, one could proceed in relatively large time steps until a market increase in pressure is detected, then backup and use smaller time increments during this period of interest, then enlarge the time step until the second bounce occurs, etc.

The frequency with which one should make dumps in order to recover from a machine malfunction is a function of the mean free error time and the amount of time required to make the dump. Unfortunately, the mean free error time is difficult to know and is certainly not a constant. For this reason, the long running codes generally have the mechanism to vary the frequency with which dumps are taken. It has been conjectured that STRETCH, even if it were extremely "sick" internally and was operating with a mean free error time of only one or two minutes, could still perform considerable useful work in a period of time only slightly (five percent?) longer than usual. A dump of all core storage can be done in approximately 0.8 second.

Several large problems during their execution "ping-pong" blocks of code, and some divide their data core storage into two or more large blocks which are emptied and refilled while the calculation proceeds in another one of the blocks.

Techniques are still being developed for satisfactory mathematical methods of dividing large meshes so as to permit convenient ping-ponging of sub-meshes.

Approximately 25 percent of the problems currently operating on STRETCH make some use of the disc during their execution. The largest and longest running problem uses the disc during approximately 10 percent of the running time. In spite of this apparent low usage, the disc provides a very valuable adjunct to the computing system for problem programs. The transmission speed of the disc system seems to be well matched to the size of the core storage and the internal speed of the central processing unit.

## CONCLUSION

The reliability of the currently installed disc system has been very high. Its high speed and large capacity have permitted the use of new data handling techniques which before were not feasible. Much remains to be learned and developed, however, in utilizing and exploiting the capabilities of the disc system.

Most of our applications involve the disc as an extension of the core storage rather than as input/output equipment, even though from a hardware and programming viewpoint it is handled in basically the same manner as the other input/output equipment. It would be desirable, we believe, if the disc did not share their indicators (and their associated interrupts) with the other input/output units. This is especially true in the case of the end-of-operation signal.

The cost per bit is almost exactly 1/100 of the cost of a bit of core storage. For our type of problems, the disc system is therefore an economical as well as a practical means of extending the effective amount of core storage.

A certain amount of care must be used in planning the organization of information on the disc in order to minimize delays due the positioning of the disc heads, which in the worst case might take one-fourth of a second. Even in the better cases, the positioning delays are often significant. The elimination of moving arms through the use of multiple head arrays on fixed arms or through other designs would be desirable.

Additional speed could be achieved through the use of additional arms so as to permit simultaneous reading and writing on one disc unit. A further extension would, of course, be to have more than one disc unit transmitting information at the same time.

Another improvement would be to have the LOCATE function included in the READ and WRITE instructions in addition to a separate LOCATE instruction. This would simplify the programming normally needed to use the disc unit since the end of operation interrupt (and servicing) due to the present LOCATE would be eliminated as well as the elimination of the issuance of a subsequent READ or WRITE instruction.

It seems certain that other applications utilizing disc files will be developed. Currently at Los Alamos some thought is being given to disc sorting of large amounts of data. Multi-programming techniques appear to require large discs or drums for optimal performance. Input/output efficiency might be considerably improved if the data could reside on a disc which can be directly accessed by another computer or by input/output units. With interrupt capabilities and real time inputs, there would appear to be some advantage to recording observations directly on the disc prior to calculation, thereby bypassing intermediate recording and the associated delays.

Present and future applications would suggest that all future large scale computing systems will require large capacity, high speed storage media such as can be provided by discs. Many smaller computers, in many cases, will also require this facility.

by B. W. Wyatt, Jr.
IBM Corporation, Western Regional Office
Los Angeles, California

# Characteristics of IBM Files

THE PURPOSE OF THIS DISCUSSION is to reacquaint you with the characteristics of the 1301 and 1311 disc files, and to indicate what the programmer may expect from these storage devices.

Let's begin by reviewing briefly the characteristics of each of these disc files. The magnetic disc used on the 1301 is a thin metal disc coated on both sides with a magnetic recording material. Data is recorded on magnetized spots located in concentric tracks on each face of the disc. There are 250 tracks for the storage of data on each disc face; 25 discs are mounted on a vertical shaft and this shaft spins the discs at approximately 1,790 revolutions per minute. The tracks are accessible for reading or writing by positioning read write heads between the spinning discs. The stack of 25 magnetic discs with its associated access mechanism makes up a disc storage module. Of these 25 discs, the upper 20 discs (40 surfaces) are used to store data. The remaining five discs (10 surfaces) are used for machine controlling purposes and as alternate surfaces. In each module of disc storage, corresponding disc tracks of each disc surface are physically located one above the other; for example, the outermost tracks on each disc surface are located one above the other forming a cylinder of 40 data tracks. These data tracks of the cylinders are numbered sequentially from bottom to top and from the outermost cylinder to the innermost cylinder.

The read write heads are mounted on the 24 arms of the access mechanism which moves the arms horizontally between the discs. No vertical motion is involved. Two read write heads are mounted on each arm. One of the heads services the bottom face of the disc above the arm, and the other head services the upper face of the disc below the arm. The read write heads on the 24 access arms are aligned one above the other and are are mechanically moved in parallel to one of the 250 cylinder positions of the module. When the access mechanism is positioned at a specific cylinder, 40 data tracks of information are available without any further motion of the access mechanism. Only electronic head switching is necessary to select a particular track in the cylinder. With the cylinder mode optional feature, it is possible to read or write a cylinder or part of a cylinder of tracks in one operation.

The access mechanism requires time to move from one cylinder to another. The time required is related to how far the arm moves within the certain machine defined limits. To calculate how much time will be required, consider the 250 cylinders of a module organized into five areas of 50 cylinders per area. Also consider each area of cylinders further divided into six sections. Access motion time for any one access is as follows: to move the access arm within a section of any one area requires 50 milliseconds. To move the access arm from one section to another section of an area requires 120 milliseconds, and finally, to move the access arm from one area to another area, crossing an area boundary, requires 180 milliseconds.

Any disc storage read or write operation includes another timing factor called rotational delay time. An index point for each circular disc track denotes the beginning and end of a track.

After a cylinder of tracks has been accessed and the proper read write head for a specific track of the cylinder is selected, actual reading or writing must wait until a specific addressed data or data area of the track is located. Rotational delay time is the time required for the disc to position the desired record at the selected read write head after an instruction has been initiated. Maximum rotational delay time is 34 milliseconds. Average rotational delay is 17 milliseconds. Data access time, therefore, includes the summation of access motion time and rotational delay time.

## THE DATA TRACK

The basic fixed recording area of the 1301 is the data track. The entire recording area of a data track cannot be used exclusively for the storage of data. Certain information must be recorded in the track prior to its use in the storage area. On subsequent read or write operations, this information is used by the system to identify the track and each of the record areas reserved for the storage of data on that track. A data track and the data to be written on a track or read from a track are identified by the means of a home address one, a home address two, and as many record addresses as there are record areas to be established on the data track. Home address one is the first information in each data track and follows the index point for that track. It is a four digit number and is the actual physical address of the track in a module. The track number is prerecorded in each data track and cannot be written by the user. Home address two, which follows home address one, in each data track is called the home address identifier. It consists of two or more characters which may be numeric, alphabetic or special characters. HA 2 must be written on the data track by the user prior to performing actual writing or reading operations for that track.

From an addressing or referenced viewpoint, home address one and home address two, together become the actual address of a data track in a module. HA 2 simply provides a method by which the user is able to further define the effective address of each data track. The data track storage area following the HA 1 and HA 2 addresses is one long continuous storage space. How this space is to be organized, how many records are to be stored, how many characters are in each record and how each record area is to be identified are determined by the user.

Each record area established for a data track is preceded by a record address. This address consists of six or more characters which may be numeric, alphabetic or special characters. They are assigned and written by the user to fit any convenient addressing scheme. A record address need not have any relationship to the home address of the track where it is written. The records on a data track can be of any length from a minimum of two characters to the full length of the data track less necessary character spaces for a home address, record address and gaps.

The information is recorded on a disc track serially by character and serially by bit. A space bit separates characters within a record. Information to be written on the disc is transferred character by character from core storage to the 7631 file control unit and odd bit parity check is performed on each character. Space bits are inserted and the character is written on the disc. Information is read from discs character by character. The space bits are removed and odd parity check performed and the characters sent to core storage. Each data track of the module has a capacity of 2,800 six-bit or 2,165 eight-bit character positions for the recording of information. These figures have been adjusted to compensate for the character positions used in the prerecorded home address one and the gaps for the home address.

## VARIOUS FORMATS FOR STORAGE

The advanced characteristics of the 1301 permit the user considerable flexibility in establishing how the disc storage space is to be allocated, organized and addressed. This concept of disc storage use makes possible a wide variety of storage formats to meet the needs of many varied applications.

Before any data can be written on or read from a data track within a cylinder, a format track for that cylinder must be written. The 250 format tracks, one for each cylinder, are located on one of the additional disc surfaces not used for data. The function of the format track is to control the use of the data tracks of a cylinder. Once a format track has been written, it establishes the location, character size and the mode of reading or writing which can take place in the home address area, the record address areas, the record areas, and certain gap areas. The layout and writing of the format track are under the complete control of the user. There is also a key operated lock which must be manipulated to write the format track. Once written, however, the format for a cylinder of tracks remains fixed until a format track is rewritten. As a final consideration of the 1301 disc storage unit itself, I should point out that various models of the file control unit permit these units to be attached between 1410 and/or 7010 systems, other 1410 and/or 7010 systems, 1410/7010 systems to other 7000 series systems such as 7094's, and finally 7000 series systems to 7000 series systems.

Now let us turn our attention to the newest member of the IBM disc file family. This is the IBM 1311 disc storage drive. The 1311 can be attached to many IBM systems, namely the IBM 1440 system, and the IBM 1401/1410/1620 and 7010 systems. As a base for comparison, let us examine how the IBM 1311 disc storage drive is attached to the IBM 1401 data processing system. As many as five IBM 1311 drives can be attached to a 1401 system—each drive capable of storing from 2 to 2.9 million alphameric characters.

The information is stored on small removable disc packs. Each disc pack has six discs. The surfaces of five of these discs are used for data storage. The access assembly contains a read write head for each disc surface, thus with proper file organization a minimum of access time is required for the access of a disc record. This concept of removable disc packs means that only those disc records needed for a particular application need be in use. Data records for other applications can be removed from the system and stored. The IBM 1311 disc storage drive and the portable disc pack allow the user to select a number of disc drives needed on the system at any one time and to place only the required disc records on the drive. Any number of disc packs are available for data storage. The number of packs needed by a user depends on his disc storage requirements. When a disc pack is not on a disc storage drive, it is inside its protective cover. The disc pack and its cover combine to make a rugged, sealed container.

Each drive has its own independent access mechanism consisting of 10 read write heads. All drives can be used by a single program in much the same way a tape system uses tape units for storing master and active input and output files. The 1311 supplies only the access and drive mechanisms. The disc records are separate. Independent portable disc packs are used with the 1311 drive. Nominal change time is two minutes. The packs are interchangeable with other 1311 drives. This means that the philosophy of tape libraries can now be applied to magnetic disc storage. A library of disc packs can be set up and when needed the packs necessary for an operation can be placed on the disc storage drives for system use.

## SURFACE DIVISIONS

The 10 disc surfaces are divided into tracks and sectors. Each disc surface contains 100 tracks which are subdivided into 20 sectors per track. Thus there are 20,000 addressable sectors per disc pack. Here again, as in the 1301, the cylinder concept is used because there is a separate read write mechanism for each disc surface. In a fashion similar to the 1301, the access assembly on each 1311 disc has five arms. Each arm has two read write heads attached to it. The read write heads move horizontally across the tracks on the record during a seek operation. Vertical movement of the access mechanism is not necessary since there is an arm for each disc. Because each disc surface has its own head, it is not necessary to perform a seek operation if the sought record is in the same vertical plane cylinder as the record previously operated upon.

The use of a comb-like access assembly greatly reduces the access time of a record in disc storage. The maximum access time is 400 milliseconds. The average access time is 250 milliseconds. Access time can be further reduced by using the direct seek feature which does not require the access assembly to return to the home position for each seek operation. The direct seek feature reduces maximum access time to 250 milliseconds, and the average access time to 150 milliseconds. In addition, of course, an additional timing factor must be considered, that is rotational delay time. The maximum time for rotational delay on the 1311 disc drive is 42 milliseconds. The minimum time is two milliseconds. Therefore, when considering the time required to make a disc record available to the system, consider both access motion time and rotational delay.

An indelible five digit sector address precedes each addressable location in a disc pack. These numeric addresses are sequential within track cylinder and disc pack. If a system uses five disc drives, the sector addresses continue in this sequence from disc drive to disc drive. A special feature of the 1311 disc storage drive is the track record feature. This provides for reading or writing an entire disc track with or without the track addresses. Track records therefore can be used for storing programs, tables, block records, and other data requiring a single large storage block.

In summary, then, a comparison of the 1301 and 1311 disc storage units might be made as follows:

The 1301 provides flexible format and a large bulk storage device on line. This file unit can be interconnected between various large scale systems. On the other hand, the 1311 disc storage unit provides unlimited storage capabilities with easily removable and portable disc packs, and in much the way that tapes can be moved from one system to another, the disc packs themselves can be transported and moved from a 1440 system to a 7010 system for additional processing.

## PRODUCTION

Most of IBM's random access file production and development work is at the San Jose, California, plant site. To mention only a few of the facilities and manufacturing techniques used:

The 1301 assembly line facility is designed and arranged specifically for production ease and cleanliness.

The manufacturing of one small component of the 1301 file – the recording head which reads and writes the impulses on the disc surface – requires great care and precision. One section of the plant is devoted to head production.

In working on the recording heads, microscopes are used for optical monitoring of each item. Production techniques have now advanced to the stage where a prototype of a unit built to produce these heads automatically has been completed.

Head testing is obviously an equally important step in producing a file which will operate satisfactorily in our customer's installation. Technicians, in surgical garb, align the heads for proper seating in their units. Closed circuit TV is used for alignment with a preset pattern on the scope. Again, non-human eyes are required to insure proper alignment of these very small objects

Another very interesting aspect of file production is the product test. We have chosen to use the 1311 as illustrative of this area. The 1311 disc pack was announced late in 1962. Surprising might be the fact that the IBM 1710 control system had been using special 1311 files for sometime prior to announcement on the 1440, 1401, 1620, *et al.*

Actual running experience – on rental – in a customer's office is certainly some evidence of product performance. However, this is not the whole story. The 1311 has been subjected to some of the longest and most exhausting tests in our Product Testing laboratories we can remember. Product Test has had this machine in its laboratory for a cumulative time sufficient to simulate almost six years of customer operations on two files. They are still going strong.

Our engineers in the Laboratory are also dedicated to provide a product requiring minimum maintenance; hence we are conducting our own reliability improvement program. The product already meets the high level specifications we required; however, we have set up a completely separate group of engineers to continue careful scrutiny of every part and point in the 1311. They are examining every gear for wearability, and they are even dumping various types of dirt and dust on the running discs to see what can be done to make the file work better under "less than desirable" conditions.

## FUTURE

No discussion of this nature would be complete without some reference to the future and what ideas and techniques might be expected. We are exploring many avenues of development and research.

One of these is a glass disc. In studying optically the characteristics of heads as they fly above the disc surface we note the disc rotates at approximately 1,700 rpm. The head flies at only 125 micro-inches above the surface – that is 125 millionths of an inch. Perhaps a more easily grasped comparison would be appropriate – the particles of cigarette smoke a smoker exhales are roughly three microns in size. A single micron is equivalent to 40 micro-inches; therefore you would be hard pressed to blow cigarette smoke particles between the flying head and the disc surface. They would just about fit! Of course, under normal operating conditions, the air currents would keep these particles from entering the lead area.

Just as we are exploring new head designs, we are also exploring surface coating techniques. As files have advanced, the thickness of coating has decreased from 1.2 mils or thousands of an inch on the 1405 file for the 1401/1410 to 0.4 mils on the 1301 and 0.2 mils on the 1311. In the experimental area "clean room" the engineer, dressed in a surgical outfit, is outfitted with an alarm which will signal presence of any particles larger than our familiar three micron cigarette smoke.

This type of research has produced the successor files to the 350 file on the original RAMAC through the 1405 on to the 1301 and 1311 and to date reaching to the 1301's cousin, the STRETCH file. The 1301's 90 kc character rate has been stepped up dramatically in the STRETCH file because this file has been designed for parallel access to binary words. If this rate were related to 1301 speed, the STRETCH file would yield a rate of 1,400 kc. Expressed another way, a novel the size of *Gone with the Wind* could be read on the 1301 in about 30 seconds. If the STRETCH file were used for this trivial task, the reading could be completed in some two seconds.

by B. W. Wyatt, Jr.
IBM Corporation, Western Regional Office
Los Angeles, California

# IBM File Concepts

Now THAT WE HAVE REVIEWED the characteristics of the IBM 1301 and 1311 disc units, let us explore some considerations and techniques of addressing these files and file organization, and also the IBM file oriented programming systems for the 1301/-1311 disc storage units. As we have seen, the two files are somewhat different in physical characteristics, however many common considerations in file organization and addressing exist.

Let us first consider file addressing. The data records which must be stored in our random access device are usually identified by a control field which can be numeric, alphabetic, special character or any combination of these. In addition, the size of the control fields could vary considerably, and typically, the distribution of the control field numbers is seldom even. For example, in any numbering system seldom are all of the available numbers in a control field used; and some coding system might be used to effect the distribution. There might be logical groupings by departments, class, sales territory, and gaps may be left to allow for expansion. All of these factors may cause a heavy concentration of numbers in some ranges and light concentration in others. Generally, random access devices in which we are going to place these data records are composed of physical record locations identified by an evenly distributed set of numbers. So, therefore, the problem is to convert an unevenly distributed set of numbers into an evenly distributed consecutive set of numbers within the limits imposed by the random device.

## ADDRESSING TECHNIQUES

There are two basic techniques for addressing — direct and indirect addressing. In the use of direct addressing, the input transaction itself would contain the actual address of the data records to be updated. If the application is so designed, it is certainly desirable from a machine and programmer's standpoint to use direct addressing. A direct addressing technique may create an additional burden on the people who must communicate with the computer system. Therefore, within the limits of practicability it is certainly desirable to design the system to fit the job rather than forcing the job to conform to the demands of the system.

We are led, therefore, to the indirect addressing technique. There are two approaches to indirect addressing. One is address modification; the other is address conversion. Address modification could be used, for example, to add a certain constant to a particular number. Take the case of a thousand-man payroll with sequential employee numbers from 1-1000. We could add a certain constant or multiply the employee number by a constant and hence arrive at a location. Any use of a table lookup technique within the machine could be considered a form of address modification. This technique is usually most practical on small files or table sizes such that they can be contained in core. The address modification technique will result in the computer processing of a control number in a predictable fashion. In other words, the results can be predicted without actually testing the technique against all control numbers in the file.

In the address conversion situation the results of the computer processing on the control number are unpredictable. It is impossible to determine how many control numbers will generate the same machine address or how many machine addresses will be unused for a particular technique or formula until such time as the formula has been applied to all control numbers in the file. It is this situation which dictates the necessity of a formula evaluation. One such technique is the digit analysis table to show the frequency of distribution of the digits 0 through 9 as they appear in each control field.

Numeric techniques such as extracting, folding, multiplication, division and others have been used successfully in different installations. Any of these techniques may produce addresses which are outside the range of the file space allotted. A compression factor may also be used to restrict the range of numbers generated. Also some determination of file pack is required. Generally one should attempt to pack as much as possible to economize on space yet not adversely affect the entire operation. And, of course, the duplicate and synonym problem arises. In the case of the 1301, however, due to the variable length feature and the record address concept it is often quite desirable to have duplicates. Ideally the number of duplicates should be the same as the number of records per track. Regardless, it is necessary to determine some technique to handle the duplicates. One approach is indexing to locate the duplicates. Another possible solution might be chaining. In the latter technique one record leads you to the next one, link by link.

In all of these addressing techniques some method of evaluation is necessary. Let us list some of the factors to be considered. First, the number of items in the master file. Second, the number of file addresses with one item assigned, two items assigned, three items assigned, etc. The average number of seeks to locate an item in the master file, the activity of items in the master file, and finally, the number of data records that can be stored per file location.

As a final comment on file addressing, it seems altogether reasonable that a periodic re-evaluation of the file addressing technique would be in order. It is certainly possible that the first technique used for a given file might be inadequate due to changing business conditions, characteristics of the file, etc.

## ORGANIZATION OF FILES

Now let us explore file organization. Just what is file organization. One definition would be that it is the systematic storing of information in such a manner that it may be easily retrieved for processing or reporting. There could be several systems objectives: 1) *process* the information within the capacity allowed; 2) *complete* a job within certain limits; and 3) *do* the job with minimum costs. Certainly the equipment itself affects these objectives. But let us remember that the job defnition itself would be quite important without regard to the equipment. One must consider the volumes, the frequencies, the peaks, and some quantitative information is needed to balance the time, cost and capacity.

What are some of the techniques of file organization? In the first place a very gross approach to this problem will come from the job definition — just how much storage is required. We might add a certain fudge factor, perhaps 20 percent extra space. Perhaps just on the gross requirements we can determine how many modules will be necessary. And maybe then we could make a further decision based upon how near our gross estimates are to a boundary value. For example, should we have one more or one less module. Secondly, we might consider any special attributes of these various data files that make up the job as a whole, which might require a particular addressing technique. We certainly must consider the effects of the growth on the size of each file.

Assuming that we have made a decision on the space to be set aside for a file, what should the records look like? Probably a single record per track will yield the maximum possible storage. We can, of course, block many logical records into a single physical track record just as we do on tape; but perhaps here we have a question of process time. How long will the channel be busy? How much time is required to scan and pick out the record we want to update? The question is really one of balancing process time available and required against the capacity. Is it possible to assign a portion of our disc storage for each file required?

In the case of the 1301, for example, we should attempt to organize into cylinders. If we can maintain a file within a given cylinder or a section of cylinders the access time is reduced considerably. There are reasons for splitting a file over several modules. One such reason might be to share the access mechanism for frequently indexed files; another would be to overlap many access mechanism movement times. Other considerations would be the ability to have at least part of the file available even if one module were off line. Other methods for saving access time — these have come to be known as revolver techniques and roundelay techniques — repeat the data in the file several times to minimize access time to any given record.

Perhaps there might be some natural sequencing of input or output that might determine the organization and addressing of files. We might, in a Tele-processing environment, want to organize the file so that the file would be indexed in the same sequence as the polling order of the communication system. On the other hand, if the reports from a given file are still an overriding requirement we would want to store the file in the order of the report. Another consideration in file organization might be the desirability of simultaneously updating multiple files from one input transaction. One example of this situation might be in a combined billing inventory and accounts receivable application.

The intent of this discussion is to provide background on some of the techniques and considerations involved in file addressing and file organization.

## FILE ORIENTED PROGRAMMING SYSTEMS

### For the 1311/1440 user

First, let us discuss briefly those programs and programming systems to be supplied to a 1440 user of a 1311 disc storage device.

The IBM 1440 Autocoder system enables the programmer to use symbolic names instead of machine addresses when referring to locations of data instructions in core storage. The programs written in Autocoder language are punched into cards, loaded into the 1440 and assembled by the 1440 Autocoder processor. This assembly program produces a self-loading machine language object program on disc or cards ready for execution.

A Report Program Generator system will be provided. Instead of writing a specific program for each report to be prepared, the user need only provide certain specifications for the generator program. These specifications include a description of the data from which the report is to be made, the calculations to be performed, and a description of the format of the desired report. After assembly and when supplied with data and executed, the machine language program produces the report in any combination of three forms — printed, punched card, or written in disc storage.

An Input/Output Control System will be developed to eliminate the need for detailed programming considerations to control input/output operations. The IOCS supplements 1440 Autocoder and consists of standardized routines that perform all card, printer and disc input/output operations. To use IOCS when writing Autocoder programs, the programmer supplies the requirements of his specific job and then writes an IOCS macro instruction at the point the input/output operation is to occur. When the IOCS macro instruction is recognized during assembly, the Autocoder processor inserts the appropriate machine language instructions in the object program. Standardized routines for input/output error correction and label handling are also provided.

Six disc utility programs are provided to assist the user in the operation of his installation. By means of these programs certain frequently required operations such as loading or unloading disc files from cards, printing out areas of disc storage for program testing purposes, etc., can be performed without programming effort on the part of the user.

A sort program for the 1440/1311 system will be provided. Sort V is a set of generalized sorting routines which are incorporated in the Autocoder system library. The programmer supplies the Autocoder processor with a description of the type of sort he requires causing the generation of a sort program conforming to his specifications. This generated object program is loaded into core storage along with control cards which describe the data file to be sorted. The object program reads the control card information and modifies itself to perform the particular sorting operation. Data records that are either punched in cards or contained in disc storage can be sorted and then written in disc storage in ascending or descending sequence.

The last set of programs to be provided for the 1311 system on the 1440 is a disc storage file organization program. A set of these programs is provided to assist users in establishing and maintaining data files and disc storage. There are two sets of programs, one for random files and one for control sequential files. These file organization programs are generated by the macro generator portion of the Autocoder system. As is evident from this brief discussion, it is important to provide the random file user with all of the programming tools necessary to minimize the time and effort required to install his file oriented system.

### For the 7094/1301 user

Let us now turn our attention to the file oriented programming systems available for the 1301/7094 systems.

The 1301 disc file has two general uses on the 7094: the system's usage, in which the systems may reside upon the disc file, as well as the systems using the disc file for scratch tape; and the IOCS usage which allows the application programmers to utilize the facilities of the disc. We'll discuss the system's usage of the disc first.

The IBJOB system has the capability of using the 1301 disc file for systems residence, and source program residence, and as system scratch as a replacement for tape. Within the IBJOB system, operation is configuration independent. This means that with no change to the programmer's programs, usage of the disc in place of tape or usage of Hypertape in place of either the 1301 disc file or 729 tapes is possible. This is done by having certain tables set up within a basic monitor. These tables are of three types. One is a *function name* table. This is a table of the functions which are required by the system: for instance, an input function, an output function, a library function, some utility tape functions.

There is another set of tables within the basic monitor called *unit control blocks*. There is a unit control block for each physical I/O device. The function of the unit control block is to assign a particular device or a portion of a hardware device to one of the required system functions. By making the proper assignment of unit control block to the function, the system will use that device to perform the particular function. This is accomplished by putting the location of the unit control block, for the device satisfying the required function, in the function table.

The third type of table is the *file control block* which contains information about the characteristics of the data in the file. One of its uses is to *point* to the UCB of the I/O device upon which the data file resides. Since the UCB contains the physical address of the I/O device, each file block or function table entry points to a UCB which in turn points to a particular I/O device. By pointing to a UCB for disc rather than tape the appropriate data file of function can utilize disc instead of tape.

All of this technique, however, requires the use of a standard IOCS. The IBJOB processor uses such a standard IOCS as its input/output control system. This is the same IOCS that is used by an applications programmer. Thus the statements about IOCS usage for the IBJOB system are true for the user of the IBJOB system. One IOCS package is brought into core for use by the user in different sized packages depending upon his requirements. The largest portion, of course, is the complete IOCS with random capability. Somewhat smaller versions of IOCS can be utilized depending upon what is required on the part of the user in his IOCS system.

The smallest of these is IOEX, which is a trap supervisor and seek scheduler. The next level is IOBS, which adds buffering to the asynchronous operation of IOEX. The next larger package is the IOBL or label package. IOEX is always required and the IOBS and IOBL packages are cumulative, that is, IOBL requires IOBS. However, RANDRM can operate with IOEX only or with IOBS and IOBL.

The IOCS system has the ability to utilize the disc in three different fashions. First, using the disc through IOEX – the executor or trap supervisor of the Input/Output Control System – allows the programmer a great deal of flexibility; yet it requires a great deal of sophistication on the part of the programmer. The executor itself performs the seeks, schedules the seeks and schedules the I/O such that the input/output operation does not precede the appropriate seek. In addition, an error recovery is made in case an error should appear. This type of operation is unbuffered; however, it accomplishes seeks and I/O asynchronously with CPU processing.

The second form in which IOCS uses the disc is the so-called sequential IOCS which uses both the buffering system of IOCS and the executor of IOCS. It stimulates tape on an area of the disc. The IBJOB system uses this IOCS whenever the disc is substituted for tape to satisfy one of the required functions. It is fully buffered and asynchronous and incorporates the same error recovery procedures found when using IOEX. The full capabilities of the 729 tape IOCS are available in the 1301 sequential IOCS, including labels. In addition to the normal protection afforded by labels, you also have the protection given by the Home Address 2 checking capabilities of the 1301. This system is buffered and asynchronous and much easier to use than the IOEX.

The third form of the IOCS usage of the disc is the 1301 Random IOCS. This portion of the IOCS makes possible the exploitation of the random access capabilities of the disc and makes it easily available. The random IOCS allows the user to find any number of "found" records identified by a "finder" record. Again, this is asynchronous with CPU operation and is quite easily used. The IBJOB systems themselves, that is, the systems within the IBJOB processor, do not use this random IOCS. The systems are designed to use tape efficiently. Information is organized in such a fashion that it is processed sequentially and, therefore, the need for random access to it is minimized.

In order for the systems to use the random capabilities of the disc, it would mean that each system would have to be re-organized in the way in which it approaches the solution of its problem. In compiling a FORTRAN program you would eliminate those steps in which you sort in order to put the information in a file which could be treated sequentially. Thus we would be faced with two systems if this were accomplished: one which worked efficiently from the disc, and one which worked efficiently from tape.

Currently two systems seem to be a rather nebulous possibility. Using the 1301 disc file for sorting on the 7090 is a problem of the same nature. The techniques that are used for sorting on the 7090 have gone to great lengths to take advantage of the large core available and produce as long strings as possible. These then, of course, minimize the difficulties that one has with the sequential nature of tape in sorting. The only advantage obtained from using the disc as an alternate device in the sorting procedure is that the disc can represent a great many more tape drives than most installations have. Thus a high order of merge is possible. However, even using this high order of merge which decreases the number of merge passes, there is not enough gain to overcome the disadvantage of the large amount of information which has to be funnelled over one channel, and of course, the number of seeks which are required to go from one of these pseudo tapes to another. The only hope for making a good efficient sorting program is to redesign a sort program such that it works efficiently from the disc. I feel it would be appropriate to mention that the 1301 programming systems support for the 1410 system matches in philosophy each of the comments made concerning the 7094 except in one regard. The use of 1301 disc file units will perform sorting in a very efficient manner. As a matter of fact, the 1410 sort using two 1301 modules will be superior in performance to tape sorting because of the much higher order of merge possible. Perhaps here we have the first indication of a breakthrough in the use of files for an application area earlier reserved for tapes.

The purpose of this paper, and the one preceding it, has been to provide a base for the understanding of the characteristics of the IBM 1301 and 1311 disc storage units, a discussion on file organization and addressing techniques, and a brief discussion on the programming systems to be supplied for typical file oriented configurations.

by W. L. Slover*
Bryant Computer Products
Walled Lake, Michigan

# Characteristics of the Bryant 4000 Series Disc File

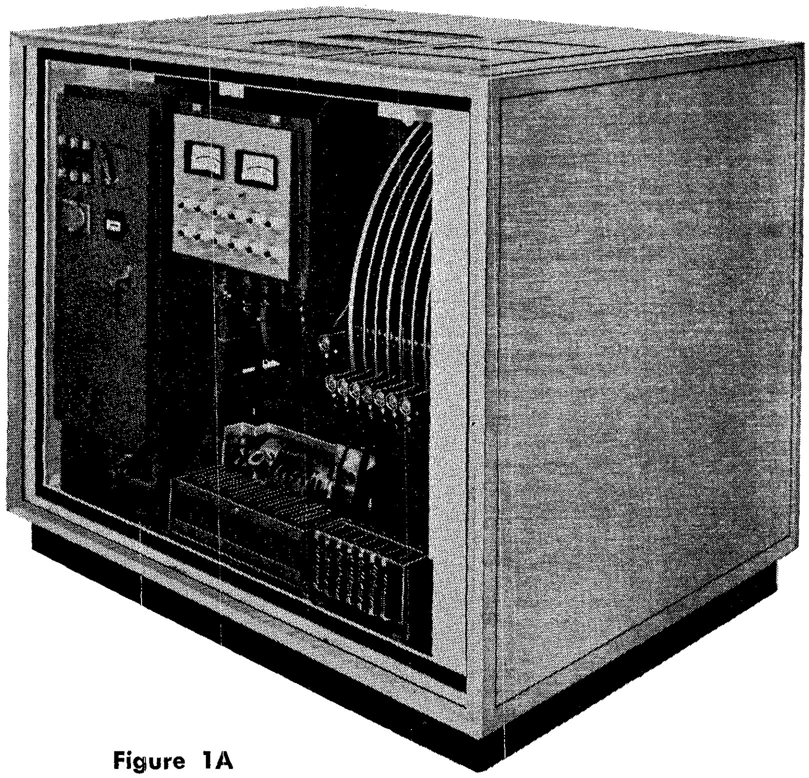


Figure 1A

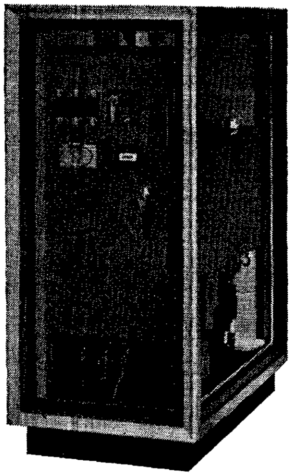
MY ASSIGNMENT IS TO DESCRIBE the Bryant 4000 Series File (**Figure** 1) and to explain why it was designed as it is. A necessary prerequisite to a thorough understanding of the "what" and the "why" of this file, however, is an appreciation of Bryant, its background and capabilities. Bryant Computer Products is an operating division of Ex-Cell-O Corporation, a publicly owned company listed on the New York Stock Exchange. In fiscal 1962, Ex-Cell-O's gross sales of over $160,000,000 produced $12,000,000 in profits after taxes.

Ex-Cell-O's product line is widely diversified. Milk packaging machines, which have captured 80 percent of the worldwide market, produce, fill and seal the familiar chisel-top milk cartons. A wide range of machine tools is also manufactured. A numerically controlled contour machine is designed to produce thin walled hemispheres to tolerances of 0.00080 inch. A massive transfer machine drills engine blocks. Jet engine fuel nozzles, hydraulic actuators, jet engine blades, and precision spindles rotating up to 120,000 rpm are also principal products.

All of these products, as well as Bryant disc files and magnetic drums, have a common denominator – precision metalworking know-how and facilities. Ex-Cell-O's 26 plants and four engineering centers in this country, and five plants in Canada, England, Germany and India incorporate more than three million square feet of precision metalworking facilities and employ more than 10,000 mechanical engineers, design tool makers, machinists, assemblers and supporting personnel. It is this vast precision mechanical capability

* Now with Data-Stor Division of Cook Electric Company.

**Figure 1. Overall views of Bryant Series 4000 Disc Files. The six data disc "A" model is a self-contained unit; the 13 data disc "B" and 25 data disc "C" models have separate cabinets for the power control unit and the disc file assembly.**



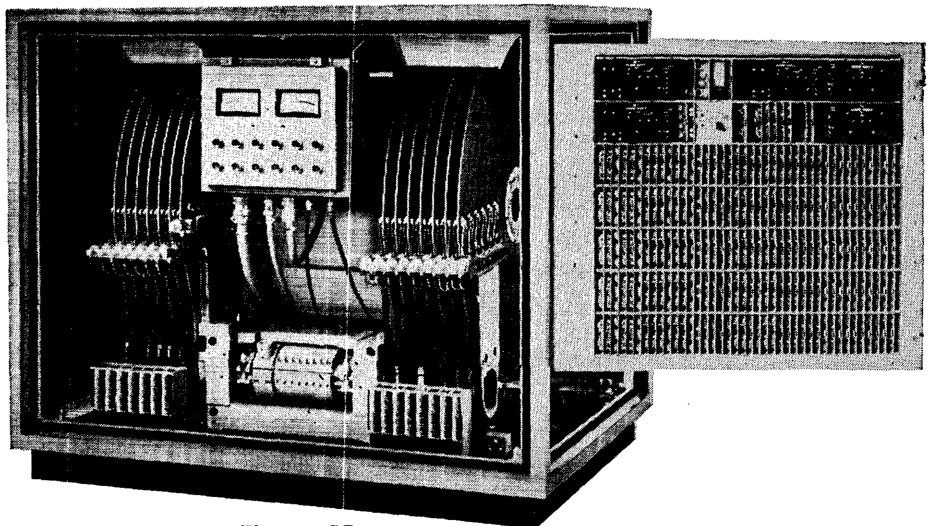


**Figure 1B**

which distinguishes Ex-Cell-O from the basically electronically oriented computer manufacturers, and, indeed, which complements their electronic competence.

Bryant Computer Products entered the "electronic" data processing field in 1954 as the result of an advertising campaign aimed at finding new applications for Bryant precision spindles. Eckert-Mauchly Division of Remington Rand ordered spindles of a special design for use as magnetic drums. Soon Bryant was producing magnetic drums for almost every company in the industry.

Shortly after entering the magnetic drum business, Bryant began designing and assembling precision magnetic heads, and later organized an electronics department to produce read/write electronics as a service to our customers. Now we have delivered many hundreds of drums and over 100,000 magnetic heads, as well as a number of successful drum systems. We are not, however,

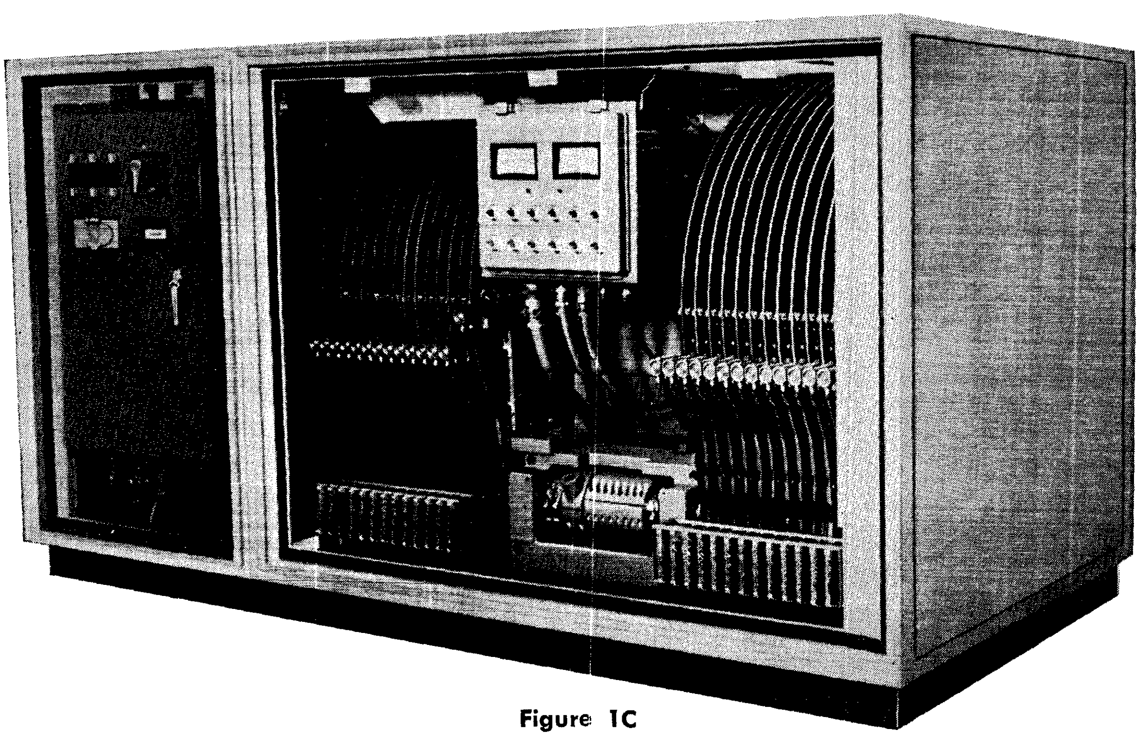


**Figure 1C**

and have no intention of becoming competitive with our customers in the electronics field. Electronics are offered as a service only.

In 1957, market surveys aimed at broadening Bryant's penetration of the data processing field indicated the need for mass random access memories and Bryant initiated development of its first disc file. A prototype model 320 file (**Figure 2**) was delivered in November 1960 and has operated satisfactorily ever since.

During the final stages of building the prototype file, it became apparent that, whereas the 320 series file more than met the original specifications, many improvements should be made in a production file. At the same time, competitive pressure was exerted by announcement of other files.

In December, 1960, we decided to abandon the 320 design except for special military applications and began designing the 4000 series file.

At that time, we recognized that our ability to produce a superior file which would serve the industry's needs for several years lay in our precision mechanical know-how, not in the application of sophisticated electronic system and programming techniques. Therefore, a basic design doctrine was established that, electromechanically, the file was to be as perfect as all of our machine tool design and precision manufacturing experience could make it, even though we were not yet ready electronically to achieve the performance levels this mechanical perfection would allow.

Thus, the 4000 series files are technique limited rather than equipment limited.

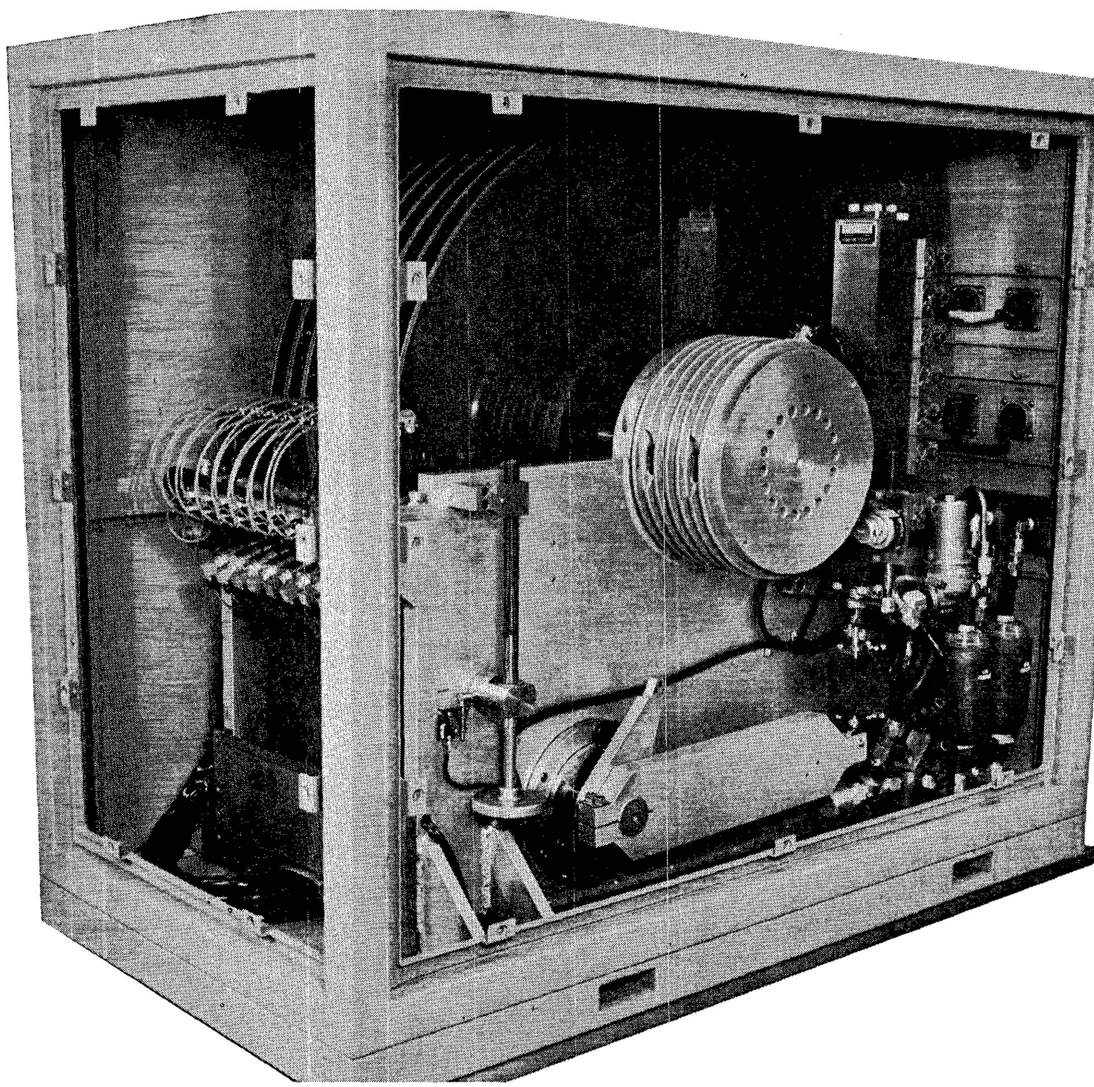
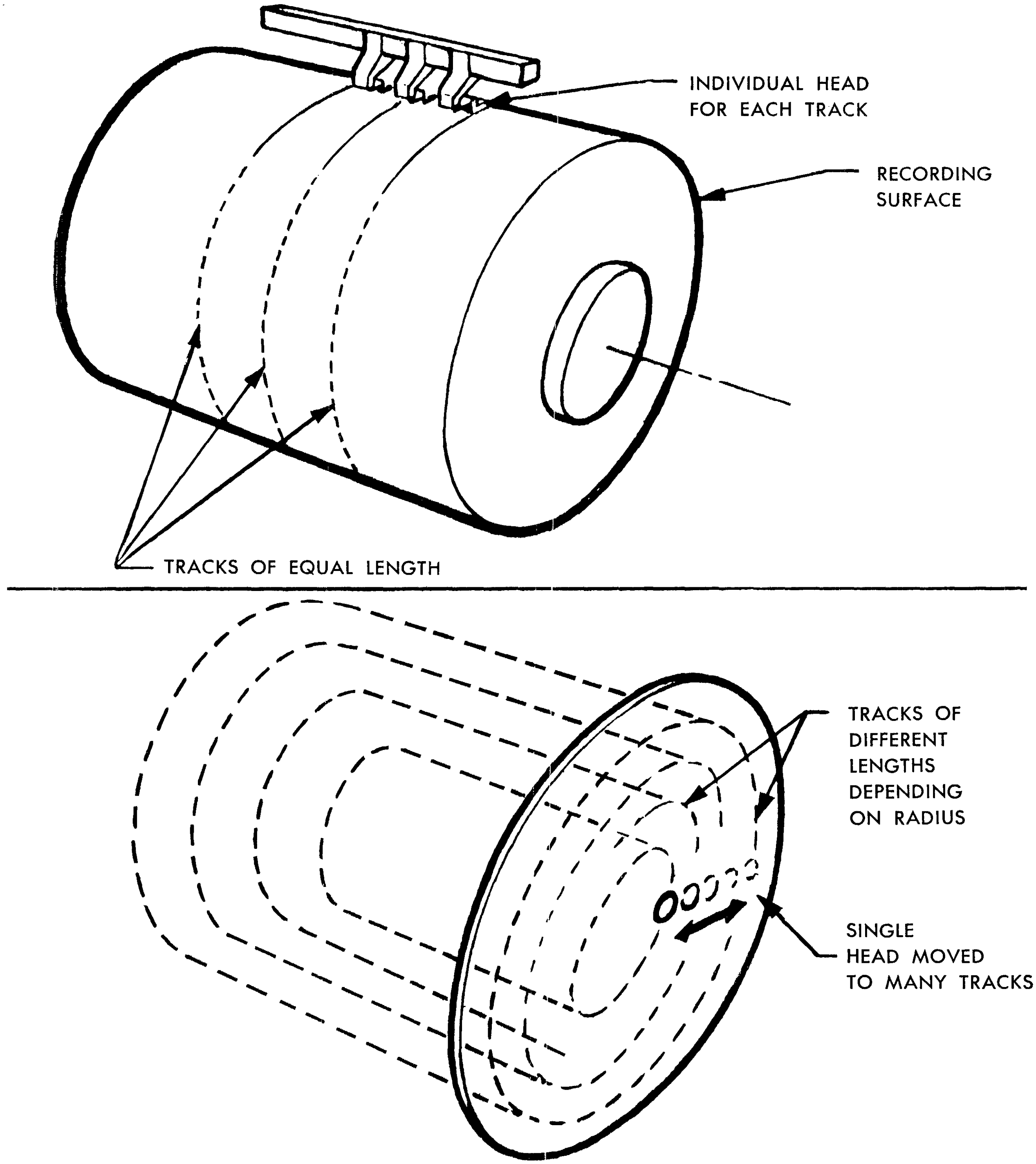


**Figure 2**

**DRUM - DISC COMPARISON**




POSITIONING HEADS ON A DISC IS LIKE MOVING HEADS FROM ONE DRUM SIZE TO ANOTHER.

**Figure 3**

## BASIC FILE PHILOSOPHY

At the same time, it was decided to continue into the 4000 series design the basic philosophy of the prototype file — that the file should be capable of simulating both magnetic drum and magnetic tape operation. In addition to self clocked operation, it should be capable of:

1. Clocked read and write
2. Serial or parallel operation
3. Bit interlacing
4. Selective alteration
5. Field expansion of capacity
6. Simulating a magnetic drum in each position of its heads.

The basic difference between a disc file and a magnetic drum (**Figure 3**) is that all of the tracks on a magnetic drum are the same length whereas the concentric tracks on a disc are each of a unique length. Usually, the drum has a magnetic head for each track. In the disc file, each head serves many tracks.

The differential lengths of the concentric tracks on a disc present the designer with these questions:

Should he use a single bit length for all tracks with only the innermost track achieving the maximum allowable bit density? This choice is very inefficient in terms of capacity.

Should he use the maximum bit density in every track? This maximizes capacity but produces a unique bit length and frequency for each track.

Should he divide the disc into concentric zones, each incorporating a number of tracks with the same bit length and frequency? Here reasonable capacity is achieved with a reasonable number of track lengths and frequencies.

Should he use zones but employ bit interlacing (interleaving, bit by bit, two or more separately addressable channels in one physical track) to achieve a single track length and a single data frequency throughout? This reduces capacity slightly and places stringent requirements on the clocking and the coating-head-read-write-amplifier system.

## FILE DESCRIPTION

In the Bryant file each disc face incorporates 768 concentric tracks serviced by six magnetic heads — 128 tracks per head. A maximum bit density of about 300 bits per inch is used for clocked systems and 600 per inch for self clocked systems. A single zone system is shown in **Figure 4.** Disc capacity is 19,000,000 bits at 300 bits per inch. It is interesting to note that with a single zone system maximum capacity is achieved by using only four heads and 512 tracks per disc face. **Figure 5** illustrates a six-zone system which provides maximum capacity of over 31,000,000 bits per disc at 300 bits per inch and 65,000,000 bits per disc at 600 bits per inch. **Figure 6** shows a two-zone, bit-interlace system which produces a capacity of 25,800,000 bits per disc at 300 bits per inch. A single track length of 11,200 bits is used to achieve a frequency of 224 kc.

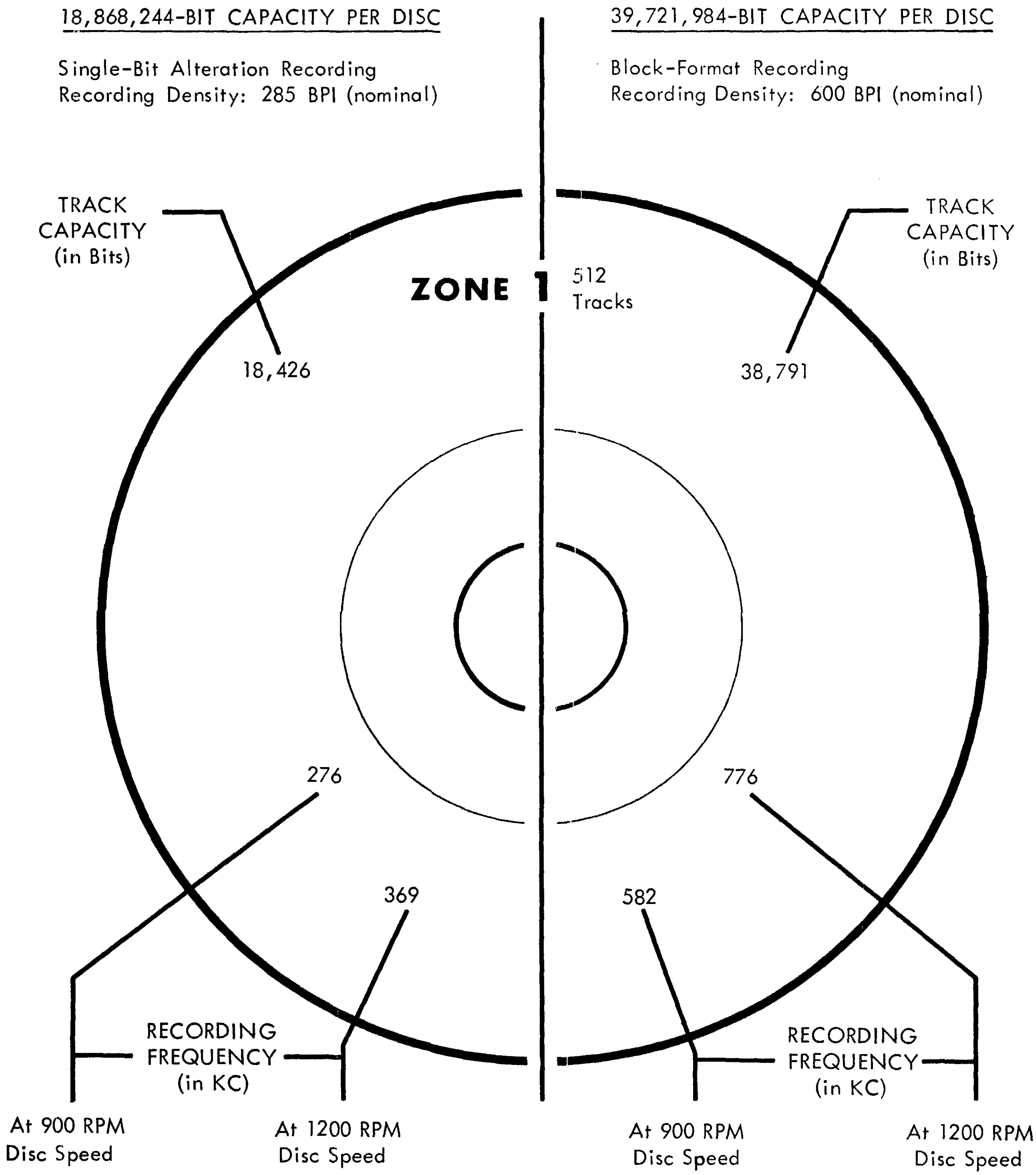
These applications are illustrative only and are not limitations. The track lengths are variable to suit individual requirements. Either 900 rpm or 1,200 rpm disc speeds may be chosen. Other bit and character interlacing arrangements may be selected.

The Bryant discs which achieve these results are 39 inches in diameter and each track is 0.010 inch wide. The intertrack guard bands are 0.005 inch wide. At 285 bits per inch, each bit recording occupies 0.0035 inch. In order to achieve reliable clocked operation, the heads must be positioned on track ± 0.0005 inch, and the data to clock phase relationship must not vary more than ± 0.0004 inch along the track.

Here's how this precision is achieved. Referring to **Figure 7**, the basic file structure comprises a massive, rigid center pedestal, surmounting a relatively flexible base which is allowed to conform to the surface on which it is located. All of the precision assemblies of the file are mounted on the pedestal and move with it relatively unaffected by flexing of the base. The disc spindle is mounted in the pedestal on grade nine angular contact bearings, the most precise available in production. The spindle is cantilevered on both sides of the pedestal and, when assembled, has a total indicated runout of 0.0002 inch at either end. The spindle is motor driven at either 900 or 1,200 rpm through adjustable length belts. Three spindle sizes are available: one is cantilevered on one side only and accommodates up to eight discs; a second accommodates a maximum of 14 discs; and a third accommodates a maximum of 26 discs. Any number of discs are available within these maxima.

The thickness, flatness and parallelism tolerances which can be achieved with large discs coated with a magnetic medium are rather loose. Therefore, to achieve precision positioning of the

*(Continued on page 37)*




Note: Maximum single-zone capacities are achieved by using only the four outermost heads serving each disc face and the total of 512 tracks they access (128 tracks per head). In short, 768n<512N where n and N are the maximum permissible track capacities in bits for the innermost and the third head out, respectively.

**Figure 4. MAXIMUM CAPACITY OBTAINED WITH ONE-ZONE DISC FORMAT**

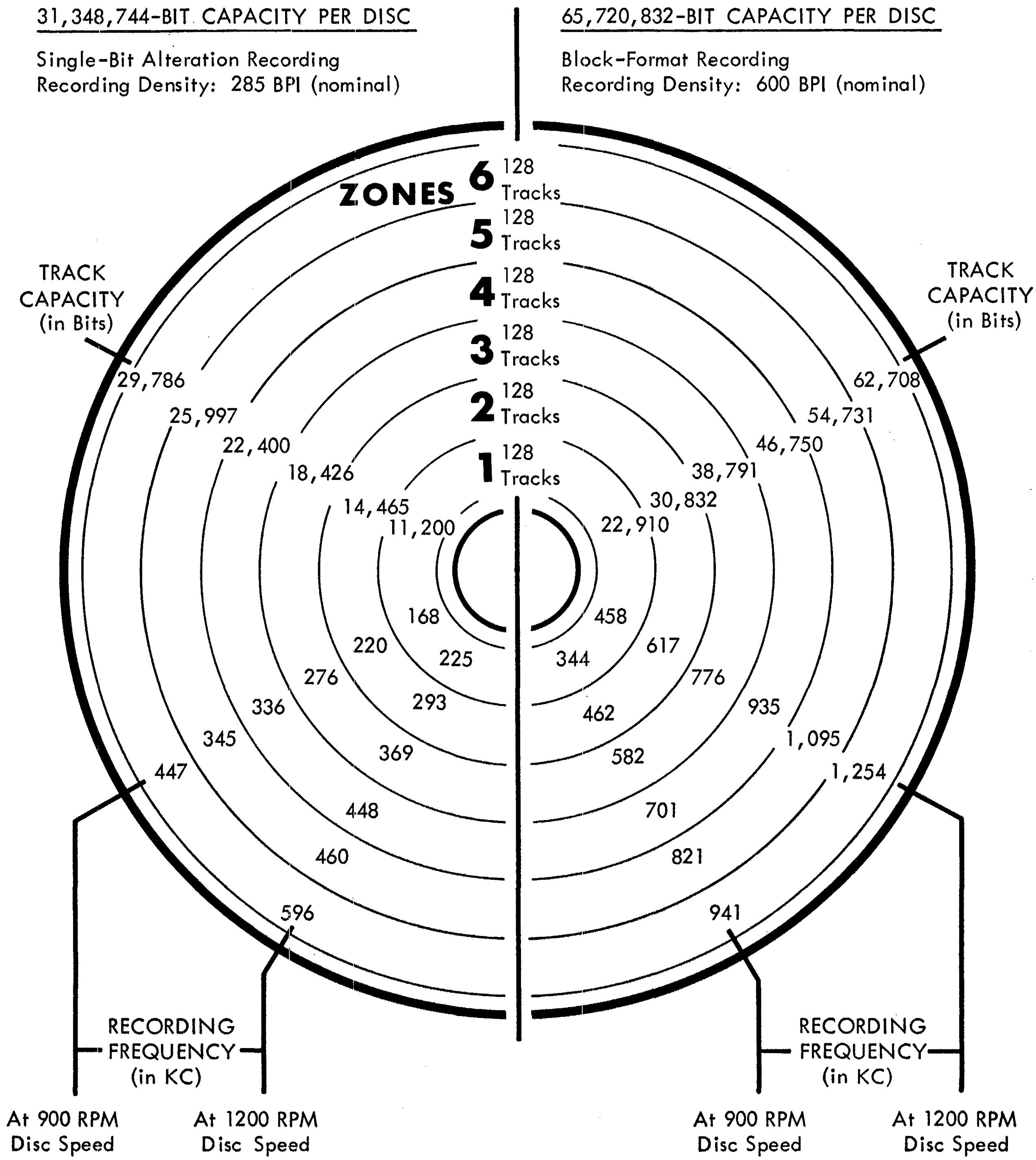



**Figure 5. MAXIMUM CAPACITIES OBTAINED WITH SIX-ZONE DISC FORMAT**

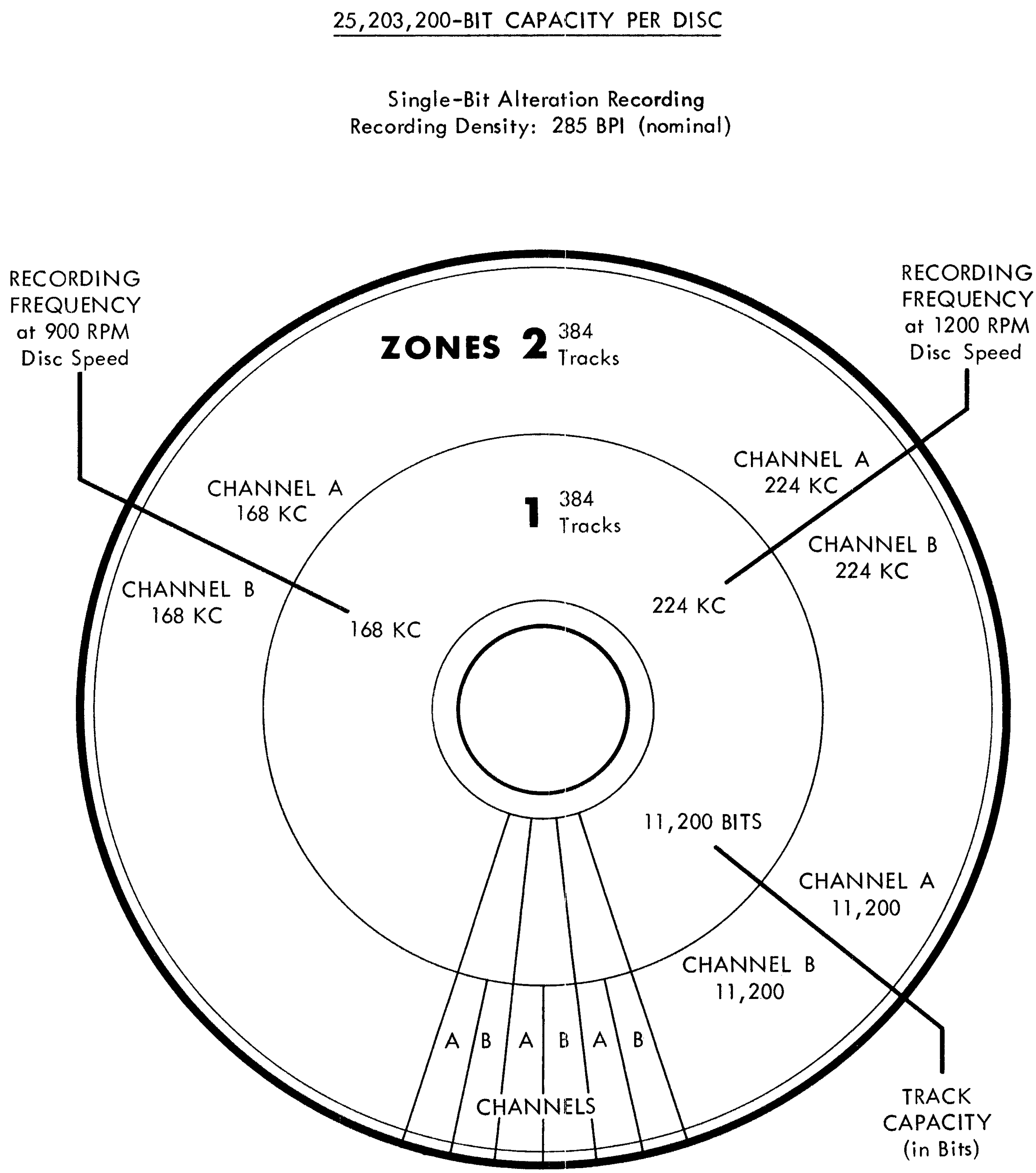



**Figure 6. MAXIMUM CAPACITIES OBTAINED WITH TWO-ZONE BIT INTERLACE FORMAT**

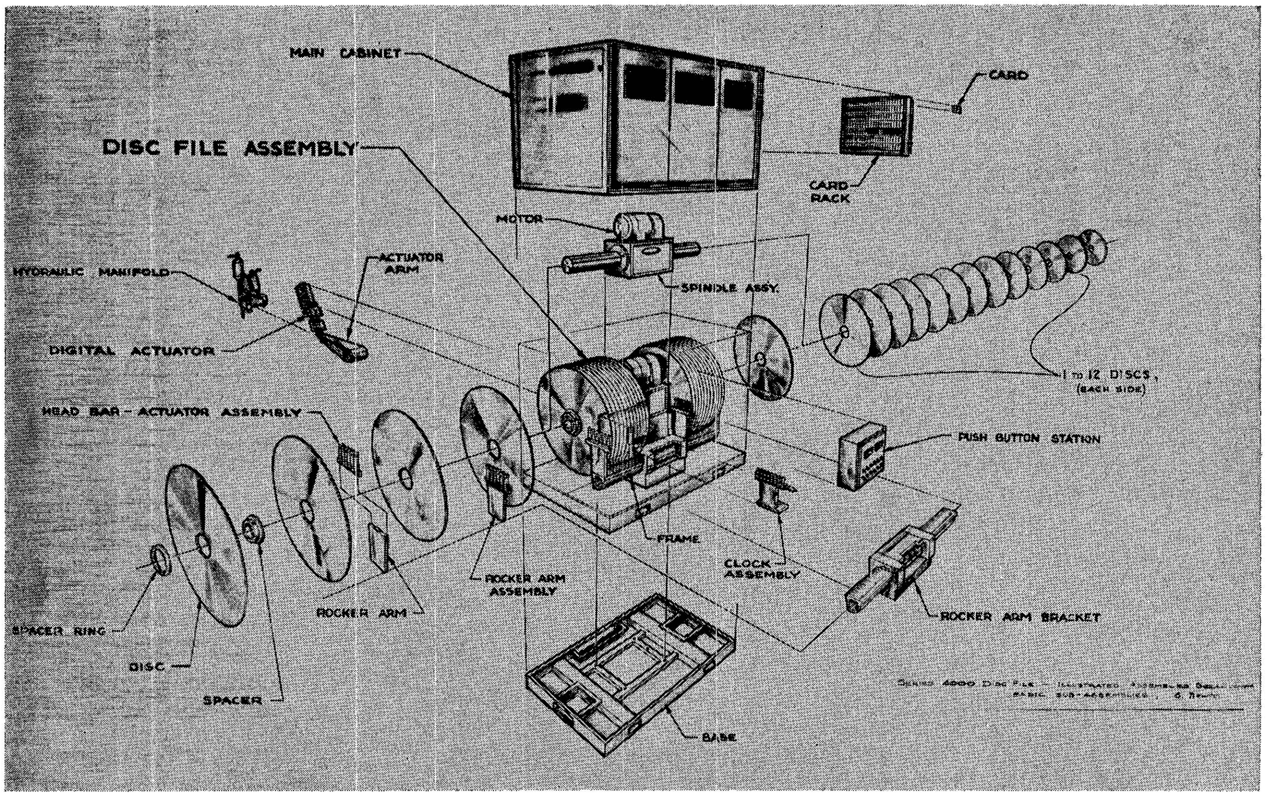



**Figure 7**

discs on the spindle, the discs are secured on accurately dimensioned spacer hubs which bolt together to locate the discs (**Figure** 8). No difficulty is encountered in removing a disc and its hub from a stack and replacing it with another.

Disc runout is held to less than 0.010 inch. Dynamic unbalanced excursion is held to less than 0.000090 inch.

Referring to **Figure 9,** the six magnetic heads which service each disc face are standard single-gap, ferrite pole piece structures mounted in aerodynamic head bodies which fly on the laminar film of air alongside the disc. Each head incorporates a center tapped coil. The coil halves are energized selectively to write binary ZEROS and ONES. Erasing is accomplished by writing new information over previously recorded data. Each head body is mounted on a precision self-aligning axle which permits the head to conform to the disc surface. The head face adjacent the disc is provided with an optically lapped surface of low friction material to minimize the effect of accidental head-surface contact.

The head axle is secured to the bifurcated ends of a spring reed extending upwardly from a head bar. A single head bar located between adjacent

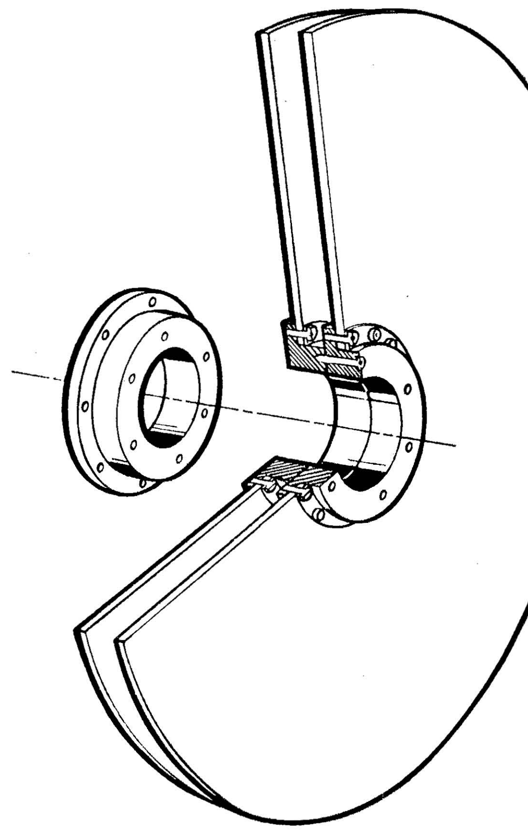


**Figure 8**

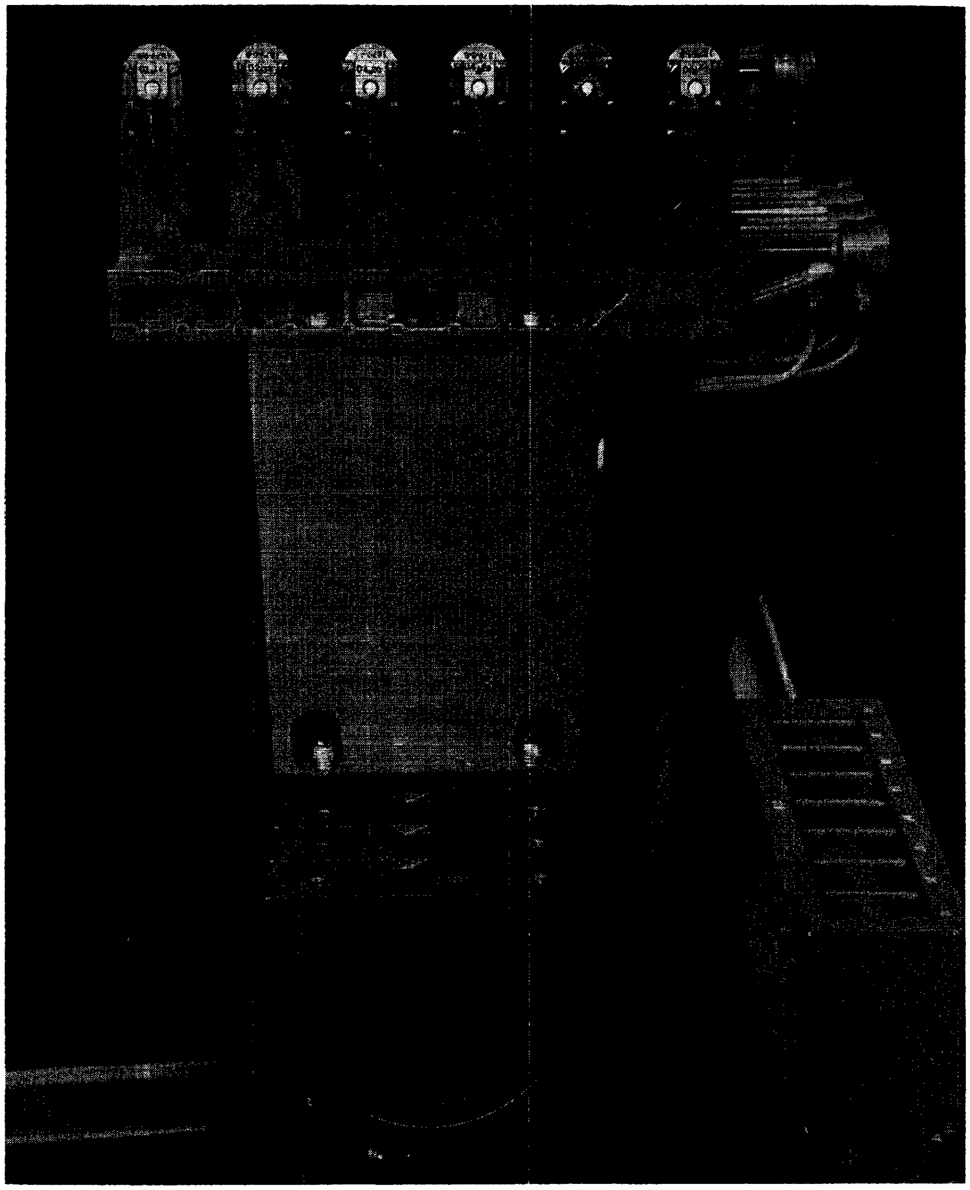


Figure 9

discs supports all 12 heads for the opposing disc faces. Individual head bars supporting six heads are provided for the end disc faces.

The heads are normally located a relatively large distance from the discs but are tensioned into flying position on operation of an hydraulic actuator at the forward end of the head bar. The mechanism whereby the actuator moves the heads includes a slide atop the head bar, an adjustable conepoint screw for each head, and a ball bearing spring reed. Adjustment of the flying portion of a head is made by turning the conepoint screw to vary the movement of the spring reed and, therefore, its preload.

Each head bar is secured on a rocker arm which swings the heads through an angle of approximately seven degrees to reach their 128 assigned tracks. All of the rocker arms are mounted on a precision rocker shaft which is journalled in a bearing housing secured to the front of the center pedestal of the machine (**Figure 10**).

The rocker shaft is rotated differentially to each of its 128 discrete positions by a digital hydraulic positioner operating through a rocker arm secured on the shaft (**Figure 11**).

## PRECISE DIGITAL POSITIONING

The digital positioner (**Figure 12**) is the most precise mechanism in the file. It has a repeatability in each of the 128 positions of its output shaft of 0.000025 inch or better. Internally, the positioner comprises seven piston and cylinder assemblies in tandem in a common bore, each controlled by a transfer valve. The seven transfer valves correspond to the seven binary values $2^0$, $2^1$, $2^2$, $2^3$, $2^4$, $2^5$, and $2^6$ in the memory address.

The seven pistons are extended and retracted—by hydraulic pressure—fixed distances proportional to the binary values $2^0$-$2^6$ under control of the valves. Each piston moves along with it all of the piston-cylinder assemblies ahead of it, creating, in effect, a binary adder. Metal-to-metal contact is utilized to achieve the required accuracy.

The output shaft of the positioner is connected to the rocker arm through a servo boost which provides moving power and also converts the rapid decelerations occasioned by metal-to-metal contacts in the positioner to an essentially cycloidal movement. The positioner shaft is pinned to the servo-spool of the boost, the power piston of the boost is tied to ground, and the boost body is connected to the rocker arm. The accuracy of the boost is better than $\pm$ 0.0001 inch – a decentering movement of this amount applying full hydraulic pressure to correct the position.

As mentioned earlier, the positioning system has an overall positioning repeatability at the heads of better than $\pm$ 0.0005 inch. In addition to this

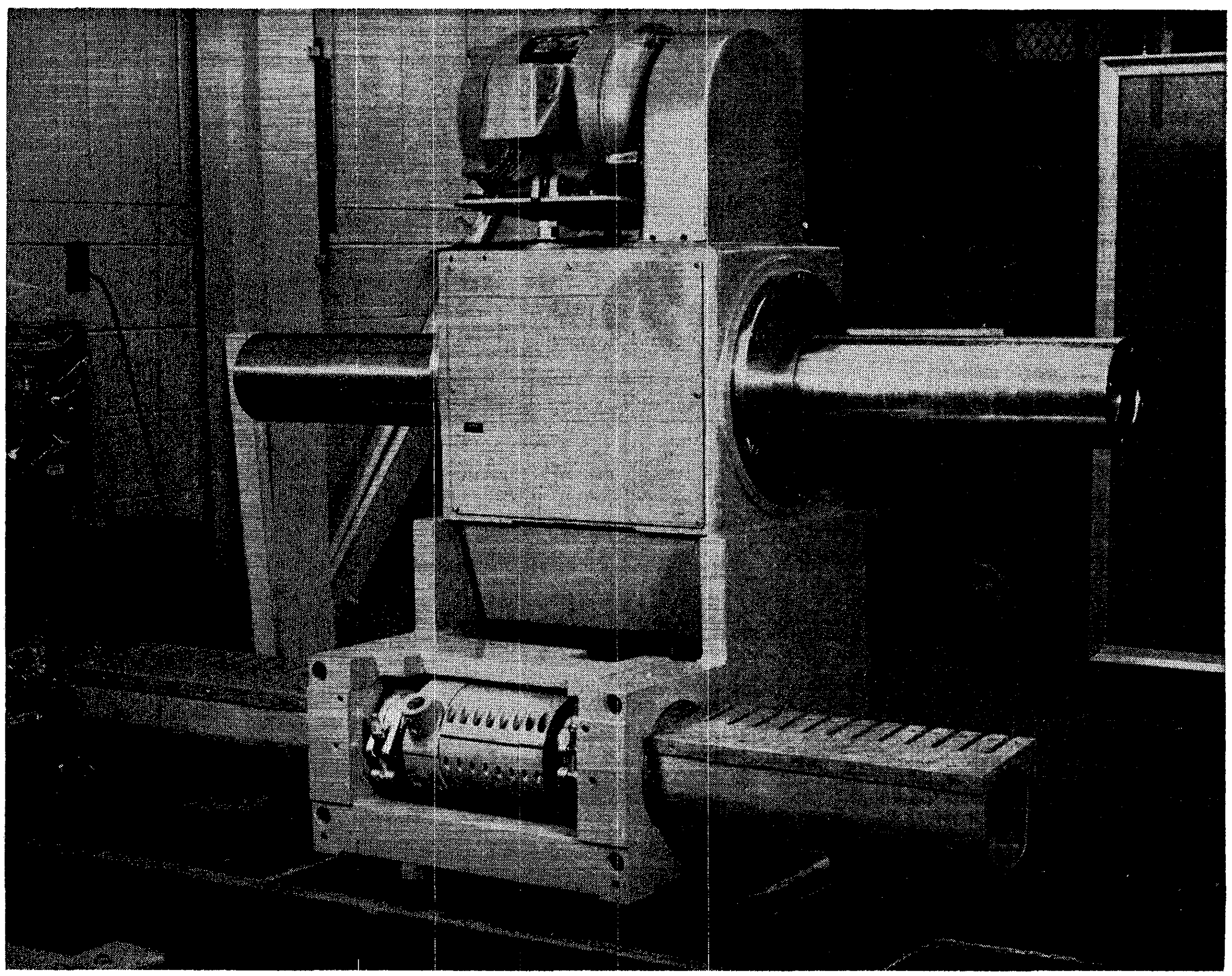


**Figure 10**

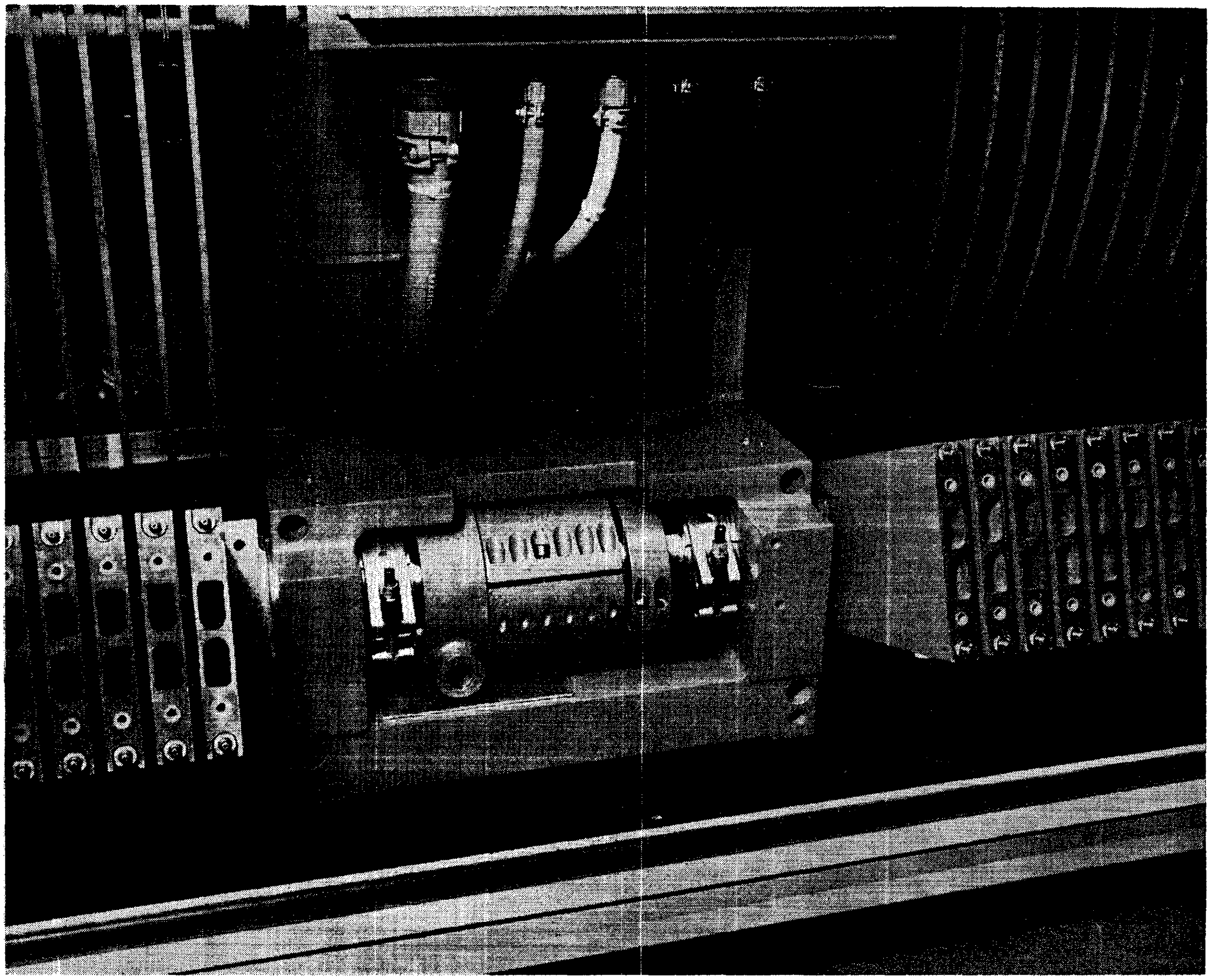


Figure 11

great precision, the system is extremely fast. **Figure 13** illustrates the access times appropriate to 6, 13 and 25 data disc files. As indicated, the longest access time, measured from application of the address signals and including settling time, is 165 milliseconds for a full two-inch (128 track) movement of the 25-disc machine. The same two-inch movement for a 13-disc machine is 140 milliseconds and for a six-disc machine it is 120 milliseconds. Track-to-track positioning times are as low as 30 milliseconds.

An extra disc is provided on each file for clocks and position verification control tracks. In a standard machine, the clock disc face is that immediately to the left of the center pedestal. The heads for this disc face are mounted on a fixed head bar atop a clock bracket secured on the rocker shaft bearing housing (**Figure 14**). The lengths of the clock head reeds and the clock bracket are chosen to have the same thermal growth characteristics as the data heads.

The clocks on this disc face serve the entire file. Additional clocks may be provided, if required, on the disc face to the right of the center pedestal.

In order to provide verification of track positioning and to check settling of the system, the outerface of the right-hand end disc is substituted for the outerface of the clock disc which is used for normal data storage. One of the heads serving this disc is assigned to reading seven-bit track addresses written continuously around each of its 128 tracks. Every eighth bit in each of these tracks is blank and is synchronized with a clock on the clock disc. Comparison of the prerecorded addresses is accomplished during these eight-bit

*(Continued on page 44)*

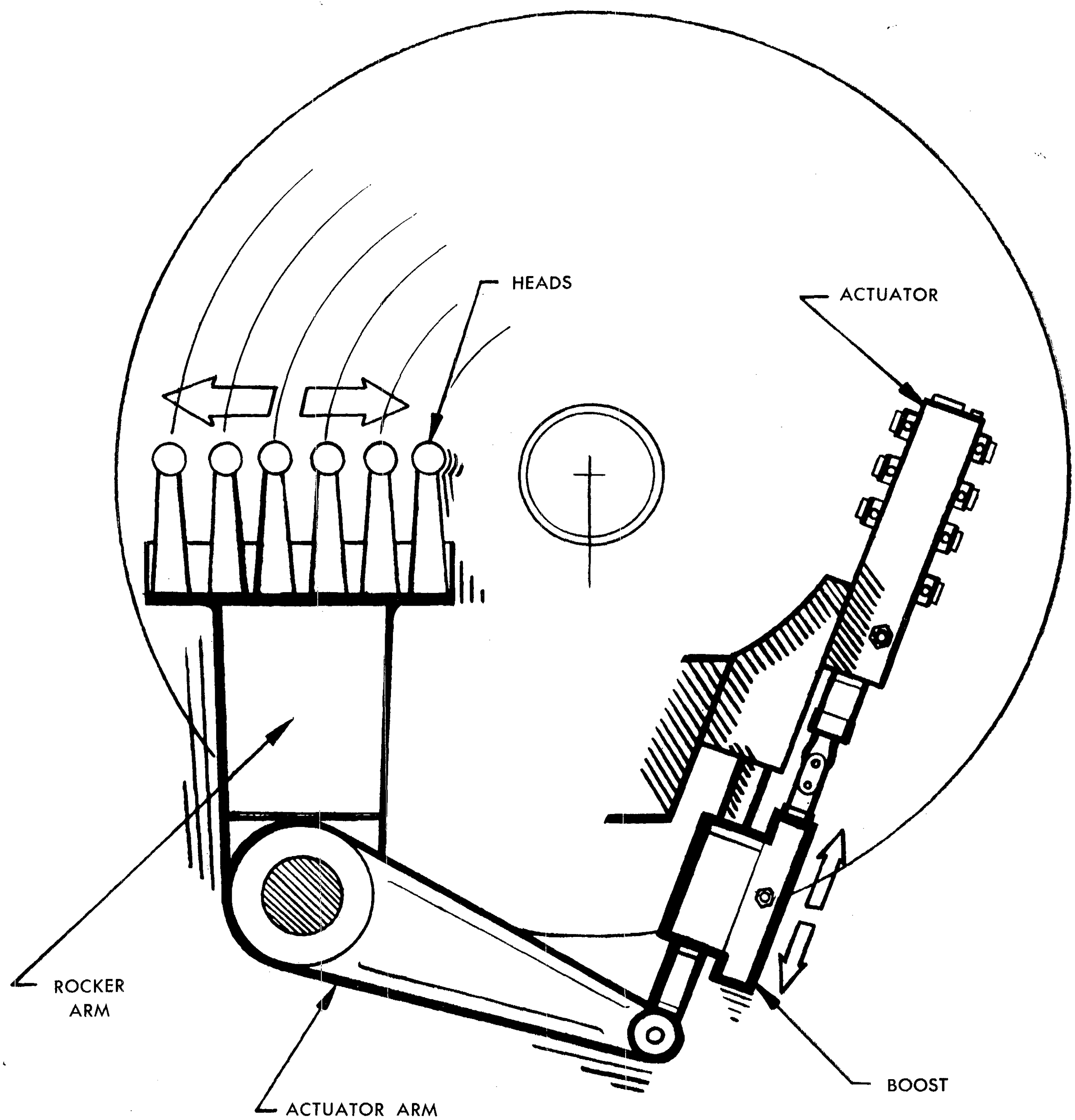



**Figure 12**

| File Size | Track-to-Track Access Time | Full-Stroke (128) Access Time* |
|---|---|---|
| Six Data Disc Series 4000A | 30 ms. | 120 ms. |
| 13 Data Disc Series 4000B | 35 ms. | 140 ms. |
| 25 Data Disc Series 4000C | 40 ms. | 165 ms. |

*Includes positioning, track verification and latency times.

**Figure 13. Average Access Times for the three Series 4000 Disc File machine sizes when equipped with a full complement of data discs.**

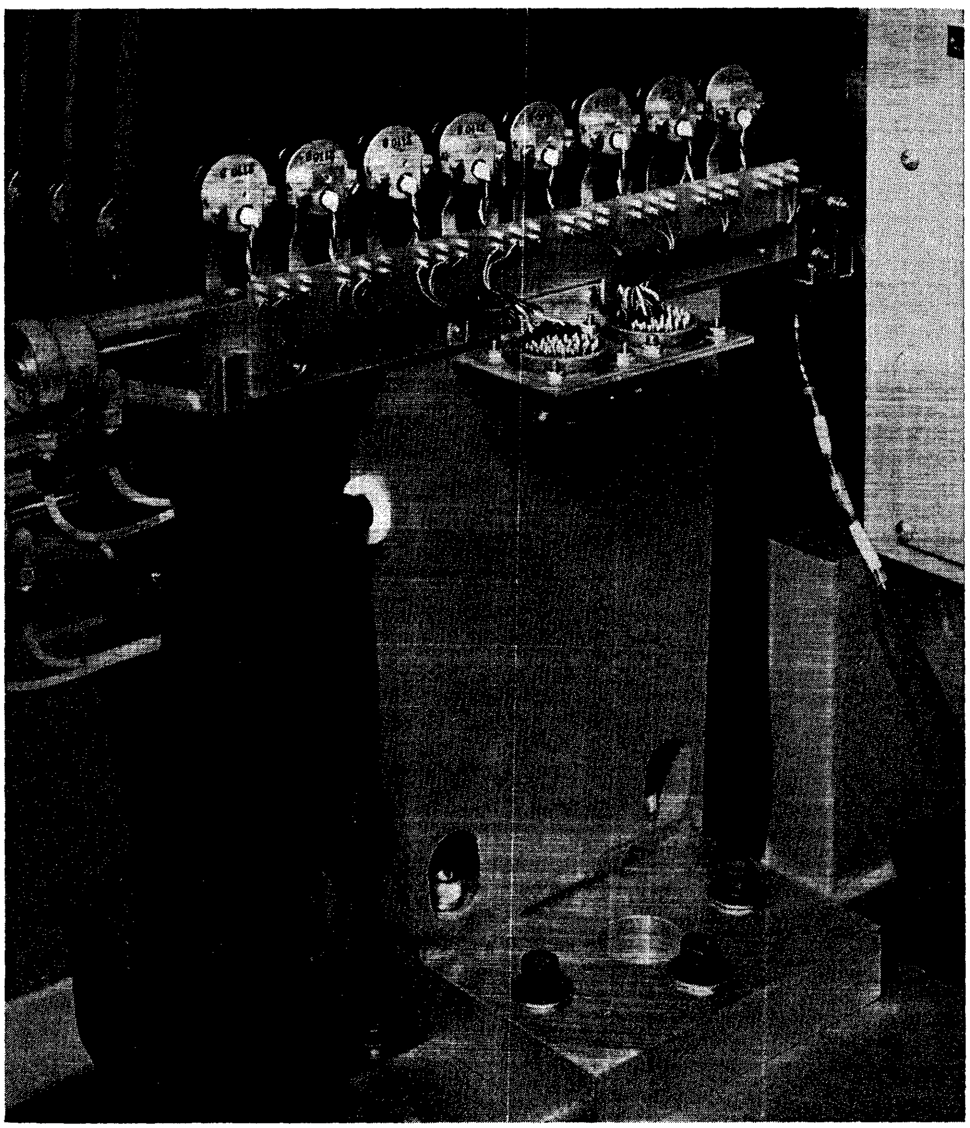


**Figure 14**

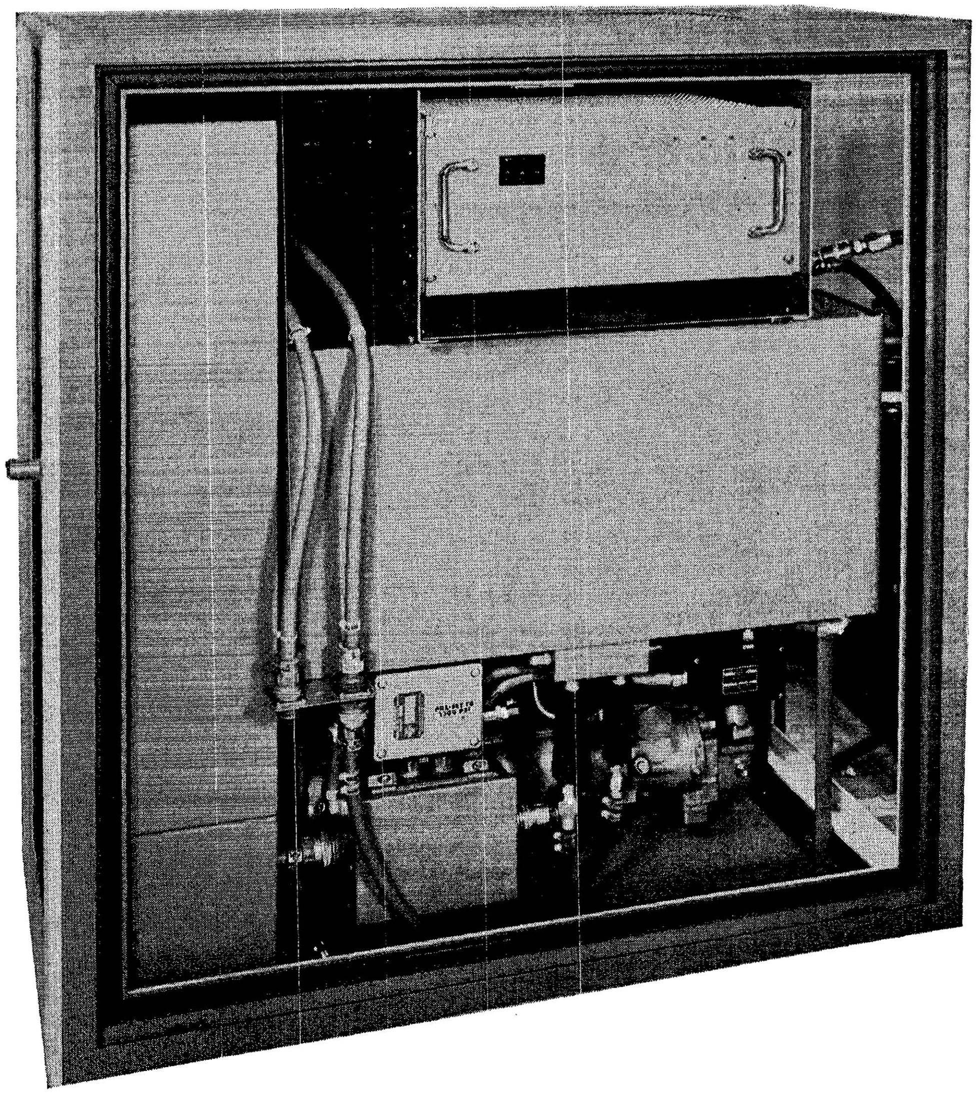


Figure 15

| Zone | Single-Bit Alteration Recording | | Self-Clocking, Block-Format, Recording | |
|---|---|---|---|---|
| | Maximum Track Capacity (285 BPI) | | Maximum Track Capacity (600 BPI) | |
| | Bits | 7-Bit Char. | Bits | 7-Bit Char. |
| 1 | 11,200 | 1,600 | 23,580 | 3,368 |
| 2 | 14,800 | 2,114 | 31,140 | 4,448 |
| 3 | 18,354 | 2,622 | 38,640 | 5,520 |
| 4 | 22,400 | 3,200 | 46,200 | 6,600 |
| 5 | 25,507 | 3,644 | 53,700 | 7,671 |
| 6 | 29,098 | 4,128 | 61,260 | 8,751 |

**Figure 16. Maximum recommended track capacities for each zone for different recording modes.**

APPLICATION NOTES

SERIES 4000 DISC FILES

224 KC SINGLE-FREQUENCY TRACK LAYOUT

25,804,800 BITS PER DISC 1200 RPM

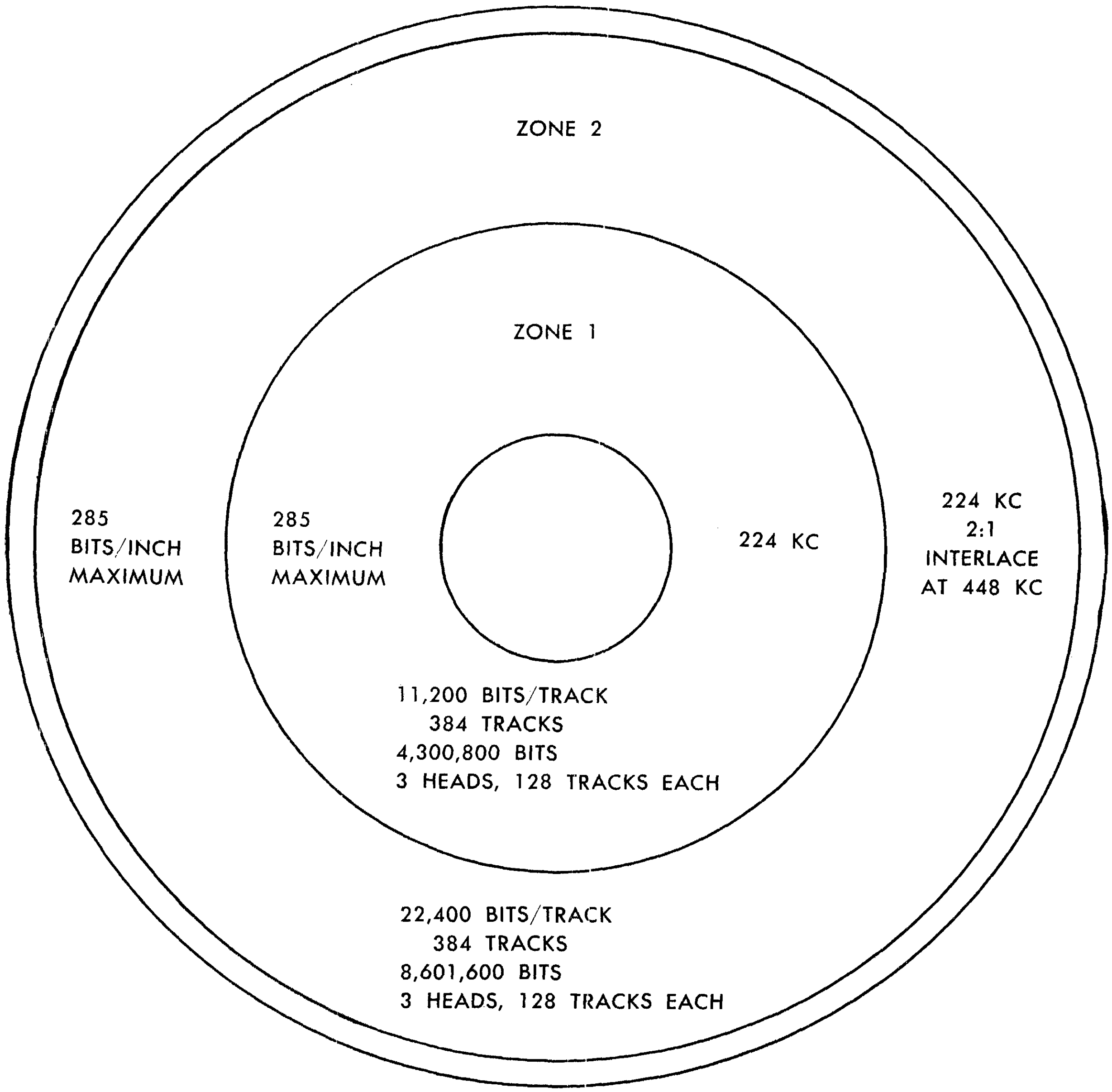



**Figure 17**

blanks. Successful comparisons for 20 milliseconds verifies positioning and system settling.

To complete the description of the 4000 series file hardware, it must be mentioned that the hydraulics supply is housed in an auxiliary cabinet (**Figure 15**). A variety of controls and safety interlocks are provided, but inasmuch as they are not of primary interest here, I will merely note them in passing.

## SERIES 4000 FILE APPLICATIONS

At this point, I would like to review the capabilities and limitations of the file in terms of record configuration, addressing, sector layout, and data transmission. The limitations imposed by the hardware are quite straightforward. At the recommended maximum pulse density of 285 bits per inch for clocked read and write, the maximum track capacity for each of the six zones is as shown in **Figure 16.** As shown in Figure 16, this can be more than doubled by giving up the clocked operation, recording at 600 bits per inch and going to block format techniques similar to tape.

Within this limitation, any desired record configuration may be used. Lower densities may be used if shorter tracks are desired. Each track may be divided into sectors as required. Tracks may be added together either in tandem or in parallel to achieve longer record lengths. Clocks are provided to fit the *user's* format including sector markers or addresses.

The format shown in **Figure 17** was generated to satisfy a user requirement for a universal track length of 10 sectors, each incorporating 160 seven-bit characters, coupled with a requirement for a data transmission rate of approximately 225 kc. The illustrated two-zone, bit-interlace system with a basic track (and interlace channel) length of 11,200 bits and a disc speed of 1,200 rpm fully satisfies the requirement.

The format shown in **Figure 18** was generated to satisfy a user requirement for handling file data at a nine-bit character rate of approximately 150 kc.

Here the six heads on a disc face read and write in parallel to achieve the high data rate. The disc is divided into two zones, each serviced by three heads with the track length in the outer zone being exactly twice that of the inner zone. The inner and outer zones are synchronized by the clock tracks so that two bits are written or read by each outer zone head while each inner zone head is writing or reading a single bit. The disc may be visualized as divided into nine-bit word sectors which are one bit long in the inner zone and two bits long in the outer zone. Thus, each nine-bit word is transferred to and from the disc in two sections — a three-bit parallel section at frequency $n$ and a three-bit parallel, two-bit serial section at frequency $2n$. Simple conversion registers in the control electronics convert this to straightforward nine-bit parallel at frequency $n$.

Another variation of this technique is shown in **Figure 19.** Here the requirement is to transfer a 60-bit word to or from the file at approximately a 56-kc word rate (3.36 megacycle bit rate). In order to achieve this, the 12 heads servicing both sides of each disc are utilized in parallel. The disc face is divided into three zones having track lengths appropriate to the ratio 3:5:7. The disc face may be visualized as divided into half word sectors, each being three bits long for the two innermost zone heads, five bits long for the two center zone heads, and seven bits long for the two outermost zone heads. Thus each sector incorporates 30 bits and back-to-back sectors on both disc faces incorporate the complete 60-bit word. The clocks for the three zones are synchronized to ensure phasing of the recording and reading processes. Conversion to and from straightforward 60-bit parallel words is accomplished in the control electronics.

Those three clocked read-write formats are typical of the wide range of user requirements which can be accommodated by the standard electromechanical file merely by providing appropriate clocks and addressing electronics. This range of formats has been significantly widened, however, through the introduction—by a user—of a clocked-write, self-clocked-read system.

**Figure 20** illustrates the first simple application of this technique. Of primary interest is the substantial increase in packing density which is achieved from 285 bits per inch to 388 bits per inch in this instance. Densities of 600 bits per inch will be standard. A simple extension of this shows that disc capacities of over 40 million bits and a file capacity up to 1.6 billion bits are readily achieved. In the illustrated initial application higher capacity was not a requirement; rather the track lengths and frequencies shown were the requirement. Even so, disc capacity is increased to over 34,000,000 bits and file capacity to over 800,000,000 bits with no change in the electromechanical file.

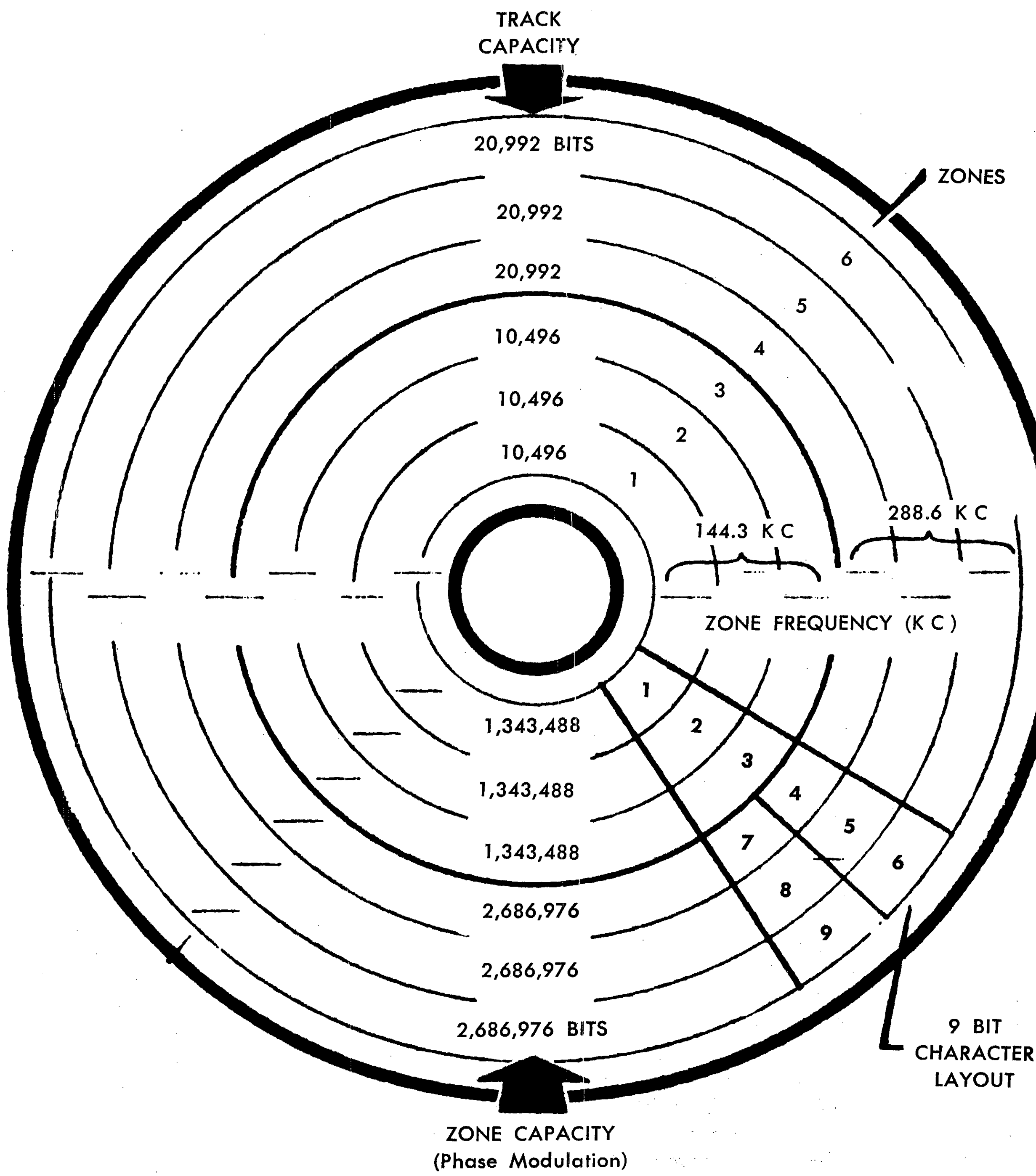
These few examples of how the files are being used indicate that we have achieved our goal of providing a basic file which, within the state of the art, is not equipment limited but rather technique limited. As more and more sophisticated users are given the opportunity to vent their creative abilities on applying this equipment to their problems, many more techniques not now apparent to us will appear.

Summarizing, the user specifies his character and word structure, desired address capacity, overall capacity and desired data frequency, and the following variables are available to achieve his specifications.

Serial or parallel operation

*(Continued on page 48)*

**SIX BIT PARALLEL — NINE BIT CHARACTER SYSTEM**

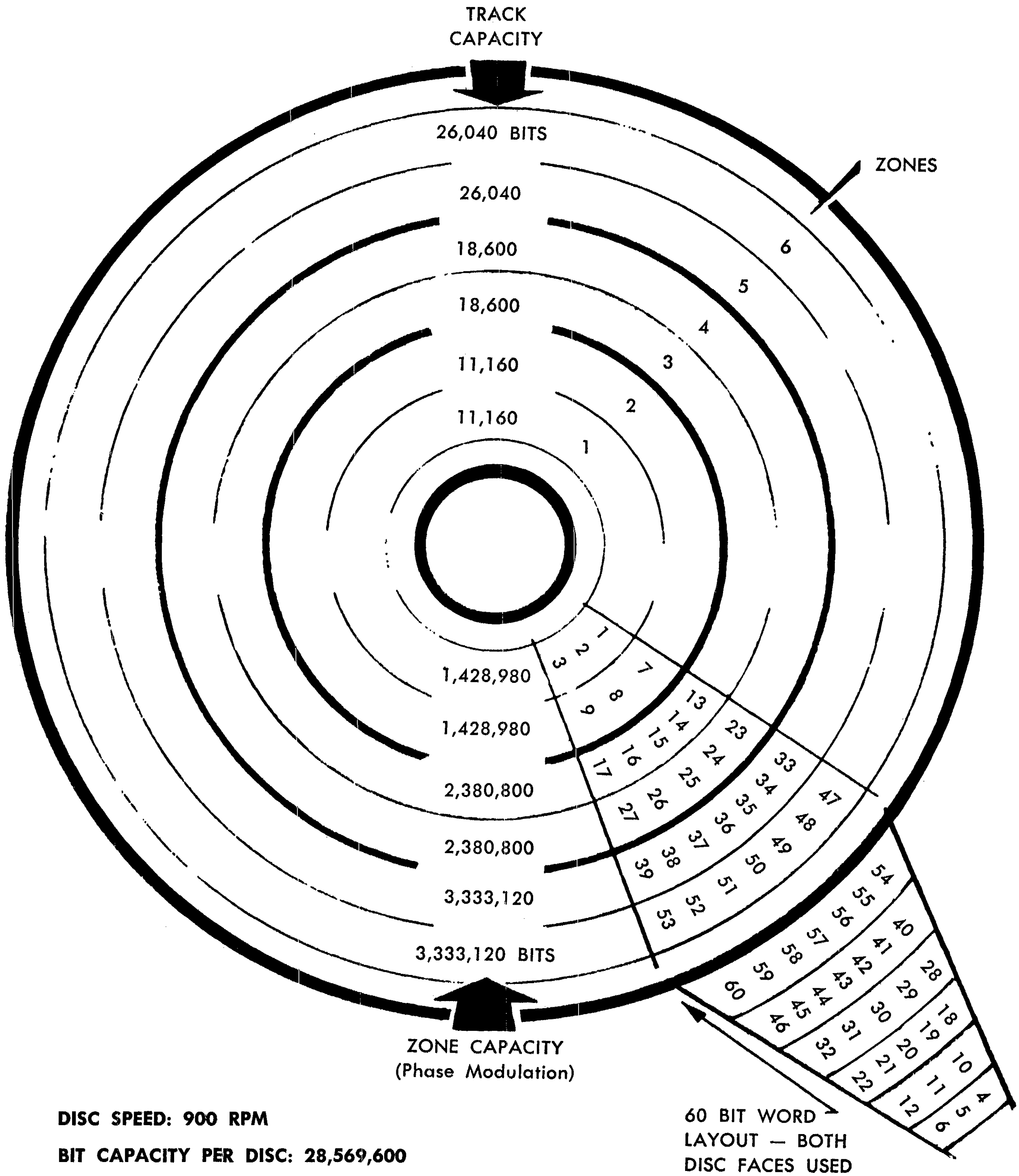



**DISC SPEED: 900 RPM**

**BIT CAPACITY PER DISC: 24,182,784**

**Figure 18**

**12 BIT PARALLEL — 60 BIT WORD SYSTEM**

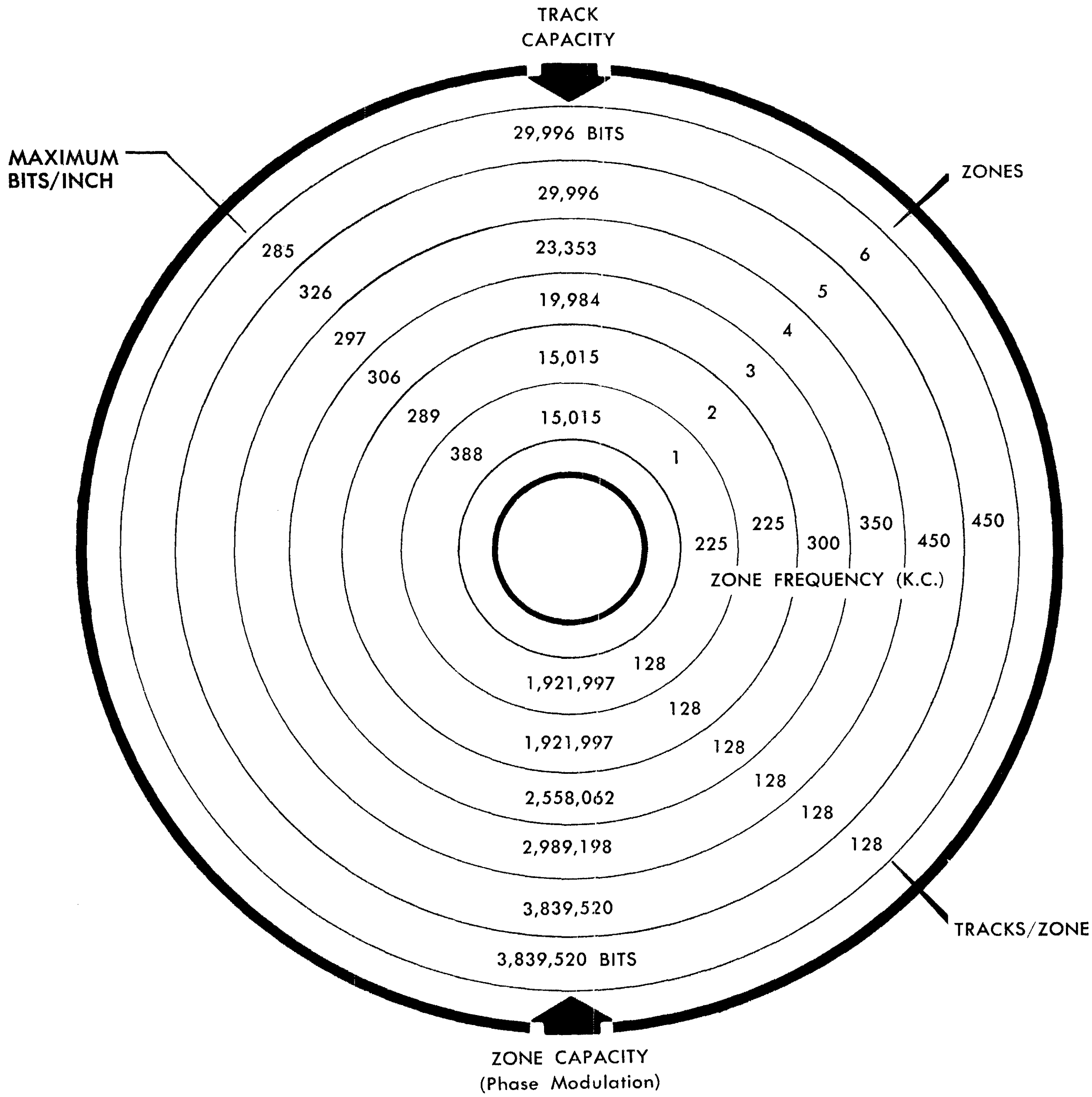



**DISC SPEED: 900 RPM**

**BIT CAPACITY PER DISC: 28,569,600**

**Figure 19**

**CLOCKED WRITE — SELF CLOCKED READ SYSTEM**




**DISC SPEED: 900 RPM**

**BIT CAPACITY/DISC SIDE: 17,070,294**

**BIT CAPACITY PER DISC: 34,140,588**

**Figure 20**

*(Continued from page 45)*

Bit and track interlacing

Clocked read-write or clocked write, self-clocked read

Track length in each zone

Disc speed

Number of discs.

## RELIABILITY

Another user consideration which must be discussed is reliability. Reliability of a magnetic memory device is frequently divided into three categories:

1. *Recoverable errors.* An error which corrects itself on repeating the operation. An error rate of better than one error in 2 x $10^8$ bits processed is readily achieved with the Bryant file.
2. *Semi-recoverable errors.* An error which can be corrected by unscheduled maintenance. Bryant file experience indicates that this type of error is extremely rare once the file is set up as described later.
3. *Non-recoverable errors.* An error which cannot be recovered by any means. This type of error is virtually eliminated by setting up the file as described later.

The error rate for a disc file is dependent upon several factors:

1. The mechanical integrity of the machine
2. The integrity of the positioning device
3. The integrity of the recording medium—read/write head—read/write circuit system
4. The integrity of the control electronics.

Factors 1, 2 and 4 are relatively straightforward and can be evaluated clearly. Factor 3, however, is one which presents a great deal of difficulty in its evaluation. The principal seat of this difficulty is in determining the rules for evaluation. Looking first at the coating itself, we can begin by stating that we want a dropout-free magnetic coating. But what is a dropout-free coating? The only answer I can give is "a coating which does not exhibit dropouts when used with specific read/write heads and circuits in a specific way". All magnetic coatings exhibit varying magnetic properties from one location to another due to undersurface variations, non-homogeneities of the coating material, variations in thickness, surface variations and other factors. Variations in the flying characteristics of the read/write heads add to the effect of these coating variations.

In order to achieve dropout-free operation, therefore, the write circuit and, to a greater extent, the read circuit must be able to compensate for these anomalies which appear principally as variations in playback amplitude, and to a lesser extent, as signal distortion and variations in the background noise level. Peak detection, zero slope amplifiers used with self-clocked-read systems, exhibit superior capabilities to compensate for wide variations in playback amplitude and are relatively immune to phase shift and noise. Threshold detection amplifiers used where selective alteration of a single bit is a requirement are much more sensitive to playback amplitude modulation and noise.

Thus a coating which is "dropout-free", if used with a peak detection system, may produce dropouts if used with a threshold detection system.

In actual practice, dropout-free coatings are a practical production item for peak detection systems. For the threshold detection systems now available to Bryant, however, the dropout rate may be as high as one in 3 x $10^7$ bits by Bryant's definition of dropouts, that is, any bit position which does not produce a playback amplitude of at least 70 percent of the envelope minimum for that track. This definition can be attacked as too stringent, but it is considered necessary for reliable long term operation.

Referring back to the three kinds of error which are experienced with files, all three varieties can be minimized and type 3, non-recoverable errors, essentially eliminated by maximizing the initial dropout rate of the file and deleting all dropout positions from storage, that is, classify any bit position which is marginal or even questionable as a dropout and provide a technique for ignoring the existence of this position in the memory.

Adoption of this philosophy has the added advantage of preparing the system to cope with dropouts which may appear later due to operational damage or deterioration of components.

A number of techniques have been investigated by Bryant to effect dropout-free file operation, even though a modest number of dropouts do exist on the discs.

These techniques are readily divided into three categories:

1. Programmed elimination of addresses containing dropouts
2. Automatic substitution of a perfect storage location for one containing a dropout
3. Error correction codes.

Programmed elimination of bad addresses simply requires restricting the programmer to a log of perfect addresses rather than permitting him to use all addresses. This technique is simple and requires no equipment, but it does require that pro-

## DROPOUT COMPENSATION TECHNIQUES

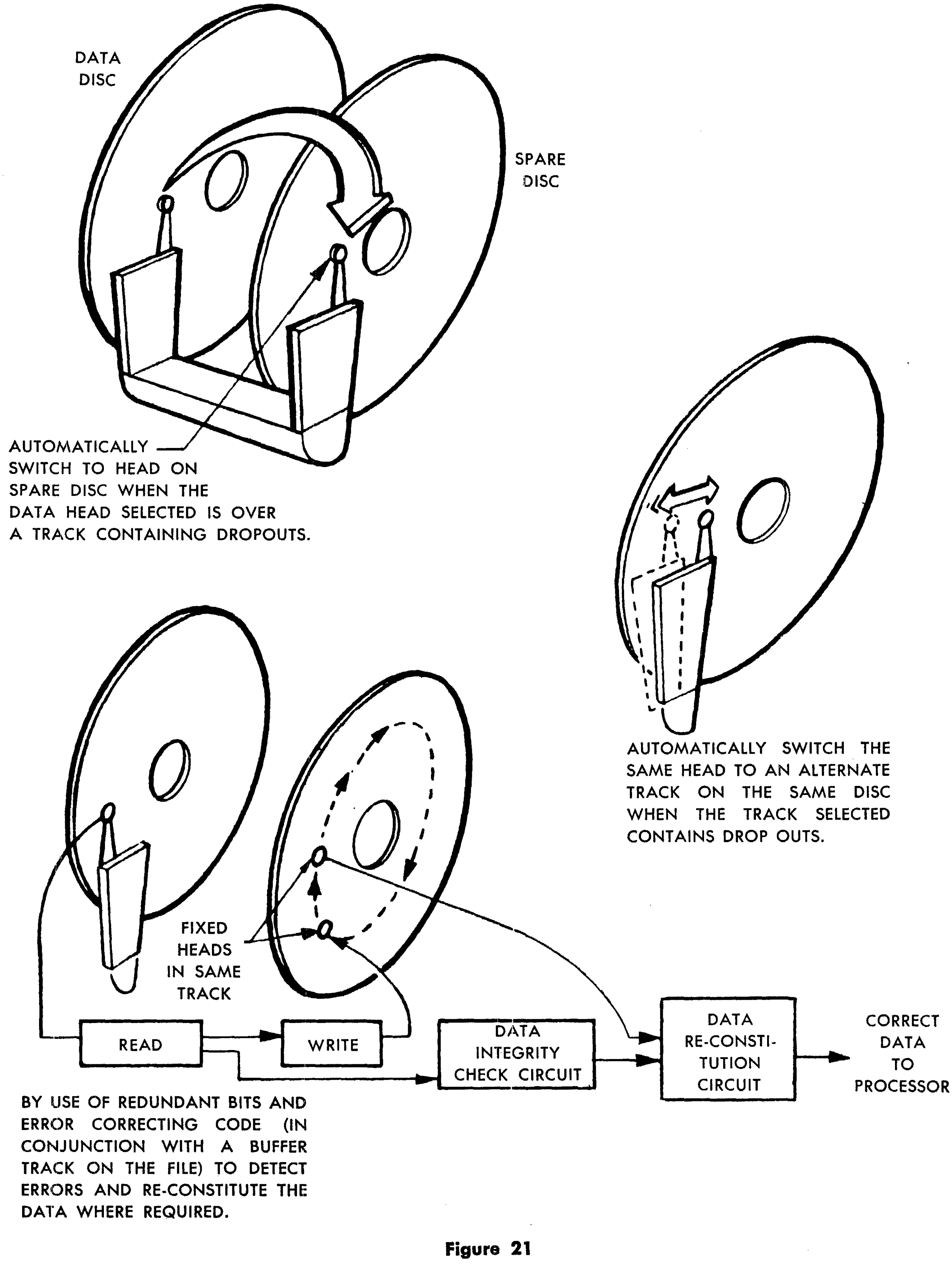



**Figure 21**

grams be modified to fit the particular dropout map for each file.

Automatic substitution of perfect storage positions may be accomplished in several ways as shown in **Figure 21.**

a. Automatic substitution of a spare head on a spare disc

b. Automatic substitution of a spare position for the same head. Here, the 128th track for each head would be reserved as a spare.

c. Automatic substitution of another position in the same track (not shown in **Figure 21**). Here, spare bit positions are available in each track and dropout bit positions are eliminated by gating off the clock pulses defining these positions.

The controls for effecting a and b are quite straightforward:

a. A "header" may be recorded in each track to identify the track as containing a dropout and to produce the substitute address.

b. A spare head on the spare disc may be used to identify the tracks which contain dropouts in each position, and to supply substitute addresses.

c. The log of bad addresses and their substitutes may be maintained in the computer memory.

Error correction codes are probably the most sophisticated approach to compensating for dropouts. The use of a correction code not only detects and corrects errors due to dropouts, but all errors due to any cause, including transients. Without going into the mathematics of it, a Fire code to detect and correct errors up to six bits in length in a storage block 1,000 bits long requires the addition of 18 control bits to the block.

As shown in **Figure 21,** data including the code bits are fed through an integrity checking circuit which produces a series of logical controls for a data reconstitution circuit which corrects any erroneous bits as the data are fed through it from a delay buffer. The computer memory, an external buffer, or a revolver track on the control disc in the file, may be used to effect the required delay, that is, at least equal to the number of bits in the data block.

The basic mathematics of this approach are described in *Error Correcting Codes* by Wesley Peterson, MIT Press and John Wiley, Joint Publications.

Summarizing, the Bryant philosophy on "dropouts" is:

1. Dropouts can be expected due to manufacturing variations and operational damage.

2. All marginal or even questionable storage locations should be classified as dropouts immediately upon their detection.

3. Techniques should be provided for achieving "dropout" free operation even though dropouts exist.

In closing, I would like to re-emphasize that Bryant's position in the data processing field is that of a supplier of peripheral digital hardware to the computer manufacturers. We support the strong electronics and systems engineering capabilities of our customers with electromechanical capability in depth. Our ability to produce hardware like the 4000 Series File depends in large measure on the requirements placed upon us by the systems analysts who study the needs of the electronic data processing market and generate equipment specifications to satisfy them. Therefore, we welcome every opportunity to learn more about the needs of the ultimate users of our equipment. All users and potential users should feel free to call upon us for our assistance in solving their mass memory requirements through application of the 4000 series files and other mass memory products under development.

by Donald K. Sampson
Data Products Corporation
St. Paul, Minnesota

# File Storage and the Data Products DISCfILE Series dp/f-5020

THE DISCfILE SYSTEM is a complete peripheral subsystem consisting of two basic elements. One of these is the disc unit and the other a logic unit. These two units are available in a single or separate cabinets. The logic unit contains the circuitry necessary to control the disc unit as well as that to connect the DISCfILE into the computer system. The logic unit is organized to accommodate the users' specified record format and interface requirement. The disc unit is produced in a limited number of models and is described in detail in the following sections.

DISCfILE systems are available in a variety of configurations. These range from systems which consist of one disc unit connected to one logic unit, composing a subsystem of which up to 32 may be interfaced to a single GE 225, to a system of four disc units connected to a single logic unit and interfaced to a Control Data Corporation 1604/A or 160A. The systems allow complete time shared seeking in the case of the GE 225, and economy of electronics for non-time shared seeking in the CDC case. There are also DISCfILE systems in which one or more disc units are connected to two logic units, allowing the ultimate in time shared seeking by a single computer or a complex of two or more computers. Connection of several disc units, as in the CDC case above, to two such logic cabinets allows for dual access to extremely large capacity files.

Delivered systems, which amount to approximately 40 DISCfILES at the date of this writing, also include four DISCfILES which read and write in parallel with electronic deskewing. These DISCfILES are operating with a Sylvania 9400 computer in the Pentagon. The majority of all systems described have been delivered during the last half of 1962, and quantity production is continuing at the St. Paul, Minnesota plant of Data Products Corporation.

These units are the culmination of a product development program for which research was completed in the period from the middle of 1959 to the first of 1960, product design to the middle of 1961, with prototype and small scale production continuing until the quantity deliveries beginning in 1962. The product development was done by a coherent group of personnel in St. Paul, which has been substantially intact from the initiation of the program.

These introductory paragraphs are intended merely to form a perspective picture of the system, the status of its availability and its origin. The main body of this paper will be an attempt at an exposition of *what* a DISCfILE is, *how* it is and can be used, and *why* this particular random access memory is designed the way it is. The first section will deal largely with the electromechanical disc cabinet, the second section will deal largely with the logic cabinet, and the third with details of both cabinets and their system configurations.

## THE DISC UNIT

### General

The disc unit is composed of a rotating assembly of discs accessed by multiple heads on independent positioners; electromechanical switching and solid state drive circuits for the positioners; diode switching for the heads, reading, writing and erase circuits; pneumatic system for the heads; cooling fans; and the required power switching for remote and off line test control of the A.C. components. The rotating assembly is mounted in a rugged frame and pressurized by filtered air. The fully enclosed cabinet is approximately five feet high, five feet long, three feet deep, and weighs about 2,500 pounds.

### The discs

The 31-inch diameter discs are made of one-eighth inch magnesium tooling plate coated with an iron oxide dispersion. They are in quantity production by the Minnesota Mining and Manufacturing Company. Each disc is permanently mounted on an iron washer. This subassembly is statically balanced, thus assuring dynamic balance of the entire rotating assembly with complete disc interchangeability in the field.

A full disc unit contains 18 discs, 16 allocated for data, the outside two providing an aerodynamic baffle. Fixed control tracks are recorded on the top baffle. DISCfILES can be supplied with four, eight, or 12 data discs for the small capacity system with complete field expandability assured.

Currently DISCfILE deliveries are being made with a maximum of 400 bit per inch modified Manchester code recording capability. All references made to reliability in this paper are based on actual tests at this density of 800 flux reversals per inch. Track spacing, which will be dealt with in detail in a later section, is a conservative 26.7 tracks per inch. The disc recording format has been optimized for two frequencies of recording, adjusted slightly to accommodate a full binary number of tracks. The resulting format is thus 128 inner zone tracks plus 128 outer zone tracks per disc side, the innermost track of each (at 400 bps density) at a radius of approximately five inches and 10 inches respectively. Each inner zone track contains about 12,500 bits and each outer zone track contains 25,000 bits. The total disc capacity (both sides) is thus 2 x 128 x 12,500 + 2 x 128 x 25,000 = 10,200,000 bits. A portion of the disc capacity is lost to formatting and control requirements depending on the application. This topic will be dealt with at length in a later section. A very significant point, however, is that *none* of this capacity is lost to bad spots in the magnetic medium.

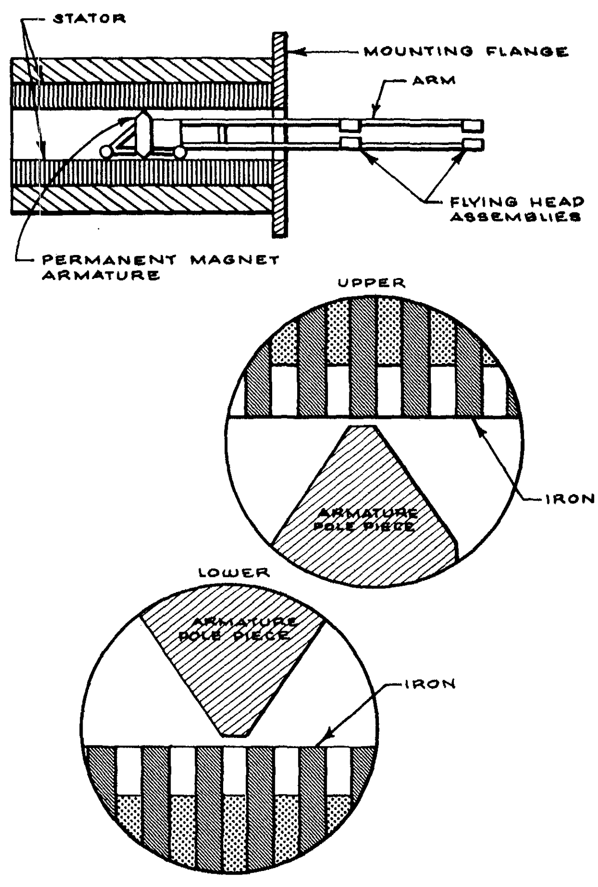
## Bad spots

It is not economically feasible to coat discs so that the surface is complete, continuous, and uniform. Pin holes and small areas of low oxide density are bound to be present. These can cause bad spots.

At one time it was considered that it was likely that discs would be damaged in service, but this is no longer a consideration. In present discs, one or two stages of "primer" are used between the oxide coating and the base metal. As a result of this the adhesion is so good that it is almost impossible to strip or flake the oxide from the metal. Further, the oxide coat is so tough and hard that when "head crashes" are deliberately induced it takes many hours of forcing the head into the disc before the oxide is worn off. Under these circumstances, although the possibility of a disc being damaged in service cannot be denied, it is very unlikely.

It is possible to design the system so that pin holes and irregularities do not introduce bad spots. In a well coated disc the probability of an imperfection above some maximum dimension is sufficiently low that the imperfect discs can be rejected. If a track width is employed which is substantially wider than the maximum acceptable pin hole, and if time-domain recording is used, then the imperfections do not introduce any perceptible effect during reading or writing. Such a system does not exhibit bad spots.

If large imperfections are present, or if the track widths must be very narrow, then imperfections will introduce bad spots. This problem must be




LINEAR POSITIONING MOTORS

**Figure 1**

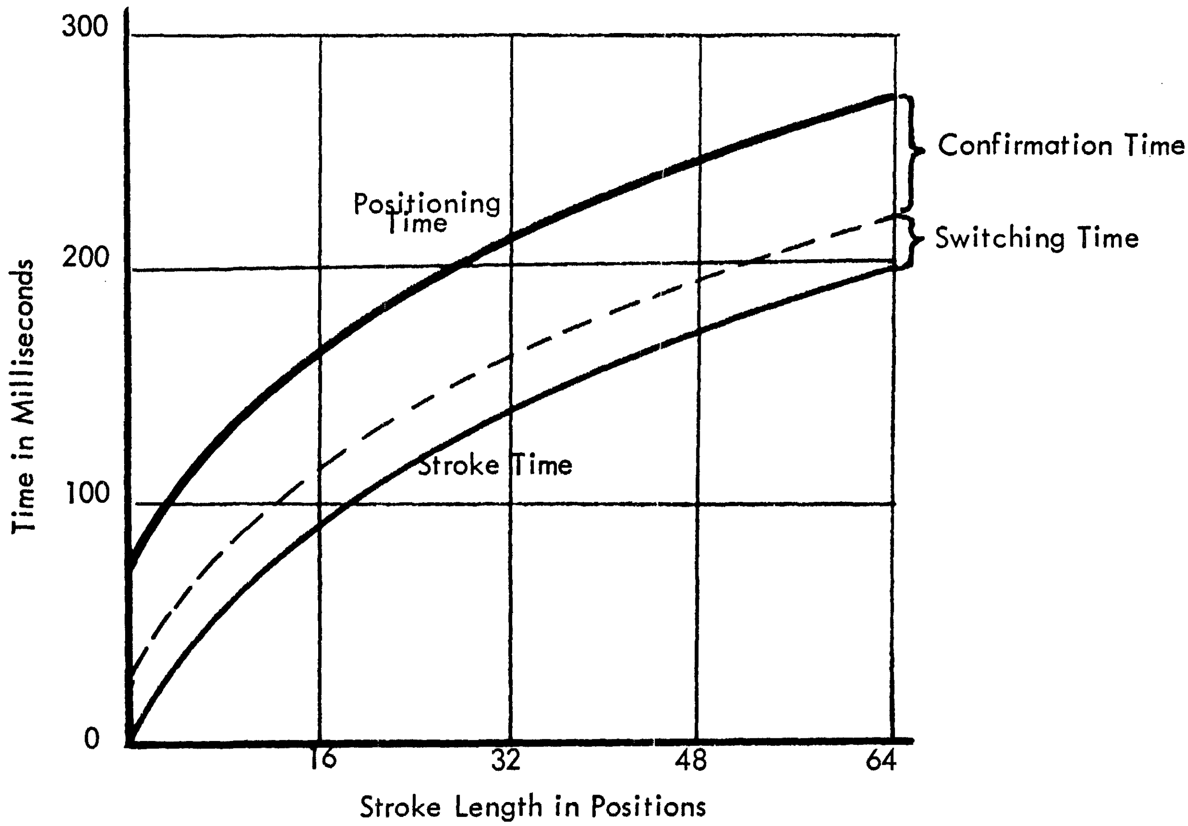



**Figure 2. ACCESS TIME OF MODEL dp/f-5020**

dealt with by incorporating spare tracks or spare discs into the file. Then bad spot logic must be used to change to an alternate disc or track in those positions where bad spots exist. The same effect can be achieved by programming, but this method is wasteful of space and time unless the probability of a bad spot existing is very low.

### The head positioners and heads

The head positioner and heads make up a complete and independent subassembly for each disc. The head positioner is made up of two parts, a stator, and armature, which together comprise a linear D.C. motor as shown in **Figure 1.**

### The stator

The stator is much more than its simple name implies. Essentially the stator is a magnetic detent mechanism comprised of two series wound, tapped iron core solenoids. The location of the detent, a restoring field, is determined by the tap at which current is applied. The resulting restoring force field at the selected position is approximately 100 pounds per inch. Since the armature is repelled from all positions other than the one selected, true random motion may be achieved without "stepping" from position to position between the last position accessed and the next one desired.

The stroke length of the position is 4.725 inches made up of 64 positions spaced 0.075 inch apart. Motion of the positioner is gentle but rapid. This gentleness is due to the fact that there are no impact accelerations due to the metal-to-metal stopping forces of the usual mechanical detent mechanisms, but instead, the eddy current damped oscillation of a permanent magnet settling into a forced field null. The speed of this motion is shown in **Figure 2.**

Stroke time, the actual travelling time of the armature carrying the heads, is a smoothly increasing time starting from zero for no motion to approximately 205 milliseconds for full stroke. An additional constant time delay of 20 milliseconds must be added to these times to allow for switching off the current to another positioner or to another position on the selected positioner. An additional constant time delay of 50 milliseconds must be added to these times to allow for sensing, by reading data reliably with the selected head, that positioning has been reliably completed and the selected head has settled on the selected track. The resulting displacement versus time curve for a positioner is thus a smoothly rising time of 70 milliseconds for no motion to 275 milliseconds for full stroke. These figures are representative of current production units. Data Products Corporation derates these times by approximately 30 milliseconds when writing specifications to allow for variations in manufacturing.

**The armature**

The armature of each positioner is a precision cast cage which carries a high efficiency ceramic magnet and a pair of silicon steel pole tips. Extensions of this cage make it a wheeled cart, the wheels providing guidance on the precision rails of the stator for the moving assembly. Two hollow rectangular extrusions extend from the front of the cage, with two flying head assemblies mounted to the rear of each arm. Ribbon conductors running along the length of the front edge of the arms and routed to the rear of the cage are the head leads. These leads are terminated in a printed circuit card containing the head select diodes and associated circuitry. This card plugs into the rear of the stator. The hollow arms carry air to the heads from a tube extending through the armature cage.

A flexible hose attached to the rear of the cage also plugs into the rear of the stator. Removal of heads for cleaning and inspection is simple and rapid, requiring only the two previously mentioned disconnections and pulling the armature from the cage. No adjustment whatever is required for track positioning since the position determining stator remains fitted to the shroud.

**The heads**

Four flying head assemblies are rigidly mounted to each armature. They are mounted to the hollow arms in two opposed pairs spaced along the arms by the stroke length of the positioner. This arrangement insures that no net force is applied to the disc by the heads, thus minimizing deflections of the discs during head motion. The head pads proper are the familiar slider bearing heads in general use in all known disc memories. Boundary layer separation of the head from the disc is approximately two to five hundred micro inches. The heads are forced towards the disc by pneumatic pressure on a ball piston integral with the pad. Clearance of the ball in its mating cylinder mounted to the arms provides a controlled air leak which lubricates the piston-cylinder pair.

The use of pneumatic loading of the heads assures many desirable features, notably constant head loading force independent of distance from the arm to the disc, the valuable feature of absolute predictability of equal loading of all heads, as well as a rapid positively controlled fail-safe retraction (by means of a small retraction spring) by dropping the air pressure in response to any abnormal operation signal. This feature of the design has resulted in heads requiring no adjustment and extremely relaxed tolerancing of disc and arm location.

Each head pad carrries two ferrite read-write core assemblies located at the rear edge of the pad. Read-write gap width (track width) is 0.025 inch and erase gap width is 0.040 inch. The track spacing of 0.0375 inch is achieved by spacing these heads an odd number of track spaces apart and interlacing the tracks laid down by the 0.075 inch step positioner.

**The shroud**

The shroud of the DISCfILE is a closed cylinder completely surrounding the discs. It is roughly divided into quarters, the front and rear quarters being closely fitting plexiglass windows and the two side quarters providing mounting area for the positioners. Most of its many functions are obvious to those familiar in the art.

One of the most important functions is to exclude the inevitable dirt; therefore, the only apertures are those provided for the head arms. The close fitting shroud also substantially reduces the power required to drive the discs and reduces noise. The use of a metal in the shroud whose thermal coefficient of expansion nearly matches that of the discs and head arms compensates, to a large degree, head position drift due to uncompensated thermal expansion occurring from the time the unit is turned on until thermal equilibrium is reached. The use of the closed shroud principle has resulted in the DISCfILES being reliably operable in widely varying temperatures from the moment the discs are up to speed.

The linear positioners are mounted on the shroud so that the motion of the head gap is purely radial. This provision obviates the tracking error of heads swung through an arc and eliminates the need for rotary head adjustment or pre-setting. The use of equally spaced arms between the discs which are always baffling the discs by at least the outer third of the radius, tends to equalize the pumping effect of the discs regardless of arm position and motion. See **Figure 3.**

### The switching and electronics

The availability of inexpensive, reliable dry reed switches is a determining factor in the economic feasibility of the DISCfILE. All positioner, position, and head group switching is done with these components. Dry reed switches are mounted on three circuit board types, one with 12 reed switches actuated by a common rectangular coil, one with 19 reed switches actuated by a common rectangular coil, and one with six reed switches with individual coils.

Sixteen of the first type (12 reed) boards are used to select the disc, 64 of the second type (19 reed) boards are used to select the positioner, and four of the third type (six reed) boards, for the position. All selection signals from the logic unit for head group (disc) and position are D.C. currents to the reed coils above. These coils are arranged in three matrices: the disc select matrix (eight lines, one of 16 select); the position matrix (10 lines, one of 16 select); and the motor zone matrix (four lines, one of four select).

Decoding of the 10-bit disc and position address (16 discs, 64 positions) to these 22 lines is done in the logic unit. The apparently redundant lines of the position matrix are required by a peculiarity of the positioner. A signal is available which indicates when the reeds are closed and is used to indicate when the current may be applied to the positioner. Extreme care is taken to avoid closing the reeds with voltage applied. Conversely, the head positioner drive circuits sense current through the reeds and will not allow opening of the reeds when current is flowing. Power-on and power-off signals to the positioners are pulses from the logic unit. The high current switch (the positioner draws approximately 1.4 amps) is a silicon-controlled rectifier.

The balance of the electronic hardware in the disc unit is made up of write amplifiers, enabled and driven from the logic unit, erase amplifiers, and reading amplifiers for data, clock and sector marks. The interface signals from these amplifiers are one's and clock on separate lines. Head selection within the group of eight heads per disc is by enables from the logic unit. Reading amplifier recovery following a write operation on a head within the group is well within 100 microseconds.

### The control disc

The top baffle disc is prerecorded with four complete sets of control tracks. Clock, inner zone sector, outer zone sector and index tracks make up the set. Virtually any number of sector marks may be recorded, allowing wide variation of record lengths. Each set of control tracks is widely enough separated from its neighbor so that the three spare sets are completely safe in case of head failure. Control track heads and control track discs are completely interchangeable.

### Air supply and A.C. control

The air supply for the heads is a major subsystem. The disc unit is normally supplied with a connection for a plant air supply, if available, and a compressor for those sites where an air supply is not available. This air supply is filtered for particles down to 0.3 micron. Several series water traps are provided to prevent a condensate problem.

The disc unit may be operated independent of the logic unit on test mode. In normal operation a single switch at the logic unit initiates an automatic sequence of turn-on which terminates with an "operable" signal indicating that the heads are landed and normal operation may begin. Over temperature protection is provided at two temperature levels. The first indicates an over-temperature condition and allows normal operation until the second level is reached. At that point, all power is disconnected but is reconnected when the first level is again reached.

Automatic head retraction in case of power failure is provided for, and a power loss detector protects against uncontrolled writing by providing bias voltage to the writer for a sufficient period.

## THE LOGIC UNIT

### General

A description of the logic unit cannot be made without describing *how* a DISCfILE is used. Although we separately price and sell disc units without the logic unit, Data Products Corporation is pleased that all units to date have been delivered with logic units of our design and manufacture.

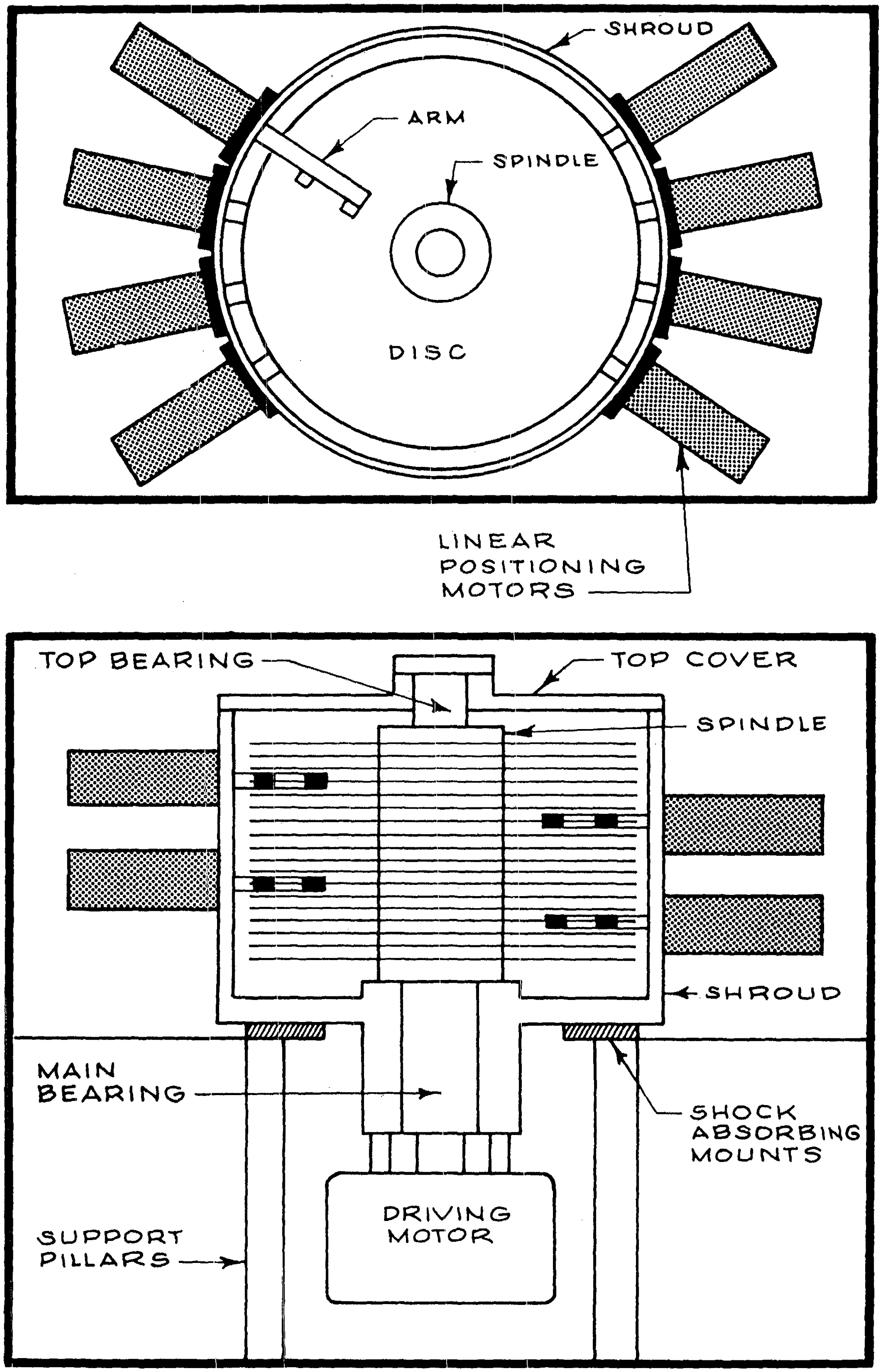

SHROUD
ARM
SPINDLE
DISC
LINEAR
POSITIONING
MOTORS
TOP BEARING
TOP COVER
SPINDLE
SHROUD
MAIN
BEARING
SHOCK
ABSORBING
MOUNTS
DRIVING
MOTOR
SUPPORT
PILLARS

**Figure 3**

The interface described for the disc unit is similar in most respects to any currently available random access memory, and to that extent may, and will in the future, be integrated into systems by our original equipment manufacturer customers. The experience we gained in supplying the complete subsystem has had a powerful influence in our thinking on future design of DISCfILES. We have, in addition, enjoyed extremely rapid system integration by our customers of a completely tested subsystem.

The logic unit as supplied to our largest customer is a representative solution of the system integration problem of any random access memory. The system requirements were:

1. Connection of up to four DISCfILE subsystems to a single customer-supplied controller with provision for simultaneous seeking of the four memories
2. Complete independence between power supplies of the DISCfILES and the customer supplied controller
3. Serial by bit transfer of data and addresses
4. Provision for independent off line operation
5. Provision for off line fixed format writing of the files
6. Complete error control and analysis within the DISCfILE subsystem.

The one logic unit described here was designed to meet the above ground rules.

### Logical organization

The basic logical elements in the logic units are:

1. A shift register to receive addresses (address staticizer)
2. A register to store addresses (address register)
3. A parallel comparator between the above registers
4. A counter whose length exceeds the prescribed record length
5. Decoding and drive circuits to decode the contents of the address register and select the disc, position, head, and address in the DISCfILE.

### Logical sequence

Following the "operable" signal the logic unit is logically connected to the controller by a "select" pulse along a line unique to that logic unit from the controller. The complete address is then transmitted on a common data bus by the controller and stored in the address staticizer. A "seek" pulse emitted by the controller logically disconnects the selected DISCfILE logic unit from the controller.

The logic unit checks the address for good parity, validity and completeness and transfers the contents of the address staticizer to the address register. Power is dropped on the last positioner used, the dry reed switches cleared, closed to the new address, and power applied to the selected positioner. Reading is enabled on the selected head and the data it reads is cycled through the address staticizer. Preceding each record is a prerecorded header containing the complete address. This header, appearing in the address staticizer, is compared with the contents of the address register. When a sufficient number of favorable compares are made to insure that the head is reliably on track, an "on track" signal is generated within the logic unit.

One bit of the address is used to condition the logic unit to send a "ready" signal to the controller at this time, or when the addressed record is approaching the head. The controller then reselects the file and commands the file to write or read the addressed record or read the record under the selected head. This last operation is used for timesaving when a search for a record-contained identifier is being made. In both cases, this technique of control allows overlapping seek in the four files on a controller, since the file is logically connected only during address and data transfer time. It also makes the fastest use of the positioner, as opposed to a constant timeout to allow the positioner to settle.

The write or read operation may be terminated on command of the controller any time during the record or may be continued for all the records on eight tracks by emitting a write or read command (without re-addressing) at the "end of record" time. The logic unit pads the remainder of a record with written "zeros" when the write is terminated within the record.

Any departure from normal timing of the seek time, latency time, and bit to bit timing in or out of the logic unit is detected and signalled to the controller.

### Record formatting and file capacity

As previously stated, any number of sector marks may be recorded into the control tracks. The resulting record length must be accommodated by the counter, and the number of addressable records must be accommodated by the registers. The storage capacity lost for each record is determined by the read-write to erase gap separation, which is lost twice per record, plus the number of address bits. This loss is increased by the requirement for complete disc and positioner replaceability, guaranteed in current DISCfILES, and complete reading amplifier recovery between records written and read back randomly or sequentially.

Most DISCfILES delivered to date in use are set up for 16 records in outer zone tracks and eight records in inner zone tracks. Total file capacity is 98,304 records of 1,365 bits each, or 134,184,960 bits per file.

## THE DESIGN CONCEPT AND ITS FUTURE

The obvious point of difference in design concept between the DISCfILE and other random access disc memories is the use of individual positioners for each disc. The obvious and ancillary advantages of the concept have been noted elsewhere in this paper: low power, proven reliability, quietness, etc., are all desirable but not sufficient reasons for the choice of this concept. The overwhelming reason for this choice of machine organization is the ease of overlapping seek within a DISCfILE for increased random transaction rate without:

1. The requirement for additional heads
2. The requirement of additional positioners precise enough to access tracks written under control of another positioner.

With independent positioners, we have achieved overlapping seek by additional switching and logic at a price for capacity and transaction rate which will be difficult to improve upon. The first DISCfILE embracing this technique, the dp/f-5025, also provides a capacity increase of approximately 50 percent by an increase in recording density to 600 bits per inch. This is the first step in a continuing program by Data Products Corporation to provide a complete DISCfILE product line for the data industries.

by Herman Hess
Informatics, Inc.
Culver City, California

# A Comparison of the Characteristics of Modern Discs and Tapes

ONE OF THE MOST INTERESTING DEVELOPMENTS in computer technology has been the coming of age of the magnetic disc storage systems. Until recently, practically all available magnetic discs were either offered as peripheral equipment only on computers with limited capability, or the disc unit was designed to be the major random access data storage of the computer. This situation has now changed so that disc units are now available for practically all computer systems. IBM offers the 1301 disc unit for all 7000 series computers as contrasted to offering their earlier disc units only for the IBM 305, 650, and 1401 systems. Other computer manufacturers are now offering disc units as standard peripherals for most of their systems.

This paper will be restricted to considering modern disc and tape units which have been publicly announced as available commercially as integrated parts of computer systems. In making comparisons, the term "character" is used to mean six bits of information plus the number of additional bits appropriate to the recording mode.

This availability of disc units has raised a series of inquiries about how best to employ these devices. This new generation of disc units has the following major features:

1. Random access to data in 17 to 350 milliseconds
2. On line storage capacities in units of hundreds of millions of bits
3. Low cost on line bulk storage
4. Inherently highly reliable
5. Small physical size per unit of storage.

These features present serious competition to other bulk storage devices such as large drums, magnetic cards and magnetic tapes. This analysis will be limited to comparing operating characteristics, costs and application techniques for disc units and magnetic tapes. Magnetic tape represents by far the greatest number of storage applications.

To illustrate the differences between magnetic tape and disc techniques a detailed analysis of computer sorting will be given later. This was selected because it is a computer application in which magnetic tapes play the dominant role, and in which tapes are displayed to great advantage. This application does not show current models of disc to good advantage. Thus, it represents a "worst case analysis" and also illustrates some of the complexities of disc usage.

## CHARACTERISTICS OF MAGNETIC DISC UNITS

The currently available disc units are the IBM 1301, Bryant 4000 Series and Data Products Corporation DISCfILE. In addition, Anelex have announced that they intend to manufacture disc units. Except for the IBM 1301 the characteristics of these units will vary considerably since each is offered to the user in different arrangements by different computer manufacturers. Hence for the purpose of this analysis a general description will be adequate.

In the past an important figure of merit for disc units has been the average access time. The average access time represents the average time it takes the unit to read or write information at a designated location. It generally includes the time of the system to perform the following procedures:

1. Decode the command and select the track and head required
2. Move the arm to the correct track
3. Confirm the track location and notify the central processor that the head is on the correct track to read or write
4. Wait until the desired record or track sector rotates into position to read or write. This last wait is called latency time.

Typical random access times are between 100 to 200 milliseconds. This includes average latency time, usually about 17 to 35 milliseconds. A related time of interest is the time to obtain access to the adjacent or a nearby track. This time is of the order of 75 milliseconds.

After access has been made to the specified track the data can be transferred at fixed rates. For transferring data serially by bit, the rates vary from 31,000 to 90,000 characters per second. The variation is caused by different bit densities, track position, and rotation rates. Transfer rates for parallel bit transfers as high as 960,000 characters per second have been announced as being available, but are not yet in use in any commercial or industrial installation.

The actual amount of data transferred before moving the heads is limited by the track capacity and the ability to switch reading or writing heads at electronic speeds. Using the cylinder concept, which implies head switching at electronic speeds, data transfers of 20,000 to almost 1,000,000 characters can be achieved. This number is useful as an indication of the amount of data available within one rotation time.

The effective transfer rate is also reduced by the requirements of formatting. If small record sizes are used the rates descend very rapidly. This can be seen clearly from **Table 1.**

The foregoing is typical for all other discs. The maximum effective transfer rate is usually less than the nominal rate and occurs when the record size and track size coincide.

An increasingly important figure of merit is the total storage capacity. Disc units now come almost in all sizes, with modules having as few as six operating discs to units with 24 operating discs.

TABLE 1

EFFECTIVE TRANSFER RATES OF A TYPICAL DISC (IBM 1301)

| TRANSFER RATE (Characters/sec.) | |
|---|---|
| Nominal | 90,100 |
| Small Records | |
| 120 characters | 68,400 |
| Small Blocks | |
| 720 characters | 85,600 |
| 256 words (of 36 bits) | 87,900 |
| Disc Track | |
| 2796 characters | 88,900 |
| Large Block (2+ tracks) | |
| 1024 words (of 36 bits) | 87,500 |

TABLE 2

PERCENT OF THEORETICAL CAPACITY OF A TYPICAL DISC (IBM 1301)

| PERCENT OF NOMINAL CAPACITY | |
|---|---|
| Small Records | |
| 120 characters | 75.0% |
| Small Blocks | |
| 720 characters | 94.1% |
| 256 words (of 36 bits) | 96.3% |
| Disc Track | |
| 2796 characters | 97.5% |

Within one package as many as 50 individual discs have been employed. In addition, disc diameters vary from 24 to 39 inches. Hence, a single disc surface can range in capacity from $5 \times 10^6$ to $15 \times 10^6$ bits per disc. For disc modules in the range of 16 to 24 discs capacities range from $22 \times 10^6$ to $103 \times 10^6$ characters.

This mass of data are generally organized in some format. For some units the format is variable in that each installation can format the module to meet its own requirements. For other units the computer manufacturer has selected a fixed record size and organization. Depending upon the format scheme employed, effective capacity can be reduced as much as 25 percent. **Table 2** shows the effective capacity for selected record sizes. If a poor choice of format is made, the effective capacity can be reduced very drastically.

Disc unit costs are relatively uniform for the units for which prices are publicly known. The cost of a module to the user is about $2,000 to $2,500 a month rental. Also required is a controller that has averaged about $1,000 a month. A very large increase in cost occurs if the computer requires a separate channel for the disc unit. This requirement for a separate data channel is not universal and varies among manufacturers.

The reliability status of disc units has suffered from a lack of detailed information on performance. Inherently a disc unit should be almost as reliable as a drum unit and far superior to magnetic tapes. The only disadvantage as compared to drums is the requirement to have moving arms. As compared to tapes the disc units are:

1. Closed systems with no human interface, no exposure to unfiltered air, and with magnetic surfaces which do not change.

2. Relatively stable physical systems which have few severe operating shocks and no contact wear on the magnetic surfaces.

One aspect of disc unit reliability to be considered is the realization that the disc unit is normally always on line. This requires that considerable thought be expended on file security. The most attractive solution to date has been the development of the disc pack units. These units with removable packs provide a neat solution to file security problems for relatively small sized files.

## CHARACTERISTICS OF MAGNETIC TAPE HANDLERS

By now the characteristics of magnetic tape handlers are relatively well known and will only be reviewed briefly.

The figure of merit for magnetic tape is the nominal transfer rate. Currently transfer rates with a range of 7,000 to 240,000 characters per second are available as standard equipment. Higher rates have been announced but are not known to be in commercial operation.

However, in evaluating magnetic tape rates the nominal transfer rates are seriously reduced by either interblock or interrecord gaps. This effect is quite pronounced at the new higher density rates as can be seen from **Table 3.**

The effective capacity of a reel of tape is also reduced by blocking requirements. A one-inch wide 3,600 foot reel of tape with a density of 2,000 characters per inch has a theoretical capacity of about $8x10^7$ characters. **Table 4** shows the effect of blocking on theoretical capacities. Most tape systems have reels and densities that are effectively packed at about four to 10 million characters per reel.

In general most tape handlers can read data in either forward or backward mode with the important exception of IBM units, which can read only in a forward direction.

Tape rewind speeds vary among manufacturers and the size of the tape reel. Rewinds of full 2,400 foot reels can be as short as one minute and as long as four to six minutes for 3,600 foot reels.

The costs of magnetic tape handlers are relatively stable. A typical handler with modest capability will rent for about $400 to $800 a month. A high performance handler will cost about $800 to $1,200. The latest very fast systems have prices slightly about $1,200. Associated with each tape handlers are controllers and channel arrangements. It is difficult to make general statements about these units because of the different capabilities and implication to the whole system. In general, for fully buffered operation and control,

TABLE 3

EFFECTIVE TRANSFER RATES OF MAGNETIC TAPES AT TYPICAL COMBINATIONS OF TAPE SPEED AND FRAME DENSITY FOR VARIOUS SIZE RECORDS

| Magnetic Tape | | | | |
|---|---|---|---|---|
| Frames per inch | 200 | 556 | 800 | 1,000 |
| Inches per second | 75 | 112.5 | 112.5 | 100 |
| Nominal Transfer Rate (characters per second) | 15,000 | 62,500 | 90,000 | 133,000* |
| Small Record | Effective transfer rates (characters per second) | | | |
| 120 characters | 6,600 | 12,500 | 15,300 | 14,700 |
| Small Blocks | | | | |
| 720 characters | 12,500 | 38,600 | 49,500 | 58,600 |
| 256 words (of 36 bits) | 13,600 | 49,300 | 64,800 | 82,500 |
| Disc Track | | | | |
| 2796 characters | 14,100 | 54,200 | 72,000 | 100,000 |
| Large Blocks | | | | |
| 1024 words (of 36 bits) | 14,500 | 58,500 | 81,800 | 116,000 |
| 4096 words (of 36 bits) | 14,800 | 61,000 | 89,000 | 128,000 |

* Nine information channel tape (six information bits per character)

TABLE 4

PERCENT OF THEORETICAL CAPACITY OF MAGNETIC TAPE AT TYPICAL BIT DENSITIES FOR VARIOUS SIZE RECORDS

| Magnetic Tape Density | 200 | 556 | 800 | 1000 |
|---|---|---|---|---|
| Small Record | | | | |
| 120 characters | 44% | 20% | 17% | 11% |
| Small Blocks | | | | |
| 720 characters | 83% | 62% | 55% | 44% |
| 256 words (of 36 bits) | 91% | 79% | 72% | 62% |
| Disc Track | | | | |
| 2796 characters | 94% | 87% | 80% | 75% |
| Large Blocks | | | | |
| 1024 words (of 36 bits) | 97% | 94% | 91% | 87% |
| 4096 words (of 36 bits) | 99% | 98% | 98% | 96% |

and where the maximum number of tape handlers is employed, the tape controllers average about 10 to 20 percent of the cost of the total number of tape handlers.

The reliability of tape handlers has improved, however, they are still considered one of the weakest links in peripheral computer equipment. One of the most difficult and vexing problems has been the requirement for the changing of tape reels. The problems of getting a tape reel changed, correctly labeled and stored, still remains a great source of annoyance and expense. The mechanical operation of tape units has remained relatively unchanged. The major reliability improvement in tape handlers has been the read after write check which gives assurance that at least the tape was readable at the time of writing. The announced tape units with sharply increased performance have aroused a certain amount of skepticism.

## COMPARISON OF TAPES AND DISCS

The major interest in comparing magnetic tapes and discs is their employment for on line scratch pad operations and for bulk storage.

**Figures 1 and 2** show the cost relationship between tapes and disc for units offered by IBM and General Electric. **Figure 1** indicates that, with IBM equipment, the cost of on line storage, per unit of capacity, is approximately equal for tapes and disc. The horizontal parts of the tape curves represent the requirement to use an additional tape controller to increase storage capacity by tape units. **Figure 2** indicates that General Electric tape units have lower costs than their disc unit. The average difference is about $600 a month rental, which represents about four to six percent of a typical General Electric 225 system. However, **Figure 2** also shows that if the T225F unit (with reel sizes comparable with IBM) was used, the cost relationship would be slightly in favor of disc units.

**Figures 3 and 4** show the effective transfer rates of disc units as compared to tapes. The results are a direct outcome of the properties of the two media. **Tables 1 and 3** clearly indicate that while both tapes and disc transfer rates were affected by blocking requirements, the effect on disc units is small, while for tapes it is large. Three-quarters of an inch at high density is a lot of empty space. The block size selected was chosen because it is a very commonly used unit in each case.

All curves have assumed that units would be placed in the most favorable positions at the start of the program. No start/stop time between blocks was used in calculating tape effective rates because it was assumed that the tape can be kept in motion for this type of reading or writing. The shape of the GE M640A curve represents the changes in transfer rate for the inner zone and the requirement to transfer to another set of tracks.

In summary, the differences between tapes and disc units are as follows:

1. The cost per unit of capacity of on line storage is approximately equal. The use of

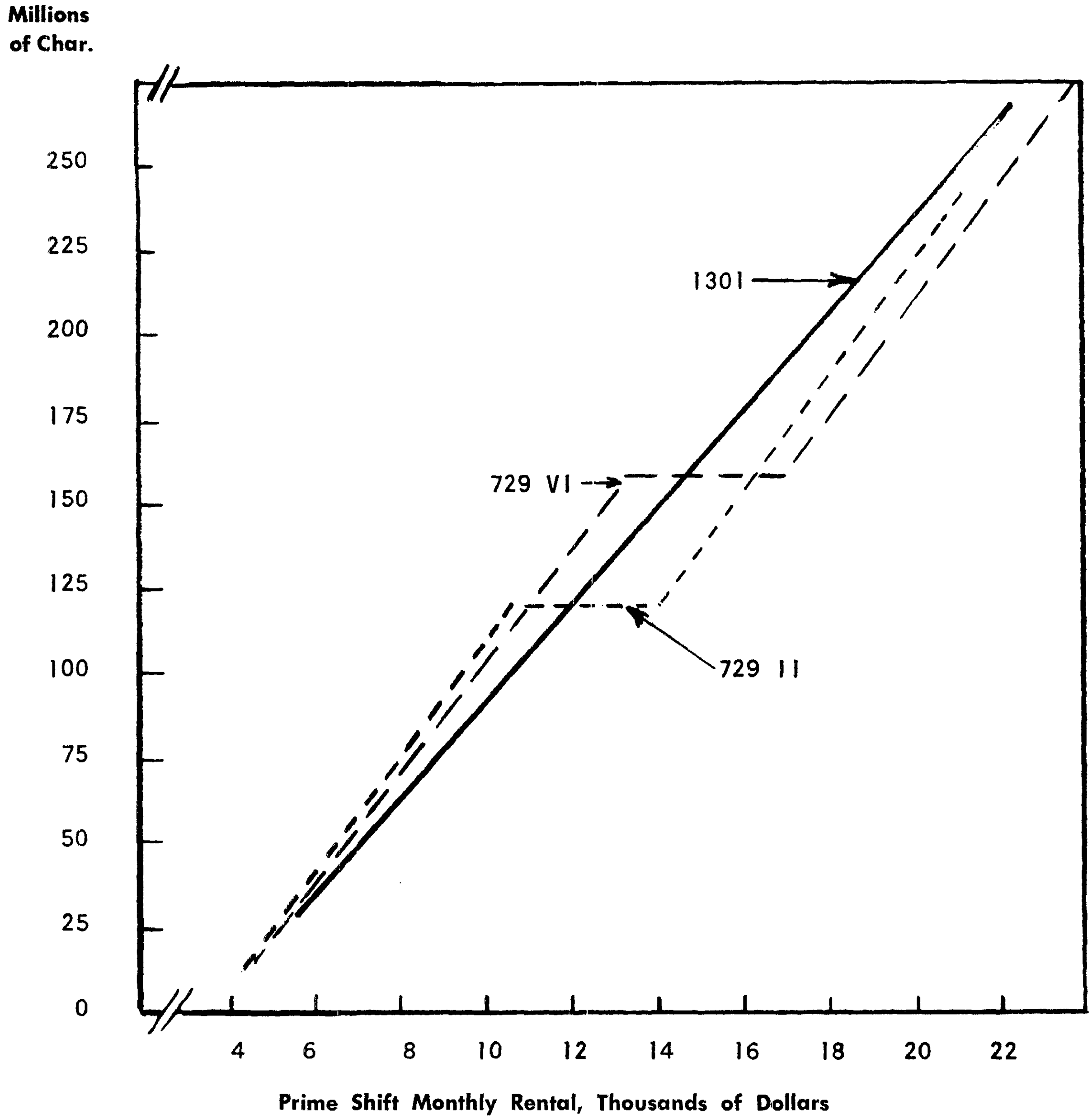



**Figure 1. EFFECTIVE ON LINE STORAGE COSTS FOR IBM 1301 DISC AS COMPARED TO IBM TAPE UNITS 729 II AND VI**

one-inch wide and 3,600 foot reels will give a cost advantage to tape.

2. The requirements of format effect both types of units. This loss in storage density for disc units is considerably smaller than for magnetic tape.
3. The effective transfer rates of disc units are always greater than tapes.
4. At the present time information in disc units cannot receive absolute file protection such as is available on magnetic tapes.
5. All information in disc units is accessible in about 250 milliseconds. For tapes, random access is of the order of a few minutes.

These characteristics indicate that disc unit costs are in the neighborhood of tape costs and offer, as a bonus, random access combined with faster effective transfer rates. The only restriction for disc units is that the information is on line and a penalty must be paid to remove and save the information.

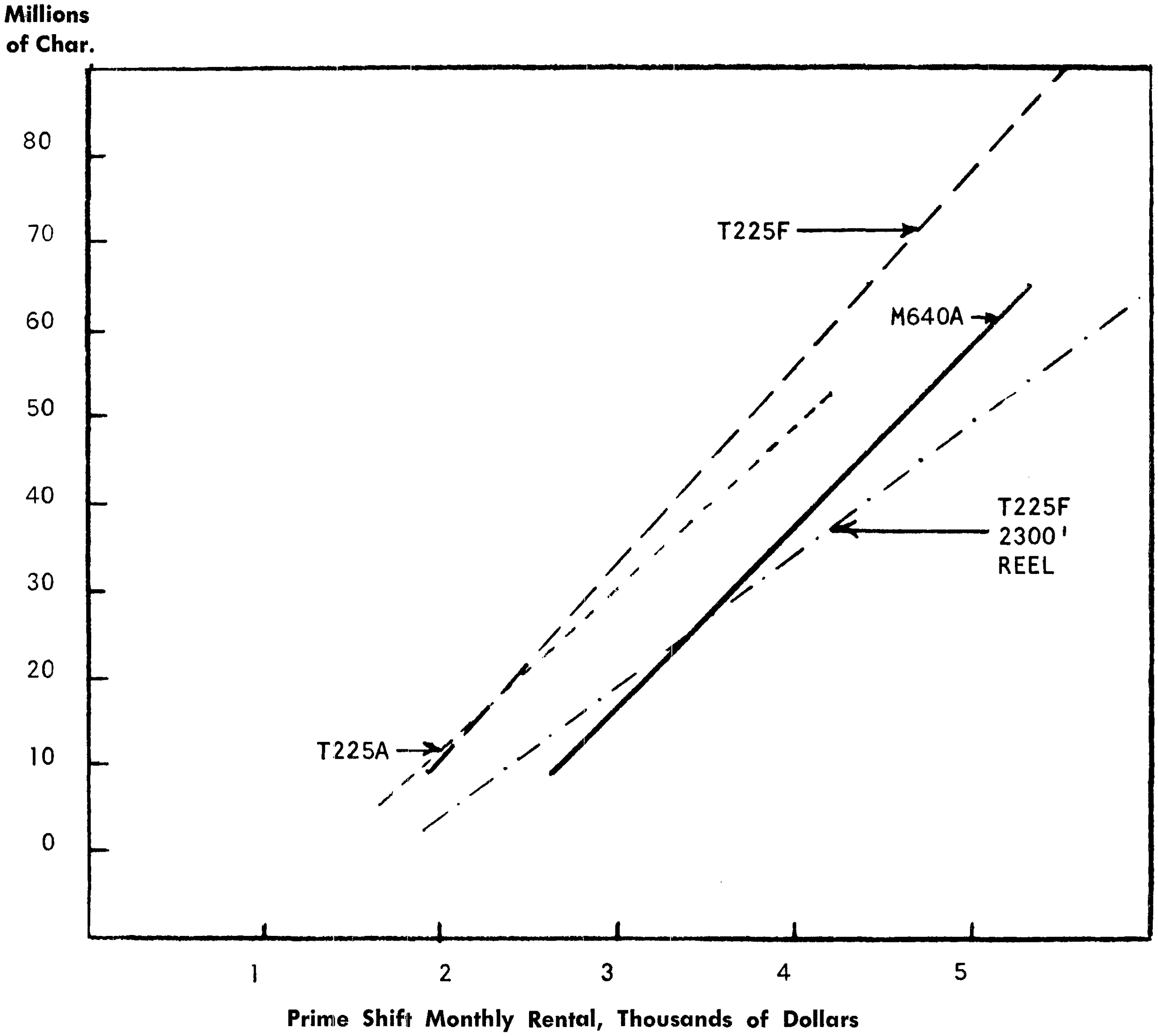



**Figure 2. EFFECTIVE ON LINE STORAGE COSTS FOR GE 225 M640A DISC AS COMPARED TO GE TAPE UNITS T225A, T225F AND T225F WITH 2300′ REEL**

## APPLICATIONS

The obvious applications for disc units have been for problems requiring immediate or short service times. Typical applications are for on line query systems and in real time applications, such as airline reservations, message centers, etc. The basic parameters for these applications are:

1. The information storage requirements are too large to store on magnetic drums or on the first few hundred feet of tape.
2. The average service time requirement is of the order of less than a few seconds, but greater than 100 milliseconds.

While the average access time for disc units is about 250 milliseconds, it is possible by system design using appropriate programming techniques and file storage arrangements, to reduce this to under 100 milliseconds.

The less obvious applications are those that make use of the fact that the cost of disc systems is competitive with tape. One application of this type is the use of a disc unit as the major auxiliary or bulk storage for computer supervisory systems.

A related application that has received some attention is the use of disc units in the implementing of compilers and assemblers. The random

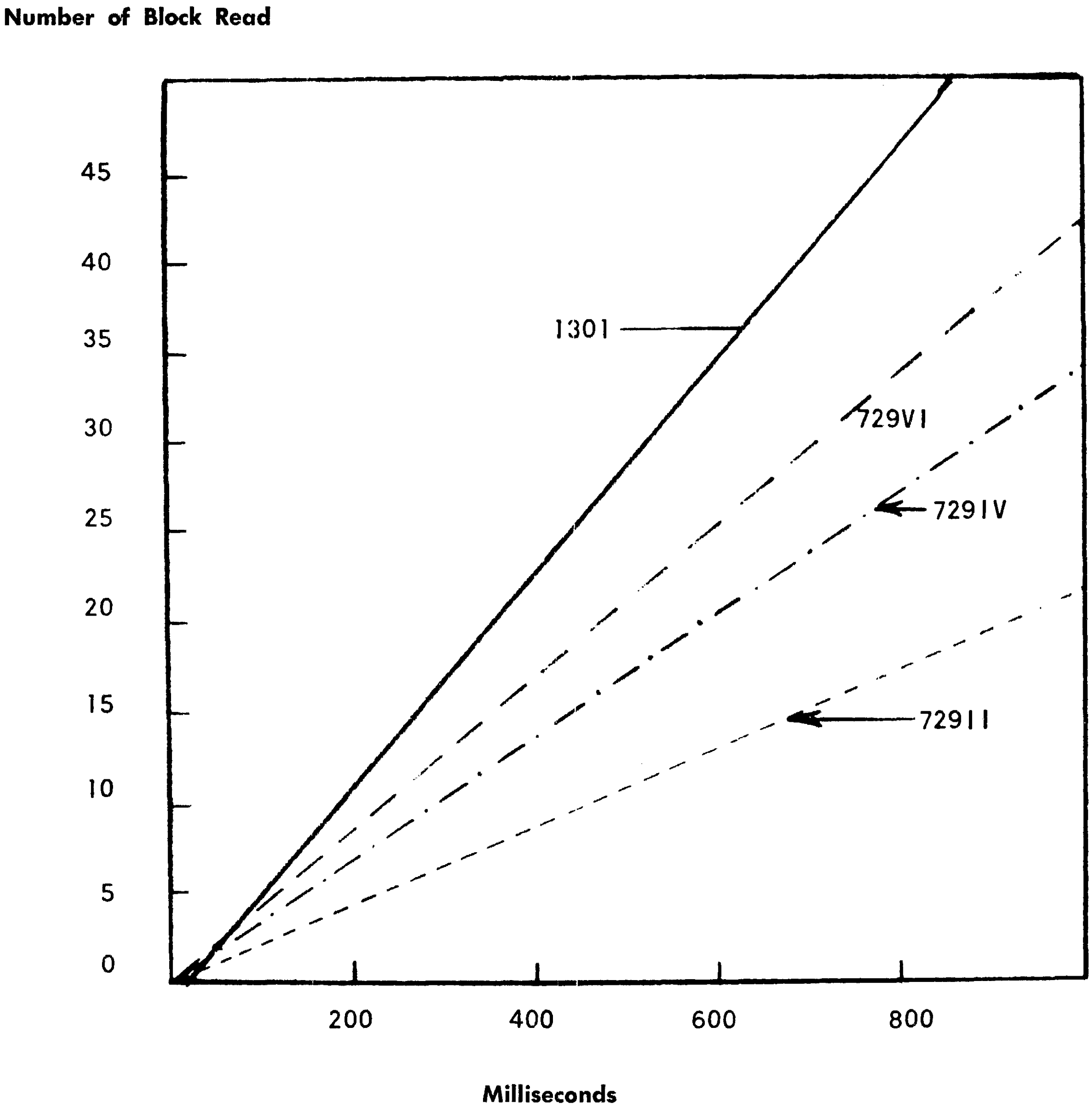



**Figure 3. EFFECTIVE TRANSFER RATE OF IBM 1301 DISC AND IBM MAGNETIC TAPE UNITS FOR READING 256 IBM 7090 WORD BLOCKS**

access feature combined with rapid transfer of large blocks makes the use of a disc seem desirable. The particular compiler operations that can use a disc unit are:

1. Storing the various lists that are generated in the compiling process. This is particularly important in small machines where the program has been forced to use tape because of space requirements.
2. Storage of subroutines which permits the direct inclusion of the subroutines when required.
3. Implementation of efficient overlay procedures to provide the effect of enlarged primary storage.

For all the applications in which a disc is used as a tape substitute, a certain amount of caution must be exercised. For existing programs (and even in new applications) there is a temptation to use the same kind of program logical arrangement for discs as for tapes. This simple procedure has been proposed for several systems as a quick method of getting into operation with disc units. The results are bound to be disappointing in most

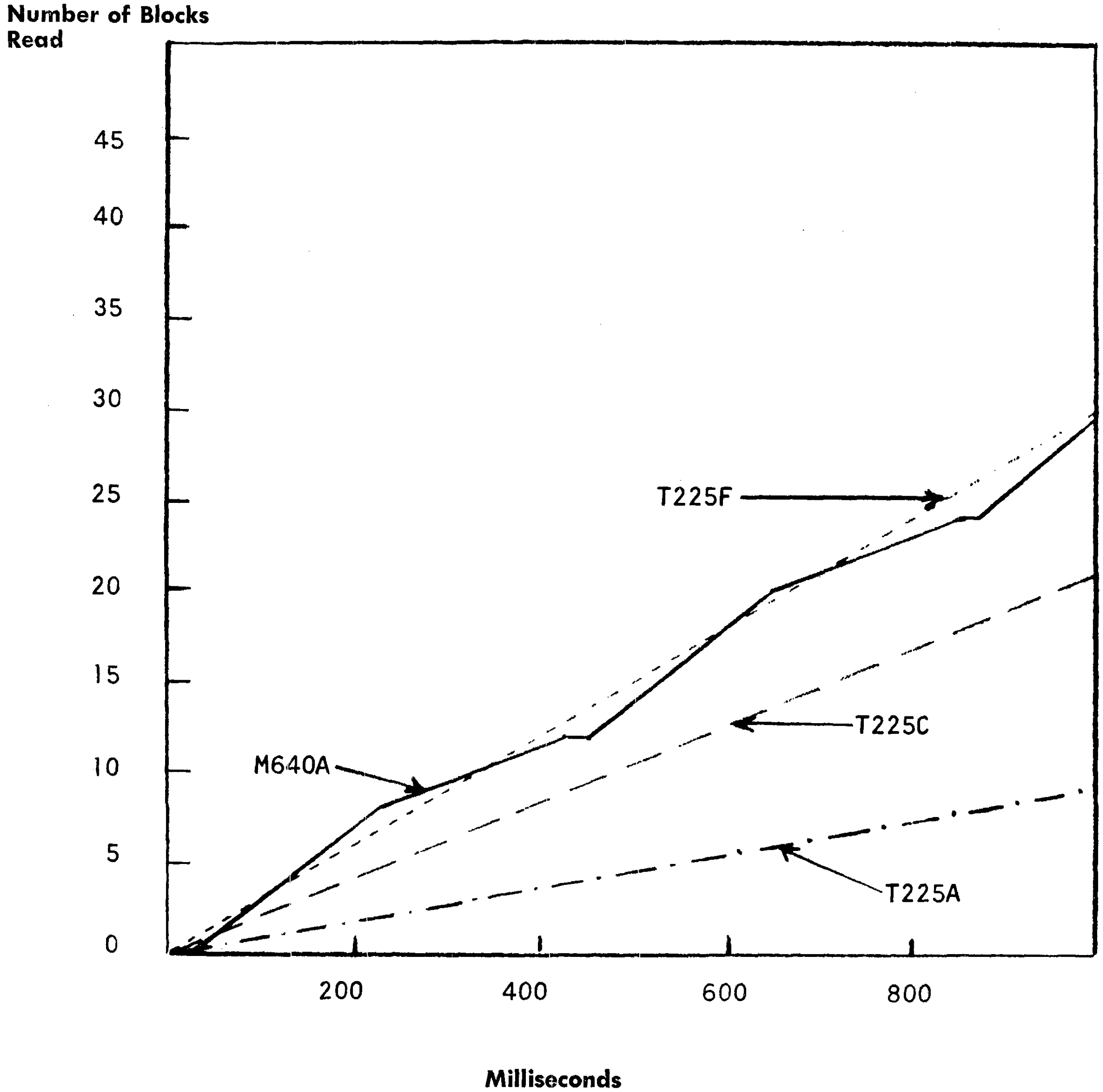



**Figure 4. EFFECTIVE TRANSFER RATE OF GE 225 M640A AND GE MAGNETIC TAPE UNITS FOR READING 1536 CHARACTER BLOCKS**

cases because good programmers take tape timing into consideration when making tape requests. However, they have not yet been required to consider average random references and timing considerations for disc units.

To illustrate some of the pitfalls of disc utilization, the following example using computer sorting was prepared. The use of computers for sorting takes a significant portion of computing time in most business applications. Sorting is required when the user insists that he wants a hard copy of every item that entered a process in an order convenient to himself. Hence, despite the availability of large random access units, a certain amount of sorting will still be required.

*Sort Example.* A file consisting of 60,000 records of 16 words each is recorded on tape in blocks of 256 words and is to be arranged in ascending sequence by an 18-character key. The sorted file is to be recorded on tape in 256-word blocks.

*Equipment.* This example will utilize an IBM 7090 having a two module IBM 1301 disc unit on its own channel and four IBM VI tape units on another channel. For the tape sort the disc modules will be preceded by an additional magnetic tape channel with four additional 729 VI units. The cost of the alternate auxiliary equipment is approximately equal.

*Procedures.* To standardize procedures the IBM 7090 Data Processing System Bulletin J28-6043-1 titled *IBM 7090 Generalized Sorting Program Sort 709 Sorting Times for the IBM 7090* was used to obtain certain critical values and to time the tape sort. The Bulletin did not provide data for the newer 729 VI units. The 729 IV units' read/write times were reduced by a generous factor of 556/800 to represent 729 VI units.

*Cases*

Method 1 – The disc configuration, used as a random access device

Method 2 – The disc configuration used to simulate 20 tapes

Method 3 – The eight-tape configuration (same price as in the disc configuration).

*Results.* **Table 5** summarizes the results of the three methods employed. Method 1 is a simple approach of using the disc as a random access device. Method 2 uses the disc as though it consisted of 20 tapes. Its sort times are slightly shorter than Method 3, which is used for comparison. Method 3 is handicapped by the lack of ability to read and write at the same time and the need to move the access arms to random cylinders. The use of multiple access arms would overcome this difficulty.

These results indicate that while the disc units have some advantages over tape in storage and transfer rate, they require considerable care in applications to avoid results like Method 1. The results all indicate the need for independent multiple access.

## DETAIL DESCRIPTION OF METHODS

(calculations are shown in appendix)

*Method 1* is an unsophisticated attempt to use the random access properties of the disc unit. In Phase I the records are read into the computer. For each record a disc location address is assigned and the 18-character key is assembled to form a four-word record. The four-word record and the input record are transferred in output blocks of optimum size to each module of the disc. The key records require 14 cylinders; the input records require 54 cylinders.

Phase II commences when all the records have been entered into the computer. The key records are returned to the computer and sorted into ascending sequence. Since the records are in adjacent cylinders and the computer core is large compared to key record size, the sorting can be done quickly in two passes. The four-word key record size is reduced to one word on the last write to the disc.

Phase III commences when the key records have been sorted. The sorted key records are returned to the computer to obtain the location address of the input record. Since the records were recorded in adjacent tracks the average access time to obtain each record is very low for average access. However, 90 milliseconds when multiplied by 60,000 is a large time interval.

This method illustrates that, despite optimum recording, the requirement to read random small records from a disc is costly in time.

It should be noted that if both modules were used to store the input records the average time for arm movement could be reduced. However, even if the movement time were reduced to zero, the average latency and record reading time

TABLE 5

TIME REQUIRED TO SORT 60,000 16-WORD RECORDS USING THE IBM 1301

(Time in Minutes)

| Method | Phase I | Phase II | Phase III | Total | Ratio of Time to Tape Sort |
|---|---|---|---|---|---|
| 1 | 1.78 | 1.20 | 90.10 | 93.08 | 6.30 |
| 2 | 1.78 | 9.34 | 1.94 | 13.06 | .88 |
| 3 | 1.78 | 10.04 | 2.96 | 14.78 | 1.00 |

would be 36 milliseconds, which would still require 36 minutes for 60,000 records.

Method 2 is the substitution of the disc for tapes. The appendix shows the great care that must be used in organization of the disc to achieve the favorable results shown. It also indicates the amount of planning required to optimize the use of only one set of access arms. The use of several arms would make the organization of the disc simpler and improve the performance.

Method 3. The tape sort was included to provide comparisons of sorting times. The IBM Bulletin indicated it was a typical tape merge sort program.

## APPENDIX

Data:

| | | |
|---|---|---|
| Number of records | 60,000 | |
| Size of records | 16 | words |
| Input block size | 256 | words |
| Output block size | 256 | words |
| Number of blocks (tape) | 3,750 | |
| Number of blocks in cylinder | 70 | |
| Number of records in each track | 29 | |
| Disc block | 87 | records |
| Number of disc blocks | 690 | |
| Recommended input block | 412 | |
| Merge block | 206 | |
| Tapes available for tape sort | 8 | |
| Merge order | 3 | |

### METHOD 1 – THE DISC USED AS A RANDOM ACCESS DEVICE

Phase I:

Input time (see tape sort): 1.78 minutes

Output time

1. Input records written in 54 cylinders in 10 track steps: 80 seconds
2. Key record written in 14 cylinders in 10 track steps: 21 seconds

Time not overlapped: 97 seconds

*Time for Phase I* 1.78 minutes

Phase II:

Sort 14 cylinders of key records organized as 28 strings of 2,300 records.

Pass 1. Input of 14 cylinders

Read in eight track steps: 23.7 seconds

Output of 14 cylinders

Written in eight track steps: 21.2 seconds

Pass 2. Input of 14 cylinders

Written in eight track steps: 21.2 seconds

Output of 3.5 cylinders

Written in eight track steps: 3.7 seconds

*Time for Phase II* 1.20 minutes

Phase III:

Read sorted keys into computer. Select records in indicated sequence.

Input of key records app. 5.7 seconds

Input of records

| | |
|---|---|
| a. Movement time 50/54 x 50 + 4/54 x 120 — 1 = | 54 milliseconds/record |
| b. Average latency | 17 milliseconds/record |
| c. Record read time | 19 milliseconds/record |
| Total | 90 milliseconds/record |

*Time for Phase III* 90.10 minutes

*Total time for Method 1* 93.08 minutes

### METHOD 2

In Method 2 judicious selection of disc organization can result in sorting times that are slightly faster than tape sorts. The following assignment of disc cylinders is an illustration.

Phase I:

*Organization.* In each module we use 30 of the 40 available surfaces. We define a "logical group" as three surfaces. Within each logical group we use 69 of the 250 available basic cylinders. We thus have a logical set of 10 x 69 = 690 "short cylinders" which we call elements. (Thus each basic cylinder slices through the 10 logical blocks, holding one element out of each group.) These 690 elements will each hold a "disc block" of 87 records (87 x 690 = 60,030 records).

The records are read into the computer and sorted into a string of 87 records. They are written out on the disc as one logical element. The next string is written in the second logic group as the first element which is also in the same physical cylinder. The eleventh string is written logically behind the first string but at the top of the second physical cylinder.

The effect of this organization is that the access arms are used only 69 times under the assumption that each string is a sequence step down.

Timing:

| | |
|---|---|
| Input same as tape sort: | 1.78 minutes |
| Arm movements | |
| 1 x 180 = | 180 milliseconds |
| 7 x 120 = | 840 milliseconds |
| 61 x 50 = | 3,050 milliseconds |
| Latency | |
| 17 x 690 = | 11,730 milliseconds |
| Write | |
| 3 x 34 x 690 = | 70,380 milliseconds |
| Total output | 86.170 milliseconds |

Note: Computer is tape limited.

*Time for Phase I* 1.78 minutes

Phase II:

*Organization.* The input which consists of 10 logical groups will be merged and written onto the other module of the disc in the same organization. A merge of order 10 requires exactly two passes in Phase II. The organization of the groups has been made so that a high probability exists that when each group is randomly selected the respective elements will lie in the same physical cylinder, or at most, a short distance apart. Hence the allowance of the arm movement time on every read will probably be an overestimate. In writing the records the sequences will tend to stay within groups requiring the arm to move to adjacent cylinders for every element. Hence the output arm movement will exceed the input arm movement and become the limiting factor. For the IBM 1301, arm movement can be overlapped, but read and write operations cannot.

Pass timing:

| | |
|---|---|
| Input | |
| Latency | |
| 17 x 690 = | 11,730 milliseconds |
| Read | |
| 3 x 34 x 690 = | 70,380 milliseconds |
| Output | |
| Arm movement | |
| 20 x 180 = | 3,600 milliseconds |
| 140 x 120 = | 16,800 milliseconds |
| 530 x 50 = | 26,500 milliseconds |
| Latency | |
| 17 x 690 = | 11,730 milliseconds |
| Write | |
| 3 x 34 x 690 = | 70,380 milliseconds |

Computer time:

To maximize the size of records, buffer areas were not used.

Estimated computer time:

| | |
|---|---|
| 2 x 50 x 690 | 69,000 milliseconds |
| Total for pass | 280,100 milliseconds |
| Total for 2 pass | 560,200 milliseconds |
| *Time for Phase II* | 9.34 minutes |

Phase III:

*Organization.* The final merge will have the same properties as the input part of the Phase II merges. However, they will not be output limited by the disc and the program will use buffer areas.

Timing:

| | |
|---|---|
| Arm movement | |
| 50 x 690 = | 34,500 milliseconds |
| Latency | |
| 17 x 690 = | 11,730 milliseconds |
| Read | |
| 3 x 34 x 690 = | 70,380 milliseconds |
| Total for input | 116,610 milliseconds |
| *Time for Phase III* | 1.94 minutes |

*Total time for Method 2* 13.06 minutes

## METHOD 3

## TAPE SORT WITH EIGHT TAPES

729 VI units estimated to be 554/800 of 729 IV units

Phase I: $1.31 + 7.8 \frac{(412-16)}{412x16} =$ 1.78 minutes

Phase II: Sequences 60000/412 =146
Passes in Phase II = 4
4 x 3131 = 5.24 minutes

Phase III: $1.31 + 7.8 \frac{(206-16)}{206x16} =$ 1.76 minutes

Rewinds allocated Phase II, 4 x 1.2 = 4.8 minutes
Rewinds allocated Phase III, 1 x 1.2 = 1.2 minutes

*Total time for Method 3* 14.78 minutes

by Clint DeGabrielle
General Electric Company
Atlanta, Georgia

# Design Criteria for an On Line Nationwide Order Processing System

An integrated communication and data processing system has been designed by General Electric engineers to serve the needs of a geographically dispersed manufacturing and distribution business.

We undertook a study, and a system evolved which ties together 65 customer service centers — formerly called sales offices — at least one in each of the 49 continental states; 40 product departments with 53 manufacturing plants in 21 states; and 18 distribution warehouses in 11 states. Each of these 136 terminal points is served by teletype circuits. At six locations throughout the country concentrator stations are installed which will take several slow speed teletype circuits and combine their traffic into a high speed voice circuit operating at 2,000 bits per second.

## CONCENTRATOR STATIONS

The concentrator stations accept multiple teletype circuits as input. These circuits may vary in speed and code as well as the type of terminal device being used. Each input is code converted to eight-level code and is accumulated character by character in the core memory of the concentrator station. When end of message is reached, the entire message is released from memory and forwarded at 2,000 bits per second to the computer center. Thus we provide an 18 to 1 increase in speed of 110 baud traffic, and a 26 to 1 increase in 75 baud traffic.

In the opposite direction the concentrator station receives messages in eight-level code from the computer center at 2,000 bits per second and stores each message in core memory. It then releases the message to the proper outgoing line, doing code and speed conversion in the process.

In the event that the teletype lines or the voice circuit are inoperative or busy, storage is provided at the concentrator station to accept a full day's traffic. Although applied here as concentrator stations, the communication processors can be used for teletype switching, data switching, remote peripheral control stations, or as computer interface units.

The six voice circuits from concentrator stations, as well as several local teletype circuits, are tied through a computer communications interface unit to a computer center. The computer center consists of a pair of GE 225 computers with mass random access disc files and a complement of peripheral equipment. All 136 remote terminals are on line to the computer center and the entire system is under computer control.

## STORAGE MEDIA

In the preliminary systems study work it became evident quite early that in order for a system of this nature to function reasonably, it must provide on line access to large volumes of data.

Various available storage media were investigated and even some future devices were considered to provide this function. The designers decided that disc storage provided the best solution to this functional requirement at the present time.

Six General Electric mass random access data storage files, model M640A, became the heart of the system. Each file provides 16 data discs of memory, and information is recorded on both sides of all discs. There is an individual positioning arm for each data disc which provides eight read/record heads, four on each face of the disc.

These 128 read/record heads and 16 positioning arms allow access to data at an average of 225 milliseconds per record. Discs rotate at 1,200 revolutions per minute and provide an average latency delay of 26 milliseconds. All eight heads on a given positioning arm are switched in 100 microseconds.

Information is recorded on each side of a disc in serial mode on 256 tracks. Sixteen records of 64 three-character words are recorded on the outer 128 tracks and eight 64-word records are stored on the inner 128 tracks. Actually, the words are recorded as images of how the word appears in computer memory. Both binary and alphanumeric configurations are retained on the disc without change. With one command from one to 64 records can be read into memory or written from memory. Information is transferred to and from inner tracks at 37,500 characters per second and from outer tracks at 75,000 characters per second.

Each 16-disc file provides storage of 18.8 million alphanumeric characters or 34.4 million binary coded digits. A maximum of four 16-disc files may be connected to each disc file controller. With multiple file units attached to a controller records may be accessed concurrently in all files, with data transfer being accomplished serially through each controller.

Multiple checking features are built into the M640A system logic. Each word is parity checked both on transfer to and transfer from central processor memory. Each word on the disc contains parity and in addition each record contains a record check word. In the event that word parity error occurs on reading of a record program, reconstruction of the record is made possible through use of the check word. Read after write checking is employed to check the parity of data on a disc after a record has been written. The priority interrupt features of the computer allow data transfer to occur from the disc files concurrently with operation of other peripherals in the computer system.

## THREE FILES

The information contained in the six mass random access data storage files is in reality three separate files: a customer information file, a product information file, and an unfilled order file.

The customer information file contains an individual customer number for each of over 30,000 active customers — the customer name, address, limit to which credit will be extended, all shipments billed but for which payment has not been received, and all material on order which has not been shipped.

The product information file contains in excess of 300,000 catalog numbers for 37,000 products; the unit price for each catalog item; the name of the department producing the item; the product line within the department which manufactures the item; as well as a complete stock record consisting of: warehouse location of all stock, quantity on hand at each location, quantity in transit to each location, the quantity back ordered, and the reorder point for each item.

The unfilled order information file contains a record of: service center requisition number, customer number, name and address, item on order, scheduled delivery and source for the item.

With this information as background, let's follow the path of various types of operations in the system.

## ORDER ENTRY

First, we will consider the entry of a customer's order into the system. At one of the 65 customer service centers, formerly called sales offices, a customer order is received by phone, by mail, or by other means. A clerk at the service center enters the order into the system by typing the information on a terminal device. The unit transmits at operator speeds up to 10 characters per second. The signals generated travel over the teletype circuits to the nearest concentrator station, or directly to the communications interface, and there are turned back on the other half of a full duplex circuit and cause the page printer to be actuated. This we call echoplexing. The operator is charged with the responsibility of checking all transmissions by reading the copy to be sure it is correct. If she detects a mistake, she corrects it in an erratum box at the bottom of the order form. If none exist, she signals O.K. and the order is then released to the computer to process. In order to assure as great a degree of accuracy as possible, we have upgraded these operators two levels above their previous grade and made 99.95 percent error free transmissions a condition of keeping the assignment. Once the operator has made all corrections or O.K.'d the order, it is released to the computer and the computer first checks the erratum box for corrections, and then performs the following operations:

First, it checks the customer number against the customer file to see if this present order added to all unpaid bills, all shipments in transit to that customer, and all unshipped orders, fall within the credit limits authorized for the customer. If not, a message is typed out at the credits and collections desk and a manual supervisory routine is started.

We feel it is important to note that in this system it is possible to check a customer's total indebtedness and not just the unpaid invoices. When credit has cleared, the computer selects the warehouse nearest to the delivery point and prints out a set of shipping papers on the dock of that warehouse. From the time the operator at the service center releases the order until the start of printout at the warehouse, the system is designed to give a maximum of 15-second service. The computer then initiates a bill at the service center for the customer and proceeds to update all affected files. Incidentally, if the customer has not specified a means of shipment or has indicated best way, the computer chooses the best way and prints this out on the shipping papers at the warehouse.

**Computer stock selection**

If the computer finds the warehouse nearest the customer's site for delivery out of stock, it then checks in succession the next nearest warehouse until it finds a stock, and then issues shipping papers at the alternate warehouse.

As products come off the assembly lines at manufacturing plants, the quantity and type of product are fed to the computer center. The computer looks at the warehouse stock status file, the unfilled orders file, and the marketing forecast and order trend files, and develops a shipping allocation of products to warehouses. In the case of back orders, it types out shipping papers at the appropriate warehouse with a notation that stock to fill these orders is in transit and will arrive on a given date.

During the normal process of servicing customer orders, the computer continually checks the product record file, the unfilled order file, the customer order rate and marketing forecast to determine when production orders should be released on a factory to satisfy projected demands. This is accomplished by analyzing the production schedules, production cycles and capacity of the plant which produces the product under consideration, and issuing production orders that provide for optimum use of the facility.

When the computer determines that an order calls for an item that is not carried in warehouse stock, it updates the unfilled order file with the order and issues a production order on the factory that will build the item. It also prints out a set of shipping notices at the factory, and upon notification from the factory of the schedule on the item, transmits the schedule to that customer's service center. When shipment is made by the factory directly to the customer, a copy of the shipping memo is transmitted to the computer center, where the unfilled order report is updated and the billing procedure is instituted.

So far we have discussed the on line processing of order input. Now, let's take a look at another on line operation.

## INQUIRY PROCESS

The inquiry process operates on line as follows:

When a customer calls a service center to ask about delivery of an item, the fast unit operator receives the call and while the customer is still on the line, she types in a request for delivery and the computer examines warehouse records and returns a status of stock. This operation takes a maximum of 15 seconds and allows the customer to receive an exact status on the item. While the customer is still on the phone, we can take the order and enter it into the system.

An open order inquiry is handled in much the same manner. The operator accepts a call from the customer, and while he is on the phone sends an inquiry to the computer, where the unfilled order file is interrogated and the status of his order is printed out on the operator's machine; she then relays the information to the customer. If there is a question on more than one item, they are all handled at the same time and the customer gets his answer while still on the phone. If the customer desires, a complete status of all orders placed can be requested and will be printed out.

In the instance of a credit inquiry, the local salesman or account representative can enter an inquiry regarding the current credit status of a customer, and the computer will read out the customer record for him. If the present limitations of credit would be exceeded by a customer's planned procurement in the near future, then the account representative can make arrangements for additional credit review or collection so that the contemplated order may be entered and serviced without delay. Once again, the total indebtedness of each customer is updated, and not just the unpaid bills.

While all of these on line activities are going on, there is continual file maintenance taking place. Changes in schedules and new promises from product departments update the unfilled orders file and result in new promises being generated at customer service centers. Changes in the price structure of products are fed to the computer from product departments, and the product record file is updated and all customer service centers are advised of the new prices.

An operation that should appeal to financial people is our on line handling of customer payments. As customer payments are received at a service center, the payment is entered into the system and the customer record is updated at the computer system. Twice a day deposits are made in local banks and deposit notices are entered into the system. These deposits are reported to treasury operations twice a day and total funds on deposit are provided to them.

## REPORT GENERATION

The system generates paperwork that is directed to the customer in several areas. First, all customer invoices and order acknowledgements are generated by the computer, as well as customer unfilled order reports. These documents are the result of system activity and transaction tape programs. It is actually possible to generate these documents on the customer's site.

Each customer service center receives, on an automatic basis from the system, an unfilled orders report, a copy of all invoices and acknowledgements, a missed shipment report, credit reports and a current and year-to-date sales report.

In this case, these reports are generated daily but they can be programmed for any interval desired.

All of the product departments receive each day a report on all unfilled orders, a report of all warehouse receipts and shipments, and a report of all missed shipments on all their departments' products.

In addition to these daily reports, there is provided at varying intervals: complete customer statements showing all business placed by the customer during the past 12 months by item and for the past three years by product line, the status of all open orders, and the current and past year's credit status, as well as any significant seasonal habits the customer may have evidenced. Aged balances for product lines, for departments, and in total are prepared by the system, also operating reports detailing inventory turnover, inventory trends, performance against schedules and operating efficiency of warehouses and product departments. Order and sales statistics are generated and total assessed investment reports are provided.

## SYSTEM BENEFITS

This, then, represents an outline of the system components and the manner in which the system is designed to operate. We feel with this system we can provide customers with the following benefits:

First, we can guarantee on 85 percent of customer orders to make 24-hour shipment. All orders received by noon local time will be shipped the same day, and all orders received after noon will go the next working day.

We can provide customers with immediate answers to their inquiries about price, delivery, stock status or open orders.

Same day acknowledgement of orders is accomplished, giving the customer a record of entry of order and all pertinent information concerning the order, such as source for material, delivery promise, internal order number and confirmation of price.

We can provide customers with much more comprehensive open order reports giving complete details on all orders and consolidating all orders on one report.

Where the volume of customer business warrants, we will put terminals on customer location, thus providing the customer with the ability to receive acknowledgements, invoices and customer reports directly from the system. It is also possible for the customer to be equipped to actually direct inquiries on line to the system if desired.

Our systems studies have resulted in a standardization of forms used by all product departments. We found that, although most acknowledgements and invoices followed the same general formats, there were enough variations to confuse customer personnel. Now, all forms are identical, and whenever a customer looks for a piece of information, he finds it in the same location regardless of the product or department involved.

We expect the system to improve our market position by offering better customer service, and by allowing us to be in a more competitive position all across our product lines.

The system will allow us to do a better marketing job; it makes available complete current statistics, thus taking a great deal of the guesswork out of our projections and sales strategy.

The preceding has been a fairly detailed description of the system design, operation and benefits. Some of the numbers that apply to the system may be of interest.

At inception, the system will handle 26,000 messages per day averaging 280 characters per message.

There are 27 computer programs which have been developed.

Nearly 16,000 circuit miles of communications will tie the system together, with over 13,000 miles of teletype circuits and 3,000 miles of voice grade circuits.

The system study and design required just under three calendar years, and represents approximately 90 man-years of effort.

The system we have described is not a blue sky dream of some systems designer, it is a present day reality. If you closely examine each of the elements of the system, you will find that we aren't doing a single thing or using a single facility that hasn't been used previously. Tied together in this fashion the total system offers, to today's manager of a business, an extremely powerful and flexible tool that can allow him to make decisions based on full possession of all current operating data. We feel that such a tool is a necessity in order to meet the ever increasing pressures of competition.

Implementation of an on line centralized system is a challenge to decentralized management. Objective analysis of the corporate benefits must replace selfish interests, and operating compromises must be sound business decisions instead of hurdles to overcome. Such a system is not going to replace managers, but it is going to spotlight indecision and buck-passing. The challenge is to make the management of our business as good as the quality we build in our products.

by J. D. Edwards
Lockheed Missiles and Space Company
Sunnyvale, California

# An Automatic Data Acquisition and Inquiry System Using Disc Files

In brief, Lockheed Missiles and Space Company has installed a large scale Automatic Data Acquisition (ADA) system which ties together the company's manufacturing facilities located in Van Nuys and Sunnyvale, California. The system includes over 200 remote input stations which collect and transmit company operating data to a central data processing center. Two RCA 301 systems are used to record and control the flow of data transmitted to the data processing center. A large capacity RCA 366 data disc file is used to store information required to provide up-to-date information in response to inquiries received from remotely located inquiry stations. In addition to storage of data on the disc files, the system automatically records all incoming and outgoing data on magnetic tape to be used as input to the company's conventional off line business data processing applications.

The ever-increasing cost of recording, accumulating, and converting operating data to machine-sensible language, and the need to provide operating management with up-to-the-minute status information concerning company operations, spurred LMSC to install one of the nation's first large scale automatic data acquisition and inquiry systems using data disc files to store status information.

## HOW SYSTEM WORKS

The basic input unit of the system is the Remote Input Station, model 6220, developed and manufactured by the Radio Corporation of America. The remote input station is a device which enables an operator (usually a factory production worker) to record a maximum of 80 columns of data from a punched card; 12 alphanumeric characters from a plastic employee identification token; 10 numeric characters from variable data levers; and one of 11 possible transaction codes from a transaction control lever.

The remote input station is designed to operate in an industrial environment where moderate amounts of dust and other air contaminants are present. Installation requirements are modest: each unit requires six square feet of floor space; a 115-volt, 60-cycle power outlet; and a two-wire voice-grade telephone line for transmission of data.

Each remote input station also includes a subset transmitter which provides the means for serial transmission of data at the rate of 27.7 characters per second to a line concentrator. The function of the line concentrator is to act as a switching intermediary to distribute the intermittent demand for service from up to 25 remote input stations in order to timeshare four trunk lines connected to central recorders.

When a remote input station requests "Permission to Transmit", it operates the associated line relay in the line concentrator causing the control circuitry to attempt to connect the requesting remote input station to an idle trunk line and its associated central receiver located in the computation center. If all the trunk lines are in use, the line concentrator will wait until a trunk line becomes available and will then make a connection to a central receiving unit.

After a successful connection, the line concentrator relays a "Start" signal, initiated by the central receiver, to the requesting remote input station, causing it to transmit its message to the central receiver, via the connection through the line concentrator.

All reading at the remote input station is done statically so that input data remain in position for automatic retransmission in the event that accuracy checks are not satisfied. Accuracy control is accomplished by character-parity and gap checks which are made at the central receivers. Upon successful completion of transmission, the line concentrator relays the verification signal from the central receiver to the remote input station. Receipt of the verification signal by the remote input station allows the associated line concentrator to release and break the connection freeing the trunk line for use by other remote input stations.

Failure to receive a verification signal causes the remote input station to automatically re-read the input data and transmit the message repeatedly within a span of 15 seconds unless a verification of correct transmission is received from the central receiver in the meantime. At the end of 15 seconds, the remote input station sets an alarm, removes itself from the line, and signals the operator to release and examine the input documents. This arrangement provides that no one input station will monopolize the wire transmitting facilities.

The development of such a network of remote input stations capable of reliable data transmission via voice-grade telephone lines was closely monitored by LMSC. Prototype equipment manufactured by RCA was installed in August, 1961, at Lockheed's Sunnyvale facility and put to use recording the receipt and subsequent inspection and storage of materials arriving at the company's receiving docks. The prototype installation consists of 10 remote input stations wired to a punched paper tape receiver. The data accumulated by the system is used as input to conventional data processing equipment to record details of the receipt of material and to update material inventory records. The punched paper tape is also used to maintain a card file which is used to store information necessary to provide answers to telephone inquiries regarding the status of materials located in the receiving areas.

During the months following installation and operation of the prototype equipment, plans were being made to make full use of improved production model remote input stations as soon as the equipment could be produced and installed in large quantity. The responsibility for development of implementation plans was given to a working group whose members represent major organizations within the company.

The implementation plans developed by the working group provided for the gradual buildup of the system in two separate stages. The first stage provided for use of remote input stations to transmit data to central receivers where incoming data are recorded on punched paper tape. The second stage provided for use of two RCA 301 electronic data processing systems to replace the paper tape receivers and to provide for routing and control of messages flowing to and from an 88-million character capacity RCA 366 data disc file.

## OFF LINE ACTIVITY

The first stage of development is identified as the off line system. The off line system was activated in March, 1962, when 36 remote input stations were installed in a single building at our Sunnyvale, California, facility. Since that time, the system has increased in size to more than 200 remote input stations located in 15 widely separated buildings. These units transmit an average of 25,000 messages a day to the central receivers where the messages are recorded on paper tape.

The messages received during the day by the off line system reflect the activity of 5,000 employees who have routinely recorded the details of their day's work in the factory. A typical employee will use his personal identification token and the remote input station to report his presence at his work station in the morning. Usually within a matter of minutes he returns to the station to report his beginning work assignments. This reporting requires the use of his identification token, a punched card which accompanies the job package to provide identification of the job, and a transaction lever which he positions to identify job status. The station automatically transmits its own three-digit identification code and the central receiver automatically appends the time of day and day code to the message. Similar transactions are transmitted throughout the day until the employee "clocks out" to record his final transaction of the day. The remote input stations are also used by authorized supervisory personnel to record the receipt and completion of job assignments for purposes of receiving credit for estimated standard hours to offset the factory workers' labor charge against the job.

The incoming data are utilized almost immediately. Approximately every two hours, the accumulated data are converted from punched paper tape to magnetic tape and an audit list is prepared. Trained labor audit personnel scan each list to identify improper or incorrect messages. A questionable message prompts a telephone call from the labor audit clerk to the supervisor in charge of the area serviced by the remote input station which transmitted the questionable message. Corrective action is initiated immediately. The corrective action may involve the preparation of an adjustment transaction, correction of punched card data which accompanies the job package, on-the-job training to correct employee reporting errors, or the dispatching of an RCA Service Company employee to inspect and repair input equipment suspected of a malfunction. Prompt action such as this has considerably improved the accuracy of data resulting from factory operations.

It may be seen that the off line system provides a direct pipeline for data flow from the production employee to the company's computation center. The need for the factory employee to prepare written input to a clerical system is eliminated, usually to the considerable relief of the factory employee. The batching and keypunching of input documents are also eliminated, insuring the availability of current data up to the time the data flow is cut off, and the accumulated data are routed to conventional electronic data processing equipment for processing.

The one way continuous flow of data through the off line system of remote input stations to the company's computation center is a significant improvement over previous methods. However, the second stage in the development of Lockheed's automatic data acquisition and inquiry system provides much greater benefit through the use of electronic data processing equipment to replace the punched paper tape central receiving equipment, and through the addition of remote inquiry reply stations to the communications network.

Each inquiry/reply station added to the communications network consists of two separate units — an inquiry unit and a reply unit. The inquiry unit operates in a fashion similar to a standard remote input station except that, instead of punched card or plastic token input, the unit features 25 variable input wheels, each of which can be set to format alphanumeric inquiry messages up to 25 characters in length. Transmission of the inquiry message is accomplished as decribed for a remote input station.

The reply unit, or units, are associated with specific inquiry units under program control. Under this arrangement, inquiries from specified inquiry stations are directed to one or more reply units. The reply unit may be an IBM 26 card punch machine, modified to receive replies in punched card form, or a receive-only page printer capable of printing reply messages at the rate of 10 characters per second.

The electronic data processing equipment installed to replace the punched paper tape receivers and to provide on line magnetic disc file storage consists of two RCA systems, each equipped with a 20K core memory, and one RCA 366 Data Disc File.

Although the two RCA 301's are used primarily for real time data processing, which always results in data being immediately stored on magnetic tape and the data disc file, several other items of peripheral equipment have been included in the central system for batch processing purposes. This peripheral equipment includes an on line high speed printer, a paper tape reader-punch and a card reader, which are used for off line operations, such as program assembly and checkout, or other miscellaneous operations, such as printing the contents of the data disc file or batch updating of record files stored on the magnetic discs.

The 366 disc file included with the on line system provides 88 million characters of storage, divided between 24 discs which are coated on both sides to provide 48 usable recording surfaces. Additional file modules can be added to provide more than 1.5 billion characters of storage.

The 88-million character file module incorporates a motor pedestal to drive (at 1,200 rpm) the shaft on which the 24 storage discs are mounted; a hydraulic read/write head positioner to move the six read/write heads which service each disc surface; and a basic logic unit which converts from bit-serial to character-serial operations, checks parity, and determines that the head positioner is on the correct track of the disc.

The read/write head positioner operates to provide an average access time of 70 milliseconds. Each read/write head covers an area of the disc two inches in width which contains 128 separate data tracks. The data transfer rate is 32,000 characters per second.

## DATA EXCHANGE AND FLOW

In addition to the data disc file and two 301 systems, the on line system includes two data processing devices which are relatively unusual in business data processing. The first of these is the RCA 375 Data Exchange Control, which permits the exchange of information between the two RCA 301 systems via an interconnecting cable. The character transfer rate of this device is 95,000 characters per second, under program control.

The second unit is the RCA 378 Communications Mode Control, which permits direct data input/output flow between a 301 system and up to 80 communications lines (two-wire, voice-grade) connected to multiple remote devices which share the communication line through the use of switching devices. Each input line is provided with a single character buffer necessary to provide a proper interface connection between the remote equipment and the central processor.

The transfer of data in either direction between the one-character buffers and the central processor proceeds independently of other operations performed by the processor.

Each one-character buffer is associated with a 100-character line slot which is used as storage for information directed to or from the buffer. Each line slot requires an allocation of 100 contiguous positions in the central processor's core memory. In the case of input data, these line slots are filled from the associated one-character buffer at the rate of 27.7 characters per second as data are received from the remote station. A flag code character is placed in the line slot when transmission is ended from the connected remote station. This is the signal which the stored program that is cycling in core memory recognizes as a signal to move the completed message from the line slot and begin executing stored program instructions. The memory cycle time to address, bring into register, and regenerate two characters of data in its original memory location is seven microseconds for a 301 system.

## MASTER COMMUNICATIONS PROGRAM

The operation of the on line equipment complex requires a master communications program to control the two 301 systems. The master communications program package was developed by RCA and delivered with the hardware system. The normal objectives of the system and its communications program are as follows:

- To accumulate the input transaction message records as they are entered at the remote input stations.
- To select and route special transaction messages to specified areas of core memory in the second 301 system where on line data processing requirements such as editing and arithmetic operations are performed.
- To transcribe all incoming messages onto a magnetic tape journal to be used for subsequent off line data processing operations on conventional electronic data processing equipment.
- To update, and inquire of, file information recorded onto the data disc file.
- To distribute inquiry reply messages to remote printers in response to messages received from remote inquiry stations.

The master program is also designed to recognize, and operate under, unusual conditions when either of the 301 systems are not available due either to maintenance requirements or off line use, such as program testing or batch updating of master records on the disc file. Operation with only one 301 system is restricted to the collection of incoming data, which is temporarily stored on magnetic tape until both systems are available to reduce the backlog of work. The program is designed to guarantee maintenance of the chronological order which is necessary to insure that items intended to update records on the data disc file are properly handled, and that backlogged inquiries requesting data from the data disc file are processed in proper sequence. This programmed flexibility insures maximum availability of central processing equipment to service the remote equipment.

Under the normal operating mode for the on line system, one of the RCA 301 systems (System A) is assigned the prime function of collecting and recording all incoming data on magnetic tape. Additionally, System A is programmed to distinguish between those input data records which are needed to update the data disc file, and those that are inquiries requesting information from the file. Upon recognition of either condition, System A transmits the information across the data exchange channel to the second 301 system (System B). System B is programmed to process incoming data received from System A, as required to update master records contained in the data disc file and to respond to inquiries by withdrawing the requested information from the data disc file for distribution to remotely located printers.

Actual use of the on line system began December 17, 1962, and has been continuous since that time. The early use of the system has been restricted to a worst-case environment in order to discover and correct any program equipment problems which might develop under such conditions. The worst-case environment was established by installing 18 remote input stations in one of the company's manufacturing facilities located approximately 400 miles from the on line central processing equipment. Several inquiry stations were also provided, along with remote printers to receive printed replies to inquiries and a card punch machine wired to receive replies in punched card format.

To further complicate matters, one of the 301 systems was scheduled for interruption on alternate hours for use in developing program routines and training of operations personnel. These frequent interruptions provided opportunities to utilize the program's ability to process backlogged data and to recover from other unusual situations occasioned by operating with only one of the 301 systems instead of the usual two-system arrangement.

Actual operating experience under these conditions has produced increased confidence that the on line hardware system and the master communications program will perform satisfactorily over extended periods of time. However, our operating experience is still quite limited, especially in regards to the data disc file. Several months will pass before the necessary operating statistics are available to formulate a reliable prediction of long range system reliability.

## MAJOR APPLICATIONS

Three major applications have been planned for extensive use of the remote inquiry/reply stations and data disc file storage capabilities of the system: shop order location, purchase order status, and material inventory status.

The first application to be activated is the shop order location system. Under this system, a master record is placed on the data disc file for each production order released to the manufacturing shops. As orders move through the manufacturing cycle, their status and locations are recorded by means of the remote input station network and the master record on the data disc file is instantly changed to reflect the most current production status and location.

The system provides for use of remote inquiry/reply stations to accept inquiries concerning individual part numbers maintained on the data disc file. Replies which indicate the current status of the shop orders pertaining to the queried part number are automatically returned to the remote inquiry/reply station, usually within a few seconds.

The shop order location application is presently in operation on a test basis. This is the application selected to test the equipment and master program under worst-case conditions requiring transmission of data between Van Nuys and Sunnyvale, a distance of approximately 400 miles. In addition to shop order location, plans have been made to accumulate data necessary to produce reports concerning shop equipment utilization, labor distribution and analysis, manpower and equipment shop load forecast, and miscellaneous exception reports of value in making management decisions.

The other two applications presently planned for extensive use of Lockheed's automatic data acquisition and data disc file system are purchase order status and material inventory status reporting. These applications will provide for storage and immediate access to information concerning procurement activity involving over 200,000 purchase orders a year and over 70,000 inventory items in 100 accounts at plants in Sunnyvale, Palo Alto, and Van Nuys, California.

In addition to the applications which make extensive use of the complete capabilities of the system, several applications are being planned which will utilize the data collection network to record data to be used as input to new or revised applications processed off line on conventional electronic data processing equipment. These additional applications cover a wide range from financial reports to product quality assurance reports.

Lockheed's new automatic data acquisition and inquiry system represents a significant achievement in improved data processing capabilities for the company. For the first time, equipment is available to accurately record meaningful data as a by-product of plant operations for immediate storage and access on a large scale data disc file. This new tool provides a means for quick solution of many of our most stubborn data processing problems, but not without offering a challenge to the user to develop new techniques of use which will revolutionize traditional electronic data processing concepts and result in the development of a completely integrated, real time system to meet all of the company's administrative data processing needs. At Lockheed we feel that we have reached a beginning and the end is not in view.

by J. A. Phillips
General Electric Company Computer Department
Phoenix, Arizona

# Disc Files in a Military Supply Depot

DISC FILES, to me, are the most interesting, and probably the least understood peripheral of the computer business. They aren't like tape or card applications where maximum speeds are controlled by a governor. With disc files the speed can be controlled by the ingenuity and imaginative capacity of the systems designer. This is the challenge to the systems designer since many alternatives have to be evaluated before the best solution can be determined.

During the past few years as senior application engineer in government sales, I have worked on many specifications requiring disc files and *no* two have required the same basic techniques. Each one has a special twist that can utilize a specific feature of the hardware. It is true that some basic concepts are developed, but there is always the realization that the system might be improved.

We in the General Electric Company refer to disc files as MRADS (Mass Random Access Data Storage). Throughout this report I will use MRADS to refer to disc files. There is a certain amount of hardware background required in order to understand any MRADS system. I will try to present MRADS in a logical sequence for easy understanding of the basic concepts and then relate it to a military supply depot.

Disc files have become a necessary way of life in the military supply area. With the implementation of the Department of Defense operating manual, MILSTRIP (Military Standard Requisitioning and Issue Procedure), which establishes time parameters for handling priority supply transactions, most supply depots are using disc files to meet the stringent time requirements.

The military installation I will use as an example of one way of handling the supply problem is the MATS (Military Air Transport Service) installation at Charleston Air Force Base, South Carolina. Charleston is the pilot installation, with seven additional ones to follow. It is planned that the supply programs developed at Charleston will be utilized at the other installations with minimum modification. Anticipated modifications include a different stock number randomizing technique to cover the stock number variation, and some local coding. Incidentally, it is not only in the supply area that the trend to disc files is noted. Personnel accounting, finance and accounting, and other government applications are finding MRADS a means to faster processing and more current reporting.

Let us now look at a MRADS file configuration (**Exhibit 1**). Five units are involved: the GE-225 central processor, controller selector, file controller, file electronics, and the MRADS file. Actually there are only four separate units since the controller selector is contained in the central processor. Note that the file controller can handle four MRADS files. All four can be seeking simultaneously although only one can be reading or writing at one time. There are 55,000 accesses/second which represent the number of words that can be transferred between the central processor and the controller selector peripherals in one second.

## CENTRAL PROCESSOR

The first of the five units to be reviewed is the central processor. There are several features of the GE-225 central processor with which you should be familiar in order to better understand the MRADS system.

1. The central processor processes data in both binary and alphanumeric form. The central processor normally operates arithmetically in the binary mode. Subroutines are provided to convert data from one mode to the other.

2. The central processor has a memory composed of 20-bit words plus a parity bit. Normally a word is considered to be three alphanumeric characters or 5½ decimal digits in binary. In the binary mode, one word can represent any number up to 524,287. Each bit within a word can be programmed individually. The 20-bit word plus parity is the transfer unit between the central

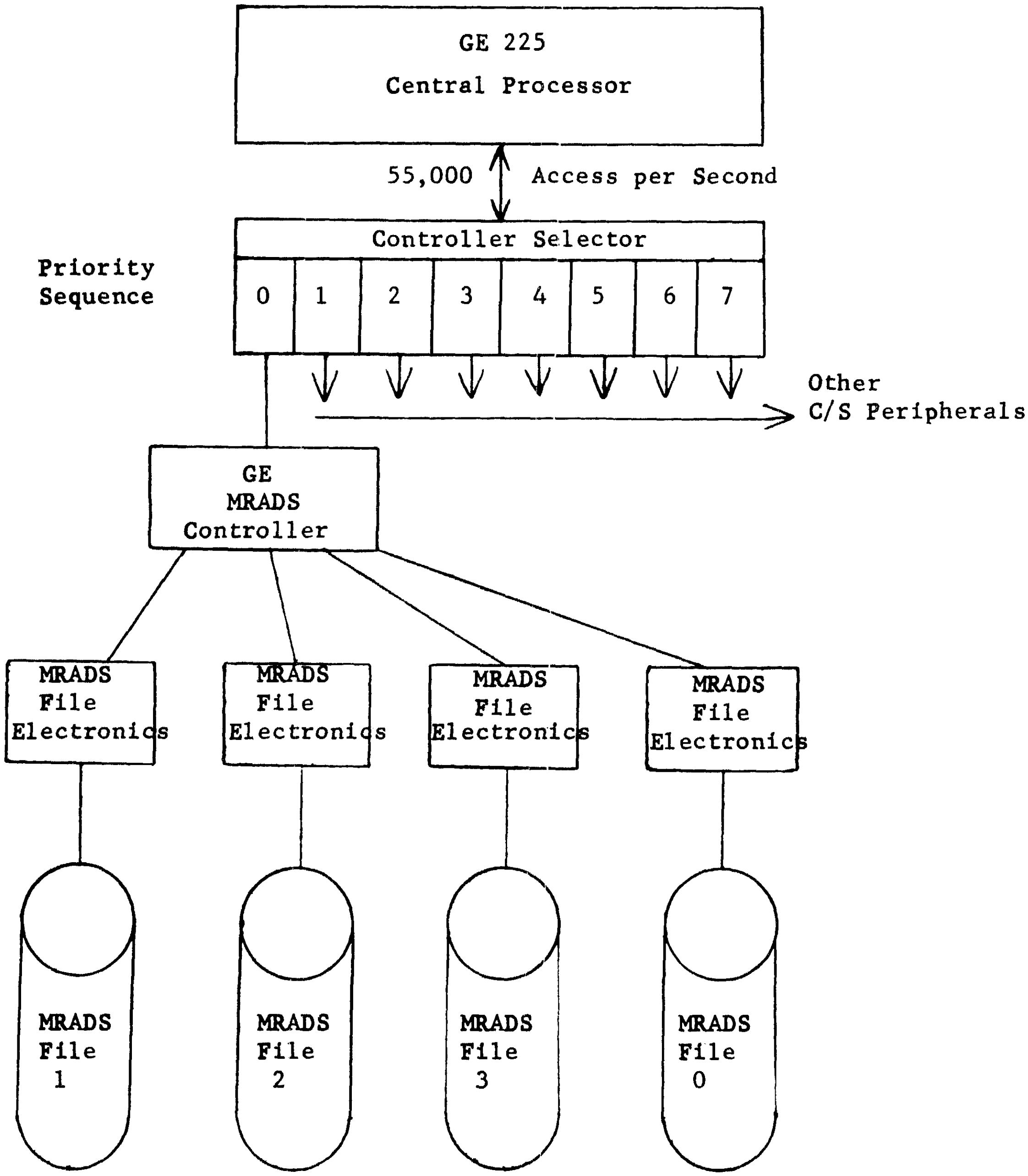



**Exhibit 1. MRADS FILE CONFIGURATION**

processor and the MRADS file. Between the central processor and the MRADS file there is, in effect, an image transfer. A great saving in MRADS requirements can be achieved by making each word contain the maximum amount of data.

3. The MRADS system is controlled by a computer program. In fact, the MRADS file is usually considered an extension of the central processor memory as many computer programs are held on MRADS file and called into the central processor memory as required.

4. The central processor's memory is available in word sizes of 4K, 8K and 16K. This is the equivalent of 12,000, 24,000 and 48,000 alphanumeric characters respectively.

5. Memory access time is 18 microseconds, which computes out to the 55,000 accesses per second indicated in Exhibit 1. In order to control these accesses, a traffic cop, referred to as the controller selector, is used to see that no traffic jams occur. There are eight input/output hubs which he must control. Each hub could have a peri-

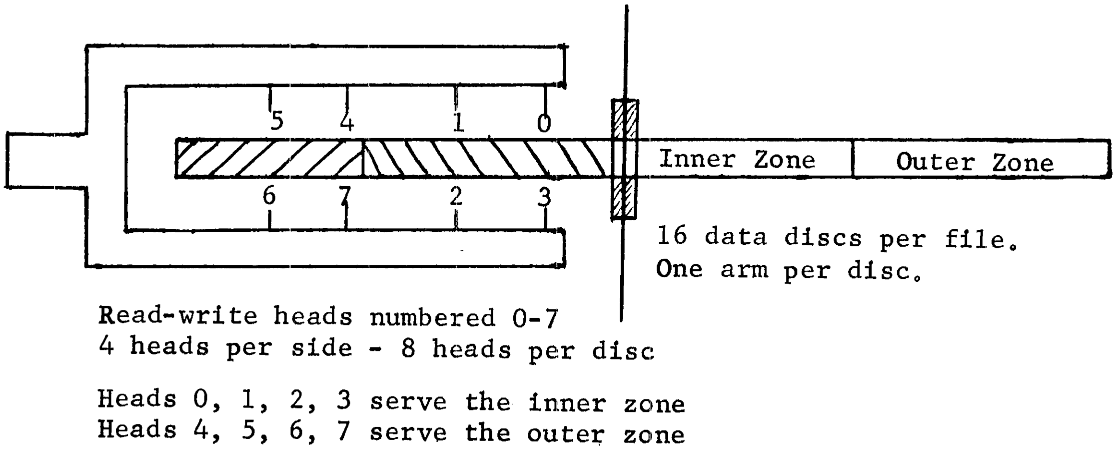



**Exhibit 2. MRADS ARM AND HEAD FORMAT**

pheral controller attached. Sufficient buffering is provided in each peripheral to meet normal operating situations. Normally the MRADS file controller is attached to hub O, which has top priority.

## MRADS FILE

A better understanding may result if we skip the file controller and the file electronics for the moment and take a look at the general description of the MRADS file itself. There are 16 discs which contain 32 data storage surfaces. There are 16 positioning arms, each one operated independently. There is an actuator for each arm which moves the arm parallel to the disc. The arm remains positioned until a new positioning instruction requires a movement to another location. There are 128 read/write heads, eight heads per disc and four heads per surface (**Exhibit 2**).

The disc revolves at 1,200 rpm. The data capacity of the file is 18.8 million alphanumeric characters or 34.4 million decimal digits in binary. The actual number usually varies as data in most applications are mixed.

Each surface of each disc is divided into an inner and outer zone (**Exhibit 3**). The transfer rate from the outer zone is 25,000 words per second and from the inner zone 12,500 words per second. I refer to words instead of characters or bits because the data content of a word can vary depending on the data mode. The maximum number of alphanumeric characters per second would be 75,000 from the outer zone and 37,500 from the inner zone. There are 128 *circular tracks* in each zone, making a total of 256 tracks on each side of each disc. The outer 128 tracks are divided into *16 sectors* and the inner 128 tracks are divided into *eight sectors.* Since there are four read/write heads for each side of the disc the actuator must move the arm only *64 positions* to serve the 256 tracks on each disc surface. This, of course, helps minimize the access time. The total number of sectors per file is 98,304.

Now let us take a look at one of the 98,304 *MRADS sectors* (Exhibit 2A).

It contains storage space for 64 words of information which, of course, can be tables, computer programs, or data. It is the smallest unit that can be transferred between the central processor and the MRADS file. The actual transfer is made in words, but in units of 64 words only. Any number of sectors from one to 16 can be transferred with

MRADS SECTOR

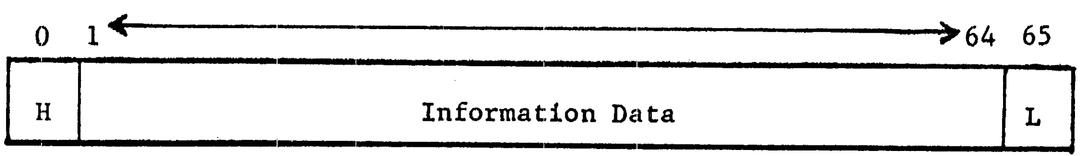



**Exhibit 2A**

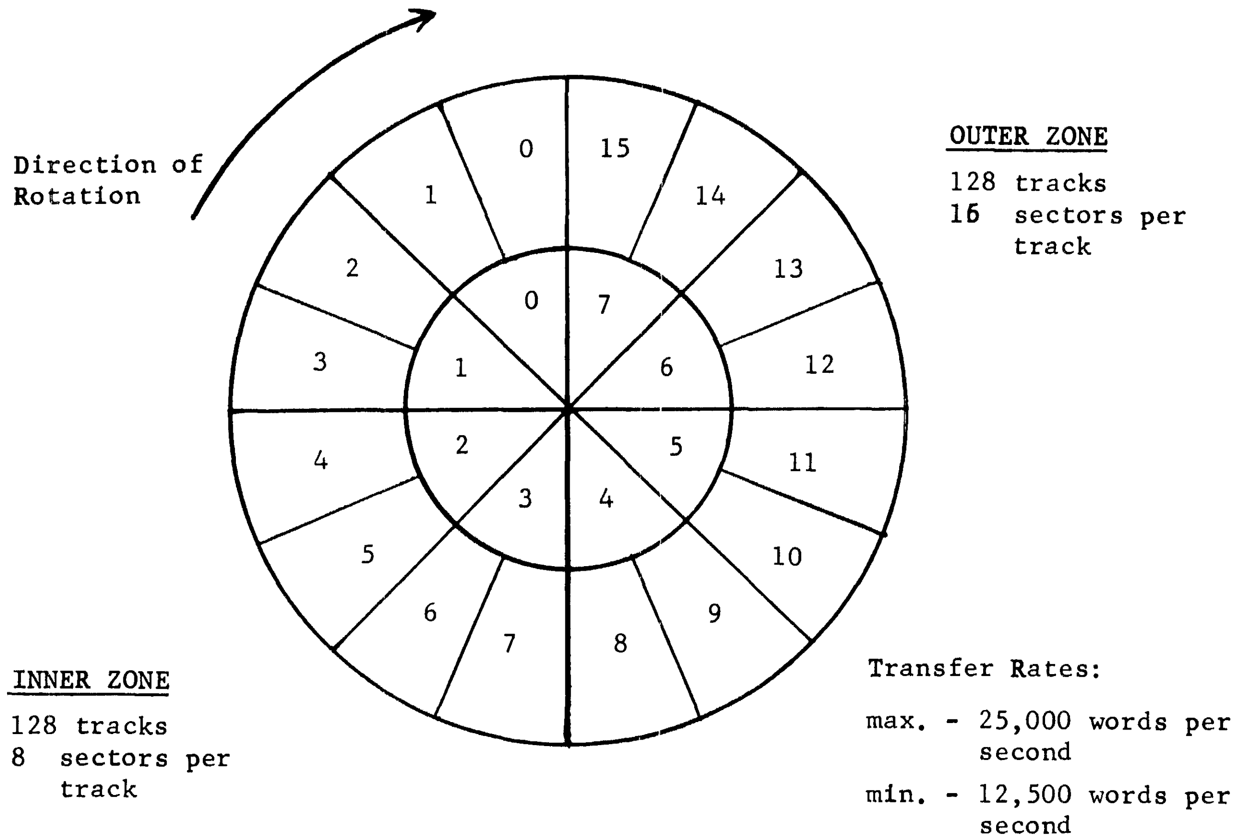



**Exhibit 3. MRADS DISC FORMAT**

one instruction, so the maximum transfer on one instruction would be 16 sectors x 64 words or 1,024 words.

The "H" refers to the *header word* which precedes a sector and contains the sector address, that is, the disc, track and sector. This header word is compared with the positioning instruction that is being held in the file electronics register. There are two phases to this comparison. *First,* the comparison of the disc and track, which is called the *positioning time.* During the positioning time, the disc is selected and the arm is moved to the track. After the disc and track have been confirmed, the sector is compared. This is called the *latency time.* Latency time should never exceed 52 milliseconds as one revolution will bring all of the sectors on that track under the read/write head. When the sector has been located, the "ready" signal is forwarded to the file controller and the central processor for processing. Normally a read or write instruction follows.

The "L" refers to the *longitudinal parity word,* sometimes referred to as the 65th word. It is generated by the file controller whenever a sector is written on the MRADS file. It is used in case a parity error is noted during a later instruction.

The 64 words of data can be in any combination of bits required for the application. There are four data modes, which I like to call *compact modes* since it implies getting the most use out of the space available. The compact modes are possible because of the individual bit manipulation ability of the GE-225. Bits can be shifted, tested, added or deleted. This makes it possible to give significance to each bit position. The *BCD mode* is the standard six-bit configuration, which provides three alphanumeric characters to the word. The *binary mode* pertains to numeric data that has been converted to a binary bit configuration. We refer to it as 5½ characters per word. The four-bit BCD pertains only to numeric data that will *not* be used arithmetically. Four bits will describe any numeric character so it is possible to put five numeric characters to a 20-bit word. *Other* data mode refers to fields in a file that would be identified by a "yes" or "no" condition. This could indicate if an item is hi-value, serviceable, reparable, or not. Each one of these fields could be indicated by one-bit position. A maximum of 20 "yes-no" type questions could be included in one word. It is possible then for one word to represent from three to 20 characters.

Perhaps this helps clear up my preference for the use of words instead of characters when referring to transfer rates. It varies according to the content of the files, and the system designer's ingenuity. By using compact modes a space savings of 20-30 percent is very possible.

## FILE ELECTRONICS AND FILE CONTROLLER

The file electronics acts as an interface between the file controller and the MRADS file; it interprets impulses from the file controller and causes the MRADS file to respond according to instructions from the central processor.

MRADS file errors are indicated on the file electronics unit. The file controller has five basic functions: It controls the flow of data between the central processor and the MRADS file. When reading or writing it will be requesting, through the controller selector, access to memory. It maintains the central processor memory location address where the data are going to or coming from. It keeps count on the number of sectors that are to be transferred (1-16). In reality, it takes charge after the central processor has forwarded the instruction, leaving the central processor free for other processing.

It interprets the instructions and takes the necessary action to notify the various units of what action to take.

It maintains a record of the status of each file – that is, whether a file is busy or not busy, or if there is an error on an individual file. This status can be interrogated by the computer program to determine the course of the program.

The file controller can hold a read or write instruction pending the ready signal from the file electronics, and then make the transfer of data without interrupting the computer program.

The file controller generates the longitudinal parity word and attaches it to each MRADS sector as it is written. It also passes the longitudinal parity word into memory when a parity error occurs on a *read* instruction.

Timing of an MRADS system is not necessarily according to the averages; in fact, it seldom is according to the averages. The published average access time for the GE-225 MRADS system averaged over an entire file is 225 milliseconds. Only on a rare occasion will a proposal be prepared using 225 milliseconds as a timing basis. As noted earlier, there is generally a special twist to each application that will reduce the average access time. A great deal depends on the concepts of the systems designer and how he plans his file arrangement and the sequencing of the various operations.

Two special features should be mentioned. First, *arm remains positioned.* Each arm moves independently of the others and maintains its last position until a new instruction is received to move it to another track. There are two conditions, "power on" and "power off", that control the status of the positioning arm. If the read or write instruction indicates a power on condition, the R/W head will remain *locked on* that particular track. If the next instruction requires the same disc and track, no positioning time is required; only latency time is necessary before reading or writing from that particular track. However, if the next instruction requires a different track that involves repositioning, power must first be dropped before positioning can occur.

If the read or write instruction indicates power off, that R/W head will stay near but *not locked on* the same track. This means that if the next instruction requires the same disc and track, the track position must be confirmed in addition to latency time before reading or writing can occur from that particular track. However, if a different track is required, power is already dropped and positioning may begin immediately. The power on and power off capability is an automatic part of the read or write instruction and is used to meet the requirements of the situation.

This feature is used to great advantage in file updating applications such as the MATS supply application. Using this feature for updating MRADS records reduces the average access time from 225 to 141 milliseconds.

Second, the *read-after-write instruction* transfers sectors to the MRADS file and checks the information written for correct transfer. This checking does not affect the operation of the central processor as no data enters memory during this parity check. If there is a parity error, the file controller will indicate this status to the central processor's program and action will be initiated to correct the error. If no error is found, the central processor program continues uninterrupted. This assures the program that data have been transferred correctly. This instruction was used consistently on the MATS supply programs.

## MATS SUPPLY SYSTEM

So much for the hardware, now let us look at the MATS supply system which represents an excellent example of what can be done on a real time basis with a moderately priced MRADS-tape combination.

A full detailed description of a MILSTRIP computer approach would be too lengthy. The best that can be accomplished briefly is to generally cover areas of major consideration and to explain in some detail the techniques and solutions that were adopted.

To set the stage for this installation let us assume we are to design a computer system to handle a maximum of 60,000 federal stock numbers and that the system must further be capable of processing receipts, issues, and all other supply transactions on a real time random basis rather than on a scheduled batched approach. The requirement of real time processing dictates the prime system configuration, i.e., it will be a MRADS rather than a tape approach. How large a MRADS is required? To begin the evaluation, **Exhibit 4** shows a breakdown of the various record

**SUPPLY INSTALLATION**

**SUPPLY RECORD TYPES AND VOLUMES**

| RECORD TYPE | VOLUME | GE-225*** RECORD LENGTH |
|---|---|---|
| *Item Master | 60,000 | 40 |
| Item Trailer | 20,000 | 24 |
| **Due In | 15,000 | 17 |
| Due Out | 15,000 | 17 |
| Work Order | 4,000 | 4 |
| On Base Stock (Pre-Issue)<br>Forward Supply Stock (FSS) | 8,000 | 6 |
| TOC Property<br>WRM<br>Project | 16,000 | 6 |

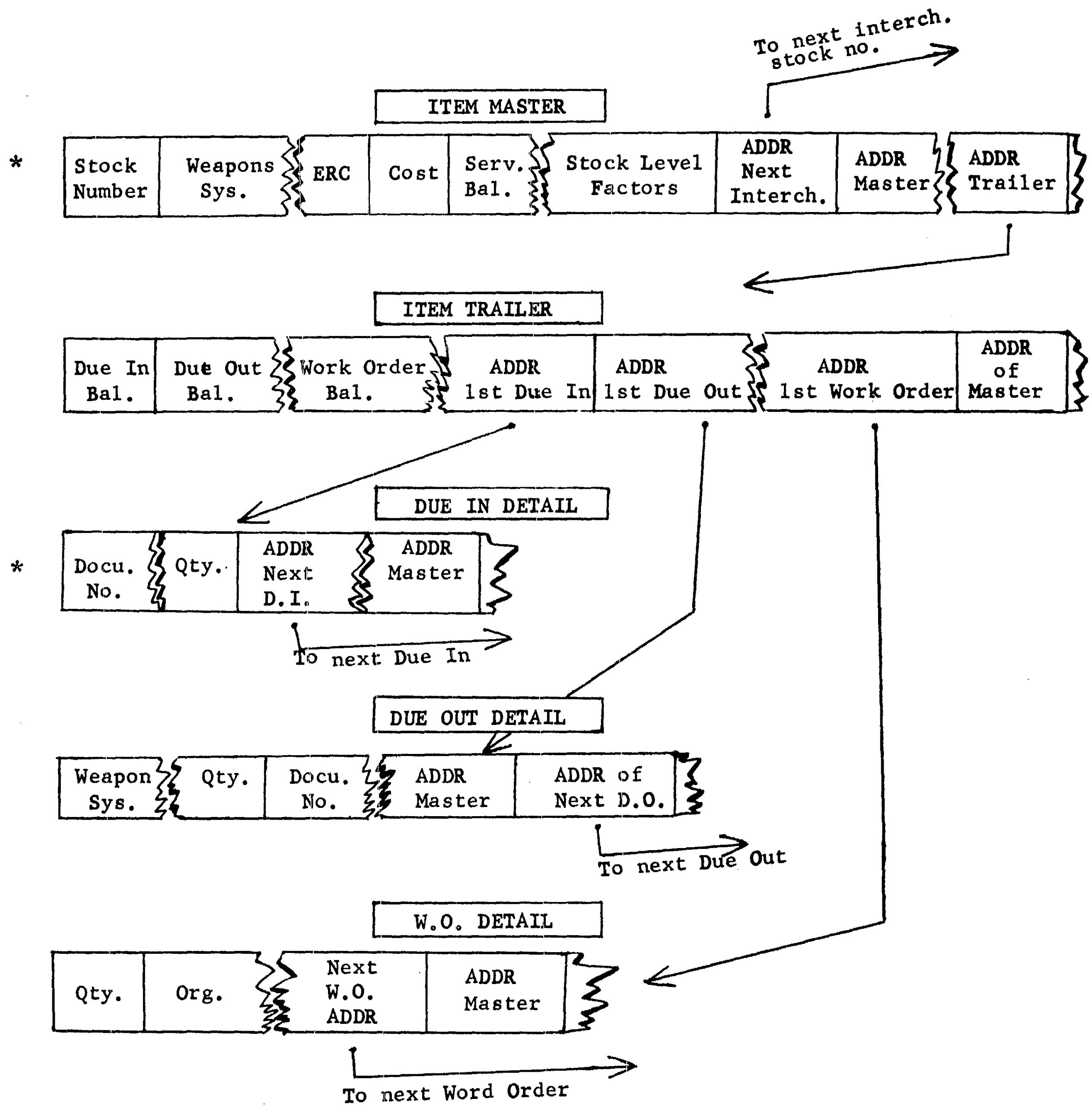
*Randomized by Stock Number

**Randomized by Document Number

***Expressed in Word Size

**Exhibit 4**

## SUPPLY INSTALLATION
## TYPICAL SUPPLY RECORD LINKING




*Randomized Into Memory -
All Other Record Types Packed

**Exhibit 5**

types and volumes involved. In studying Exhibit 4, we note that:

1. Only 33 percent of the stock numbers (20,000 out of 60,000) normally have due in, due out, work order, etc., balances. Because of this characteristic it is possible to break the stock number master record into two different segments, i.e., item masters, which contain the basic information about each item, and item trailers, which contain detail amount balances and the MRADS address where more detail can be obtained. This technique permits us to more fully utilize the MRADS.
2. Trailer, due in, due out, WRM, etc., individual records are to be established and cross-referenced to the appropriate item master in MRADS memory.
3. The number of GE-225 words (within a 64-word record) required by each record type is shown to the far right under the heading "GE-225 Record Length". Whenever possible amounts are maintained in binary to increase the data storage capacity.
4. Only the item master and due in records are to be placed on the MRADS and referenced in random order; all remaining records will be "packed" and cross-referenced to the item master. Expressed differently — the file will be normally approached with one of two different control keys — stock number (in the case of issues, etc.) and document number (in the case of receipts).

Having determined the record volumes and types, attention can now be directed to the detailed layout of record formats. In reviewing **Exhibit 5**, which contains a graphic picture of five major record types and their linking, note that:

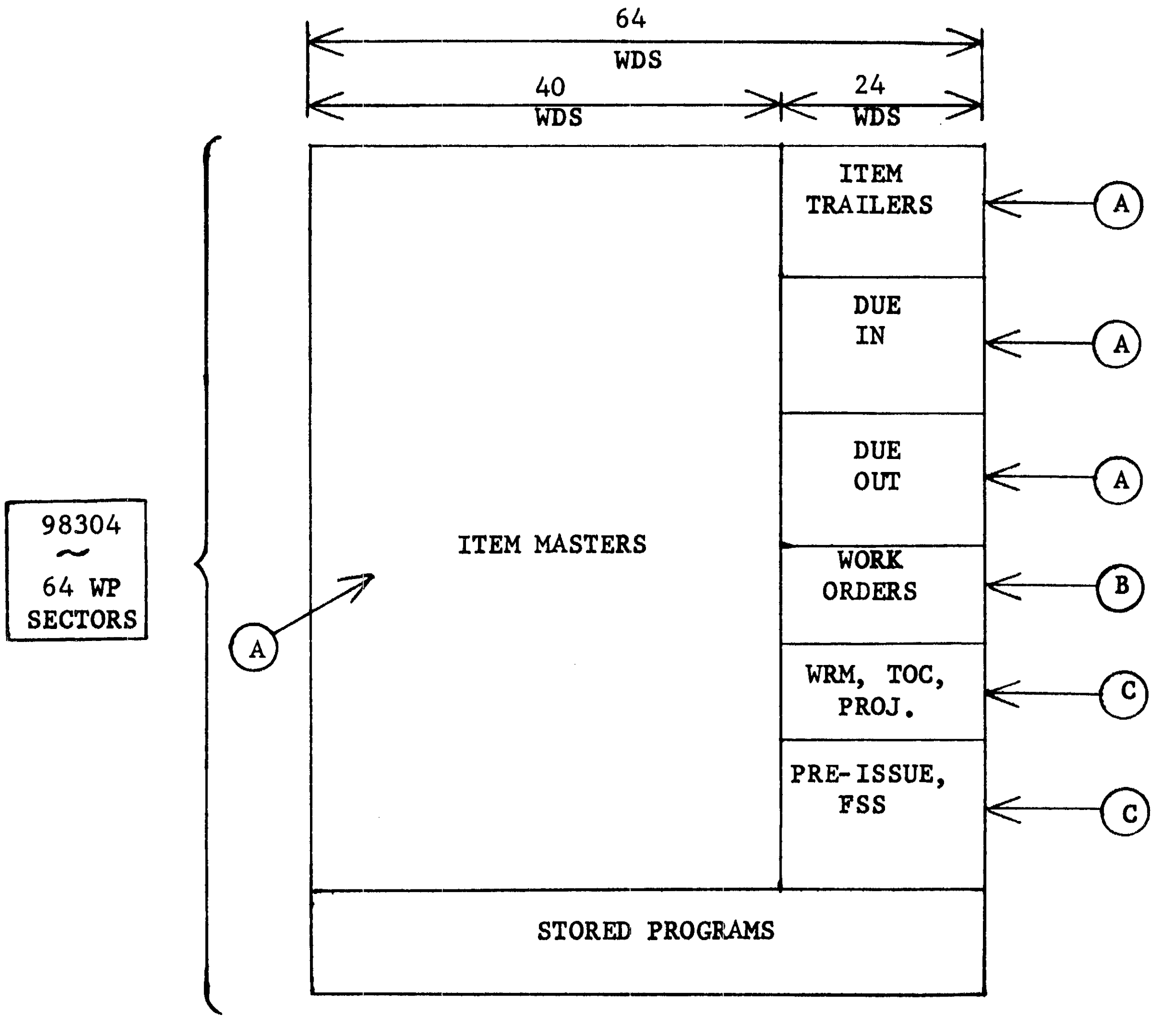
1. There will be one item master for each stock number.
2. Prime item masters are cross-referenced to interchangeable stock numbers. The last item record in an interchangeable chain is blank in the address of next interchangeable field.
3. Each item master has an address of master field used only in the case of interchangeables to find the prime master in the event a due out must be established, etc.
4. The address of trailer field in each item master will contain one of two different types of information:
   a. Blank if there is no trailer
   b. Address of trailer if there are due in, due out, etc., balances applicable to the appropriate stock number.
5. Item trailers are packed in the MRADS and contain the summary balances of due ins, due outs, and other detail balances.
6. Item trailers carry the MRADS address of the first due in record, first due out record, etc. Each of the first due in, due out records are in turn cross-referenced to subsequent details (of like type) in the chain, etc.
7. Due in detail records are randomized (by document number) in the MRADS and their address placed in the item trailer (if the first due in) or in the preceding due in (if the second or subsequent due in).

As previously mentioned, the GE-225 MRADS contains 98,304 sectors of 64 words each. The 98,304 sectors can be thought of as a "plane" memory as shown in **Exhibit 6**, which presents a graphic picture of the disc file layout. Note that:

1. Item masters are randomized into the first 40 words of a 64-word sector.
2. Item trailers are packed in the last 24 words of sectors on the top few MRADS discs.
3. Due in records are randomized into the last 24 words of sectors on subsequent MRADS discs.
4. All other records, e.g., due outs, WRM, etc., are packed into the last 24 words of sectors on succeeding discs.
5. A sufficient amount of space is left on the lower disc for holding programs.
6. More than one detail logical record, such as work orders, are packed into a 24-word physical space.

By packing is meant the assigning of MRADS addresses in a sequential order from a known base number. Counters are maintained in memory for assigning these sequential locations. Each item detail area, except due in, has an established range that cannot be exceeded without a repacking of the file being accomplished. As a new item detail is processed, the next available MRADS address is assigned from the appropriate counter. This did not create any problem so long as the assignments were on a one detail to one 24-word record relationship, such as due out and item trailer. In the case of item details packed more

## SUPPLY INSTALLATION
## DISC FILE LAYOUT



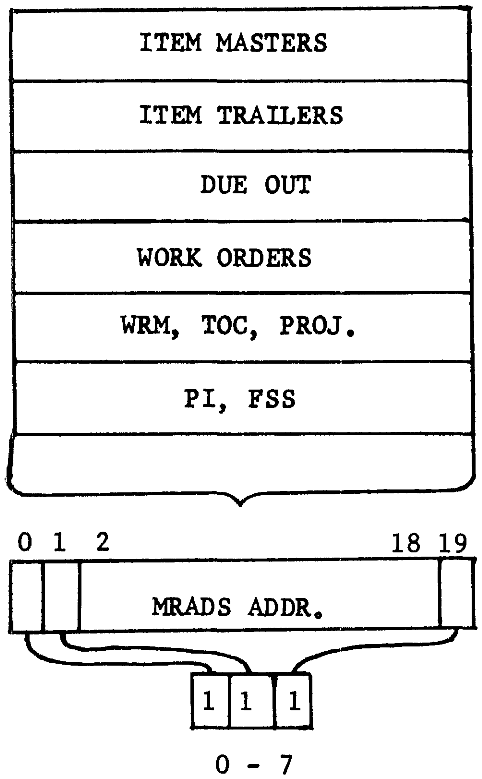

Ⓐ Packed 1 - Per 64 WD Sector

Ⓑ Packed 6 - Per 64 WD Sector

Ⓒ Packed 4 - Per 64 WD Sector

Exhibit 6

**WORD COUNTERS IN MEMORY**




COUNTER

**Exhibit 6A**

than one to a 24-word record, an additional control had to be developed (see **Exhibit** 6A).

The MRADS address utilizes 17 bits of the 20-bit word thus leaving three bits available for additional packing control. The three bits – 0, 1, and 19 – are combined into a word and used to pack data into the assigned MRADS address. When the specified number of item details have been packed into the MRADS address, the counter is increased by one and bits 0, 1, and 19 are zeroed. Having the ability to manipulate individual bits within a word made this type of program control possible.

Having determined that one disc file will do the job, the computer configuration can now be set. **Exhibit 7** shows the components of the configuration placed on rental by the Military Air Transport Service to process a MILSTRIP system similar in volume to the figures just reviewed. In studying this exhibit certain considerations should be kept in mind:

1. The disc file is to contain data and programs.
2. A 4K memory is not large enough to contain all the programming required. An overlay technique (to be covered later) must be adopted.
3. Input transactions are to be read in the card reader.
4. The typewriter is used for error indications and instructions to the operator.
5. The card punch (100 cpm) is utilized for error and action output.
6. Only one tape handler (15kc) is used for capturing the transaction "picture" as well as recovery and restore purposes. This transaction tape will be used for later processing.
7. The 300 lpm printer is used principally for preparing general purpose supply documents during real time operations and for reports run during off hours.
8. Equipment is run on a real time basis for approximately seven hours per day processing an average of 4,500 supply transactions in random order. An executive program is in memory at all times during this period.

PROGRAM OVERLAY

In MRADS applications all relevant programs are usually run against a unit of input data, whereas in tape processing data are normally run against the programs one at a time with the intermediate results stored on tape or card. In a MRADS-oriented real time supply program such as MILSTRIP, it is vitally important that all the actions and reactions triggered by a unit of input data be performed and completed prior to proceeding to the next transaction. Unfortunately, it is not economical to buy a core memory sufficiently large to hold all the programs that might conceivably be called upon. To circumvent the need for (and expense of) unusually large core memories, a coding technique of program overlay was adopted. This approach worked as follows:

1. The entire problem was broken into multiple segments (called programs) of a maximum size, such as issued, receipts, etc.
2. Each program was assigned a number.
3. If the program to be performed in a given leg of the program could not be completed within the maximum allowed overlay size, the work was divided into two or more programs with the last few steps of the first

## SUPPLY INSTALLATION COMPUTER CONFIGURATION

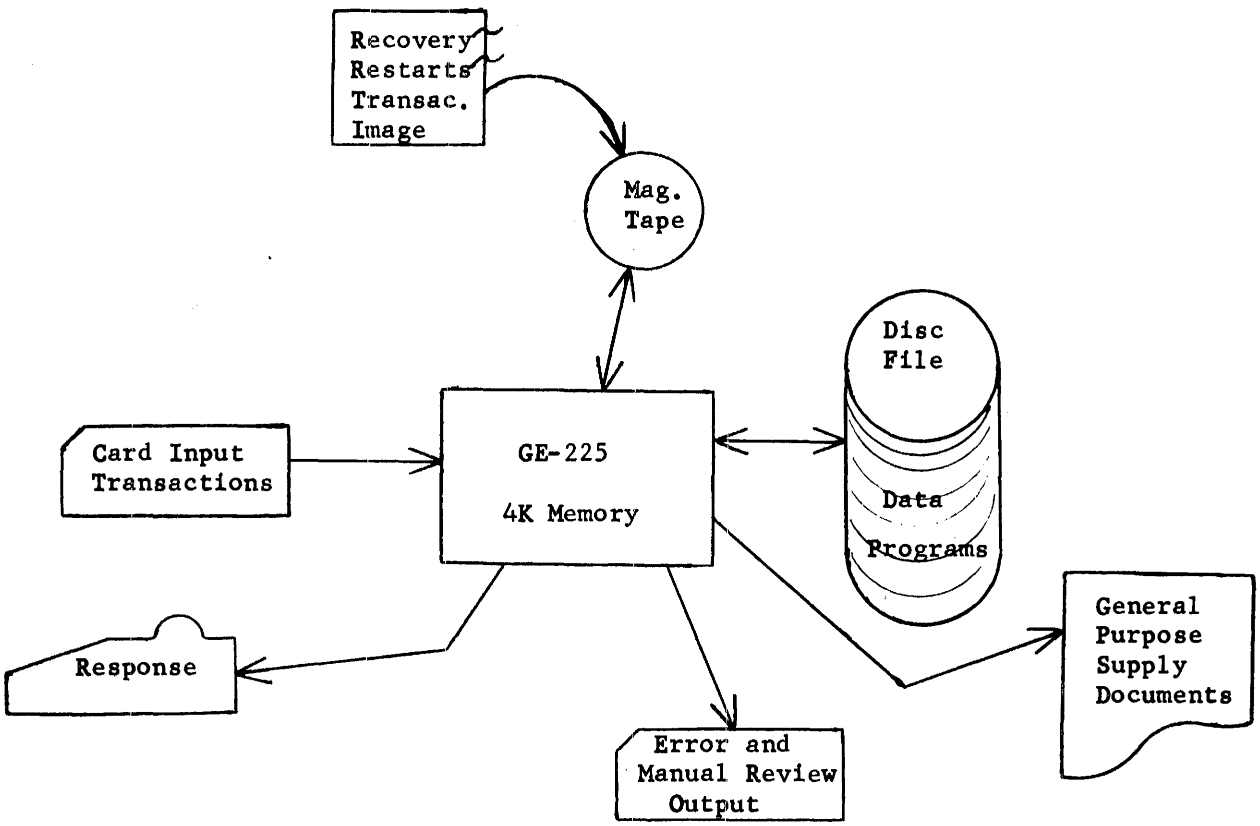



**Exhibit 7**

program "calling out" the number of the subsequent program that must be performed, etc.

4. Programs were loaded in MRADS.
5. Core memory was divided into distinct areas and functions as shown in **Exhibit** 8. Note that:
   a. A program overlay area (called "1024 wd. Overlay Area") is provided in memory for the programs to be read from the MRADS.
   b. Standard areas are allocated for card input, card output, MRADS data records, etc. images.
   c. Standard subroutines utilized by programs when brought into core from MRADS, e.g., BCD to Binary, etc., are left in computer memory at all times.
   d. Interim working storage of 512 words is provided in memory for storing temporary results to be worked on by subsequent programs loaded into the overlay area from MRADS.
6. Processing would occur as follows:

   The executive program would read a card from the card reader and call into memory the basic audit program from MRADS. Control would pass to the basic audit program to audit the card. If acceptable, the basic audit program will call into memory, from the MRADS, the applicable program for processing. Control would then be given to the program in the overlay processing area which would cause a MRADS data record to be brought into memory and perform the necessary processing. The overlay program, when completed with its work,

**SUPPLY INSTALLATION**

**MEMORY LAYOUT**

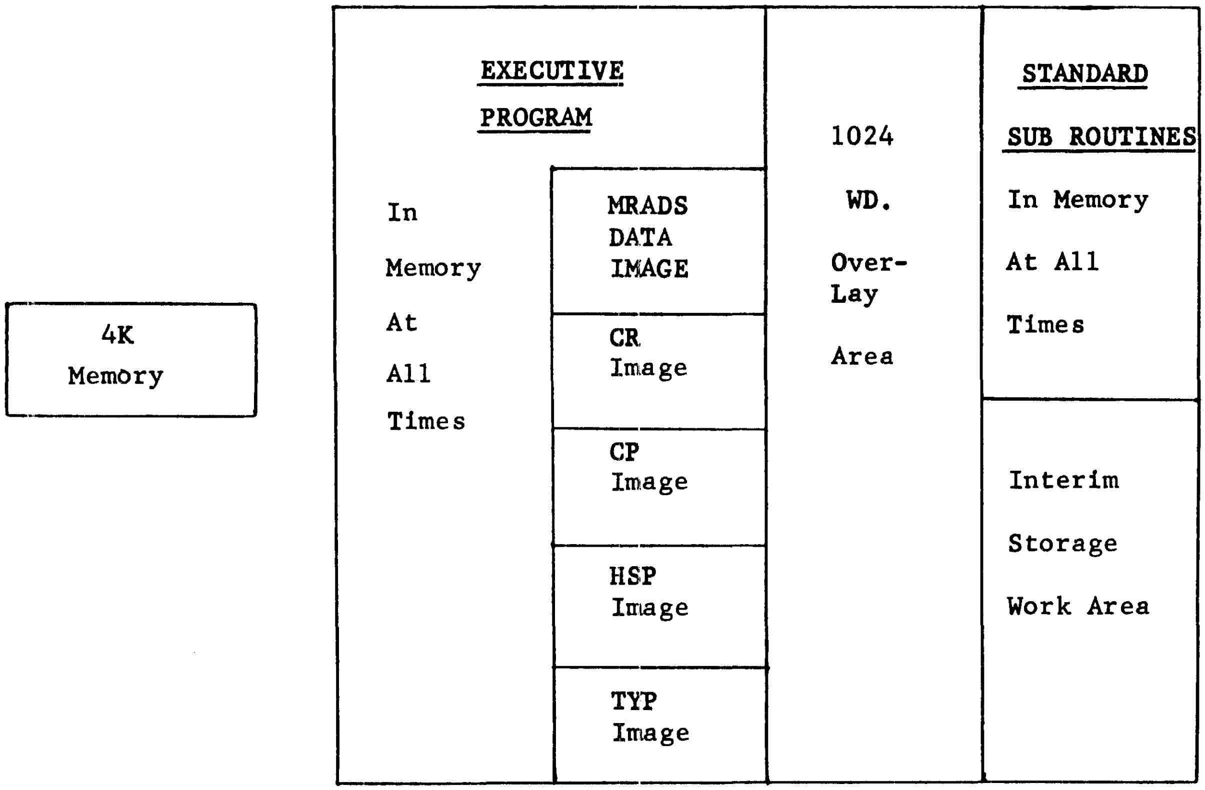



**Exhibit 8**

might cause subsequent programs to be brought into memory to continue processing, or, if the job was completed in one overlay, return control back to the executive program (after the MRADS record has been written back, appropriate cards punched, etc.) **Exhibit 9** shows a possible overlay processing for normal issues. Note that the executive program reads a card and immediately calls in the basic audit program. If the card is not valid the error routine, within the basic audit program, prepares error output and returns control to the executive program, which in turn reads a new card.

If the card initially read was found to be valid, the basic audit program calls in the appropriate program from MRADS and passes control to the overlay area. Note when the issue program is being processed that if back orders are required, the issues overlay program calls in the back order program, etc. A major problem was the coordination between programs and programmers when transferring from one major program to another, such as from "issues" to "back orders". Control, after all work has been performed, is always returned to the executive program.

In the Charleston AFB supply system there are 46 individual programs with an average of 3,000 instructions each. This means there were approximately 140,000 instructions required to handle this real time supply system. This does not include other programs required to process data generated during the real time processing.

With such a small memory (4K) to accomplish a MILSTRIP supply application there was, of necessity, a great amount of duplication of pro-

## SUPPLY INSTALLATION
## EXECUTIVE OVERLAY LOGIC

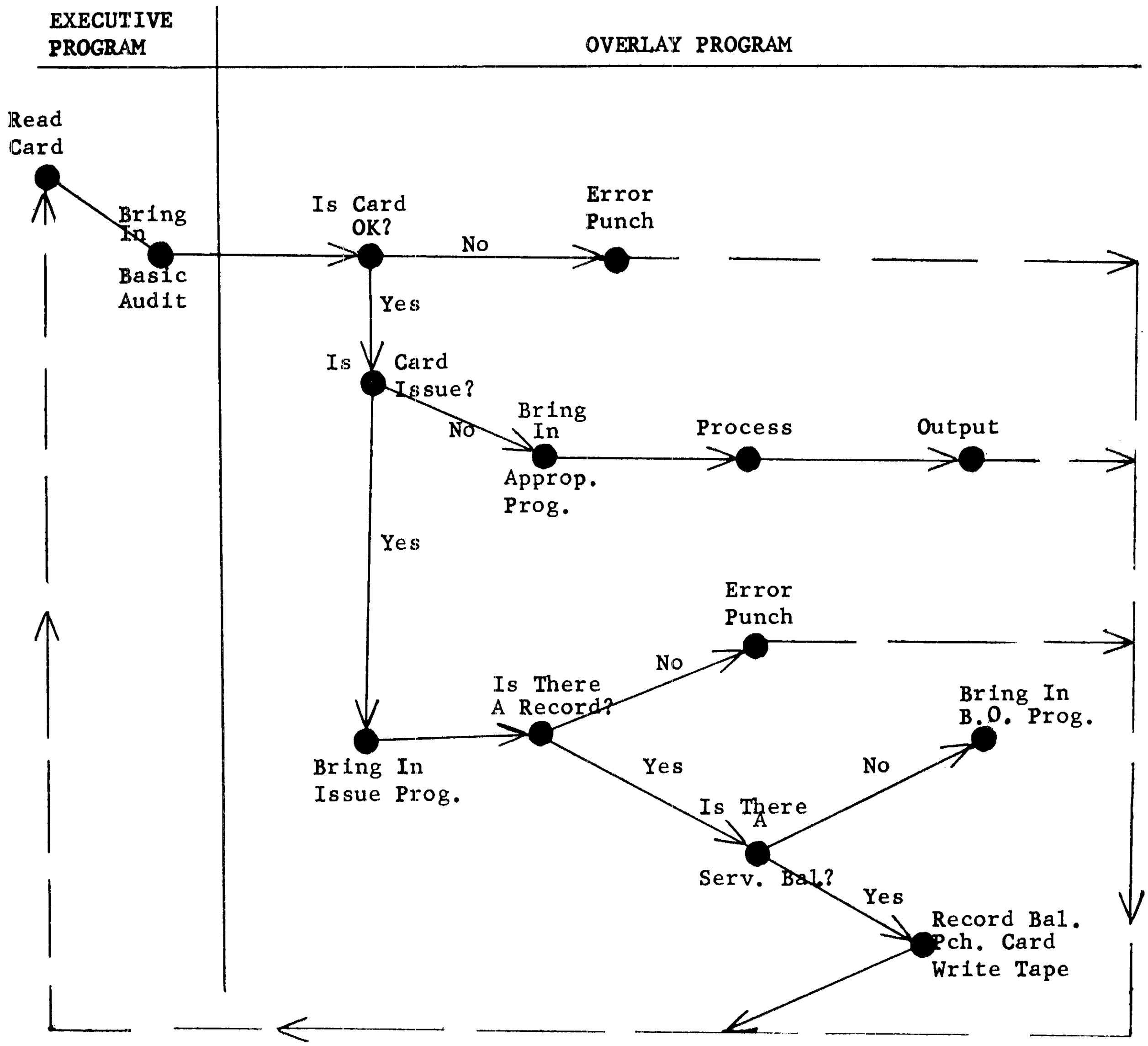



**Exhibit 9**

gramming effort. Many programs had their own error punch routines, preparation and writing of transaction tape routines, linkage error card routines, and special level review and index card routines. With a larger memory many of these routines would have been included as part of the executive program, and not duplicated in many programs. The number of overlays required to complete an average transaction varies between three and five using a 4K memory. Overall processing time would, of course, be reduced if a larger memory were utilized.

It is conceivable during MRADS processing that a series of MRADS records may be changed or updated and written back onto the MRADS prior to reaching a point in processing where an invalid condition is detected; the condition being severe enough to require that all records previously changed be "restored" to their original content and the transaction "rejected". The technique used

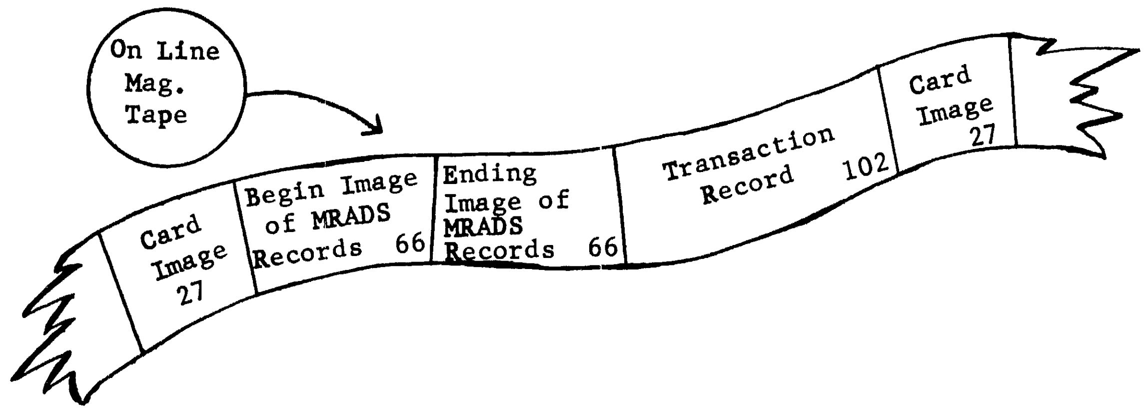



**Exhibit 10. RESTORE — TRANSACTION TAPE**

for accomplishing restores was to write a tape image each time a card was read, a record was read from MRADS, a record was written on MRADS, and for each transaction output (see **Exhibit 10**).

If MRADS image restore is then required, the tape is backspaced and the beginning balance image for this series of updating written on to the MRADS, thus restoring the MRADS record to its original content prior to transaction processing. You will note that each MRADS record has 66 words instead of the sector length of 64 words. Two words have been added to assist in this restore. The 65th word is the MRADS address of this record and the 66th word is either zeros or all "1's" to indicate whether the record is the original record (a read) or the updated record (a write).

This restore tape is used for two major purposes over and above restore capability:

1. The transaction records on tape (priced) can be used to feed tape oriented accounting routines (FIA), and
2. The tape itself can be saved for a period of time to serve as a recovery tool if a MRADS failure is detected.

When complex federal stock numbers of 15-17 alphanumeric characters are reduced to a random number, it is quite normal to find that 20 percent to 35 percent of the stock numbers result in duplicate random numbers. One of the techniques used to ease this problem is to provide a MRADS area larger than that required to hold all data. The larger the excess area provided, the less likelihood of duplication and more likelihood that free space — in case of duplication — can be swiftly found by a programming technique. Duplicate stock numbers are normally loaded into one of the free locations provided, then linked back to the prime location. Open locations can be found by a forward search program that continually advances one record at a time looking for unused record space or by a table technique. One of the problems encountered in the preceding approach is that a duplicate stock number may be placed in a location subsequently belonging to a prime number, thus requiring that the prime number be displaced into a free location and cross-linked. A prime number in this instance refers to a stock number that would randomize to a specific location rather than one that is forced into a specific location due to its duplicate relationship.

As a compromise and to partially alleviate the problem of duplicate stock numbers being loaded into prime locations, an overflow area for duplicates was established (see **Exhibit 11**). Under this system — assuming 60,000 stock numbers are to be loaded with estimated 30 percent duplicates (18,000) — the MRADS file was divided into two areas. The first area, item masters, provides sufficient space to store 80,000 records, and the second area, called overflow, incorporates space for 9,000 duplicate items. As duplicates are found the item master is loaded into the next sequentially available open location in the overflow area. A counter is maintained in the executive program for assigning these sequential locations. The prime location is then cross-referenced to the overflow item by placing the address of the overflow item in the "Go to" portion of the prime record.

It is apparent that 9,000 overflow locations did not handle the 18,000 duplicate items. After the overflow area was filled the program goes into a forward search program loop that continually ad-

**SUPPLY INSTALLATION**
**ITEM MASTER LOADING TECHNIQUE**

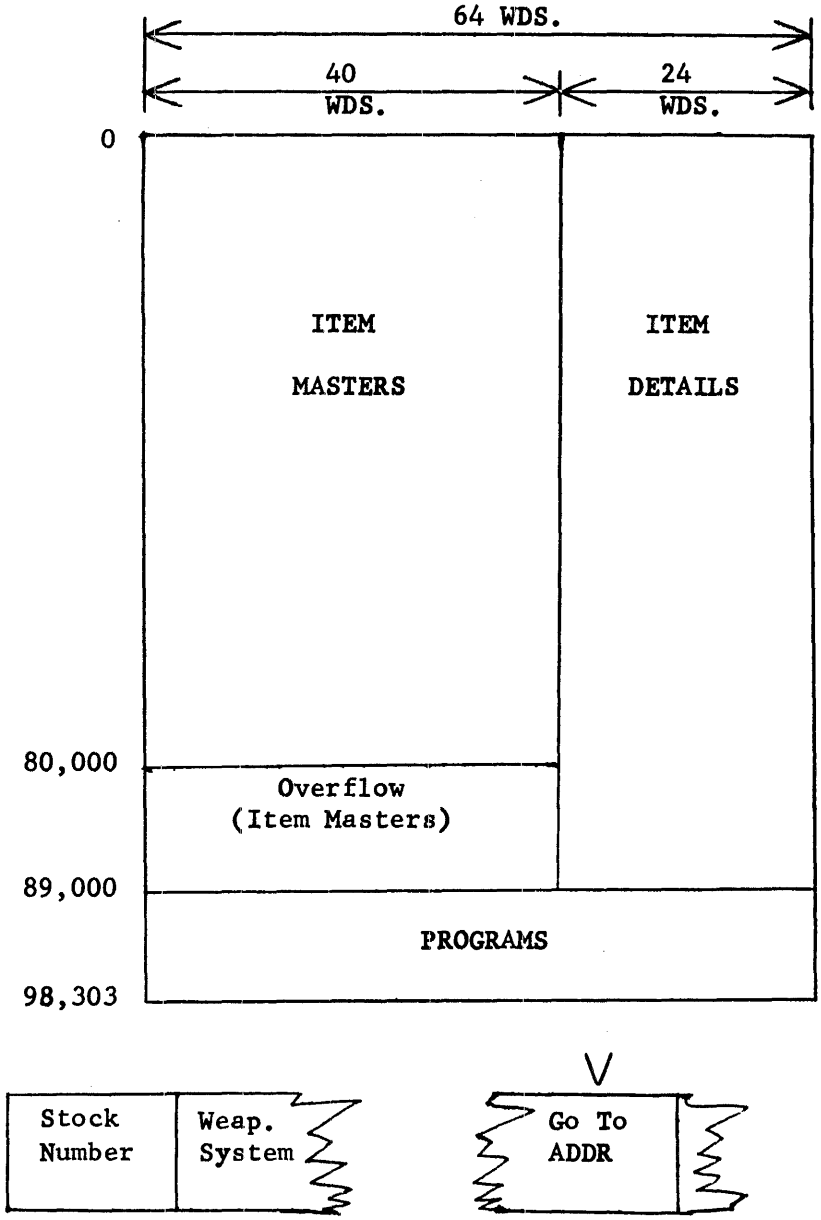



**Exhibit 11**

vances one record at a time looking for unused record space. **Exhibit 12** is a general flow chart of the storing item master routine. The flow chart starts with a randomized stock number MRADS address that has been generated by the executive program and given to the overlay program for processing. The overlay program determines if the MRADS address is available. If available the item is loaded. If not available the overlay program requests an available MRADS address from the executive program. The executive program maintains control until an available MRADS sector is located.

Periodically (presently planned for quarterly) the MRADS item master area will be repacked, using a program called FILEPAC, which will:

1. Move overflow items into vacated prime areas, and
2. Repack and cross-reference the overflow area so that open locations may be quickly located.

Periodically, as required, the item detail areas, except due in, will be repacked also using the FILEPAC program. Estimated time for repacking is two hours. The due in area has a continuous repacking routine as part of the system and never requires repacking.

Experience has disclosed that the techniques covered to this point permit normal MILSTRIP processing on the GE-225 at an average rate of 3.9 seconds per transaction, based on an average daily transaction volume of 4,500, and indications are this time can be significantly reduced. During the GSA acceptance test recently held at Charleston AFB the computer configuration was operated 408.97 hours with only 5.33 hours down time—an availability of 98.69 percent.

## SUPPLY INSTALLATION
## STORING ITEM MASTER ROUTINE

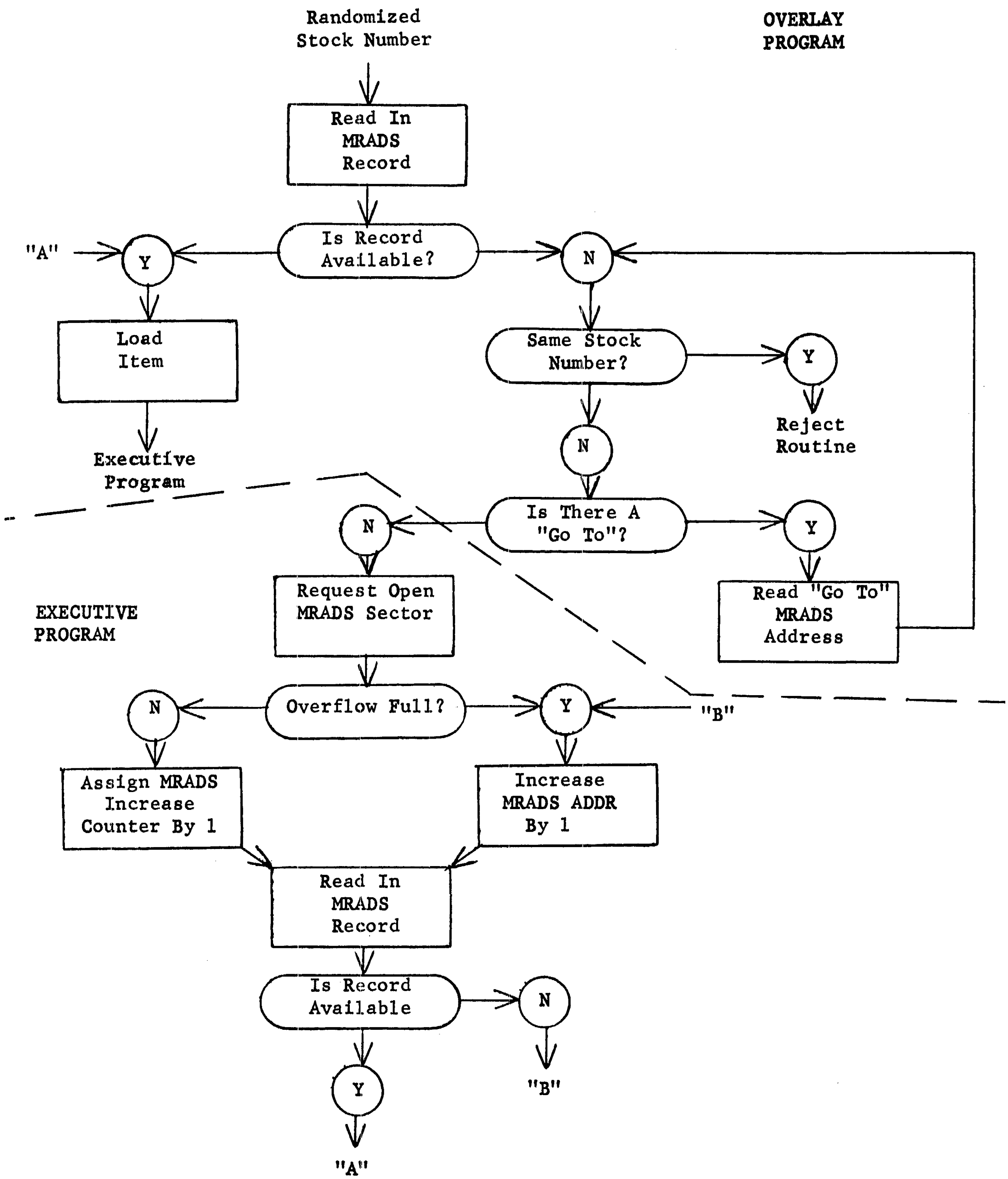



Exhibit 12

by W. R. Hoover and C. A. Seafeldt
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

# Disc File Applications in a Real Time Multi-Computer System

THE CALIFORNIA INSTITUTE OF TECHNOLOGY – Jet Propulsion Laboratory (JPL) is responsible for National Aeronautics and Space Administration projects for the unmanned scientific exploration of the moon, the planets, and interplanetary space. The use of unmanned spacecraft has resulted in a need for rapid processing, display and analysis of data from the spacecraft and from the tracking equipment in order that ground control, using transmitted commands to the spacecraft, may be accomplished. Commands to the spacecraft are for instructions for midcourse and terminal maneuvers; instructions for telemetry and mode control; and instructions for deployment and activation of spacecraft subsystems.

The general data acquisition, processing, and display problem for space flight support is represented in **Figure 1**. The data handling problem is logically divided into four phases:

1. Spacecraft to Deep Space Instrumentation Facility (DSIF) tracking sites
2. DSIF on site processing
3. DSIF to JPL communications
4. Central acquisition, processing, and display.

The central processing facility provides information to the engineering, flight path and scientific analysis groups and to the flight operations director. A centralized command and control system approach has been established for the execution of space flight operations at JPL. An integrated facility is now under construction at JPL, called Space Flight Operations Facility (SFOF), and a multi-computer system providing on line acquisition processing, display and control is being developed for data processing support.

The data processing hardware configuration under development is indicated in **Figure 2**. The criteria which governed the design of this system were:

1. Provide automatic on line acquisition and conversion of all data for automatic processing.
2. Provide a form of parallel processing of analysis programs. This processing to be at the millisecond level for restricted classes of computation, such as conversion to engineering units and display, and at the 15-minute level for the major analysis programs such as orbit determination, midcourse guidance analysis, science analysis, and engineering systems analysis.
3. Provide automatic on line alarm monitoring and distribution of both raw data and reduced data in three analysis centers and at a central operations center. The system must have the capability to automatically retrieve past telemetry and tracking data.
4. Provide capability for distribution of selected parameters to a central status display.
5. Provide processing capability for the complete non-real time processing of all data associated with the spacecraft test program and flight.

The multi computer system shown in **Figure 2** has a large amount of flexibility to meet the criteria. The IBM 7040 computer will act as an input/output processor responsible for:

1. The on line acquisition of all data directly from the communications lines and from the telemetry processing station
2. The conversion of data to engineering units
3. The transfer of data to disc storage
4. Alarm monitoring on incoming telemetry data
5. Quick look output of raw and converted data
6. Distribution of data to the analysis centers and to the DSIF sites.

*(Continued on page 102)*

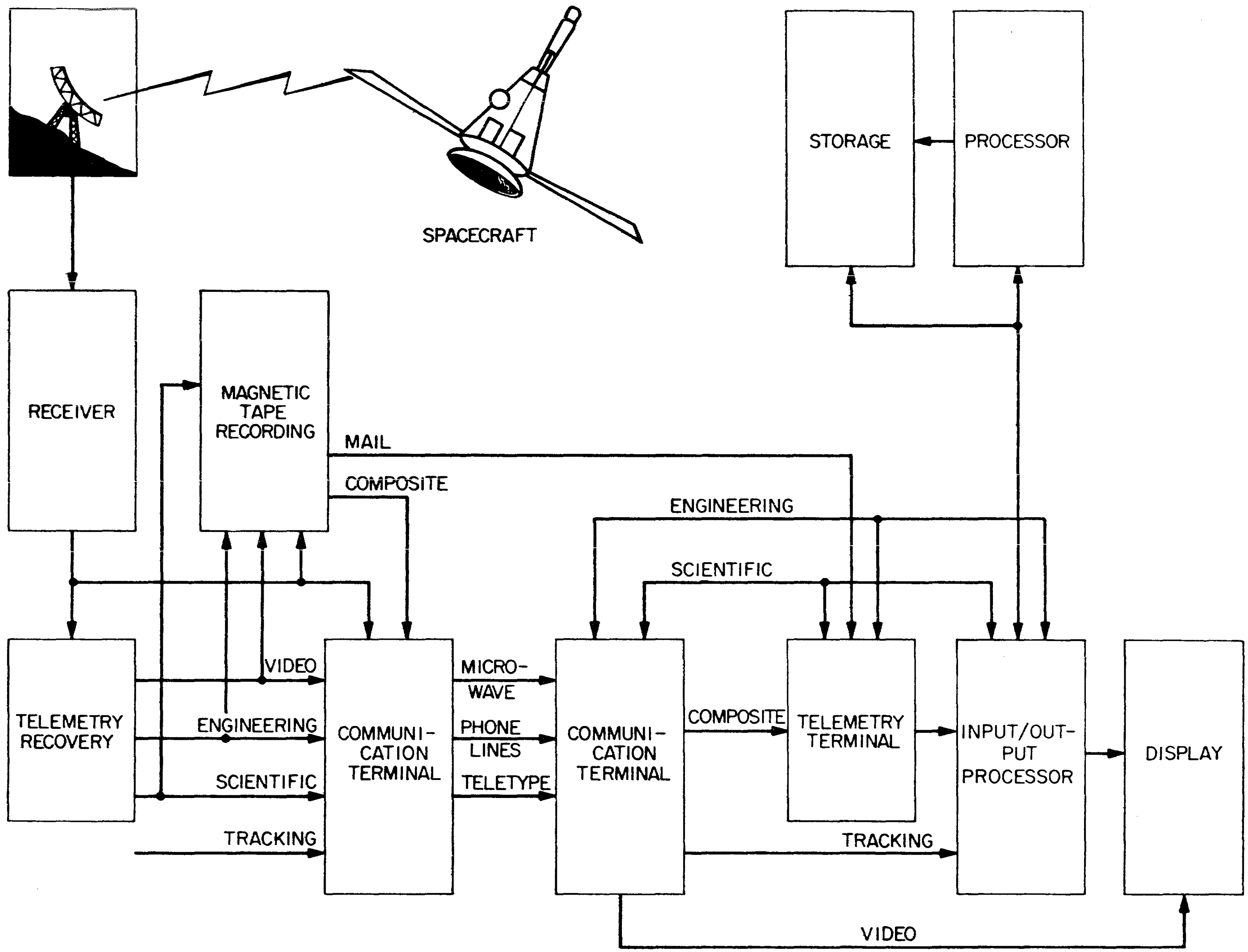



Figure 1. SPACECRAFT DATA SYSTEM

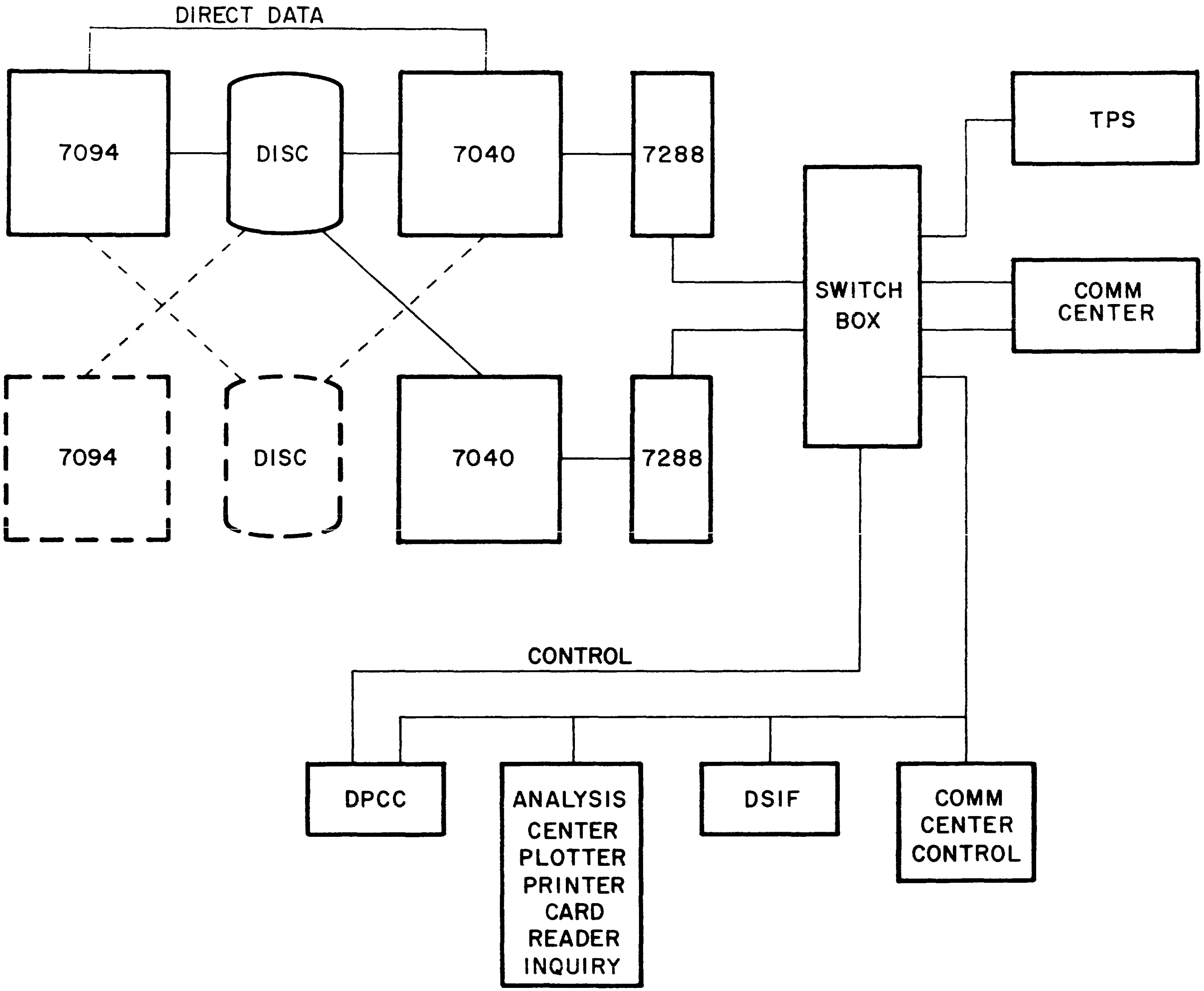



Figure 2. HARDWARE SYSTEM

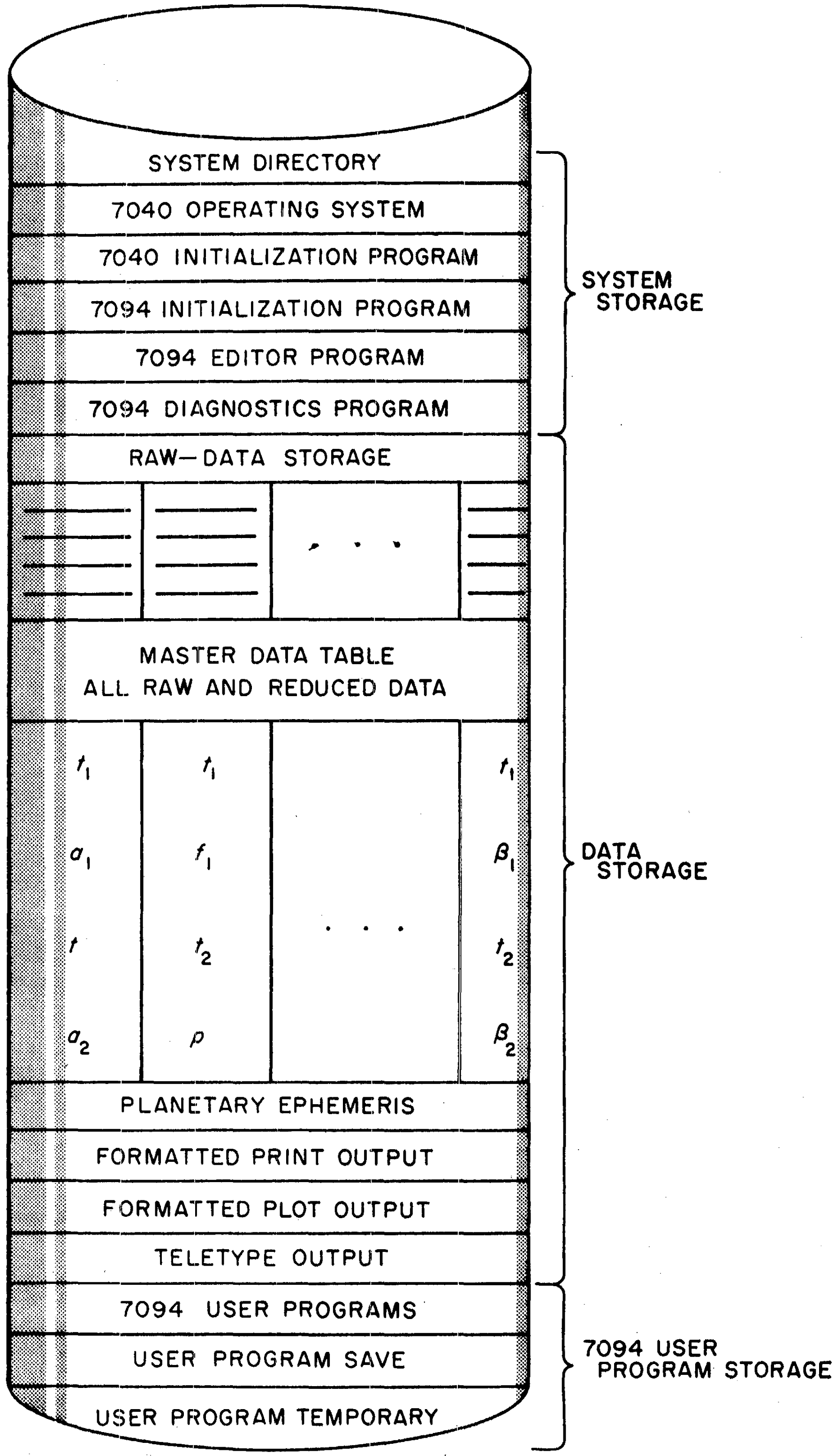



**Figure 3. DISC FILE LAYOUT**

The IBM 7094 computer, under control of an executive monitor, will perform the computation for orbit determination, science analysis, guidance analysis, etc. The 7094 will return its output to disc so that the 7040 may distribute it to the appropriate area. The 7094 will also write output on magnetic tape for off line tabulation and plotting using IBM 1401 systems and a G.D.E. 4020 microfilm printer plotter.

The remainder of this paper will emphasize the function of the disc file system, the disc control programming system, and the layout of the disc file storage. A brief description of the general programming system will be included.

## FUNCTION AND ORGANIZATION OF DISC FILE SYSTEM

The disc file system consists of two 1301-II disc units with a total storage capacity of 112 million six-bit characters, a 7904 data channel on the 7040, a 7909 data channel on the 7094, and a 7631-II file control capable of connecting to either the on line 7040 or the 7094 under program control, and capable of switching between 7040's under manual control. The 1301 units are standard single arm units. Consideration was given for a special order for dual arms but a simulation verified that this would be unnecessary for our problem. For the purpose of this paper we will include a direct data connection between the 7904 on the 7040 and a 7607 data channel on the 7094 as part of the disc system.

The disc file system has six functions: storage of raw telemetry and tracking data for 7094 processing; storage of 7094 outputs for distribution by the 7040; scratch storage for 7094 processing; 7040 and 7094 program storage; lunar and planetary ephemerides storage; and storage of the disc file directory. The disc storage is capable of storing all data during a single tracking station pass for missions in the planning stage. The direct data connection is used to transmit control information between computers, such as card images, 7040 priority request for disc access, priority control information for the 7094, etc.

The disc storage is organized in different ways, depending on how the data are to be accessed. **Figure** 3 shows the types of data kept on disc.

1. The *Master Data Table* (MDT) contains all the raw data measurements that have come into the system, sorted into files by type of data. The MDT consists of many separate data files. Each of the data files represents a unique type of raw data. These data files contain the raw measurements paired with a time value. Associated with each of these individual data files is a directory track. This directory track allows the computers to access any specified time interval of data for any specified data type.

2. The *Raw Data Files* contain the raw data as written by the 7040 onto disc. There is a telemetry raw data file and a tracking raw data file. These are circular files and are accessed in a sequential mode. There are two pointers in the disc dictionary which indicate to the 7040 the location it should write the next record, and to the 7094 the location from where it should read the next record.

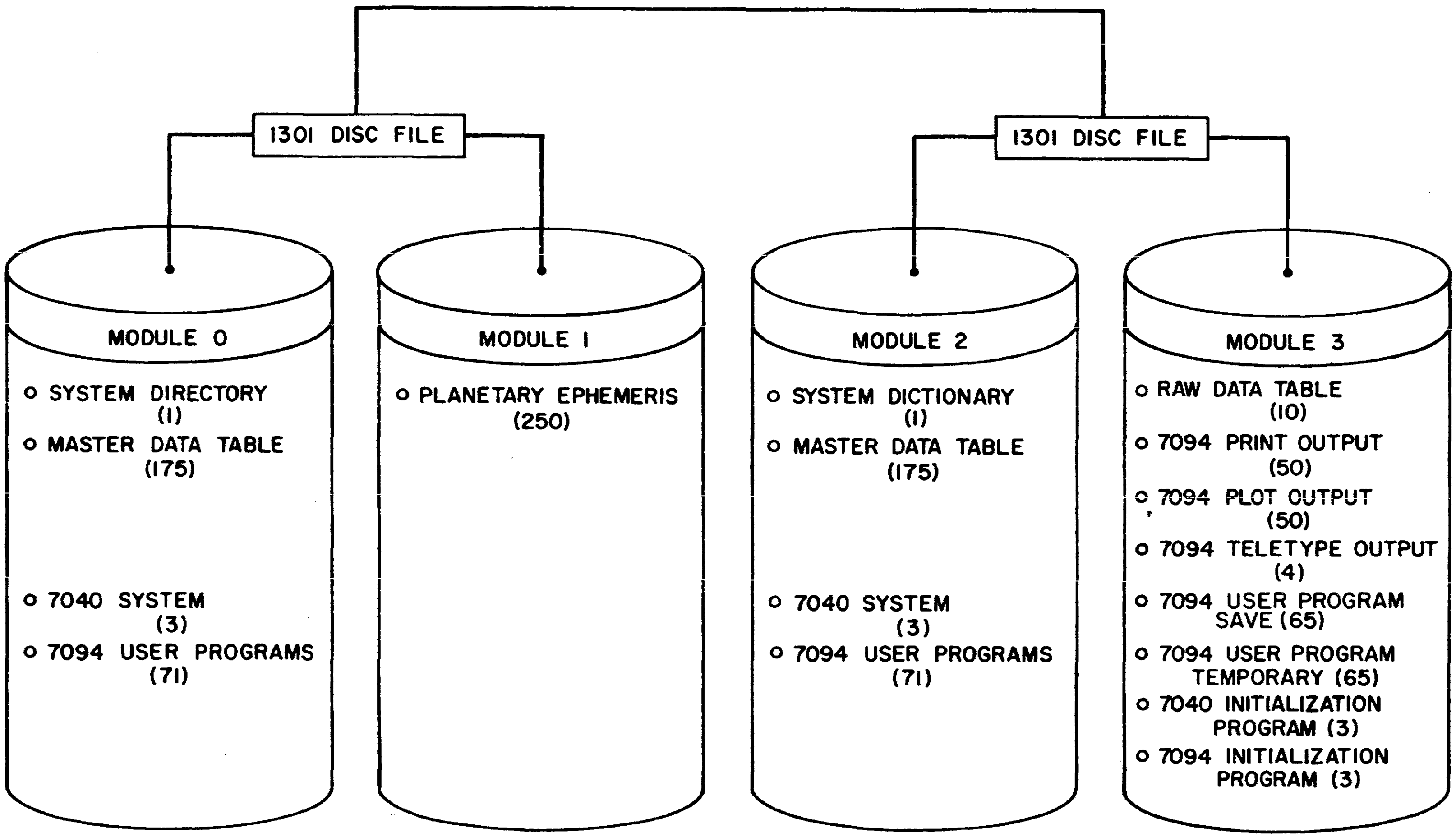
The 7094 reads the raw data and sorts it into the MDT under control of the editor program.

3. *Scratch* storage is provided for the 7094 analysis programs. Each analysis program will have available a fixed number of blocks. These blocks will be numbered in sequential fashion. The 7094 program reads and writes scratch disc by referencing the desired block number.

4. *Formatted 7094 Output* consists of the fully formatted plots, prints, and teletype outputs which the 7094 generates and stores on disc for 7040 output processing. These outputs are identified by symbolic names and the 7040 is notified of their availability on disc. Depending upon previous requests from the remote areas and on initialization parameters, the 7040 will perform various functions upon indication that 7094 output is on disc, which include notification to remote areas of display availability, queuing up of display on various output devices, requesting verification for teletype transmission to the DSIF, etc.

There are in addition to these types of disc storage several types which represent straightforward reading and writing of absolute files. The *Operating System, Initialization Programs,* and *Editor* represent the programs necessary to initialize the system, run the system, and restart the system. The *User Programs* are the various analysis programs which operate under the system. The user programs are stored in both their primary form and their interrupted form and are saved in the *Program Save* area.

**Figure** 4 shows the actual breakdown of the disc storage onto the respective modules. The numbers in brackets indicate the number of cylinders delegated to each type of disc storage. Modules 0 and 2 are identical. This data is considered critical and is duplexed onto both 1301's.




MODULE ASSIGNMENT MADE ON BASIS OF:

- OPTIMUM DUPLICATION FOR BACKUP
- MAXIMUM RECOVERY CAPABILITY

Figure 4. DISC MODULE ASSIGNMENT

This scheme allows operations to continue if any disc or any module goes out. The operating system can run on two modules since we can stop duplexing data and we can go to tape for the Planetary Ephemeris.

In addition to duplexing, certain items are write checked or not depending upon the degree of reliability required. The necessity for check summing will be determined after the discs have been in use long enough to truly evaluate their reliability. This feature will not be used in the initial operations.

The entire disc will be formatted with two 200-word records per track. This layout was dictated by the overall computer programming system. Core space limitations restrict buffers in the 7040 to half track length. Since it is necessary to access successive records, a dummy space is left at the end of each record, in order that successive records can be accessed without waiting full disc revolution.

The *disc directory* was mentioned above. The actual disc access consists of a dictionary-directory system for the master data table. The dictionary is in core and provides the disc programs with the address of a particular directory. This directory record is then brought in to core. The disc program will then use this directory to locate a particular data type for a particular time period.

Since both computers access the same data, it is obvious that the dictionaries must be updated between computers. This is accomplished by use of the *direct data connection* which is independent of the disc files and allows transfer of information at a 62.5 kc word rate.

The 7040 system must be able to achieve immediate access to disc in order to service the high speed input buffers without loss of data. In case the 7094 has the disc when a high priority requirement exists in the 7040, the 7040 uses the direct data connection to interrupt the 7094 and gain control of the disc. The maximum time delay for the 7040 to achieve control of the disc using this method is 50 microseconds plus the time to complete reading the current record, which is a maximum time of 17 milliseconds.

## PROGRAMMING SYSTEM

The overall programming system (**Figure 5**) logically consists of three different subsystems. Each system operates asynchronously but must be cognizant of the status of the other subsystems. Mutual control between subsystems is required. The three subsystems are: the 7040 system, 7094 system, and the data control system which operates in both the 7040 and the 7094 and is responsible for sequencing and issuing all disc file transactions and all direct data communications. The 7040 and 7094 systems both have timesharing schemes; however, these schemes are completely different due to the difference in the functions performed by the two computers.

### 7040 system

The 7040 system's function is to handle all raw data inputs, time tag the incoming data with SFOF time, record all data on magnetic tape, transmit data to the disc file, distribute output from the raw data stream and from disc to the analysis centers and to the DSIF and control communications from the remote consoles. The 7040 can also convert data to engineering units and perform out of tolerance alarm monitoring on the telemetry data. The 7040 must be able to perform acquisition and monitoring on two missions in parallel. **Figure 6** is a generalized flow diagram of the 7040 system.

The basic structure of the 7040 includes the following types of routines:

1. Trap processors

The function of the trap processor is to process the data channel traps and move data from the trap buffer to a line processor buffer and to a tape output buffer. Trap processors are further divided into input trap processors servicing the input subchannels on the 7288, output trap processors servicing the output subchannels, a tape trap processor, and a disc trap processor.

2. Input processors

Input processors are controlled by the priority control program, PRIO, and are the ordinary processors which perform data identification and separation of teletype data, phone line, microwave line data, card readers, and console inputs. These processors are responsible for operations on data stored in the line processor buffers by the trap processors. These routines also decode requests from the consoles, convert data to engineering units, perform the alarm monitoring and transfer data to sort buffers and to the plot and print buffer pool. The sort routine initiates requests to the disc control program when sort buffers are full and are written onto disc.

3. Output processors

These ordinary processors control the display of information in the analysis centers and the dis-

*(Continued on page **107**)*

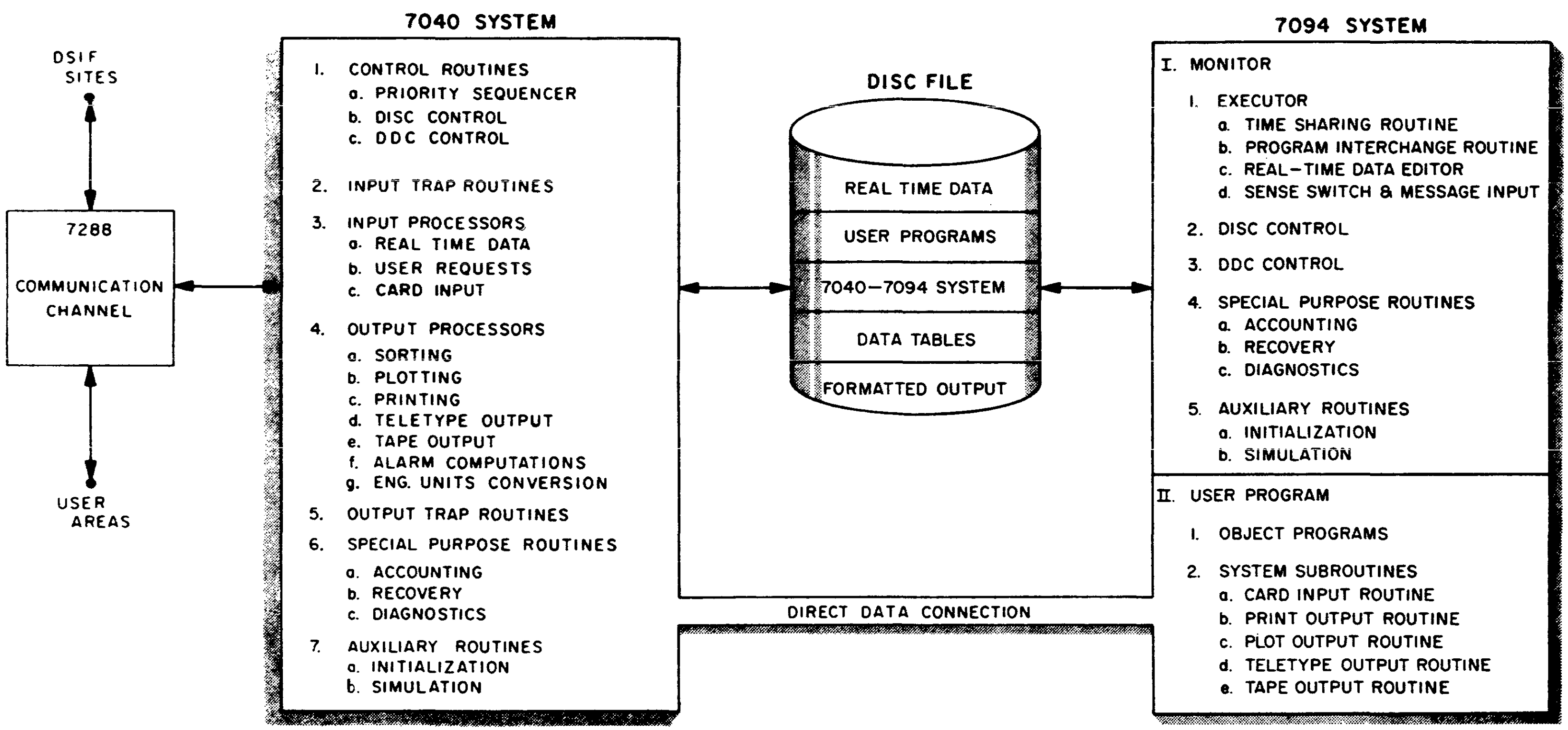



Figure 5. SFOF PROGRAMMING SYSTEM

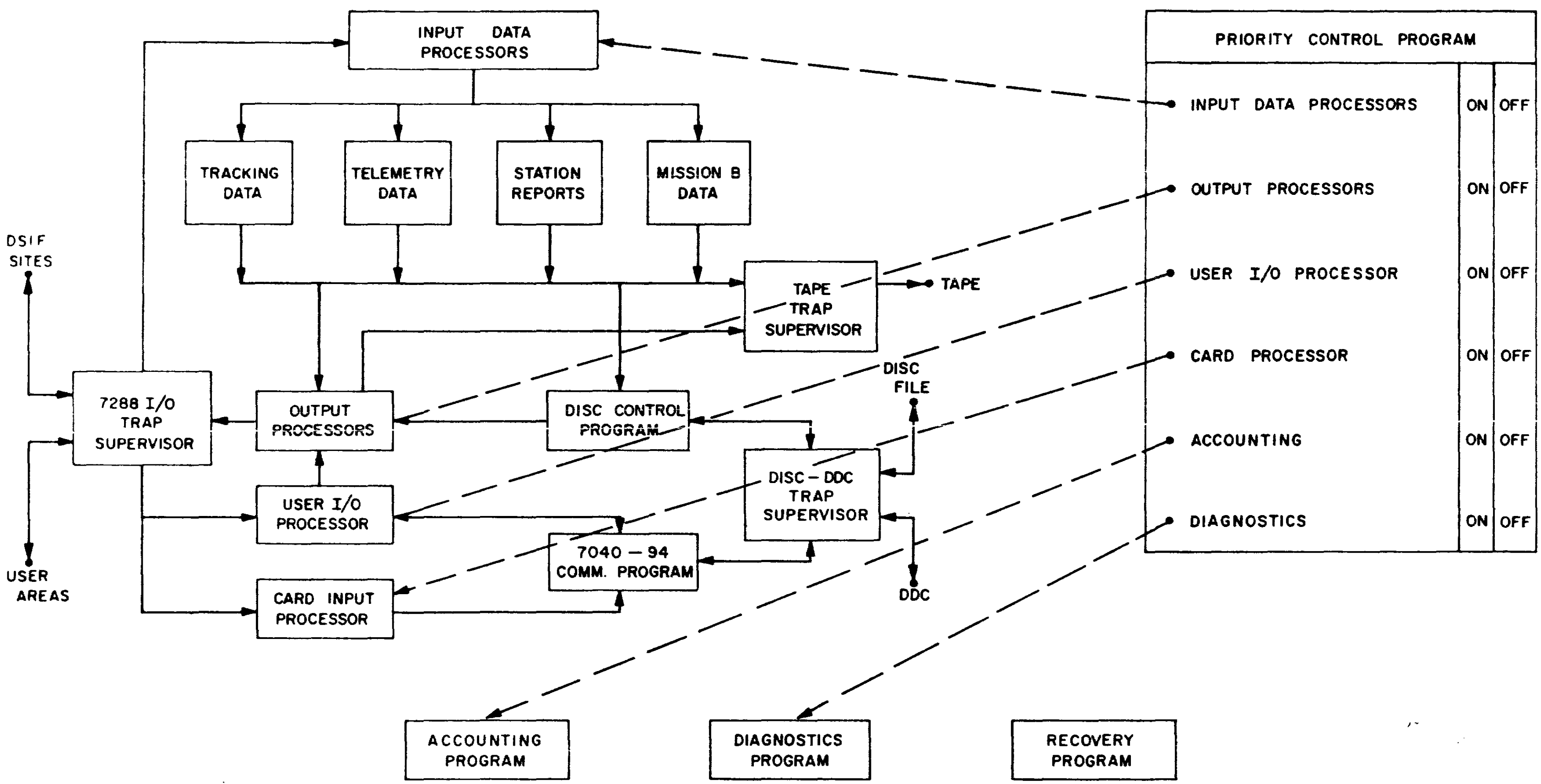



**Figure 6. 7040 PROGRAMMING SYSTEM**

tribution of data to the DSIF sites. Displays are of two types: 7094 generated prints, plots, status, and commands; and display of data from the real time data stream.

4. Priority control

Trap processors are assured of highest priority by the 7040-7288 hardware system. Trap processing will interrupt ordinary processors immediately upon a trap condition. Priority among the ordinary processors is established by a list, PLIST, which governs the operation of PRIO, Priority Control Program. Routines in PRIO may be turned on by a trap processor or by another ordinary processor. A diagnostic routine is at the low priority end of PLIST and will always run when no other routine in PLIST is turned on.

**7094 system**

The 7094 system (**Figure 7**) consists of three basic parts: the monitor, the input/output system, and the analysis programs. The analysis routines include orbit determination, science analysis, guidance analysis, etc., and will not be discussed here.

1. Monitor

The monitor is responsible for the control of analysis program execution with a priority scheme, intra computer communications, editing and sorting of the raw data table to the master data table, status reporting, and accounting and creation of old raw data prints and plots. Since the flight missions divide into logically different operation periods, the priority control must be capable of being modified on line. All modifications of this type are made from the central data processing control console. The 7094 timeshares the analysis programs by keeping a list of the programs which should be sharing time in the 7094. Associated with each program in this list is the percentage of real time which that program should get. An interval timer trap will periodically cause entry to the monitor. The monitor will guarantee the time sharing by interrupting programs, saving them on disc, and starting a new program which needs time.

2. Input/output routines

Input routines are necessary in order that an analysis program may request cards to be read from the appropriate analysis center. Output routines provide the analysis program capability to print and plot in the analysis center and to transfer teletype output to the DSIF.

**Data control program**

Data control programs exist in both the 7040 and 7094. These programs control all disc transactions and all direct data transactions. An overall diagram of the data control program is shown in **Figure 8**.

The *control* programs are referenced by the user to obtain disc transmissions. The primary function of these programs is to interpret the user's request and to make appropriate entries into various queue lists. The *trap supervisors* interpret traps from disc and issue new requests by accessing the queue list entries stocked by the control programs. The communication programs provide the necessary inter computer communication necessary to provide dictionary updating and system control. The *disc routines* are special subroutines used by 7094 trap supervisor and control program.

1. 7040 disc program

**Figure 9** represents a more detailed breakdown of the 7040 disc program.

The Disc Control Program (DCP) is a closed subroutine which is called by the users for the purpose of accessing the disc file units for data reads or data writes. By use of the user's calling sequence and the dictionary, DCP translates the essentially symbolic request into the required sequence of entries for the queue lists. These lists contain information relating to module selection, seek order, record address, and type of transmission.

The DCP Trap Processor (DCPTRP) is responsible for keeping disc I/O going continuously by servicing traps and the requests queued by DCP. It is also responsible for direct data connection activity.

When a trap occurs on the disc channel, DCPTRP examines the flag bits to determine the cause of the trap. If necessary, the sense lines are then sampled for additional information. If the channel is active, and 7040 is using disc, nothing is attempted with the exception of issuing any possible SEEK activity on inactive modules. If the channel is inactive, and any satisfied seeks are queued, data transfer activity is initiated. In the event of an unusual end signal, the cause is determined, and appropriate action is taken. This may involve retrying a read and write, trying a write in an alternate region, etc.

*(Continued on page 110)*

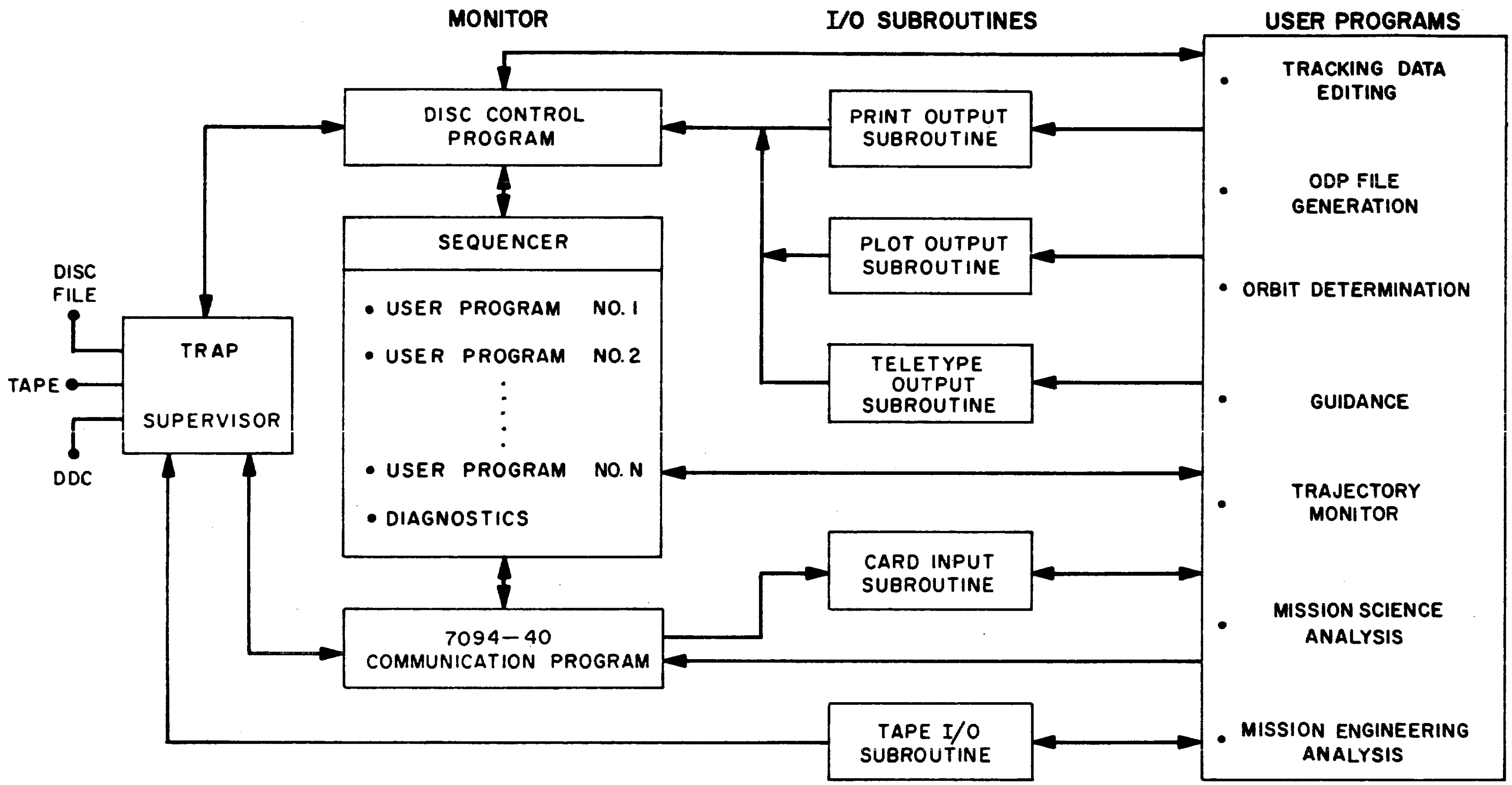



Figure 7. 7094 PROGRAMMING SYSTEM

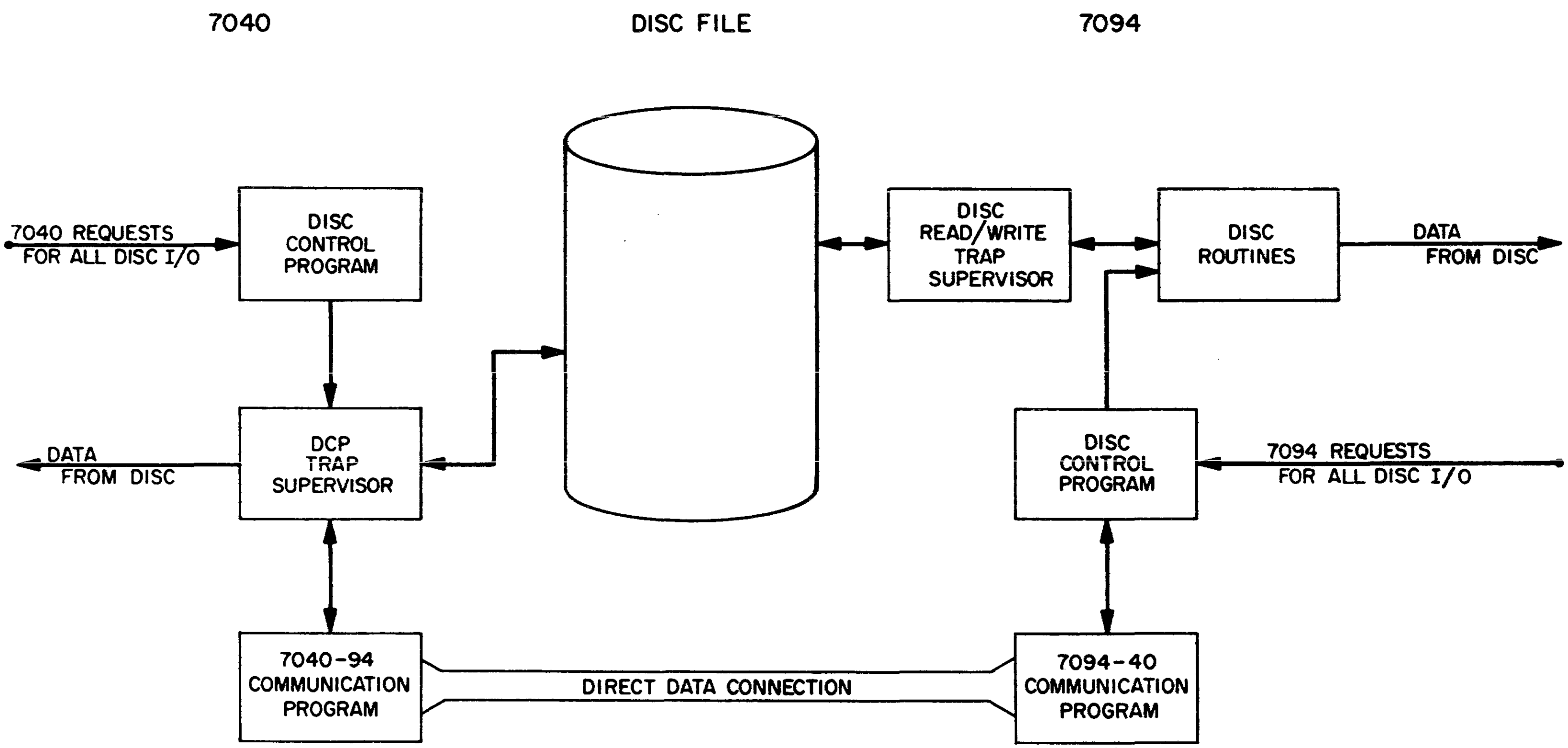



Figure 8. DISC PROGRAMMING SYSTEM

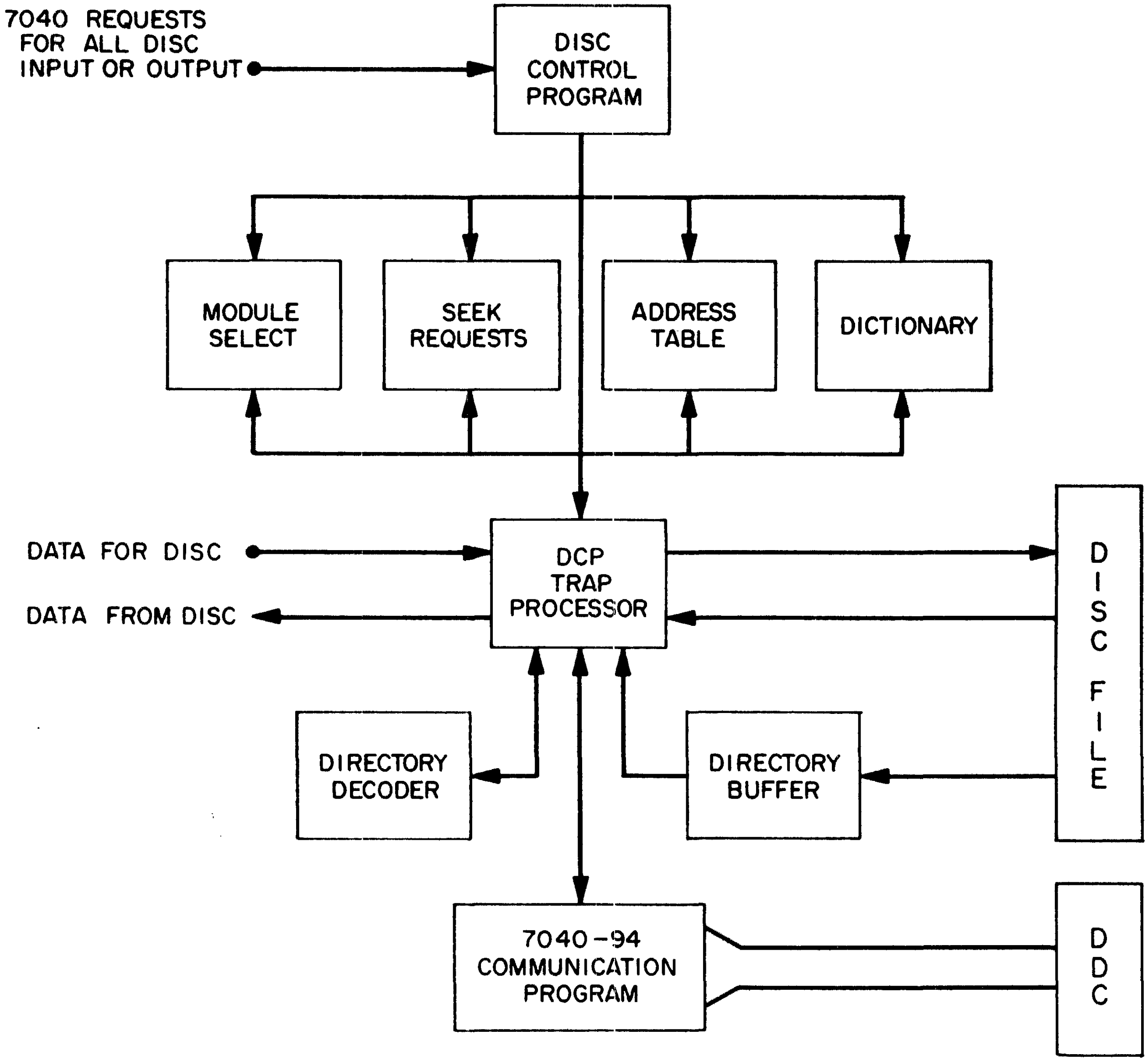



**Figure 9. 7040 DISC PROGRAM**

If it is necessary to use a directory track to find data, the directory decoder is used by DCPTRP to insert the necessary entries into the queue lists.

2. 7094 disc program

**Figure 10** represents a more detailed breakdown of the 7094 disc program. This system varies somewhat from the 7040 system because:

1. There are more types of disc I/O in the 7094, and
2. It was desired to make use of two existing routines.

The Disc Read/Write Program (DSKRW) is a standard JPL disc routine and the trap supervision is done under IOEX.

The Disc Control Program translates requests which are essentially symbolic into the form required by DSKRW by use of the dictionary.

If the disc reference requires no directory reference, the request is sent to DSKRW. DSKRW stacks the requests into queue lists and returns control to the user. If the disc reference requires a directory track, DCP issues a request to DSKRW for the directory track and returns control to the users. Upon completion of the read of the directory track DSKRW turns control over to the disc routines.

There are four disc routines shown in Figure 10. Master Data Table I/O Disc Routine, and

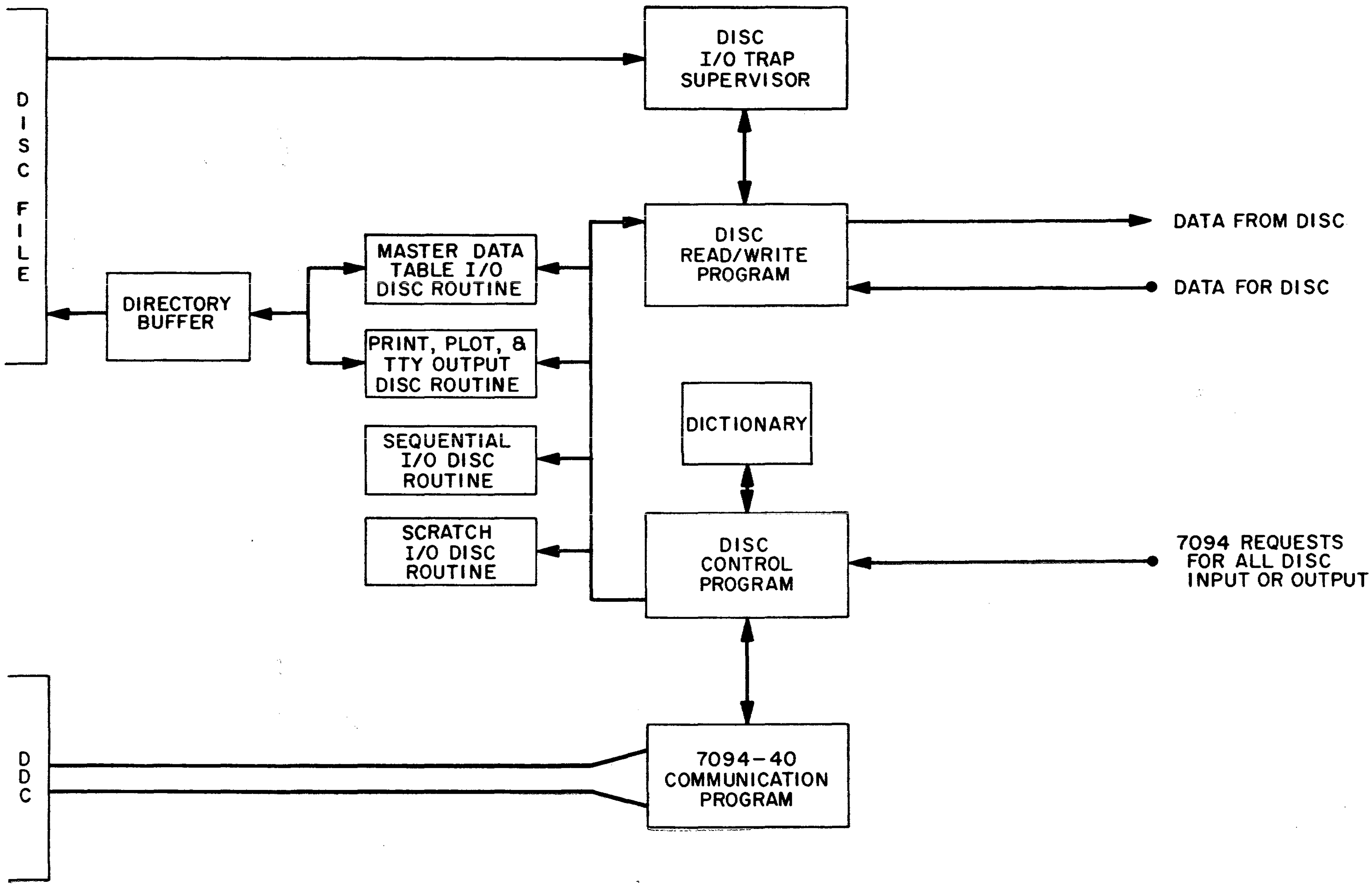



Figure 10. 7094 DISC PROGRAM

the Print, Plot, TTY Output Disc Routine interpret the directory track and make requests to DSKRW; these two disc routines and the sequential/scratch disc are all entered by DSKRW when a transmission is complete and have the responsibility to set a flag word for the user.

In summary, the disc read/write routine provides the following capabilities.

1. Receives requests for disc operations from DCP
2. Forms request queues and stacks them for efficient service
3. Allows priorities to be assigned to each request
4. Provides a method for random disc access
5. Provides necessary communications with IOEX
6. Can use closed subroutines (disc routines) to process disc data at the time it becomes available in core.

The SFOF was scheduled for flight support on January 1, 1964. The 7040-Disc-7094 system with two analysis centers became operational in July of 1963, with a complete programming system. This system will be used to verify the design concept, conduct familiarization exercises and to allow operational checkout of the hardware and programming.

The programming system design was a joint JPL-Computer Sciences Corporation effort and the programming system implementation, exclusive of the analysis programs, is being carried out by Computer Sciences Corporation. The hardware system was a joint JPL-IBM design.

by E. D. Reilly, Jr.
Knolls Atomic Power Laboratory*

# A Disc Oriented Remote Inquiry Philco 2000 System

MOST PRESENT DAY COMPUTING CENTERS make highly efficient use of computing time itself, but very poor use of the time of the humans who communicate with the machine. The days when programmers were allowed to sit at the console and think their way through their program bugs may have been wasteful of machine time, but this method insured that the longest part of a programming job was the *writing* of the program, not the debugging, as is presently the case in many installations.

The tendency toward off line card-to-tape and tape-to-printer facilities tends to create first shift turnaround times of an hour at best; this goal is difficult to achieve if an installation cannot process all of its lengthy jobs (10 minutes or more) on second and third shifts. With the continuing arrival of faster and faster machines, the disparity between problem execution time and total turnaround time is becoming worse.

The solution proposed by several authors[1,2] is to decentralize operations to some extent by establishing remote stations from which short jobs can be processed. The technical requirements for such a system for use with the KAPL Philco 2000-212 that are being considered are:

1. The system shall consist initially of 8-10 stations, but should be expandable.
2. Each station shall consist of a combination typewriter-punch unit such as a Friden Flexowriter or KSR communications device.
3. The furthest station may be up to 25 miles from the central computer.
4. Some type of scanning device shall scan the remote stations in cyclic order. The scanner shall have sufficient buffer storage and fast enough logic to process messages from up to 10 stations simultaneously.
5. The scanner shall be capable of communicating information to and from the Philco 212 at a rate not less than 60 kc.
6. A disc storage unit shall be coupled to the scanner in such a way that messages can be transmitted between the disc and remote stations without control of the Philco 212.
7. The disc shall have a storage capacity equivalent to at least two million 48-bit Philco 2000 words.
8. The disc shall have sufficient read/write arms to enable the simultaneous processing of data between 10 different stations and their assigned portions of the disc.

The principal uses of the disc would be:

1. To hold an input queue of small problems which accumulate while the 212 is busy with a large problem
2. To hold an output queue of problem answers which are received from the 212 at 60 kc, but which must then be disbursed through the remote stations at 10 characters per second after the 212 disconnects from the scanner
3. To hold a library of frequently used programs and subroutines in order to minimize file tape mounting at the central computer.

## IMPLEMENTATION

At least one means of implementing such a "Round Robin" system is commercially available. All of the requirements outlined above can be met by using a GE Datanet 30 Communications Device in conjunction with the GE RAM memory that is presently being supplied as part of the GE-225 product line.

The RAM has 16 discs, each with its own read/write arm, so that the desired remote station buf-

* Operated for the United States Atomic Energy Commission by the General Electric Company, Schenectady, New York.

[1] Adams, C. W., "Cottage Computing", *Datamation*, Oct. 1961, p. 53.

[2] Bright, H. S., and B. F. Cheydleur, "On the Reduction of Turn-Around Time", AFIPS Conference Proceedings, Vol. 22, Dec. 1962, p. 161.

fering could be achieved. Each disc contains the equivalent of approximately 140,000 Philco 212 words.

The Datanet has a memory of its own, equivalent to approximately 3K Philco 2000 words, together with microsecond logic that is capable of performing the scanning and message assembly functions.

Although a real time interrupt device is available as part of the Philco 2000 product line, its monthly rental is equal to that of the Datanet. Necessity for such a device can be avoided if the large programs being run at the central station are provided with programmed interrupt points no longer than, say, five minutes apart.

Since all 212 problems are run within the BKS operator system,[3] there is a natural break-in point between system jobs where the contents of memory would not need to be saved and restored in order to process a Round Robin job. Since most BKS jobs are less than five minutes in duration, only a few Laboratory programs would have to be modified to provide break-in points.

## SYSTEM ADVANTAGES

The remainder of this paper is a discussion of the advantages of the Round Robin system for particular types of problems.

### Advantages for sequential design problems

Long turnaround time is indirectly wasteful of great amounts of machine time. Knowing that he will get only one shot per hour (at the very best), an engineer, confronted with the necessity of investigating the effect of several variables on his design, will deluge the machine with a large number of problems that will map out a function space so large that it is bound (he hopes) to encompass the optimum design he is seeking. With two to 10 minute turnaround, however, he would be content to proceed in a more conservative manner, changing a variable or two at a time and observing their effects first hand. In addition to reducing his total machine time requirements by an order of magnitude, he gains the invaluable first hand "feel" for his design which is being lost in the brute force approach.

### Debugging advantages

At the beginning of the day, large source decks for programs to be debugged are brought to the central station where they are placed on tape, read into the machine in the usual manner, compiled, and both source and object programs then stored on the disc corresponding to the station from which they will be debugged. (Small programs may be loaded directly from remote stations provided the programmer is willing to prepare a punched tape.) During the day, the programmer then makes debugging shots from the nearest station to him, communicating with an executive program in the Datanet with short messages such as:

Modify the disc image of card number 261 as follows . . .

Insert the following statement after card (image) 47

Delete card (image) 133

Re-compile program 2706, etc.

Voluminous dumps, wasteful of paper, high speed printer time, and personnel time, would no longer be processed through the central station printer but would be stored on the disc. (Each station's disc could hold approximately five programs of 40,000 machine language instructions each—large programs by any standard—and still have room for 10 memory dumps of 12,000 words each.) Programmers would never call for their dumps to be printed but would call for selected portions of them, as needed, *in symbolic language.* This would be accomplished by keeping each program's symbol table on the disc along with the program itself. Note that calling for FLUX (13,27), for example, *does not require access to the Philco 2000;* the Datanet itself would be capable of fetching selected data from the disc-dump and returning it through the appropriate remote station with a delay measured in milliseconds!

The programmer will thus proceed to make periodic debugging shots through a remote station throughout the day. When his program is debugged, he sends a message which says, in effect,

Punch a corrected copy of my debugged program (via the path disc→ Datanet→ Philco 212 →output tape→off line card punch) and return it to me.

### Advantages for users of library data

Consideration will also be given to disc storage of important library data. At KAPL, there exist several programs which use libraries of nuclear cross section data. If duplication were eliminated, one or two 140K discs should be sufficient to

[3] Smith, R. B., and C. H. Hunter, "The BKS System for the Philco 2000 Computer", WAPD-TM-233, April 1961.

contain the data. This would be an extremely worthwhile investment for two reasons:

1. The number of library tapes mounted and dismounted per month at KAPL would be decreased by 200, and
2. An efficient mechanism would exist for insuring consistent use of the same cross-section data for all Laboratory programs.

### Advantages for one-shot problems

Using remote station access, one-shot form sheet calculations of the type normally done by an engineering aide at a desk calculator could be done rapidly and accurately. Furthermore, the occurrence of logarithms, sines, Bessel functions, etc., that would be the cause of tedious and error prone table lookup and interpolation procedures by a human calculator, would not faze the machine in the least.

### Advantages for long production runs

Even with the three-fold or more speed advantage of the 212 over the Philco 2000-211, large reactor problems will still require two hours or more of computer time. Although running such lengthy problems during the day shift would no longer have drastic effects on turnaround time for short problems (because minimum five minute interrupts would be programmed in), most such problems will inevitably be postponed until second or third shift. The advantage of the RR system for such problems would not be decreased turnaround in itself—a 24-hour schedule would prevail at best, but the RR system could help insure getting *no worse* than 24-hour service by providing a mechanism for quick "production checkout" runs whereby problems scheduled for evening running could be input-checked on the machine during the day.

A second advantage would accrue to those who prepare large problems which require hundreds of input cards. Usually, a reactor designer sets up one reactor design or "model" and then proceeds to work with it over a period of days or weeks. During that time he will submit many problems based on this model, each differing with respect to some material property or characteristic operating condition. The present system requires him to spend considerable card shuffling time in the EAM room producing the number of problem deck variants that he needs.

With a disc, a convenient parking place exists for each basic model with which he is working. Short messages through a remote station, analogous to the "delete card", "add card" type commands described under debugging advantages would enable the user to produce all of the variants he needed without touching a card deck or crowding into the data processing room.

## SUMMARY

A system for remote station access to a large scale digital computer has been described which offers advantages for the running of debugging procedures, sequential design problems, form sheet calculations, and problems which reference library data. A specific method was outlined for implementing such a system through the combination of a Philco 2000-212, a GE Datanet 30, and a GE RAM disc file.

by J. L. Atwood, J. E. Volder, and G. N. Yutzi
Collins Radio Company
Newport Beach, California

# Application of a Disc File to Message Switching Systems

WE WILL DESCRIBE the hardware and software of a Collins C-8000 electronic message switching center with special emphasis on areas relating to the use of disc file. This center is primarily a store-and-forward teletype switching terminal utilizing data processing techniques. Installations with a majority of these features implemented are presently in operation.

The following tabulation of functional requirements for this switching center give an insight into the scope of hardware and software required:

1. Transmit and receive four standard speeds of serial mark-space teletype on 200 duplex lines.
2. Maintain supervisory control over all end devices.
3. Transmit and receive to two high speed data channels.
4. Monitor all lines for malfunctioning or improper operating procedures.
5. Accept all incoming messages regardless of the size of output line queues.
6. Edit incoming messages and determine destination, routing and priority. Queue output traffic according to priority and time of receipt.
7. Service subscriber requests for retransmission of any message less than 24 hours old with no "noticeable delay". Service subscriber requests for retransmission of any message less than 30 days old.
8. Provide a high degree of "message assurance".
9. Be easily adaptable to changing system requirements.

Disc files provide the necessary storage for message queueing, rollback and fast recall. Magnetic tapes provide the necessary storage for slow recall. Collins C-8401 processors execute the necessary control and processing required by the overall communications complex.

## RELIABILITY CONSIDERATIONS

Although teletype transmissions may be characterized by a high error rate, this error rate is tolerated in this system and in other person-to-person message switching systems because of the high redundancies in language structure. The important criterion for this system is a high degree of message assurance. It is most important that there be a high probability that the intent of a message be correctly interpreted by the intended addressee within a reasonable time period. Complete system down time (scheduled or otherwise) must be minimized and the number of lost messages must be negligible.

These requirements dictate duplicated capability in one of the following forms:

1. Operating redundancy
2. Standby subsystems
3. Load sharing with partial capability during partial system down time.

The system we will describe employs the third method above. Two identical processors and two identical disc file systems are employed. Each processor has the necessary hardware for continuous access to all teletype lines and the necessary adaptors to connect to any peripheral device. Both processors have access to both d[illegible] however, both processors and both disc files switch interconnections as a pair. The interconnections are electrically interlocked such that two processors cannot be simultaneously connected to the same disc file and such that two disc files cannot be simultaneously connected to the same processor. Both processors must agree before a connection can be changed unless one of the processors is in a power off condition.

With both disc files operational, one is designated by the system supervisor as the master and one as the slave. A disc file that is inoperative or just coming on line after maintenance would be designated as the slave. All write operations are duplicated at the same location on both disc files (if both are operational), but reading is first directed to the master disc file. In the event of a disc system malfunction or scheduled maintenance period the affected disc is simply switched out of the system without disturbing in-progress operations or degrading system capability.

With both processors operational, each is assigned a responsibility for approximately half of the communication lines by the use of stored responsibility tables. Duplicate line queueing tables are maintained in each processor by means of an interconnecting data channel. In the event of a processor malfunction or a scheduled maintenance period, it is only necessary for the remaining processor to accept responsibility for all lines and process all messages in the line queueing tables. This is a partial capability mode which only degrades system operation during high traffic periods. Saturation of the system is avoided by restricting output and reducing, where possible, polling requests for additional input messages. Only those messages incoming to and outgoing from the affected processor at the time of malfunction need be retransmitted.

In order to minimize complete system down time, emphasis must be placed on minimum down time of each subsystem. A comprehensive diagnostic program system at both the micro and macro levels for both preventative maintenance and malfunction diagnosis is stored on magnetic tape. The "stored logic" or "microprogram" concept employed in the Collins C-8401 processor is especially useful in this area. Special diagnostic sequences, not available in the operating instruction repertoire, are loaded into the microprogram memory for each diagnostic testing phase.

A generally accepted requirement for a data switching center is store-and-forward operation with unlimited queueing. This is necessary to obtain a reasonable line and end-device utilization. In electromechanical teletype switching centers, this requirement is instrumented by storing incoming data on paper tape by means of a tape reperforator and then either manually or electrically transferring this tape to an outgoing tape reader. This same queueing capability must be retained in most electronic switching center applications.

In this system the amount of storage required for message data waiting for a not-busy outgoing line dictated a more economical storage media than core storage. Reliability and access restrictions ruled out magnetic tape. Considering only queue storage requirements, a magnetic drum would suffice. However, the additional requirement of immediate recall of any message less than 24 hours old boosted mass storage requirements into the disc file category.

## DISC FILE ORGANIZATION

The disc file used in Collins' message switching systems is the Bryant series 4000 with a "B" frame. This unit is capable of expansion to thirteen 39-inch discs with a total data storage capability of over 226 million bits or approximately 110,000 typical teletype messages. The exact number of data discs used in any particular system will depend on the system requirements. All but two disc surfaces are equipped with movable heads which may be positioned to any of 128 different tracks (see **Figure 1**). The remaining two disc surfaces are equipped with fixed heads that are used for timing signals and fast access storage. The movable heads are all controlled by a central hydraulic actuator which enables positioning within a maximum time of 200 milliseconds. There are physically six movable heads on each disc surface, the outer three heads of which may be addressed as six different heads. This is made possible by interlacing the data for each of the two addresses assigned to each outer head. Data for the even number head assignments will be recorded during the even phase of the disc clock, and data for the odd number head assignments will be recorded during the odd phase of the disc clock. On the inner track of the inner double address head, data will be recorded at a maximum density of 235 bits/inch. The disc rotates at a nominal speed of 1,200 rpm which is equivalent to 50 milliseconds per revolution.

All tracks are divided into 16 angular sectors. The selection of an arm position, head and sector, uniquely addresses a single cell of 528 bits. In this system these bits represent thirty two 16-bit words and one parity word. This cell size was chosen to give an optimum trade-off between MCS storage requirements and disc file initiation frequency. Adjacent cells are separated by an intersector gap equivalent to seven 16-bit words. Head switching is allowed in each intersector gap and the adapter is designed so that independent read/write operations may be performed during

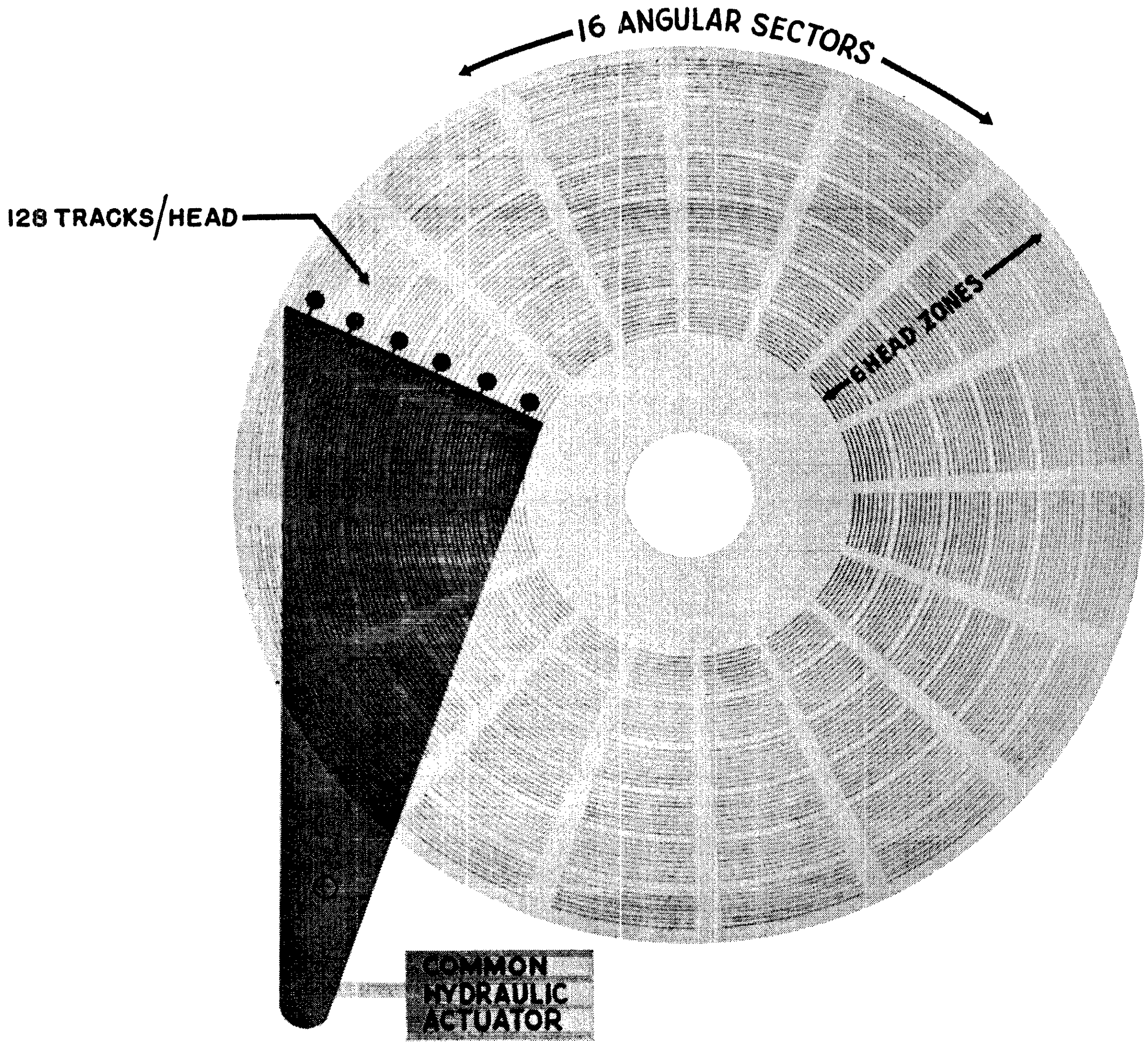



Figure 1

consecutive sector times regardless of the head or read/write selection.

The two disc surfaces reserved for timing signals and fast access data storage are each equipped with 13 fixed heads. Three of the heads on each disc surface are designed to read timing signals only and the remainder are available for fast access storage. Six of the outer heads are addressable as 12 different heads through data interlacing thus making available a total of 32 effective fixed fast access heads. These heads may be used while the hydraulically controlled movable heads are being positioned.

DISC FILE-PROCESSOR INTERFACE

Communication between the Bryant disc file and the Collins C-8401 processor requires a Collins C-8076 disc file control, a Collins C-8072 disc file adapter and a 16-bit I/O register. The disc file control contains 130 three by three inch circuit cards to accomplish reading or writing to the selected head. The disc file adapter contains 374 cards to accomplish disc file-processor interconnection selection, command code interpretation and word transfer control. The 16-bit I/O register is a standard I/O register which handles all control and data communication between the processor and the disc file system.

The adapter is designed to recognize four unique command codes as follows:

1. Status request
2. Arm position
3. Head select
4. Sector and read/write select.

Upon transfer of any of the first three command codes to the adapter's I/O register, the adapter cycles through a corresponding sequence which terminates with current status information being set into the same I/O register. This status information identifies the next disc sector available, the current disc-processor interconnection, and the dynamic state of the arm positioner. The total cycle time in each case is less than five microseconds. The second command code repositions the movable heads to the specified location. The cycle time is five microseconds although it may require up to 200 milliseconds to actually position the heads. The third command code results in the selection of the specified read/write head. This head remains selected until the generation of a new head select command code.

The fourth command initiates the transfer of a buffered sequence of thirty three 16-bit words between the disc file and the processor. If a movable head is selected and arm positioning is in progress the adapter automatically waits until arm positioning is complete. If a fixed head is selected, the state of the movable heads is ignored. When the selected head is positioned and the intersector gap preceding the selected sector is under the heads, the adapter connects the selected head to the selected read or write circuitry and sets a communication bit in the I/O register indicating to the processor that data must be exchanged with the adapter within the next 60 microseconds. After each word transfer the processor resets the communication bit. This set-reset sequence is repeated 33 times, after which status information is set into the I/O register and the adapter is ready to accept a new command code.

The C-8072 adapter disc-processor interconnection logic is designed such that if two working processors have interconnection codes in their respectable I/O registers agreeable to a configuration change, the two adapters will electrically switch disc files within a microsecond. Either processor may set a code in its adapter's I/O register which will freeze the current configuration. It is also possible for either processor to use a code which will allow the other processor to change configuration at will. The interconnection selection codes may be set into the I/O registers at any time and will remain in the last set state until set to another state by the processor.

## THE C-8401 PROCESSOR

The Collins C-8401 processor is a medium size, stored logic data processor whose design is oriented toward application in communication systems. The C-8401 main frame includes a coincident current magnetic core storage (MCS) memory, an arithmetic unit, a logical operations unit, several utility registers, input/output registers, a transfer matrix and a non-destructive memory (Biax) for the stored logic sequences (see **Figure 2**). The main frame contains approximately 1,500 three by three inch circuit cards. The basic word length of the processor is 16 bits. The main memory (MCS) has a five microsecond read/write cycle time and is expandable to 65,536 words. The arithmetic unit is a binary one's complement adder with a two-third microsecond add/cycle time.

The stored logic Biax memory has a capacity of 1,024 thirty six-bit words and a read/cycle time of one microsecond. Each output word from the Biax is commutated into three sequential register transfer commands giving a transfer rate of three million transfers per second.

The transfer matrix accomplishes 16-bit word transfers between the various utility registers, memory access registers, I/O registers, arithmetic registers and logic registers. Each and every I/O operation, main memory access, arithmetic unit operation and Biax memory access is directly controlled by transfer commands. The list of transfer comands stored in the Biax memory is referred to as either a transfer sequence list or a microprogram since it controls basic machine functions at the lowest level. The actual process of microprogramming follows conventional programming disciplines. The microprogram has capabilities for performing tests, making conditional and unconditional jumps, and following loops. Micro-instruction modification is limited since write-in to the Biax memory must be preceded by a complete clear of all locations. Variable connectors, address modification and other forms of instruction modification are affected by using main core storage for the variable quantities.

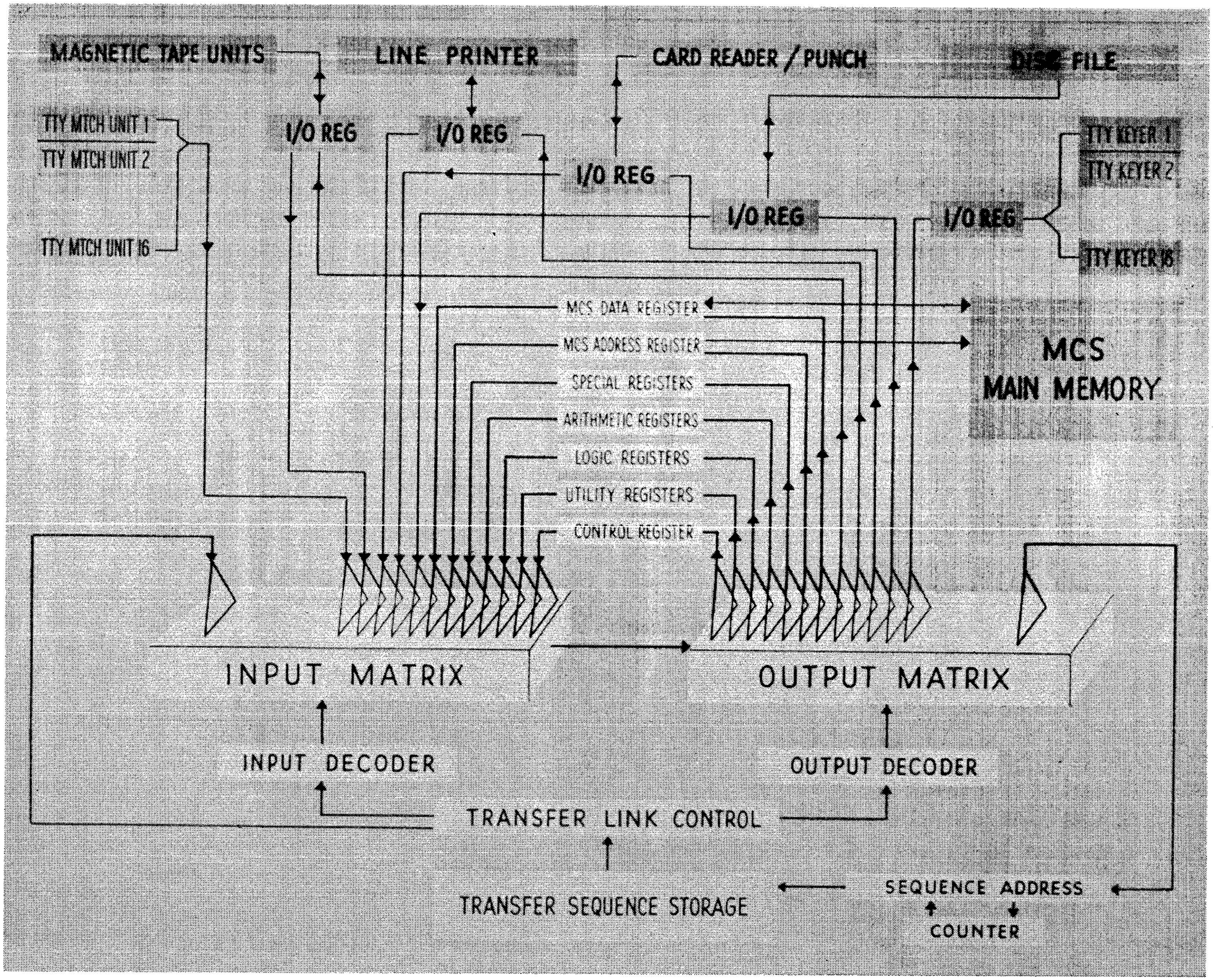



Figure 2. COLLINS C-8401 PROCESSOR ORGANIZATION

The C-8401 can be microprogrammed to interpret any main core program instruction repertoire (macro program). The repertoire used in the message switching system discussed herein stresses transfer capabilities and logical manipulations which expedite the processing and routing of message data. The capability of the C-8401 to simulate existing processing systems will not be discussed in this report inasmuch as the capability is not exercised in the message switching system under discussion.

## PROGRAMMED LOGIC SYSTEM

The C-8401 Processor Programmed Logic System consists of macro instruction interpretation micro routines, real time input/output micro routines and a macro assembler. The real time input/output micro routines consist of the *teletype input/output sequence* and various peripheral I/O buffering sequences. The teletype I/O sequence is a continuously active I/O sequence which performs the conversion between teletype signals and the processing format. Peripheral devices such as the disc file, magnetic tape and high speed printer employ micro routines that effect block transfers of data between the peripheral device and the processor magnetic core storage concurrent with the execution of other machine operations.

The macro assembler accepts programs written in a human-oriented source language and translates them into the machine language. This assembler is identified by the acronym COGENT (*C*ommunications *O*riented *GEN*eral *T*ranslator). The instructions produced by this assembler and interpreted by the instruction interpretation micro routines are called COGENT instructions. COGENT instructions have from zero to four operands and extended precision capability. All COGENT I/O instructions are single operand instructions specifying the direct address of an I/O service packet.

There are three basic disc instructions:

1. Request status and disc-processor interconnection select
2. Position arm
3. Read/write.

(Note that these instructions are macro instructions and should not be confused with the previously defined disc file commands which are used only in the disc file I/O register under micro program control.)

When the first instruction is used, the I/O service packet parameter defines the disc-processor interconnection to be selected. When the second instruction is used, the I/O service packet parameter defines the arm position to which all of the movable heads are to be positioned.

When the third instruction is used, the I/O service packet defines the following parameters:

1. Read or write operation
2. Read/write head number
3. Sector number
4. Main memory address where read or write data is to be stored or accessed
5. Initial parity word.

This instruction results in the execution of an initiate microprogram which selects the proper head number, the proper sector number, and the read or write mode. When the selected sector approaches the read/write heads, a second microprogram will be executed which will exchange data (read or write) with the disc file adapter. The time lapse between the initiate microprogram and the first execution of the data transfer microprogram, and between each word transfer of the data transfer microprogram, is used for execution of other processor sequences. This overlapping of the disc file read/write instruction microprogram with other processor sequences is enabled through periodic tests of the disc file adapter communication bit in all overlapping microprogram sequences. When completion is recognized, control is returned to the macro program which originally generated the disc instruction.

Maximum reliability of disc read/write operation is achieved by using an accumulative spiral parity scheme in conjunction with an initial parity assignment of a 16-bit code that is unique for each cell recorded. This procedure forms the modulo-2 sum of all 32 data words and the initial parity word. The spiral parity characteristic is developed by shifting the parity accumulator left one bit preceding the modulo-2 addition of each new data word. This minimizes the probability of cancelling errors in specific bit positions. The 33rd word written on the sector is the parity word. When this data is read from the disc, the same initial parity word must be supplied by the macro programmer; otherwise a parity error will be indicated by the disc file read microprogram. The use of a unique initial parity for each sector of data stored minimizes the possibility of disc addressing errors not being detected.

## MESSAGE FLOW THROUGH SYSTEM

A diagram illustration of message traffic flow through the system is shown in **Figure 3.** The teletype I/O sequence interrogates incoming teletype lines to detect the presence of character bits. These bits are assembled into five-bit baudot characters and packed three characters per computer word. These words are then stored into 32-word bins in MCS (magnetic core storage). The teletype I/O sequence must also output messages to the outgoing teletype lines in a fashion similar to inputting.

When the first word of a message is assembled, the teletype I/O sequence must determine an MCS storage location for the message. A bin availability table consisting initially of the MCS addresses of 512 bins is used for assigning buffer storage locations for messages or parts of messages. A message averages about four bins in length. Therefore each bin of a message, except the last, must be chained to the next bin of the same message since the bins are not necessarily contiguous in MCS. An MCS chaining address is supplied by the teletype I/O sequence in the 32nd word of each bin of the message, except the last.

After a message bin is filled, the *traffic storage routine* replaces the MCS chaining address with a disc file chaining address. The traffic storage routine looks up the disc file address previously assigned to this bin, generates an I/O service packet, and transfers control to the *peripheral I/O service routine.*

Disc file addresses are pre-assigned in a sequential order such that all cells of each arm position are assigned before assigning the cells of the next sequential arm position. This tends to minimize positioning time and assures that all cells within each arm position will be used. This also assures a relatively constant disc file "lifetime" of all messages. Each bin is written as it is filled and each bin area of MCS is then made available for other messages by placing the bin address in the bin availability table.

The first bin of a message contains message header information regarding the sending station and each receiver station's address.

The *message editing routine* checks for proper format and provides to the *message routing routine* a list of all receiving station addresses, the message priority, and other header information. The outgoing line number for each station address is determined by the message routing routine. The *output message queueing routine* makes the appropriate entries, according to priority class, in the outgoing message queue table for each teletype line to which the message is to be sent.

A station address consists of a variable number of mnemonic characters (2-8). The routing procedure is complicated by many rules involved and many exceptions to these rules. Approximately 10,000 computer words are required for the routing tables. In some cases several accesses to these tables are necessary before the outgoing line number is determined. These tables are used frequently and are stored on the fixed head area of disc for fast access purposes.

The *output line service routine* must monitor all outgoing lines and supply the teletype I/O sequence with data to be outputted if a message is waiting in queue for the line. Once a message has been started on an outgoing line, each succeeding bin must be in MCS by the time the teletype I/O sequence has finished with the current bin; however, the new bin will not be brought to MCS from disc too soon or MCS space will be unnecessarily wasted.

Once a message has been sent over an outgoing teletype line, this message becomes a part of the historical file. It must remain on the disc file for at least 24 hours in case the receiving stations or the sending station wish a fast recall of the message. Before the message is overwritten on the disc, it is transferred to magnetic tape where it is available for recall for a period of 30 days. The *history routine* generates the necessary locator tables and also transfers messages from disc to magnetic tape. One track (16 cells) from disc is written as one record on magnetic tape. The *retrieval routine* has the ability to locate and retrieve any message in the historical file.

## PERIPHERAL INPUT/OUTPUT CONTROL

All disc read and write requests must be made to the *peripheral I/O control routine.* The large volume of data that must be handled requires that efficient use be made of each disc revolution. This is accomplished by queueing all requests. Entries to this queue are arranged (sorted) by arm position and sector. Head and read/write selection is not important as far as queueing read/write requests are concerned since sufficient time exists within the intersector gap for selecting the head and the read/write mode. The position of the disc can be determined by interrogating the adapter. Thus the execution of read or write

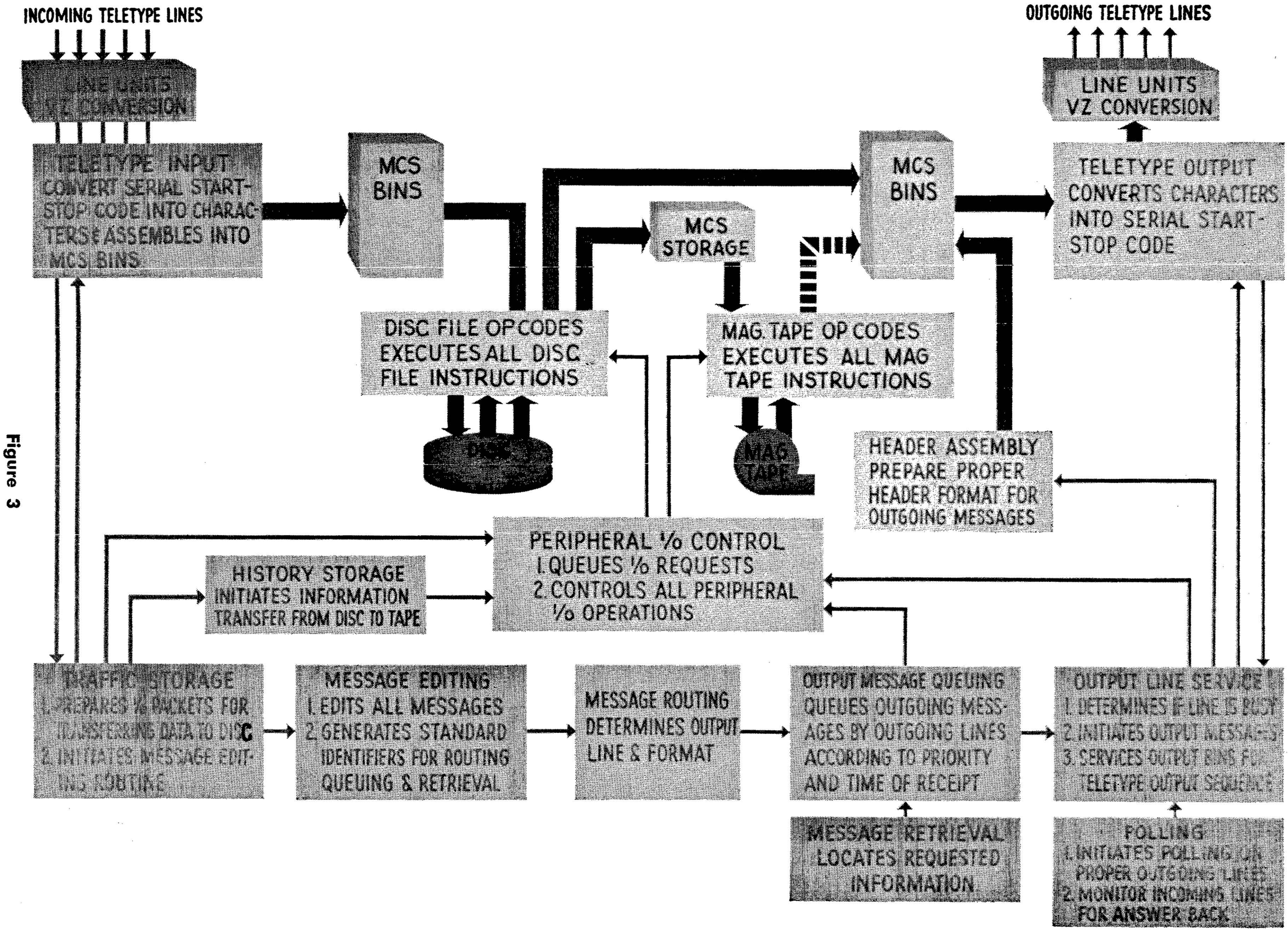



Figure 3
MESSAGE AND PROCESSING FLOW

| 1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12. 13. 14. 15. 16 | | | |
|---|---|---|---|
| STATUS WORD | | | Required by the disc file I/O Op-codes |
| MAIN CORE STORAGE LOCATION | | | |
| INITIAL PARITY WORD | | | |
| IND. BITS | HEAD | SECTOR | |
| NUMBER OF BINS | ARM | | Required by the Peripheral Input/Output Control Routine in addition to the first 4 words. |
| CHAIN ADDRESS TO NEXT PACKET | | | |

**INPUT/OUTPUT SERVICE PACKET**

**Figure 4**

requests may start at the next sector to pass under the read/write heads.

Provisions are also included for handling priority requests. This feature is used whenever processing must be delayed until a read or write operation is complete. Two examples are when reading routing tables from the fixed heads, and when reading a routine from disc to overlay an existing routine. This last read is necessary since the 32K main core memory used in this system is not large enough for all the switching routines and MCS table storage requirements.

A routine initiating a read or write request must build an I/O service packet and then transfer control to the peripheral I/O control routine. The packet contains all parameters relating to the I/O request. These include: 1) the arm, head, and sector address; 2) the number of successive disc cells or MCS bins involved; 3) a read/write indicator; 4) a priority operation indicator; 5) a bin availability table indicator; 6) a status word; 7) an initial parity word; and 8) a word used to chain packets in the I/O queue which fall within the same sector and same arm position.

Upon completion of a disc read or write operation, status information is left in the first word of the packet. The routine originating the request must test specific status bits to determine if the operation is complete or if an error occurred. This status information will also identify the next disc sector that can be accessed, the current disc-processor interconnection and the dynamic state of the movable heads.

The disc I/O op-codes are designed to transfer only one 32-word bin between disc and MCS per read or write instruction. By giving the total number of bins involved in the operation, the requesting routine is relieved of having to call for each group of 32 words individually. Peripheral I/O control will provide for this ability. An indicator is available which will cause bin addresses to be released to the bin availability table following a write or which will cause bin addresses to be obtained for storage of messages.

The routine requesting a read or write need not worry about the status of the disc files or the position of the arm. This is also taken care of by the peripheral I/O control routine. The packet supplied to the peripheral I/O control routine contains the necessary information that is needed by the disc I/O op-code to perform a read or write and sufficient additional information for housekeeping functions within the peripheral I/O control routine.

An important reliability feature of the disc file system involves the use of an initial parity word when reading or writing message traffic. The initial parity word is derived from an identification code such that each bin of each message is uniquely (limited by 16-bit word) identified and the identification of each successive bin is easily computed from the previous identification. When reading back message traffic the same initial parity word is used as when writing.

After initiating an I/O operation, peripheral I/O control releases control to the routine initiating the request so that other processing can be performed while the I/O operation is being completed. When the disc I/O op-codes complete their function, an interrupt of the switching routines takes place. Control is then given to peripheral I/O control so that another I/O operation can be initiated, or if an error condition occurred, an attempt can be made to recover from the error condition.

Other features of the peripheral I/O control routine are listed as follows:

1. All data are written on two disc files, for backup reasons, and then read back to insure that data were written properly. If a parity error is detected after the read-back from either disc, the data are written in an alternate location on the affected disc and an entry is made in an alternate cell table. The completion bit in the I/O service packet is set when both writes and read-backs are completed successfully from both disc files.
2. Alternate cell tables are maintained for each disc file and each arm position. These tables are stored in the fixed head area of the associated disc file. During arm positioning time, the table corresponding to the old arm position is written to disc and the table corresponding to the new arm position is read into MCS. Each entry in one of these 32-word tables corresponds to an alternate head and sector which is reserved for recording data which cannot be correctly recorded at the packet address specified by the user routine. Each entry, if occupied, will contain the disc address (head, sector) of a cell for which a failure occurred. Thereafter, if a read failure occurs, the peripheral I/O control routine will search this table to determine if the data were relocated. Thus the switching routines need not be concerned whether or not the data have been relocated. The relocation of data that do not give a correct read-after-write check provides a practical solution to the inherent disc file problem of "bad spots".
3. A read will be attempted X times (X being a parameter), if necessary, in order to get a good read without a parity error. The completion bit in the packet will be set when the read is accomplished successfully or after X attempts have been made to read from each disc file. The error bit will be set if an error exists after the last read.
4. All disc file-processor interconnection switching is initiated by this program. One disc file is treated as the master and the other as the slave. All reading will be attempted from the master disc file first. If the read is unsuccessful after X attempts, it is tried from the slave disc file. The slave disc file is used as backup for the master, and will take its place if the master is taken off line for maintenance or because of failure.

## CONCLUSIONS

The important factors we discussed concerning the application of a disc file to message switching systems are summarized below:

1. Sufficient storage capacity for immediate message retrieval requirements.
2. A comprehensive disc file I/O queueing routine operating in conjunction with an efficient disc file data organization scheme to handle the volume of message data without requiring exorbitant MCS storage.
3. Protection against catastrophic failure of the disc by the use of a redundant disc file system.
4. Error control by the use of an effective error checking scheme, read back after write, relocation of data away from temporary and permanent "bad spots", and backup of data retrieval with the redundant disc file.

# OTHER MASS STORAGE TECHNIQUES

by R. W. Carriker*
UNIVAC-Division of Sperry Rand Corporation
Aerospace Industries Branch
Los Angeles, California

# UNIVAC FASTRAND Mass Storage – A UNIVAC 490 Subsystem

THE UNIVAC DIVISION of the Sperry Rand Corporation announced on December 26, 1962 a new dual-drum mass storage device known as FASTRAND, a subsystem for the 490 Real Time Computer.

FASTRAND mass storage units have a capacity of 64,880,640 characters each. Average access time is 92 milliseconds to a fixed length sector of 165 six-bit characters, recorded in bit serial manner.

Each FASTRAND unit consists of two drums revolving at 870 rpm; the drums are contained in a cabinet measuring 122 inches long, 33 inches wide, and 63-3/8 inches high. Sixty-four flying heads mounted on flexure springs are used to search the rotating drums for desired information.

Very few moving parts are utilized to position all 64 heads in each unit. Positioning is accomplished through use of a linear transducer directly coupled to the positioning carriage on which the heads are mounted. Only 11 bearing surfaces are employed in each FASTRAND unit. Two motors, one integrally mounted to each drum, are used to drive the system, which is completely self-contained with its own power supply.

Existing programming packages will operate with the FASTRAND mass storage subsystem.

This report explains how the FASTRAND mass storage subsystem is organized and how communication between FASTRAND and the computer is effected.

## SYSTEM ORGANIZATION

### *File units*

A FASTRAND mass storage subsystem is composed of from one to eight FASTRAND storage units and a FASTRAND control cabinet. The control cabinet includes a control unit and a channel synchronizer. These units provide the computer system with the capability to store in excess of 500,000,000 characters of information per input/output channel. A storage unit, containing two drums in one cabinet, is treated as one physical and logical unit.

### *Data format and access*

Data is recorded in a bit serial manner in fixed length sectors of 165 six-bit characters; an equivalent of 33 complete UNIVAC 490 words. The data are organized to provide continuous binary addressing through any one storage unit. A storage unit is composed of two drums and 64 data heads, each of which may be positioned to any one of 96 tracks. Each track has 64 thirty-three word sectors. All data heads move simultaneously. Any one storage unit has 393,216 sectors of 165 characters per sector. This produces a capacity per unit of 64,880,640 characters. A FASTRAND unit revolves at 870 rpm. The maximum latency time is one drum revolution or 70 milliseconds; the mean latency is one half drum revolution or 35 milliseconds. The maximum time to access or position a data head is 86 milliseconds. The mean time to position a head is 57 milliseconds and the minimum time to reactivate a head already positioned is five milliseconds.

The lookup time is the sum of the latency plus access time. Maximum lookup time, therefore, is 156 milliseconds; mean lookup time is 92 milliseconds. Thirty milliseconds are required to move from track to track. When reading or writing continuously, 20 microseconds are required to switch data heads. A "Position" function code positions all the read/write heads to the desired

* Mr. Carriker has since left the employ of UNIVAC.

track where they remain until the unit has been re-addressed to a different position. Independent simultaneous movement of the positioners on all the cabinets connected to one control unit is possible, because as soon as the positioning has been started, the subsystem is released for operation with the other units.

*Track format*

A variable interlace is provided and must be specified before delivery. These options are:

| *Interlace* | *Skip-Read* |
|---|---|
| 3 | 2-1 |
| 7 | 6-1 |
| 9 | 8-1 |
| 21 | 20-1 |

*Data checking*

Data are checked on a sector basis by the "Data Check Generator Method". This method not only incorporates effective error detection but also reliable data coverage ability as well. During the write operation the FASTRAND subsystem generates and writes data check character for each sector of data received for storage. The check character will be used by the *data recovery routine* should no errors be detected when reading this information.

*Addressing*

Addressing is binary continuous through an entire unit, each individual number defining a discrete sector of data. Any invalid or non-existent addresses in a subsystem are detected and the computer is notified by a status word. The addressing scheme is as follows:

| 3 bits Unit | 8 bits Position | 6 bits Head | 6 bits Sector |
|---|---|---|---|

*Functions*

A six-bit function code is located in the high order portion of the function word. The starting address of the function is 23 bits positioned in the lowest order positions of the function word.

1. Positioning

   The position function positions the read/write heads on the desired track. Only the unit and track portion of the address is significant.

2. Read

   The read function causes data to be read and transferred to the computer continuously, starting at the specified address and continuing until terminated by one of the following:

   a. Terminate function
   b. Time out
   c. Parity error
   d. Sector length
   e. End of position input
   f. Address error
   g. Control unit sequence error.

   An interrupt and appropriate status word is generated for all of the above.

3. Write

   The write function causes data to be written continuously from the computer, starting from the specified address and continuing until terminated by one of the following:

   a. Terminate function
   b. Normal ending
   c. Sector length error
   d. End of position output
   e. Error address
   f. Control unit sequence error.

   If a full sector of data is not transferred within the time allowed, the data transferred are recorded and the remainder of the sector is padded out with zero bits and correct parity. Upon completing the writing, an interrupt and appropriate status word is generated. Note that the read and write functions are both continuous functions and a number of sectors can be read or recorded with one function.

4. Search

   a. Identifier

      Initiation of the search function requires a transfer to the mass storage control unit from the computer of a search function word followed by an identifier word.

   b. First word

      The search first word function causes the first word of each sector starting at the specified address to be transferred to the synchronizer for comparison with an identifier word. When a positive compare is made, the sector is read immediately (without another drum revolution) and transferred to the computer starting with the first word in the sector as in a normal read function following the same rules. If a negative compare is made, the search is automatically continued to the first word of the next sector and so on until the positive compare is made or the function is stopped by one of the following:

1) Terminate function
2) Parity error
3) End of position input
4) Address error
5) Control unit sequence error
6) Sector length error.

An interrupt and appropriate status word is generated for any of the above.

c. All words

The search all words function is similar to the search first word function except that all words in the sector are compared.

d. Length of search

If a positive compare is not made in a search operation between the time the last addressed sector of a track has been searched, the function is terminated and the computer so notified. An appropriate interrupt and end of position status word is sent to the computer.

5. Write lockout

Provision is made to optionally inhibit writing in the first 4,096 consecutively addressed sectors starting with address zero. This option is effected by a key-lock switch.

In addition to the position function, the read, write, search first word, and search all word functions cause the read/write heads to position to the specified track address. Position confirmation is made to assure that the track address and the track used are identical.

### ERROR AND STATUS CONDITIONS

Various error and status conditions are generated for the computer as defined under the given functions and specifically a unique status word is generated for each:

*Parity error*

This error is developed during reading or search operations only. Reading is not interrupted in a sector for parity conditions. Reading continues until the end of the sector at which time a parity indication is made by means of an interrupt and status word which is sent to the computer.

*Address error*

Incorrect or impossible address.

*Sector length error*

Sector recorded incorrectly.

*Unoperable condition*

Unit not operating properly.

*Invalid function code*

Function code not recognized by the control unit.

*Invalid address*

Incorrect or impossible core address.

*Channel synchronizer*

Sequence or operational error in synchronizer.

*Channel synchronizer character error*

*Time out*

Time out error indicates that data have not been accepted by the computer within the allotted time. If the computer does not keep up with the specified data rate of the file for the given interlace, the function is terminated. The status word contains the address of the sector in which the time out occurred. Therefore, all read or successful search functions will normally end with a time out which will generate a status word with the address of the last sector read. This is due to the fact that these functions are continuous and that the function is terminated by the computer buffer area being exhausted and the subsequent output channel termination causing the control unit to time out.

*Normal ending*

This will only occur on a write function. When an interrupt is requested with the write function on completion of the function, an interrupt and normal ending status word is sent to the computer. If no interrupt is requested, the function is terminated and no indication made.

*End of position*

During the execution of a function when the address is advanced to a point which requires repositioning of the read/write heads in order to continue, the function is terminated and the condition indicated to the computer with a status word. This status word contains the address of the first sector at the new position. The function must be resubmitted to the control unit with a new address.

*End of position output*

During the execution of the function when the address is advanced to a point which requires repositioning of the read/write head in order to continue, the function is terminated and the condition indicated to the computer with the status word (no address). The function must be resubmitted to the control unit with a new address.

The FASTRAND mass storage units rent for $3,300 a month each and sell for $160,000. The FASTRAND control and synchronizer rents for $2,750 a month each and sells for $135,000. Availability is nine months.

by Emory A. Coil
and
Simon A. Goodman*
Librascope Division
General Precision, Inc.
Glendale, California

# Librascope Mass Memory — A "Working" Storage System

THIS PAPER describes a new mass memory system that provides the user with fast and highly flexible access to a large data base. The most significant attribute of this system is its ability to do much of the "work" of a processor.

Retrieval is accomplished on the basis of content alone. The system provides a full complement of logical criteria with which to perform the search.

Data may be retrieved and recorded without concern as to physical location.

The computer industry has long needed a bulk storage media that does not require extensive software systems, conversion of existing document identification and predetermination of query.

The Librascope Mass Memory satisfies the aforementioned requirement, through its ability to locate data on the basis of content, where it may be specified that the content qualify according to one or all of a full complement of logical criteria.

## PHYSICAL CONFIGURATION

The mass memory consists of six discs, 48 inches in diameter, rotating at a rate of 900 rpm. Fixed heads are employed for reading and writing of all data, so that access speeds are similar to those attained with drum memories.

Associated with the actual discs within the Mass Memory unit itself, are a high speed core buffer, switching and control logic, and extensive error checking facilities. The overall dimensions of the unit are 5′ x 5′ x 3′, 1,500 pounds; power requirement is 22w, 60~, 3∅.

The data within a mass memory module are subdivided into zones (only one zone is actually being implemented in those units currently under construction). Each zone consists of two twins which can be operated either independently of or in parallel, under program option. This latter feature provides the capability of dual recording and retrieval of any critical data without loss of time. The information is thus safeguarded against loss due to electronic failure.

Each twin includes 60 files; each file consists of 1,350 blocks, where each block contains 128 alphanumeric characters, for a total capacity of 19,736,000 characters. The system can be expanded to a maximum capacity of 1,161,216,000 characters. The block is the basic unit of data to be transferred between the mass memory and the central processor. One file is the amount of information which can be examined during a single revolution of the unit. This can best be understood through a consideration of the arrangement of data within a given file.

## PHILOSOPHY

The concept employed by Librascope in the past[1] and retained in the unit under discussion has been separation, by hardware means, of each data block into two portions. The primary part (in this case the first 18 characters of each 128 character block) is written into a special band, common to all blocks of the file. This creates a configuration analogous to a group of folders (blocks) in a file cabinet drawer (file). Each folder has a tab on which identifying information is written and all tabs are physically located so as to be easily readable.

The actual procedure by which the Librascope central processing unit retrieves information from its mass memory unit(s) follows this analogy exactly. The program loads three registers of 128 characters each which are physically contained

* Now members of the staff of Informatics, Inc.

[1] Coil, E. A., "A Multi-Addressable Random Access File System", 1960 IRE WESCON Convention Record, Part 4, pp. 42-47.

within the core buffer portion of the mass memory. These registers, "Key", "Mask", and "Control", specify to the mass memory all criteria for identifying and transferring the desired data. The relevant portions of the contents of each data block (as specified by the mask) are compared against the key in the manner determined by the control for each individual field. The types of comparison which may be ordered in any combination on any combination of data fields within the block are:

1. Content of the specified data field *equal* to corresponding masked key
2. Content of the specified data field *not equal* to corresponding masked key
3. Content of the specified data field greater than corresponding masked key
4. Content of the specified data field less than corresponding masked key
5. Content of the specified data field between specified upper and lower bounds.

When a file search is ordered, the mass memory examines the 18 character "tab" portion of every block of an entire file during one revolution time. If all criteria are satisfied, then the entire block is transferred to the core buffer and the remaining (secondary) fields are checked. If all criteria are again satisfied, then those fields designated by the control register are transferred to the central processor. It is significant that only the desired data are read in, as opposed to automatically reading the entire block. It is possible, in fact, to merely count the blocks which satisfy given conditions without actually transferring anything.

It is possible, under program option, to read in the fixed addresses of matching blocks.

The program is able, either by special instruction or automatically at the time of reading (optional) to mark any block position as available for re-use. This being done, the central processor can turn over to mass memory the entire task (or any part of it) of locating the positions into which new data are to be written.

The discussion thus far has been limited to data retrieval on a content basis. Although this is certainly the most significant feature of the mass memory unit, it should be noted that a full complement of conventional fixed address instructions are also available. Whenever convenient the Librascope Mass Memory can also be operated in this manner.

## UTILIZATION

The design of the Librascope Mass Memory is based on overcoming the restrictive ordering, key translation, predetermination of query and access time that have been imposed on the user of conventional disc equipment.

The following is a short description of the manner in which the aforementioned problems can be solved through the use of the mass memory.

a. *Ordering*

   On conventional systems it is necessary to serially order information, so that records that use a file space that is greater than one hardware block may be retrieved. Through the use of content search, sub-blocks of any given record may be retrieved by including in their key, a classification flag; thus they may be retrieved individually, in part, whole, or along with the rest of the record.

b. *Key translation*

   Through the use of content search, it is no longer necessary to convert existing document identification and numbering systems.

c. *Predetermination of query*

   Since any combination of relationships may be specified by using the search criteria, large and restrictive indexing systems are no longer necessary.

d. *Access time*

   Average access time for content search on primary key is 33.3 milliseconds.

e. *Selective data transfer*

   It is particularly significant that only the desired data are read in. Other disc systems have a limited version of this capability, however they require the use of special characters and combinations of characters to be recorded in file, using valuable space, creating significant debugging problems, and magnifying the problem of system changes and growth, which is an inherent requirement of every data processing system.

In light of its particularly powerful retrieval facilities and the fact that only a minimal amount of ordering is required, the Librascope Mass Memory lends itself to the speedy and efficient implementation of information retrieval and maintenance applications, particularly real time.

by K. H. Rash
The National Cash Register Company
Electronics Division
Hawthorne, California

# NCR's Card Random Access Memory (CRAM)

THE NCR 315 SYSTEM with CRAM was announced in 1960 and since that time has been impressive both within the electronics data processing industry and in sales.

The success of the CRAM is based mainly on the following factors:

1. It is a unique device which was the first to combine low cost file storage and efficiency for *both* random and sequential processing. This is in addition to providing relatively low cost off line file storage capability.
2. In addition to being a low cost unit the CRAM is the main factor in providing lower total system costs. The typical 315 CRAM systems have fewer handlers than typical tape systems.
3. It is a proven device with a great deal of potential for future development.

The 315 system can be classified as a low to medium cost general purpose data processing system with its major market being in commercial data processing areas.

## REVIEW OF CHARACTERISTICS AND OPERATION

The following summary of characteristics and operation is presented in order to provide a better understanding of the CRAM with its advantages and potential in current and future data processing systems.

### Physical description

*Handler*:

Cabinet dimensions

35 inches wide; 24 inches deep; 60 inches high

*Card*:

Dimensions

Length: 14 inches
Width: 3.25 inches
Thickness: .0056 inches (.005 inch Mylar base)

Coding

Eight binary-coded notches at top of each card provide unique addresses for a full magazine of 256 cards.

Recording density

250 bits per inch

Card speed

400 inches per second

Data format

Channels/card: 7
Tracks/channel: 8
Six bit alphanumeric character recorded in parallel across each channel with clock bit and parity bit.

### Maximum capacity

| | *No. of Cards* | *Alphanumeric Characters* | *Decimal Digits* |
|---|---|---|---|
| *Channels*: | ---- | 3,100 | 4,650 |
| *Card*: | ---- | 21,700 | 32,550 |
| *Magazine*: | 256 | 5,555,200 | 8,332,800 |
| NCR 315 System —16 units: | 4,096 | 88,883,200 | 133,324,800 |
| *Off Line*: | | Unlimited | |

### Timing

*Average Access*:

Card access time: Approximately one-fifth of a second

Re-access (average rotational delay): 23 milliseconds (one half of card rotational time)

*Transfer Rate*:

Instantaneous: 100,000 characters/second

Block transfer: Approximately 65,000 characters/second

**Operation**

When loaded into a CRAM unit the cards hang from eight rods which may be turned in such a way as to cause only one of the cards to be released. Each card has a set of binary coded notches which permits automatic selection, at random, of any one of 256 cards from the CRAM magazine.

After the card is released it is allowed to fall freely until it reaches a rotating drum, and the card is then accelerated to the surface speed of the drum, 400 inches/second. Shortly after attaining this speed the leading edge of the card reaches the read-write heads. The read head, besides its normal function of reading, is also used to perform an immediate check after writing.

After reading or writing, the card may remain on the drum, or it may be released and returned to the magazine. If a card that is on the drum has not been released by instructions or has not been accessed for 750 milliseconds, the card will automatically be returned to the magazine. When the card is released, momentum sends the card through the raceway and into the magazine. The loader plate then pushes the card back onto the selection rods.

## ADVANTAGES OF CRAM SYSTEMS

**Performance**

The CRAM is a flexible device which is efficient in applications involving the following types of jobs.

1. File maintenance, both in line and batch processing
2. Sorting
3. Inquiry
4. Compiling
5. Many other data retrieval and manipulation jobs.

The CRAM may be used efficiently in the three basic file updating techniques of random, serial selective and serial copy. Since most data processing problems do not require clear cut versions of any of these, the CRAM can be used to blend these methods into an optimally efficient system.

For random file processing, commonly used addressing methods are employed to select a CRAM unit, drop a card, and then read a channel. These methods include the use of directories, direct addressing or calculated addresses.

The CRAM has built into it the capability of having various levels of simultaneity of data access without the expense of controllers or synchronizers. For example, in a single CRAM there may be three active cards at one time in the following stages of activity:

a. A card may have been just released from the magazine area.
b. A card may be in the process of being read or written onto.
c. A card may be returning to the magazine area.

Furthermore, in multiple CRAM installations these same operations may be taking place simultaneously in more than one handler. Theoretically up to 16 CRAM's may be in various stages of operation simultaneously.

This simultaneity is facilitated by combining the use of photocells in a CRAM unit with the interrupt and branching features of the 315. That is, as the leading edge of a card approaches the read/write heads an automatic program interrupt occurs. This flexibility is extremely useful in on line inquiry applications.

**Storage capacity and costs**

A 315 CRAM system may include just one CRAM unit initially having 5.6 million characters of on line storage. This same system is capable of expanding to 16 units or 88.9 million characters of on line storage, thus providing on line storage costs of only $.172/1,000 characters per month.

Off line storage costs are also quite low. The cost per character for off line storage ranges from $.027/1,000 characters to $.035/1,000 characters depending on whether one card or 256 cards complete with canister and dust cover are stored.

The same features which facilitate low cost off line data storage also provide for low mailing costs and less space for physical media storage.

**Other advantages**

1. On a one-CRAM data processing system it is quite a simple task to duplicate a CRAM card and physically remove that card and place it in another magazine. This is ex-

tremely useful when various files use the same transactions for file updating purposes. Rather than just copy the transactions they would most likely be re-sorted into a sequence which would be most useful to the destination file; in either case the advantages are obvious.

2. Reliability and Backup

   The CRAM does not have the problems associated with mechanical movement of a head assembly, the only mechanics involved in card selection being the positioning of selection rods.

   The removable file concept provides backup capability in case one of the CRAM units becomes inoperative. The magazine on that unit can easily be removed and mounted on a spare unit, thus making the data available to the system with a minimum of down time.

3. In most instance it is necessary to check that data have been written on a file correctly. The CRAM does both the writing and checking operations in one card revolution.

## REFERENCES


1. Angel, A. M., "The NCR Magnetic Card Random Access Memory", presented at the Symposium on Large Capacity Memory Technique for Computing Systems, sponsored by Information Systems Branch of Office of Naval Research, May 1961.
2. Bloom, L., I. Pardo, W. Keating and E. Mayne, "Card Random Access Memory (CRAM): Functions and Use", presented at the 1961 Eastern Joint Computer Conference, Washington, D. C.