ELECTRONIC DIGITAL MACHINES
FOR HIGH-SPEED INFORMATION SEARCHING

by

Philip Rutherford Bagley

SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

(1951)

Signature of Author .....
Department of Electrical Engineering, August 24, 1951

Certified by .....

.....
Thesis Supervisors

.....
Chairman, Departmental Committee on Graduate Students

Abstract

ELECTRONIC DIGITAL MACHINES

FOR HIGH-SPEED INFORMATION SEARCHING

by

Philip Rutherford Bagley

Submitted for the Degree of Master of Science in the Department of Electrical Engineering at the Massachusetts Institute of Technology

1951

The object of this thesis is to examine methods and machines designed to make possible the high-speed location of related items in a large body of suitably-indexed information.

A flexible and comprehensive indexing plan exists which requires for the selection of a document the simultaneous occurrence of a specified combination of index terms. However, a search of the entire index is required, which for a sizable collection of documents can be satisfactorily performed only by a high-speed machine. Some machines and punched-card systems have been developed which can perform very limited searches at relatively slow speeds. No machines have been contemplated up to the present which have the requisite speed and flexibility to cope with an information searching problem of the magnitude encountered in the United States Library of Congress (approximately 15 million documents including articles in periodicals). It has been conclusively shown here that without major modifications an electronic digital computer designed for general mathematical work is inherently unsuited to the information searching process. It is possible, however, to construct an electronic machine from standard digital computer components (flip-flops, gates, etc.) which would be capable of scanning the index to nearly 5 million documents per hour (at 900 binary digits per document) and of preparing a list of the serial numbers of the selected documents. Metal magnetic tape seems to be the most suitable storage medium for the index. The rate of scanning by the machine is limited by the physical characteristics of the tape alone; it is anticipated that technological improvements will eventually enable the scanning rate to be increased by a factor of 3 or 4.

August 1951

TABLE OF CONTENTS

Page

Title page i

Abstract ii

Table of contents..... iii

Foreword..... vii

PART ONE. INTRODUCTION TO THE INFORMATION SEARCHING PROBLEM

CHAPTER 1. THE NATURE OF THE INFORMATION SEARCHING PROBLEM..... 1

A. Summary

B. What Is Happening to Our Storehouse of Information

C. Why Is This Deactivation So Critical?

D. What Are the Kinds of Information Searching?

E. What Are Some of the Specific Literature Searching Problems at Present?

F. What Are Some of the Problems Encountered by Persons in Libraries?

CHAPTER 2. THE CAUSES OF THE INFORMATION SEARCHING PROBLEM 12

A. Summary

B. The Inadequacy of Conventional Indexing and Classification

C. The Brain as an Indexing System

D. The Multi-Dimensional Nature of Information

E. General Observations on the Construction and Use of an Indexing Scheme

F. New Indexing Schemes Made Possible by Mechanized Searching

G. Indexing a Large Collection of Documents

H. Prospects of Machine Searching

I. Envisioning a Bibliographic Searching Machine in Operation

CHAPTER 3. RECENT APPROACHES TO THE PROBLEM OF INFORMATION SEARCHING 30

A. Present Equipment and Systems

B. Comparisons of the Various Systems

PART TWO. INVESTIGATION OF ELECTRONIC DIGITAL METHODS FOR HIGH-SPEED SCANNING AND SELECTION

CHAPTER 4. CODING INFORMATION FOR MACHINE HANDLING 38

A. Summary and Conclusions

B. The Need for Coding; The Binary Code

C. Requirements for a Coding System

D. Basic Methods of Coding: 1. Superposition

E. Basic Methods of Coding: 2. Generic Array

F. Binary Digit Requirements of the Two Methods of Coding

CHAPTER 5. GENERAL DESCRIPTION OF THE SELECTION PROCESS..... 52

A. What the Selection Process Must Accomplish

B. Form of the Stored Index Information

C. The Meaning of the Flow Diagram

D. General Steps in the Process of Scanning, Matching, Combining, and Identifying

E. Continuous Storage Media (Film, Tape) Versus Punched Cards

CHAPTER 6. THE GENERAL-PURPOSE ELECTRONIC DIGITAL COMPUTER...67

A. The General-Purpose Digital Computer as a Flexible Information-Processing System

B. A Description of the Computer

C. Preparing a Program for Searching

D. Results of Minor Modifications of the Computer

E. Results of Major Design Changes in the Computer

F. Relative Speeds of Operation of the Various Programs

G. Input and Output Media

H. Comment on the General Philosophy and Abilities of the Electronic Digital Computer

Page

CHAPTER 7. A SPECIAL-PURPOSE ELECTRONIC DIGITAL MACHINE... 77

A. Philosophy of Sequential Versus Simultaneous Operations

B. The Machine Operations in Detail

C. Detailed Description of the Searching Machine

D. The Effect of Storage Media and Input Equipment upon Searching Speed

E. Time Estimates of Operation

F. Weaknesses and Capabilities of the Searching Machine

PART THREE. CONCLUSIONS AND RECOMMENDATIONS

CHAPTER 8. EVALUATION OF HIGH-SPEED SEARCHING EQUIPMENT.... 96

A. The Cost of an Electronic Searching System

B. The Cost of Punched Card Systems

C. The Desirability of Electronic Versus Punched Card Systems

D. The Practical and Ideal Features of Searching Systems

E. The Intellectual Consequences of a High-Speed Searching System

F. The Evolutionary Aspect of High-Speed Searching Systems

CHAPTER 9. SUGGESTIONS FOR FURTHER WORK 107

A. Aspects Needing Further Development

B. Description of Equipment with Possible Applications to the Field of Information Searching

C. Organizations Interested in Machine Methods of Information Handling

D. A Note about Translating Machines

APPENDICES

Page

APPENDIX 1. DERIVATION OF SUPERPOSED CODING FORMULAS...114

APPENDIX 2. DESCRIPTION OF MAJOR ELECTRONIC COMPUTER COMPONENTS.....117

A. Flip-Flop

B. Gate Tube

C. Crystal-Diode Matrix

APPENDIX 3. GUIDE TO CODING FOR THE WHIRLWIND I COMPUTER.119

A. Short Guide to Coding; List of Orders

B. Sample Program For Whirlwind I

APPENDIX 4. SEARCHING WITH THE WHIRLWIND I COMPUTER...124

A. The Fundamental Process of Recognizing Equality of Two Numbers

B. "Logical Computation"--The Mathematical Solution of Equations in the Algebra of Logic

C. A complete Program For Searching

D. Special Orders

E. Major Modifications to the Whirlwind I Computer to Increase Adaptability to Information Searching

FOREWORD

The first suggestion to the author to enter the field of information searching came from Mrs. Ruth Lane, head of the Vail Library at the Massachusetts Institute of Technology. In May, 1950, Russell C. Coile completed for Course 6.68 at M.I.T. a report entitled, "A Study In Problems of Documentation in Electrical Engineering." In his report, Coile investigated the present and predicted volume of literature of interest to electronic and electrical engineers. He reviewed some of the existing machines designed for searching for information and proposed a modification of the Microfilm Rapid Selector for which he drew up the basic specification. It was through Mrs. Lane that the author met Russell Coile and became interested in carrying on his work. At a conference with Professor Harold Hazen of the Electrical Engineering Department and Russell Coile it was agreed that carrying on Coile's work was a worthwhile project for a Master's thesis.

The author was then introduced to Professor James W. Perry of the Modern Language Department who had been studying for some time the problems of documentation, principally in the field of chemistry. Finally it was mutually agreed that an investigation was in order for determining the applicability of an electronic digital computer to the information searching problem. Fortunately there existed in one of M.I.T.'s own laboratories one of the newest and fastest general purpose electronic digital computers in the world (Whirlwind I). The

cooperation of the Whirlwind staff in supplying information was secured, and the study of the computer got under way about December, 1950. Before the work had progressed very far, however, it became evident that this application of the general-purpose digital computer was a very awkward and inefficient one. The aim of the study was then modified to that of determining a more suitable method and machine.

It was felt unnecessary, and perhaps impossible, to document every statement in this report. A bibliography has been included at the end of each chapter to facilitate checking and further study.

Some of the ideas contained herein originated in personal discussions with the persons mentioned below, but it is virtually impossible to trace all the ideas back to their original sources. I hope that those who were kind enough to help me will not feel slighted if they are not given specific credit for their own contributions.

The list of persons who have been of valuable assistance in carrying on this thesis work is a lengthy one. At the top of the list must be put the names of those who gave so freely of their time. This thesis has been supervised by Mr. Jay W. Forrester, in charge of the Electronic Computer Division of the M.I.T. Servomechanisms Laboratory (better known as Project Whirlwind), and Professor James W. Perry of the Modern Language Department at M.I.T. Especial thanks is due Professor Perry, upon whose writings I have leaned heavily in the preparation of Chapters 1 and 2. In addition to my advisers,

two others also gave considerable time: Mrs. Ruth Lane, of the M.I.T. Vail Library, and Russell C. Coile, of the Navy Department.

It would not be fair to omit the mention of all those whom I consulted at one time or another, for suggestions or for specific bits of information:

Jordan J. Baruch, M.I.T. Acoustics Laboratory
Robert R. Everett, Project Whirlwind
Jack Gilmore, Project Whirlwind
Freda Harold, Dartmouth College Library
Harold L. Hazen, M.I.T. Electrical Engineering Department
L. B. Hitchcock, United States Government
John Howard, Burroughs Adding Machine Company
David Israel, Project Whirlwind
David Kemper, Project Whirlwind
Calvin N. Mooers, Zator Company
Albert V. Morris, United States Government
Benham Morriss, Project Whirlwind
J. A. O'Brien, Project Whirlwind
Jack Porter, Project Whirlwind
Edward Rogal
Anola Ryan, Project Whirlwind
Frank M. Verzuh, M.I.T. Electrical Engineering Department
Thyllis Williams, M.I.T.

For clerical assistance I am indebted to Miss Alice Perry and to Mrs. Eleanore Galant.

For reviewing the method of presentation of this report, I extend my thanks to Mr. Forrester and Professor Perry, and also to Professor Thomas F. Jones, Jr., of M.I.T.

Philip R. Bagley

Cambridge, Massachusetts
August, 1951

PART ONE

INTRODUCTION TO THE

INFORMATION SEARCHING PROBLEM

CHAPTER 1

THE NATURE OF THE INFORMATION SEARCHING PROBLEM

A. Summary

The location of written material on a specific topic or the location of a specific piece of information has today become a difficult task, if not on occasion an impossible one. In particular, the complexity of conducting searches in libraries has already become a serious problem for scientists and engineers. Our storehouse of information has grown so large that it has exceeded our capacity for making full use of it. Furthermore, this mountain of information continues to grow at an ever-increasing rate. Difficult accessibility of information impedes the planning of experimental research and also hampers those investigations which consist of gathering and correlating data in order to demonstrate new relationships or new points of view. The fundamental problem involved is the inadequacies of our handling information so that we can retrieve what we want when we want it.

B. What Is Happening to Our Storehouse of Information?

Dr. Bush in his now famous article "As We May Think" expressed the nature of the problem very colorfully: "There is a growing mountain of research. The investigator is staggered by the findings and conclusions of thousands of other workers — conclusions which

he cannot find time to grasp, much less to remember as they appear.

"The difficulty seems to be not so much that we publish unduly in view of the extent and variety of present day interests but rather that publication has been extended far beyond our present ability to make real use of the record. The summation of human experience is being expanded at a prodigious rate and the means we use for threading through the consequent maze to the momentarily important item is the same as was used in the days of square-rigged ships.

"A record if it is to be useful to science must be continuously extended, it must be stored, and above all it must be consulted. Today we make the record conventionally by writing and by photography, followed by printing; but we also record in film, on wax discs, and on magnetic wires. Even if utterly new recording procedures do not appear, these present ones are certainly in the process of modification and extension.

"The repetitive processes of thought are not confined, however, to matters of arithmetic and statistics. In fact, every time one combines and records facts in accordance with the established logical processes, the creative aspect of thinking is concerned only with the selection of the data and the process to be employed and the manipulation thereafter is repetitive in nature and hence a fit matter to be relegated to the machines."

Let us now examine some of the statistics which give some idea of the extent of publication today and attempt to forecast the publication trends of the future.

To quote Professor Perry: "Recently published statistics relating to chemical publication show that a search of Chemical Abstracts would have been complete in 1920 after considering twelve volumes containing some 184,000 abstracts. But in 1935 there would have been fifteen more volumes to search, and these new volumes alone contain about 382,000 abstracts. By the end of 1950 the forty-four volumes of Chemical Abstracts to be searched contained well over a million abstracts." If the present trend in publication continues, the total abstracts published in this one field by 1960 will be almost 1,800,000.

"Not only the absolute number of papers and patents published but also the rate of publication is increasing. The Chemical Abstracts statistics show that the increment in rate was 1,500 publications per year during the period 1920-1939. Since the war the increment in rate has jumped to 2,400 papers per year."

Let us look at the figures in another representative field — Electrical Engineering. In a study made by Russell Coile, he estimated that the total number of published papers of current interest to electrical engineers was between 250,000 and 500,000. He further estimates that the current rate of publication of these papers is 10,000 to 20,000 per year. Hence by 1960 the total will lie between 350,000 and 700,000 papers of interest.

What does all this mean? It means that there is a progressive deactivation (to use another phrase of Professor Perry) of our

storehouse of knowledge produced by current research and publication. Furthermore, it means that the situation is rapidly getting worse. Research continues to generate a vast amount of new knowledge, in spite of the handicaps caused by the size of our storehouse of knowledge.

C. Why Is This Deactivation So Critical?

Research is big business. Let us look at some more statistics. In the table below are given the research expenditures by American industries for the year 1947 (from a survey made by the National Association of Manufacturers).

Industry	Expenditures
Chemical and allied products	$27,624,000
Products of petroleum and coal	47,232,400
Rubber products	10,360,065
Electrical machinery, supplies, equipment	9,261,500
Professional, scientific and controlling instruments	15,589,258
TOTAL	$110,067,223

The adequate planning of research requires a thorough knowledge of previous activities. No right-thinking director of research willingly embarks on a program without first knowing the "state of the art." Acquiring this background for a research program is sometimes referred to as "literature searching." It is the opinion of the author that literature searching is fast becoming a controlling factor in the amount and direction of original research. The tremendous amount of published information, coupled as we shall see later with

inadequate methods of locating desired information, has rendered the task of literature searching a complicated, tedious, and expensive process. If the data has already been collected and compiled by some writer, the searcher may be fortunate enough, after a short examination of a card catalog, a periodical index, or an abstract journal, to locate the desired data. If, however, the searcher must investigate every compound in the group for its applicability to the process, an almost endless search through chemical books and papers may be required. While this example illustrates in a particularly striking fashion how formidable a search may become, any librarian can enumerate many searching problems of a similar nature.

D. What Are the Kinds of Information Searching?

There are essentially two types of activity which come under the heading of information searching. They are: Bibliographic Searching, and Correlation. Bibliographic searching is the investigation of a body of stored information in order to locate documents relating to one or more topics. A bibliographic searching system may provide, for instance, a list of all the available material on the subject of "radar used in weather forecasting." Correlation is the process of retrieval of related items or ideas from a body of stored information. A correlation system might answer such a question as: What diseases have the symptoms A, B, C, and D in common? Bibliographic searching is simply a finding operation, while correlation is the organizing of a mass of facts in an attempt to perceive new relationships.

Both types of searching have certain features in common. The stored information must be "tagged" or indexed in a suitable fashion. Each block of index entries pertaining to a single article or document consists of entries "characterizing" the subjectmatter of the document, and some designation identifying the document. Example: An article dealing with the production of wheat in Kansas for the year 1949 would be characterized at least by the words Wheat, Kansas, and 1949, followed by the serial number of the document. The process of selection is one of inspecting each block of stored index information for some specified index word or for some combination thereof.

E. What Are Some of the Specific Literature Searching Problems Today?

One profession which faces an acute problem is chemistry. There are over a million published abstracts of chemical papers. When a chemist wishes to conduct an investigation of some chemical process, he either has to have ready access to the previous work along similar lines, or he may have to repeat the work himself. Here time is money — big money. Chemical Abstracts and the great German compendia have done a great deal to alleviate the chemist's information problem. These time-honored roadways to needed information are experiencing a severe traffic jam caused by the volume of research publication. The situation in chemistry is sufficiently critical that the American Chemical Society has had a committee studying for over four years the possibilities of bettering the methods of information storage and retrieval. (Here follows a quotation from "Chemists and Librarians" by Professor James W. Perry.)

"From the chemist's point of view the starting point for re-evaluation of the chemical information problem is the fact that the day is long since past when any one chemist could hope to read all the available chemical information of possible eventual interest to him in his work. It would be even more impossible to retain in his memory the enormous mass of useful detail involved. Human limitations have forced chemists to transfer to an ever greater degree the important function of memory to bound volumes stored on the shelves of libraries and to collections of more or less private reports.

"Not too long ago when the number of bound annual volumes of the abstract periodicals was relatively small, the amount of time required to consult the record of chemistry in this fashion was considerably less than it is at present. Looking into the not too distant future, it is possible to anticipate that the labor involved in consulting indexes and abstract journals will become excessive."

The problem of the electrical engineer is similar to that of the chemist. While there is a smaller number of papers of direct interest to him, perhaps, he has no single source of carefully indexed abstracts. To complicate matters further, electrical engineering is beginning to have applications in many heretofore unrelated fields, such as medicine, bio-chemistry, geology, and meteorology.

The search problems encountered by the United States Patent Office are becoming increasingly acute. Each patent application requires the investigation of all the other patents and publications

relating to the alleged inventions. The relationships involved have become so complex that the present classification scheme copes with them less and less successfully. The labor required for searching a patent application tends to increase with ever severer challenges to the skill of the searcher if his review of the prior art is to be complete and accurate. A similar situation has been developing for years in other fields of learning — particularly law and medicine.

F. What Are Some of the Problems Encountered by Persons in Libraries?

Most questions asked in research libraries can be classed as one of three types. First we have questions requiring a specific answer. For example, one might ask for the principal commonly used commercial power voltage and frequency available in Japan. A single specific answer will usually suffice.

The second type of question asks for all the information concerning a specific topic. For example, one might ask for all the published material about the "theoretical limit on the rate of transmission of information via an electric channel." In response to this question, the searcher expects to get a list of all the publications which bear on this specific topic.

The third type of question — perhaps the most frequent — is "something about." The searcher is interested in, for example, "something about" the production and construction of printed circuits. In this case, he is not interested in finding all the information on the subject, nor is he interested in finding a detailed coverage of

the subject. He is interested only in one or two papers which treat the subject generally.

These types of questions may be further illustrated by questions directed to the M.I.T. Vail Library during the past year. Some persons have desired to make a comprehensive search of unpublished information, that is, finding reprints of conferences and projects. Industrial requests have been received asking for lists of theses covering a broad range of subjects for various specified periods. Some requests, directed to very specific subjects, ask for thorough information if possible. Another is the requests of the students and staff for material to bring themselves up to date on various subjects.

Technical literature appears in many forms other than books and magazine articles, as is indicated by the following list:

technical papers
mathematical tables
handbooks
specialized handbooks
technical reports
Government bulletins
seminars
theses
test codes
statistics
bibliography
standards
catalogues
manufacturers data sheets.

It is often desired to have other information than subject headings in an index or card catalog. In the opinion of Mrs. Lane, the bibliographic information which should be indexed in some manner

in addition to subject material should be the following: language, date, and place of publication; publication source (bureau, committee, company); extent of bibliography; number of illustrations, diagrams, and tables; paging; exact title.

Another problem is posed by outdated material. This is a difficult problem for that material which is apparently obsolescent today may not be so tomorrow. Even if it is obsolete, it may be of some definite historical value. There is little doubt that some percentage of the material is definitely of sub-marginal value. Here the problem is to determine what is obsolete and to remove from the file in a practical manner all of the indexing information on an obsolescent item. Another problem is that of locating the material of the most value. This again is a problem of judgment for which there is no easy universal criteria. It would be advantageous if the relative value or relative usefulness of a document could be included along with the other index entries, but the determination of relative value presents problems for which there is no practical solution at present.

BIBLIOGRAPHY

Bush, Vannevar. "As We May Think." Atlantic Monthly, vol. 176, July 1945, pp 101-108.

Ball, Norman T. "Very Basic Research." Cambridge, Massachusetts Institute of Technology, May 1948, 6 pages mimeographed.

Casey, R.S. and Perry, J. W. Punched Cards, Their Application to Science and Industry. New York: Reinhold, 1951, Chapter 25.

Coile, Russell. "A Study in Problems of Documentation in Electrical Engineering." Cambridge, Massachusetts Institute of Technology, May 1950. 122 pages, typewritten.

Perry, James W. "Chemists and Librarians." College and Research Libraries, January 1949, pp 12-16.

Perry, James W. "Storing and Processing Information." Cambridge, Massachusetts Institute of Technology, March 1951. 12 pages, lithographed.

CHAPTER 2

THE CAUSES OF THE INFORMATION SEARCHING PROBLEM

A. Summary

In order to provide a means for retrieving material related to a certain topic, we have in the past ordinarily made use of subject indexes and classification schemes. Deficiencies in these conventional methods have become increasingly apparent in recent years. The prospect of using mechanized methods of searching opens new possibilities for "tagging information for retrieval." A tagging scheme can be devised based on the multi-dimensional nature of information. A concept is located at the intersection of these "dimensions," (akin to specifying a point in multi-dimensional space by giving its coordinates).

A machine selection of a document takes place by locating a certain "value" in each dimension (in the index), such as the time and place of an event. Hence the index earns the name "coordinate" index. If higher-order generic terms (often called "higher levels of abstraction") are included in the indexing, a document indexed under "wheat" will be selected if "grain" or "agricultural plant" is searched for. A flexible and comprehensive coordinate indexing system can be constructed from "abstraction ladders" (series of related terms of increasing order of generality). It is with this

coordinate-array indexing that the thesis shall be concerned in later chapters.

B. The Inadequacy of Conventional Indexing and Classification

Classification is a means for organizing information into a hierarchy of classes and sub-classes — usually mutually exclusive — so that related items of information will fall into the same sub-class, and hence may be located with greater ease. The common characteristics for each sub-class are often neither independent nor unique, however. We may divide the plant kingdom, as an example, in many ways: according to the color of the plants, or to the physical similarities, or to the uses which may be made of the plant, or the localities in which it grows. In any single conventional classification, because of its physical construction, a class may be subdivided in only one way. This requires one or more decisions which usually are made on the basis of what will be most useful to the users of the classification. It is evident, then, that any single scheme of classification will not be of equal adequacy for users with differing points of view.

A simple example will show the practical difficulties of a classification scheme. Let us assume there is a file of patents relating to dyestuffs, classified according to the chemical structure. It would be fairly simple to locate patents relating to a certain specific type of diazo dye. It would not be a simple task, however, to locate those patents concerned with all the dyes stated to be

applicable to celanese (a type of artificial silk). In such a search, classification on the basis of chemical structure is of very little help as it is necessary to examine all the patents in the file. (Taken from Perry, "Analysis of Information for Searching and Correlation by Machine Methods.")

Conventional classification is compelled to operate on a faulty basis. A conventional classification scheme is forced to arrange knowledge in one dimension (that is, in a single line) in such a manner that all related material will fall adjacent. This is simply impossible. If one looks at the Universal Decimal Classification he cannot decide where to place Biochemistry, Geophysics, diathermy machines, and information theory.

(From Perry, "A New Era in Scientific Information Methods.")

"... it is important to keep in mind that it is not the intellectual basis of classification — namely the grouping of items in terms of desired criteria — which imposes limitations. These arise rather from the practical impossibility of using a simple mechanical device — namely, a set of pigeon holes — to provide as many groupings as may be desired and eventually useful as new developments and new needs raise unforeseen questions."

Another method for the organization of information is the index. An index is an ordered list of terms (generally alphabetical) or combination of terms, containing opposite each entry a designation

(generally a page number) which uniquely specifies the location of the item to which the index entry refers. If the index is constituted of single terms only, the index will be only as long as the list of terms. If, on the other hand, we are completely indexing chemical compounds, for example, not only by name, but by color, family, and chief use, in all combinations, the number of entries in the index becomes the factorial of the number of attributes or terms. For each of 100 compounds, indexed by name, color, family, and use, there will be 4! (four factorial) = 24 entries, or a total of 2400 entries altogether. Practically, this is never done because of the sheer bulk of the index. Instead, the most meaningful entries — in the judgment of the person who makes up the index — are used, the rest discarded. It is again evident that conventional indexing is inadequate for users with different points of view.

C. The Brain as an Indexing System

An ideal tagging system should provide associative links between concepts or items of information in a manner similar to the human brain. When one thinks of kitchen, his chain of thoughts may progress in one of the following manners:

kitchen, food, partaking of daily meals, one's nutrition, state of health, what one's life expectancy will be, a concern for the adequacy of one's life insurance policy, etc.

kitchen, mother, women in the family, women in general, etc.

kitchen, kitchen utensils, hardware business, etc.

Each concept is related to one or more others by "association."

At present the brain's system of association is not sufficiently well understood that it can be duplicated by artificial means. At best, a system can be constructed, in which each associative link is specifically noted down. The mere bulk and tedium of this process limits what can be accomplished to a small fraction of what might be desired for exhaustive completeness. One artificial system, which seems to be a major step in the direction of achieving a practical scheme is the "generalized classification scheme," or "multi-dimensional classification," or system of "coordinate indexing." This scheme will be dealt with at greater length in succeeding sections.

It should be noted that the seemingly indiscriminate use of the words classification and indexing interchangeably is not careless confusion of terminology. The aspect of multi-dimensionality tends to reduce both classification and indexing to a common denominator. The machine indexing scheme will be seen to embody the characteristics of both of the more conventional methods.

D. The Multi-Dimensional Nature of Information

(From "Analysis of Information for Searching and Correlation by Machine Methods," by James W. Perry)

"Documentary information — the record of human experience — is essentially polydimensional in character. An historical event, for example, may be analyzed in terms of such dimensions as time, place, persons and organizations involved, and nature of action occurring. The record of a surgical operation is concerned with such different

types of variables as the patient's symptoms, the pathological conditions thereby deduced, treatment, and the result of treatment. A chemist describes his experiments in terms of the interacting materials, the reaction conditions, the substances produced and accompanying effects, such as the evolution of heat.

"...Since there is no a priori relationship between time, persons, and action, the analogy to orthogonal coordinates is so suggestive as to be useful in further discussion ....We will thus have the following general classes of dimensions: time, place, persons, things, actions. These five types are inherently orthogonal in nature, that is to say, it is not possible to deduce from one of these — in the absence of recorded information — anything concerning any one of the others.

"Our analogy between the analysis of information and Cartesian coordinates must not be pushed too far. This becomes evident when we consider the problem of designating material objects — all of which we can think of as pertaining to a single coordinate — the "thing" dimension. In conducting the analysis of information it will not prove convenient to establish a single array of terminology along the "thing" dimension in the way that numerical values are ordinarily used to measure a distance along a coordinate axis. Rather, we will often wish to designate things from different points of view, such as purpose (means of transportation, power source, weapon), mode of functioning (electrical, internal combustion, atomic fission), materials of construction (steel, copper, plastics), etc.

These different points of view constitute in their own right something akin to a set of sub-dimensions, which, however, are not necessarily orthogonal in nature. An electric motor will scarcely be used to shoot bullets and a machine gun is not a recognized power source. Hence — from a purely theoretical point of view — there may be no absolute need for specifying a thing in terms of several non-orthogonal sub-dimensions. It is, however, extremely useful — even downright necessary from a practical point of view. Otherwise, conducting searches becomes exasperatingly difficult."

E. General Observations on the Construction and Use of an Indexing Scheme

In order for an indexing scheme to be of practical use, it must have certain features. No matter how capable the scheme theoretically, it is valueless unless it is at the same time practical.

1) The analysis of the information must be done in such a fashion that the searcher's question is either "in harmony" with the indexing plan or is capable of being brought into harmony with it. Furthermore, the phrasing of the searcher's question may be quite vague; the system must not require that he be able to specify his question with hairline detail.

2) The system should permit the accomplishment of searches not envisaged when the system was constructed. It is difficult to give a good illustrative example until the concept of "logical combinations" is discussed. For the present, let it be pointed out that if documents were classified by states (among other things) it would

be possible to select documents relating to the "industrial New England States" by looking for those dealing with either Massachusetts, Rhode Island or Connecticut — a "logical sum" of terms.

3) Semantic confusion should be kept to a minimum by the precise definition and use of terminology. A dictionary of allowable terms may be needed so that non-existent terms will not be used for conducting a search.

4) The system must not be excessively complex, or it will defeat the purpose it is supposed to serve — that of providing a convenient means of locating information.

F. New Indexing Schemes Made Possible by Mechanized Searching

The ability of mechanical or electrical devices to perform routine operations at high speed makes possible indexing schemes based on the requirement that the entire index be searched for each question. Machine searching makes unnecessary the arrangement of the index in any particular order, hence the preparation of the index is simplified to the extent that one coherent block of index entries is sufficient for each document — there is no need for duplicative index terms or entries in other parts of the file, such as is now done to provide cross-indexing.

Before entering upon this discussion, it would be well to define a few terms:

index term = a single word or phrase similar to a subject heading, used to designate a single concept; for example, in referring to a document concerned with the production of wheat, one index term might be "wheat."

index entry = a component unit of an index; may be either an index term, e.g. "wheat," or a group of index terms considered as an integral unit; for example, "wheat, production of, in Kansas."

index block = all the index entries pertaining to a single document; for example: "wheat, production of, in Kansas; Agricultural Experimenter Journal; 1949, July."

document = any publication considered as a single unit for indexing purposes; for example, a book or a chapter therefrom; an article from a periodical; a separately published paper.

logical combination = the grouping of units (in this context, index entries) which satisfy an equation written in the algebra of logic (sometimes referred to as Boolean algebra).

generic array = an ordered sequence of index terms, arranged in increasing order of specificity (decreasing order of generality); for example, a generic array of terms stemming from a specific term such as "biology, body, tissue, cell, nucleus." In the language of semantics, a generic array of terms is called an "abstraction ladder," because progressing from a less general to a more general term is the process of "abstracting."

Various approaches are possible in developing plans for indexing, based on a machine's ability to search an entire index.

1.) A plan using specific index terms, with relationships between terms not indicated.

2.) A plan using specific and generic terms, with relationships between terms not indicated.

3) A plan similar to either 1) or 2), with relationships between terms expressed by a rudimentary grammar.

4) A plan using specific and generic terms, arranged in a single generic array stemming from each specific term, with relationships between arrays not indicated.

5) A plan using specific and generic terms, arranged in a plurality of generic arrays stemming from each specific term, with relationships between arrays not indicated.

We shall proceed to describe each of the plans in detail.

As an example of the first type, we may return to our previous example of the indexing of 100 chemical compounds. The 100 compounds need be listed only once (making a total of 100 index entries), with the associated attributes (family, color, chief uses) beside each compound. The order of the attributes within a given entry is of no consequence. The machine will search each entry for the simultaneous presence of, say, "green" and "detergent." We thus locate all the compounds in the index which are green detergents. The machine is not able, however, to search directly for the broader expression "surface active agents." We may overcome this drawback if we are willing to go to the slight additional effort of searching for "green," and any one of several terms "emulsifiers, detergents, or wetting agents." This approach resembles an ordinary index but with the distinction of having no entries repeated with its terms permuted (permutation of terms being the way in which

"cross-index entries" are constructed — in this plan no cross-indexing is necessary).

Another approach is similar to the first, but in addition are included a few generic terms for each specific term. The number of generic terms (the number of steps in the abstraction ladder) is based on convenience. For example, in our index of compounds previously mentioned for illustrative purposes would be included such terms as "surface active agents." We now have the advantage that in looking for surface active agents we will locate the index entries dealing specifically with detergents, wetting agents, etc. This plan is utilized by some of the punched-card indexing systems in current use.

A third approach involves the use of index entries such as are used in both of the approaches above, with the additional feature that the relationships between terms or arrays of terms are expressed by a rudimentary grammar. Some of the relationships which may be expressed are: synonyms, antonyms, inclusions, constituent, experimental relation. One of the simplest ways to construct the grammatical framework is to add artificial prefixes or suffixes to the various index terms. The index would then contain expressions such as: Substance X (acts on) Insect Y; Substance X (has constituent) Color Green. This type of indexing plan has been advocated by J. E. L. Farradane (see Bibliography).

A fourth approach involves single generic arrays stemming from each index term. The Universal Decimal Classification is an

example. A document classified under the U.D.C. might have the following associated numbers: 621.385.3 : 621.315.592 ; 518.4. Each number (such as 621.385.3) represents a generic array of terms. The document is more or less uniquely specified by the combination of these various arrays of terms.

A fifth approach has been called variously a generalized or multi-dimensional classification, or a "coordinate-array index." In this plan, not one but a plurality of generic arrays may be built upon each specific term. (Examples will be given presently.) The relationships between the arrays are not expressed in the coding (although with additional effort they might be, if it were found desirable). Each array built from the same index term would be in a different ladder of abstraction: such as time, place, person, utility, thing, action, mode of functioning, material.

In order to make effective use of this fifth approach, the machine must be able to search for a "logical combination" for index entries pertaining to a single document: For example, to be able to search for entry A plus either entry B or entry C. There follows a demonstration of the reason for the use of a plurality of arrays stemming from a single index term, and for the desirability of finding "logical combinations."

A document might be classified under the U.D.C. scheme of single generic arrays. Take, as an example, a document concerned with the Effect of Rainfall on the Production of Durum Wheat in Kansas for 1949. This document will never be turned up if a search

is directed to "climatic effects on commercial foodstuffs." It is not likely that it can be located if the search is directed to "foodstuffs grown in the Midwest."

Under the coordinate array indexing plan, there might be assigned such arrays as these:

meteorology, climate, precipitation, rain

plant life, grain, wheat, durum

North America, United States, Mid-west, Kansas

foodstuffs, bread, wheat, durum

commerce, agriculture, grain production, wheat, durum

20th century, 5th decade, 10th year.

Now this document will be located when searches involving generic terms are conducted. For example: "Midwest grain production 1940-1949." With the indexing broken apart into its elements, these elements may be recombined in many ways. This form of indexing is essentially a n-dimensional classification scheme wherein a particular subject lies at the intersection of n dimensions. The five dimensions above are substance, location, time, realm of activity, and treatment (processing). There are obviously many more dimensions. Dimensions can be set up arbitrarily if desired; for instance, a subdivision of the "thing" dimension might be "ordnance firing mechanisms."

The terms upon which a selection is based are stated in what is called a "logical combination." For example, the statement "A document will be of interest if it is indexed under the terms A and (B or C) and (D but not E)" is a statement in the language of "logic."

In the algebra of logic the statement would be written

$$A \cdot (B + C) \cdot D \cdot \overline{E} = \text{Selection}$$

There are a great many ways in which such statements may be constructed in order to specify a topic.

Thus it is seen that the coordinate array indexing plan possesses flexibility and a high degree of what might be called cross-indexing. It is upon this coordinate array indexing scheme that the majority of the work in this thesis is based.

G. Indexing a Large Collection of Documents

Let us assume we wish to plan a suitable scheme for handling a very large collection of documents, say 50 million. This is not an unreasonable figure, since the Library of Congress has at present about 15 million documents (including periodical articles individually). At the present rate of publication, the total number of books, articles, and pamphlets which might be indexed — could a library handle the job — will approach 50 million within a few decades.

1) Indexing scheme. It has been suggested by persons familiar with problems of indexing that the most desirable indexing scheme is a multi-dimensional classification constructed of arrays of terms (previously referred to as the coordinate array index).

2) Number of entries per document. Chemical Abstracts at the present time uses about 10 index entries per document indexed, and apparently is limited to this number by the sheer bulk and unwieldiness of the index thus created. With a comprehensive machine

indexing scheme in which bulk and unwieldiness are not limiting factors (such as might be the case if the index were in the form of binary digits on magnetic tape), the useful index entries relating to subjectmatter might be increased to 25. Add an additional 5 for publication data, authors, dates, countries of publication, journals, etc., to give an approximate total of 30 index entries per document.

3) Size of index vocabulary. The estimated total number of entries in the Universal Decimal Classification in 1946 was 860,000. This seems far too small a number to be used as a basis for a comprehensive indexing plan, as the plant and animal kingdoms alone require better than 1 million names; the number of known chemical compounds borders on 2 million; the number of different authors' names runs well into the millions. A total assumed size of indexing vocabulary of 10 million ought to be adequate for the present; a vocabulary of 50 million more than adequate for the foreseeable future.

It remains for Chapter 5 to show how the indexing may be translated into a form which a machine can handle.

H. Prospects of Machine Searching

The tremendous speed and accuracy of modern machinery compared to human capabilities suggest the use of machinery to search for information. The greater speed of a machine makes possible more extensive searching than could ever be accomplished by human effort. It is not inconceivable that a machine could search in a reasonable time through

all the information to which it had access — a task manifestly impossible for even a group of persons. A possible consequence of a machine searching system is that material might be indexed under every conceivably useful heading. Assuming for the moment that an appropriate indexing procedure makes it possible to use the machine to scan and select non-numerical information, it is apparent that a machine might relieve some of the burden of extensive searching, and there is reasonable prospect that a machine searching system can be made to provide ready access to nearly all published information.

Searching through a file of index cards (or other media bearing index entries) reduces to a repetitive routine of inspection, comparison with a standard, and acceptance or rejection based on this comparison. If the stored material is suitably marked or coded, it is possible to relegate the mechanical process of examining the index entries to a machine, which can perform the job much faster than humans.

I. Envisioning a Bibliographic Searching Machine in Operation

Let us speculate for a moment on how a library of the future might be able to provide us with the answers to questions. Having decided on the subject of our search, we convey our wishes to a specially trained librarian. The librarian uses a code book to prepare a punched card or the like to control the operations of the machine. The instructions are given to the searching machine by the insertion of the punched card. The machine performs the requested search and types on an electric printer the document numbers located by the search. With

the numbers thus obtained, the entire documents or abstracts thereof can be selected, either by automatic means, or by looking them up in a file arranged simply by document number. The convenience of having the machine produce, in a single operation, the entire original document is a service which is not likely to be feasible until some radically new methods of storage and printing are developed, whose practicality will be determined by cost considerations.

BIBLIOGRAPHY

Bernier, Charles L. "Relations Among Words." Cambridge, Massachusetts Institute of Technology, 1948. 8 pages mimeographed.

Casey, R.S. and Perry, J.W. Punched Cards, Their Application to Science and Industry. New York: Reinhold, 1951.

Cochran, S. W. and Perry, J. W. "Conventional and Mechanized Search Methods." Industrial and Engineering Chemistry vol. 42, August 1950, pp 1456 f.

Fairthorne, Robert A. "The Mathematics of Classification." Proc. Brit. Soc. for Intl. Bibliog., vol. 9, pp 35-42.

Farradane, J. E. L. "A Scientific Theory of Classification and Indexing and Its Practical Applications." Journal of Documentation, vol. 6, June 1950, pp 83-99.

Hayakawa, S. I. Language in Action. New York: Harbrace, 1941.

Holmstrom, J. E. "A Classification of Classifications." Royal Society Scientific Information Conference Proceedings, 1948. Paper No. 33.

Larkey, S. V. et al. "Categorization as a Basis for Machine Coding." American Chemical Society, Washington. Paper to be presented at the fall meeting, 1951 of the Division of Chemical Literature.

Mooers, Calvin N. "Theory of Digital Handling of Non-Numerical Information and Its Implications to Machine Economics." Boston: Zator Company, 1950, 17 pages (Zator Technical Bulletin 48).

Perry, James W. "Analysis of Information for Searching and Correlation by Machine Methods." Cambridge, Massachusetts Institute of Technology, 1950. 18 pages, mimeographed.

Perry, James W. "A New Era in Scientific Information Methods." Cambridge, Massachusetts Institute of Technology, 1950, 13 pages, lithographed.

Taube, Mortimer, and Thompson, A.F. "The Coordinate Indexing of Scientific Fields." Washington: American Chemical Society. Paper to be presented at the fall meeting, 1951, of the Division of Chemical Literature.

CHAPTER 3

RECENT APPROACHES TO THE PROBLEM OF INFORMATION SEARCHING

A. Present Equipment and Systems

1) Bratner Notched Cards. Bratner of E. I. DuPont has designed a system which uses patented steel cards with 100 slot positions across the base of the card. Each index word is represented by n slots punched in a random pattern from the 100 possibilities (using a superposed coding system). Details as to the maximum useful vocabulary size and the maximum useful number of cards in a single file were not available at the time of writing. Each card is approximately 5 x 8 inches, and is able to hold up to 1100 pages of microphotographically printed material. The selection of the desired cards is performed by magnetic and mechanical means, which sorts 100,000 cards in 100 seconds (60,000 cards per minute). This system is still in the course of development.

2) Calvin Mooers (of the Zator Company, Boston) has developed a system using cardboard cards (known commercially as Zatocards), 5 x 8 inches in size, with 40 punching positions along the bottom edge of the card. Each index word is represented by a pattern of four punches out of the possible 40, and up to 7 words may be superposed upon one card. The total vocabulary is 91,000 words, and the maximum convenient file size is estimated to be

100,000 cards. A second type of card is available with 36 punching positions on each of the two edges top and bottom. Selection is performed by means of a "shake-box" holding sorting needles (similar to ordinary knitting needles) upon which rest the cards to be sorted The whole box is agitated by a small motor, and appropriately notched cards drop down about 1/4 inch where they may be readily seen. Approximately 800 cards per minute may be sorted in the manner. The entire system is available on a fee basis: installation $500, $45 monthly.

3) "Keysort" punched cards. The McBee Company of New York manufactures patented cards (5 x 8 inches) which have a double row of holes along the top and bottom and a single row of holes along each side. Each hole or group of holes may be assigned a specific meaning. Holes are notched, that is clipped to extend all the way to the edge of the card, in order to provide for the selection of cards by a group of (knitting) needles. Many different coding schemes have been proposed for using these cards and are described in a recent book (Casey and Perry, Punched Cards, Their Application to Science and Industry, pp 3-6). Carl Wise reports that a vocabulary of 456,976 words is possible, of which any 16 words can be superposed on a single card. The cost is $18.00 per thousand cards.

4) Batten System. Batten of England has developed a punched card system in which a large card (or alternately, an IBM card) is used for some one predetermined aspect or subject heading. Each document is assigned to a particular numbered hole position

(the same numbered hole position on every card). If a document possesses a particular aspect, the corresponding aspect card is punched in the numbered hole position assigned to the document. Selection is performed by manually picking out the cards bearing the desired aspects, aligning them, and searching for any hole positions which are punched on all the aligned cards. Such holes correspond to numbers of documents having all the aspects represented by the cards used for selection. About 10,000 holes can be accomodated on each aspect card, thus one complete set of aspect cards is required for every 10,000 documents. This system works well for modest-sized collections of documents in relatively narrow fields. No costs are known.

5) IBM Punched Cards and Associated Equipment. Punched cards produced by IBM (International Business Machines Corporation) are 3 1/4 x 7 3/8 inches, containing 80 columns and 12 rows of punching positions. The punching plan enables each column on the card to contain one alphabetic or numeric character. Cards are sorted at 450 to 600 cards per minute on automatic card-sorting machines. It must be known in which columns desired characters will appear, because the sorting selectors can be set on the present equipment only to specific columns. It is believed, however, that new machines are under development which will obviate this disadvantage and which will detect a specified sequence of characters, no matter in which group of columns they may fall.

It would be possible theoretically to utilize IBM cards in ways other than for which equipment has been made available by IBM. The total number of different ways to punch a single column on the card is 2^{12} or 4096. (The total number of different ways to punch an entire card is 2^{960}.) If we adhere to punching one datum per column, selection could take place by inspecting the card column after column (running the card lengthwise through the selecting machine instead of crosswise as is presently done), overcoming the necessity for punching certain information in specified regions or "fields" on the card.

Card punching machines are rented from IBM for approximately $35 a month, card-sorters for approximately $60.

6) Samain Selector. Jacques Samain of France has invented a selecting device which utilizes a special punched card containing provision for 144 characters. His machine performs selection at 400 cards per minute, searching for one six-letter word at a time. To obtain the effect of searching for more than one word at a time, he successively reruns the selected group of cards. Details on his machine have not been published.

7) Microfilm Rapid Selector. Engineering Research Associates have designed and built a machine to specifications drawn up by Dr. Ralph Shaw, Librarian of the Department of Agriculture. This machine stores index information and an abstract of each document on 35 mm motion-picture film. The film is scanned by photoelectric means

at 300 feet per minute — less than 10,800 documents per minute. Each document may have up to six index words consisting of seven decimal digits each. The Rapid Selector permits selection to be based on a single index entry. Inability to search for combinations of entries severely limits this machine's capabilities. A photographic reproduction of the abstract is made at the time of selection. The original cost of the machine was $75,000. Processed 35 mm film costs only a few cents per foot. The predecessor of this machine was the Bush Rapid Selector, which was similar except that only one index word per document was used.

8) Card-Selecting Machine, U.S. Patent 2,539,998 (Assigned to International Business Machines). This is a card-selecting device which is specifically designed to inspect IBM cards column by column for a single specified sequence of characters, plus one additional character whose location is not specified. The machine is intended principally for use in inspecting cards containing the Dyson cipher for chemical compounds. The patent covers construction with either relays or vacuum tubes.

B. Comparisons of the Various Systems.

It is perhaps enlightening to compare the above searching schemes in terms of the number of "bits" of information which the selecting device can process in one second. A bit of information is the amount of information contained in one binary digit; a binary digit is either a 0 or a 1, hence the amount of information it may represent is the choice between two alternatives. In computing

the rate of processing of a system, we determine the number of possible bits of information which the selector can examine in a unit time. If, for example, the index consisted of a single alphabetical symbol for each document, and the selecting mechanism could examine these at the rate of 10 a second, our computations would look like this.

Each letter represents a choice of 1 out of 26 possibilities. Each letter contains n bits where $2^n = 26$, and hence n = 4.71. In one second we examine 10 letters or 47.1 bits of information. The digit-processing rate is then 47.1 bits per second. The following table has been made up by calculating the number of binary digits required to express the same number of possibilities as a single index entry in any given searching system. This is multiplied by the rate of processing the index entries to give rate of processing binary digits or binary bits of information.

Batten cards	(indeterminate)	equivalent bits/sec
Mooers Zatocards (40 punching positions)	530	
Samain selector	3,990	
IBM punched cards as presently utilized	9,600	
ERA Microfilm Rapid Selector	25,200	
Bush Rapid Selector	60,000	
Coile's proposed selector (modification of ERA machine)	71,800	
Bratner cards	100,000	
Electronic digital machine proposed in Chapter 7 of this thesis	1,400,000	

It will be stated without proof at this point that none of the machines above except the electronic digital machine have either the speed or the capability of handling the indexing for a large collection of documents. In discussions to be presented later, it will be seen that the requirements on selection and on the number of binary digits will automatically disqualify the existing machines from competition for devices to handle a large collection of documents.

BIBLIOGRAPHY

Casey, R.S. and Perry, J. W. Punched Cards, Their Application to Science and Industry. New York, Reinhold, 1951. Chapters 1, 4, 6.

Bailey, M. F. et al. "Mechanized Searching in the United States Patent Office." American Chemical Society, Washington. Paper to be presented at the fall meeting 1951 of the Division of Chemical Literature.

de Gorter, B. "The Indexing of British Patents on Plastics: A Punched-Card Method of Indexing Technical Documents in a Limited Field: The Batten System." Royal Society Scientific Information Conference Proceedings, 1948. Paper SIC/192 (48).

Coile, Russell C. "A Study in Problems of Documentation in Electrical Engineering." Cambridge, Massachusetts Institute of Technology, May 1950, 122 pages, typewritten.

Engineering Research Associates. "Report for the Microfilm Rapid Selector." St. Paul, Minnesota and Arlington, Va., 1949. 30 pages lithographed.

Ferris, L., Taylor, K., and Perry, J.W. "Bibliography on the Uses of Punched Cards." Journal of Documentation, vol. 4, 1948, pp 250-271.

Mooers, Calvin N. "Theory of Digital Handling of Non-Numerical Information and Its Implications to Machine Economics." Boston: Zator Company, 1950, 16 pages (Zator Technical Bulletin 48).

Mooers, Calvin N. "Zatocoding Applied to the Mechanical Organization of Knowledge." American Documentation, vol. 2, no. 1, Jan., 1951, pp 20-32.

Perry, J. W. "A New Era in Scientific Information Methods." Cambridge, Massachusetts Institute of Technology, 1950. 13 pages, lithographed.

Samain, Jacques. "New Apparatus for Classification and Selection of Documents and References by Perforated Cards." Royal Society Scientific Information Conference Proceedings, 1948. Paper SIC/191(48).

United States Patent Office. "Card Selecting Apparatus, Patent 2,539,998." January 30, 1951. Washington: Government Printing Office, 1951, 17 pages.

PART TWO

INVESTIGATION OF

ELECTRONIC DIGITAL METHODS

FOR HIGH-SPEED SCANNING AND SELECTION

CHAPTER 4

CODING INFORMATION FOR MACHINE HANDLING

A. Summary and Conclusions

In order to be handled by electronic digital machines, index terms and symbols must be translated into binary numbers (a number system using only zeros and ones).

The coding may be carried out in one of two basic ways: either the coded index terms are superposed upon one another — that is, the logical sum of the terms is formed — (superposed coding), or an array of the related index terms is formed (array-coding), the order of the terms indicating generic relationships. It can be shown that for the coding of a coordinate-array index, fewer binary digits are required with array coding than with superposed coding. Furthermore, the array coding can be adapted to an expanding collection without redesigning the entire coding plan, while the superposed coding cannot.

For documents each requiring on an average 30 index entries (representing 300 index words or terms) approximately 900 binary digits will be required to code the index entries for a single document. The machine searching methods are shown to be applicable to large files, e.g., a collection of 50 million documents.

B. The Need for Coding; The Binary Code

In order for a machine to manipulate index terms or other symbols, the terms must be translated into a "language" which the machine can "understand." (While this language need not necessarily be a numerical one, it is difficult to conceive of a practical one on any other basis.) The machine language might be, for instance, decimal numbers such as are used in the Universal Decimal Classification. It happens, however, that machines (electronic machines in particular) are inherently better adapted to a base-2 or binary number system. Hence it is desirable to convert the indexing scheme into a binary number system.

C. Requirements for a Coding System

A satisfactory system of coding information for machine handling must take cognizance of the following desiderata:

1) Where speed is a major consideration, efficient use must be made of the digits. The total number of combinations possible with the digits used must not greatly exceed the number of combinations which are usable. In other words, the system must have a high utilization factor.

2) A wide range of vocabulary requirements must be accomodated. The same principles should apply, whether the system is designed for a vocabulary of 2,000 or 2,000,000 terms.

3) The translation from index words to machine language must not be difficult. It will probably be accomplished in a practical

system by persons not highly skilled. The simpler the translation process, from English to machine language, the less errors in coding are likely to be made.

4) The system should be adaptable to coding not only index terms, but also concepts which are not usually expressed in words: such as chemical formulas, chemical structures, musical themes, mathematical formulas, statistical data, electrical networks, relative degrees of an attribute.

5) The code should be able to express relationships between various coded entries. Coding for numerical information does not ordinarily require the expression of relationship between the various terms, but, when coding for ideas, there are advantages. One type of relationship is expressed between the various terms arrayed in descending generic order, indicating essentially the order of the terms in the array. Relationships of a grammatical nature between different index terms or arrays in a block of index entries may or may not be expressed.

D. Basic Methods of Coding: 1. Superposition

The superposition type of coding is applicable where either association (but not order) of the various terms in a generic array or grammatical relationships between arrays, or both, are required. The coded form of a generic array appears as a single field of n tally positions, in which each of the t terms in the array has a random assignment of x tallies.

Example: for n = 13, t = 3, x = 3:

poodle	=	1 0 0 1 0 0 1 0 0 0 0 0 0
dog	=	0 1 0 1 0 1 0 0 0 0 0 0 0
animal	=	0 1 0 0 0 0 1 0 0 0 0 1 0
RESULT	=	1 1 0 1 0 1 1 0 0 0 0 1 0

The chief advantage of superposed coding is that it is very economical of tally positions compared to the simplest type of coding where each coded word occupies a separate group of digits or "field." Its chief disadvantages are 1) the lack of flexibility for expansion once the system is tailored to accommodate a certain number of terms, 2) the requirement that every term must have an absolutely random assignment of tally positions, and 3) that a certain percentage of selection, which can be kept small, will occur that are unrelated to the ones for which the search is being conducted. This latter effect is analogous to "noise," and is inherent in the superposed coding scheme, although the percentage of unwanted items can, at least in theory, be kept below any chosen value.

Where the expression of association of various terms in generic arrays is needed, but which term is associated with which array is not indicated, a single field of tallies may be used. The design formulas for optimum utilization of the tally positions are given below. (The formulas are adapted from work published by Calvin Mooers; the adaptations are carried out in detail in Appendix 1.)

v = total number of terms (generic and specific) in vocabulary

g = number of different types of grammatical relationships to be expressed — this is never less than 1.

t = number of terms applicable to a single array

x = number of tallies (not tally places) required for each term

n = total number of tally places required — total number binary bits required. n = 1.45tx

s = number of terms used in specifying selection

Solution for x and n by trial and error in the following equation:

$$C_x^{(1.45tx)} = \frac{(1.45tx)!}{x!\ (1.45tx - x)!} \geq vg \qquad \text{where } n = 1.45tx$$

The noise level $N \leq (\frac{1}{2})^{sx}$

Example: If the characterization of a document requires 6 arrays of 5 terms, or a total of 30 terms, a total vocabulary of 50,000 is assumed, with 5 types of grammatical relationships, and at least 2 terms specified for selection, then:

v = 50,000
g = 5
t = 30
n = 132
x = 3
s = 2
N = 1/64 = .0156

Where the index terms must be designated as belonging to a particular array, the coding field must have a separate part for each array. Now, for a arrays, the design formulas are as follows:

v, g, t, x, and s are as defined above

n = number of tally positions per array

a = number of arrays

r = total tally positions for a arrays

Solution for x and n:

$$C_x^{(1.45\frac{t}{a}x)} = \frac{(1.45\frac{t}{a}x)!}{x!\ (1.45\frac{t}{a}x - x)!} \geq vg \text{ where } n = 1.45\frac{t}{a}x \qquad r = na$$

The noise $N \leq (\frac{1}{2})^{sx}$

Example: Continuing the previous example, with 6 arrays of 5 terms each:

$g = 5$ $\qquad$ $n = 36$

$a = 6$ $\qquad$ $x = 5$

$\frac{t}{a} = \frac{30}{6} = 5$ $\qquad$ $N = (\frac{1}{2})10 = \frac{1}{1024} = .00097$

$s = 2$

E. Basic Methods of Coding: 2. Generic array

This method of coding is based on sequences of symbols, and is applicable where the expression between various terms in an array is required; by extending the coding, grammatical relationships may be indicated.

In order to construct a code of the generic array type, a choice must be made of a symbol system. If the symbols are to be stored as binary numbers, then the system of symbols should be some base 2^n if possible combinations are not to be wasted.

Each symbol in a sequence of symbols standing for a single index array (equivalent to an index entry) represents a sub-division of the class indicated by the preceding symbol. The number of

subdivisions in any one class may have a wide range. There may be as few as 2 subdivisions, or as many as 400 (as in the case of plant families). Our symbol system must be flexible enough to accommodate this wide range. Using a high-base symbol system (base 20, for example) where the necessary number of subdivisions is small means that a majority of the symbols have no possibility of being used. It follows that maximum utilization will result when the symbol number-base is as small as possible. If we are dealing with a system base 2^n, then the smallest base is 2^1 or 2.

For whatever base we choose, we will find that in a certain percentage of cases the number of needed subdivisions will be greater than the symbol base. By using a pair or a triad of symbols, we have at our disposal symbols of base 2^{2n} or 2^{3n}. Any required number of subdivisions may be accommodated in this way. We are not restricted to 10 subdivisions in a given class, as in the Universal Decimal Classification.

Now we appear to have a dilemma. We need a variable base if the system is to be most efficient; that is, we need a symbol-base with 2, 4, or 8, representations according to the number of subdivisions we require, and to be efficient we must adjust the size of the symbol-base so that the number of useful representations is as large a percentage as possible of the total possible representations. One the other hand, remembering that our symbols must all be reduced eventually to binary numbers, we must have a means for knowing how

many binary digits are used in combination to represent a single symbol. In other words, in a variable-base system, reduced to a series of binary digits, we need some means for telling where one symbol stops and the next one begins. Hence it appears that we must adopt a standard symbol system having a fixed number of binary digits per symbol.

Upon more critical examination, we see that this need not be true. Adopting a base-2 system, and using symbols of any base 2^n where necessary to provide sufficient sub-classes, artificial sub-classes may be created which have no intellectual meaning. For example: let an "internal-combustion engine cycle" be represented by a symbol whose binary number representation is ...010. Let us now choose to divide this into four subdivisions: intake, compression, explosion, exhaustion. We need base-four symbols in order to represent each of the subdivisions; our base-four symbols in binary form are 00, 01, 10, and 11. Had we needed only two subdivisions we could have used simply 0 and 1. In effect, the first of the two binary digits of each symbol represents a division of internal-combustion engine cycle into one of two classes which by itself has no meaning. That is to say, intake and compression are not related to each other in a way which is distinct from the way in which explosion and exhaustion are related. All four are related to each other because together they form an internal-combustion engine cycle. Our coded

representations are now:

...010	internal-combustion engine cycle
...010 00	intake
...010 01	compression
...010 10	explosion
...010 11	exhaustion

The creation of artificial sub-classes without meaning, however, is without adverse consequence.

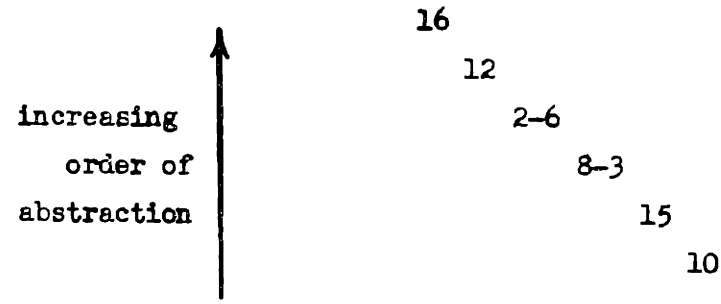
Retracing the procedure on coding the information: In assigning coding symbols to each level of subdivisions, we choose a symbol system of sufficiently large base to just encompass the number of subdivisions. Each symbol, of whatever base, has a binary equivalent. The array of symbols representing successive subdivisions is translated into its binary equivalents, which will give a binary number unique to the smallest subdivision we have chosen. Example: (Capital letters represent the successive levels, progressing downward in a classification tree from the letter A. Each number following the letter (which indicates the "level of abstraction") designated the ordinal of the sub-class. A21 means the top level of the classification tree, and the 21st division at this level.) Let us complete the coded representations begun above.

A1. All human knowledge	(1)	1
B13. Mechanical engineering	(25)	00111
C4. Theory of machines	(7)	100
D3 Internal-combustion engines	(9)	0011
E1. Internal combustion engine cycle	(2)	1
F1. intake	(4)	00
F2. compression		01
F3. explosion		10
F4. exhaustion		11

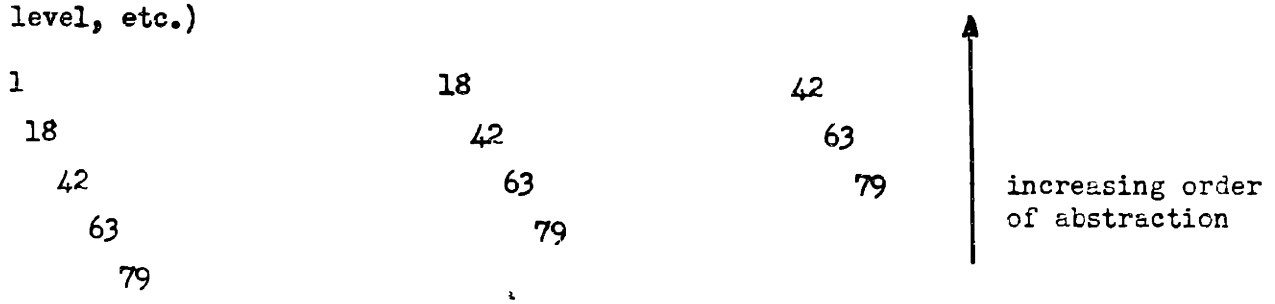
In parentheses at the right is given the total number of subdivisions at each level. The last column at the right gives the binary coded representation of the symbol at the corresponding level. The concept "compression" can now be uniquely specified by the binary number 1 00111 100 0011 1 01, or written without spaces, 1001111000011101.

We now have a flexible system, adaptable to any base symbols at any level of subdivision. We have the option of constructing codes as complex as we may need to conveniently express the different types of information that must be coded, that is, names, arrays of words, and other previously developed coded forms such as the Dyson chemical cipher.

Where a single system of symbols is used, and where any symbol (or combination thereof) may appear at any level of abstraction, it is necessary to know the level at which the array begins. This may be, by convention, at the "top" (i.e., "all human knowledge"), or any other level, as long as it is used consistently, or else the level at which each array begins is separately specified. For these conditions, an array might appear as shown below, where for clarity decimal numbers have been used instead of binary coded symbols.



In the case where we use a different set of symbols at each level, and where all binary-coded symbols consist of the same number of binary digits (not an efficient utilization of binary digits), we are at liberty to begin the array at any level we choose, as the set of symbols to which the first symbol in the array belongs will uniquely indicate in which level the first symbol lies. By way of example, all the arrays shown below are the same array but with a varying number of terms absent from the beginning of the array. (Symbols 1 to 16 belong to the first level, 17-32 to the second level, etc.)



If, for technical reasons to be apparent later, we decide null combinations (binary numbers consisting of all zeros) are to be invalid, the symbol bases available to us which are expressible by binary numbers are $2^n - 1$ instead of 2^n. But n cannot be 1, for that would give a base 1 system, which would be useless, having only one subdivision at each level. Our usable bases would in this case be 3, 7, 15, 31, etc.

This plan of variable symbol bases is extremely useful. We can use the regular alphabet for coding author's names. We can express in binary code any convenient number system, e.g., octal or decimal, for the purpose of designating the location of the documents to which the index entries refer.

If, again, for technical reasons, the entire binary number representing an index entry cannot be stored as a single group of digits, then a series of groups must be stored, with an appropriate indication as to where the binary number starts. This can be done by adding a group with a unique meaning at the beginning of each sequence of groups constituting a single binary number, or alternatively by adding an additional digit on each group. The choice most economical of digits will depend on the numerical values adopted for digits-per-group, and groups-per-sequence.

F. Binary Digit Requirements of the Two Methods of Coding

Referring to Chapter 2, we will examine the number of digits required to index a collection of 50,000,000 documents.

1) Using the superposed coding plan, there are two possible approaches: We may take 30 entries, each one of which is an array of 10 terms, and make a potpourri of 300 terms; or we may keep the entries separate, superposing only the 10 terms within each entry. Using 300 terms superposed in a single number or field, each index term will require three tallies in a field of 1305 tally-places — that is 1305 binary digits will be required to index each document. Using 30 separate fields, one for each index array of 10 terms, each field will contain 6 tallies per term in a field of 87 tally-places. Thirty fields will require altogether 2610 tally places (equivalent to 2610 binary digits). These numbers are derived by the formulas given in Section D of this chapter.

2) Using the generic array coding: assuming a ratio of usable vocabulary arrays (binary numbers, in the final analysis) to the possible number of arrays as being .05 (the utilization factor), the total possible number of arrays is 20 times 50 million, or 1 billion. One billion different binary numbers can be represented with 30 binary digits. Thirty arrays then require a total of 900 binary digits.

It is fairly obvious, then, that the generic array coding plan offers two distinct advantages: it requires about 35% of the digits required by the "equivalent" (more or less) in superposed coding, and has greater flexibility in regard to expansion of the number of entries. The remainder of this thesis will be concerned specifically with the generic array coding plan, but the principles involved will be applicable no matter which coding plan is used.

The arrangement of the binary digits on the storage medium will be to some extent influenced by the requirements of the machine, but this detail will be considered in a later chapter.

BIBLIOGRAPHY

Casey, R. S. and Perry, J. W. Punched Cards, Their Application to Science and Industry. New York: Reinhold, 1951, Chapters 6, 20.

Mooers, Calvin, N. "Theory of Digital Handling on Non-Numeridal Information and Its Implication to Machine Economics." Boston: Zator Company, 1950. 17 pages (Zator Technical Bulletin 48).

Mooers, Calvin N. "Application of Random Codes to the Gathering of Statistical Information." Boston: Zator Company, 1949, 28 pages (Zator Technical Bulletin 31).

Mooers, Calvin N. "Making Information Retrieval Pay." Boston: Zator Company, 1950. 13 pages (Zator Technical Lulletin 55).

CHAPTER 5

GENERAL DESCRIPTION OF THE SELECTION PROCESS

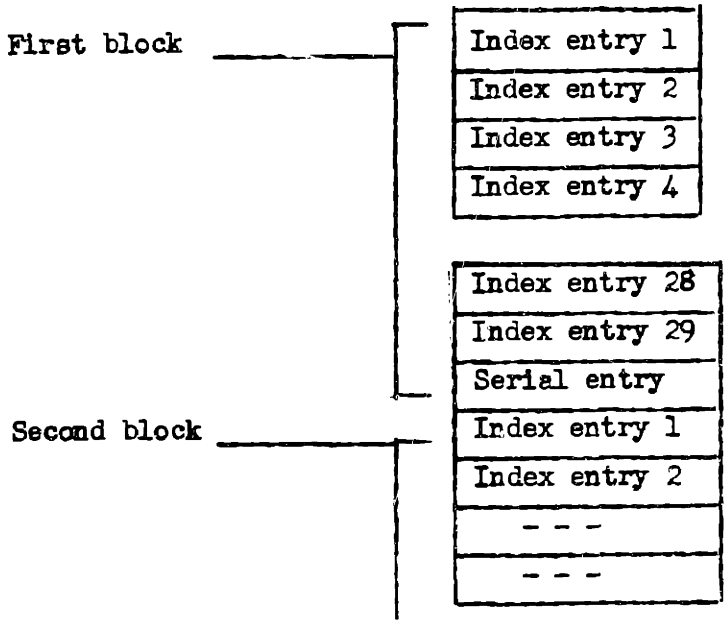
A. What the Selection Process Must Accomplish

In order to retrieve information from a collection of stored material, we must perform a process of selection which will inspect all the stored index information, decide whether or not it is closely enough related to the information requirement (subject) for which we are searching, and, if so, we must be provided with a direct means for locating the original documents. The selection process, then, consists of scanning the encoded index, matching the various index entries and/or terms against a list of criteria (which characterize the information requirement for which the search is being conducted), combining the index entries thus matched and decide whether or not the proper "logical combination" of criteria is satisfied. Last of all, if the document represented by such inspection of index block results in positive identification of a document, then it must be identified appropriately, e.g., by a serial number, in order that the original document be found in the storage file.

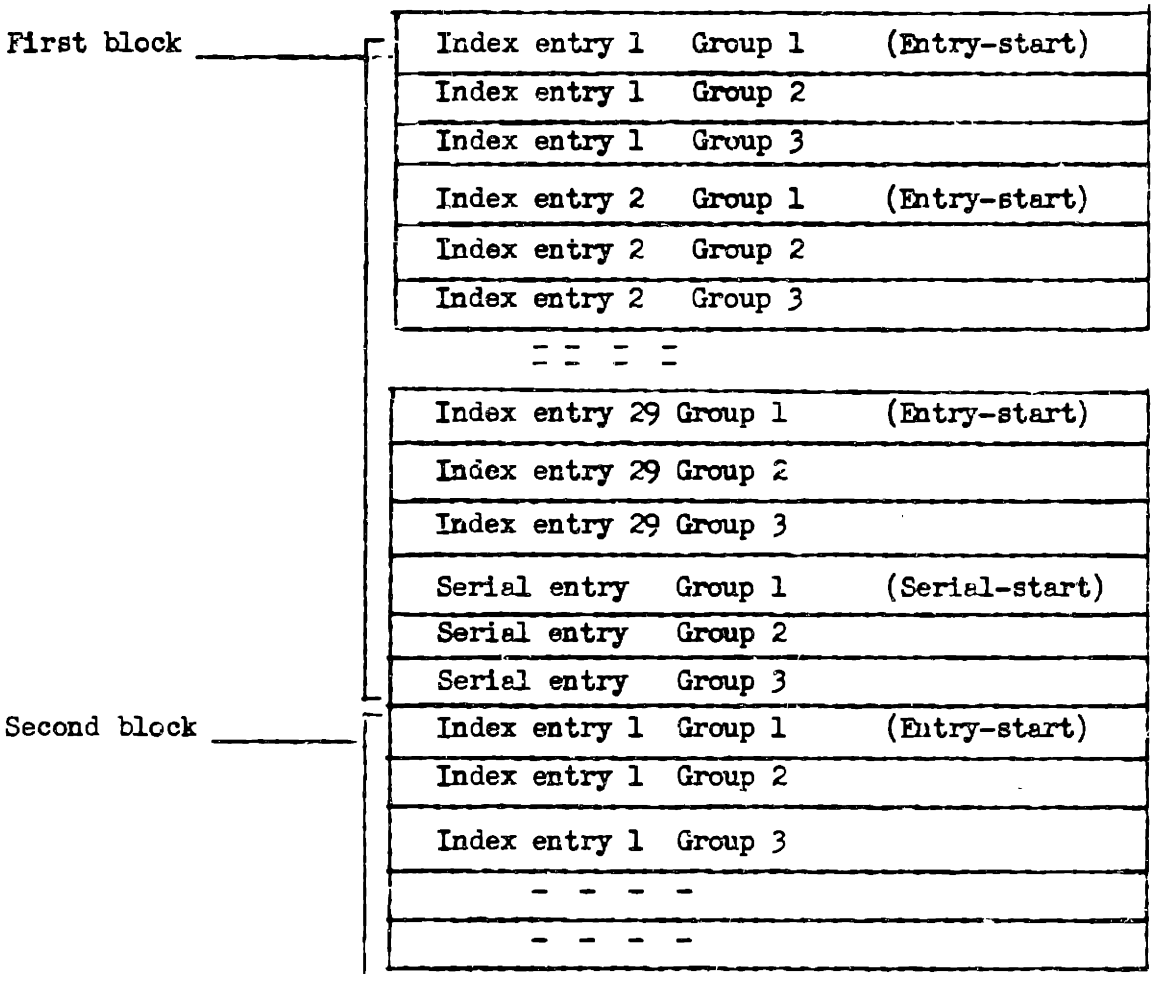
The first step, the scanning process, is essentially one of examining the index for the entire body of stored information. The index will consist of blocks of index entries, each block referring

to one document. For the case where the index is stored on a continuous medium, such as film or tape, the index entries must be scanned in sequence. Because of the physical limitations of the storage media, it may be necessary to record an index entry in several sections. Hence in the scanning process it may prove necessary to reconstitute the entire entry from its component parts. The following examples illustrate in a general way two possible layouts of a block of index entries referring to some one document.

A block of entries is all the encoded information stored together which relates to a single document. A block consists of a series of index entries followed by a serial entry (the serial number of the document).



Alternatively each entry may be broken up into groups of digits, which will form successive rows; an indication must then be provided to show where each index entry or serial entry begins.



The second step, the matching process, consists of comparing each index entry on the storage medium with all the criteria which, in encoded form, are used to define the search being conducted. The matching process may be for the purpose of establishing the absence of certain entries as well as the presence of others. Because of the

properties of the coding scheme (generic array coding) the encoded criteria used to define the search will often require a match with only the initial portion of a recorded index entry. In other words, in the search, we may require only the occurrence of prescribed initial portions of certain index entries. In such a case if the encoded search criterion matches the initial part of an index entry, then the index will be considered to have satisfied that particular criterion.

The third step in the selection process is the combining of the results of a search of a single block of index entries to determine if the required combination of criteria has been met. In our coding scheme, we are not only requesting the simultaneous presence or absence of certain index entries, but we may also require that these terms appear in certain combinations (termed "logical combinations" because they are expressible in the formal algebra of logic). For example, we may require the terms A and B as well as any one of the terms C, D, or E. This decision must be made when the end of the block has been reached.

If a block which has just been scanned satisfies the required combination of criteria, it is then necessary to provide a means for identifying the original document. This can be done by arranging to have the last entry in the block a number representing a document or serial number, which number will be recorded whenever the accompanying index block meets the criteria. The results of a machine search will then be a list of these serial numbers, which may serve as a guide in looking up the original documents.

B. Form of the Stored Index Information

In order to move the stored index information into a machine for scanning at a maximum rate, we have tentatively chosen the medium which seems most suitable at the present time — metal magnetic tape. We have also chosen to make use of the benefits of generic-array coding. In previous discussion, we have seen that the coded information relating to one document will be a block of about 30 large binary numbers, the last of these numbers being the block number or document number for purposes of identifying the document to which the coded block of index entries refers. An illustration of a coded block of information is given in the figure below. Each of these large binary numbers will hereinafter be referred to, for brevity, as "index entries," while the last entry in a block, being the document number, we will give the name "serial entry."

Block 1	10110110011100111001110001110011101	(Index entry 1)
	01101110111011111100101110011001101	(Index entry 2)
	11100111011010111100010100011001111	(Index entry 3)
	00011100111000111100001010110001110	(Index entry 4)
	11011101110000011101110011011000010	(Index entry 28)
	11000011001100011001111001101011101	(Index entry 29)
	00000000000011110011101111110000011	(Serial entry)
Block 2	11100111001000110011000001001001101	(Index entry 1)
	00111001010011001110011100110011101	(Index entry 2)

At present it is not thought feasible to record more than 16 channels or tracks in parallel on a magnetic tape, hence it is not possible to record a complete entry (which would require a digit

in each of 30 channels across the tape). Each entry must be broken into digit groups, each one of which will be a single row of digits across the tape.

Now the necessity arises for indicating in some manner how to reconstruct the entry from the series of digit-groups which occupy successive rows on the tape. This indication may be accomplished simply by marking the first group in each series of groups representing a single entry. Each group so marked will be considered as the start of a new entry. The machine can be constructed to "remember" that each digit group it "sees" is a continuation of the entry the beginning of which was indicated by the last observed marker (entry-start indication).

Because the machine reads only a row of digits at a time from the tape, there must be some indication provided so that the machine can know how to reconstruct the block of index entries. Each block must be distinguished from the next. Furthermore, the serial entry must be distinguished from the rest of the entries. One simple method of carrying this out is to add a marker beside each entry which starts or ends a block. Since the entries will be fed in from the tape in succession the machine can remember that each entry it sees belongs in the same block as the last entry which had an associated mark.

The index information, which pertains to a single document, will appear as follows when encoded. In this form, it is not ready for

machine scanning as coded designations to indicate entry starts and block starts have not been included.

8 tape channels	Special instructions	
10101010	block start, entry start	Block 1
01101101		
10111010		
10011101		
00001000		
11101101	entry start	
00110010		
10111000		
11110001		
10111000		
01010001	entry start	
. .	. . . .	
11110011	entry start	
11110111		
01011101		
01010101		
10011100		
11111100	serial start	
01010111		
00111101		
01110101		
11010110		
10001110	block start, entry start	Block 2
11001101		
. .	. . . .	

The required indications for entry-and serial-start can be made in one of two ways. In one scheme, a unique group of digits may be used for each of the two types of indications, as represented in the figure below:

00000001 (Entry start)
10101110
11011101
11011110
00111011
00001101
00000001 (Entry start)
00110110
11011010
01011101
00000111
00001110
11111111 (serial-start)
11000000
10101010
01111011
01111011
10101110
00000001 (entry-start)

If this method be used and if all the blocks are formed in a similar manner, in that they contain index subject entries followed by a single serial entry, the machine can remember to consider the entry succeeding the serial entry as the first index entry in a new block. We would then not need to separate a separate marker to indicate the beginning of a block. Markers to indicate the start of an index entry and the start of a serial entry would suffice.

The alternate method for indicating entry- and serial-starts utilizes additional channels on the tape to enter binary digits which signify either of the two types of start indications, as shown in the figure below:

(Index entries)	(Entry-start)	(serial-start)
11011101	1 ◁	0
11011010	0	0
11101110	0	0
01101101	0	0
01111101	0	0
10101101	1 ◁	0
10101101	0	0
00100110	0	0
10011010	0	0
01000010	0	0
11001100	0	1 ◁
11011101	0	0
00001111	0	0
00111011	0	0
00011101	0	0
11110110	1 ◁	0
00011010	0	0
00011011	0	0
01110101	0	0
00111101	0	0
. .	.	.

(The simultaneous indication of entry-and serial-start might be used to mean block-start, if it were found desirable.)

To sum up, then, a block of index entries referring to one document will be made up of any desired number of index subject entries followed by a single serial entry. Each entry will be composed of a sequence of binary digit groups. Each group which begins an entry must be identified as an initial group, and must indicate the type of entry, either index or serial. If the final entry is invariably the serial number, then a block start indicator is not needed.

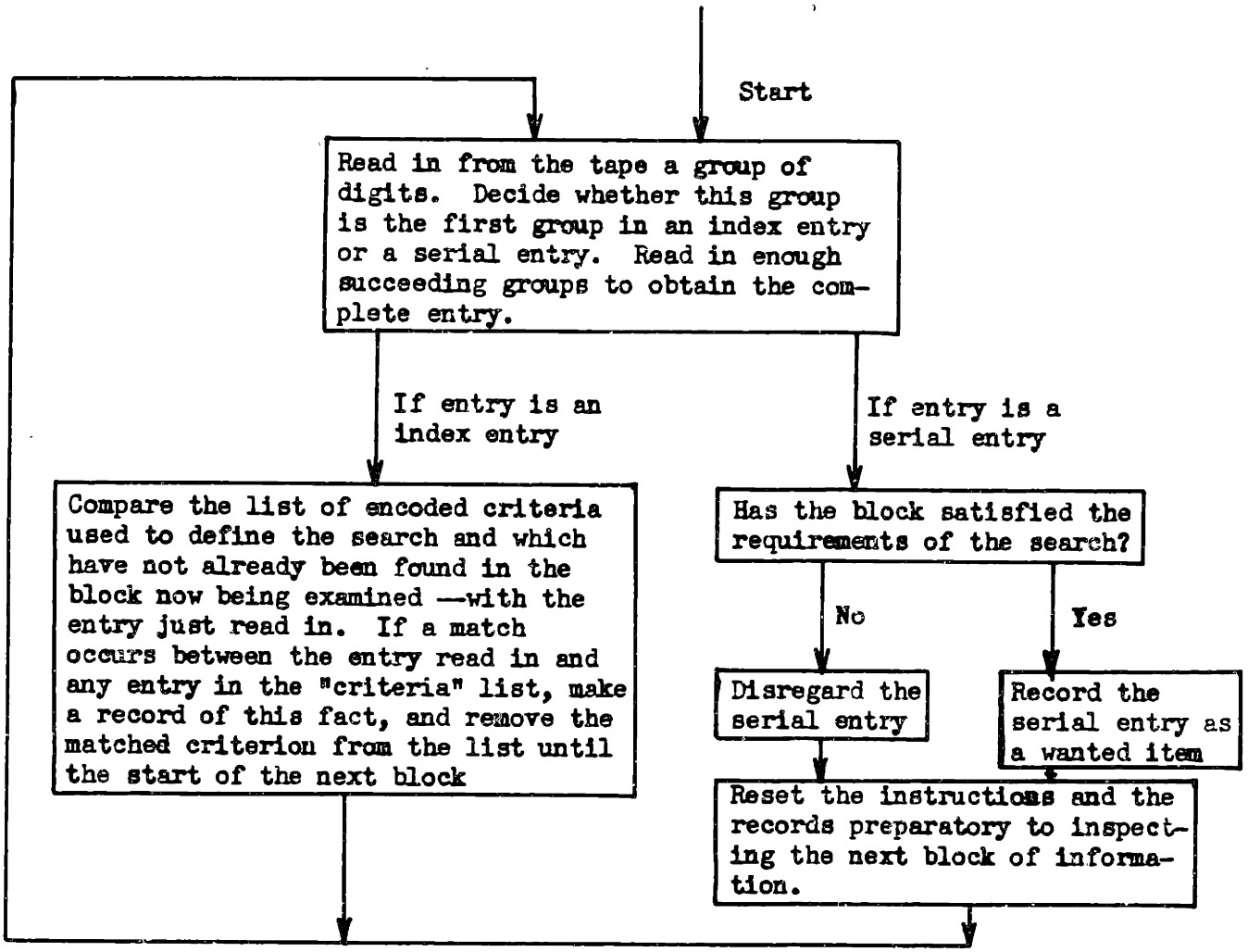
C. The Meaning of the Flow Diagram

A flow diagram or flow chart is a pictorial way of describing a sequence of operations. The choice of which part of a sequence of operations is to be performed may be based on the result of decisions arrived at through previous operations. To read a flow diagram, we begin at the box labeled "start." We perform the instructions which are enclosed in that box. If a decision is required, the decision is made, and an appropriate exit from the block is made, based upon which of the alternative decisions was correct. Following the arrows or lines of flow, we proceed to the next block. The process then repeats.

In this manner it is possible concisely to diagram complicated sequences of operations. Depending upon some of the decisions arrived at, operations may be performed in cyclic form; that is, a certain sequence of operations may be performed repeatedly until a change in one of the decisions carries the line of flow to another section of the flow diagram.

The flow diagram in the figure below shows the order of operations which a machine must perform in searching a block of index information.

Depending upon whether the entry being read by the machine is an index subject entry or a serial entry we take one of two alternate courses of action. When we have reached the bottom of the flow diagram, we follow an arrow which returns to the beginning to initiate a new cycle, that is, to start again the entire sequence of operations.



Flow diagrams are, in actuality, merely a detailed logical analysis of the sequence of inspections, computations, and decisions which the human mind makes almost without cognizance of the logical sequence of operations which it goes through. Any complex course of action, which a rational human being can follow in a consistent logical manner results in sequences of operations which can be represented in graphical form in a flow diagram.

Flow diagrams are used extensively to plan the programming for logical devices, such as digital computers. In this case we have used the flow diagram to demonstrate the general operations which a scanning and selecting device must perform in order to perform an information search. In Chapter 6 a very detailed flow diagram showing the particular steps taken by a general-purpose electronic digital computer will be presented.

D. General Steps in the Process of Scanning, Matching, Combining and Identifying

The flow diagram above shows the order of operations which must be performed in searching a single block of index information to determine if it contains a specified combination of index entries. This sequence of operations is essentially the same regardless of whether a machine or a human being performs the search. The human mind performs the steps so quickly, of course, that there is usually no consciousness of the discrete steps which comprise the operations.

In searching a storage tape containing recorded index information the object is to obtain the serial number of each block which

contains a specified combination of index entries. Now the selection process which the machine must perform is the following:

a) Obtain from the tape a group of binary digits. If it is a group of digits which begins an index entry, inspect it to see if it corresponds to the first group of any of the encoded criteria used to define the search being conducted.

b) Read in succeeding index entry groups to see if they match criteria for which there was a match in preceding groups of the block being searched.

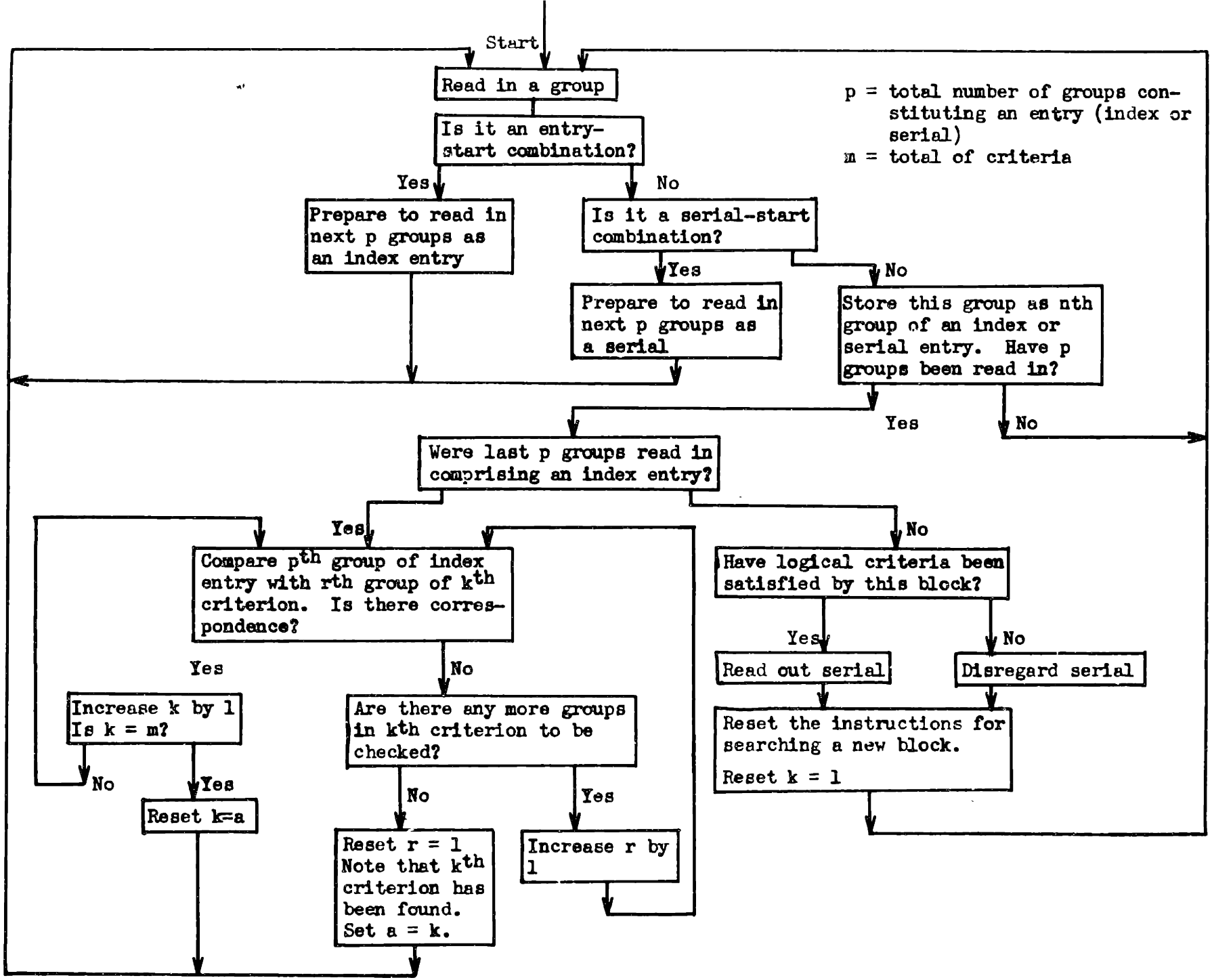
c) Observe and note all the criteria matched by the various sequences of digit groups which constitute the index block.

d) Repeat a), b), and c) for each index entry in the same block.

e) When the serial number is read in, hold it temporarily. Determine if the "logical combination" of index entries was located within the associated block.

f) If so, record the serial number.

Because a piece of machinery cannot think, in the same same sense that a human being thinks, the sequence of operation must be specified in great detail, in such a simple-minded fashion, one might say, as to make it unmistakeably understandable to the simplest moron — for the machine is, in fact, much less intelligent than a moron. The drawing on the next page is a more detailed flow diagram showing the order of operations performed by the machine in searching a block of index entries. The two flow diagrams given in this chapter are


Start
Read in a group
Is it an entry-start combination?
Yes
No
Prepare to read in next p groups as an index entry
Is it a serial-start combination?
Yes
No
Prepare to read in next p groups as a serial
Store this group as nth group of an index or serial entry. Have p groups been read in?
Yes
No
p = total number of groups constituting an entry (index or serial)
m = total of criteria
Were last p groups read in comprising an index entry?
Yes
No
Compare pth group of index entry with rth group of kth criterion. Is there correspondence?
Yes
No
Increase k by 1 Is k = m?
No
Yes
Reset k=a
Are there any more groups in kth criterion to be checked?
No
Yes
Reset r = 1 Note that kth criterion has been found. Set a = k.
Increase r by 1
Have logical criteria been satisfied by this block?
Yes
No
Read out serial
Disregard serial
Reset the instructions for searching a new block.
Reset k = 1

strictly applicable to a machine which can perform a single operation at a time. The human brain is not this limited in its ability. It is able to attend to more than one detail at once, or at least in such rapid succession that it may be considered essentially simultaneous. It is possible to design electronic machines so that they may perform two or more operations simultaneously. Design details will be presented in Chapter 7.

E. Continuous Storage Media (Film, Tape) Versus Punched Cards

Most of the discussion in this chapter has applied principally to the selection of information which is stored on a continuous medium, and which presents to the machine the index entries in sequence. The remarks are not strictly true where the storage medium is a film or a punched or similarly marked card. Because the information on a film or card is physically spread out, it is possible to examine an entire block or portion thereof simultaneously. (With magnetic tape this cannot be done. It is impossible to read adjacent digits in the same channel simultaneously because the reading heads cannot be placed close enough together.) It is evident that in the case of film or cards there is no necessity for assembling the various stored group of digits into the entries which they constitute. The various entries must, however, be compared in some fashion to all the criteria and a decision must be made based on whether or not logical combinations of criteria are satisfied, and the desired serial numbers obtained.

The physical features of film or cards makes possible a few simplifications in the mechanical operations necessary for scanning. Where it is permissible to mark (by spot or hole) n places out of p possibilities so that the selecting device is always looking for n out of p marks constituting an index entry or portion thereof, a simple optical system based on the principle of blackout may be used. To illustrate, let us suppose a character is indicated by n marks out of p possible marks. In order to identify the character, an optical mask is used with marks corresponding to the unmarked locations of the desired character. When this mask is coincident with the desired character, then and only then will an optical blackout occur. This principle is utilized in the Microfilm Rapid Selector.

BIBLIOGRAPHY

Engineering Research Associates. "Report for the Microfilm Rapid Selector." St. Paul, Minnesota, and Arlington, Va.: Engineering Research Associates, 1949, 30 pages, lithographed.

CHAPTER 6

THE GENERAL-PURPOSE ELECTRONIC DIGITAL COMPUTER

A. The General-Purpose Computer as a Flexible Information-Processing System

The problem of developing a machine searching system may be discussed under two headings: designing a machine to do the searching, and designing an information code that the machine can handle. Because the two parts of this problem are interdependent, the machine cannot be designed without some knowledge of the code, but the code cannot be completely determined without some knowledge of the operations that the machine is able to perform. Either some assumptions must be made about the coding in order to plan a machine, or some assumptions must be made about a machine in order to plan a code. Since an electronic digital computer represents as fast and as flexible an electronic machine as has been designed, it would seem reasonable to adopt this machine as a logical starting point for a study of the capabilities of electronic machines. By studying the applicability of the general-purpose computer, some modifications may become apparent which will make the machine more suitable for information searching.

B. A Description of the Computer

In carrying out the study of an electronic digital computer, it is intended to take as a representative computer the machine known

as Whirlwind I, now being developed at the Electronic Computer Division of the Servomechanisms Laboratory at the Massachusetts Institute of Technology. This machine is a general-purpose electronic digital computer operating on a parallel-digit basis (all the digits of a number treated simultaneously instead of successively), and using the binary number system. Each number (or word, as it is often referred to) handled by the machine is 16 binary digits in length. The machine is designed to have 2048 storage registers which will store 2048 words. In addition to internal storage in the machine, there will eventually be external storage of library films containing permanent sets of instructions (subprograms) for computing trigonometric functions, square-roots, and the like. Whirlwind I is at present the fastest electronic digital computer in operation. The average time for an operation such as addition, transfer to storage, or comparison, is approximately 50 microseconds (.00005 seconds).

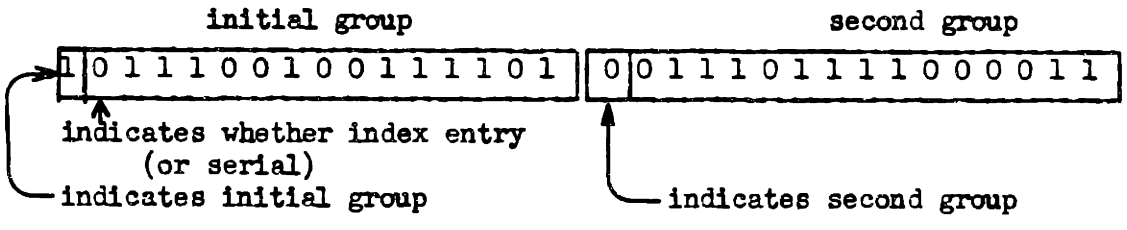
In Appendix 3 are given the general principles of writing sets of instructions (programs) for the computer and a detailed list of the operations the computer can perform. A short sample program is included in Appendix 3 for illustration.

C. Preparing a Program for Searching

The details of a searching program will vary, depending on 1) the number of coded entries and the number and size of the pieces into which the coded entries must be broken down for storage purposes,

2) whether the number and size of the coded entries is variable or fixed, 3) the manner in which various entries are tagged to indicate the start of an index entry or serial entry, 4) what logical combination of entries must be located, and 5) whether all criteria must be test-matched against all entries or some method of elimination of useless comparisons can be developed. Since these details are not of major importance, however, some convenient assumptions may be made in order to be able to work out a specific program. The resulting program will be representative only; it will be neither the only way nor the optimum way. It will be shown later that optimization is not a decisive factor, since there are no significant advantages to be gained by changing any of the details.

Because the contemplated storage media (magnetic tape) and the present version of Whirlwind are both limited to handling groups of 16 binary digits, it will be assumed that an entry is constituted by a pair of 16-digit numbers. The first digit in each group will indicate whether the group is the initial or second group of a pair. The second digit in the initial group will indicate whether the pair is an index entry or a serial number. Pictorially, a single entry is:

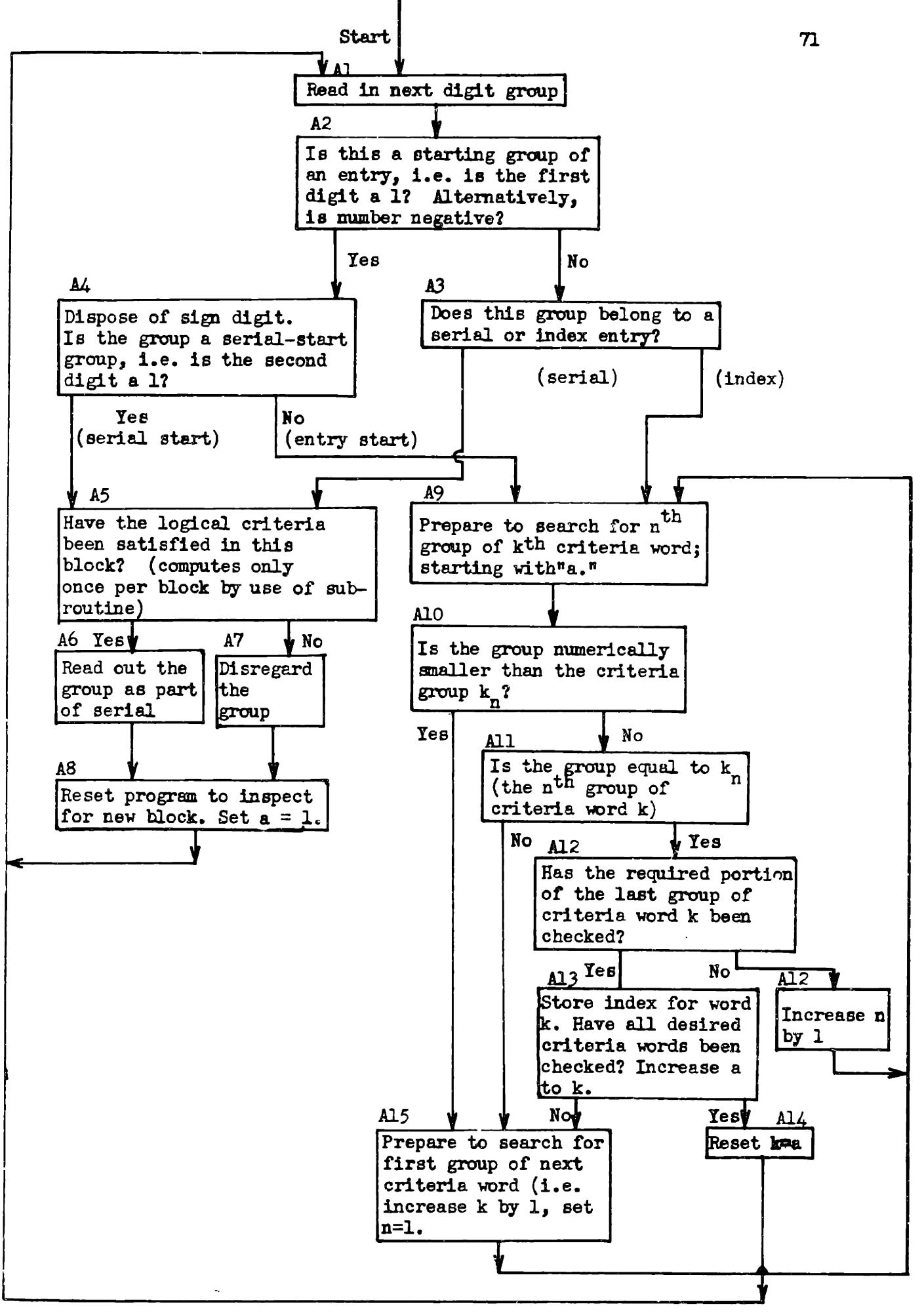


Each encoded criterion, used in accomplishing a search by matching index entries, will in the general case be shorter than — i.e. consist of fewer digits than an entry. The criterion will be in an integral number of groups, but with each group will be associated a number indicating the number of digits in the group which constitute the criterion, the remainder of the digits being meaningless and not to be matched. A criterion might be illustrated pictorially thus:

initial group	second group
1 0 1 1 1 0 0 1 0 0 1 1 1 1 0 1	0 0 1 1 1 0 1 1 1 1 1 0 0 0 0 0
16 digits to be matched	V digits to be matched —the last (16-v) digits need not be matched.

If criteria and index entries are stored and handled in numerical order, then checking of every criterion against every index entry can be avoided.

The flow diagram on page 71 shows the detailed steps the computer must go through to search a block of index entries, based on the assumptions made in the preceding paragraphs. The flow diagram is essentially a more detailed version of the flow diagram appearing in Chapter 4. An explanation of flow diagrams may be had by referring to Section C of Chapter 4. In summary, the flow diagram is merely a graphical way of depicting the order in which various operations are to be performed and various decisions made. In general it has internal loops or cycles, around which the sequence of operations may proceed repeatedly. The entire sequence of operations to inspect one block of index entries is a tortuous path through the flow diagram.


Start
A1
Read in next digit group
A2
Is this a starting group of an entry, i.e. is the first digit a 1? Alternatively, is number negative?
Yes
No
A4
Dispose of sign digit. Is the group a serial-start group, i.e. is the second digit a 1?
Yes (serial start)
No (entry start)
A3
Does this group belong to a serial or index entry?
(serial)
(index)
A5
Have the logical criteria been satisfied in this block? (computes only once per block by use of sub-routine)
A6 Yes
Read out the group as part of serial
A7 No
Disregard the group
A8
Reset program to inspect for new block. Set a = 1.
A9
Prepare to search for n^th group of kth criteria word; starting with "a."
A10
Is the group numerically smaller than the criteria group k_n?
Yes
No
A11
Is the group equal to k_n (the nth group of criteria word k)
No
Yes
A12
Has the required portion of the last group of criteria word k been checked?
No
A12
Increase n by 1
A13 Yes
Store index for word k. Have all desired criteria words been checked? Increase a to k.
No
Yes
A14
Reset k=a
A15
Prepare to search for first group of next criteria word (i.e. increase k by 1, set n=1.

In Appendix 4, the flow diagram has been transformed into an actual program, which requires, on the average, four computer operations for each rectangle in the flow diagram. This is a straightforward program, written without any programming short cuts or tricks. By careful reworking, and making full use of the many extra facilities of the machine, the program might be shortened a small amount. There appeared to be no advantage in doing this, however, since the program is intended principally to give an idea of the order of magnitude of the operations involved.

Also in Appendix 4 is a discussion of the basic operations required for determining if two numbers match, and for performing computations according to the rules of logical algebra.

D. Results of Minor Modifications of the Computer

By making a few minor changes in the computer, such as wiring in some "orders" or operations which are not included in the present list of orders, some parts of the program can be simplified without affecting the suitability of the computer for mathematical purposes. Unfortunately, the net saving in the total number of operations required for searching an index turns out to be only a small percentage.

Some of the special orders which might be instituted would be the following:

1) Logical addition, the addition of two binary digits according to the rules of logical algebra.

2) Logical multiplication, the multiplication of two binary digits according to the rules of logical algebra.

3) Comparison, the choice of one of two courses of action depending upon whether or not two numbers are equal.

4) Complementing, the exchanging of 1's for 0's and of 0's for 1's.

These special orders would perform the equivalent of the processes which require, with present orders, 4, 3, 6, and 4 operations respectively.

More details concerning the special orders are given in Appendix 4.

E. Results of Major Design Changes in the Computer

With some major additions to the computer, it is possible to obtain a faster processing speed than can be had with the present design.

The principal changes which will permit a faster searching process involve adding a switching circuit for logical computation, and extension of the length of certain registers to accommodate a total of 32 to 38 digits instead of the present 16. This latter change will allow an entire entry to be handled as a unit instead of in groups. With a few special orders, involving additional switching facilities, the tedious bookkeeping now necessary can be virtually eliminated; the number of operations would be reduced nearly to the theoretical minimum for this type of machine which is characterized by making comparisons in sequence.

These modifications are discussed in greater detail in Appendix 4.

F. Relative Speeds of Operation of the Various Programs

It is impossible to make a general, accurate prediction of the length of time required to search a block of index information, as the number of operations required is directly dependent on the number of matches which occur between the criteria and the index entries. Some rough estimates may be arrived at, however.

The maximum processing time for one piece of data (one 16-digit group) is that for the group succeeding the initial group of an index entry. For a block containing 30 entries in 60 groups, a search using a program with existing orders, directed toward 8 search criteria will require for a single group a maximum time equivalent to processing 8020 orders. At 50 microseconds per order, the searching required for a single group would be about .4 seconds. Because the input medium must be run at a constant speed (for mechanical reasons) this maximum processing time for a single group limits the rate of input to 2 1/2 groups, or 1 1/4 entries, per second.

If special orders are used, such as those mentioned in Section D above, some portions of the program may be slightly shortened. An examination of the program leads to the estimate that up to 1000 orders might be eliminated and the maximum processing time for a single group might be reduced to .35 seconds, a saving of 12 1/2%.

Taking an overall view, it is seen that even if the time required could by some means be squeezed to .1 second, the processing time for a single 30-entry block would still be 3 seconds — hardly as fast as a mechanical device could operate. (A search of 50 million

blocks would take 150,000,000 sec, which is 41,700 hours, or 4170 ten-hour days!)

G. Input and Output Media

Input and output media are not decisive factors because of the relatively slow speed with which the (unmodified) computer can process the stored information. In other words, it is no problem to keep the computer supplied with information to process, and it is no problem to record all the information which the computer can put out. The computer with major modifications does operate significantly faster. The problems involved where a high rate of input and output are necessary are discussed in Chapter 7.

It is, of course, fairly obvious that any input medium or output device which has mechanical inertia must be operated at a constant speed if serious mechanical handling problems are to be avoided. This further implies that the longest sequence of operations which the computer must perform between reading successive entry groups fixes the maximum rate at which the input device can be run.

H. Comment on the General Philosophy and Abilities of the Electronic Digital Computer

As will become evident in Chapter 7, the applicability of the general-purpose computer to the information searching problem is impaired because of the necessity of performing operations singly and in sequence.

One of the purposes of studying the computer was to determine whether there were any special features or abilities of the computer

which might have a particular application to the problem of information searching. Up to the present time, there appear to be none.

BIBLIOGRAPHY

Engineering Research Associates. High-Speed Computing Devices. New York, McGraw-Hill, 1950.

Huskey, H. D. "Terminology and Block Diagrams For Use In Connection With Automatic Digital Computing Machines." Washington: National Bureau of Standards, 1950.

Institute of Radio Engineers, Standards Committee. "Standards on Electronic Computers: Definitions of Term, 1950. I R E Proceedings, March 1951, pp 271-277.

Project Whirlwind (John Salzer). Functional Description of the Whirlwind I Computer. Cambridge, Mass.: Massachusetts Institute of Technology, 1950. 2 volumes, hectographed. (Whirlwind Report R-180.)

Project Whirlwind (David Israel). Introduction to Coding. Cambridge, Mass.: Massachusetts Institute of Technology, 1950. 82 pages, hectographed. (Whirlwind Engineering Note E-2000-1.)

Project Whirlwind. (Hrand Saxenian). Brief Description of the Whirlwind I Computer. Cambridge, Mass.: Massachusetts Institute of Technology, 1951. 60 pages, hectographed. (Whirlwind Report R-196.)

University of Pennsylvania, Moore School of Electrical Engineering. Automatic High-Speed Computing: A Progress Report on the EDVAC, September 30, 1945. Philadelphia: Univ. of Pa., 1945. 111 pages.

University of Pennsylvania, Moore School of Electrical Engineering. Description of the ENIAC and Comments on Electronic Computing Machines. Washington: National Defense Research Council, Applied Mathematics Panel, 1945. 65 pages.

University of Pennsylvania, Moore School of Electrical Engineering. Theory and Techniques For Design of Electronic Digital Computers. Philadelphia: Univ. of Pa., 1946. 4 volumes.

CHAPTER 7

A SPECIAL-PURPOSE ELECTRONIC DIGITAL MACHINE

A. Philosophy of Sequential Versus Simultaneous Operations

Because the selection process is one that can be written out in terms of simple checks for identity and a systematic method of keeping a record of the intermediate results, it is entirely possible to use a general-purpose digital computer to scan a coded index for identifying and selecting purposes. (A general-purpose computer is defined here as one which is designed for general mathematical work.)

The basic philosophy of general-purpose digital computers is to have one computing element which performs all the operations on the data. The rest of the machine is devoted to storage, control, and input-output mechanisms. By using a single section which is common to all the operations to be performed, one can afford to make this element very complex and very versatile. A necessary concomitant, however, is the fact that operations are restricted to being performed one at a time. In many cases, because of the high speed of the machine, the fact that all operations must be done in sequence is of little or no consequence. On the other hand, where the aim is the highest possible speed, the sequential operation approach creates a bottleneck. It would seem advisable, then, in considering the specialized requirements of index searching, to turn our attention to a special-purpose machine which performs as many operations simultaneously

as it is logically possible to perform. It is not logically possible, of course, to perform two operations simultaneously where one operation is based on the results of the other operation.

It must be emphasized that no matter how fast a digital machine may operate, a requirement that certain operations be sequential limits the possibility of conserving time. In order to take the next big step to higher speed of scanning and comparing, all criteria words used to define the search must be compared simultaneously to each index entry which is being scanned.

The general-purpose computer can perform only a single elementary step at a time, such as an addition, a comparison, or a transfer. The process performed by the computer is built out of a great many such simple steps taken in appropriate order. It has been shown that such a search as the one outlined in Chapter 6, performed on an unmodified computer at 50 microseconds per step (roughly the time required for a single order in the Whirlwind I computer) will require a time on the order of 3 seconds to search a single block of index entries! Realization that the elementary steps are performed in sequence one at a time leads to the speculation that there might be a way to perform more than one step at a time. It turns out to be quite possible to build a machine which can perform groups of operations simultaneously, and hence which can carry on a routine search in a more efficient manner than the general-purpose machine. This special-purpose computer is equipped with a unit whose sole function is to watch for a single sequence of digit groups constituting a specific index entry and to notify the central control element whenever this entry is recognized. As many units of this type are utilized simultaneously as there are different search criteria to be matched with index entries.

By introducing radical modifications to the general-purpose computer (such as are presented in Appendix 4), the searching time for a single block may be decreased to .01 seconds. However, since radical modification is nearly as much effort as redesigning the computer, but produces results not nearly so effective, it seems unwise to consider that modification provides a satisfactory solution.

The general-purpose digital computer is virtually disqualified as a selection device because of the fact that a substantial portion of the total time is spent in transfer operations, moving numbers back and forth between the storage element and the operations element of the machine. The special-purpose machine, on the other hand, can devote a single unit to each operation and be relieved of the burden of transferring information about internally, which is necessary in the general-purpose machine in order to utilize the single operations element for all the different operations.

B. The Machine Operations in Detail

The special-purpose digital-computer, hereinafter referred to as the "searching machine," will be able to compare simultaneously a single index entry against all the criteria words. The operations of controlling the behavior of the machine, of making logical decisions, and of reading out the desired information are operations which are performed in sequence exactly as they would be performed on the general-purpose machine. In the searching machine, however, the time required for this sequence of operations can be reduced. This is possible because a special-purpose device, limited in its scope of operation, can be made with a simpler switching mechanism than can a general-purpose machine. We may consider, then, that the figure on page 71, which describes in a detailed flow diagram the operations to be performed

by the general-purpose machine, is applicable to the proposed searching machine.

One object in the design of the searching machine will be to reduce the various operation times to the minimum possible.

C. Detailed Description of the Searching Machine.

The input information (the stored index information) will be a succession of binary digit groups which must be identified as either the initial group of a binary number representing an index entry, the initial group of serial number, or a succeeding group of either type. The type of a succeeding group need not be specified, since the machine can "remember" what the last "initial group" indication was.

There must be provided a synchronizing signal so that the machine will know when to act. One might assume that as long as there appears at least one digit in any row, the machine can use this digit as an indication to read and compare the entire group or row of digits. By using a large numerical base for the symbols used in the indexing, however, it may happen that a digit group will occur which is composed entirely of significant zero-digits (which are recorded as unmagnetized spots on the tape). (For example: the first symbol in a base-256 system would be one; expressed in binary form, 00000001. If the index entries were recorded in groups of seven or fewer binary digits, it might then happen that a digit group would be composed of the seven zeros.) Provided that symbols which are designated by all-zeros are not used in the indexing, and that the longest symbols used are not greater than twice the recorded digit-group length, then two successive

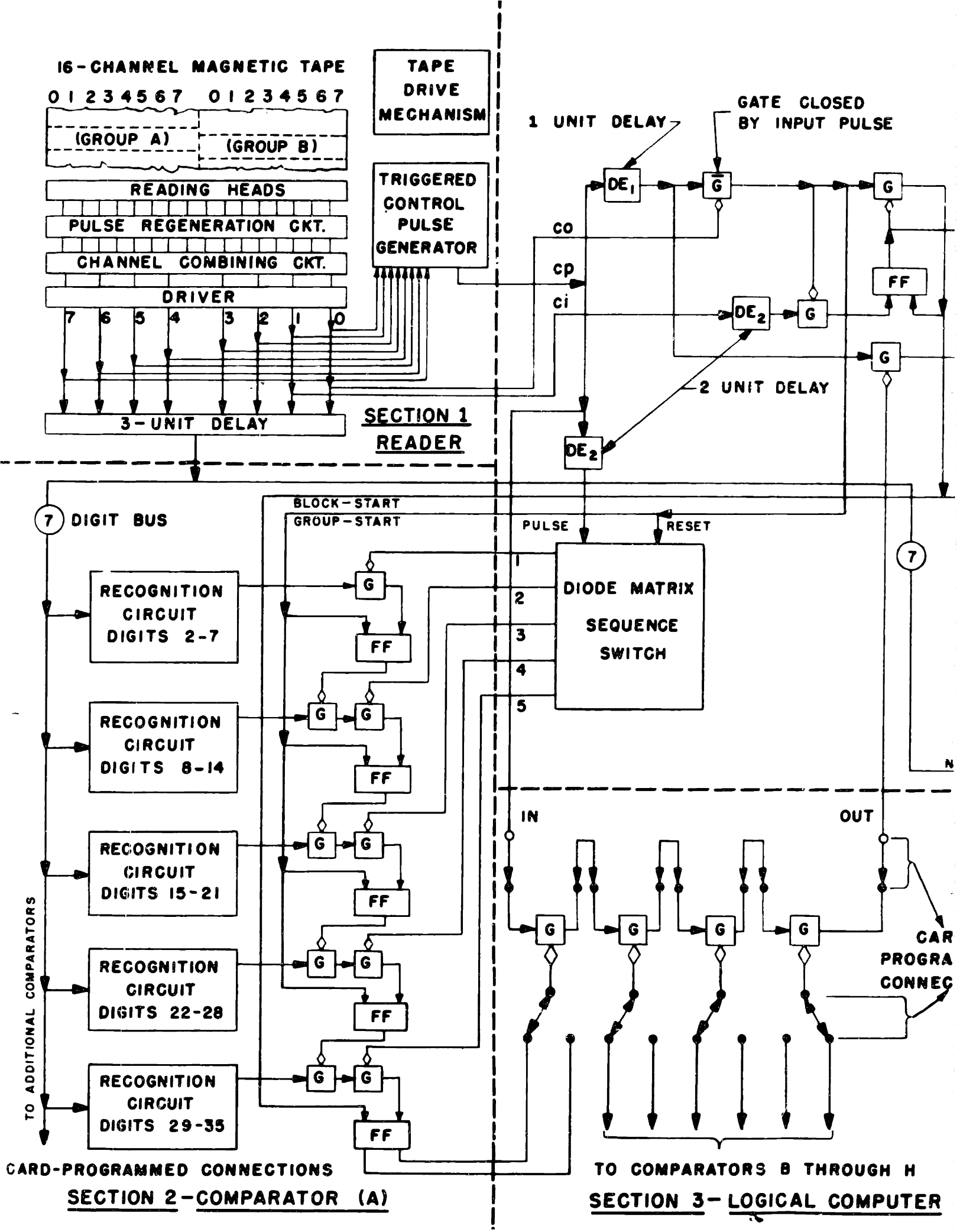
digit groups could never both be all-zeros. This fact permits a simple solution to the problem of providing a synchronizing signal. Arrange a triggered sync pulse generator to be triggered by the logical sum of the total of all the digits in a digit-group, but provide the feature that the pulse generator create one additional pulse after the last trigger pulse.

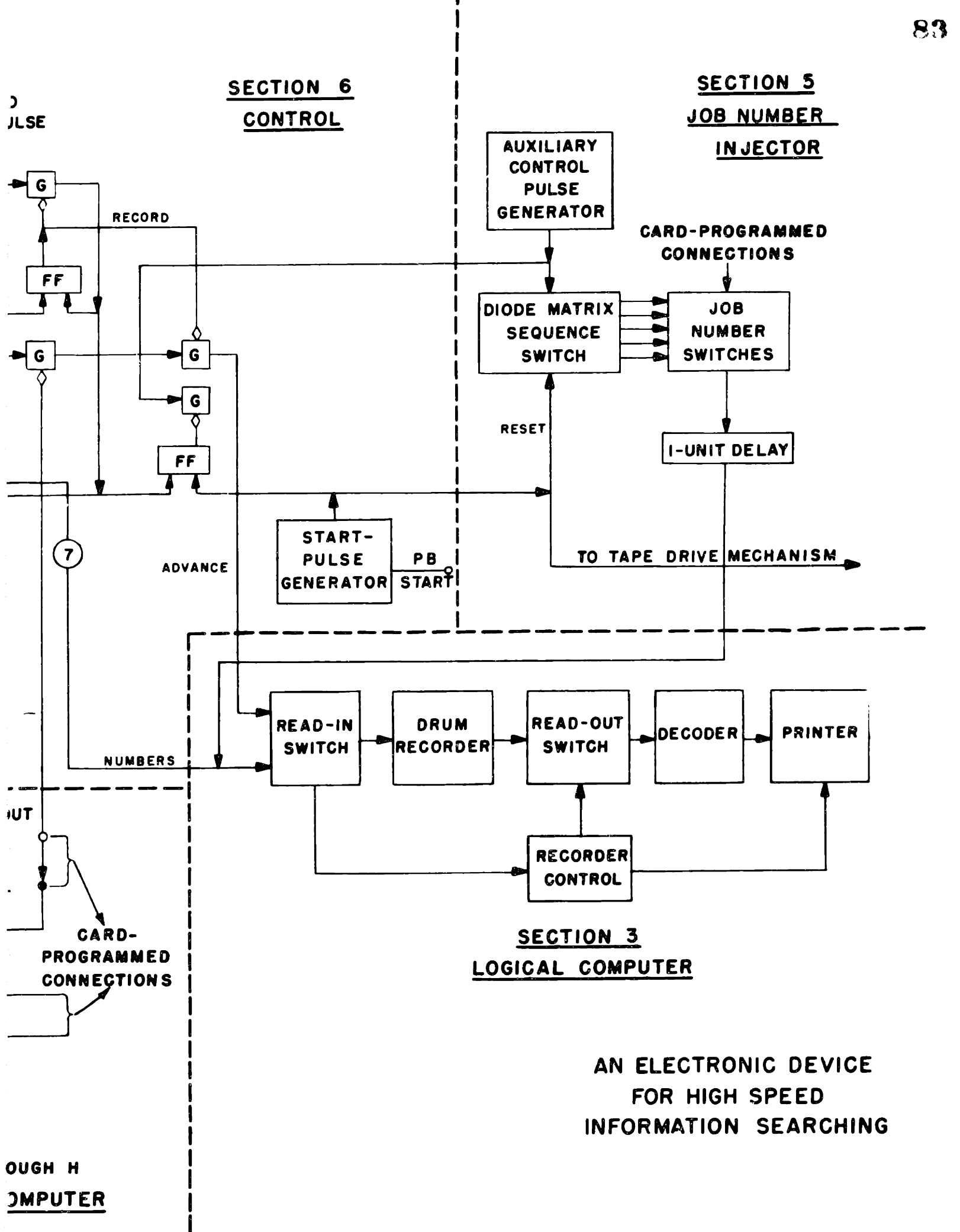
An alternate and less complex method is to arrange the coding so that an initial group will never be all zeros. Make provision for an additional digit in each row by adding a "control" channel. Then use the absence of a digit in the control channel to indicate the initial group and the presence of a digit in the control channel to indicate a succeeding group. All succeeding groups of digits will have at least a digit in the control channel, and the initial group is arranged never to be zero, hence the logical sum of all the digits in a row (including the control channel) can be relied upon to furnish a synchronizing pulse whenever a digit-group of any type occurs. Because serial numbers may start with a number of zeros, choose the initial digit of all serial numbers to be a 1. The design of the machine will be based on this latter scheme of providing sync pulses.

There are several sets of data which are to be put into the machine before a search is started. 1) Each of the encoded criteria corresponding to the index entries or portions thereof which must be found, 2) The required logical combination of the index entries which must be satisfied, and 3) The "job" number, or order number, which is merely transmitted through the equipment in order to indicate

on the printed record produced by search a number identifying the latter. The setting of switches to provide the proper connections can be simplified by preparing cards punched with the appropriate binary data. These cards can be electrically sensed and used to set up the necessary connections in a shorter time and with greater accuracy than could be done manually.

The READING MECHANISM will read the digit groups from the tape and generate a pulse corresponding to the presence of each binary digit. It is estimated that the maximum number of channels which is practical at the present time is 16. For reasons which will become apparent later (see page 91), the reader will read alternately from Channel group A (channels 0 to 7) and Channel group B (channels 8 to 15). Channel groups A and B are combined to form a single group of eight channels. The pulses picked up by the magnetic reading heads must be regenerated into sharp trigger pulses (of .1 microsecond or so duration) with sufficient power to drive other portions of the circuit. Channels 0 and 1 will contain control information and must be fed to the control equipment. Because the logical sum of all channels will be used to generate the synchronizing pulses, outputs from these channels must be fed to a Control Pulse Generator. The channels carrying index information (channels 1-7) must be fed to the Comparator and to the Recorder. Appropriate time delay of the pulses must be provided, however, so that the switching circuits will have time to prepare the proper transmission paths. The Reader is shown in Section 1 of the block diagram on page 83.


16-CHANNEL MAGNETIC TAPE
0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
(GROUP A)
(GROUP B)
TAPE DRIVE MECHANISM
READING HEADS
PULSE REGENERATION CKT.
CHANNEL COMBINING CKT.
DRIVER
7 6 5 4 3 2 1 0
TRIGGERED CONTROL PULSE GENERATOR
3-UNIT DELAY
SECTION 1 READER
1 UNIT DELAY
GATE CLOSED BY INPUT PULSE
DE1
G
CO
CP
CI
DE2
FF
2 UNIT DELAY
7 DIGIT BUS
BLOCK-START
GROUP-START
PULSE
RESET
RECOGNITION CIRCUIT DIGITS 2-7
RECOGNITION CIRCUIT DIGITS 8-14
RECOGNITION CIRCUIT DIGITS 15-21
RECOGNITION CIRCUIT DIGITS 22-28
RECOGNITION CIRCUIT DIGITS 29-35
DIODE MATRIX SEQUENCE SWITCH
1
2
3
4
5
N
IN
OUT
TO ADDITIONAL COMPARATORS
CARD-PROGRAMMED CONNECTIONS
TO COMPARATORS B THROUGH H
SECTION 2-COMPARATOR (A)
SECTION 3-LOGICAL COMPUTER



AN ELECTRONIC DEVICE
FOR HIGH SPEED
INFORMATION SEARCHING

A COMPARATOR unit is shown in Section 2 of the block diagram. The Comparator must accept successive groups of binary digits previously identified by the control unit and check each group against the criteria set up by the operator of the machine. The principle of selection is based on accepting each group if it matches the corresponding section of a criterion, and preparing the way for the acceptance of the next group. The next group must match the succeeding section of the same criterion. If it matches, a path is prepared for checking another succeeding group, and so on. If a group does not match, the acceptance chain is broken and the circuit will accept no more groups until after it has been reset to begin inspection of another sequence. If the entire criteria is matched, then a signal is sent to the Computer. This signal remains continuously until the end of the block of index entries relating to one document is reached. (Technically, it remains until a reset pulse is initiated by the succeeding block.)

As many comparators may be used as are needed, one comparator unit being required for each index entry to be found. The recognition circuits are arranged so that when the appropriate combination of pulses is present, all the associated vacuum tubes are cut off and a grid attached to a common plate load resistor is then raised above cut-off to provide an output pulse.

The LOGICAL COMPUTER or COMPUTER (Sec. 3 of the Block Diagram on page 83) is provided to ascertain if the proper combination of index entries has been located within a single block. The logical computation

to be performed is that of substituting values in an expression written in logical algebra. The values to be substituted indicate the presence or absence of the various criteria. For example;

$$J = A \cdot B \cdot (C + D + E + F) + G \cdot \overline{H}$$

This means J is true if either: A and B and one or more of the factors D, D, E, and F are true, or if G is true and H is simultaneously not true. In the sense in which this algebra is used here, the word "present" may be substituted for the word "true". In other words, if the desired criteria designated A to H are properly matched, then the corresponding document is desired, and hence the serial number is to be copied from the tape.

The circuit for performing this check is the electronic counterpart of a relay circuit. The relay circuit can be constructed according to the rules of "switching algebra", which is closely related to "logical algebra". Gate-tubes are used instead of relay contacts. Turning a gate tube "On" is the same circuit-wise as closing a relay contact. By properly controlling the gate tubes by connections indicating the presence or absence of the various criteria, a circuit can be made which will give an output only if the proper combination of criteria has been matched within a single block. When suitable criteria are met, a circuit is completed through the computer; a pulse emitted from the control unit passes through the computer and returns to the control unit, indicating that circuits are to be prepared to read the forthcoming serial number.

The RECORDER is shown in Section 4 of the block diagram on page 83. The task of the recorder is to accept the job number of the search and all the desired serial numbers selected from the tape, which numbers will be fed in at random intervals. There is no reliable device at present which will translate and print numerical information (in this case ten-digit decimal numbers) arriving at random intervals as short as 250 microseconds (which would be necessary if two serial numbers were to be recorded from two successive blocks, where the first block contained the probable minimum number of entries, j = 10). A buffer storage unit must necessarily be provided. Magnetic drums are suitable for storing the serial numbers as fast as they may be sent from the reader unit (up to 1 million pulses per second per channel), provided that the backlog of unprinted number does not exceed the capacity of the drum. Selectors for controlling the recording and reading on the drum must be provided, as well as a device which will decode the binary numbers into numbers more readily usable by human beings. Octal-base (base 8) numbers are strongly recommended for this, as the translation from binary to octal numbers is trivial, while the translation from binary to decimal numbers is very complex. A control mechanism must be provided to direct the activities of the printer and of the read-in and read-out selectors associated with the drum.

The JOB NUMBER INJECTOR (Section 5 of the block diagram) is provided principally as a convenience in identifying the sheets of serial numbers produced by the printer. In the event that one printer

is used for more than one searching machine, either simultaneously or alternately, then printed identification of the particular job that the machine is performing will avoid confusion in identifying the results with the original search request.

The job number is set up on switches (either manually or by punched card) and is transmitted through the recorder in a manner similar to the handling of a serial number. The job number is identified with a preliminary 0, whereas the serial numbers are each identified with a preliminary 1.

The Control Unit (Section 6 of the Block Diagram) is the "brain" which directs the activities of the rest of the machine in proper sequence. The control unit must perform the following functions:

1) Provide various reset pulses to the Comparator when the beginning of each block and of each index entry or serial occurs.

2) Provide gate pulses in proper sequence to the Comparator indicating the ordinal of the group of digits being read from the tape (that is, whether the group is the first, second, third, etc., of a sequence of groups representing either an index entry or a serial number).

3) Provide pulses to the Recorder to indicate each digit group of a serial or job number which is to be recorded.

4) Provide a means for starting the search, giving a start-pulse to Job Number Injector, and then starting the tape-drive mechanism.

5) Accept pulses from the Computer for use in creating the pulses to indicate to the Recorder when to record.

6) Accept synchronizing pulses from the Reader (Control Pulse Generator) to provide a time reference for the operations of the entire machine.

With the exception of the Recorder, the entire machine is designed to be built from standard circuits and components which have been built and tested (such as diode matrix switches, flip-flops and gates, pulse generators, gate generators, bus drivers, magnetic reading and recording heads and associated amplifiers); all of these circuits are used in the Whirlwind I electronic digital computer at M.I.T. The Recorder can use a commercially available magnetic drum, but the selectors and decoder will require special design; the proposed recorder is at best a makeshift arrangement until some more promising development is made in high-speed recording and printing techniques.

A word is perhaps in order here concerning the practical construction of a searching machine using electronic components. The entire machine is designed for flexibility; it should be so constructed that Comparator units may be added or changed where necessary to provide the needed capacity (in terms of the number of criteria which may be used at one time). In addition, the full utilization of the possibilities of miniaturized tubes and components, as well as printed circuits, should not be overlooked.

The advent in recent years of a new circuit component called the "transistor", which in many cases can take the place of a vacuum tube, promises to revolutionize the design of electronic circuits. No filament power is required, hence the power dissipation of a transistor is of the order of one-millionth that of a vacuum tube. Furthermore,

transistors are in general not subject to mechanical or electrical failure, thereby almost negating the problem concerned with circuit failures. The construction of transistor circuits for digital information processing is currently being done by the Bell Telephone Laboratories and other groups. Developments are for the most part still in the experimental stage, however,

D. The Effect of Storage Media and Input Equipment Upon Searching Speed.

The primary object in high-speed searching will be to process the stored index information at the greatest possible rate. Since the index information will be stored in the form of binary digits, the object can be said to be one of processing the greatest number of binary digits per second. An important factor influencing speed is the storage medium used. Of all the permanent storage media, magnetic tape seems to be the most suitable at the present time. Giant magnetic drums may be possible but do not appear practical at present.

There has not been time to make a thorough study of the possibilities of using 35mm microfilm as a permanent storage medium. The resolution of film, in a sense the number of discrete spots which can be recorded in a given area, is greater than that of magnetic tape. While 16-channel magnetic tape can accommodate about 1600 binary digits per inch of tape, film can accommodate at least 3500 binary digits per linear inch, perhaps more. If we can assume that film can be run as fast as tape, we would be able to obtain a scanning speed more than double that possible with magnetic tape. There may be some severe optical problems to be overcome in this application of film. There is

also the question of durability, which has been insufficiently investigated.

Assuming the choice of magnetic tape as the most suitable storage medium at present, a plan of digit arrangement will be formulated. Two different kinds of data will appear on the tape: index (subject) entries, which are to be scanned and compared, and serial numbers, some of which are to be transferred out of the machine. Both types of data are coded in binary numbers of approximately 30 digits each. There must be some additional indication to distinguish between the two so that the machine does not test serial numbers nor transfer out index entries. Because there is almost certainly to be more digits in each binary number than there will be channels on the tape, the binary numbers must be broken into digit groups which will be arranged in sequence.

These requirements (namely, provision for breaking numbers into successive digit groups) suggest adding an additional channel on the tape, which channel is to contain "control" information. The presence of a certain digit in the control channel will be an indication that the corresponding group of digits is the first group of a sequence of groups forming an index entry or a serial number. One additional distinction is necessary to differentiate between index entries and serial numbers. The logical choices are perhaps another control channel or the assignment of the first digit of each initial digit group. (This machine outlined in Section C of this chapter utilizes the second plan.)

If it is anticipated that the number of digits per index entry will vary widely, say from 15 to 35 binary digits, it would seem desirable to arrange the digit groups in such a manner that the number of groups can be varied and yet still be economical of digits. Although only a single digit might be needed to complete a sequence of groups, additional digits must be supplied to fill out an entire group simply because the succeeding binary number must begin with a full digit group. In order, then, not to have wasted too many digits filling out incomplete digit groups, the digit groups must be as short as possible. On the other hand, breaking up the number into a lot of small digit groups requires an increased number of control digits, since each group requires a control digit.

Sixteen-channel magnetic tape is, in the opinion of an expert in the field, about the maximum practical capacity at the present time for high-density pulse recording (80+ digits per inch per channel). By staggering the digits in various channels, the equivalent of an 8-channel tape running at twice the actual speed, or 5-channel tape running at three times actual speed can be obtained. On 8-effective - channel tape, one channel will be reserved as the control channel, allowing 7 channels for index information. This means that each binary number entry must be some multiple of 7 digits long. This appears to be a reasonably flexible arrangement.

While the maximum number of channels available on the tape influences to some extent the arrangement of the data on the tape, it does not affect the principles of design of the equipment. We have

discussed the design of the machine on the basis of using 8-effective-channel tape.

A word should be devoted here to the mechanical difficulties involved in operating magnetic tape at high speeds (400 or more inches per second). Because the tape mechanism necessarily has inertia, and because the tape is intended to be run at a speed of around 100 feet per second (1250 inches per second, to be exact) a rather complicated servomechanism is required to properly accelerate the tape to normal speed before reading can commence, to maintain proper tension in the tape while it is being run, and to bring the tape to a safe stop before the end of the reel has been reached.

In a word, then, handling magnetic tape at high speeds is a nasty mechanical problem which has been ignored while talking about the digital principles of design. It has been tacitly assumed that the mechanical difficulties can be overcome, since (according to Engineering Research Associates) tapes have been run successfully at 1600 inches per second.

E. Time Estimates of Operation

The limitation on the processing speed of the searching machine is at present the maximum tape speed. Electronic circuits can be constructed to handle 2,000,000 digit groups per second, higher by a factor of ten than is at present required for the searching machine. A greater rate of input might possible be achieved by running several tapes simultaneously. A very complex switching circuit and buffer storage would be needed, however, in order to keep the blocks intact

and read them into the machine at the proper time. The requirements on such a device would make the design unduly complex and it will not be considered further in this report.

The design figure for the digit-handling rate of magnetic tape which it seems reasonable to adopt at present is 10^5 pulses per second per channel. At a digit concentration of 80 digits per inch per channel, this requires a tape speed of 1250 inches per second, which is entirely within the realm of reason. On a sixteen-channel tape we can by recording alternately in the first eight and second eight channels (or in odd and even channels) obtain effectively 8 channels at 2×10^5 pulses per second per channel. This represents a total digit-processing rate of 1.6×10^6 digits per second. Hope for an increased digit processing rate lies either in being able to develop magnetic tape with greater than 16 channels or in methods of handling tape at higher speeds.

Assuming 5 groups of 7 digits (septads) per index entry, and 30 entries per document, the above tape speed will permit processing the index to 1,333 documents per second, or 4.66 million documents per hour.

F. Weaknesses and Capabilities of the Searching Machine.

As has already been pointed out by others, one of the chief weaknesses with any high-speed device is the difficulty of providing input information fast enough to keep the machine fully occupied, and of providing a means of recording the machine's output fast enough to keep the machine from having to slow down.

Also here should be mentioned something about the reliability of electronic equipment. It has been tacitly assumed during the designing that the reliability of electronic equipment is 100 percent perfect. This is most certainly not true in practice. The art of electronic handling of information by digital equipment is processing, however, to the point where we may hope to get almost perfect accuracy. Occasional machine errors are not nearly so critical in this machine as they would be in a computing device, where one wrong number is sufficient to invalidate an entire calculation. A mistake in a searching machine may mean that a single document will be missed, but in general this type of error will not invalidate the results of an entire search.

One of the obvious weaknesses of the electronic searching machine is that it provides no really useful output besides the serial numbers of the documents. Ideally, a machine should supply abstracts or perhaps even the entire original documents themselves.

An additional disadvantage is that working with a continuous medium such as would be necessary in order to achieve the high speed, it would be impossible to handle the results of the search as discrete units of information. In other words, the results of a search cannot be perused visually and a second search conducted on the results of the first (without scanning the entire file the second time).

BIBLIOGRAPHY

Bell Telephone Laboratories. Notes on the Design of Switching Circuits For Automatic Control. New York: Bell Telephone Laboratories, Inc., 1950. Mimeographed. (To be published in book form.) (X-63883)

Engineering Research Associates. High-Speed Computing Devices. New York: McGraw-Hill, 1950.

Engineering Research Associates. Magnetic Drum Storage For Digital Information Processing Systems. St. Paul: Engineering Research Associates, Inc., 1949. 19 pages.

Harvard University Computation Laboratory. Synthesis of Electronic Computing and Control Circuits. Cambridge, Mass.: Harvard University Press, 1951.

See also the Bibliography to Appendix 2.

PART THREE

CONCLUSIONS AND RECOMMENDATIONS

CHAPTER 8

EVALUATION OF HIGH-SPEED SEARCHING EQUIPMENT

A. The Costs of An Electronic Searching System

It is virtually impossible to estimate accurately the costs of a searching system based on an electronic digital machine. Any estimates will be at best only rough guesses based on previous experience with similar equipment and procedures.

An electronic digital machine for searching might be comparable in construction and cost to a modern electronic digital computer, for which engineering and development of a prototype machine is in the neighborhood of one million dollars. Since an electronic searching machine makes use of many electronic features which have already been successfully worked out for computers, it is possible that the engineering task might be to some extent simplified. The fruition of promising new developments such as transistors and magnetic-core storage quite possibly may revolutionize construction and maintenance costs of digital machines. Furthermore, copies of a prototype machine can be made for substantially less cost than the original.

The cost of preparing a coded index for documents numbering in the millions will be comparable to that of the prototype searching machine itself. The analyzing, indexing, and coding of each document will be the major preparation cost, inasmuch as it will be in dollars per document, while the storage medium itself will most probably cost

a few cents per document. The medium may be subject to wear and tear and require periodic replacement, but it can be duplicated for essentially the cost of the storage medium involved.

A complete search of the index for 50 million documents at 4.66 million per hour (the design figure for the machine of Chapter 7) will take 10.7 hours. The cost of such a search, requiring the services of a costly machine for an entire day, cannot be assigned a meaningful figure. One can only guess that it will be some astonishing number. Before dismissing the searching cost as being unreasonable, however, it is well to recall to mind that operation costs of present day digital computers or of television networks seem to be comparable, and yet are no longer considered impossibly high. For industrial and government work, the costs of high-speed searching systems may well prove to be less expensive in the long run than the present slow and incomplete, and sometimes inaccurate, searches. Furthermore, when high-speed searching systems are put into actual service, an increase in searching speed through new technical developments and a decrease in cost per search through widening use of the facilities will almost certainly result.

B. The Cost of Punched-Card Systems

It is very difficult to ascertain some comparative costs for the use of punched card systems with respect to electronic searching systems because of the fact that punched-card systems are generally inadequate for performing the same tasks that can be performed by an electronic searching machine as outlined in the previous chapter. Assuming

for the moment that full use could be made of the 960 punching positions on an IBM card, we might be able to index an entire document (with a coordinate-array index) on a single card. The IBM sorting equipment will sort at the rate of approximately 600 cards per minute, or 36,000 cards an hour. The searching time for 50,000,000 documents would be 1390 hours, or approximately 126 days, (four months) based on eleven hours of operation a day. The rental cost of an IBM card-sorting machine is approximately $60.00 per month, hence the machine cost alone for the entire search would be approximately 126/24 x 60 or about $315.00.

Of course, the IBM card-sorting equipment is not equipped to perform the type of search that we have previously outlined in Chapter 5, but the figures give some idea of the orders of magnitude involved in using punched-card equipment.

In using a system involving hand-sorted punched cards, it is almost impossible to make a suitable comparison in terms of scanning a large index.

C. Desirability of Electronic Versus Punched-Card Systems

If the searching system is to be designed for a collection of more than one million documents it is of the utmost financial importance that the maximum utility be made of the tape. This requires that maximum possible searching speeds be used. The additional cost incurred in making the searching machines perform high-speed operations will be more than saved in returns resulting from more searches per unit time. To put it a little more forcefully, it would be economically

unsound to run a tape valued in millions of dollars at a speed less than that which would keep the cost of a search down to a reasonable figure.

Where a smaller collection is involved, say, 50,000 to 100,000 documents, the cost of the machine rather than the cost of preparing the index tapes becomes the significant factor. In this case it may not be economically desirable to insist on the highest possible searching speed. It will probably turn out to be more convenient and less expensive to adopt a system of hand- or machine-sorted punched cards instead. It is not until the number of documents which are indexed becomes very large that high-speed methods of searching become necessary or feasible.

The suitability of punched card systems is especially significant in applications which require a fairly restricted amount of indexing. An example of this might be a medical application in which it is desired to make a file of all the diseases with their attendant combinations of symptoms. If this file were limited to 100,000 cards, the system could be made to operate quite suitably and efficiently with punched cards using superposed coding (Zatocards or Keysort cards).

In drawing comparisons of various systems, the comparing of costs of indexing and coding for a library searching system is hardly significant since virtually the same amount of effort will be required to index and code the documents, regardless of the type of searching system under consideration.

D. The Practical and Ideal Features of Searching Systems

The "ideal" system specifications for a searching system might be something like the following:

1) The scheme of coding to be relatively simple, capable of being handled by unskilled clerical help once the index terms were assigned to a document.

2) The indexing system to be comprehensive enough to provide access to documents through generic terms.

3) The machine to store the index information in permanent form.

4) The machine to search through the entire index in a reasonable time.

5) The machine to search for logical combinations of concepts.

6) The machine to present the results of a search in usable and convenient form.

7) The machine operator to require only a small amount of specialized knowledge.

8) The machine cost not to be prohibitive.

9) The indexing system capable of handling 50 million to 100 million documents.

One of the ultimate requirements of a suitable searching system would be to supply copies of abstracts and/or of the original documents themselves. The searching machines previously discussed have been capable of simply scanning the coded index and determining

which documents are related to the search; the actual obtaining of the information contained in the documents has been overlooked. In the design of a practical system, a compromise must be reached between high speed of searching and the convenience of having printed documents as the result of a search. It is unreasonable to try to make a high-speed machine which has to carry along the original document or an abstract thereof along with the indexing information. The abstract and document should be stored in some other fashion. In a future machine, then, we would expect that additional machine features would select and copy the abstract and/or document from permanent storage other than the medium which carries the index.

Where the speed of punch-card systems is suitable for the searching system desired, punched cards can be used to distinct advantage. They provide first of all a physical copy of the index information or the abstract, just as soon as the scanning operation is completed. The second advantage is that on a first search a large group of cards may be selected and visually inspected; this group may then be put back into the machine and resorted in any way that appears desirable.

It is perhaps enlightening to outline an entire searching system using punched cards. This system might use cards bearing both the index information and an abstract; possibly the document itself would be included, through the process of microprinting or microphotography The cards should be of cardboard stock similar to IBM cards and of a

standard size, somewhere between 3 x 5 and 5 x 8 inches. One long side of the card would be used for the storage of index information either by combinations of light and opaque spots, or by magnetized portions of a magnetic material such as the present magnetic tape. (There are other methods.)

If the cards are in random order, the practical size of the file is on the order of 500,000 cards. For collections in this range, the system has definite advantages over other searching systems.

1) Material selected on a first sort can be visually inspected. If it is not quite what is wanted, this group alone can be re-sorted. In a search not readily definable, a search directed toward generic topics may be made to obtain a rough idea of what is available, and the cards thus found may be sorted again according to probably useful headings.

2) In a group of desired cards the machine can re-sort the cards in many useful ways depending on the needs of the user. For example, a group of cards dealing with plant life might be sorted first according to families, then according to locale, or habitat, then according to uses of plant products, etc.

3) If the original document is not microprinted on the card, the need for a search in a second file to obtain the original document is reduced to a minimum, for most of the decisions as the usefulness of each document can be made from the abstracted information.

4) The material (both the index information and the abstract) on the cards is readily available for review and editing.

5) Cards lend themselves to easy reproduction for private sub-files, even to mass production for distribution to other library files.

6) If not enough information can be gotten on one card, it is readily possible to arrange for the use of trailer cards.

In the preparation of an magnetic tape index for a large document file, it would seem advisable to make the original file on punched cards. The tapes would then be prepared automatically from the punched-card file. The punched-card master file would have the distinct advantages of being easy to check entries for accuracy, being easy to make revisions where necessary (adding or deleting headings or eliminating entire items) and durable enough to prepare a number of tapes.

Another possibility for a searching system would be to combine the best features of magnetic tape and punched cards. A suitable system might be arranged to have a major search performed by an electronic device to locate the document serial numbers. Another machine might then look up the corresponding card in the master file, which card could then be visually inspected and handled on a card-sorting machine.

Some of the value of a high-speed searching system, or any searching system for that matter, will depend to an extent on the operator. There will occasionally appear questions which seem to be beyond the capacity of the searching equipment. If the criteria were, among other things, any one of 48 chemical compounds, the proper programming to keep the search within the capacity of the machine (say

10 criteria) would not be to search for any one of the 48 compounds but rather to search for one or more of the common features of the group of compounds. This will result perhaps in a selection of more material than is wanted, but will enable the machine to do a major amount of the preliminary screening. The fact that one asks for any one of 48 items almost insures that the 48 items have characteristics in common.

E. The Intellectual Consequences of a High-Speed Searching System

It has been suggested that one of the outstanding results of a practical high-speed searching system will be to assist in the production of a greater volume of more profound research. It is felt by some people that one thing which is holding back research at the moment is the inability to make use of related reports of previous work which have been written and filed away. Making this stored information more readily available — in a sense, doing part of the job of correlating known information — should be a great assistance toward this end.

Information searching and research are likely to form a feedback network: if high-speed searching improves the planning of research, there will be a tendency for more effective research to be done and more documents to be written, and, in turn, more documents will be available upon which to base further research. The net result may possibly be a significant increase in the rate of scientific progress. (We will studiously avoid, however, any philosophical discussions concerning what constitutes scientific progress.)

F. The Evolutionary Aspect of High-Speed Searching Systems

It is perhaps interesting to conjecture how, if at all, high-speed searching systems may come to be a reality. The author believes that many people acquainted with the present problems of searching will not dispute the fact that high-speed searching systems are much needed, and will probably be adopted. The establishment of a radically new system for doing something we have been doing for years presents, however, quite a few practical problems. In the first place, no one really wants to take the responsibility for designing and building the first system because there is financially so much at stake. When commercially-made searching machines are available, it may take some time for them to come into widespread use, just as was the case of radio programs transmitted by frequency modulation. It is difficult to imagine that the task of building the initial system will be undertaken by an individual or corporation. It will most probably be a job to be promoted and financed by the United States Government.

There are a multitude of possibilities in applying high-speed searching systems to government work. Among those who most urgently need such systems are the Patent Office, the Library of Congress, and the military intelligence. It seems almost inevitable that the government will eventually be under the necessity of developing the systems to handle its information problems.

It may be said that the costs of setting up a comprehensive high-speed searching system are too fantastic. As time goes by, however, it will become more and more evident that, unless we are content to be without guidance in a sea of knowledge, that the cost of not building systems for searching will be even more fantastic. It is not too much to hope that a positive attitude toward the development of high-speed searching devices may develop.

BIBLIOGRAPHY

Mooers, Calvin N. "Making Information Retrieval Pay." Boston: Zator Company, 1950. (Zator Technical Bulletin 55), 13 pages.

CHAPTER 9

SUGGESTIONS FOR FURTHER WORK

A. Aspects Needing Further Development

The chief bottleneck in making full use of the capacity of the high-speed searching machine is the input and output equipment. First of all, there is a limitation on the speed at which the magnetic tape carrying the stored information can be run. Until some way can be found to increase this speed, it remains the factor limiting the rate of searching.

There are perhaps two ways which ought to be examined in regard to finding ways to process more digits per second. One method is to find a permanent storage medium which can be run faster than a tape and which will hold approximately the same concentration of binary digits as the tape, or else a storage medium which will hold a greater concentration of digits. One of the most promising methods would be the use of microfilm, which presumably could be run as fast as magnetic tape, and which might hold a considerably higher concentration of digits.

A second aspect which needs some attention is the output or recording device. At the present time we must provide buffer storage and do some fancy switching in order to accommodate the spurts of serial numbers which the machine may turn out. The buffer storage is necessitated by the fact that there is no method of printing at

the rate which would be necessary. It would be worthwhile to investigate a new development in high-speed printing announced by Daniels of Engineering Research Associates. At the Association for Computing Machinery Conference in Washington, D.C., in September, 1950, it was claimed that it is possible to record 5,000 characters per second. No further details have been found, however.

B. Description of Equipment With Possible Applications to The Field of Information Searching.

Nearly all the material in this section has been gained from personal conversations with engineers. Little of the material is conveniently available in published form as yet.

This section is essentially an enumeration of equipment which might possible have some application. Under the heading of permanent and temporary storage there are eight items to be mentioned.

1) The first is the electrostatic storage tube, developed by the Whirlwind Project at M.I.T., which stores charges representing single binary digits on the surface of a 16 x 16 array of beryllium squares. The approximate access time to a given square (a certain "address") is at present about 50 microseconds. It is hoped that the access time will some day be reduced to the order of 6 microseconds, and the storage to 1024 addresses.

2) The Flexowriter is a modified IBM electric typewriter which punches paper using a six-hole code similar to the teletypewriter code. The maximum rate of punching is approximately 8 characters a second. Each character is represented by one of the 64 combinations possible with 6 binary digits.

3) The Engineering Research Associates Tape Reader is designed for reading punched Flexowriter tape at the rate of 140 lines per second (= 140 characters per second).

4) The Eastman Kodak Film Unit was designed for handling 35mm. film. The reading and recording speed is approximately 2000 lines per second with up to 25 channels. The film speed is approximately 20 inches per second. The time required to stop the film unit is 10 milliseconds.

5) Magnetic Tape Recording. Standard coated tape at the present time has a recording density of 100 pulses per inch and 5 channels on tape $\frac{1}{4}$ inch wide. Engineering Research Associates reports in "High Speed Computing Devices" that 100,000 digits per second per channel is a reasonable recording and reading rate. An engineer in the magnetic recording field says that a pulse repetition frequency of 1,000,000 cycles per second is not too optimistic. Vicalloy tape made by the Arnold Engineering Company is presently being operated at 400 inches per second with negligible tape wear. It is reportedly possible to use Vicalloy with a recording density of 200 digits per inch at a speed of 400" per second, giving a pulse repetition rate of 80 kilocycles. A 14" reel of tape is approximately 2500 feet. Magnetic heads are currently available (Armour) which can be stacked side by side twelve heads to the inch. Another manufacturer is developing heads which will stack 40 to the inch.

6) Magnetic Drum Recording. According to an expert, recording 1,000,000 pulses per second is now possible. Drum speeds vary from

3600 to 10,000 rpm. Engineering Research Associates has produced a drum which has 2048 "addresses" and 208 channels. The average access time (that is, one-half a revolution-time) is 8 milliseconds, and the recording and reading rate is approximately 1.25×10^5 pulses per second per channel.

7) Electrochemical matrix experiments have been reported in which tantalum wires laterally and gold wires horizontally with a sulfate solution surrounding each cross-point can be used for temporary or semi-permanent storage. Each crosspoint acts as an individual storage cell.

8) Ferromagnetic memory is a three-dimensional array of ferromagnetic cores, each core a bistable element. Any one core can be selectively magnetized by selection of appropriate coordinate wires. Work on this device is currently being done by William Papian at Project Whirlwind.

One switching device in addition to the conventional gate-tubes and crystal-diode matrices (see Appendix 2) is an electrostatically controlled multi-position switch. This is a concept rather than a piece of finished equipment. The principle is to use an electrostatically-deflected beam to select coordinate addresses, which will in turn open associated gate-tubes controlling access to the coordinate wires in an electrochemical or ferromagnetic storage matrix. This device can conceivably be of less complexity than the whiffle-tree method of gate-tube switching.

Among the output devices are:

1) the Flexowriter mentioned above.

2) ADCAP, a device built by William Santelmann of M.I.T. which prints 100 decimal digits per second on Teledeltos tape. The character matrix is 5 x 3 units. The inventor says that the recording of 500 decimal digits per second is possible.

3) Control Instrument Company printer. This printer uses printing pins arranged in a character matrix 6 units by 4. The cam actuation mechanism is similar to the IBM electric typewriter. This device prints 48 characters per line at 1000 lines per minute from an input of punched-card data. The equipment is now being built and tested by the Control Instrument Company.

4) The ERA (Engineering Research Associates) Super-Speed Printer. This was mentioned earlier in this chapter; it prints 5000 characters per second. No further details are available.

5) Alfax paper, made by Alden Products. Simultaneous printing by the conduction of an electric current is possible over any area of the tape. Supplying the necessary moisture, however, presents a problem.

6) Times Hecto Paper is a paper similar to Western Union Teledeltos tape, but a hectograph master is produced by the recording process.

C. Companies Interested in Machine Methods of Information Handling.

The following is a list of organizations in the United States which have an apparent interest in machine searching.

Eckert-Mauchly Computer Corp.
3747 Ridge Avenue, Philadelphia 32, Pa.

Potter Instrument Company
135-56 Roosevelt Avenue, Flushing, New York

Barber-Colman Company
Rockford, Illinois

Engineering Research Associates
1902 Minnehaha Avenue, St. Paul W4, Minnesota

Raytheon Manufacturing Company
Waltham, Massachusetts

International Business Machines
Poughkeepsie, New York

Control Instrument Company
Brooklyn, New York

National Bureau of Standards
Washington 25, D.C.

Zator Company
79 Milk Street, Boston, Mass.

Metropolitan Life Insurance Company
1 Madison Avenue, New York 10, New York

John Hancock Life Insurance Company
200 Berkeley Street, Boston 17, Mass.

Division of Chemical Literature
c/o American Chemical Society, 1155 Sixteenth St., N.W.
Washington 6, D.C.

D. A Note About Translating Machines.

The problem encountered in translation of language is partly that of the meaning of individual words and partly of correlation of groups of words with ideas. The part of translation dependent upon merely looking up the meaning of words in a dictionary is a mechanical one which can be performed with a selecting device similar to a library searching machine. The task of correlating groups of words and revising them idiomatically is a job for a machine which

can make use of probability data and which can make repeated trials. This suggests machines such as the digital searching machine, and the "synaptic" telephone system being developed in Holland by the Phillips concern.

It seems apparent, at this point anyhow, that the translator will eventually be composed of several types of devices: a storage device, a device for scanning the storage, a device for correlating, a device which can make repeated trials and perform various operations with the same piece of information, and a device which can remember and make choices based upon known probabilities.

BIBLIOGRAPHY

Dodd, S. H., et al., "Electrostatic Storage Tube," Electrical Engineering, November 1950.

Eastman Kodak Company. "Kodak Digital Reader-Recorder, Model A, Rochester, Eastman Kodak Company, June 1950.

Engineering Research Associates. High-Speed Computing Devices. New York: McGraw-Hill, 1950. (Magnetic tape recording, p. 439.)

APPENDICES

APPENDIX 1

DERIVATION OF SUPERPOSED CODING FORMULAS

The design formulas applying to superposed coding systems have been somewhat simplified from the original expressions derived by Calvin Mooers. No consideration has been taken of the frequency of occurrence of the various index terms. (In a coordinate index, the higher-order generic terms may be expected to occur more frequently than the lower-order ones, and these should be given shorter coding patterns, that is, a smaller number of tallies per index word.) The conditions for the validity of the noise formula may not be stringently met; nowhere in its derivation, however, could be found any statement which would invalidate it for the condition that various terms have different frequencies of occurrence.

Let v = the number of terms in the indexing vocabulary

g = the number of different grammatical relationships (if no relationships are expressed, this is 1)

t = the number of terms applicable to a single entry

x = the number of tallies (1-digits) required for each index term

n = the total number of tally places (for 0- and 1-digits both)

s = the number of terms used in specifying the selection

Now, t terms require x tallies each **or** a total of tx. The greatest efficiency in the use of digits occurs if approximately half the total tally places are occupied by tallies (1-digits); in other words, if the total number of tallies is equal to one-half the total possible number of tallies. tx tallies distributed randomly in a

n-place field will take less than tx tally places because some tallies will fall in the same tally-place; actually .69n tallies will require .5 of the n tally-places, hence tx = .69n. The total number of combinations of n tally-places filled x at a time is equal to or greater than the total vocabulary multiplied by the number of grammatical forms each vocabulary word can take (to express various grammatical relationships).

Hence, in mathematical notation,

$$C_x^n \geq vg$$

But, substituting for n using tx = .69n or n = 1.45 tx,

$$C_x^n = \frac{(1.45tx)\,!}{x!\ (1.45tx - x)\,!} \geq vg$$

Mooers has demonstrated that, for the conditions assumed above, the noise, or percentage of selections unrelated to the search, is given by

$$N \leq (\tfrac{1}{2})^{sx}$$

For the second case described in Chapter 4, where there are a fields of n tally positions, the equations may be adapted to give a solution on the basis of a single field, thus:

By substituting $\frac{t}{a}$ for t in the equation above, we have:

$$C_x^n = \frac{(1.45\frac{t}{a}x)\,!}{x!\ (1.45\ \frac{t}{a}x - x)\,!} \geq vg$$

The total number of tally positions (the total size of a fields) is r = na. The noise formula holds as before.

BIBLIOGRAPHY

Mooers, Calvin N. "Zatocoding for Punched Cards." Boston, Zator Company, 1950. 19 pages (Zator Technical Bulletin 30).

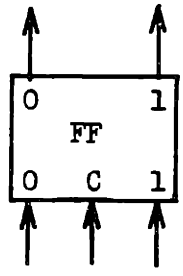
Mooers, Calvin N. "Application of Random Codes to the Gathering of Statistical Information." Boston, Zator Company, 1949, 28 pages, (Zator Technical Bulletin 31).

APPENDIX 2

DESCRIPTION OF MAJOR ELECTRONIC COMPUTER COMPONENTS

A. Flip-flops

A "flip-flop" is the nickname for a bi-stable circuit, an electronic circuit which will remain indefinitely in one of two stable states (tube A or tube B conducting) depending upon which input connection last received an electric pulse. The conventional indication for a flip-flop circuit is as follows:



output leads

input leads

An input on the "0" lead switches the flip-flop to provide a continuous output on the corresponding zero lead. A pulse on the "1" lead switches the flip-flop to provide a continuous output voltage on the corresponding 1-lead. A pulse on the "C" lead switches the flip-flop from its present state to the other (or "complementary") state.

It is possible to build a bi-stable circuit with a single tube such as the Kliotron, which is capable of changing its state up to 100,000 times a second. With two tubes, such as 6AS6's or 6AK5's, a more reliable circuit operating up to 10 megacycles (10 million times per second) can be constructed.

B. Gate Tube

A gate tube is to an electronic circuit as a garden gate is to a human being. When the gate is held open, an electronic pulse

can pass through the gate tube; when the gate is closed the electronic pulse is blocked. Because the gate tube is opened by another electronic pulse, arriving on another lead, the gate tube can be said to give an output when there is a coincidence of pulses on two different input to the gate tube. In electronic language, a gate tube will emit a pulse if and only if the control and suppressor grids are simultaneously above cut-off voltages. Gate circuits can be constructed with tubes such as the 6AS6 and 7AK7. The time delay encountered by a pulse traveling through a gate tube can be shorter than .04 microseconds.

C. Crystal-diode matrix

A crystal-diode matrix is a simple switching device made principally of germanium crystals, which can translate a combination of input pulses (arriving simultaneously on different leads) into another combination of pulses.

BIBLIOGRAPHY

Adams, Charles W. "Notes on the Logical Design of Digital Computers and On Special Coding Techniques." Cambridge, Mass.: Massachusetts Institute of Technology, 1951. Hectographed.

Huskey, H. D. "Terminology and Block Diagrams for Use in Connection with Automatic Digital Computing Machines." Washington: Bureau of Standards, 1950.

Brown, DAVID ~~Douglas~~ R. "Rectifier Networks for Multi-Position Switching." Institute of Radio Engineers Proceedings, February 1949, p. 139.

Verzuh, Frank M. "Reference Notes and Experiments for Course 6.538T, Electronic Computation Laboratory." Cambridge, Mass.: Massachusetts Institute of Technology, 1950, hectographed.

APPENDIX 3

GUIDE TO CODING FOR THE WHIRLWIND COMPUTER

A. Short Guide to Coding; List of Orders

Included here is a list of simplified instructions concerning the"programming" for the Whirlwind electronic computer. The method of writing a program is illustrated by a short sample program.

A SHORT GUIDE TO CODING

Using the Whirlwind I Code of October 1949

Electronic Computer Division
Servomechanisms Laboratory

Massachusetts Institute of Technology
Cambridge, Massachusetts

A SHORT GUIDE TO CODING

(using the Whirlwind I code of October 1949)

COMPUTER PROGRAMS

Program. A program is a sequence of actions by which a computer handles a problem. The process of determining the sequence of actions is known as programming.

Flow diagrams. A flow diagram is a series of statements of what the computer has to do at various stages in a program. Lines of flow indicate how the computer passes from one stage of the program to another.

Coded program. Programs and flow diagrams are largely independent of computer characteristics, but instructions for a computer must be expressed in terms of a code. A set of instructions that will enable a computer to execute a program is called a coded program, and the process of preparing a coded program is known as coding.

Orders and operations. Individual coded instructions are known as orders and call for specific operations such as multiply, add, shift, etc.

The computer code. The computer code described here is that of Whirlwind I, an experimental computer using binary digits, single-address order code, parallel operation, and electrostatic storage. It is expected that computers of this type will ultimately achieve an average speed of 50,000 operations per second.

COMPUTER COMPONENTS

Registers and words. A register has 16 digit positions each able to store a one or a zero. A word is a set of 16 digits that may be stored in a register. A word can represent an order or a number.

Arithmetic element. Arithmetic operations take place in the arithmetic element, whose main components are three flip-flop registers, the A-register, the accumulator, and the B-register (AR, AC, BR). The 16 digit positions of AR starting from the left are denoted by AR 0, AR 1, . . ., AR 15. Similarly for AC, BR. Words enter AC through AR; BR is an extension of AC.

Storage. The term "register" by itself refers to the main electrostatic storage, which consists of 2^{11} or 2048 registers, each of which is identified by an address. These addresses are 11-digit binary numbers from 0 to 2047. The computer identifies a register by its address.

Input-output. All information entering or leaving the computer is temporarily stored in the input-output register (IOR). The computer regulates the flow of information between the internal storage and IOR, and also calls for any necessary manipulation of external units. The descriptive names of the input-output orders were chosen for photographic film reader-recorder units, but the orders are applicable to other types of external equipment.

Control element. The control element controls the sequence of computer operations and their execution. Instructions are obtained from storage in the form of individual orders, each of which is represented by a single word.

Inter-connections. The four main elements (storage, control, arithmetic, and input-output) are connected by a parallel communications system, known as the bus.

REPRESENTATION OF ORDERS

Operation section. When a word is used to represent an order the first (left-hand) 5 digits, or operation section, specify a particular operation in accordance with the order code.

Address section. The remaining 11 digits, or address section, are interpreted as a number with the binary point at the right-hand end. In the majority of orders this number is the address of the register whose contents will be used in the operation. In orders sl, sr, the number specifies the extent of a shift; in rf, rb, the number specifies an external unit; in ri, rs, the address section is not used.

Example. The order ca x has the effect of clearing AC (making all the digits zero) and then putting into AC the word that is in the register whose address is x. If q is a quantity in some register, the order needed to put q in AC is not ca q but ca x, where x is the address of the register that contains q.

REPRESENTATION OF NUMBERS

Single-word representations. When a word is used to represent a number the first digit indicates the sign and the remaining 15 are numerical digits. For a positive number the sign digit is zero, and the 15 numerical digits with a binary point at their left specify the magnitude of the number. The negative -y of a positive number y is represented by complementing all the digits, including the sign digit, that would represent y. (The complement is formed by replacing every zero by a one and every one by a zero.) In this way a word can represent any multiple of 2^{-15} from $2^{-15} - 1$ to $1 - 2^{-15}$. Neither +1 nor -1 can be represented by a single word. Zero has two representations, either 16 zeros or 16 ones, which are called +0 and -0 respectively.

Overflow — increase of range and accuracy. With single-word representation the range is limited to numbers between $2^{-15} - 1$ and $1 - 2^{-15}$. Programs must be so planned that arithmetic operations will not cause an overflow beyond this range. The range may be extended by using a scale factor, which must be separately stored. Accuracy can be increased by using two words to represent a 30-digit number.

COMPUTER PROCEDURE

Sequence of operations. After the execution of an order the program counter in the control element holds the address of the register from which the next order is to be taken. Control calls for this order and carries out the specified operation. If the order is not sp or cp(-) the address in the program counter then increases by one so that the next order is taken from the next consecutive register. The sp and cp(-) orders permit a change in this sequential procedure.

Transfers. A transfer of a digit from one digit position to another affects only the latter digit position, whose previous content is lost.

Negative zero. The subtraction of equal numbers produces a negative zero in AC, except when AC contains +0, and -0 is subtracted from it.

Manipulation of orders. Words representing orders may be handled in the arithmetic element as numbers.

Procedure in the arithmetic element. The execution of an addition includes the process of adding in carries; this process treats all 16 digits as if they were numerical digits, a carry from AC 0 being added into AC 15. A subtraction is executed by adding the complement. Multiplication, division, shifting and round-off are all executed with positive numbers, complementing being performed before and after the process when necessary. For round-off the digit in BR 0 is added into AC 15.

NOTATION FOR CODING

Addresses. A coded program requires certain registers to be used for specified purposes. The addresses of these registers must be chosen before the program can be put into a computer, but for study purposes this final choice is unnecessary, and the addresses can be indicated by a system of symbols or index numbers.

Writing a coded program. Registers from which control obtains orders may be called action registers, and should be listed separately from registers containing other information, which may be called data registers. A coded program is written out in two columns; the first contains the index number of each action or data register, and the second column indicates the word that is initially stored in that register. In many cases part or all of a word may be immaterial because the contents of the register in question will be changed during the course of the program. This state of affairs is indicated by two dashes, for example, ca --.

The abbreviations RC, CR. Abbreviations used in referring to the register that contains a certain word or to the word in a certain register are

RC . . . = (Address of) Register Containing . . .
CR . . . = Contents of Register (whose address is) . . .

The symbol ri x. When an address forms part of an order it is represented by the last 11 digits of a word whose first 5 digits specify an operation. An address x that is not part of an order is represented by the last 11 digits of a word whose first 5 digits are zero, which is equivalent to specifying the operation ri. Thus the word for an unattached address x may be written ri x. It could also be written $x \times 2^{-15}$.

THE ORDER CODE

AC = Accumulator AR = A-Register BR = B-Register

x is the address of a storage register; n is a positive integer; k designates an external unit

Order	Operation			Function
	Name	Code		
		Decimal	Binary	
ri --	read initially	0	00000	Take words from external unit until internal storage is full.
rs --	remote unit stop	1	00001	Stop external unit.
rf k	run forward	2	00010	Prepare to use external unit k in forward direction.
rb k	run backward	3	00011	Prepare to use external unit k in backward direction.
rd x	read	4	00100	Transfer to register x a word supplied by external unit.
rc x	record	5	00101	Arrange for transfer of contents of register x to external unit.
ts x	transfer to storage	8	01000	Transfer contents of AC to register x.
td x	transfer digits	9	01001	Transfer last 11 digits from AC to last 11 digit positions of register x.
ta x	transfer address	10	01010	Transfer last 11 digits from AR to last 11 digit positions of register x.
cp(-)x	conditional program	14	01110	If number in AC is negative, proceed as in sp; if number is positive disregard the cp(-) order, but clear the AR.
sp x	subprogram	15	01111	Take next order from register x. If the sp order was at address y, store y + 1 in last 11 digit positions of AR.
ca x	clear and add	16	10000	Clear AC and BR, then put contents of register x into AC. If necessary, add in carry from previous sa addition.
cs x	clear and subtract	17	10001	Clear AC and BR, then put complement of contents of register x into AC. If necessary, add in carry from previous sa addition.
ad x	add	18	10010	Add contents of register x to contents of AC, storing result in AC.
su x	subtract	19	10011	Subtract contents of register x from contents of AC, storing result in AC.
cm x	clear and add magnitude	20	10100	Clear AC and BR, then put positive magnitude of contents of register x into AC. If necessary add in carry from previous sa addition.
sa x	special add	21	10101	Add contents of register x to contents of AC, storing result in AC and retaining any overflow for next ca, cs, or cm order. Only orders 1 through 15 may be used between the sa order and ca, cs, or cm orders for which the sa is a preparation.
ao x	add one	22	10110	Add the number 1×2^{-15} to the contents of register x. Store result in AC and in register x.
mf x	multiply and round off	24	11000	Multiply contents of register x by contents of AC; round off result to 15 numerical digits and store in AC. Clear BR.
mh x	multiply and hold	25	11001	Multiply contents of register x by contents of AC and retain the full product in AC and the first 15 digit positions of BR, the last digit position of BR being cleared.
dv x	divide	26	11010	Divide contents of AC by contents of register x, leaving 16 numerical digits of the quotient in BR and ±0 in AC according to sign of the quotient. (The order sl 15 following the dv order will round off the quotient to 15 numerical digits and store it in AC.)
sl n	shift left	27	11011	Multiply the number represented by the contents of AC and BR by 2^n. Round off the result to 15 numerical digits and store it in AC. Disregard overflow caused by the multiplication, but not that caused by round-off. Clear BR.
sr n	shift right	28	11100	Multiply the number represented by the contents of AC and BR by 2^{-n}. Round off the result to 15 numerical digits and store it in AC. Clear BR.
sf x	scale factor	29	11101	Multiply the number represented by the contents of AC and BR by 2 sufficiently often to make the positive magnitude of the product equal to or greater than 1/2. Leave the final product in AC and BR. Store the number of multiplications as last 11 digits of register x, the first 5 digits being undisturbed.

NOTES ON THE ORDER CODE

Effect of operations. The functions of the various orders are described above. It is to be assumed that AR, AC, BR, and the register whose address is x are undisturbed unless the contrary is stated.

AR. AR is primarily a buffer register for passing words into AC. After orders ca x, cs x, ad x, su x, sa x, and ao x it contains the number originally contained in register x. After orders cm x, mr x, mh x, and dv x it contains the magnitude of the contents of x. The effect of sp x and cp(-)x is stated above. No other order changes the contents of AR.

BR. A number stored in BR always appears as a positive magnitude, the sign of the number being assumed to be that indicated by the sign digit in AC. This convention has no effect on the logical result of the operations involving BR except that when BR contains a number that will be used later it is necessary to retain the appropriate sign digit.

Alarms. If the result of an arithmetic operation exceeds the register capacity (i.e., if overflow occurs), a suitable alarm is given except as mentioned in connection with orders sa x and sl n.

Shift orders. A multiplication overflow in sl is lost without giving an alarm, but an overflow from round-off gives an alarm. Orders sr 0 and sl 0 only cause round-off, an alarm being given if an overflow occurs. The integer n is treated modulo 32, i.e., sl 32 = sl 0, sl 33 = sl 1, etc.

Scale factors. If all the digits in BR are zero and AC contains ±0, the order sf x leaves AC and BR undisturbed and stores the number 33 in the last 11 digit positions of register x.

Division. Let u and v be the numbers in AC and register x when the order dv x is used. If $|u| < |v|$ the correct quotient is obtained and no overflow can arise. If $|u| > |v|$ overflow occurs and gives an alarm. If $u = v \neq 0$ the dv order leaves 16 ones in BR and round-off in a subsequent sl 15 would cause overflow and give an alarm. If $u = v = 0$ a zero quotient is obtained.

REVISIONS IN THE WHIRLWIND I ORDER CODE

October, 1949 through May, 1951

Order	Operation: Name	Operation: Code Dec.	Operation: Code Binary	Function
A. Permanent addition to the Code:				
ck x	check	11	01011	Stop the computer and ring an alarm if contents of register x is not identical with contents of AC; otherwise proceed to next order.
B. Minor changes not affecting previous functions of certain orders:				
sa x	special add	21	10101	Add contents of register x to contents of AC, storing result in AC and retaining any overflow for the next ca, cs, or cm. Only orders 1 through 15, 23, 30, and 31 may be used between sa and the ca, cs, or cm for which the sa is a preparation. Use of any other operation between sa and ca, cs, or cm will result in the overflow being lost completely, with no other effect on the normal function of the intervening operation.
sl*n	shift left without roundoff	27	11011	Multiply contents of AC and BR by 2^n, as in sl n, but do not roundoff nor clear BR.
sr*n	shift right without roundoff	28	11100	Multiply contents of AC and BR by 2^{-n}, as in sr n, but do not roundoff nor clear BR.
C. Temporary order likely to become permanent:				
qe x	exchange	13	01101	Exchange contents of AC with contents of register x (original contents of AC to register x, original contents of register x to AC).
D. Major changes in the order code:				
rs--	(remote unit) stop	1	00001	To be discontinued: at present it has the temporary function of stopping the computer.
rf k	run forward	2	00010	Discontinued. Replaced by operation si (described below).
rb k	run backward	3	00011	Discontinued. Replaced by operation si (described below).
E. Tentative new order (not yet completely defined):				
si k	select in-out unit			Select the particular piece of terminal equipment (with mode and direction of operation, if necessary) specified by the address.
F. Present temporary orders (to be discontinued when replaced by the more general si, rd and rc operations):				
qh x	h-axis set	6	00110	Transfer contents of AC to register x; set the horizontal position of all display scope beams to correspond to the numerical value of the contents of AC.
qd x	D-scopes display	7	00111	Transfer contents of AC to register x; set the vertical position of the beams of the display scopes to correspond to the numerical value of the contents of AC; display (by intensifying) a spot on the face of the D-display scopes.
qf x	F-scope display	23	10111	Same as operation qd, except display a spot on the face of the F-display scopes.
qr n	read/ shift right	30	11110	Perform two logically distinct functions: 1) Cause the photoelectric tape input reader to read the next character containing a hole in position 7 from tape into digits 0 through 5 of FF Register #3 (holes or no holes in tape positions 1 to 6 becoming ones or zeros in digit columns 0 to 5 respectively). 2) Shift the contents of AC and BR to the right n times. The sign digit is shifted like any other digit and zeros are introduced into the left end. (no roundoff, no BR clear, no sign control). Note: If more than 3 milliseconds (= about 60 orders) elapses between one qr and the next, the reader stops. This must not happen except where 3" of blank tape has been provided for the purpose.
qr*n				Same as qr n above, except the mechanical tape reader, rather than the photoelectric tape reader, is caused to read. The mechanical reader can read line-by-line (i.e. no blank tape is required at places where the reader is allowed to stop).
qp n	punch/ shift right	31	11111	Perform two logically distinct functions: 1) Cause the paper tape output equipment (punch or printer or both, depending on the settings of the switches) to record one character with holes or no holes in positions 1 to 6 on tape corresponding respectively to ones or zeros in digit columns 10 through 15 of AC, and with a hole in position 7. 2) Shift right as in operation qr.
qp*n				Same as qp n above, except no hole is put in position 7.
qs n	index camera	12	01100	Perform two logically distinct functions: 1) Move the next frame of film into place in the display scope camera and open the shutter if it is not already open. The shutter is closed manually when the film is removed for developing. 2) Shift right as in operation qr.

*The fact that the six largest binary digits of the address section of orders sl n, sr n, qr n and qp n are normally zero (and in any case are disregarded by the step counter which counts the shifts) is exploited to permit the addition of an extra variant, as described, to each of the orders mentioned. The star (*) in sl*n, sr*n, qr*n, and qp*n implies that digit 6 (the 2^9 = 512 digit of the address) will be a one, while in sl n, etc., with no star, digit 6 is to be a zero. This can be accomplished during preparation of tape by typing sl (800 + n) for sl*n.

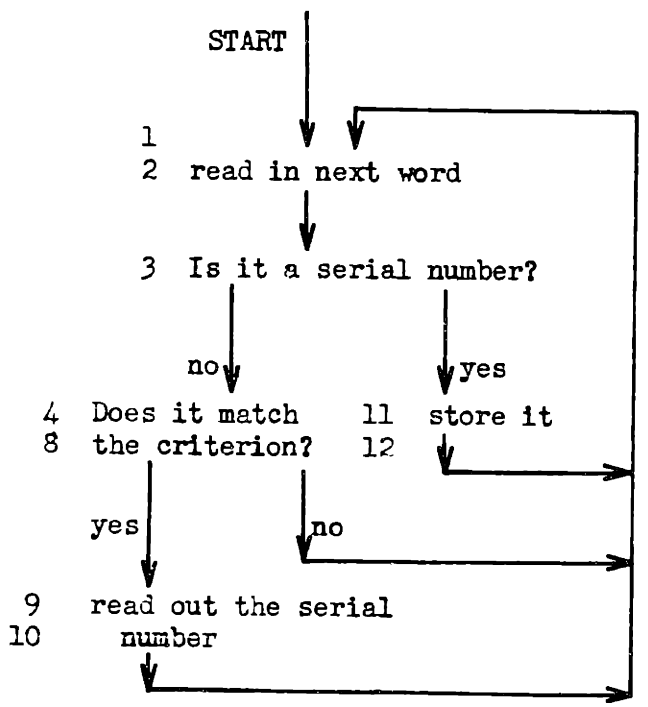
B. Sample Program for Whirlwind I

Arrangement of data which is being fed into the computer; Each "block" of data consists of a serial number (negative), followed by approximately 30 index words (of 16 binary digits each). The computer is to inspect each block of data for a particular word (criterion); and to read out the serial number of each block in which the criterion is found.

Program

Storage register	Order
1	rd 13
2	ca 13
3	cp 11
4	su 14
5	cp 7
6	sp 1
7	ad 16
8	cp 1
9	rc 15
10	sp 1
11	ts 15
12	sp 1
13	---
14	(criterion)
15	(stored serial)
16	1×2^{-15}

Flow Diagram



APPENDIX 4

SEARCHING WITH THE WHIRLWIND I COMPUTER

This appendix is concerned with the details of using the Whirlwind I computer for carrying out the information searching process. It has been included so that the discussions in Chapter 6 would not be obscured by highly technical details. This appendix presupposes familiarity with Appendix 3.

A. The Fundamental Process of Recognizing Equality of Two Numbers

The basic pattern of orders which must be used with the present version of the computer to determine if two numbers are identical is as follows: The arithmetic element (AC) is cleared, and a number X is added in. The second number Y is subtracted from X. Because subtraction is accomplished by the addition of the complement, the difference of two equal numbers results in a representation of zero which the computer recognizes as a negative number. (See Appendix 3, "Short Guide to Coding.") Hence, if X - Y is a positive difference, X cannot equal Y, and the comparison is abandoned as unsuccessful. The computer can choose one of two courses of action based on whether the result of a computation is positive or negative; if the difference X - Y is negative, X may equal Y, and the computer proceeds to make a second test by adding to the difference the smallest increment possible (1×2^{-15}). If X equals Y, the difference would be negative zero, and the addition of the smallest possible positive

number n would result in a positive number. Hence if the result of the second step is positive, then X equals Y. This sequence of orders to compare two numbers is shown as orders 4 though 8 in the sample program in Appendix 3.

Were the computer to be used extensively for the comparison of pairs of numbers, it would be practical to endow the computer with a single order which would initiate one of two possible courses of action dependent on whether or not two numbers were equal. (See below, Special Orders.)

B. "Logical Computation" — The Mathematical Solution of Equations in the Algebra of Logic

The searching process is carried out on the basis of searching for certain combinations of index entries. A set of alternative combinations of events (index entries, in this case) can be expressed as an equation in logical algebra, such that by substituting in the equation an appropriate symbol indicating the presence or absence of an event, the occurrence of events in previously specified combinations will lead to one answer, while all other combinations will lead to another answer.

In the algebra of logic, the events are postulates, and the equation is a syllogism. For certain postulates being true, the syllogism will be true. For all other combinations of true and false postulates, the syllogism will be false. Given the syllogism in the form of an algebraic equation, and given the truth or falsity of each postulate, the truth or falsity of the entire syllogism can be obtained by formal algebraic manipulation.

The rules upon which logical algebra are based are the following:

$P = A + B$ means P is true if either A or B (or both) are true.

$P = A \cdot B$ means P is true if both A and B are true.

$P = A \cdot \bar{B}$ means P is true if A is true and B is simultaneously not true.

Mathematically a true value may be expressed by a 1, and an untrue or false value by a 0. The addition table which will satisfy the first expression for P above is:

$1 + 1 = 1 \qquad 1 + 0 = 1 \qquad 0 + 1 = 1 \qquad 0 + 0 = 0$

The multiplication table is then:

$1 \cdot 1 = 1 \qquad 1 \cdot 1 = 0 \qquad 0 \cdot 1 = 0 \qquad 0 \cdot 0 = 0$

By way of example, let us solve an equation. Suppose that a search were to be directed as follows: locate all documents indexed under either 1) A, B, and at least one of C, D, E, and F, or 2) G, provided not H. Mathematically,

$$P = A \cdot B \cdot (C + D + E + F) + G \cdot \bar{H}$$

Suppose a document is indexed under A, B, D, E, and H. A 1 is substituted in the equation for A, B, D, E, and H, and a 0 is substituted for the remaining C, F, and G.

$$P = 1 \cdot 1 \cdot (0 + 1 + 1 + 0) + 0 \cdot \bar{1})$$

The $\bar{}$ indicates the complement. ($\bar{1} = 0$ and $\bar{0} = 1$). Hence,

$$P = 1 \cdot 1 \cdot (0 + 1 + 1 + 0) + 0 \cdot 0$$

Simplifying according to the addition and multiplication rules above,

$$\begin{aligned} P &= 1 \cdot 1 \cdot (1 + 1) + 0 \\ &= 1 \cdot 1 \cdot 1 + 0 \\ &= 1 + 0 \end{aligned}$$

$$P = 1$$

The presence of A, B, D, E, and H is seen to satisfy the equation; hence this combination is one that satisfies the criteria for the search.

In order to carry out such a computation on the computer, sequences of orders must be devised which perform addition and multiplication and complementing according to the rules of logical algebra. These sequences of orders are as follows, together with an example of how the computation appears in the computer. (The notation RC is an abbreviation for "register containing.")

Logical addition: $1 + 0 = 1$

		Example:
ca RC A		0.000 000 000 000 001
ad RC B		0.000 000 000 000 000
	sum	0.000 000 000 000 001
ad RC 1×2^{-15}		0.000 000 000 000 001
	sum	0.000 000 000 000 010
sr 801 (shift right one place; do not round off)		0.000 000 000 000 001 ans

Logical multiplication: $1 \cdot 0 = 0$

ca RC A		0.000 000 000 000 001
ad RC B		0.000 000 000 000 000
	sum	0.000 000 000 000 001
sr 801		0.000 000 000 000 000 ans

Complementing:

ca RC $\bar{H}$		0.000 000 000 000 000
su RC 1×2^{-15}		1.111 111 111 111 110
	sum	1.111 111 111 111 110
ts C		
cm (obtain positive magnitude)		0.000 000 000 000 001 ans

The number of orders required in a program to obtain the solution to a logical equation will be approximately (number of addition signs) x 3, plus (number of multiplication signs) x 2, plus (number of complements) x 4.

Again in a practical application of the computer, it might be found advisable to provide single orders which would perform a logical addition, multiplication, or complement. (See below, Special Orders)

C. A Program for Searching

On the following page is a complete program for the present Whirlwind computer containing all the orders required to perform the search of a coded index. The alphanumerical combinations in the program are a form of serial notation for addresses (numbers designating the various storage registers). The first part of each address, such as "A5.—" indicates the box on the flow diagram with which the order is concerned. The flow diagram for the program is the one given in Chapter 6, page 71.

The index information is stored in 16-digit groups, where a pair of groups constitutes one complete index entry. (See Chapter 6, page 69.)

This program is intended primarily to be illustrative. Time and space do not permit a detailed tracing through of all the steps of the program; by reference to the "Short Guide to Coding" in Appendix 3 and to the flow diagram (page 71), the reader may, with sufficient time and patience, trace it himself. The program begins with

A1.1	rd B1.1	A9.5	su B4.8	A15.4	sp A9.1	C1.20	cm C3.2
A2.1	ca B1.1	A9.6	ad B9.5	B1.1	—	C1.21	ad C2.7
A2.2	cp A4.1	A9.7	ts B9.7	B3.1	—	C1.22	sr 801 1m
A3.1	sp —	A9.8	td A10.9	B4.1	+0	C1.23	ad C3.1
A4.1	sr 815	A10.1	ca B9.7	B4.4	ri B4.6	C1.24	ad B4.8
A4.2	qe B4.1	A10.2	ad B10.4	B4.6	sp A9.1	C1.25	sp —
A4.3	sl 815	A10.3	td A10.7	B4.7	sp A5.1	C2.1	(A tally)
A4.4	ts B1.1	A10.4	ad B10.4	B4.8	1×2^{-15}	C2.2	(B tally)
A4.5	sr 815	A10.5	td A13.2	B5.1	--	C2.3	(C tally)
A4.6	ad B4.4	A10.6	ca B1.1	B9.1	(a)	C2.4	(D tally)
A4.7	td A3.1	A10.7	sr --	B9.2	(D1.1 - 1)	C2.5	(E tally)
A4.8	td A4.11	A10.8	ts B1.1	B9.3	(ℓ)	C2.6	(F tally)
A4.9	ca B4.8	A10.9	ca —	B9.5	(n)	C2.7	(G tally)
A4.10	ts B9.5	A10.10	su B1.1	B9.6	(k)	C2.8	(H tally)
A4.11	sp --	A10.11	cp A11.1	B9.7	—	C3.1	—
A5.1	sp C1.1	A10.12	sp A14.1 cf	B10.3	(p)	C3.2	—
A5.2	ts B5.1	A11.1	ad B4.8	B10.4	(2ℓ)	D1.1	(criterion 1a)
A5.3	cp A8.1	A11.2	cp A14.1	B12.1	--	D1.2	(criterion 1b)
A6.1	rc B1.1	A12.1	cs B10.4	C1.1	ts C1.25		—
A8.1	ca B4.8	A12.2	ts B12.1	C1.2	ca C2.3	D1.2ℓ	
A8.2	ts B9.1	A12.3	cp A12.5	C1.3	ad C2.4	D2.1	(shift factors)
A8.3	cs B4.1	A12.4	sp A13.1	C1.4	ad B4.8 1a	D2.2ℓ	
A8.4	ts B5.1	A12.5	ao B9.5	C1.5	sr 801	D3.1	(criteria tallies)
A8.5	cs B9.3	A12.6	sp A9.1	C1.6	ad C2.5	D3.ℓ	
A8.6	ts B8.1	A13.1	ca B4.8	C1.7	ad B4.8 1a		
A8.7	ca B10.3	A13.2	ts --	C1.8	sr 801		
A8.8	td A8.10	A13.3	ca B9.6	C1.9	ad C2.6		
A8.9	ca B4.1	A13.4	ts B9.1	C1.10	ad B4.8 1a		
A8.10	ts —	A13.5	ca B9.3	C1.11	sr 801		
A8.11	ao A8.10	A13.6	su B9.6	C1.12	ad C2.2		
A8.12	ao A8.1	A13.7	cp A15.1	C1.13	sr 801 1m		
A8.13	cp A8.5	A14.1	ca B9.1	C1.14	ad C2.1		
A8.14	sp A1.1	A14.2	ts B9.6	C1.15	sr 801 1m		
A9.1	ca B9.1	A14.3	sp A1.1	C1.16	ts C3.1		
A9.2	sl 801	A15.1	ao B9.6	C1.17	ca C2.8		
A9.3	ad B9.2	A15.2	ca B8.1	C1.18	su B4.8 ct		
A9.4	ts B9.6	A15.3	ts B9.5	C1.19	ts C3.2		

a = no. of criteria found in current block

ℓ = no. of criteria

k = ordinal of criterion being checked

n = no. of groups per entry = 2

p = first register of tally table

tallies stored as 1×2^{-15}

logical combination subroutine for $A \cdot B \cdot (C+D+E+F)+G \cdot \bar{H}$

the order at address A1.1, and continues to perform the orders in successive registers except where otherwise directed by sp and cp orders.

The groups of orders encompassed by brackets are those which can be replaced by single special orders, which orders are discussed in the succeeding section.

D. Special Orders

The special orders which have been mentioned above are described here in more technical detail.

la x, logical add: Obtain the logical sum of each digit in the AC (accumulator) with the corresponding digit in register x, retaining the sum in AC. (This replaces three of the present orders -- ad, ad, and sr.)

lm, x, logical multiply: obtain the logical product of each digit in the AC with the corresponding digit in register x, retaining the result in AC. (Replaces two orders.)

cf x, compare: if contents of register x is identical to the contents of the AC, store a positive zero in the AC; if not identical store a negative zero in the AC. This order to be followed by a cp order. (Replaces four orders.)

E. Major Modifications to the Whirlwind Computer to Increase Adaptability to Information Searching

Major additions to the present Whirlwind computer which will permit faster searching of the type outlined in Chapter 6 are the following:

1) Extend the present Check Register from its present 16-digit length to 32 digits. (The Check Register is a register which is at present used for checking purposes only; it gives an output only if the same number is read in (transferred to it) twice in succession.) Modify the Check Register so that it will compare only in the left-most N digit positions.

2) Provide two Input Registers and one Output Register (for temporary storage), each of 32 digit positions. Input register No. 2 stores the complete entry while it is being checked; at the same time the succeeding entry is being read from the external storage medium (tape) and assembled in Input Register No. 1..

3) A storage register of 37 digit positions must be provided for each criterion to which all the entries will be compared.

The necessary switching facilities and wiring would also be included to provide for the performance of the following operations:

1) Assemble the entry in Input Register No. 1, the first group in the left half, and the second group in the right half. This might require an order of the nature of an si order (see Appendix 3, "Short Guide to Coding."), or it might be done automatically.

2) Transfer the assembled entry to Input Register No. 2..

3) Read entry from Input Register No. 2 to Check Register. Read from appropriate criterion storage register to Check Register and simultaneously read left 5 digits (the binary-coded value of N) from the same criterion register to a pulse control unit, which prepares the Check Register for checking only in the left-most N digit positions.

If Check Register indicates that the entry and the criterion match (have the first N digit positions equal to each other), a pulse is created which stores a 1 (a tally) in digit position 6 of the associated criterion storage register.

4)) "Index" to (prepare to read from) the next criteria storage register. If the register contains a criterion, it must contain a 1 digit (a tally) in digit position 7, in which case operations 3 and 4 are to be repeated. If the criterion register is empty, all criteria have been checked, the machine should return to operation 1, (or to operation 2 if operation 1 has already been automatically performed).

5) As each entry is assembled in Input Register No. 1, the first digit is sensed (examined to determine whether it is a 0 or a 1). A serial number has an initial 0, and index entry an initial 1. If the entry is a serial, all digit positions 6 in the criteria storage registers are "checked" by a "logical compute" pulse. The storage elements comprising these digits can be wired in such a way that a single pulse will pass through the network only if the predetermined combination of criteria have been matched. If the desired combination has been satisfied, the serial number is transferred from the Input Register (on the next cycle) to the Output Register, whence it is recorded in some fashion. (The logical computing circuit can be set up by card-programmed connections of gate tubes associated with digits 6.)

It is not likely that the present "electrostatic storage" registers will be utilized at all. All the operations could be performed with a group of registers similar to the present Test Storage, which would require about 25 microseconds for each order. With 8 criteria, the searching time would be roughly 240 microseconds per entry, or 7200 microseconds per block of 30 entries; the searching rate would then be about 138 blocks per second.

BIBLIOGRAPHY - see Bibliography to Chapter 6.