La Fan Stuart

BULLETIN 3026

Preliminary Edition

DATATRON

ELECTRONIC DATA PROCESSING SYSTEMS

HANDBOOK

Model 560 DATAFILE Multiple Bin Tape Unit

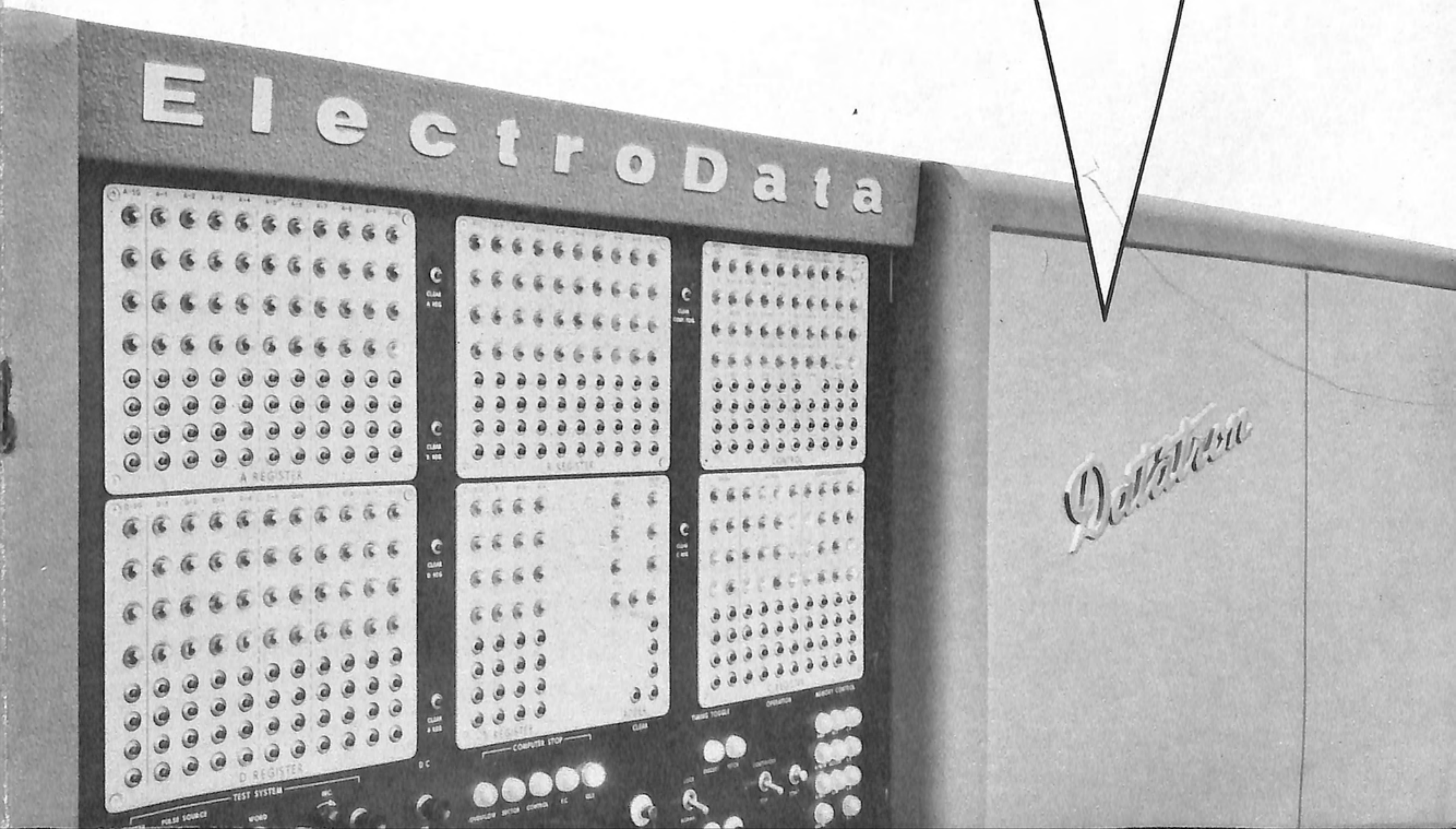

TABLE OF CONTENTS

MODEL 560 DATAFILE MULTIPLE TAPE UNIT

General 1

CHARACTERISTICS OF THE DATAFILE

Physical Description 1

Speed of Head Movement 2

Speed of Tape Operation 3

Random Access Time 3

DATAFILE COMMANDS

Magnetic Tape Search 4

Magnetic Tape Read 6

Magnetic Tape Write 7

APPLICATIONS

General 7

Methods of Arranging Files

1. Straight serial 8

2. Lane-parallel 10

3. Tape-parallel 12

4. DataFile parallel 14

5. Random Access 14

6. Frequency Distribution of File Data 15

7. Static and Dynamic File Separation of Data . . . 17

Sorting on the DataFile 18

General Conclusions 19

APPENDIX

MODEL 560 DATAFILE MULTIPLE BIN TAPE UNIT

GENERAL

The Model 560 DataFile is an auxiliary storage device providing relatively short access time to very large volumes of records. Up to ten DataFile units may be associated with the DATATRON at one time, or any combination of DataFiles and DataReaders up to a total of ten. DataFile units and DataReaders are controlled by the same Model 543 Tape Control Unit. The physical characteristics of the magnetic tape used in the DataFile are the same as that used in the DataReaders. The Magnetic Tape System Handbook describes the Tape Control Unit and the magnetic tape.

CHARACTERISTICS OF THE DATAFILE

PHYSICAL DESCRIPTION

The DataFile contains fifty separate 250 foot lengths of magnetic tape within the cabinet. Each hangs free between static-free metal partitions. A pair of magnetic reading and writing heads capable of selecting any particular tape moves beneath the tapes at a high rate of speed. Two lanes of information are recorded on each tape in addressed blocks of twenty words each. (See Magnetic Tape System Handbook.) On each of the fifty tapes, 1000 blocks, numbered 000 through 999, are recorded in each lane. Thus, there are 2000 blocks on each tape and a total of 100,000 blocks, or 2,000,000 words, per DataFile. Logically, each lane is addressable. There are a total of 100 lanes on which to read

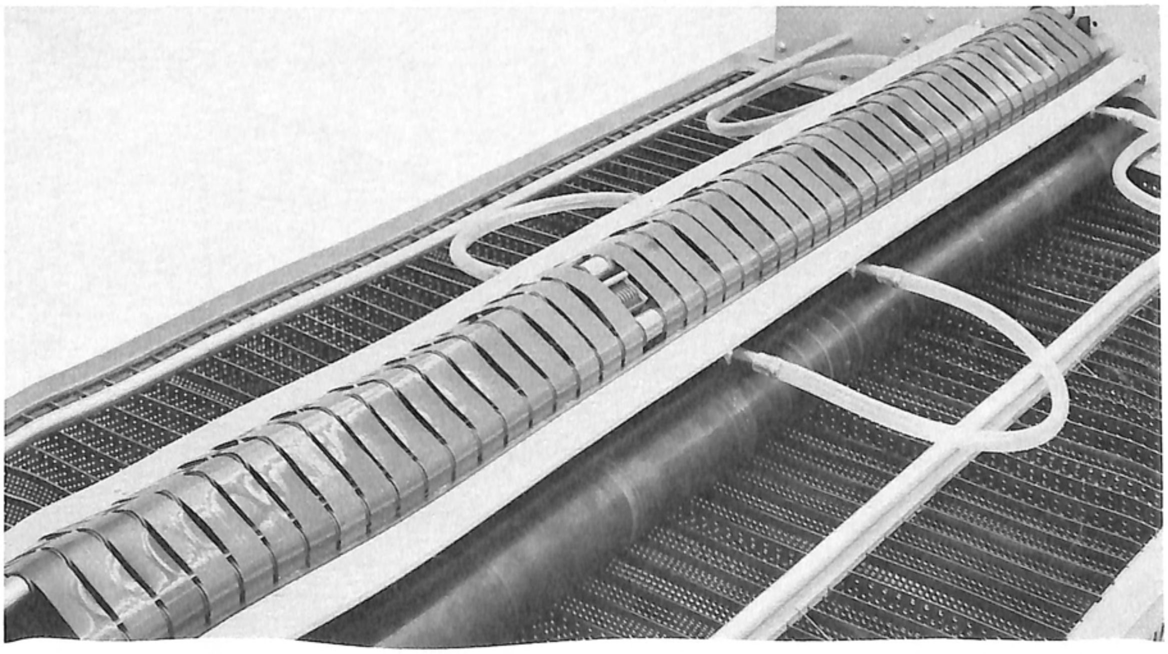

Figure 1. Read-Write Head Beneath the Tapes

or write information. The lanes are designated 00 - 99. As with the DataReader, a separate read-write head is used for each of the two lanes on one tape. As shown in Figure 1, the two read-write heads move as a unit beneath the 50 tapes. The carrier which drives the heads is controlled in such a manner that the heads will stop precisely in position to read or write.

One tape of the 50 may be moved backward or forward at any one time. When the read-write head leaves one tape to operate with a different tape, all other tapes remain stationary. When the head returns to the first tape for another read-write operation, the tape will be in the same position as it was after its last operation.

SPEED OF HEAD MOVEMENT

The carrier holding the two read-write heads moves the entire distance of the 50 tapes in less than two seconds. The increase in head movement time is approximately linear with respect to the number of tapes traversed.

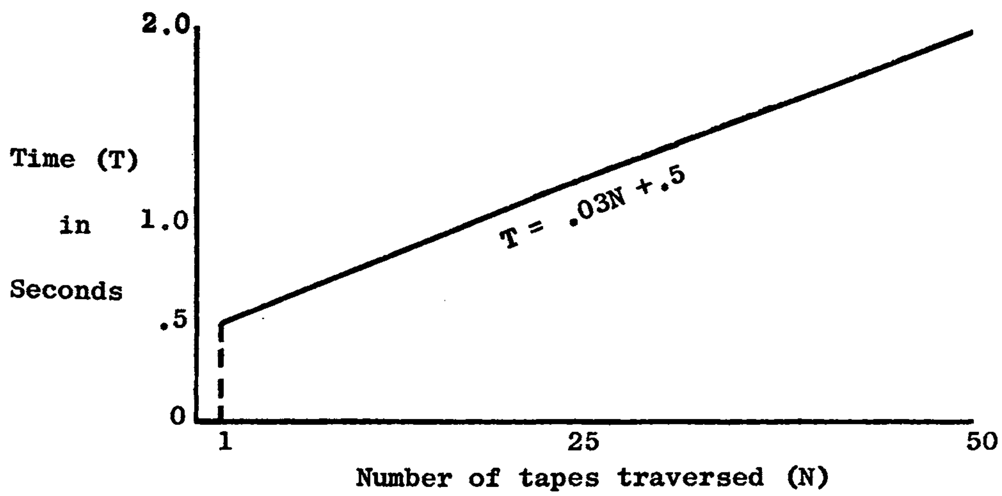

Figure 2. Approximate Read-Write head movement time

Whenever the head is moved from one tape to another, a relay release and final latch-in requires a maximum of .50 seconds. The average time to traverse one tape after initial acceleration is 30 milliseconds. The formula for calculating the approximate head traverse time may be expressed as:

$$T = .03N + .5$$

where N is the number of tapes traversed, .03 is the traverse time per tape in seconds and .5 is the latch-in time.

Example: If the head was to be moved from Tape Number 2 to Tape Number 5, Tapes 3 and 4 would be skipped and the total traverse time for the read-write head would be 3 (.03) + .5 = .590 seconds.

SPEED OF TAPE OPERATION

The operating speed of the magnetic tape is 60 inches per second. When a tape search is completed, the block addressed is in position to be read or written. When tape reading or writing is completed, the block following the last one read or written is in position to be read or written.

Most applications do not require the reading of an entire file since addressed blocks can be located by searching. File information read from tape to be updated is usually rewritten in the same position on the same tape after the updating.

RANDOM ACCESS TIME

It can be shown that the average time to locate one of a number of records stored at random on magnetic tape is the time required to travel past 1/3 of the total number of records. The formula can be stated as $D = \frac{a}{3}$, where a is the total number of records on a tape and D is the average distance traveled to obtain the record.

As with the DataReader, the time required to travel past one block of information in the DataFile is 46 milliseconds. Since 1000 blocks are stored on each lane, the average random access time for any one of the 2000 blocks on one tape is (1000 x .046)/3 = 15.3 seconds.

In order to obtain the random access time for the entire 100,000 blocks contained in the DataFile, the random access time to reach one of the 50 tapes must be added to the time given above. This average random access time is calculated by using the two formulas: $T = .03N + .5$ and $D = a/3$. The $\underline{N}$ in the first formula is obtained by taking the total number of tapes, <u>50</u>, and dividing by <u>3</u>. Thus, the average random time for head movement is $.03(50/3) + .5$ or 1.0 second, approximately. The average random access time for any one of 100,000 blocks (200 characters each) in one DataFile is therefore approximately 15.3 + 1.0, or 16.3 seconds. The time is the same for access to any one of 1,000,000 blocks stored on ten DataFiles.

Because of the independent search feature and parallel storage effect, which will be described in following sections, the average access time to any record is usually reduced to something less than two seconds.

DATAFILE COMMANDS

MAGNETIC TAPE SEARCH (MTS)

The SEARCH command takes the form:

s hhup 42 Oxxx

s	sign position (See section on B register Modification in Magnetic Tape System Handbook.)
hh	lane number
u	DataFile unit number, 1, 2.....8,9, 0.
p	breakpoint code position (See Central Computer Handbook)
42	numerical code for MAGNETIC TAPE SEARCH
Oxxx	address of the desired block

There are 100 lanes located on 50 tapes. The two read-write heads are designated odd or even. Thus, if <u>hh</u> is even, the even head will be used on an even lane in the searching operation; if <u>hh</u> is odd, the odd head will be used on an odd lane in the searching operation. Figure 3 shows the relationship between the head number and the tape and lane numbers.

Lane Number	Tape Number	Head Number
00	1	0
01	1	1
02	2	0
03	2	1
04	3	0
05	3	1
06	4	0
07	4	1
.	.	.
.	.	.
.	.	.
.	.	.
96	49	0
97	49	1
98	50	0
99	50	1

Chart of Head Designation, Tape, and Lane Number

Figure 3

For example, the command 5420 40 0693 causes the read-write heads on DataFile 2 to move to the 28th tape and latch into place. The even read head is selected and the search for block 693 under that head is begun. If, by a previous command, the tape was positioned at the beginning of block 438, the search command causes the tape to move past 255 blocks. The time required is 255 x .046 = 12.33 seconds. If the read-write heads were previously at tape 47, they would traverse 19 tapes. The time required is 19 (.03) + .5 = 1.07 seconds. The total time for the search operation would be approximately 13.4 seconds.

Only one DataFile or DataReader may be searching or reading or writing at one time. On one unit the head designated on the last search command will be the head used for all following read-write commands until the execution of a subsequent search command changes the head designation.

The DataFile has no rewind command. Tape movement to block 000 is effected by a MTS 0000 command.

MAGNETIC TAPE READ (MTR)

The READ command takes the form:

s nnup 40 xxxx

s	sign position (See section on B Register Modification in Magnetic Tape System Handbook.)
nn	the number of blocks to be written, 01 through 00 (100).
u	DataFile unit number, 1, 2....8, 9, 0.
p	breakpoint code position (See Central Computer Handbook)
40	numerical code for MAGNETIC TAPE READ
xxxx	address in main storage to which first word in first block is to be transferred

For example, 0 0710 40 3000 is the command to read seven consecutive blocks of twenty words each from DataFile 1, transferring words to consecutive storage cells on the DATATRON drum beginning with 3000.

The lane read from is the one designated in the last tape search on that DataFile. The Magnetic Tape System Handbook gives further detail on the transfers of information to drum storage. Successive TAPE READ commands may be given without an intervening TAPE SEARCH command.

MAGNETIC TAPE WRITE (MTW)

The WRITE command takes the form:

s nnup 50 xxxx

s	sign position (See Section on B Register Modification in Magnetic Tape System Handbook.)
nn	the number of blocks to be written, 01 through 00 (100).
u	DataFile unit number, 1, 2....8, 9, 0.
p	breakpoint code position (See Central Computer Handbook)
50	numerical code for MAGNETIC TAPE WRITE
xxxx	address in main storage from which first word in first block is to be transferred.

For example, 0 0420 50 1060, is the command to write four consecutive blocks of twenty words each on DataFile 2 transferring words from consecutive storage cells on the DATATRON drum beginning with 1060.

The lane written on is the one designated in the last tape search on that DataFile. Successive TAPE READ and TAPE WRITE commands may be given without an intervening TAPE SEARCH command. The Magnetic Tape System Handbook gives further details on the transfer of information from drum storage. a TAPE READ or TAPE WRITE command which is preceded by a TAPE SEARCH command at some point in the program (not necessarily the preceding step) must be followed by a CONDITIONAL CHANGE OF CONTROL. A TAPE READ OR TAPE WRITE command preceded by another TAPE READ or TAPE WRITE command need not be followed by a CONDITIONAL CHANGE OF CONTROL command.

APPLICATIONS

GENERAL

The concept of a serial-parallel magnetic tape storage is desirable in general in data processing techniques, particularly when associated with addressable tape systems. With the DataFile several dimensions of space-storage are available.

1. There are two lanes on each tape. Thus, when one lane is searched, the information on the other lane is bypassed. This situation is desirable since only one half the file need be searched and access will have been available for the entire file.

2. There are 50 tapes, and the rapidity of head movement among them offers the advantage of skipping large segments of a file. Addressable tapes are necessary to realize a saving in skipping one or more tapes when the head moves from tape to tape.

3. Each lane is serial in the storage of information. Serial storage is kept to one-tenth that of a reel unit, although the capacity of DataFile is five times that of a reel unit.

4. More than one DataFile may be utilized to contain a file offering a third parallel effect for file storage.

METHODS OF ARRANGING FILES

1. Straight serial

A file of records may be arranged in a serial sequence much the same as with storage on conventional reels. With the DataFile, this arrangement of file storage should be avoided if possible. If storage must be serial, the DataFile offers only a slight advantage over reel storage.

Consider a file-reference problem in which there are 3000 records of 200 characters each. The records are numbered 0-2999 and are recorded on the length of two tapes. It is desired to print out the information stored in certain records. Figure 4 shows the arrangement of records on the two tapes of the DataFile.

LANE NUMBER

Record Number

00	01	02	03
000	1000	2000	
001	1001	2001	B
002	1002	2002	
003	1003	2003	L
.	.	.	
.	.	.	A
.	.	.	
.	.	.	N
.	.	.	
996	1996	2996	K
997	1997	2997	
998	1998	2998	
999	1999	2999	

(Lanes 04-99 are also blank)

Figure 4. Tape layout for serial method.

Assuming the tapes are initially set at block 000, the tape searching distances for the desired blocks are given in Figure 5.

Records Desired	Lane Number	Tape Number	Distance Searched (Blocks)	Head Movement Time
0004	00	1	4	0
0027	00	1	23	0
0237	00	1	210	0
0491	00	1	254	0
0719	00	1	228	0
1171	01	1	548 *	0
1456	01	1	284	0
1777	01	1	321	0
1801	01	1	24	0
2098	02	2	98**	.53
2732	02	2	634	0
2803	02	2	71	0
		Total	2700	.53

*Change to lane 01 and search backward for 1171 which is the same as 171 on lane 00.
**In switching from tape 1 to tape 2, tape is left at block 1801, and the head moved to tape 2 to locate 2098.

Figure 5. Search distance table for serial method.

From Figure 5, the total time in searching is 2700 x .046 + .53 = 124.7 seconds. Therefore, the average access time per record is 124.7/ 12 = 10.4 seconds.

2. Lane-Parallel Method

The same file of 3000 records may be arranged in a lane-parallel sequence reducing the distance searched considerably. By this method, consecutively numbered records are stored alternately on the two lanes of each tape. Figure 6 shows this layout. As with the serial method, only two tapes are used.

LANE NUMBER

Record Number	00	01	02	03	
	0000	0001	2000	2001	
	0002	0003	2002	2003	
	0004	0005	2004	2005	
	.	.	.	.	
	.	.	.	.	(Lanes 04-99 are blank)
	.	.	.	.	
	.	.	2996	2997	
	.	.	2998	2999	
	.	.	B	B	
			L	L	
	1994	1995	A	A	
			N	N	
	1996	1997	K	K	
	1998	1999			

Figure 6. Tape layout for lane-parallel method.

In the lane-parallel method, searching progresses forward using both heads on each tape. Thus, the search backward for a record on the second lane of the same tape is eliminated. The searching distances for the lane-parallel method are given in Figure 7. All tapes are positioned at 000 to start.

Records Desired	Lane Number	Tape Number	Distance Searched (Blocks)	Head Movement Time
0004	00	1	2	0
0027	01	1	11	0
0237	01	1	105	0
0491	01	1	127	0
0719	01	1	114	0
1171	01	1	226	0
1456	00	1	143	0
1777	01	1	160	0
1801	01	1	12	0
2098	02	2	49	.53
2732	02	2	317	0
2803	03	2	35	0
TOTAL			1301	.53 seconds

Figure 7. Search distance table for the lane-parallel method.

From Figure 7, the total time in searching is 60.0 + .53 = 60.53 seconds. Therefore, the average access time per record is 60.53/12 = 5.0 seconds.

3. Tape-Parallel Method

The tape-parallel method is an extension of the lane-parallel method which further reduces access time to any record by spreading the file over all 100 lanes. The previous example is used to illustrate this method. The tape layout for the tape-parallel method is shown in Figure 8.

The tape-parallel method increases the head movement time considerably, but this is more than offset by the reduction in tape movement time. Since each tape contains only sixty records, a number of tapes are skipped during each search. Figure 9 shows the distance searched for each block and the approximate head movement time.

LANE NUMBER

	00	01	02	03		96	97	98	99
	0000	0001	0060	0061		2880	2881	2940	2941
	0002	0003	0062	0063		2882	2883	2942	2943
Record Number	0004	0005	0064	0065	. . . .	2884	2885	2944	2945
	..	..	..	..		..	..	..	..
	..	..	..	..		..	..	..	..
	..	..	..	..		..	..	..	..
	0058	0059	0118	0119		2938	2939	2998	2999
	Blank		Blank			Blank		Blank	

Figure 8. Tape layout for tape-parallel method.

Records Desired	Lane Number	Tape Number	Distance Searched (Blocks)	(Tapes)	Head Movement Time (Approx.)
0004	00	1	2	0	0
0027	01	1	11	0	0
0237	07	4	28	3	.59
0491	17	9	5	4	.65
0719	23	12	29	3	.59
1171	39	20	15	8	.74
1456	48	25	8	5	.72
1777	59	30	18	4	.62
1801	61	31	0	1	.53
2098	68	35	28	4	.62
2732	90	46	16	11	.83
2804	92	47	22	1	.53
	TOTAL		182 blocks		6.42 seconds

Figure 9. Search distance table for tape-parallel method.

From Figure 9, the total time in searching is 182 x .046 + 6.42 = 14.97 seconds. Therefore, the average access time per record is 14.97/12 = 1.25 seconds.

4. <u>DataFile Parallel Method</u>

This method employs lane-parallel and tape-parallel methods using more than one DataFile. The records are arranged the same as in the third method except divided among the DataFiles.

5. <u>Random Access Method</u>

In many applications, it is impossible to have the input data to the system in sequence. In this case, the four previous methods discussed cannot be used. It is in this type of problem that the DataFile has its greatest advantage over reel storage. The problem is to set up 2000 records on a DataFile using the random access method. Each lane will contain 20 records. Assuming that the reference to any one of the 200 digit records is in a completely random fashion, the average access time for any record can be calculated as follows:

> Average random head movement time for 50 tapes = 1.0 seconds.
>
> Average random tape movement time (20 records x .046 seconds)/3 = .31 seconds.
>
> Average random access time to any record is 1.0 + .31 = 1.31 sec.

In this particular example, the head movement is much greater than the tape movement time. Obviously, the 2000 records should not be spread across all 100 lanes for optimum efficiency. To illustrate this, suppose only 80 lanes are used. Each lane will contain 25 records. The average random access is then:

> Head movement for 40 tapes = .90 seconds
> Tape movement time = .38 seconds
> Average Random Access Time = 1.28 seconds

The number of lanes over which the records are spread is the key to the average random access time for all records. The optimum number of lanes to select can be determined. An approximation is

> Number of lanes (b) = $.12\sqrt{ac}$
> Where $\underline{a}$ is the number of records in the file,
> $\underline{c}$ is the number of characters per record.

Thus, in the example above, the optimum number of lanes is:

$$.12\sqrt{2000\ (200)} = 78 \text{ lanes}$$

Since 78 lanes result in an uneven number of records per tape, 80 is probably a better figure, giving 25 records per lane.

Using the above formula, it is interesting to note that all 100 lanes should be used whenever the file contains 3500 or more records of 200 characters each.

Appendix I contains a formula relating access time to any combination of lanes, records, and record lengths.

6. Frequency Distribution of File Data

Problems such as inventory control, billing, accounts receivable and payable, where a pattern of record reference is usual, a distribution favorable to processing the records is possible. Normally, certain of the records are referenced or up-dated more than others. In general, a small percentage of the records represent a large percentage of activity. Searching time can be reduced by taking advantage of this situation to the degree that independent computer processing time is sometimes equal to or greater than the searching time. The entire file may be maintained on the DataFile, but with such a structure that high activity records are more readily available than low activity records.

A new concept of file structure is introduced called "bell" format. Using the bell curve of normal popularity distribution, the more active records can be concentrated on each lane in a central area surrounded on each side by less active records. The file will be completely out of sequence identification-wise, but in sequence demand-wise.

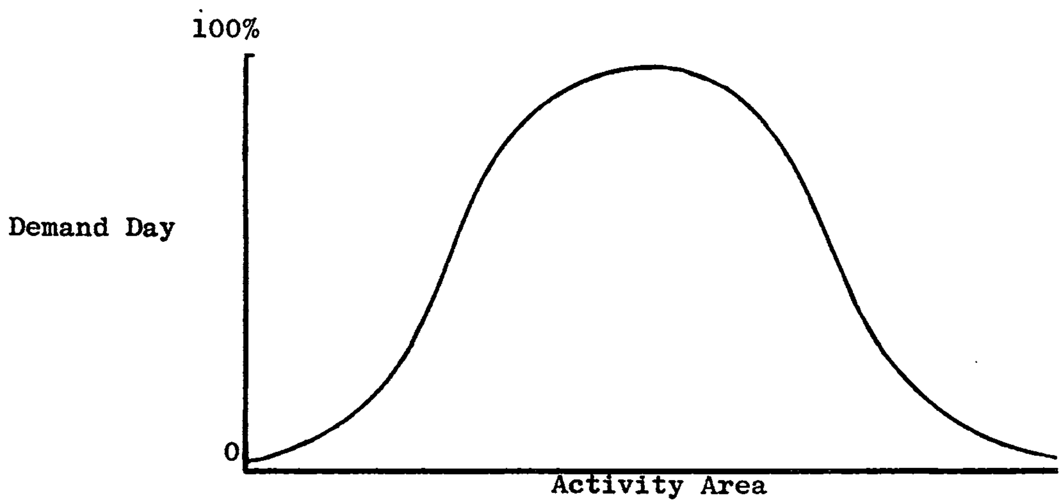

Figure 10. Record activity

An entire file will be equally distributed over all the available lanes. On each lane, the central area will contain the high volume items, with popularity tapering off to either side of the central area, i.e., to the ends of the tape.

Example

Assume a billing problem where an order is completely processed line by line. The stock consists of 100,000 items, each requiring 200 characters, or one twenty-word block. Suppose the stock items have the following demands:

(a) 10% of the items are ordered 80% of the time.

(b) 40% of the items are ordered 15% of the time.

(c) 50% of the items are ordered 5% of the time.

The file is made up as follows:

(a) The 10,000 items ordered most frequently are stored in blocks 0450 - 0550 across the 100 lanes.

(b) The 40,000 items ordered with intermediate frequency are stored in blocks 0250 - 0449 and 0550 - 0750 across the 100 lanes.

(c) The 50,000 items ordered least frequently are stored in the remaining blocks.

The average access to any item in the three classes is:

(a) $1.0 + \frac{100}{3} \times .046 = 2.5$ seconds

(b) $1.0 + \frac{500}{3} \times .046 = 8.7$ seconds

(c) $1.0 + \frac{1000}{3} \times .046 = 16.3$ seconds

Average access to any item in the entire 100,000 item file is:

$2.5 \times .80 + 8.7 \times .15 + 16.3 \times .05 = 4.1$ seconds

If the 100,000 items were stored on two DataFiles -- to allow room for expansion -- the access time would be cut to approximately 2.5 seconds.

A further favorable effect toward bell distribution can be achieved by designating the lanes near the center of the

DataFile as high activity lanes and the outer lanes as those for records of lower activity. Note that the computer is free for calculation or input-output during the entire access time. Thus, search time should not be added to calculate time, but should be partially absorbed by it.

7. <u>Static and Dynamic File Separation of Data</u>

In many problems such as inventory control, accounts receivable, and accounts payable, where a change is made to a record, the entire record is stored as a unit on magnetic tape. Usually only one or two items in the record are referred to and considerable amount of time is expended reading and writing unchanging information. If the static information and dynamic information are separated into two files, only the short dynamic file need be up-dated. When a periodic summarization is required, the two files may be run simultaneously (not merged) to yield a required print-out, including all the totals and the descriptive information.

The dynamic information might constitute only one-tenth of the total file, in which case only one-tenth need be read or written. Consider an integrated personnel file used for payroll labor distribution, personnel statistical work, etc.

<u>Items and Number of Characters</u>

Name	30	Address	20	City	15
State	5	Job Title	20	Dept.	5
Number	5	Section	2	Shift	1
Pay Rate	5	Dependents	1	Bond Code	1
Credit Union	6	Credit Union	4	Start Date	4
Ret. Bal.	7	Ret. Ded.	4	Education	1
Race	1	Height	2	Weight	3
Age	2	Last Dept.	5	Last Sect.	2
Date of Transfer	4	Security	1	Birth	4
Birthplace	30	Other	11		

TOTAL: 200 characters

To perform a payroll calculation with employee's name, hours worked, number, job code, and date as input data, a dynamic file might require:

Number	5 Char.
Dependents	1
Bond Code	1
Pay Rate	3
Tax to Date	6
Earnings to Date	6
Retirement Code	1
Credit Union Deduction	4
	27 Char.

In this hypothetical case, 27/200 or 13.5% of the personnel file is needed for payroll calculations. Only 13.5% of the file need be read or written when the paycheck is printed.

Periodically, a personnel review is made requiring the use of both the static and the dynamic file. In this case, the DataFile, by virtue of the numerous lanes, can divide a file into many sub-files which may be easily coordinated to produce the necessary consolidation.

SORTING ON THE DATAFILE

The DataFile, in combination with the 4000 word drum of the DATATRON, as a temporary storage device, provides a means for high speed sorting. Since each DataFile contains 50 tapes, the most readily apparent way to reduce sorting time is to block sort on the highest digits.

The drum is used as a 50 "pocket" sorting storage medium preparatory to searching and writing on the DataFile. The records must be less than 800 digits in length to make use of the drum and the 50 pocket method. The tapes are searched and written on only when one of the 50 pockets is full. This technique reduces the head movement and searching time considerably. An approximation for block sorting into 50 blocks to the DataFile is as follows:

$$\text{Total time to distribute (seconds)} = \frac{1.0 + (.046 \times 4)}{80} \times \text{Number of words in the file} =$$

$$.0148 \times \text{Number of words in the file.}$$

Following the block sorting of the entire file to the DataFile, each tape is read back onto the drum for fine sorting on the remaining digits. The advantages of this technique are:

a. The entire file is written onto magnetic tape only twice; once when distributed to the DataFile, and once after final sorting.

b. Fine sorting can be accomplished by comparison or by address approximation techniques, rather than by the actual value of the digits. The fraction of the file on each tape should be small enough (less than 150 blocks) to make a high speed drum sort possible.

GENERAL CONCLUSIONS

The DataFile provides a variety of methods of arranging file format. By correct choice of the type of file structure on the DataFile, access times may be reduced considerably. New approaches to certain problems are possible with DataFile, but not feasible with reel units.

When used skillfully, the DataFile and DATATRON makes possible processing speeds and techniques unmatched by any data processor, large, medium, or small.

APPENDIX I

COMPLETE AVERAGE RANDOM ACCESS FORMULA FOR THE DATAFILE

The formula for head and tape movement to locate any record on a DataFile unit may be expressed as

$$\text{Time in Seconds} = \frac{.000077(ac)}{b} + \frac{.005b}{d} + 0.5$$

where
- a is the number of records per file
- b is the number of lanes to be used
- c is the number of characters per record
- d is the number of DataFiles to be used

Explanation of Formula:

a. The time required for the tape to travel past the reading head to locate a particular record is as follows:

$$T = \frac{ac}{200b} \times \frac{.046}{3}$$

Where 200 is the number of digits per block and .046 is the time in milliseconds to travel a tape block.

Example: The average random access time for tape travel to locate any 200 character record of 100,000 records stored on 100 lanes is:

$$\frac{(100{,}000)(200) \times .046}{200(100) \times 3} = 15.3 \text{ seconds}$$

b. The time required for the read-write head to move to any required tape on which the record is stored is calculated by the formula:

$$\text{Time in seconds for head movement} = \frac{.005b}{d}$$

Where b is the number of lanes used,

d is the number of DataFiles used.

Example: Find the average random head movement for any record stored on any one of 100 lanes on one DataFile.

$$\text{Time in seconds for head movement} = \frac{.005(100)}{1} = 0.5 \text{ seconds}$$

c. .5 seconds is the time for the read-write head to release from its last position and latch-in at the designated position.

The combined formula is

$$\frac{ac(.046)}{200b(3)} + \frac{.005b}{d} + 0.5 = \text{Time in Seconds}$$

The simplified condensed formula is shown above. The number of characters per records, c, may be a multiple or sub-multiple of 200, 5, 10, 20, 25, 50, 100, 400 etc., or may be 40, 60, 80, etc., so that a block may contain a mixed number of records and some one record may be in two blocks. For example, with an 80 character record,

Block No. 1 contains 2-1/2 records
Block No. 2 contains 2-1/2 records

A record length with obvious disadvantages is one in which space is wasted, such as 53, 61, 79, etc.

EXAMPLES OF AVERAGE RANDOM ACCESS USING THE DATAFILE

EXAMPLE	NO. OF UNITS	NO REC. IN FILE	DIGITS PER REC.	LANES USED	BLOCKS PER LANE	AVG. ACCESS IN SECONDS
A	1	22,000	200	80	275	5.12
B	1	52,000	100	100	260	4.99
C	1	400,000	40	100	800	13.24
D	1	6,000	800	100	240	4.68
E	2	200,000	200	200	1000	16.30
F	3	18,000	2000	240	750	12.40
G	10	18,000	2000	1000	180	8.25

DATATRON

ELECTRONIC
DATA
PROCESSING
SYSTEMS

ElectroData
DIVISION OF BURROUGHS CORPORATION
460 SIERRA MADRE VILLA, PASADENA, CALIFORNIA

Printed in U.S.A.