PROJECT
WHIRLWIND

R-232

INFORMATION SORTING IN THE APPLICATION OF ELECTRONIC DIGITAL COMPUTERS TO BUSINESS OPERATIONS

by

HAROLD H. SEWARD

DIGITAL COMPUTER LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Copy

Report R-232

INFORMATION SORTING IN THE APPLICATION OF ELECTRONIC DIGITAL COMPUTERS TO BUSINESS OPERATIONS

by

HAROLD H. SEWARD

DIGITAL COMPUTER LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
Cambridge 39, Massachusetts

May 24, 1954

ACKNOWLEDGEMENT

The author is indebted to his thesis supervisor, Professor Charles W. Adams of the M.I.T. Digital Computer Laboratory, for his many helpful criticisms and suggestions. Also, to Mr. Jay W. Forrester, Director of the Digital Computer Laboratory, for making the facilities of the Laboratory available for this work, to John B. Bennett for editing the thesis, and to Joseph Kelley of Remington-Rand Inc., Norman Grieser of Underwood Corporation, and M. E. Femmer of International Business Machines for their helpful assistance.

The research reported was partially supported by Air Force Contract No. AF 19(122)-458.

Report R-232

ABSTRACT

Primarily, two methods of sorting are of interest in automatized business operations, digital sorting and sorting by merging. In the majority of applications, sorting may be achieved most economically with punched-card machines or magnetic-tape devices. In a system incorporating a general-purpose computing element it is generally faster and more economical to use special-purpose sorting equipment for the sorting operations. For applications involving large amounts of information and requiring rapid sorting operations, the use of a high-speed, high-density photographic storage may be practical. However, the processing time of the photographic medium should not be more than several seconds, and the information should be of such volume that the medium-processing time is not significantly large in comparison with the total read-record time.

Because it presents information of general interest, this thesis report, which has had only very limited distribution, is being issued as a Digital Computer Laboratory R-series report.

TABLE OF CONTENTS

Page

Chapter I - Introduction to the Sorting Problem

1.1 Description of a General Purpose Digital Computer 1

1.2 Speed Characteristics of a Digital Computer 2

1.3 Business Applications of a Digital Computing Equipment 2

1.4 The Large Scale Information Storage Problem and Sorting 4

Chapter II - The Methods of Sorting

2.1 Introduction 6

2.2 Digital Sorting 6

2.2.1 Machine-Sorted Cards 6

2.2.2 Manually Sorted Cards 8

2.2.3 Digital Sorting with Magnetic Tape Units 11

2.2.4 Time Analysis of Digital Sorting 12

2.2.5 Optimum Arrangement of Magnetic Tape Units in Digital Sorting 13

2.3 Sorting by Merging 14

2.3.1 Time Analysis of Merge Sorting 17

2.3.2 Optimum Arrangement of Magnetic Tape Units in Sorting by Merging 17

2.4 Initial Internal Sorting to Aid Merge Sorting 20

2.4.1 Internal Sorting by Merging Pairs 21

2.4.2 Internal Sorting by Finding the Smallest 23

2.4.3 Internal Sorting by Interchanging Pairs 23

2.4.4 Internal Sorting by Sifting 24

2.4.5 Internal Sorting by Partial Sort 25

2.4.6 Internal Sorting by Floating Digital Sort 25

2.4.7 Comparison of the Methods 28

TABLE OF CONTENTS (Continued) Page

2.5 Use of Internal Memory to Aid Digital Sorting 30

2.6 Combination of High-Speed and Slow-Speed Memories 30

Chapter III - Sorting with General Purpose and Special Purpose Equipment

3.1 Equipment Considerations 34

3.2 A Special Purpose Magnetic Yoke Sorting Element 36

3.3 A Special Purpose Photographic Device for Sorting 43

3.4 Comparison of Some Equipment Combinations Applied to a Sorting Problem 48

3.5 Summary 53

Appendix I - Characteristics of Magnetic Tape Units 55

Appendix II - Comparison of External Memory Media 57

Appendix III - Comparison of Internal Memory Media 58

Bibliography 59

LIST OF ILLUSTRATIONS

Figure Number		Page
1	Edge of "Keysort" Card.	9
2	Example of Digital Sorting with "Keysort" Cards.	10
3	Merging Two Strings.	15
4	Merging Pairs.	22
5	Finding the Smallest.	22
6	Interchanging Pairs.	22
7	Sifting.	22
8	Partial Sort.	22
9	Floating Digital Sort.	26
10	Tape Coding.	37
11	Block Diagram of Magnetic Tape Sorting Unit. . .	38
12	Schematic Diagram of Photographic Sorting Unit.	46
13	Time Comparisons for some Equipment Combinations Applied to Merge Sorting.	51
14	Comparison of External Memory Media. . . . . . . (Appendix II)	57
15	Comparison of Internal Memory Media. . . . . . . (Appendix III)	58

CHAPTER I

INTRODUCTION TO THE SORTING PROBLEM

1.1 Description of a General-Purpose Digital Computer[1]

A general-purpose digital computer may be divided into 4 basic elements: the memory or storage element, the arithmetic element, the control element, and the terminal equipment or input-output equipment.

The memory element consists of a number of storage locations or registers in which information may be stored or from which it may be extracted. Each memory register is designated by a number, its address. The information contained in a register is called a "word." A word may be either an instruction or data. An instruction consists of 2 parts: the operation code and the address of words which are to be used in executing the operation. Data is any information not used as an instruction.

The arithmetic element is an electronic accumulator with varying degrees of flexibility to permit addition, subtraction, multiplication, division, and various logical manipulations of words obtained from storage.

The control element consists of electronic counters and switches which take each instruction in proper sequence from the program stored in the memory and send appropriate signals to various parts of the computer to read or write in the memory, manipulate digits in the accumulator, etc.

The input-output element may contain a variety of equipment - teletypewriters, punched-card readers and punches, magnetic-tape units, paper-tape readers and punches, printers, film readers and recorders, etc. The purpose of the input-output equipment is to enter information into the computer and send out results in the desired form.

[1] See Reference 16

1.2 Speed Characteristics of a Digital Computer[2]

The foremost characteristic of a digital computer is its speed. The arithmetic and control elements perform additions, multiplications, etc.,at speeds up to several thousand operations per second. The memory element offers access to its stored instructions and data within several milliseconds in magnetic-drum memories and within several microseconds in magnetic-core memories. The input-output element, however, operates at a much slower speed due mostly to the mechanical motion required to link the computer to the outside world. Mechanical paper-tape readers and output printers operate at about 7 alphabetic or numeric characters per second although more costly models operate several times faster. Magnetic tape operates considerably faster at up to about 10,000 characters per second, while punched cards may be read at the rate of about 200 characters per second.

With such high internal and low terminal operating rates the electronic digital computer is usually applied most efficiently to problems characterized by relatively little input-output data and extensive computations. Scientific and engineering computational problems have such characteristics, and the need for solutions to these problems has been in the past the major impetus for the development of these computers.

Business applications of computers, on the other hand, involve vast amounts of input, output, and filed information which for the most part entail little computation.

1.3 Business Applications of Digital Computing Equipment

We might, for example, consider an inventory-control problem of electronically maintaining an accurate account of some 10,000 items in a

[2] See Reference 2

mail-order house. As each order is received, its stock number is manually inserted into one of several keyboards which transmit the information to the computer. The computer selects the storage location corresponding to the stock number, subtracts one from the "amount in stock" count found at the location, and returns to receive more information from the input keyboards. During off hours the computer is placed in another mode of operation during which each item's count is examined and checked against its reorder level; the results are printed out on an output typewriter. A person later examines the printed report and takes appropriate action to maintain the inventory.

To assign a general-purpose digital computer to a single problem such as this would be gross misuse of equipment since nearly all of the time is spent waiting for input information or output operations. The few simple arithmetic operations which are required might take less than 50 milliseconds for each stock number and utilize little of the computer's flexibility. Usually such simple operations as inventory control are solved quite readily and economically by punched-card systems or, if time is a critical quantity, by special-purpose high-speed computers such as the "Distribution"(or "Speed Tally"), built by Engineering Research Associates, Division of Remington Rand, for John Plain and Company, a Chicago mail-order house.[3,4] "Distribution" consists of a magnetic-drum memory, 10 keyboard inputs, and a printer output and operates in a manner somewhat similar to that described in the foregoing example of inventory control.

A similar example of inventory control is the "Reservisor" built for American Airlines by the Teleregister Corporation.[5] A large magnetic drum is used to store up-to-the-minute information on availability of

[3] See Reference 10

[4] See Reference 21

[5] See Reference 1

flight reservations. By punching keyboards at remote locations, agents may obtain information immediately on several flights and make desired reservations or cancellations.

Both the Distribution and the Reservisor serve to illustrate some of the characteristics of information processing, but they are unique in one important respect: the filed information is stored entirely within the high-speed memory element. Hence, after the input element has received its items of information, the corresponding filed item is made accessible for processing within a few milliseconds.

1.4 The Large-Scale Information-Storage Problem and Sorting

Unfortunately, however, most information in a business application is of such volume as to require a much more extensive storage capacity than the capacity required by the stock counts illustrated in the foregoing example of inventory control. A charge-account file for a department store might include coded characters for automatically printing a customer's name, address, description of purchases, type of payment plan for all purchases, amounts outstanding, etc.[6] Even in an inventory-control application, each item might contain such information as supplying firms and their addresses, amounts to reorder, seasonal history, and the like. An insurance-policy file might contain even more information for each filed policy.[7]

As the volume of data increases, storage in costly rapid-access memories becomes most impractical. Less expensive storage media such as punched cards, magnetic tape or wire, or possibly photographic film must then be considered. Because of their serial characteristics these media offer relatively low-speed access to information at random locations.

[6] See Reference 12

[7] See Reference 18

Several minutes may be required for a magnetic-tape of magnetic-wire unit to locate an item on a single reel of tape.

However, access to sequential items in such serial-storage media is considerably faster. Thus, if the order in which the information is to be used is known, arranging or sorting the items in that order will allow the information to be read in serially at a much higher rate.

Fortunately, in most instances this order is known. In inventory control, for example, the input items are desired in order by stock number, assuming that the inventory file is also ordered in that manner. If the input items are arranged in order, the computer may gain access to each ordered input item and filed item at the same time, requiring only one pass through the input and filed information. A census report, for example, might require that totals be computed and printed out in order by income brackets. However, the sorting operations necessary may required that a considerably high number of passes be made through the information in order to attain the desired arrangement of items.

CHAPTER II

THE METHODS OF SORTING

2.1 Introduction

Two basic methods of sorting appear practical for use with most present-day computer equipment. They are digital sorting and sorting by merging. Digital sorting is the same type of sorting as that used with punched-card sorters. Sorting by merging takes 2 or more ordered groups and orders them into a single group.

2.2 Digital Sorting

Digital sorting requires that the serially stored items be run through the computing unit once for each digit of the "key," the number by which the items are being sorted into ascending (or descending) order. On the first pass through, the least significant digit of the key of each item is examined and according to the value of the digit is sent to one of a number of output units. There is one output unit corresponding to each possible value of the digit.

2.2.1 Machine-Sorted Cards

For example, when punched cards are fed into a sorter for decimal sorting,[8] electric brushes detect the decimal value of the punched hole of a digit and send the card into one of 10 card pockets numbered 0 to 9. After all cards have been fed through they are removed from the pockets and stacked together in order according to their pocket numbers. These cards are again fed through the machine with the brushes reset to examine the next more significant digit of the key of each card. After the most significant digit has been examined the cards are in order.

[8] See Reference 4

Present sorters produced by International Business Machines, Inc. will process cards at the rate of 450 cards per minute per digit.

The process described above considered the use of only one sorter. More than one sorter may be used, and the time required is about inversely proportional to the number of sorters used. For example, if one sorter is used to sort 5000 cards on a 9-decimal-digit key, the time required is

$$\frac{\text{5000 cards/pass x 9 passes}}{\text{450 cards/minute}} = \text{100 minutes}$$

If 5 sorters are to be used, the following procedure might be employed. The cards are first distributed evenly among the 5 sorting machines and sorted on the most significant digit. The cards at the output of each digit in each sorter are then grouped with the cards of all other sorters to give 10 groups of cards corresponding to each possible value of the most significant digit of the key. If we assume that the cards are distributed fairly evenly among the 10 groups of cards, each sorter is then assigned 2 groups of cards which are then sorted separately on the other 8 digits. After all digits have been sorted upon, the 10 original groups are placed in order to yield the single ordered array of cards. With the even distribution assumed above, the time required is seen to be equal to 1/5 of the time required when only one sorter is used.

However, the distribution is more frequently uneven and causes the sorting operation to be held up by the sorter with the highest load. A possible alleviation of such a situation is to further break down the groups with another pass using the second most significant digit. This allows a more even distribution among the sorters to be obtained, but the number of groups (100) may become difficult to handle. Some of the groups could be placed together even though this could mean sorting on the digit again. The course of action chosen depends upon the particular problem

2.2.2 Manually-Sorted Cards

"Keysort" or other manually sorted cards[9] are sorted in a manner similar to automatic card sorters except that sorting is done on binary digits rather than decimal digits. (Binary digits have one of two values, zero or one. They are the basic numbers used in automatic systems; several binary digits may be coded to represent a decimal digit or letter.) These cards have holes punched along their edges and are coded either by punching a hole out to the edge to designate a ONE or by leaving the hole intact to represent a ZERO (see Fig.1).

In sorting, the cards are stacked, and a needle similar to a knitting needle is inserted through the hole corresponding to the least significant binary digit of the key. The needle is then lifted away from the stack bringing with it all ZERO cards and leaving behind all ONE cards since the needle slips through the cut-away portion of the ONE cards. For example, if the cards were arranged in the random sequence shown in Fig. 2.A1, the needle would be placed through the holes corresponding to the rightmost digit; lifting the needle away from the stack would remove the cards shown in B1; the cards in C1 would remain. The ZERO group (B1) is then placed in front of the ONE Group (C1) to form A2. The next pass of the needle is then made on the middle digit and results in B2 and C2. Similarly, the last pass is made on the most significant digit giving B3 and C3 which, when grouped together (A4), contain the cards in increasing order. Keysort cards may also be sorted by several sorters operating simultaneously if an initial breakdown sort is done as in 2.2.1.

[9] See Reference 4

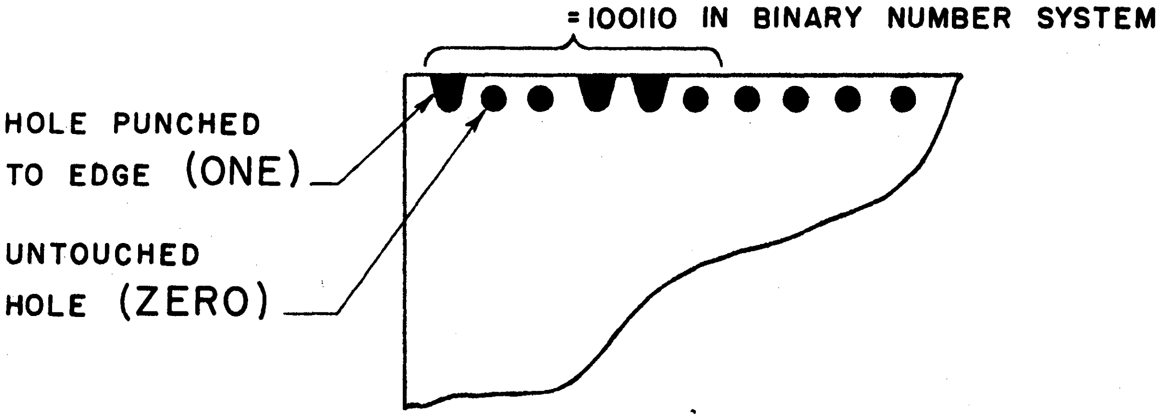

FIG. 1

EDGE OF A "KEYSORT" CARD

A1	B1	C1
101		
011	010	101
010	110	011
110	100	001
100		
001		

A2	B2	C2
010		
110	100	010
100	101	110
101	001	011
011		
001		

A3	B3	C3
100		
101	001	100
001	010	101
010	011	110
110		
011		

A4
001
010
011
100
101
110

Fig. 2 Example of Digital Sorting with "Keysort" Cards

From 500 to 1000 cards are usually handled at once with each binary digit taking from 5 to 15 seconds. The number of cards handled at once could unquestionably be increased to several thousand using an arrangement with a much longer sorting needle. The sorting rate would be considerably faster than machine-sorted cards, where each card must be examined in sequence. At present, however, needle-sorted cards are not used in conjunction with automatic machinery.

2.2.3 Digital Sorting with Magnetic-Tape Units[10,11,12]

A computer having magnetic-tape units as auxiliary memory elements may also be used for digital sorting. A decimal sort might be accomplished with 20 magnetic-tape units. As each item is read into the computer the least significant decimal digit of the key of the item is examined to determine upon which of the 10 output tapes the item is to be recorded. After the items have all been read in and recorded onto their corresponding output tapes, the tapes are rewound and read into the computer in order of their corresponding digits (i.e., 0, 1, 2 ... 9). As the items are received by the computer on this second pass, the next more significant digit is examined as before, and the item is recorded onto one of the other 10 tapes. On the following passes the 2 banks of 10 tape units alternate between the roles of input and output until all digits have been considered. After consideration of the most significant digit, the tapes are rewound, and the items are available for reading in the desired order.

Many tape units are capable of reading in both directions and recording in the forward direction. This property allows the sorting

[10] See Reference 11
[11] See Reference 15
[12] See Reference 20

process to take place with up to twice the speed of the preceding example since the rewinding time of a reel of tape is usually nearly equal to the reading or recording time. The process used with such tape units is identical to the former except that after each pass has been completed the first tape which is to be read is started up in the rewind direction and the computer begins reading this tape as it rewinds. The tapes are rewound and read in order of their corresponding digits, this order being decreasing on one pass and increasing on the next, depending upon whether the number of digits in the key is odd or even. After the final pass, the items on the output tapes may be read in order during the last rewinding operation.

In both of the above examples of tape sorting, only one of the input reels is read at a time. In view of this it is not necessary to use two banks of tape units, but rather one bank of tape units for the output and a single tape unit for the input. As soon as one reel is read into the input, it is replaced with the next reel manually. Reel changing is possible within 10 or 15 seconds in the Univac computer and constitutes only a fraction of the time required to run through an entire reel. One objection to manual reel changing is that the operator is quite likely to make mistakes which could seriously disrupt the operations involved, but such mistakes may easily be detected and announced to the operator by the computer program.

2.2.4 Time Analysis of Digital Sorting

The time required for digital sorting is seen to be the product of the time required to pass the entire volume of information through the control element and the number of digits in the key. Hence, if the range of the key is 0 to R, the time required for the sorting operation is

What [illegible]

$$T = NA \left[\log_d R \right] ^{*}$$

? T = [illegible]

where N is the number of items being sorted, A is the access time to read (write) sequential items in the memory, and d is the base or radix used in representing the key. In decimal card-sorting machines, d is equal to 10; in "Keysort" cards, d is equal to 2. When a computer is used in digital sorting, however, the key may be expressed in any base and easily converted to any other base. The base chosen depends upon the characteristics of the application and of the available equipment.

2.2.5 Optimum Arrangement of Magnetic-Tape Units in Digital Sorting

The optimum value of the base (number of output units) in a magnetic-tape digital-sorting element might be defined as that base which would result in a minimum value of the product:

(Time required for sorting operation) x (number of tape units required)

If reels are changed manually, the number of tape units is one input unit plus d (the value of the base or radix) output units, and the product desired to be minimum with respect to d is

$$\begin{aligned} & T(d+1) \\ &= NA \left[\log_d R \right] (d+1) \\ &= NA \left[\frac{\log_{10} R}{\log_{10} d} \right] (d+1) \\ &= f(N,A,R) \quad \frac{d+1}{\log_{10} d} \end{aligned}$$

* The notation $[x]$ means "the smallest integer greater than x."

Substituting the values of d = 2,3,4,etc., the product is seen to be minimum at d = 4; hence, the optimum arrangement would be one input unit feeding 4 output units.

If manual reel-changing is not to be used the number of tape units is 2d and the product to be minimized with respect to d is

$$
\begin{aligned}
&T(2d)\\
&= NA \left\lceil \log_d R \right\rceil (2d)\\
&= NA \left[\frac{\log_{10} R}{\log_{10} d}\right] (2d)\\
&= f(N,A,R)\,\frac{d}{\log_{10} d}
\end{aligned}
$$

Substituting d = 2,3,4,etc., the product is seen to be minimum at d = 3. The optimum arrangement of tape units without reel changing between passes is therefore 2 banks of 3 tape units each.

These results indicate that the sorting speed is increased more effectively by increasing the number of tape units used for each input and output rather than by increasing the base of the sort. That is, if 25 tape units are available and manual reel changing is used, the best deployment would be to use 4 parallel-operating tape units for input (i.e., one-fourth of each item is recorded on each tape; the tapes operate simultaneously) and 4 parallel-operating units for each of the 4 outputs. Conversely, a less effective arrangement would be to use one tape unit for input and the other 24 for output.

If simultaneously operating equipment is not practical, the next best choice is to use 5 independently operating 1-on-4 arrangements, each arrangement being assigned one-fifth of the items through the use of an initial breakdown sort. This alternative, although requiring a breakdown sort, is simple equipment-wise since the problems involved in synchronizing parallel reading or recording operations of tape units are avoided.

Exceptions to the optimum case might be found because discrete rather than continuous variables are involved, but in general the optimum arrangement will utilize the available equipment most effectively.

2.3 Sorting by Merging

Sorting by merging consists basically of taking 2 or more ordered groups ("strings") and merging them into a single ordered group (a single string). For example, consider 2 strings of items which are recorded on 2 input magnetic-tape units as shown in Fig. 3. Initially,

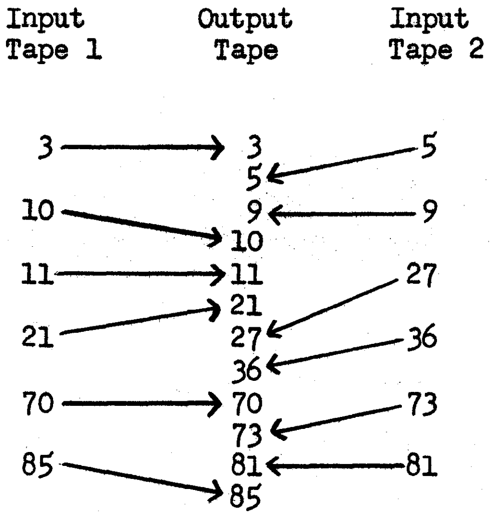

Fig. 3 Merging 2 Strings

the first items of each tape unit (3 and 5) are read into the computer and compared. The item with the smaller key (3) is then recorded on the output tape. Another item is then read into the computer from the tape unit which supplied the last output item (unit 1 in this example). The new item and the item not chosen for output on the previous comparison are then used in the next comparison. The process is repeated until both strings have been exhausted and merged into a single string and recorded on the output tape as shown in Fig. 3.

In the general case the number of strings on an input tape may be any number from one to the number of items on the tape. (In the latter case the items are arranged in the opposite order; hence each string is only one item long.) When there is a multiple of strings on the input tapes, the merging takes place in the same manner as in the foregoing example. For instance, if 2 input tapes are used, the first string of each input tape is merged with the first string of the other input tape to produce a single string which is recorded on one of 2 output tapes. Next the second strings of the 2 input tapes are also merged and put on the second output tape. The following output strings are recorded alternately on each of the 2 output tapes so that the number of strings on each output tape differs by no more than one after completion of the pass.

After all input items have been recorded onto the output tapes, the output tapes are rewound to become the input tapes; the input tapes are then used for the output tapes, and a second pass through the items is made in the same manner. Since pairs of strings are merged into single strings on each pass, the number of strings is seen to be halved on each pass until a single string results.

2.3.1 Time Analysis of Merge Sorting

In the above example it is seen that for each successive pass the number of strings is reduced by a factor of one-half. The number of passes is therefore equal to $\left[\log_2 S\right]$, where S is the initial number of strings. In the worst case, where the input items appear in the opposite order, the number of strings is equal to N (the number of items) and the number of passes equals $\left[\log_2 N\right]$.

As in digital sorting, merge sorting may take place with more input-tape units, and the number of passes required is seen to be $\left[\log_b S\right]$, where b is the number of input-tape units. Rewinding time may also be eliminated as in digital sorting if the tape units are capable of reading in both directions. To accomplish this the items are merged in increasing order on one pass and decreasing order on the next pass.

If manual reel changing is used between passes, the number of output units may be reduced to one by producing several successive reels (the number at least equal to the number of inputs) on the single output unit (i.e., if 2 input units were used with one output unit, the first half of the strings would be recorded on one output reel and a new output reel would then replace the former reel to record the second half of the output items. On the succeeding pass at least 2 reels are then available for the input units.)

Note that in order to sort any given arrangement of a number of items without reel changing, it must be possible to store all of the items on one reel of tape. Otherwise, manual reel changing is necessary. This applies to digital sorting as well as sorting by merging.

The number of strings will in all probability never appear as in the worst case, i.e., equal to the number of items. To estimate the expected situation, let us select one key at random from a long serial array of keys which are random fractions of unity.

The probability that an increasing sequence of at least n keys will occur in a designated direction from this key is

$$P(x_1 \leq x_2 \leq \ldots \leq x_n) = \frac{\int_0^1 \int_0^{x_n} \int_0^{x_{n-1}} \cdots \int_0^{x_3} \int_0^{x_2} dx_1 dx_2 \; . \; . \; . \; dx_n}{\int_0^1 \int_0^1 \int_0^1 \cdots \int_0^1 \int_0^1 dx_1 dx_2 \; . \; . \; . \; dx_n}$$

$$= \frac{\int_0^1 \frac{(x_n)^{n-1}}{(n-1)!} dx_n}{1} = \left.\frac{(x_n)^n}{n!}\right|_0^1 = \frac{1}{n!}$$

The probability that exactly n keys are in order from the selected key is the probability that at least n keys are in order less the probability that at least n+1 are in order or

$$\frac{1}{n!} - \frac{1}{(n+1)!}$$

The expected average length of an increasing sequence from the selected key is then approximated for a long array of numbers by the expression

$$\bar{n} = \sum_{n=1}^{\infty} n \left[P(n)\right]$$

$$= \sum_{n=1}^{\infty} n\left(\frac{1}{n!} - \frac{1}{(n+1)!}\right)$$

$$= \sum_{n=1}^{\infty} \frac{n}{n!} - \frac{1}{n!} + \frac{1}{(n+1)!}$$

$$= e - (e-1) + (e-2)$$

$$= e-1$$

This value is also the expected average length of a decreasing sequence occurring in the opposite direction from the selected key. Hence the expected average length of the string containing the selected key is

$$\begin{aligned} \bar{S} &= 2\bar{n} - 1 \\ &= 2(e-1) - 1 \\ &= \text{approximately } 2.4 \end{aligned}$$

the one being substracted because the selected key cannot be counted twice. With this value the number of strings in the random situation cannot be expected to be considerably less than the number of items, but the number of passes necessary for sorting randomly ordered items is likely to average around one less than the number of passes required in the worst arrangement of items.

2.3.2 Optimum Arrangement of Magnetic-Tape Units in Merge Sorting

The optimum value of the base (number of input units) in a magnetic-tape merge-sorting element might be defined in the same manner as that given for the digital tape sorter, i.e., the value which would give the minimum value of the product

(Time required for sorting operation) x (number of tape units required)

If manual reel changing is used, the number of tape units is b+1, and the product desired to be minimum with respect to b is

$$\begin{aligned} T(b+1) &= NA \left[\log_b S\right] (b+1) \\ &= f(N,A,S) \frac{(b+1)}{\log_{10} b} \end{aligned}$$

This result is similar to that found for digital sorting and the minimum arrangement is also found to be 5 tape units, one for output, and 4 for input.

If reel changing is not to occur during the sorting operation, the product to be minimized is

$$T(2b)$$

which is also similar to the case of the digital sorter. The minimum value occurs at b equal to 3; that is, 3 tape units each are used for input and output. It should be noted, however, that the 3-on-3 arrangement offers only about 5 per cent improvement over the case where 2-on-2 tape units are used.

If greater speed is desired, it may be best attained by splitting the load between several 3-on-3 arrangements which operate simultaneously.

2.4 Initial Internal Sorting to Aid Merge Sorting

In the description of sorting by merging, it was shown that the number of passes required was proportional to the log of the number of strings. Hence, if an initial pass is made through the data in order to sort it into strings of Q items in length, the maximum number of passes is reduced from $\log_b N$ to $\log_b \frac{N}{Q} + 1$, the one being the initial pass required to form the strings of Q items. This reduction of sorting time may be appreciable if the number of passes eliminated by the initial sort is a substantial fraction of the number of passes required without an initial sort. On the other hand, if a large number of passes are required, the effectiveness of an initial sort may be quite small and possibly uneconomical depending upon the characteristics of the available equipment.

2.4.1 Internal Sorting by Merging Pairs [13] (Fig. 4)

In this method the Q items are read into one section of the computer memory, and space is reserved for Q items in a second section of the computer memory. The computer first takes each pair of items in turn and merges them into a sorted group of 2 items in length. The resulting strings of two items are stored in the second section of storage as they are being formed. On the following pass, the strings of 2 items in length are merged into strings of four items, the resulting strings being placed in the table which originally held the Q items. The process continues with the 2 sections of storage alternating between the roles of input and output until the string size has been increased to Q items. The base of the sorting procedure (that is, the number of groups merged on each pass) might be increased above 2; however, the number of comparisons and program complexity increase in such a manner as to offer little if any reduction in sorting time.

If the amount of information in each item is in excess of one or 2 registers, it is usually advisable to sort the addresses of the items rather than to sort the items themselves. That is, the items are stored in locations Q.1 to Q.N in one section of storage, the addresses Q.1 to Q.N are stored in a second smaller section of storage, and space is reserved in a third section of storage for the addresses contained in the second section.

[13] See Reference 7

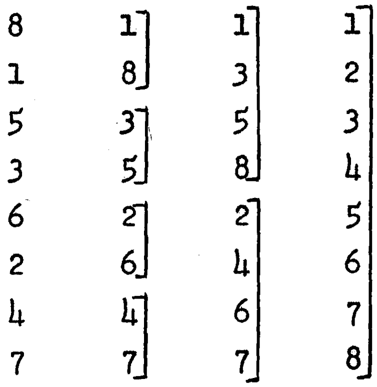

Fig. 4 Merging Pairs

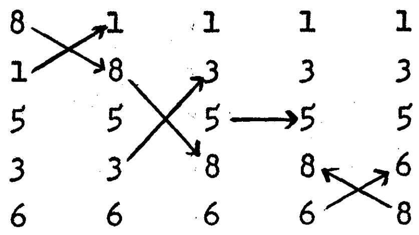

Fig. 5 Finding the Smallest

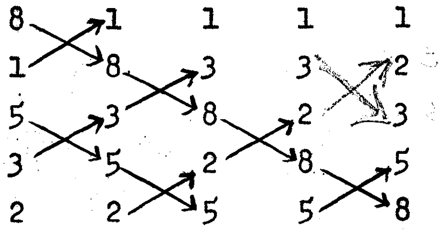

Fig. 6 Interchanging Pairs

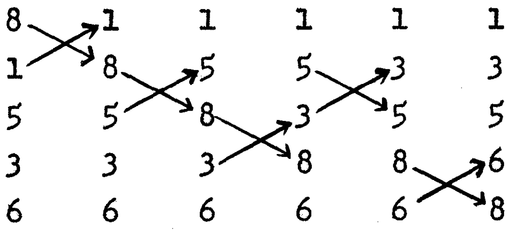

Fig. 7 Sifting

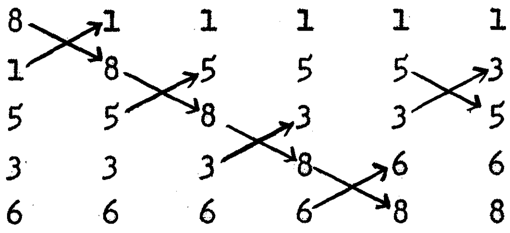

Fig. 8 Partial Sort

The addresses of the items are then merged in the same manner as explained above, but the items remain in their original storage positions. After the addresses have been sorted the items are available in order by referring to the sorted table of addresses. Sorting by addresses is usually much faster since only the address of each item is shifted during the process rather than the entire item. Space also is usually conserved since the extra storage required is that necessary to store 2Q addresses rather than that necessary to store Q entire items, each of which may require several storage locations.

Regardless of the original order, the number of items (or addresses) transferred to other storage locations is

$$N \left\lceil \log_2 N \right\rceil$$

2.4.2 Internal Sorting by Finding the Smallest[14] (Fig. 5)

This method consists of first finding the smallest of all items and storing it. The next smallest is then found and stored after the first. The process continues until all items have been considered. The extra storage required is negligible. The number of items transferred is N, but the number of comparisons is seen to be

$$= (N-1) + (N-2) + \ldots + 3 + 2 + 1$$

$$= \frac{N(N-1)}{2}$$

2.4.3 Internal Sorting by Interchanging Pairs[15] (Fig. 6)

The first 2 items, Q.1 and Q.2 are compared; if Q.2 is smaller they are interchanged, thus placing the item with the smaller key before

[14] See Reference 20
[15] See Reference 7

the other item. The same is done with Q.3 and Q.4, Q.5 and Q.6, etc., until the N items have been considered. On the second pass, the same process is used on pairs Q.2 and Q.3, Q.4 and Q.5, etc. The third and following odd-numbered passes use the same pairs as the first pass; the following even-numbered passes use the same pairs as the second; and the process continues until no interchanges are made on one pass. If the items are given in opposite to the desired order, the number of comparisons and interchanges is seen to be about

$$\frac{N^2}{2}$$

The extra storage required is negligible.

2.4.4 <u>Internal Sorting by Sifting</u>[16] (Fig. 7)

The items are examined in order until one is found which is smaller than the previous item. When one is found to be smaller than the preceding item, it is interchanged with each preceding item until it reaches an item which is smaller. After the item has been "sifted" upward to its proper position, the process continues, starting with the item which originally followed the last sifted item. The items are ordered after the sifting of the last item. In the worst case the number of comparisons and interchanges is about

$$= (N-1) + (N-2) + \ldots + 3 + 2 + 1$$

$$= \frac{N(N-1)}{2}$$

[16] See Reference 20

Again, the extra storage required is negligible.

2.4.5 Internal Sorting by Partial Sort[17] (Fig. 8)

The first and second items are compared and interchanged if the latter is smaller. The second and third items are then compared, the third and fourth, and so on down to the last pair. If any interchanges were made, the process is repeated. The number of comparisons and interchanges is also found to be at worst about

$$\frac{N(N-1)}{2}$$

2.4.6 Internal Sorting by Floating Digital Sort (Fig. 9)

The principle used in this method is identical to digital sorting except that in addition the items are counted before each pass to obtain the number of storage locations required for each value of the digit on the next pass. This precount of items enables the computer to make much better use of the available storage.

To illustrate, let us consider that N numbers (e.g., N = 6) which range from 0 to R are to be sorted into ascending order. These numbers are stored in locations X.1 to X.N (the "X-table"). An equal amount of storage is available in locations X'.1 to X'.N (the "X'-table"). These 2 tables alternate between the roles of input and output on each succeeding pass. Another table, the "A-table," has one register for each possible value of a digit. This table is initially used to count the number of times each digit appears in the numbers which are being sorted. After the count the counters are added in a manner which yields the proper address in which to store each item in the output table.

[17] See Reference 7

		Contents During First Pass (Units Digit)			Contents During Second Pass (Tens Digit)		
Address of Register	Initial Contents of Register	After Phase I	After Phase II	After Phase III	After Phase I	After Phase II	After Phase III
A.0	0	0	X'.0	X'.0	0	X.0	X.0
.1	0	0	X'.0	X'.0	0	X.0	X.0
.2	0	0	X'.0	X'.2	0	X.0	X.2
.3	0	2	X'.2	X'.2	2	X.2	X.3
.4	0	0	X'.2	X'.2	1	X.3	X.5
.5	0	0	X'.2	X'.2	2	X.5	X.5
.6	0	0	X'.2	X'.4	0	X.5	X.5
.7	0	2	X'.4	X'.5	0	X.5	X.5
.8	0	1	X'.5	X'.6	0	X.5	X.6
.9	0	1	X'.6	X'.6	1	X.6	X.6
X.1	22	Same	Same	Same	Same	Same	22
.2	42						26
.3	37						37
.4	88						42
.5	26						46
.6	46						88
X'.1	Optional	Same	Same	22	Same	Same	Same
.2				42			
.3				26			
.4				46			
.5				37			
.6				88			

Fig. 9 Floating Digital Sort

To explain the process in more detail it might be broken into 3 phases per digit pass. A decimal sort is assumed in the following.

Phase I: Each number is taken in order from the X-table and the appropriate digit is examined. (The digits on each succeeding pass are taken in increasing order of significance as in the former explanation of digital sorting.) The value of the digit, n, is examined, and the A-table counter in location A.n+1 is increased by one. Upon completion of this phase, the A-table contains a count of the numbers for each value of the digit (e.g., A.3 contains 2 after phase I of the first pass because of the presence of 2$\underline{2}$ and 4$\underline{2}$ in the X-table).

Phase II: The value of the address X'.0 is stored in the location A.0 in the A-table. This value is added to the count in A.1 and the result is stored in A.1. Following this the number in A.1 is added to the number in A.2, and the result is stored in A.2. This continues until A.8 is added to A.9, and the result is stored in A.9.

Phase III: A second pass is made through the numbers in the X-table and the same digit is considered. The value of the digit, n, is again examined, and the X'-table address in location A.n is increased by one. This resulting address designates a position in the X'-table; the item is then stored at this position.

After completion of all three phases, the next more significant digit is considered, and the X-table and X'-table exchange the roles of input and output.

As in the other methods, addresses may be sorted rather than the items themselves in order to eliminate the shifting of large items on each pass.

The number of passes is seen to be

$$\left[\log_d R\right]$$

where R is the range of the key and d is the number of counters in the A-table.

2.4.7 Comparison of the Methods

Except for sorting by merging and digital sorting, the time required for the above sorting methods is about proportional to the square of the number of items being sorted; hence these methods would probably not be used unless the data were arranged in a manner favorable to the particular method. For example, sorting with the interchange or partial-sort method might be especially profitable if the items were nearly in order. Or, if the memory has serial characteristics like a magnetic drum and if rapid access is possible to 2 successive items on the drum, a complete pass through the data might be made in one revolution of the drum using either the interchange or partial-sort method.

For the general case one might expect to use the principle of digital sorting or sorting by merging. The choice depends upon the nature of the key. If the range of the key is greater than the number of items, sorting by merging would be more favorable. If the opposite is true, digital sorting would probably be the better choice. The former case might occur when a number of names are to be sorted in alphabetic order; the latter case, when many items have the same key, as in an inventory control.

In the initial internal sort prior to the merge sorting of magnetic tapes, an attempt might be made to produce longer strings than those limited by the internal storage capacity. That is, as the internally sorted string is being placed on the output tape during the first pass, input items might, as storage space became available, be read in and placed with the items in the proper order. The probability that an input item's key will be greater than the last original output item (and will therefore be included in the string being formed) is seen to be

$$\frac{N-n}{N}$$

where n is the number of original items already on the output tape and N is the number of items in the initial internal sort. Hence after the N original items have been placed on the output tape, the number of items added to the original string might be expected to be

$$\sum_{n=1}^{N} \frac{N-n}{N} = N - \frac{N+1}{2}$$

which represents about a 50 per cent increase in string length.

Actually the above analysis does not represent the true situation; it assumes that the number of items in each new string is random. This might be true on the very first string formed, but after the first string the process tends to favor starting new strings with keys which are predominantly low. This effect is also seen to be cumulative to some degree as each new string is started; therefore, one could expect the average increase in string length to be something greater than 50 per cent.

2.5 Use of Internal Memory to Aid Digital Sorting

Because of the nature of digital sorting an initial internal sort is of no value. However, if the purpose of the sorting is to process the items with corresponding filed items on another tape the number of sorting passes may be reduced from

$$\left[\log_d R\right] \text{ to } \left[\log_d \frac{R}{F}\right]$$

where R is the range of values for the key, d is base (number of output tapes) of the sort, and F is the number of items on the file tape which may be stored in internal storage. To accomplish sorting with this reduced number of passes, the items are sorted on the digits of the integral part of the value of the key divided by F (i.e., $\frac{R}{F}$) rather than by the actual key value (R). After the sort, the items are arranged in order not by single items but by groups of items. Each group of F filed items is then stored in the computer, and all input items corresponding to the F filed items are read in sequentially and collated with the stored file items using random-access techniques.

This feature is most useful if the filed items are small in size, since F varies inversely with item size. Such small items might appear in an inventory application where the filed stock counts are to be collated with the sales recordings.

2.6 Combination of High-Speed and Low-Speed Memories

A high-speed memory may be used to increase the effective (average) access time of slower speed memories.

In slower types of storage the serial access time is generally much lower than the random access time; e.g., the serial access time to successive binary digits on a magnetic drum is as much as several-hundred times less than its random access time of 1/2 revolution of the drum. A memory using several-hundred magnetic discs is being developed by Dr. J. Rabinow[18] of the National Bureau of Standards. With this memory less than 2 seconds are required to move the read-record mechanism to the desired disc; the disc is then spun past the heads for one revolution during which time up to a half-million bits may be read or recorded.

The use of purely random access memories in a sorting operation offers very little in terms of time or economy. However, if a high-speed memory is used in conjunction with a slow-speed memory such as described above the effective access time can be considerably reduced. If the input and output items are required in a sequential fashion, as is the case in digital or merge sorting, several items may be read into the high-speed memory during one access to the low-speed memory. Hence, the access time to the low-speed memory is effectively divided by the number of items which can be taken into the high-speed memory on one pass.

To evaluate the above, let us consider that a low-speed, high-capacity memory of random access time A is to be used for merge sorting in conjunction with a high-speed, lower-capacity memory of negligible access time. Computing time is assumed negligible. The high-speed memory is capable of storing Q items. A fraction P of the high-speed memory is used to store the output items of the merging process. When this fraction

[18] See Reference 13

becomes full the items are transferred to the output section of the lower-speed memory. The remaining fraction of the high-speed storage, 1-P, is distributed equally among the b inputs being used. As each input fraction of high-speed storage becomes empty it is refilled by the input section of the low-speed memory. The effective input access time is therefore

$$T_i = \frac{A}{\text{Number of items received in one access to low-speed memory}}$$

$$= \frac{A}{Q\,\frac{1-P}{b}}$$

and the effective output access time is

$$T_o = \frac{A}{\text{Number of items sent out in one access to low-speed memory}}$$

$$= \frac{A}{QP}$$

Assuming other time requirements negligible, the time required to sort a given group of numbers is

$$T = N \left[\log_b N\right] \left(T_i + T_o\right)$$

$$= N \left[\log_b N\right] \frac{A}{Q} \left(\frac{b}{1-P} + \frac{1}{P}\right)$$

$$= f\ (N,A,Q)\ \frac{1}{\log_{10} b} \left(\frac{b}{1-P} + \frac{1}{P}\right)$$

The expression for minimum time with respect to P is seen to be

$$0 = f\ (N,A,Q)\ \frac{1}{\log_{10} b} \left(\frac{b}{(1-P)^2} - \frac{1}{P^2}\right)$$

$$0 = bP^2 - (1-P)^2$$

$$0 = P^2(b-1) + 2P - 1$$

$$P = \frac{-1 \pm \sqrt{b}}{(b-1)}$$

Substitution of the values $b = 2,3, \ldots$ etc., into the above equations shows that minimum time occurs when $b = 4$, $P = 1/3$ and hence

$$T = N \left[\log_4 N\right] A \frac{9}{Q}$$

These results would indicate that a considerable amount of high-speed storage could be used as described before a point of diminishing return was reached. The results are also identical if digital sorting is to be used in an analogous situation.

CHAPTER III

SORTING WITH GENERAL-PURPOSE AND SPECIAL-PURPOSE EQUIPMENT

3.1 Equipment Considerations

Because of the simplicity of sorting it might be suggested that sorting should be done with a smaller special-purpose computer rather than with a more expensive general-purpose computer. The choice might depend upon several factors.

The comparative speed and cost of the general-purpose computer and the special-purpose sorter may be the foremost factors. Underwood Corporation's Elecom 125[19], a moderate-priced general-purpose computer employing a magnetic-drum memory, includes a special-purpose sorting unit which uses part of the magnetic-drum memory for comparing the keys of the items being sorted. Two input and two output tapes are used in sorting by merging. The cost of the sorter is not significantly lower than the computer in this instance because of the low cost of the computer. However, the over-all speed characteristics are considerably increased since programmed sorting in the general-purpose section of the computer is much slower than in the special-purpose sorter and in addition the computer is left free to process other information.

The type of work and its scheduling may have an important bearing on the sorting methods. If the application is such that the computer is often left idle while awaiting a sorting operation, it might be more feasible to sort with the computer rather than with a special-purpose sorter. On the other hand, if a reasonable schedule is possible the computer may be kept well occupied while a special-purpose sorter orders information for later processing.

[19] See Reference 19

Another alternative is to place all operations under the control of a large high-speed general-purpose computer using a sizeable number of high-speed magnetic-tape units. The computer would necessarily be equipped with buffer storage which would allow computational work to be carried out while information is being transferred between the several tape units and buffer storage. The desired amount of buffer storage required is that amount necessary to maintain nearly continuous information transfer between the tape units and buffer storage and at the same time allow the computer to be efficiently time-shared between computational work and the transfer of information between the computer and buffer storage. When a high-speed computer is used in such an arrangement, the cost of necessary buffer storage is a small part of the cost of the tape units involved. In some computers such as Remington-Rand's Univac, a portion of the regular memory is set aside for buffer storage, and sorting is done with magnetic-tape units under the control of the computer.

The operation of the control element of a high-speed general-purpose computer can be considerably more economical if the computer can be given a reasonable supply of work. This might be effected by using multiple input-output units to increase the information flow or by supplying some items which require more processing time and less input-output time. More complex problems such as those found in scientific and engineering applications might be carried out while the input-output operations required for sorting or tabulating are taking place. A more reliable and flexible system may result; manual procedures such as real changing may be ordered in advance and automatically checked; in event of a mistake the computer could designate the trouble to an operator without stopping the processing of other items. However, the problem of scheduling the work might easily outweigh all advantages.

3.2 A Special-Purpose Magnetic-Tape Sorting Element

In order to obtain some estimate of the cost involved in a special-purpose sorting device, the block-diagram design of a merge sorter is outlined and described below.

The tape is coded in the following manner: one channel provides the synchronizing pulses which signal that a new line of pulses has appeared at the reading heads. Another channel stores the key upon which the items are being sorted. The remaining channels store the block of information following each key (the key channel is also used for storing this block of information since the key is stored ahead of the block on the tape--see Fig. 10).

A "key mark" is recorded at the end of each item block. This mark signals the control element of the sorter to prepare for the next comparison of keys.

There are 4 main parts to the sorting unit (Fig. 11). One is the magnetic-core static-delay lines which store the keys of the next blocks of information recorded on each input tape.

The second part is the comparing element which consists mainly of 3 flip-flops for each input tape unit. The comparing element is used to choose that key which is the smallest of all keys equal to, or greater than, the last key recorded on the output tape. If the last key recorded is greater than all keys, the smallest key is chosen to begin a new string of items.

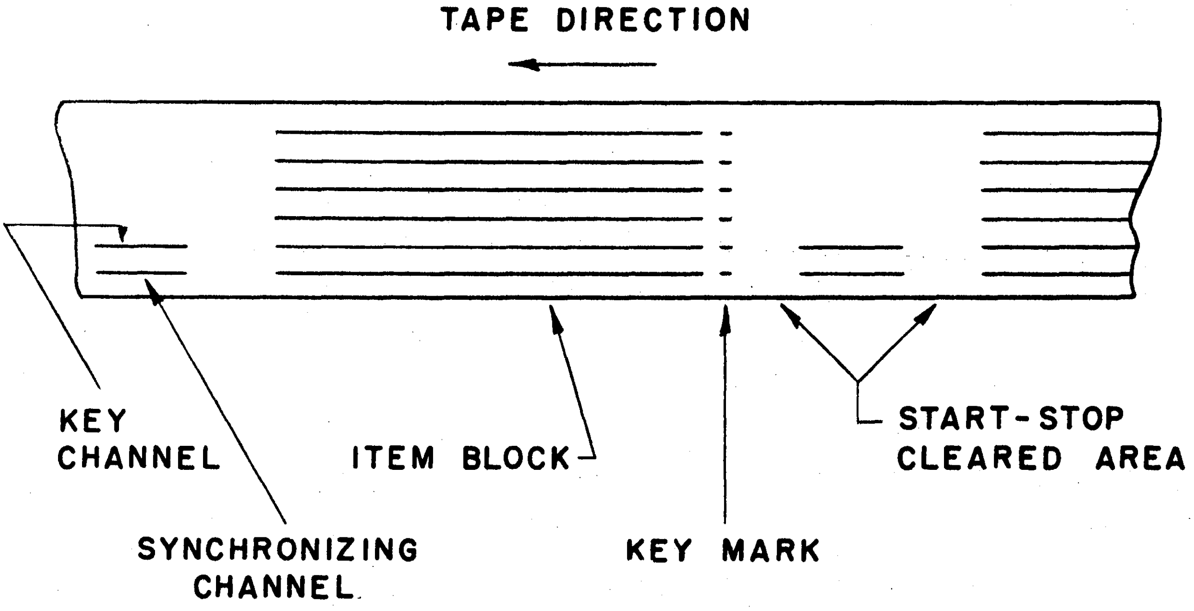

FIG. 10

TAPE CODING

A-58845

D-47119

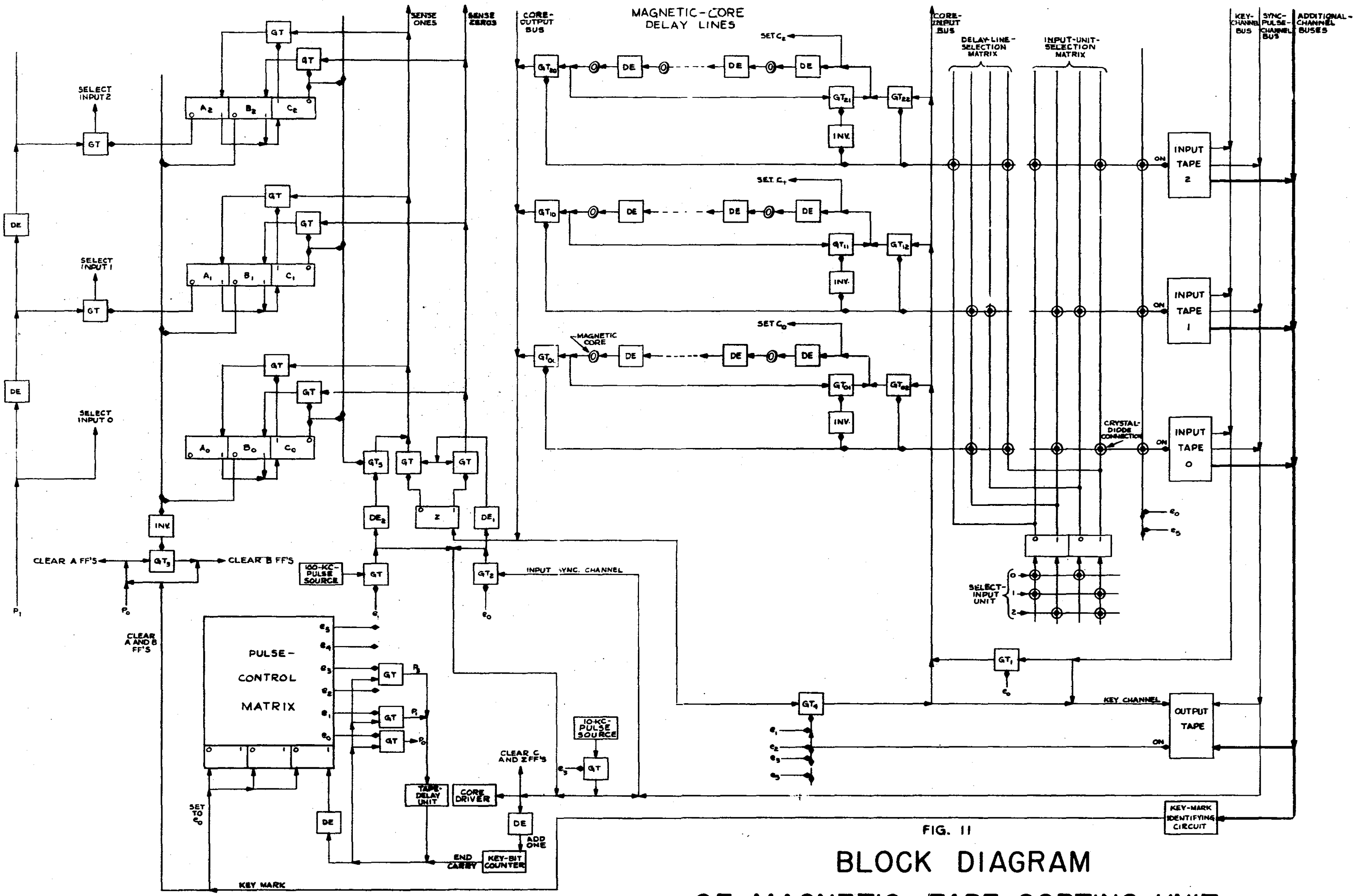

FIG. 11

BLOCK DIAGRAM OF MAGNETIC-TAPE SORTING UNIT

The third main part of the sorter, the pulse-control matrix, controls the sequencing of the sorter's operations.

The fourth main part of the sorter is the unit-selection switch. Each position of this switch causes a different input tape unit to be selected by the input-unit-selection matrix. An identical matrix, the delay-line-selection matrix, operates in parallel with the input-selection matrix and is used to select the delay line corresponding to the selected input unit.

For purposes of explanation, assume that the pulse control is set to e_5; hence, one of the input tapes is in the process of recording one of its information blocks onto the output tape via the three buses shown. The key of the item being recorded is stored in its corresponding core register.

After the last line is recorded, the key mark appears, is recorded, and is also sensed by the key-mark identifying circuit. The identifying circuit sends out a pulse which sets the pulse control to e_o and also clears all A and B flip-flops in the comparing element. Since voltage e_5 has been lowered, the output tape is shut off and begins decelerating to a stop.

The selected input tape continues reading; in addition, e_o opens GT_1 which connects the key-channel bus to the core-input bus. GT_2 is also open, placing pulse control under the control of the synchronizing-pulse channel of the input tape.

When the first synchronizing pulse arrives thrqugh GT_2, it clears all C and Z flip-flops to the ZERO position and pulses the core driver which advances all delay lines one bit. In the nonselected delay lines,

GT_{n1} is open. Hence, a ONE coming off the end of the delay line will be cycled into the recording and at the same time will set its corresponding C flip-flop to the ONE position.

The delay line and C flip-flop for the selected unit are similarly recorded, except that the ONE comes from the core-input bus rather than being cycled from the other end. A ONE coming off the output end of the selected core line is fed into the core-output bus rather than recycled. This will cause the Z flip-flop to register a ONE. Hence, the key of the last item recorded on the output tape is read serially into the Z flip-flop, and the keys of the next items on each input tape are read serially into the C flip-flops, the more significant digits appearing first.

After the first significant digit has been recorded in the C and Z flip-flops, the synchronizing pulse appears at the output of DE_1 and senses both Z gates. One of three occurrences may then change the status of a key:

1. If Z is a ONE, all of the C's are sensed at the ZERO gates. If a ZERO gate is on, this means that the corresponding key is <u>less than</u> the last key on the output tape. If so, the corresponding B flip-flop is set to the ONE position to record this fact.

2. If Z is a ZERO, all of the C's are sensed at the ONE gates. If a ONE gate is on, this means that the key is <u>greater than</u> the last key on the output tape, and the corresponding A flip-flop is set to the ONE position to record this fact.

3. If both Z and C are the same, no change takes place.

In either of the first two cases, the ONE output of the A or B will cause the corresponding C to be locked in the ONE position. This is done to avoid setting a B if its corresponding A has been set previously on a more significant digit.

After each such digit comparison, the delayed synchronizing pulse indexes the key-bit counter. If the last bit has not yet been compared, the control element waits for the next synchronizing pulse to start another digit comparison.

After the last synchronizing pulse has arrived, the key-bit counter sends an end carry to the pulse control. This yeilds a pulse P_0, which clears all A flip-flops and also all B flip-flops if GT_3 is on. GT_3 is on only if no B's are in the ZERO position (i.e., all keys have been found to be less than the last key on the output tape, and a new string must be started).

The pulse-control matrix has now been set to e_1; hence the absence of e_0 causes T_1 to deselect the input-tape unit and also switches control from the synchronizing pulses to the 100--kc--pulse source. The key digits are then cycled in the same manner as before except that the selected delay line cycles through GT_4, rather than being destroyed. Z is read into, but the result is not used.

After each bit is read into its corresponding C, the delayed pulse is received from DE_2. This pulse senses GT_5; if any C is in the ZERO position, the "sense ONE" line is pulsed, and all C's in the ONE position have their corresponding A's set and locked in the ONE position.

After all digits have been cycled through in this manner, only the lowest key (or keys) has its corresponding A in the ZERO position. At this time the key-bit counter gives an end carry which results in pulse P_1. This pulse senses the ZERO gates of each A in sequence and causes the highest-numbered eligible unit to be selected by the unit-selection switch. P_1 also initiates a delay to allow the output-tape unit to accelerate to recording speed after being started by voltage e_2.

After the acceleration time has elapsed, the tape-delay unit indexes the pulse control to voltage e_3. The output tape continues moving, and the slower 10--kc--pulse source begins recording in the synchronizing channel of the output unit. The selected key is recorded simultaneously and also is recycled into its delay line.

Upon recording the entire key, the end carry from the key-bit counter indexes the control switch and causes P_3 to initiate another tape delay for decelerating the output-tape unit. When the delay pulse again indexes the control switch, e_5 turns on the output tape and the selected input tape, and a block of information is again transferred after the tape has accelerated to speed.

The tape-delay unit shown is mainly for ease of explanation. Actually a more economical and accurate method would be to use the key-bit counter for the delay unit. Slightly more involved timing could also reduce acceleration time considerably by accelerating the units during, rather than after, the previous operations. Also, the magnetic-core registers might be eliminated if a section of the main computer memory were available for storing the information.

The design indicates that the cost of the control element for a special purpose sorter is probably a small fraction of the cost of the magnetic-tape units which are normally upwards of $8000 apiece. In view of this, it might be concluded that auxiliary sorting equipment would be most desirable when a moderate-speed computer (e.g., magnetic-drum memory) is used for the general-purpose work and that, when a high-speed computer is available, the choice between special-purpose and general-purpose equipment for sorting is not too definite.

3.3 A Special-Purpose Photographic Device for Sorting

Another approach to the sorting problem, although it is generally considered most uneconomical, is that of the construction of a vast high-speed memory which is perhaps hundreds of times larger than those now in use. Clearly, the cost of such a memory if constructed with Williams tubes, mercury delay lines, or magnetic cores is out of the question. However, it may be noted that in merge sorting or digital sorting within a random-access memory, the items are read from one table, processed in a small amount of storage, and recorded in a second table. The 2 tables are read and recorded into alternately until the sorting is complete. Since the time involved in this transfer of information is in the order of seconds, it may be feasible to use photographic media for storage of the 2 tables without seriously affecting the time factors.

The developing time may not be at all prohibitive if a large volume of information is transferred on each pass. It is possible to process some photographic media and project an image from them within a fraction of a second.[20] A significant disadvantage of photographic

[20] See Reference 17

recording is that the media are not available for rerecording after use; in sorting, the results of each pass would have to be disposed of after one reading. Obviously, photographic-film recording is economically impractical for sorting involving anything but a very small number of passes. However, if mechanical motion of the read-record equipment relative to the recording media is avoided, a considerably higher recording density is available with photographic media. As many as one million bits may be easily recorded in a square-inch area.[21] With such a density, the contents of a 1200-foot, 6-channel magnetic tape with a recording density of 200 bits per inch could be recorded on a photographic surface about 4 inches square.

The reading of the surface could be done most effectively with cathode-ray tubes (CRT's). This has been done successfully at frequencies above 1 megacycle per bit for successive bits along the surface.[20, 21] Random access to the first bit of a group, however, is something less than about 40 microseconds with electrostatic deflection. The random-access time to a 100-bit item is therefore

$$100 + 40 = 140 \text{ microseconds}$$

This may be effectively reduced by transferring more than one successive item at a time to a higher-speed memory, if available. Recording as well as reading with a CRT is likewise possible at the above frequencies.

[20] See Reference 17

[21] See Reference 9

A possible schematic design of such a machine is presented below (Fig. 12). The storage capacity is about equal to that of one standard reel of magnetic tape.

Two binary digit-to-analog decoders, each containing 10 binary digits, supply the horizontal and vertical deflection voltages to 2 identical banks of 16 cathode-ray tubes. Hence, each CRT contains about a million bits or about 10^3 times the capacity of a Williams tube. Each bank has associated with it a photographic plate and automatic developing equipment. On the other side of the plate from the CRT's is a photomultiplier tube which acts as a sensing element. One bank of CRT's is used for reading information from the plate corresponding to the bank; the other bank is used for recording information on its plate. Each bank is capable of either reading or recording, depending upon the state of its photographic plate.

The bank in the reading state contains a developed plate. This plate is read by (1) setting up the deflection voltages, (2) pulsing the beam of each CRT in sequence, (3) sensing the transmission of the beam of each CRT through the developed plate and to the photomultiplier tube (PMT) on the opposite side of the plate.

The bank in the recording state contains an initially unexposed plate. This plate is recorded on by (1) setting up the deflection voltage and (2) pulsing each CRT (in parallel) which is to record a ONE.

The general- or special-purpose computer using this storage medium for sorting would read small amounts of data out of the reading bank, conduct sorting operations within computer storage, and then record the data onto the undeveloped plate of the recording bank. After all data

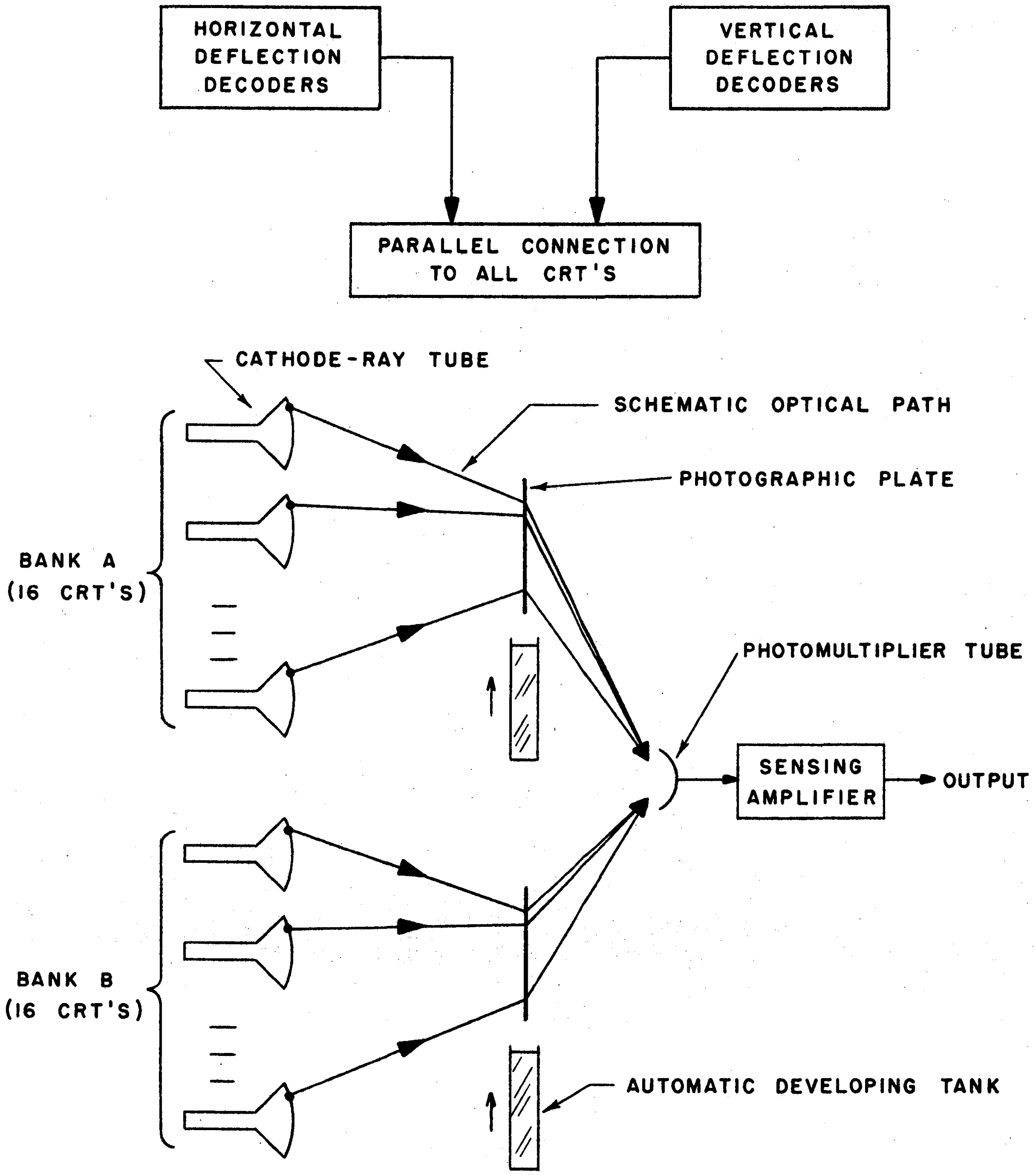

FIG. 12

SCHEMATIC DIAGRAM
OF PHOTOGRAPHIC SORTING UNIT

has been processed and transferred to the recording plate, this plate is automatically developed without being detached from its supports. The reading plate, having served its purpose, is ejected and replaced by an unexposed plate. The reading bank now becomes the recording bank and vice versa. Processing and transferring continues for several cycles until sorting has been completed.

To avoid any errors due to minute flaws in either the CRT or photographic surfaces, each bit might be recorded by 2 CRT's on 2 separate areas of the plate. Parity checks or self-correcting codes might also be used. The time required to transfer one table to the other is something less than 40 seconds. This is based upon an item 100 bits in length which is read in a period of 140 microseconds and recorded in 50 microseconds (parallel operation is possible in the recording phase). The developing interval is assumed to be about 8 seconds. The effective access time may be reduced, however, by a factor of up to 10 by using parallel sensing elements and by reading and recording several items at each access to the memory. Another possibility would be to have an additional pair of photographic plates served by the same CRT's. While one pair is waiting for the developing process, the other pair may be in use. A shutter provided with each bank would isolate the bank being developed. In view of the time involved, aging of the CRT's and deflection equipment should for the most part produce negligible variations in beam positioning between the recording and reading of information. Atmospheric and temperature effects will similarly be reduced to a minimum because of the short duration between recording and reading.

If the items to be sorted contain a considerable number of digits other than the key it may be more economical to have only one bank of the described arrangement and to sort the addresses of the items within a general or special-purpose high-speed memory. After the addresses are sorted the items are available in order from the photographic plate. This requires only one plate for each complete sorting operation, but it also requires that the high-speed memory be capable of handling all of the addresses involved.

The cost of such a photographic device is high, especially as its use is most likely restricted to sorting. Moreover, the input-output rate should necessarily be high to keep the machine in efficient operation. This might be effected with multiple units operating in parallel and/or higher-speed elements such as film readers and recorders, but the cost of this terminal equipment may be high.

However, for an application demanding a high-speed sorting process of this magnitude, such a device might well prove to be a more economical solution than that of using several tape sorters or other equipment operating in parallel.

3.4 Comparison of Some Equipment Combinations Applied to a Sorting Problem

The results of the foregoing analyses may be used for comparing 2 basic equipment combinations applicable to a sorting operation. Each combination used in the comparison will include a high-speed memory (e.g., mercury delay lines, magnetic cores) capable of storing about 200 items of 100 bits each. (It is assumed that strings of 100 items have been initially formed with the high-speed memory.) All combinations also include a high-capacity auxiliary memory.

The first combination is that in which the access time to the items of the high-capacity memory is mainly the sequential access time. This characteristic is present when sorting is done with basically serial memories such as magnetic tape or punched cards. The second combination is that in which the access time to items is mainly the random-access time. This would include magnetic drums or discs which cannot be stopped between successive items because of the high mechanical inertias involved in their operation. The high-speed memory will be used as described previously in order to reduce the effective access time of the latter combination.

In the former combination the effective access time to sorted items is defined as

$$\frac{\text{Sorting time required}}{\text{Number of items to be sorted}}$$

and (neglecting the time required for forming the initial strings of 100 items each) is equal to

$$\frac{n\left[\log_b \frac{n}{100}\right] A_s}{n} = \left[\log_b \frac{n}{100}\right] A_s$$

where n is the number of items to be sorted, b is the number of strings being merged together at once, and A_s is the serial access time to items in the high-capacity memory. The value of A_s for an International Business Machines (IBM) collator used for merging punched cards is 130 milliseconds. A_s would average about 2 milliseconds for a magnetic tape with the following characteristics:

Speed - 100 inches per second;

Recording density - 100 bits per inch;

Number of information channels - 6;

Fraction of tape used for acceleration, deceleration area - 1/6.

In the latter combination the effective access time to sorted items has been previously shown to be

$$\left[\log_4 \frac{n}{100}\right] \frac{9}{Q} A_R$$

where Q is the number of items which may be contained in the high-speed memory (200) and A_R is the random access time to items in the high-capacity memory. In a large drum of about one-million-bits capacity, A_R might average about 16 milliseconds. In the Rabinow disc memory, A_R is expected to be about 1.5 seconds. The photographic device as described previously would have an expected A_R of about 200 microseconds although, unless the high-speed memory is several times faster than this device, little reduction of the effective access time can be expected from the use of the high-speed memory.

The sorting times required with the above equipment for various numbers of items are presented in Fig. 13. The underlined values indicate that the capacity of a single memory unit has been exceeded.

The IBM punched-card collator used for merging with a sorting base b = 2 is the most time consuming. However, items of 3 times the size may be sorted without increasing the sorting time, since only about a third of a card's capacity is used with the 100-bit item being considered. In addition, punched cards offer comparatively fast access time to random locations in an ordered file. Although this access is manual it is often necessary in business applications, and the procedure is often preferable to searching through a long reel of magnetic or photographic records with expensive reading and recording equipment. Clearly, many considerations other than sorting time must be considered in the choice of a storage medium.

	n = number of items being sorted				
b = base of sort	K^4 items	K^5 items	K^6 items	K^7 items	(K = 10)
2	140 sec	2200	$\underline{28K^3}$	$\underline{340K^3}$	Sorting time for magnetic-tape units $T = n\left[\log_b \frac{n}{100}\right] 0.002$ sec
4	80	960	$\underline{14K^3}$	$\underline{180K^3}$	
8	60	800	$\underline{10K^3}$	$\underline{120K^3}$	
16	40	600	$\underline{8K^3}$	$\underline{100K^3}$	
2	$9.1K^3$	$143K^3$	$1.8K^6$	$22K^6$	IBM collator $T = n\left[\log_2 \frac{n}{100}\right] 0.13$ sec
4	$2.7K^3$	$33.8K^3$	$473K^3$	$\underline{6K^6}$	Magnetic-disc memory $T = n\left[\log_4 \frac{n}{100}\right] \frac{9}{200} x 1.5$ sec
4	28.8	$\underline{360}$	$\underline{5.05K^3}$	$\underline{64K^3}$	Magnetic-drum memory $T = n\left[\log_4 \frac{n}{100}\right] \frac{9}{200} x 0.016$ sec
4	8	96	$\underline{1.4K^3}$	$\underline{18K^3}$	Photographic device (total read-write time) $T = n\left[\log_4 \frac{n}{100}\right] 0.0002$ sec
4	32	40	$\underline{56}$	$\underline{72}$	Photographic device (total developing time) $T = \left[\log_4 \frac{n}{100}\right] x\ 8$ sec
	T = time in seconds				

Fig. 13 Time Comparisons of Some Equipment Combinations Applied to Merge Sorting

The use of magnetic discs for storage is seen to be only about 4 times as fast as the IBM collator even though the access time is effectively reduced to about 5 per cent of its normal value through the use of a high-speed memory. However, as with punched cards, the random access time and capacity of this medium make it most useful in applications other than sorting.

The large magnetic drum is shown to be about 100 times as fast as the disc memory in a sorting application. However, in view of the cost of the equipment involved, the comparatively low storage capacity of the medium,and the availability of more suitable media, such equipment would not be desirable solely for sorting operations.

The magnetic-tape-unit arrangement is shown to be from about 1/6 to 1/3 as fast as the drum arrangement and about 100 times as fast as the punched-card machine when 4 input tapes are used. The capacity of a single reel is exceeded only by that of the magnetic-disc memory unit. Moreover, extra capacity is easily available because of the low cost of the tape and ease of manual reel changing. These advantages, together with the relatively moderate cost of a magnetic-tape unit (upwards of $8000), have resulted in the recent trend in the use of fast tape units for high-speed information-processing systems, especially where the emphasis is on flushing the information through the control unit rather than providing fast and versatile arithmetic and control elements.

The photographic-storage unit described is shown to be over 15 times as fast as the tape units used in the $b = 2$, $N = 10^5$ arrangement which requires 4 tape units without manual reel changing between passes. (However, it is shown that developing time seriously retards speed of the photographic device if a small number of items is being sorted.) Hence, it would necessitate in the vicinity of 60 tape units to compete with the speed of the photographic device if the latter were provided with a substantial number of items. However, as noted before, the speed of faster computer elements such as the photographic device is often rendered less effective by the bottleneck in information flow caused by the slower input-output equipment.

In digital sorting the above results are similar proportionally, except that the IBM card sorter rather than the collator is used, and the former is seen to be about 4 times as fast in a comparative situation (i.e., N = R).

3.5 Summary

A sorting operation is solved most effectively through the use of either present punched-card methods or magnetic-tape units. The difference in economy of the two media is indefinite because complete cost estimates are not yet available, but the difference is estimated to be small. Other features such as the speed and access characteristics would probably be the more determining requirements in a choice between the two media for a particular application.

A high-speed general-purpose computer, if available, can add appreciably to the speed of a sorting process, but (because of the economic factors involved) a computer would usually not be employed for sorting if any other data were ready for processing. If a computer is not available, the control of a sorting operation is comparatively inexpensive and sometimes even faster with special-purpose control units.

The advantages of photographic storage may be used effectively in sorting operations requiring high speed. However, the storage must (1) be of considerable capacity to minimize the effect of developing time and (2) be equipped with terminal equipment capable of providing the device with a reasonably high rate of information transfer.

Signed HH Seward
Harold H. Seward

Approved CW Adams
Charles W. Adams

APPENDIX I

CHARACTERISTICS OF MAGNETIC-TAPE UNITS

The storage medium used is a magentizable tape of metalor coated plastic. The tape is stored on reels in the same manner as movie film. The tape is moved past stationary reading (recording) magnetic heads which read (record) in one or several channels along the tape. The number of channels is usually less than 7 or 8, since the tape tends to skew as it becomes wider and the pulses in adjacent channels appear at the recording heads at different time producing synchronization difficulties. The width of the tape is usually less than 3/4 inch, depending upon the number of channels, and the length ranges usually from 800 to 1200 feet. Longer tapes involve larger reel inertias which demand higher powered diving units and increase the tape acceleration and deceleration time.

The information is recorded in "blocks" which have spacings of 1/4 to 1 inch on the tape between them in order to provide blank areas of tape in which the tape may be accelerated or decelerated to the necessary speed. The time required for acceleration (deceleration) is about 5 to 10 milliseconds in faster tape units. The pulses are recorded at a density of about 100 to 200 per inch, and the tape travels at speeds up to about 100 inches per second. Considering the above characteristics a reading (recording) rate of about 50,000 binary digits per second is not too optimistic if the tape is not accelerated and decelerated excessively. This rate is high but not too well matched to the high processing rates of some of the faster computers; especially when short processing programs are being used, as is so often the case in business applications.

One method used to minimize the time requirements of a computer using magnetic-tape units is through the use of "buffer" storage.

Buffer storage is a small amount of storage used in the transfer of information between the computer and tape units. Information may be transferred in either direction between the computer and the buffer-storage element at rates near or equal to the speed of the computer memory. After this transfer of information between the buffer and computer memories, the buffer is used to transfer information between itself and the tape units at the slower rates required by the latter. This buffering of information frees the computer to perform other operations while information is being transferred in either direction between the buffer element and the tape unit. Other tape units also having buffer elements may be operating simultaneously.

APPENDIX II

COMPARISON OF EXTERNAL MEMORY MEDIA[22]

Media Only

Technique	Cost ($/ decimal digit)	Spatial economy (d.d./in.^{3})
Print on paper	10^{-6}	7×10^{3}
Punched cards	1.2×10^{-3}	5×10^{2}
Punched paper tape	2×10^{-6}	3.6×10^{3}
Photographic film	5×10^{-3}	4×10^{4}
Magnetic tape	1.25×10^{-6}	2.5×10^{4}
Magnetic wire	4×10^{-6}	14×10^{4}

Read-Record Equipment and Media

Technique	Cost ($/ decimal digit)	Time to procure successive words from file (sec)	Read time (sec)	Record time (sec)
Print on paper	--	3	$30x10^{-3}$	$100x10^{-3}$
Punched cards	10^{-2}	0.1	$4x10^{-3}$	$4x10^{-3}$
Punched paper tape	$2.8x10^{-3}$	$20x10^{-3}$	$3x10^{-3}$	$83x10^{-3}$
Photographic film	$1.3x10^{-3}$	$83x10^{-6}$	$83x10^{-6}$	$83x10^{-6}$
Magnetic tape	$2x10^{-3}$	$5x10^{-3}$	$5x10^{-3}$	$5x10^{-3}$
Magnetic wire	$3x10^{-3}$	$13x10^{-3}$	$13x10^{-3}$	$13x10^{-3}$
Magnetic Disks	$.4x15^{-3}$	$.5x10^{-3}$	$.5x10^{-3}$	$.5x10^{-3}$

Fig. 14

[22] See Reference 2

APPENDIX III

COMPARISON OF INTERNAL MEMORY MEDIA[23]

Technique	Cost ($/decimal digit)	Spatial economy (d.d/in^3)	Access time (microsec)	Read or record time (microsec)	Permance	Life expectancy of storage medium (years)
Relays	120	8x10^{-3}	10	3x10	No	1
Vacuum tube flip-flop	100	1.25x10^{-2}	0.5	less than 1	No	0.5
Electric delay line	75	5x10^{-3}	20	less than 1	No	5
Mercury delay line	4	0.05	300	less than 1	No	2-5
Quartz delay line	3.5	0.1	300	less than 1	No	more than 10
Magnesium delay line	3	0.08	300	less than 1	No	more than 10
Magnetic drum	0.30	2	8000	less than 10	Yes	more than 10
Electrostatic storage tube (Williams type)	4	0.1	25	less than 1	No	0.3
Static magnetic shift registers	20	5x10^{-3}	200	less than 10	Yes	more than 10
Core-diode matrix	4	2	5	less than 1	Yes	more than 10
Coincidence-current core-matrix	1	25	10	less than 1	Yes	more than 10
Cerebral cortex (human)	10^{-6}	5x10^{8}	2x10^{6}	30x10^{3}	No	65

Fig. 15

[23] See Reference 2

BIBLIOGRAPHY

1. Andrews, Craig and Quick, H.R. "Magnetic Memory Inventory," Electrical Manufacturing, Vol. 52, No. 4, October 1953, pp. 124-143.

2. Auerbach, I. L. "Fast Acting Digital Memory Systems," Electrical Manufacturing, Vol. 54, Nos. 4-5, 1953.

3. Bagley, P. R. Electronic Digital Machines for High Speed Information Searching, M.S. Thesis, M.I.T., 1949.

4. Casey, R. S. and Perry, J. W. Punched Cards, Their Applications to Science and Industry, Reinheld Publishing Corp., New York.

5. Eckert, W. J. Punched Card Methods in Scientific Computation, Columbia University, 1940.

6. Forrester, J. W. Digital Computers as Information Processing Systems, Report R-166-1, Digital Computer Laboratory, Cambridge, Mass., M.I.T., 1949.

7. Goldenburg, D. Time Analysis of Various Methods of Sorting Data, M.I.T. Digital Computer Laboratory, M-1680, October 17, 1952.

8. Goldstine, H. H. and John von Neumann Planning and Coding for an Electronic Computing Instrument, Princeton, N. J., Institute for Advanced Study, 1947.

9. King, J. W., Brown, G. W., and Ridenour, L. N. "Photographic Techniques for Information Storage," Proceedings of the I.R.E., Vol. 41, No. 10, October 1953, pp. 1421-1428.

10. Lessing, L. P. "Computers in Business," Scientific American, Vol. 190, No. 1, January, 1954, pp. 21-25.

11. Marimount, R. B. and Chin, J. W. Estimates of Sorting Rates for Digital Computer, National Bureau of Standards Report 1875, August 1952.

BIBLIOGRAPHY

12. Morriss, B. E. — *Department Store Information Processing by Digital Computer Techniques*, M.S. Thesis, M.I.T., 1951.

13. Rabinow, Jacob — "The Notched-Disk Memory," *Electrical Engineering*, August, 1952.

14. Roberts, F., and Young, J.Z. — *Proc. Inst. of Elec. Eng.*, Vol. 99, Series III A, No. 20, 1952; *Nature*, Vol. 167, p. 231, 1951 and Vol. 169, p. 963; 1953.

15. Shiowitz, Morc — *Analysis of the Application of Large Scale Digital Computers to a Sorting Process*, U. S. Bureau of Standards Report 1519, 1952.

16. Thomas, W. H. — "Fundamentals of Digital Computer Programming," *Proceedings of the I.R.E.*, Vol. 41, No. 10, October 1953, pp. 1245-1249.

17. Tyler, Arthur W. — "Recording Techniques for Digital Coded Data," *Review of Input and Output Equipment Used in Computing Systems, Joint AIEE-IRE-ACM Computer Conference*, American Institute of Electrical Engineers, March 1953, pp. 3-7.

18. *The Application of Automatic Electronic Digital Techniques to the Operations of the John Hancock Mutual Life Insurance Company*, Raytheon Manufacturing Company, Waltham, Mass., 1950.

19. *The Elecom 125 Electronic Business System*, Underwood Corporation, Long Island City, N. Y.

20. *Sorting and Collecting with the CRC 107 or 102A General Purpose Computers*, Computer Research Corp., Hawthorne, California.

21. *Systems Magazine*, Vol. XVII, No. 11, November, 1953.

MIT