NATIONAL
PHYSICAL
LABORATORY

Automatic Digital Computation

PROCEEDINGS OF A SYMPOSIUM
HELD AT THE
NATIONAL PHYSICAL LABORATORY
ON MARCH 25, 26, 27 & 28, 1953

LONDON: HER MAJESTY'S STATIONERY OFFICE
1954
Price £1 1s 0d net

NATIONAL PHYSICAL LABORATORY

AUTOMATIC
DIGITAL
COMPUTATION

LONDON: HER MAJESTY'S STATIONERY OFFICE
1954

PREFACE

The International Symposium on Automatic Digital Computation, held at the National Physical Laboratory, Teddington in March, 1953, was the third of its kind in this country, the first having been held at Cambridge in June, 1949, and the second at Manchester in July, 1951 in connexion with the inauguration of the Manchester University computing machine.

In the belief that a great part of the value of such symposia lies in the discussions, copies of the papers to be presented were distributed in advance and speakers were asked to assume that they had been read. This saving in time in the presentation of the papers gave a useful increase in the time available for the discussions. The summarized reports of the discussions are based on notes taken at the time by the various reporters, checked and supplemented by reference to a magnetic-tape recording of the proceedings. It has, of course, been necessary to condense the reports of the discussions very considerably in order to keep the record to a reasonable size, but it is hoped that nothing of outstanding importance has been omitted.

The organizers of the Symposium wish to take this opportunity of thanking those who prepared and presented the papers, particularly those who helped very materially by supplying the copies needed for the advance distribution.

National Physical Laboratory,
TEDDINGTON, Middlesex.

1954.

CONTENTS

Page

OPENING ADDRESS
by Professor D. R. Hartree,
Cambridge University. 1

BRITISH MACHINES

1. THE PILOT ACE
by J. H. Wilkinson,
National Physical Laboratory 5

2. THE EDSAC
by M. V. Wilkes,
Cambridge University Mathematical Laboratory 17

3. OPERATING AND ENGINEERING EXPERIENCE GAINED WITH LEO
by J. M. M. Pinkerton,
J. Lyons & Co. Ltd. 21

4. MADAM
by F. C. Williams,
Manchester University 35

5. MOSAIC - the "Ministry of Supply Automatic Computer" -
by A. W. M. Coombs,
Post Office Research Station, London 38

6. NICHOLAS
by N. D. Hill,
Elliott Bros. (London) Ltd., Research Laboratories 45

7. ADVANCE NOTES on RASCAL
by E. J. Petherick,
Royal Aircraft Establishment, Farnborough. 46

8. THE TRE HIGH-SPEED DIGITAL COMPUTER
by R. H. A. Carter,
Telecommunications Research Establishment 56

PROGRAMMING

9. OPTIMUM CODING
by G. G. Alway,
National Physical Laboratory. 65

10. MICROPROGRAMMING AND THE CHOICE OF ORDER CODE
by J. B. Stringer,
Cambridge University Mathematical Laboratory 71

11. CONVERSION ROUTINES
by E. N. Mutch and S. Gill,
Cambridge University Mathematical Laboratory 74

12. GETTING PROGRAMMES RIGHT
by S. Gill,
Cambridge University Mathematical Laboratory 80

Page

DESIGN

13. SPECIAL REQUIREMENTS FOR COMMERCIAL OR ADMINISTRATIVE APPLICATIONS
by T. R. Thompson,
J. Lyons & Co. Ltd. 85

14. INPUT AND OUTPUT
by D. W. Davies,
National Physical Laboratory 102

15. ECHELON STORAGE SYSTEMS
by D. O. Clayden,
National Physical Laboratory 117

16. SERIAL DIGITAL ADDERS FOR A VARIABLE RADIX OF NOTATION
by R. Townsend,
The British Tabulating Machine Co. Ltd. 120

THE UTILIZATION OF COMPUTING MACHINES - I

17. MATHEMATICS AND COMPUTING
by A. van Wijngaarden,
Mathematisch Centrum, Amsterdam 125

18. LINEAR ALGEBRA ON THE PILOT ACE
by J. H. Wilkinson,
National Physical Laboratory 129

19. THE NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS
by L. Fox and H. H. Robertson,
National Physical Laboratory 137

20. THE SOLUTION OF PARTIAL DIFFERENTIAL EQUATIONS
by N. E. Hoskin,
Manchester University 147

THE UTILIZATION OF COMPUTING MACHINES - II

21. MATHEMATICAL TABLES
by E. T. Goodwin,
National Physical Laboratory 155

22. APPLICATIONS OF ELECTRONIC MACHINES IN PURE MATHEMATICS
by J. C. P. Miller,
Cambridge University Mathematical Laboratory 160

23. THE APPLICATION OF AUTOMATIC COMPUTING MACHINES TO STATISTICS
by K. D. Tocher
Imperial College, London 166

24. GENERAL DISCUSSION ON "MACHINE UTILIZATION" 178

CIRCUITRY AND HARDWARE

25. GATES AND TRIGGER CIRCUITS
by W. W. Chandler,
Post Office Research Station, London 181

25a. PARALLEL FERRORESONANT TRIGGERS
by J. Garcia Santesmases,
Consejo Superior de Investigaciones Cientificas, Spain 186

26. MERCURY DELAY LINE STORAGE
by M. A. Wright,
National Physical Laboratory 195

27. APPLICATIONS OF MAGNETOSTRICTION DELAY LINES
by R. C. Robbins and R. Millership,
Elliott Bros. (London) Ltd., Research Laboratories 199

28. CATHODE RAY TUBE STORAGE
by T. Kilburn,
Manchester University 212

29. MEMORY STUDIES AND OTHER DEVELOPMENTS AT THE NATIONAL BUREAU OF STANDARDS
by Ralph J. Slutz,
National Bureau of Standards 217

SERVICING AND MAINTENANCE

30. PREVENTIVE OR CURATIVE MAINTENANCE
by E. A. Newman,
National Physical Laboratory 235

31. EXPERIENCE WITH MARGINAL CHECKING AND AUTOMATIC ROUTINING OF THE EDSAC
by M. V. Wilkes, M. Phister, Jnr., and S. A. Barton,
Cambridge University Mathematical Laboratory 239

32. DIAGNOSTIC PROGRAMMES
by R. L. Grimsdale,
Ferranti Ltd. 246

33. COMPONENT RELIABILITY IN A COMPUTING MACHINE AT MANCHESTER UNIVERSITY
by A. A. Robinson,
Ferranti Ltd. 252

MEDIUM SIZE DIGITAL COMPUTING MACHINES

34. THE HARWELL COMPUTER
by E. H. Cooke-Yarborough,
Atomic Energy Research Establishment,
Harwell 259

35. THE APE(X)C: A LOW-COST ELECTRONIC CALCULATOR
by A. D. Booth,
Birkbeck College Electronic Computation Laboratory 264

MEDIUM-SIZE DIGITAL COMPUTING MACHINES (Contd.)

36. THE ELLIOTT-NRDC COMPUTER 401 - A DEMONSTRATION OF COMPUTER ENGINEERING BY PACKAGED UNIT CONSTRUCTION
by W. S. Elliott, H. G. Carpenter and A. St. Johnston,
Elliott Bros. (London) Ltd., Research Laboratories 273

37. MEDIUM-SIZE DECIMAL COMPUTING MACHINE
by N. Kitz, Instituto Nazionale per le Applicazione
del Calcolo, Italy 276

38. THE DESIGN REQUIREMENTS OF A LOW-COST COMPUTING MACHINE
by K. D. Tocher,
Imperial College, London 281

APPENDICES

I. Alphabetical List of Organizations Represented and their Delegates. 285

II. Alphabetical List of Delegates. 293

AUTOMATIC DIGITAL COMPUTATION

Opening Address

by

Prof. D. R. Hartree

Cambridge University

It will be two years this July since there was last a conference on automatic digital calculating machines in this country, the occasion for that conference being the inauguration of the Ferranti machine at the University of Manchester. Since then there have been various developments: on the engineering side, progress with and completion of other machines, and, more particularly in America, research on new kinds of storage elements and other machine components; on the maintenance side, in connexion with testing and diagnosis of machine faults; and on the operating side, in connexion with programming. The papers to be given at this conference, and the numbers present to listen to them, show that the time is ripe for another conference on this subject, and I would like on your behalf to thank the Director of the National Physical Laboratory for making it possible to hold it here, and those of his staff who have been responsible for the detailed arrangements.

Five or six years ago, it was possible for someone interested in the subject of automatic digital calculating machines to know something of most of the main projects in this field in the world. But this is hardly possible any longer. This is not only because of the much greater number of groups concerned with work on these machines, but because the scope of the subject has widened greatly. With the successful completion of several machines and the accumulation of experience in using them, there have been developments in programming, and also in various branches of numerical analysis in which advance has been stimulated by the potentialities of these machines, and it would be something like a full-time job to keep well acquainted with all that is going on in this field in this country alone. I therefore apologize in advance if I show myself less familiar than I would like to be with some of the work on which some of you are engaged.

One development which I will mention specifically is the work on the machine called LEO at the head office of Messrs J. Lyons & Co. Ltd; and it is worthy of special mention because it is, as far as I know, the first high-speed automatically sequenced machine to be built primarily for commercial and clerical work. A characteristic feature of such work is that the amount of input and output, relative to the amount of computation, is very much larger than in most calculations relating to scientific and technical problems. This feature, and its influence on the machine, is discussed in Mr. Thompson's paper; for the present let me commend the enterprise of Messrs. J. Lyons & Co. Ltd., collectively and Mr. Thompson individually in undertaking and carrying through this project.

One of the important steps in the development of the more recent automatic digital calculating machines was the recognition of the possibility of representing operating instructions within the machine in the same form as numbers, and of the advantages which would follow from this. So long as there was no completed and operating machine making use of this idea and giving opportunity for experiments in the practical exploitation of it, interest was mainly concentrated on projected machines and their functional and detailed design. It is of course possible to do paper programming for an imaginary machine, or for one under construction but not yet operational; but such programming may be rather unrealistic, and, as far as my own experience goes, one's ideas of programming are apt to change rapidly and considerably as soon as one has a working machine on which to try them out.

But when machines did become available and some experience of using them accumulated, the subject of programming took a bound forward, in much the same way as the subject of machine design and construction developed from exploratory experiments to the successful completion of machines two or three years earlier. This is illustrated by the number of papers concerned with programming in the joint A.I.E.E. - I.R.E. - S.C.M. conference at

Pittsburgh (June 1952) as compared with slender reference - almost as an afterthought - to programming in the report of the New York conference (December 1951). It seems that in the interval the power of programming techniques such as the use of the library of subroutines, interpretive routines, and conversion or translation routines, had become much more widely appreciated and studied; and I understand from those who were at the Toronto conference and the M.I.T. summer school last year that there is now considerable activity in America on these more sophisticated aspects of programming.

Engineering and Programming

I do not want to give the impression that I regard engineering and programming as unconnected features of an automatic digital calculating machine. The facilities provided by the engineer are closely related to programming, and conversely experience of programming may suggest facilities which one would like to have built into the machine. Further, the operation specified by a single machine instruction may involve a sequence of more elementary operations - for example, multiplication involves a sequence of additions, shifts, counts, and tests - and the design of the control system can be systematized by applying the ideas of programming to the sequence of elementary operations involved in carrying out one machine instruction. This is the subject of Mr. Stringer's paper on "Microprogramming".

An example of the effect of engineering on programming is provided by the facilities afforded by the B-registers on the Manchester machine. In a machine with a single adder and accumulator, any modification of instructions in the course of the calculation involves the use of the accumulator, and to make it available for this purpose, any other intermediate results must be moved out of it, and then replaced when the modification of instructions has been carried out. Provision of a second adder and accumulator, perhaps only with limited facilities, enables such modification of instructions to be carried out without the disturbance of partial sums being accumulated in the main accumulator, and this, as well as simplifying programming, probably increases the overall speed and decreases the total storage space required for instructions. I am aware, of course, that the B-registers in the Manchester machine are not actually used quite in this direct way for altering instructions in the store; the effect is obtained in another way, involving similar hardware additional to that of a machine with a single adder and accumulator. Incidentally, some such facilities could be provided alternatively by microprogramming.

Another effect of engineering on programming is the effect of the provision of an auxiliary store. This may well affect not only programming but the calculation to be programmed. For example, in the numerical integration of ordinary differential equations some methods require special starting procedures, and, for that reason, have been regarded as less suitable than methods not requiring such special starting procedures. But with an auxiliary store it may be quite practicable first to have a special starting subroutine in the main store and, once the integration has got under way, to replace it by the main integration procedure.

Before I leave the engineering aspect, let me mention two possible future developments.

In most calculations, the number of storage locations whose contents are altered - apart from the result of transfers from an auxiliary store - is small compared with the total number used. It is only for those whose contents *are* changed that one needs a short read-in time; for the rest only a short access time and read-out time is necessary. Thus if removal of the condition of short read-in time could lead to a storage system, with short access time, simpler than anything at present available, it could be used for the greater part of the store.

Secondly, means of scanning rapidly a sequence of storage locations and determining which contained the number of greatest modulus would be of great service in any calculation of a kind which in a hand calculation would be done by "relaxation" methods.

Development of Programming

In the development of programming there has been a marked tendency to depart further and further from direct programming in terms of the machine instruction code.

One step in this direction was Wheeler's free use of relative addresses, the appropriate origin for the address in each instruction being specified by a marker symbol which is interpreted by the machine in the course of reading the input tape.

Another step was the use of "interpretive" subroutines which specify special interpretations to be given to "words" of certain kinds, so that a single "word" - used by the programmer as a single instruction - in the main program can be used to initiate a whole sequence of operations involving a long sequence of machine instructions.

Another step is the use of "floating addresses" for cross-reference between instructions; each instruction which is referred to by other instructions is marked by a special indication and is referred to by that mark; as the instructions are read in, a list is made of the addresses into which the marked instructions are placed, and the addresses in the other instructions which refer to them are entered accordingly.

A more recent development is the use of "conversion" or "translation" routines to convert into a sequence of machine instructions a program written - in whole or in part - in some other form. For example, a conversion routine might convert the data and formulae (for the evaluation of a sextic polynomial):

$$\begin{array}{l} x = C(100),\ a_n = C(120+n), \\ y_0 = a_0,\ y_n = y_{n-1}\, x + a_n, \\ y_6 \quad \text{to } 200, \end{array}$$

supplied in suitably coded form, into a sequence of machine instructions. The main problem in such conversion routines, I think, is devising means of building up not the instructions for the purely arithmetical operations but those for the organizational side of the work such as counting and switching.

A feature of this departure from the machine instruction code in programming is that it may increase the difficulty of finding mistakes in programs. Even in programming in terms of machine instructions, use of subroutines sometimes causes difficulties, for example when a mistake in a program results in a spurious instruction being planted within a library subroutine; one of the advantages of library subroutines is that they have been - or should have been - adequately checked, and one does not immediately go looking for program faults in them so a fault planted in a library subroutine by the operation of instructions elsewhere in the program may take some time to diagnose. The difficulty is likely to be much greater when conversion routines are used, because the programmer will usually not know what instructions ought to be in what places in the store, so that a "post-mortem" procedure will be of little use in finding mistakes.

In view of these developments in programming, I would like to see more publication of papers on programming and on the use of these machines. There are, I know, a number of reports with limited circulation, but issue of these does not constitute publication in the sense of making generally known or available. Even the report of the Manchester conference of July 1951 appears to have been issued and distributed privately. This is a pity, as the papers in it are of considerable interest, but they cannot be regarded as generally available for reference; and the same applies to some other reports. In particular the ACE system of programming, with its three-address code and timing numbers, and its facilities for optimum programming, is a good deal different from any other system I know, and is of considerable interest, but I do not know of any accessible reference to give to anyone enquiring for information about it.

Terminology

I would like to end with a few remarks on terminology. It is inevitable that each group working on the construction or use of a machine should develop a terminology particularly appropriate to the local conditions. But it is unfortunate if in general discussion or in published work the local dialect is too freely used without explanation. For example with a serial cathode ray tube store the term "line" for a storage location, or for its content, is a natural one; but the term "line" for a storage location would be inappropriate even in a parallel c.r.t. store, and use of this term in this sense without explanation in a general context is merely confusing to those unfamiliar with the particular storage system to which it is appropriate. For this reason I used earlier the term "B-register" rather than "B-line", in referring to the facility provided by these registers in the Manchester machine.

About 6 months ago a letter by Professor Kapp appeared in "Nature" on this subject of terminology in connexion with automatic digital machines; this was well answered by Dr. Booth *(ref. 1)*. I find myself in agreement with Professor Kapp in his disapproval of the term "memory" for "store"; but he also objected to the term "instruction" or "order" and suggested the term "setting" or "adjustment" which he said "have exactly the same meaning". Here Professor Kapp's suggestions seem to be quite inadequate and also misleading; I can only think that he had not at all understood what is meant by an "instruction" in this context. A "setting" suggests something that is fixed and is determined by the operator, whereas a machine instruction may be altered by the machine and whether or not this is done may depend on the course of the calculation and may not be known to the operator; or it may be constructed by the machine by means of a conversion routine and the operator may never know what it is or where in the store it is; he may even be entirely ignorant of its existence. This is hardly covered by the terms "setting" or "adjustment".

A term which I would like to see used with more care is the word "control", which seems to be used indiscriminately for an operation (as in "transfer of control") and for the piece of hardware which carries out this function (as in the "current instruction is sent to control"). This can lead to confusion, and I suggest the word "control" be restricted to the operation, and the hardware should be called the "control system" or "control unit".

No great harm is done if different people use different words for the same thing; it is the use of the same word for different things (as in the case of "control") which leads to confusion, and in this connexion I would like to mention the term "floating address". In the sense in which I have used it earlier, this was introduced by Professor Adams of MIT as being preferable to the earlier term "free-address" which in speaking is very easily confused with "three-address", but it is also used in the sense of "relative address".

Lastly, may I press for the use of the spelling "program" without the superfluous terminal "-me". We do not write the French forms "telegramme", "diagramme"; why use the French form "programme"? there has been objection to "program" on the ground that it is an Americanism and, for this reason (so it is implied) reprehensible. But it is not an Americanism; reference to the O.E.D. and Fowler's "Modern English Usage" will show that it is a well-established English word, of respectable age, derived from the Greek through Latin, and that "programme" is a reintroduction through the French. When we have a perfectly good English word, why should we prefer a Gallicism? And why, anyway, should a Gallicism be acceptable and an Americanism - even if it were one - unacceptable?

References

1. *Nature*, 1952, **170**, 547.

BRITISH MACHINES

Chairman: Mr. F. M. Colebrook

I. The Pilot ACE

by

J. H. Wilkinson

National Physical Laboratory

Introduction

A machine which was almost identical with the Pilot ACE was first designed by the staff of the Mathematics Division at the suggestion of Dr. H. D. Huskey during his stay at the National Physical Laboratory in 1947. It was based on an earlier design by Dr. A. M. Turing and its principal object was to provide experience in the construction of equipment of this type. It was not intended that it would be used on an extensive programme of computation, but it was hoped that it would give practical experience in the production of subroutines which would serve as a useful guide to the design of a full scale machine. An attempt to build the Pilot Model, during Dr. Huskey's stay, was unsuccessful, but a year later after the formation of an Electronics Section at the NPL a combined team consisting of this section and four members of the Mathematics Division started on the construction of a Pilot Model, the design of which was taken over almost unchanged from the earlier version. The machine first worked, in the sense that it carried out automatically a simple sequence of operations, in May 1950 and by the end of that year it had reached the stage at which a successful Press Demonstration was held. The successful application of the machine to the solution of a number of problems made it apparent that, in spite of its obvious shortcomings, it was capable of being converted into a powerful computer comparable with any then in existence and much faster than most. Accordingly a small programme of modifications was embarked upon early in 1951, but the machine was not functioning satisfactorily again until November of that year. After a month of continuous operation it was transferred from the Electronics Section to Mathematics Division where it has since been in use on a 13-hour day. During its first year of full scale operation it achieved a 65% serviceability figure based on a very strict criterion. Its performance during its second year has so far been considerably better than this.

General Description

The Pilot ACE is a serial machine using mercury delay line storage and working at a pulse repetition rate of 1 megacycle/sec. Its high speed store consists of 11 long delay lines each of which stores 32 words of 32 binary digits each, with a corresponding circulation period of 1024 microseconds, 5 short lines storing one word each with a circulation period of 32 microseconds and two delay lines storing two words each. It was inevitable that in the design of a machine originally intended for experimental purposes, overriding consideration should be given to the minimization of equipment rather than to making the machine logically satisfying as a whole. This is reflected to a certain extent in the code adopted for the machine and in its arithmetic facilities, which are in general fairly rudimentary. The design of the machine was also decisively influenced by the attempt to overcome the loss of speed due to the high access time of the long storage units. The machine in fact uses what is usually known as a system of "optimum coding".

Code of Pilot ACE

The Pilot ACE may be said to have a "three-address code" though this form of classification is not particularly appropriate. Each instruction calls for the transfer of information from one of 32 "sources" to one of 32 "destinations" and selects which of eight long delay lines will provide the next instruction. This third address is necessary because consecutive instructions do not occupy consecutive positions but are placed in such relative positions that, in so far as is possible each instruction emerges during the minor cycle in which the current instruction is completed. An unusual feature of the instructions is that the transfers they describe may last for any number of consecutive minor cycles from one to thirty two. The instruction word contains three other main elements which are known as the wait number, the timing number and the characteristic which together determine when the transfer starts, when it stops and which instruction in the selected instruction source is the next to be obeyed. The structure of the instruction word is as follows:

Next instruction source	Digits 2-4
Source	Digits 5-9
Destination	Digits 10-14
Characteristic	Digits 15-16
Wait number	Digits 17-21
Timing number	Digits 25-29
Go digit	Digit 32

The remaining digits are spare.

Coding of a problem takes place in two parts, in the first of which only the source, the destination and the period of transfer are specified, the last being a function of the characteristic, wait number and timing number. In the second part, the detailed coding, the other elements are added.

The sources and destinations

Simplest among the sources and destinations are those associated with the short delay lines. The six one-word delay lines are each given numbers and these for reasons associated with the history of the machine are 11, 15, 16, 20, 26 and 27. They are usually referred to as Temporary Stores or TS's because they are used to store temporarily those numbers which are being operated upon most frequently at each stage of a computation. In general TSn has associated with it a source, source n, and a destination, destination n. An instruction of the type

15 - 16

in the preliminary stage of the coding represents the transfer of a copy of the contents of TS15 via source 15 to TS16 via the destination 16. After it has taken place both stores contain the number originally in TS15. The period of the transfer is not mentioned in the coding because a transfer of more than one minor cycle is irrelevant. Most transfers are for 1 minor cycle and hence the period of transfer is not specified unless it is greater than one minor cycle. Associated with the TS's are a number of functional sources and destinations. TS16 for instance has two other destinations 17, and 18 associated with it, in addition to destination 16. Any number transferred to destination 17 is added to the contents of TS16 while any number transferred to destination 18 is subtracted from the contents of TS16. TS16 may be said to have some of the functions associated with the accumulator on an orthodox machine. The period of transfer to destinations 17 and 18 is very important. Thus

15-17 (n minor cycles)

has the effect of adding the contents of TS15, n times to the contents of TS16. This prolonged transfer is used in this way to give small multiples (up to 32) of numbers. Similarly, we may have

15-18 (n mc)

The instruction

16-17 (n mc)

is of special significance because it has the effect of adding the content of TS16 to itself for each minor cycle of the transfer, that is it gives multiplication by 2^n or a left shift of n binary places.

TS26 has associated with it a number of functional sources. Source 17 gives the ones complement of the number in TS26, Source 18, the contents divided by 2, and Source 19, the contents multiplied by 2. The instruction

18-26 (n mc)

thus has the effect of dividing the contents of TS26 by 2^n, that is a right shift of n places. Similarly

19-26 (n mc)

gives a left shift of n places.

There are two functional sources which give composite functions of the numbers in TS26 and TS27. These are Source 21 which gives the number

TS26 & TS27

and Source 22 which gives the number

TS26 $\not\equiv$ TS27.

There are a number of sources which give constant numbers which are of frequent use in computation. These are Source 23 which gives the number which has a zero everywhere except in the 17th position, usually known as P17, Source 24 which gives P32, Source 25 which gives P1, Source 28 which gives zero and Source 29 which gives a number consisting of 32 consecutive ones. These sources are valuable because they provide numbers with an access time of one minor cycle and are thus almost as useful as several extra TS's.

The use of a number of TS's with the arithmetic facilities distributed among them makes it possible to take advantage of the placing of instructions in appropriate positions in the long storage units so that they emerge as required. The coding of a trivial example will illustrate the uses of the TS's and their associated sources. It is required to build up the successive natural numbers, their squares and their cubes simultaneously. It is natural to store the values in TS's and we may suppose TS15 contains n, TS20, n^2 and TS26, n^3.

Instruction			Description	
1.	28-15		zero to TS15 *i.e.* 0	These 3 instructions set the initial values
2.	28-20		zero to TS20 *i.e.* 0^2	
3.	28-26		zero to TS26 *i.e.* 0^3	
4.	26-16		TS16 contains n^3	
5.	20-17	(3mc)	TS16 contains $n^3 + 3n^2$	
6.	15-17	(3mc)	TS16 contains $n^3 + 3n^2 + 3n$	
7.	25-17		TS16 contains $n^3 + 3n^2 + 3n + 1$	
8.	16-26		TS26 contains $(n + 1)^3$	
9.	20-16		TS16 contains n^2	
10.	15-17	(2mc)	TS16 contains $n^2 + 2n$	
11.	25-17		TS16 contains $n^2 + 2n + 1$	
12.	16-20		TS20 contains $(n + 1)^2$	
13.	15-16		TS16 contains n	
14.	25-17		TS16 contains (n + 1)	
15.	16-15		TS15 contains (n + 1) Next instruction (4)	

The instructions (1) to (3) set the initial conditions. The instruction (4) - (15) have the effect of changing the contents of 15, 20, 26 from n, n^2, n^3 to (n + 1), $(n + 1)^2$, $(n + 1)^3$. As remarked earlier, each instruction selects the next instruction and here instruction (15) selects instruction (4) as the next instruction. In the preliminary coding this is usually denoted by using an arrow; it must be catered for in the detailed coding by the correct choice of the timing number, as will be shown below.

The branching of a programme is achieved by the use of two destinations, destination 24 and destination 25. If a transfer is made from any source to destination 24 then the next instruction is one or other of two according as the number transferred is positive or negative. Similarly if a transfer is made to destination 25 then the next instruction is one or other of two according as the number transferred is zero or non-zero. In the preliminary coding the bifurcation is denoted by the use of arrows, thus.

$$\begin{tikzcd} & 15 - 24 \arrow[dl] \arrow[dr] & \\ +\text{ve} & & -\text{ve} \end{tikzcd}$$

In the detailed coding the effect is that if the number transferred to destination 24 is negative then the timing number is increased by 1. Similarly for destination 25; the two possible next instructions are consecutive in the store.

The two double word stores are numbered DS12 and DS14 DS12 has only source 12 and destination 12 associated with it, but DS14 has, in addition to source 14 and destination 14, a number of functional sources and destinations. Source 13 gives the contents of DS14 divided by 2, while transfers to destination 13 have the effect of adding the numbers transferred to DS14. In specifying transfers from, and to, the double length stores, the time of the transfer must be specified, *i.e.* whether it takes place in an even or an odd minor cycle or both. Thus the transfer

12-14 (odd minor cycle) usually written
12-14 (o)

represents the transfer of the word in the odd positions of DS12 to the odd position in DS14 while

12-14 (2 minor cycles)

represents the transfer of both words in 12 to the corresponding positions in 14. The operation

13-14 (2n)

gives us a method of shifting the contents of TS14 n places to the right while

14-13 (2n)

produces a shift of n places to the left.

The machine is not equipped with a fully automatic multiplier. To multiply two numbers, a and b together, a must be sent to TS20, b to DS14 odd, zero to DS14 even and a transfer (source irrelevant) made to destination 19. The product is then produced in DS14 in 2 milliseconds, but a and b are treated as positive numbers. Corrections must be made to the answer if a and b are signed numbers. To make multiplication fast, it has been made possible to perform other operations while multiplication is proceeding. Thus the corrections necessary if a and b are signed numbers may be built up in TS16 during multiplication, and signed multiplication takes only a little over two millisecs. It is, of course, therefore, a subroutine but a very fast one. The amount of equipment associated with the multiplier is very small. The main part of the store consists of the long storage units known as DL1, DL2 DL11. Each of these has a source and a destination with the same number as the DL number. The words in each DL are numbered 0 to 31 and the nth word in DLM is usually denoted by DLM_n. Transfers to and from long lines in the preliminary coding are denoted thus

8_n-16 (transfer n^{th} word of DL8 to TS16)
8_{m-n}-17 (add all the words from 8_m to 8_n *i.e.* n-m+1 consecutive words of DL8 to TS16).

Detailed coding

In the second stage of the coding the true instruction words are derived from the preliminary coding. This is a fairly automatic process and recent experience has shown that it can be carried out satisfactorily by quite junior staff. The timing of each instruction is given relative to the position of that instruction in the store. This is an incidental feature of the code which arose from the attempts to minimize equipment. It would be dropped in any future machine in favour of an absolute timing system. If an instruction occupies position m in a DL and has a wait number W and timing number T then the transfer always begins in minor cycle (m + W + 2) and the next instruction is always in minor cycle (m + T + 2) of the selected next instruction source. The period of transfer depends on the value of the characteristic. If the characteristic is zero then the transfer lasts for the whole period from (m + W + 2) to (m + T + 2), that is (T-W + 1) minor cycles. If the characteristic is one, then the transfer is for one minor cycle, that is minor cycle (m + W + 2). If the characteristic is three then the transfer is for two minor cycles (m + W + 2) and (m + W + 3). The characteristic value, two, is not used. The characteristic value zero, gives a prolonged transfer which is peculiar to the Pilot ACE. The characteristics 1 and 3 are analogous to the facility on EDSAC whereby full length or ½-length words may be transferred. On the Pilot ACE we transfer single or double length words. This facility is invaluable for double length, floating and complex arithmetic. In the above definitions the numbers (m + W + 2) etc. are to be interpreted modulo 32. In general timing and wait numbers are simpler than they appear from the definitions because they are very frequently both zero, corresponding to a transfer for one minor cycle. The detailed coding of the problem given earlier will illustrate the procedure. All the instructions are in DL1 so that the next instruction source is always one. The key to the headings in the following table is

m.c.	Minor cycle position of instructions in DL1.
N.I.S.	Next instruction source.
S	Source.
D	Destination
C	Characteristic
W	Wait number
T	Timing number

The last column gives the position of the next instruction in DL1; it is given by (m + T + 2). The first 4 instructions occupy minor cycles 0, 2 and 4, 6 and each takes two minor cycles, and gives a transfer for one minor cycle only. The next instruction occupies minor cycle number 8 and it requires a transfer lasting 3 minor cycles. The simplest and fastest way of getting this is to have W = 0 and T = 2 giving a transfer of (2 - 0 + 1) minor cycles. The next instruction is in position (8 + 2 + 2), that is minor cycle 12, and so on. When we reach the instruction in minor cycle 31, viz. 25-17, a transfer for one minor cycle is required. The simplest way is to have W = 0 T = 0 and this makes the next instruction occupy position (31 + 0 + 2) i.e. position 33 which is position 1. If position 1 had been already occupied, a value of T could have been chosen in order to land in an unoccupied position. In order to ensure that a transfer of one minor cycle only took place, the characteristic could have been made 1. It should be appreciated that the choice of C, W and T is far from unique. Whenever possible T = 0 and W = 0 are chosen because this gives the highest speed of operation besides being simplest. The instruction occupying position 1 is of special interest because this is the last instruction of the cycle needed to build up a square and cube and it must select as its next instruction the first of the cycle, which is, in position number 6. This is achieved by making T = 3 (giving the next

Minor cycle position of instructions in DL1	Next instruction Source	Source	Destination	Charact-eristic	Wait No.	Timing No.	Minor cycle position of next instruction
0	1	28	15	0	0	0	(2)
1	1	16	15	0	0	3	(6)
2	1	28	20	0	0	0	(4)
3							
4	1	28	16	0	0	0	(6)
5							
6	1	26	16	0	0	0	(8)
7							
8	1	20	17	0	0	2	(12)
9							
10							
11							
12	1	15	17	0	0	2	(16)
13							
14							
15							
16	1	25	17	0	0	0	(18)
17							
18	1	16	26	0	0	0	(20)
19							
20	1	20	16	0	0	0	(22)
21							
22	1	15	17	0	0	1	(25)
23							
24							
25	1	25	17	0	0	0	(27)
26							
27	1	16	20	0	0	0	(29)
28							
29	1	15	16	0	0	0	(31)
30							
31	1	25	17	0	0	0	(1)

instruction in m.c. 1 + 3 + 2 = 6). This incidentally gives a transfer lasting four minor cycles but since it is a transfer from one TS to another and no functional source or destination is in use, the prolonged transfer produces no harmful effect. If a prolonged transfer had to be avoided then the characteristic could be taken as 1. It is seldom necessary to use any characteristic other than zero for transfers to and from TS's but when transfers are made to and from DL's, characteristic values of 1 or 3 are almost universal. All 12 instructions which comprise the repeated cycle of the computation take a total time of one major cycle exactly (32 minor cycles) the last instruction of the cycle having been specially designed to get back to the beginning of the cycle. This is in contrast to the position in a machine not using optimum coding, where 12 major cycles would be necessary quite apart from the fact that the multiplications by factors of 3 and 2, each of which uses one instruction, would normally need more than one instruction if a prolonged transfer were not available. Fig. 1 gives a simplified diagram of the machine. The sequence of events in obeying the instruction

N	S		D	C	W	T
2	16	-	20	0	8	10

occupying $DL1_2$ for example is as follows. Starting from the time when the last instruction was completed, the instruction from $DL1_2$ will have passed into the special TS marked TS COUNT during minor cycle number 2. By the end of minor cycle number 3, S switch number 16 will be over and also N switch number 2. The contents of TS16 will be passing into HIGHWAY and those of DL2 into INSTRUCTION HIGHWAY. At the beginning of minor cycle number 12 (*i.e.* 2 + 8 + 2), D switch number 20 will go over, and TS20 will stop recirculating and the number on the HIGHWAY will pass into TS20. The transfer will continue until minor cycle 14 (*i.e.* 2 + 10 + 2) when the D switch number 20 will switch back. At the beginning of minor cycle 14, the switch X on COUNT will go over and the number on INSTRUCTION HIGHWAY during this minor cycle, $DL2_{14}$, will pass into COUNT. At the end of minor cycle 14, the X switch will close again and $DL2_{14}$ will be trapped in COUNT. The cycle of events is now complete. COUNT is associated with a counter and it is this counter which determines from the wait, timing, and characteristic numbers of the trapped instruction, when the D and X switches go over and back.

Input and Output

The only part of the instruction word not described is the GO digit. If the GO digit is a one, the instruction is carried out at high speed, but if it is a zero the machine stops and does not proceed until a manual switch is operated. The GO digit is omitted in strategic instructions when a programme is being tested. It also serves a further purpose in synchronising the input and output facilities with the high speed computer. Input on the machine is by means of Hollerith punched cards. When cards are passed through the reader the numbers on the card may be read row by row as each passes under a set of 32 reading brushes. When a row of a card is under the reading brushes, the number punched on that row, regarded as a number of 32 binary digits, is available on source 0. In order to make certain that reading takes place when a row is in position and not between rows, transfers from source 0, have the GO digit omitted and it is arranged that the Hollerith reader has the same effect as operating the manual switch each time a row comes into position. The passage of a card through the reader is called for by a transfer from any source to destination 31. No transfer of information from the card takes place unless the appropriate instruction using source 0 is obeyed during the passage of the card. Output on the machine is also provided by a Hollerith punch. The passage of a card through the punch is called for by a transfer from any source to destination 30. While a card is passing through the punch a 32 digit number may be punched on each row by a transfer to destination 28. Again synchronisation is ensured by omitting the GO digit in instructions calling for a transfer to destination 28, and arranging that the Hollerith punch effectively operates the manual switch as each row comes into position. The reader feeds cards at the rate of 200 cards per minute and the punch, at the rate of 100 cards per minute. The speed of input for binary digits is 200 x 32 x 12 per minute or 1280 per second. The output speed is 640 digits per second. Data may be fed in and out in decimal, but it then requires conversion subroutines. The computation involved in the conversion is done between the rows of the card and up to 30 decimal digits per card may be translated. This speed of conversion is only possible because of the use of optimum coding. The facility for

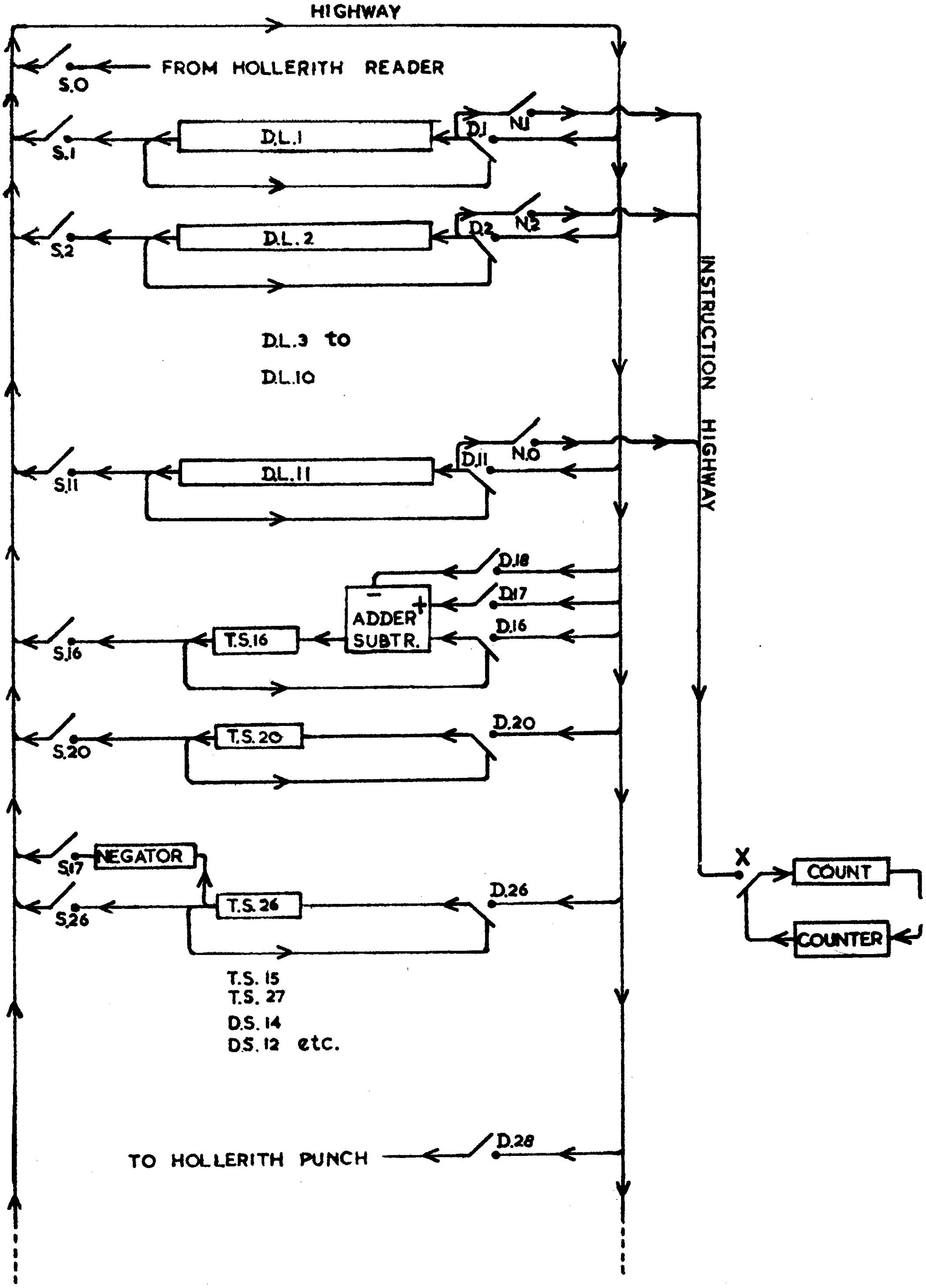


FIG. 1 SIMPLIFIED DIAGRAM SHOWING SOME SOURCES, DESTINATIONS AND NEXT INSTRUCTION SOURCES.

carrying out computation between rows of cards is used extensively particularly in linear algebra when matrices exceeding the storage capacity of the machine are involved. The matrices are stored on cards in binary form with one number on each of the 12 rows of each card, all the computation being done either between rows when reading or when punching. Times comparable with those possible with the matrices stored in the memory are often achieved in this way, when the computation uses a high percentage of the available time between rows. Up to 80% of this time may be safely used.

Initial Input

The initial input of instructions is achieved by choosing destination 0 in a special manner. When a transfer is made to destination 0, then the instruction transferred becomes the next to be obeyed and the next instruction source is ignored. Source 0 has already been chosen specially since it is provided from a row of a card. The instruction consisting of zeros has the effect of injecting the instruction punched on a row of a card into the machine as the next to be obeyed. The machine is started by clearing the store and starting the Hollerith reader which contains cards punched with appropriate instructions. Destination 0 is also used when an instruction is built up in an arithmetic unit ready to be obeyed.

Miscellaneous sources and destinations

Destination 29 controls a buzzer. If a non zero number is transferred to destination 29 the buzzer sounds.

Source 30 is used to indicate when the last row of a card is in position in the reader or punch. This source gives a non-zero number only when a last row is in position. The operation of the arithmetic facilities on DS14 may be modified by a transfer to destination 23. If a transfer with an odd characteristic is made from any source to destination 23 then, from then on, DS14 behaves as though it were two single length accumulators in series. This means that carries are suppressed at the end of each of the single words. This condition persists until a transfer is made to destination 23 using an even characteristic, when DS14 behaves as an accumulator for double length numbers with their least significant parts in even minor cycles and more significant parts in odd minor cycles.

The operation TS20 is modified by transfers to destination 21. If a transfer with an odd characteristic is made to destination 21 then TS20 ceases to have an independent existence and from then on is fed continuously from DL10. Source 20 then gives the contents of DL10 one minor cycle later than from source 10. TS20 reverts to its former condition when a transfer with an even characteristic is made to destination 21. The facility is used to move the 32 words in DL10 round one position so that the word in minor cycle n is available in minor cycle $(n+1)$

Assessment of optimum coding

A detailed assessment of the value of optimum coding is by no means simple. Roughly speaking, subroutines are on an average about 4 or 5 times as fast as on an orthodox machine using the same pulse repitition rate. In main tables a somewhat lower factor is usually achieved. The factor of 4 or 5 would be exceeded if less of the advantage given by optimum coding were used to overcome disadvantages due to the rudimentary nature of the arithmetic facilities on Pilot ACE. Even so, the bald statement of the average ratio of speeds does not do full justice to the value of optimum coding on the Pilot ACE. Its value springs as much from the fact that it has made possible the programmes in which computing is done between the rows of cards and also the high output speed of decimal numbers. The binary decimal conversion routines for punching out several decimal numbers simultaneously on a card and also decimal-binary conversion routines for reading several numbers, achieve a ratio of something like 14 to 1, and on a machine which is being used extensively for scientific computation on a commercial basis this is of immense importance.

Future Programme

Engineered versions of the Pilot Model are now under construction by the English Electric Company. These machines will be similar to the Pilot Model but will have a littl more high-speed store, an automatic divider, two quadruple length stores and a subtractive input on the double length accumulator beside several minor modifications including a rationalization of the numbering of the stores. In addition a magnetic drum intermediate store with the equivalent of 32DL's storage capacity will be added. A full scale machine will probably soon be under development employing a 4 address code. Typical instructions will be of the form

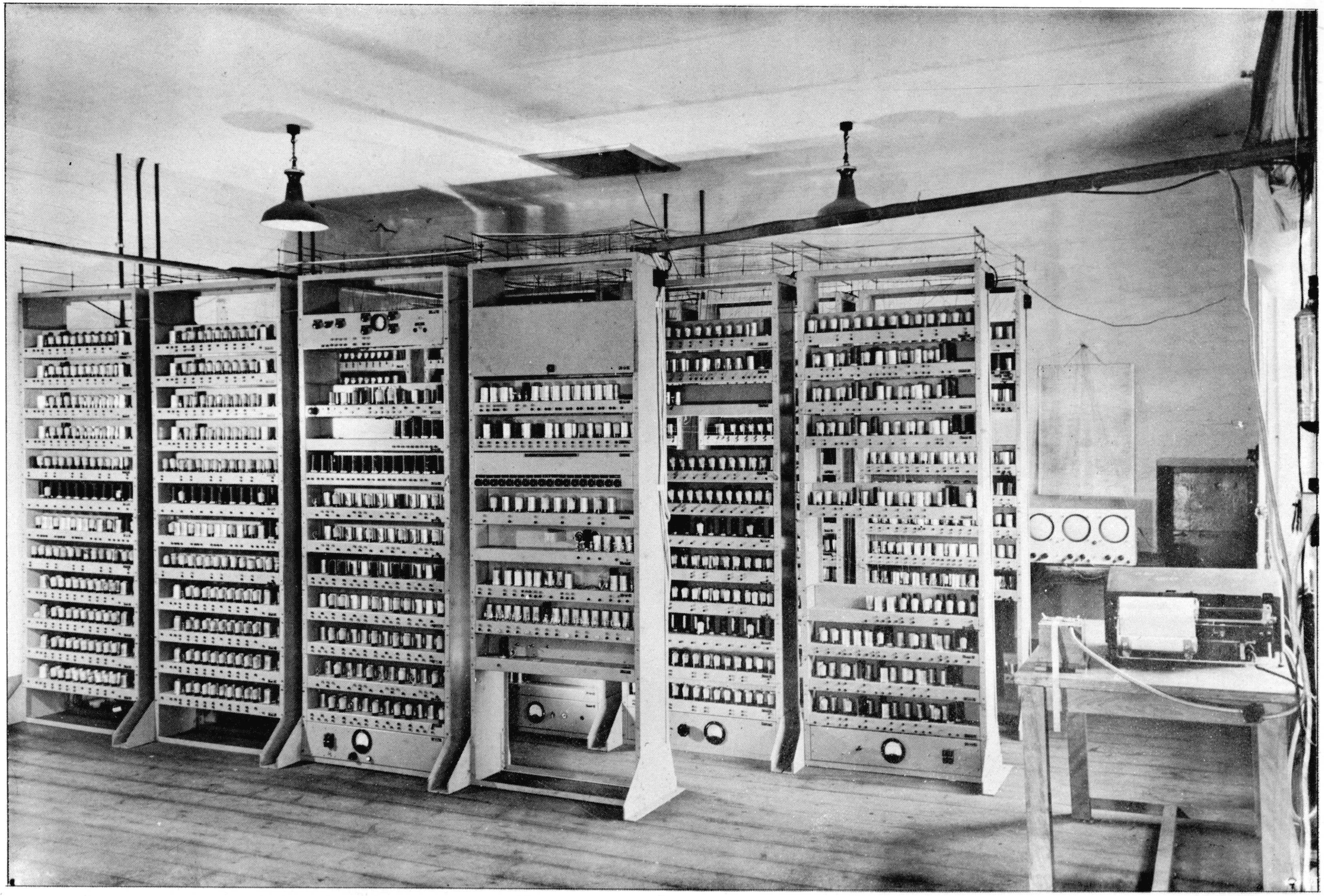
$$A \pm B \quad C$$

and will select the next source of instruction. This code is more economical in instructi storage space and since all single word stores will then become complete accumulators with all facilities except multiplication on them, it will be possible to take much fuller advantage of optimum coding.

SOURCES, DESTINATION AND NEXT INSTRUCTION SOURCES

SOURCES		DESTINATIONS		NEXT INSTR. SOURCE	
0.	Input	0.	INSTRUCTION	0.	DL11
1.	DL1	1.	DL1	1.	DL1
2.	DL2	2.	DL2	2.	DL2
3.	DL3	3.	DL3	3.	DL3
4.	DL4	4.	DL4	4.	DL4
5.	DL5	5.	DL5	5.	DL5
6.	DL6	6.	DL6	6.	DL6
7.	DL7	7.	DL7	7.	DL7
8.	DL8	8.	DL8		
9.	DL9	9.	DL9		
10.	DL10	10.	DL10		
11.	DL11	11.	DL11		
12.	DS12	12.	DS12		
13.	DS14 + 2	13.	DS14 add		
14.	DS14	14.	DS14		
15.	TS15	15.	TS15		
16.	TS16	16.	TS16		
17.	TS26	17.	TS16 add		
18.	TS26 ÷ 2	18.	TS16 subtract		
19.	TS26 x 2	19.*	MULTIPLY		
20.	TS20	20.	TS20		
21.	TS26 & TS27	21.	Modifies Source 20		
22.	TS26 ≢ TS27	22.	-----		
23.	P17	23.	Modifies Source 13, Destination 13		
24.	P32	24.	DISCRIMINATE on sign		
25.	P1	25.	DISCRIMINATE on zero		
26.	TS26	26.	TS26		
27.	TS27	27.	TS27		
28.	Zero	28.	Output		
29.	Ones	29.	BUZZER		
30.	Last row of card	30.*	PUNCH		
31.	----	31.*	READ		

* Independent of source used.



PAPER 2

Fig. 1. The EDSAC

2. The EDSAC

by

M. V. Wilkes

Cambridge University Mathematical Laboratory

I would like to tell you a few things about how the EDSAC project has developed from its inception at the end of 1946 to the present time. The original idea was to build a computing machine for experimental work, and we started off with the idea that we would build something very simple, with no frills; but, as in most projects, we soon got beyond that stage and found ourselves building quite an ambitious machine, It has always been regarded as an experimental machine, both from the point of view of engineering design and of mathematical use. The desire to get a machine on which we could try out programmes, instead of just dreaming them up for an imaginary machine, governed the development throughout, and we probably pushed ahead with the electronic work in a rather bolder manner than we might otherwise have done.

Theoretical proposals had been put forward for a purely binary machine with no divider or any other "special function" units, and there was every reason to suppose that such a machine would be quite usable, but this had to be proved by practical experience. The EDSAC (fig. 1) was therefore designed without any of these features. In particular, it was expected that it would be possible to develop the use of subroutines in such a way that special function units would not be necessary.

The most suitable type of high-speed store from the point of view of easy development was the ultrasonic delay line. At that time Professor Williams' experiments with the electrostatic store had not got as far or were not generally known to have got as far as showing that it was a practicable proposition. We decided to be conservative about the pulse length and chose a pulse interval of 2 microseconds, that is, a repetition rate of half a megacycle. By doing this I believe we made things a little easier for ourselves than if we had chosen a megacycle for the pulse repetition rate.

The mercury tanks that we have used are 5 ft long and are assembled in batteries of 32. The construction is very solid and the design has proved to be very successful except that after a time there is some interaction between the mercury and the rubber washers that are used to seal the ends of the tubes. There is a lot to be said for using a design in which rubber is not used for the seal, as I believe is done in the ACE Pilot Model and in the mercury tanks developed at the Moore School.

The machine did its first calculation in the summer of 1949. There have been a number of modifications since then which I will talk about later. By 1950 we had started operational use of the machine and began the mathematical experimenting I referred to earlier. We made a rule that no alterations would be made to the machine except those which would lead to better serviceability. We did not alter the order code since mathematicians could not be expected to build up a system of programming and a library of subroutines if the order code was likely to be changed.

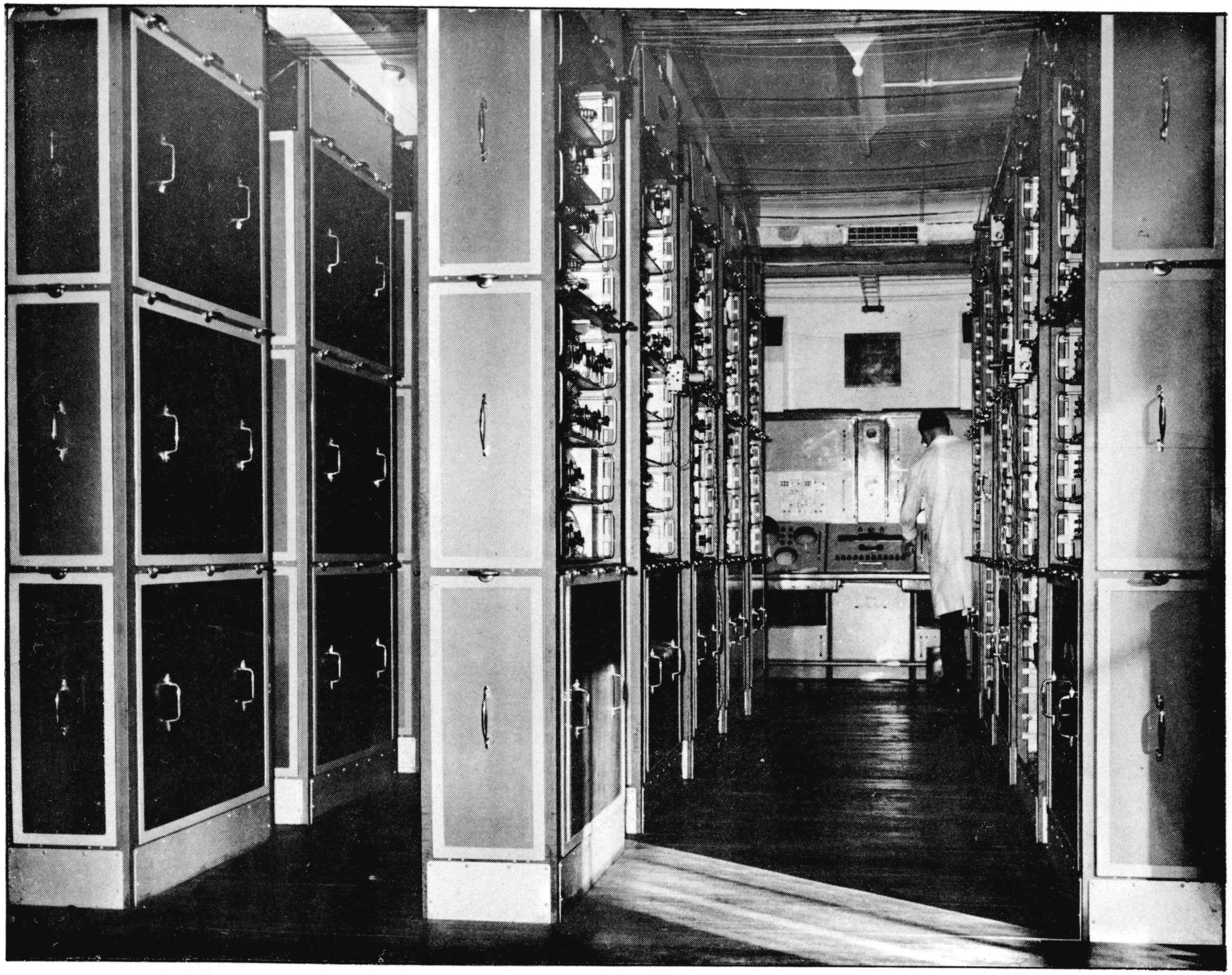
Early in 1950 the original mechanical tape-reader was superseded by a photoelectric one, and this has proved to be a great improvement. We realised then that it is much more important to have a reasonably fast input mechanism in a machine used for mathematical work than a fast output since, for one thing, many programmes get very little further than the tape-reader! During this phase of improvement we gradually eliminated relays - for example, the mechanical tape-reader had relays associated with it for control whereas the photoelectric one does not have any. In our experience, the fewer relays one has the better, unless they are used in a situation where you can repeat what you have just tried to do without much trouble or loss of time. For instance, in an automatic telephone system you can re-dial a number if the call does not connect in the first instance. In a computer, however, intermittent failure due to a relay can be a great nuisance.

It has been a special feature of our development work at Cambridge that we have been particularly interested in equipment for preparing tapes. Now we are moving in the direction of using hard valves rather than relays in equipment intended for this purpose, with the idea of achieving greater reliability. For example, an electronic duplicator has just been put into service. This has two photoelectric tape-readers and there are certain facilities for switching from one tape-reader to the other so that two tapes can be combined. It is too early yet to say whether, in this instance, purely electronic techniques will justify our confidence in them but it seems to me that there is everything to be said for using hard valve circuits rather than relays wherever one can.

In 1951 we shut the machine down for some time and rebuilt a good proportion of it, incorporating improvements based on experience up to that time. In a serial machine it is often necessary to delay the pulses slightly, amplify them, and use them to gate clock pulses. In all these cases we introduced regenerative feed back to hold up the output of the amplifier until the end of the clock pulse. A similar technique is used in the SEAC in Washington. About the same time we also introduced a logical checking output code. Originally we had a direct teleprinter fitted with checking circuits which could read back into the machine a character that had just been passed to the teleprinter. These checking circuits caused a lot of trouble. With this new system, in which each character has two 1's and three 0's, we could use an output punch instead of a direct teleprinter without any uneasy feelings about accuracy. This was a great step forward since the output punch is about three times as fast as a direct teleprinter. There was some discussion before we put this system on as to whether it would be as useful to the operators of the machine as a direct teleprinter. The doubt arose because of the delay between the results coming from the machine and the actual printing of these results. The punch produces a tape, which is then passed to a tape-reader (after a sufficient loop of tape has accumulated between the two units) before it is printed by the teleprinter. At first the teleprinter was left on the machine as an alternative to the punch but after a time we discovered that it was not being used at all and it has now been removed from the machine.

For the past year we have been doing a certain amount of engineering development work on the machine as well as using it for mathematical work. Amongst other things we added some new orders. Previously we could only transfer control according as to whether the accumulator was positive or negative; now we can transfer control if the contents are zero. We can also arrange that when control is transferred the accumulator is cleared. There is now an absolute transfer of control and we have provided what we call a "mix" order which enables the function part of an order to be altered by direct substitution, without altering the address. Previously if we wanted to alter the function part of an order it had to be done by adding something to it, and it was necessary to work with the binary equivalents of the various function letters. These are all only small conveniences, but well worth having.

Another development was marginal checking, but this will be discussed more fully later in the Symposium. Recently we have started experiments with magnetic tape equipment and we have had an auxiliary magnetic tape store working with the machine. This is now being "cleaned up" and should be fully operational quite soon. We hope to report on its operation later.



PAPER 3

Fig. 1. LEO: General view of the Calculator Section showing three of the four rows of racks and in the background the control desk with oscilloscopes and pushbuttons for control.

3. Operating and Engineering Experience Gained with LEO

by

J. M. M. Pinkerton

J. Lyons & Company Limited

Introduction

This paper is not intended to give a full description of all the equipment at Cadby Hall, since much of it is a development of equipment at Cambridge which has been described elsewhere *(ref. 1)*. It may, however, be of some assistance to discuss some of the special features of LEO and our experience in using them.

In setting out to build a calculator for Lyons the intention was to get equipment into operation as quickly as possible, because it was felt that until equipment of some kind had actually been used over a period for clerical purposes, the optimum form of such equipment could not be decided. To expedite the construction it was decided in the main to copy the logical design of an existing machine. At the time the EDSAC seemed the most suitable, and much friendly assistance was given to us by Mr. M. V. Wilkes, Director of the Mathematical Laboratory at Cambridge. Using the circuit designs of the EDSAC, with many revisions in detail where necessary, we built the calculator which is the basis of the LEO system (fig. 1). It was realised at the start that to make the installation effective on clerical work additional features would be required, including a larger store, means for introducing data into and extracting results from the calculator much faster than was possible with the EDSAC, and a foolproof method of checking data recorded for the machine.

At the outset it was decided that Lyons should themselves undertake the design and construction of the calculator similar to the EDSAC forming the heart of the installation and, shortly afterwards, that the development and construction of auxiliary equipment for handling data and results should be done by Standard Telephones and Cables Limited. The requirements for this equipment were outlined by Lyons and in subsequent discussions with Standard Telephones & Cables Ltd. were formulated in detail. Construction started in the summer of 1949 and the calculator itself has now been in use for rather more than eighteen months; prototypes of each of the different items of auxiliary equipment are installed and on operational trial. The installation as a whole is, however, not yet as reliable as would be necessary for carrying out a regular and intensive programme of clerical work.

It was recognised that equipment to be used to do the work of a large office would have to be at least as reliable as that used by a public utility such as an Electricity Board. Clerical work on a large scale is normally undertaken to a very tight schedule; it is not unusual for the papers to pass through the office in a few hours so that goods can be despatched on the same day as the order is received from the customer. Obviously therefore any calculator used on this kind of work must be extremely dependable and moreover, when breakdowns occur, the trouble must be put right in minutes rather than hours.

ENGINEERING FEATURES

Logical Design of the Calculator

This is extremely similar to that of the EDSAC and the order code is in fact nearly identical. All the orders provided in the EDSAC are available in LEO and have the same code, with the exception that testing for zero accumulator in LEO is given the code letter B instead of F as in EDSAC. Numerous modifications to the logical structure of the units have been made from time to time where this seemed likely to improve the performance or reliability. In addition to the EDSAC orders there are further orders concerned with high speed methods of feeding and extracting data, and with the use of alternative slow speed inputs and outputs.

Control

The operational control of the machine is generally similar to that of the EDSAC, with buttons for Start, Halt and Restart. Push-buttons are provided for inserting numbers under manual control directly into the store. These are used in conjunction with test programmes and for fault finding should the machine be unable to read the programme tapes. There are also push-buttons for Single Step and Slow-Speed operations. Three alternative slow speeds are provided by a ticking relay circuit. Several monitoring oscilloscopes are grouped in a control desk which also houses the starter unit which puts the initial orders into store. The monitor displaying the store *(ref. 2)* is very convenient in use, numbers being depicted on a raster display with vertical strokes for the units and dots for the zero digits. An electronic switching arrangement is provided so that the contents of any one of the 64 storage tubes can be examined at will. Attachments have recently been made to allow a Cossor camera to be quickly attached for photographing the contents of the store.

An unusual feature, which has been found valuable in fault finding, is a loudspeaker connected to a waveform in the central control circuits of the machine. This loudspeaker makes a noise depending on the sequence of orders being carried out and every large programme has its own characteristic rhythm; should the machine stop or go permanently into a closed loop the fact is instantly apparent. In testing the machine on simple repetitive programmes, a single failure is easily detected as a break or click in the continuous tone.

Storage system

The Store is identical in principle to that of the EDSAC but one or two improvements of detail may be mentioned. The frequency of the clock pulses is regulated by an automatic control system which has been described elsewhere *(ref. 3)*. This uses one delay tube as a reference standard to maintain the frequency at a value which should suit all the others provided they are maintained at the same temperature. A preset control enables the frequency to be varied and the variation which can be tolerated by the store as a whole to be measured in arbitrary units. This is used as a form of marginal check on the store. Sixty-four tubes are in regular use giving a capacity exactly double that of the EDSAC; there are in fact five batteries each of 16 tubes, making a total of 80, in the constant temperature enclosure so that a number of spare tubes are available in case of breakdown

About eight months ago it was found that some storage tubes had gradually deteriorated in performance; the main noticeable change was a considerable reduction in the bandwidth. This naturally had the effect of delaying the pulses in some tubes more than in others so that it became difficult or impossible to choose a clock pulse frequency to suit them all. The trouble was found to be due to a slow loss of the alcohol used for wetting the quartz and mercury surfaces, either by evaporation or diffusion through the rubber washers, so that the acoustic pulses no longer passed freely from the crystal to the mercury or vice versa. A much better wetting agent has been found in a mixture of five parts of glycerine and two parts of water. Glycerine is a very efficient matching agent between the quartz and mercury as it has an extremely high acoustic impedance (ρc), 2.5 times that of ethyl alcohol. The water is added to make the mixture less viscous. One per cent of sodium benzoate is also added to the solution as a rust preventative. Since this technique of filling the tubes has been adopted none has deteriorated through a loss of bandwidth. In fact the tendency seems to be for the overall attenuation to fall and the main problem now is with the reflected pulses which have travelled three times down the five foot column of mercury.

Slow speed Input/Output Facilities

The input and output facilities of LEO may be divided into two categories the slow-speed channels, which are broadly similar to those used at Cambridge but with certain modifications, and the high-speed channels which are peculiar to LEO and, as already mentioned, have been developed by Standard Telephones and Cables Ltd; the latter are dealt with later in this paper.

On the slow-speed input side there are two readers, one photoelectric reader and one mechanical, both using five-hole tape. There is a changeover instruction in the order code which has the effect of operating relays, for changing the connexions from one reader to the other. All subsequent input orders then operate on the alternate reader until a further changeover instruction is carried out. The photoelectric reader is extremely similar to those in use at Cambridge and so need not be described in detail. It has proved to be quite reliable now that the original design difficulties have been overcome. Its operation is fully mechanically and electrically interlocked so that in proper adjustment it is impossible to read twice without moving on the tape.

On the slow-speed output side there are two instructions in the order code, one working a teleprinter using the five-wire system as originally employed with the EDSAC. The other output order, known as the Perforate order, operates either a standard telegraph reperforator or a standard teleprinter, or both in parallel. Both these output channels operate at seven characters per second. In the Perforate channel, telegraph characters are formed by a special circuit, designed by Standard Telephones & Cables Ltd., using gas trigger tubes and multicathode tubes *(ref. 4)*. This circuit, introduced partly as a check on the performance of the gas tubes, gave a good deal of trouble at first but latterly it has proved very reliable.

We have not introduced into LEO the "inhibition" scheme in use at Cambridge by which, after a mechanically executed order, the calculator is allowed to carry on calculating before the mechanical action is completed, but if a second mechanically executed order appears in the order register before the first one has been completed it is inhibited.

It is worth noting in connexion with all mechanically executed orders that the most serious difficulty arises with the provision of the so-called "end-pulse" indicating that the instruction has been completed. It appears to be quite difficult to devise a scheme for linking a mechanical contact to electronic circuits in such a way as to produce one pulse, but only one pulse, when the contacts exhibit bounce as they usually do.

Marginal check facilities

Following the experience gained at Cambridge and elsewhere, marginal testing facilities are being introduced gradually throughout the whole of the calculator and its auxiliary equipment. They have already proved themselves to be of great value. The ways in which they are used are discussed further on under the headings "Methods of maintenance" and "Experience from running programmes". Nearly all the marginal test circuits employ 50-cycle alternating voltages injected in series with some D.C. bias. This has the effect of raising and lowering the threshold over which a pulse must climb before it can be retransmitted to a subsequent circuit. Thus if a pulse in a circuit, following a marginal check point, is already very near the lower limit of functioning it will be modulated above and below the failure limit at 50 cycles. The normal practice is to operate the entire machine on a suitable test programme with marginal voltages applied at a large number of points and to increase the voltage gradually by means of a Variac until failure occurs. Marginal tests are being regularly applied to all storage units, either collectively or in groups as required, and to the majority of pulse amplifiers, particularly in the computing circuits. They are also applied to all decoding circuits. Ultimately it is intended to apply marginal test voltages to all trigger circuits in addition, in such a way as to vary their trigger sensitivity. When this has been done we believe that we shall be able to detect drifts in the performance of all the circuits.

Marginal tests are also applied to the photoelectric reader simply by increasing and decreasing the voltage applied to the lamp filament by about 10%. This is an extremely effective test.

Constructional features

In its construction LEO differs more widely from EDSAC than in its logical design. It was considered vital to be able to replace units as easily as was consistent with reliable electrical connexions. Units slide into tall racks holding up to twelve units and are locked in position by hinged screws fitted with knurled finger nuts. Pulse connexions are made by flexible leads fitted with spade terminals which come from a gantry on the unit mounted opposite a tag strip fixed to the rack. It is arranged that each pulse lead comes exactly opposite to the tag to which it corresponds, and there is thus little risk of error in connecting up the pulse leads. At the same time it is easily possible to introduce an attenuator in any pulse lead for special test purposes. Power is fed into each unit on eight-pin miniature Jones plugs, which have been found entirely satisfactory. Every unit carries its own filament transformer which is individually fused; each valve is connected by a short lead to a common filament busbar along the centre of the chassis. There is a fuse in the main H.T. supply to each unit with a small neon indicator in parallel which lights if the fuse blows. Thus short-circuits can readily be tracked down to the unit concerned. The racks themselves, which are of substantial construction, form an integral part of the ventilation system. With the removable sheet metal covers in position each rack forms a vertical duct. Cold air is blown in at the bottom from a common duct linking together all the racks in each row and the warm air is extracted at the top through another common duct. Filtered cold air is blown into the machine by a large fan and a second large fan extracts the warm air and passes it back to the outside. The room itself is ventilated with filtered air by a third fan which maintains a slight excess pressure in the room at all times and distributes fresh air to various parts through ducting. By having all windows permanently closed it has been found possible to reduce greatly the amount of dust settling on the apparatus. This is particularly helpful because of the many relays and switches used in the auxiliary equipment. The calculator proper consists of twenty racks of equipment housing approximately 200 panels, each carrying twenty-five or more valves, the total number of valves being about 6000. In addition to these racks there is the control desk already mentioned.

Power Supply

This has been the subject of a number of alterations since the equipment was first constructed, so the present arrangement will be described. The power for the valve filaments is taken from the supply mains via an oil-filled regulating transformer which is similar in principle to a Variac. When the valves are switched on the voltage applied to the primary of the heater transformers is raised gradually from 0 to 230 over a period of two minutes by an automatic control. Subsequently the voltage is maintained at 230 within 1% by the control circuit. The high tension supplies are four in number and are rectified by four selenium rectifier sets supplied by Standard Telephones and Cables Ltd. The largest of these is a three-phase rectifier with a maximum output of 30 amps at 259 volts. To prevent surges on the public supply from affecting the calculator a voltage controlled alternator of 12 kilowatts is installed in the basement driven by an ordinary motor from the supply mains. Thus unless the surge is of unusual duration or intensity, it will be smoothed out by the inertia of the rotating machinery. A duplicate alternator set is available in case of breakddwn of the first. In three and a half years' of operation no power cuts have been experienced, but Diesel generators are available on the site and these could be used to supply the power if necessary.

HIGH SPEED INPUT AND OUTPUT ARRANGEMENTS

Data recording

For feeding data into the calculator at high speed a recording medium is required which can be used to effect a change of speed from that at which data can be recorded by an operator from a keyboard and that at which it must be read by the calculator if the latter is not to be kept waiting. Magnetic tape on which characters are recorded in teleprinter code meets this need. A suitable machine for making such records was under development by Standard Telephones and Cables Ltd., when we first approached them to produce input and output equipment for use with LEO. An endless loop of tape is employed which is held in a narrow vertical box from which it is drawn independently past reading and recording heads; after passing each head the tape passes back into the box and is allowed to fall under gravity.

A standard transmitting teleprinter is used to transmit signals representing data to the recording head. For each character sent, a small length of tape is drawn past the head and falls into the box forming an inner loop carrying the recorded data. This inner loop lies between the recording and the reading heads; during reading out, tape is drawn from this loop past the reading head and then again falls back into the box to form an outer loop. Characters recorded on this outer loop of tape are automatically erased as the tape passes the recording head. It will be seen that no rewinding is needed. Because only a small amount of tape has to be accelerated at any time the tape can be started from rest very rapidly. Signals from the reading head may be fed after amplification to a teleprinter, to the calculator, or to other electronic equipment for checking purposes.

Data checking

The greatest possible importance is attached to preparing clerical data free from recording errors. Errors affecting the numerical value of the data are serious enough but those which would cause the calculator to misinterpret the form of the data, for instance by taking a man's wage rate to be the number of hours worked, could cause havoc in carrying out a job. It was felt to be essential, therefore, to have special apparatus for checking the original recording character by character. A second operator working from the same original documents transmits the information from a teleprinter to the checking apparatus. As each character is received by this checking apparatus it causes the next character on the tape to pass the reading head of the tape machine. This character is compared in the checking apparatus with that sent by the checker and if the two agree the character is re-recorded by the checking apparatus, which is connected also to the recording head of the same tape machine. If the character sent by the checker disagrees with that read from the tape a special correction procedure is carried out. As checking proceeds the original message is overwritten by the checked version which follows it on the tape. The checker's teleprinter is made to produce a proof sheet of what is recorded and of corrections made to the original recording. This enables a re-checker to make sure that all corrections were in fact made correctly.

Because the tape drives on the tape machine are independent it is possible for characters to be removed altogether from the original message by reading them from the reading head without re-recording them. Likewise, additional characters can be introduced by moving the tape under the recording head only. Facilities have been provided for both inserting and removing information in this way.

Input to calculator

Once a box of tape has been filled with checked information it can be taken off the standard speed tape machine, and fitted to another one having high speed continuous drive, so that the information passes into the calculator at many times the speed of the teleprinter. The information read from the tape is passed through electronic equipment which is designed to assemble and prepare the data in the form required by the calculator. This equipment is the subject of patent applications which have so far not been published and for this reason further information cannot at the moment be given.

There are in fact a pair of tape machines associated with each input channel, the first box of data being read from the first tape machine. When all the information has been read from this box the reading then commences from the second box on the second tape machine. In the meantime the first box is taken off the first tape machine and the third put in its place, and so on.

As an alternative to these input channels we have recently experimentally coupled to LEO a high speed photoelectric punched tape reader made by Ferranti Ltd. This appears to work very satisfactorily at about 120 rows per second, starting and stopping on every row. The amount of operational use has so far been so small that we cannot at the moment report on the performance of this method of input.

Output

The arrangements for taking information out of the calculator are complementary to those for putting it in. Results are put out in the form which the calculator finds most convenient and are suitably arranged according to the form required by further electronic circuits which are more elaborate than those provided on the input side. They are then recorded at very high speed on magnetic tape. This recording includes all the punctuation and interpretation characters necessary to cause a teleprinter to print the results in the desired layout. A pair of tape machines is used so that, when the tape box on the first machine is completely full, recording can automatically start on the second while the box on the first machine is replaced by an empty one.

Information recorded on the tapes can, if required, be fed back into the calculator for a subsequent job, or if the results are required in printed form the boxes can be fitted to tape machines and fed to a battery of receiving teleprinters.

Printing Results

When the calculator is working continually on clerical results it will be able to keep several teleprinters, possibly as many as 20, busy printing the results recorded through one of the output channels. Many of these results will be required on preprinted forms so that some printers will be fitted with a sprocket-feed carriage to ensure exact alignment.

Alternative high-speed input and output channels

It is fully recognised that the method described using magnetic tapes for input and output is only one approach to the problem. It has the disadvantage that a lot of work is involved in handling the boxes of tape. Another disadvantage is that results produced are not available for scrutiny as they are produced by the calculator. No doubt as faster printers and other methods of recording are developed these objections will be overcome. It should be made clear, however, that the basic principles of the system we are using can be applied also to other means of reading and recording information.

GENERAL EXPERIENCE FROM OPERATION

Methods of maintenance

One of the large unanswered questions about operating calculators at present is to what extent their intrinsic unreliability can be countered by efficient servicing. This question is perhaps the most important to be answered about the engineering aspect of LEO. Maintenance can be considered under two headings:

(i) preventive maintenance designed to stop faults from occurring whilst the machine is in operation, and

(ii) curative maintenance, designed to reduce the amount of time spent on finding and correcting faults.

There is nothing novel about the measures being used for these purposes on LEO. We hope we are applying well known methods efficiently but we are by no means satisfied yet with the results.

Preventive Maintenance

The preventive maintenance programme is intended to eliminate faults caused by dry joints, gradual deterioration in valve performance, and failures in the mechanical equipment of the system. Methods of dealing with valves are described in the next section. All mechanical equipment is periodically cleaned and oiled and checked for wear or wrong adjustment. An unfortunate feature of the maintenance of mechanical parts of electronic calculators is that in spite of all precautions items which have recently been serviced tend to cause faults due to slightly wrong adjustment or other human error made during the servicing.

failures shown as catastrophic caused an immediate breakdown of the calculator. In order to assess how many failures caused a machine breakdown, the proportion of failures found on routine overhaul has been shown in the bottom of the table. Out of 893 rejections, 514 occurred on routine overhaul. A further point is that 326 failures of 6D2 valves were thought to be due to a form of cathode poisoning from which the valves may be recoverable by suitable processing. If we deduct from 893 the diodes failing in this way and also the 43 SP61 and 6F32 valves accidentally damaged we obtain a figure of 524 rejections. Of these it can be assumed that approximately 40% would have resulted from a failure in the machine rather than as a result of routine tests, that is to say about 200 valves have probably caused breakdowns in approximately 6000 hours running time.

Conclusions

The conclusions tentatively drawn from the above figures are that it pays to remove valves and test them at reasonably frequent intervals but that this cannot reduce breakdowns due to valve failure to zero. This can only be achieved by an improvement in the quality of the valves used. Nevertheless, this quality does seem to be higher than was first supposed. At a later date it will be possible and worthwhile to plot survivor diagrams for suitable batches of valves of various types, so as to determine whether there is any advantage in replacing valves which have reached a certain age whether or not they pass the acceptance test for used valves. It is felt that insufficient data has yet been accumulated to make it worthwhile to plot such diagrams.

Experience from running programmes

For the past fifteen months the machine has been used for part of its time on calculations of a technical and scientific nature, carried out for outside organizations. These include calculations of shell trajectories, crystallographic structure, work in connexion with Meteorological forecasting, and calculations for actuarial purposes. In addition certain regular clerical work has been carried out week by week. All these jobs have had to use a slow-speed output system until recently because the other equipment was not ready to be used. A great deal of valuable experience has been gained which has emphasized the vital importance of checks and tests at every stage to pick out both human errors and faults in the machine itself.

Every step from the construction of the programme to the scrutiny of the final results is subject to its own particular check, so that work should proceed no further than the point at which the error has occurred. The programme is prepared in sections, each of which can be independent since the initial orders used with the EDSAC (and with LEO) *(ref. 5)* allow the address to be written with reference to its place in the subroutine or stage. Each section is, whenever possible, tried independently, for it has been found that a programme of any size is hardly ever devoid of flaws as first written. These flaws are very much easier to detect when only a part of the programme is being tested.

When feeding the programme into the machine a check is made to see that it has been correctly read and stacked; this is done by adding together the digits of the addresses and instruction numbers to give a check total for comparison with the value punched at the end of the programme tape. A disagreement may signify either that the programme has been incorrectly perforated or that the machine has read it incorrectly through the reader. In either case time is saved by not going any further until the fault is discovered.

Likewise the most stringent checks are introduced into the data. Where this is recorded by hand the usual practice is to perforate the two tapes independently and compare them visually, or to use the collating equipment with the magnetic recording machine mentioned in paragraph 3. Carried forward data produced by the machine at an earlier stage or during an earlier repetition of the same job in clerical work can very conveniently be checked by producing a check total in the machine which is punched or otherwise recorded at the end of the tape to be carried forward. In carrying out the subsequent job the check total is again accumulated in the machine and compared with that recorded on the tape at the end of the job or stage of the job. Again when data is being taken in through several channels, code numbers are used so that a check can be made that corresponding data is being read at the same time through all channels.

Further checks are used in the actual execution of the programme, for example, to see that the magnitudes of particular quantities do not exceed predetermined limits or to test their internal self consistency, e.g. $\mathrm{Sin}^2\theta + \mathrm{Cos}^2\theta = 1$. On scientific problems where the volume of fresh data is small a check total may be perforated at the end of the data tape, but in clerical work this would be too laborious in most cases.

As a further safeguard on the accuracy of calculations, the practice has been adopted of working out results from test data using the same programme before and after or during the course of the job itself. It is now usual to produce these test results with marginal voltages applied to numerous parts of the circuits. Proceeding in this way it is extremely rare for answers to be produced which are initially thought to be correct and are subsequently shown to contain an error.

Finally it is worth stressing the great advantage to be gained from the use of a monitoring oscilloscope to view the contents of a selected storage tube. Programme faults may frequently be detected quickly in this way, and a running supervision of a calculation can also be effected very conveniently especially if, as is usually convenient, all the varying quantities are stacked in one or two tubes. The camera can often be used to photograph selected parts of the store and thus allow the contents held in it to be studied at leisure, with a minimum of expenditure of valuable machine time. The cost of the film used is very small.

Conclusions

The conclusions to be drawn from the work done so far must necessarily be only tentative. A large amount of information has been recorded about the day to day performance of the machine and its auxiliary equipment but very little analysis of this has so far been attempted. However, certain things do appear to stand out:

1. There is no doubt that if the machine is working properly it will do what we require.
2. The operation and maintenance of calculators used for clerical work must be made more foolproof than it is, before the best service can be obtained from them; even in their present unreliable condition a high proportion of failures is due to some human omission or injudicious intervention.
3. The design should be conservative and straightforward; the control circuits particularly should be as simple as possible and sequential in operation, and the number of valves in the equipment should be reduced to the absolute minimum.
4. To simplify maintenance work and reduce time lost through faults to a minimum all parts should be easily replaceable and adequate spares for them should be provided. The mechanical parts are the most important in this respect. Spares for the electronic panels should be physically and electrically identical with their counterparts.

References

1. WILKES, M. V. and RENWICK, W. *The EDSAC – Report of Cambridge Conference*, July 1949, 9 (and further references there quoted).
2. LENAERTS, E. H. *Electron. Engng*, 1951. 23, 140.
3. SHARPLESS, T. K. *Proceedings of Harvard Symposium, January 1947*, Annals of Computation Lab. 1948, 16, 108.
   PINKERTON, J. M. M. *Electron. Engng.* 1951 23, 142.
4. HOUGH, G. W. and RIDLER, D. S. *Electron. Engng.* 1952, 24, 152, 230, 272.
5. WHEELER, D. J. *Proc. Roy. Soc., A*, 1950, 202, 573.

TABLE I

Valve failure analysis

This analysis is compiled from data obtained up to 6th February 1953.

Hours worked to 6th February 1953: 6545 hours

Total number of valves in machine			Total number of failures	Percentage
ECC 33	1431	(Double Triode)	160	11.2
KT 61	340	(Pentode)	34	10.0
SP 61	1008	(Pentode)	193*	19.1
EF 50	250	(Pentode)	93	37.2
EF 55	113	(Pentode)	13	11.5
6F32	36	(Pentode)	9	25.0
6D2	2144	(Diode)	381	17.8
EB 91**	614	(Diode)	10	1.6
Totals	5936		893	Average 15.0

Date	Hours run	No. of Valves in use at end of period	Gross Total of failures to date	Mean rejection rate per 1000 per 1000 hours
1.1.51.	1136	2749	52	16.6
21.5.52.	4400	5033	369	16.7
19.9.52.	5380	5562	729	24.3
6.2.53.	6545	5936	893	23.0

* Includes 41 accidentally damaged.

** The low proportion of failures of this type is due to the fact that it has not been long in use.

TABLE II

Analysis of Valves from Machine rejected on test

	Gradual			Catastrophic					Accidental	Miscellaneous			
Type	Emission Failure	Gassy	Total	Inter Elect. Ins.	Internal o/c **	Envelope Cracked	Heater o/c	Total	Top Cap Broken	Pins Loose, etc.	Other Causes	Total	Grand Total
ECC 33	63	12	75	63	9	3	2	77	-	2	6	8	160
EB 91	4	1	5	1	2	-	-	3	-	2	-	2	10
EF 50	68	-	68	10	11	-	1	22	-	1	2	3	93
EF 55	4	-	4	5	-	-	1	6	-	-	3	3	13
KT 61	13	3	16	12	1	-	2	15	-	-	3	3	34
SP 61	36	16	52	84	4	4	2	94	41	2	4	6	193
6D2	326	4	330	31	3	5	6	45	-	-	6	6	381
6F32	3	-	3	4	-	-	-	4	2	-	-	-	9
Totals	517	36		210	30	12	14		43	7	24		
			553					266				31	
% of Grand Total	57.8	4.0		23.4	3.3	1.4	1.6			1.1	2.6		
			61.8					29.7	4.8			6.5	893
Proportion of failures on overhaul			380					110	5			19	514

NOTE * This includes low slope or total emission and high conducting resistance for diodes.

** This accounts for zero emission

PAPER 4

Fig. 1. MADAM.

4. Madam

by

Professor F. C. Williams

Manchester University

By stupendous efforts we did just succeed in producing for the opening of this conference the report of the Manchester conference in July 1951 so I shall assume that I need say nothing about the Manchester machine before that date and that I can take as my subject a summary of the experience that we have had with this machine in the 20 months that have elapsed since then. Throughout this period this machine *(Fig.* 1) has been used purely as a computing instrument in a mathematical organization, and has been kept quite separate from the electrical engineering department, and no engineering research has been undertaken on the machine. The policy of the mathematical organization has been to investigate the method of using computing machines and to investigate their application to a reasonably wide range of problems. We have never attempted to set up a computing service for the solution of routine problems by means of our own computing machine staff. As part of this policy we have devoted a good deal of time to the training of users drawn from as wide a field as possible, including other departments in our own and other Universities, Government Departments, and such industrial organizations as we have been able to interest. Thus the emphasis has been rather on the training of users and the dissemination of machine "know-how" rather than on producing specifically interesting numerical results. It is our intention to leave the production of such results very largely in the hands of newly trained users, who in general have undergone training with the object of being able to do that kind of production work.

In parallel with the policy decided upon for the use of the machine, we have been keeping a very close eye on the machine from the engineering point of view, and it is important to realize that we have approached this from an operational-research aspect. We have not been doing engineering research on the machine, - we have been watching the behaviour of the existing machine, from an engineering point of view. In this connexion it is important to remember, as I have already said, that the machine has been severed completely from its parent department, and that its servicing is in fact, done not by people who assisted in making it, but by people who have been trained for that work. In much the same way as radar mechanics, during the war, took a course on a particular radar device, so our service engineers have taken a course on the machine and then attempt to service it. Now I am not suggesting that these specially trained service people do not get assistance from more knowledgeable people when they are in trouble, - there are a variety of people to whom they can apply for help - but in the main they do manage to keep the machine going most of the time. The other thing that I want to mention about this machine is that it is not a specially built laboratory model. It was not even built in the University. It was delivered there and is a commercially made article. As a result of this experience with the machine it has become quite clear that, apart from a lot of little mistakes, we made two major mistakes in the original design. In spite of these two mistakes we have managed to limp along at an average rate of computation of about 70 hours a week during the period under review, but these two mistakes did become very clear during this period.

The first mistake was that we made our magnetic drum of too small a diameter. As just looked at, on the oscillograph, it seemed to work very well, but in the computing machine there were errors of the order of a few wrong digits in a million. This was not good enough. We have now replaced the old drum by a drum of double the diameter with, therefore, a more open packing for digits, and trouble from this source has practically vanished.

The second rather serious mistake that was made in the design was the decision to use D.C. machinery for the high tension supplies to the machine. We find that these machines are rather like the curly headed girl, when they are good they are very very good but when they are bad they are quite horrid. The so-called D.C. can, under such certain circumstances, be accompanied by every variety of high frequency mush that you can imagine and this is developed at such a low impedance, by brush arcing and so on, as to make it practically impossible to eliminate it by filter circuits. The result is that from time to time the machine has just become intolerable in this respect and it has been necessary to dismantle the machines, skim the commutators and generally attend to them mechanically. This is always a long job and it closes down the machine for rather a long time. We have done nothing more about this yet, but we believe that the preferred arrangement is to use a motor alternator, to isolate the machine from the mains and its attendant switching transients, and then subsequently to use some form of dry plate metal rectifiers to look after the high tension supplies.

Throughout this period too, we have been keeping a very careful watch on the behaviour of the cathode ray tube store. This has given considerably less trouble than either the drum or the power supplies. In fact if we had not had any trouble with our power supplies the faults due to storage itself would have been negligible. Most of the faults we have had with the store have been due to machine interference getting through rather sensitive time base circuits to the deflection circuits of the cathode ray tubes. The time base has been improved recently, and that has given some considerable assistance, but the real solution is to root out the source of the trouble and get rid of the D.C. machines.

Now before I give you detailed figures of the performance over this period, I think it would be as well to outline the present normal running routine of the machine. The normal process is to switch the machine on at 8.30 on a Monday morning, and it then runs night and day until the following Friday week, when it is nominally switched off at 10 p.m. in order to give the maintenance men one free weekend in two. Frequently however the machine runs through this second weekend as well, without a maintenance engineer in attendance but with one at the other end of a telephone with a fairly powerful car at his disposal. Each day there are two periods set aside for maintenance. The first is from 8.30 until 11, during which preventive measures are taken and general investigations made, and another from 3 to 4 in the afternoon. The first period is always used. The second is used if circumstances suggest it would be a good idea; otherwise the calculations proceed right through it. Thus, in any period of 14 days the machine is actually switched on for a period of 277½ hours, of which 38½ hours, or something like 14%, are set aside for maintenance. We have kept continuous records, excluding all holidays, but including involuntary closing down of the machine. We have recorded under a number of headings the total time the machine was on, the total scheduled maintenance and engineering time, the scheduled computing time, and the fault time. The figures are shown in Table 1. The total "on time" is about 8000 hours, scheduled engineering and maintenance 1300, scheduled computing time 6500 odd, fault time 860 hours. The serviceability can be expressed either as 87% of the scheduled computing time, or as 72% of the total power-on time. I think those are fairly satisfactory figures, bearing in mind the nature of the maintenance effort that we have used, and also the fact that very little of the machine's time is employed entirely on production. If the machine is running on pure production then a fault is found just about as soon as it happens and can be relatively quickly traced. If the machine is being used for training when it may be idle for lengthy periods, or for programme preparation and correction, then it is possible for minor faults to accumulate until you have several, before you spot any of them, and this always makes life rather more difficult.

TABLE I

Serviceability

Total "on" time	7792 hours
Scheduled maintenance and engineering time	1322 hours
Scheduled computing time	6470 hours
Fault time	860 hours
Percentage serviceability in terms of scheduled computing time	87%
Percentage serviceability in terms of total "on" time	72%

We have also kept records of all component failures during the period under review. These are shown in Table II. The 357 valve failures involved 200 pentodes, 108 diodes, and 49 other types. There were 30 cathode ray tube failures, 10 condenser failures and 24 resistor failures. We arrive therefore at an average valve failure rate of approximately 10 valves per mega-valve-hour, or an average life of 88 000 hours per valve - which is a reasonably satisfactory figure. The rate for the cathode ray tubes is less good, and it is of course pointless to express the answers for condensers and resistors in that way because the valves already are showing about 10 years life, so the corresponding figures for the condensers and resistors will take us well into the next century. It seems from these figures that our policy of running the machine fairly cool and not running the valves right up to their rated level has paid off reasonably well. Turning to other aspects of the machine, the performance figures showed that there had been about 5600 hours available for computation. During this period about 50 different people from various sources have been trained in the use of the machine and our experience with these trainees shows that it is definitely possible to have a flair for this kind of work, and that the standard of ability required for good programming is quite high. It is not in fact at all easy to select suitable people for work on programming, and certainly a high level of mathematical knowledge, although it is valuable as a starting point, is certainly neither necessary nor sufficient as a criterion of suitability as a programmer.

TABLE II

Component Failure

Valve failures	357	200 pentodes 108 diodes 49 other types
C.R.T. failures	30	
Condenser failures	10	
Resistor failures	24	

Average rate of valve failure is 1 per 88000 hours.

Average rate of C.R.T. failure is 1 per 3000 hours.

The requirement for highly efficient operators arises from the rather high cost of machine time. This point is always remembered when the engineering efficiency of the machine is under discussion but it sometimes gets forgotten when one is speaking of the efficiency of the actual operation of the machine by the user. Most of the people that we have trained continue to use our machine, usually part time. This happens for instance with people from other Universities. They may come up for one or two days a fortnight to work off their accumulated questions. The same happens with our local industrial connexions, and it is therefore difficult to say how many people are effectively using the machine full time. Our guess is that the total effort is equivalent to about 15 full time mathematical people. Set against this, the scale and the cost of the maintenance effort required is of course quite negligible. From this point of view therefore it is really false economics to economise on the standard of engineering staff employed to maintain the machine. With effort on that scale, i.e. 15 or so full time people, the machine is not kept busy. I do not say that it is ever actually idle at any time, but there is no particular pressure to use what time is available in a highly efficient manner, and one could readily accommodate both more people and a greater volume of work per person. A good deal of this total time I have mentioned has been spent on the development of subroutines of general utility; but this work is now almost complete though we do still use quite a lot of machine time in the development of particular programmes built up out of these subroutines. Our general conclusions, as a result of this experience are that if we regard the machine as a scientific computing machine then it is reliable enough and fast enough to meet present demands. We could easily accommodate more users. This situation might be rectified either by employing more people in the University or by undertaking some routine production; but in the University sphere this does not seem to be the right solution and we seek to use our machine more fully by increasing the circle of users from outside, over and above those permanently associated with the machine staff.

There is one other lesson that I now know we have learnt during this period, but which I had not realized until I heard remarks made here about new facilities that might be supplied by the engineer. I would like to give a word of warning to all engineers on this point - namely, never give to a group of mathematicians any facility or any speed that you are not prepared to maintain for all future time. For they will say: "Any machine that is slower than the one we have had is intolerably slow and any machine that does not contain at least all the facilities of the machine that we are accustomed to is useless!"

5. MOSAIC

The 'Ministry of Supply Automatic Computer'

by

A. W. M. Coombs

Post Office Research Station, London

History and general nature of machine

Originally required by MOS for the rapid solution of one particular problem in the analysis of radar data, "MOSAIC" has in fact been designed with complete flexibility as a general purpose computer. The mathematical design was by NPL, the engineering design by GPO Research, and the manufacture and assembly by All-Power Transformer Co.

It is a high-speed electronic machine (*Fig.* 1) working on the "serial" principle, and using a "3-address" form of instruction.

Store

Ultrasonic mercury delay lines, using quartz crystals, and with a capacity of 1040 numbers or words, each of 40 binary digits.

Speed

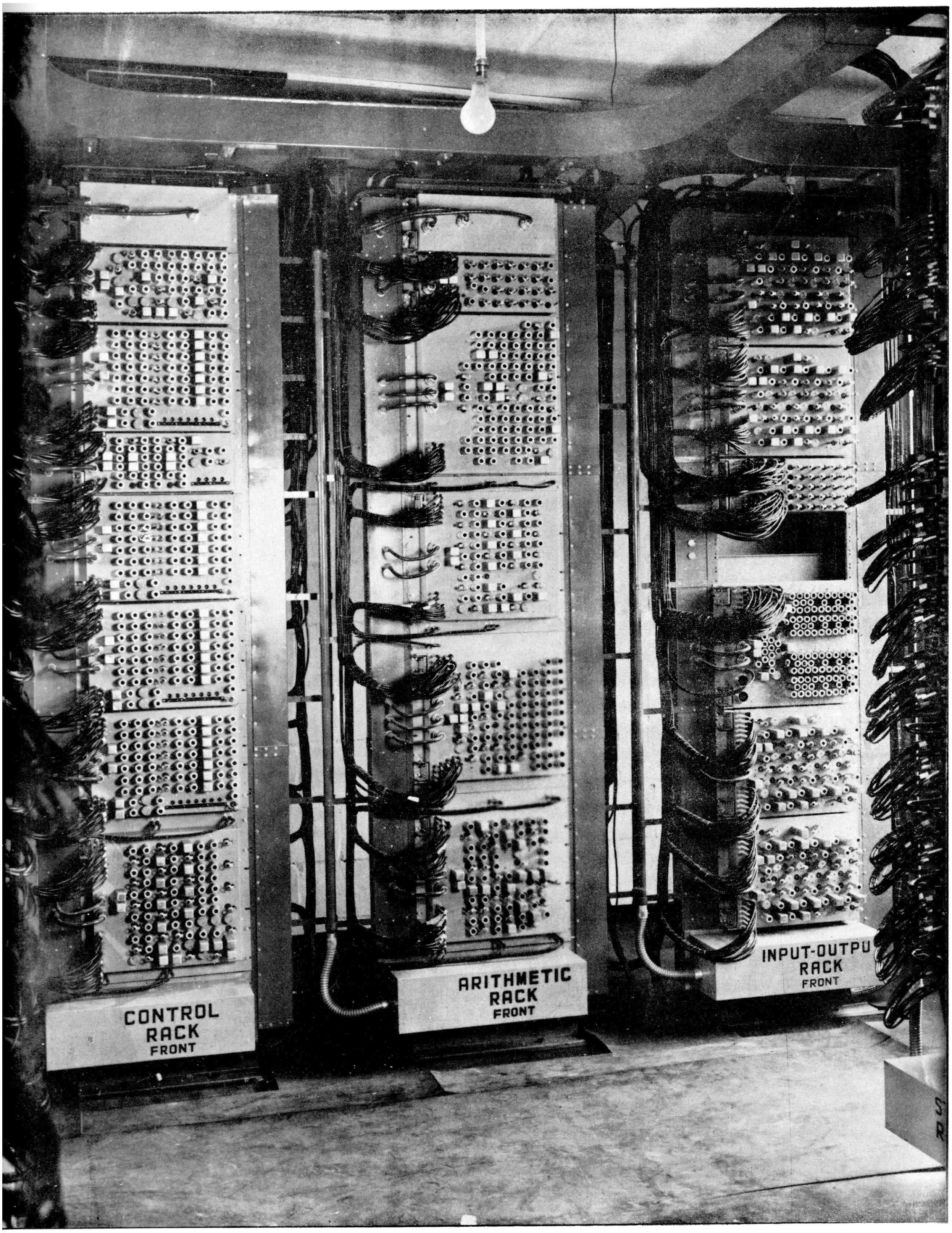
Fundamental pulse frequency 570 kc/s. Addition time for 40 digit numbers 70 μs, multiplication time 6 ms.

Facilities

Addition, subtraction, multiplication, certain logical operations and delay up to 48 digits carried out in response to a single instruction. Other operations must be programmed.

Input and output

Input usually by punched card, but alternative form using 4 in. wide punched paper tape available for the particular problem of radar data analysis. Output by punched card or by automatic typewriter.



PAPER 5

Fig. 1. MOSAIC: Three typical racks.

(14567)

Size of equipment

The machine uses about 6000 thermionic valves and 2000 germanium diodes, spread over 600 sq ft of panel area. It dissipates 30 kw and therefore forced circulation of air is used for cooling. The store contains about ¾ ton of mercury, arranged in 64 long tanks (16 numbers) 3 short tanks (1 number) and one special tank (2 numbers).

General description of operation

Logical principles. The unit operation of the machine consists in deriving two numbers from the store, operating on them in some specified way, and consigning the resulting single number back to the store. The instruction word of 40 digits therefore contains information as to two sources (of numbers), an operation to be performed on the numbers, and a destination for the resulting number - hence the name "Three Address". In addition, the instruction word specifies a further source from which the next instruction is to be drawn while the present operation is proceeding, a timing number and "Characteristic" which together govern the instant of start and the duration of the present operation, and a "Go" digit, without which the machine will "Set up" the instruction, but will then wait without obeying the instruction until some external source authorises continuance of the programme. This latter facility is used for input, output, and manual testing.

The machine operates in successive periods of "Set-up" and "Obey Instruction". The "Set-up" period, during which various leads are arranged under the control of the instruction word to govern the next operation, invariably lasts for one minor cycle (70 µs). The "Obey Instruction" period succeeds it, but is of variable length depending on the timing number and the characteristic. In general, it includes a "Wait" period, a "Transfer" period and a further "Wait" period, all of which will be integral numbers of minor cycles, and the first and third of which may be, and usually are, of zero length. The commonest form of instruction is the "Immediate, 0" type, which makes the two wait periods zero and the transfer period one minor cycle, or 70 µS, so that the interval between successive instructions is then 70 µS, and the complete period from the setting up of one instruction to the setting up of the next is 140 µS. The optimum arrangement can be obtained to some extent by arrangement of the instruction words in the store; there are however some instructions which *sui generis* require more than the optimum period to be obeyed.

The usual discriminate and subroutine facilities are of course available.

Engineering principles. The basic principle embodied in the circuitry of "MOSAIC" is that, from a reliability and maintenance point of view, a large number of simple circuits is to be preferred to a small number of complex circuits, despite possible increased bulk and initial cost. With this goes the corollary that valves may be used in profusion, on the principle of "One electrode, one job", rather than avoided by the use of ingenious but "edgy" networks. For after all, valves are cheap, easily replaced and liable to become rather more than less reliable, and they have the merit of isolating sections of the circuits, which may then be tested very easily.

Thus, the greater part of "MOSAIC" consists of CV138 valves arranged as binary trigger pairs, or as multiple "and" gates. The rest consists of a few "Or" gates, and the necessary linking, phase reversing or buffer valves, which are either CV138 or CV2127, the latter if large low-impedance signals are required. There is a small amount of diode switching using thermionic diodes; the germanium diodes are used solely as clamps.

The standard trigger and gate circuits will be dealt with by Mr. Chandler in a later paper. We shall here mention two of the problems associated with high-speed computers of this type and the technique employed in "MOSAIC" to meet them.

The first problem is that arising from temperature variation of velocity in the delay lines. In the long lines, this causes a delay variation of one pulse per 3°C, which is clearly too great to be tolerated.

Either the temperature must be kept constant, or the clock frequency varied to suit the temperature; greater flexibility is obtained with the second method, which has accordingly been adopted. In effect, all the mercury lines are packed close together in a thermally insulated enclosure, in such a way that they may reasonably be assumed to be at the same temperature. One of the long lines is then used to control the master clock frequency in the required way, using what amounts to a feed-back circuit. An overall manual control is provided in case the wide variation of temperature between say, winter and summer should be outside the capacity of the automatic circuit. The circuit has been wholly successful to date (over 4 months operation).

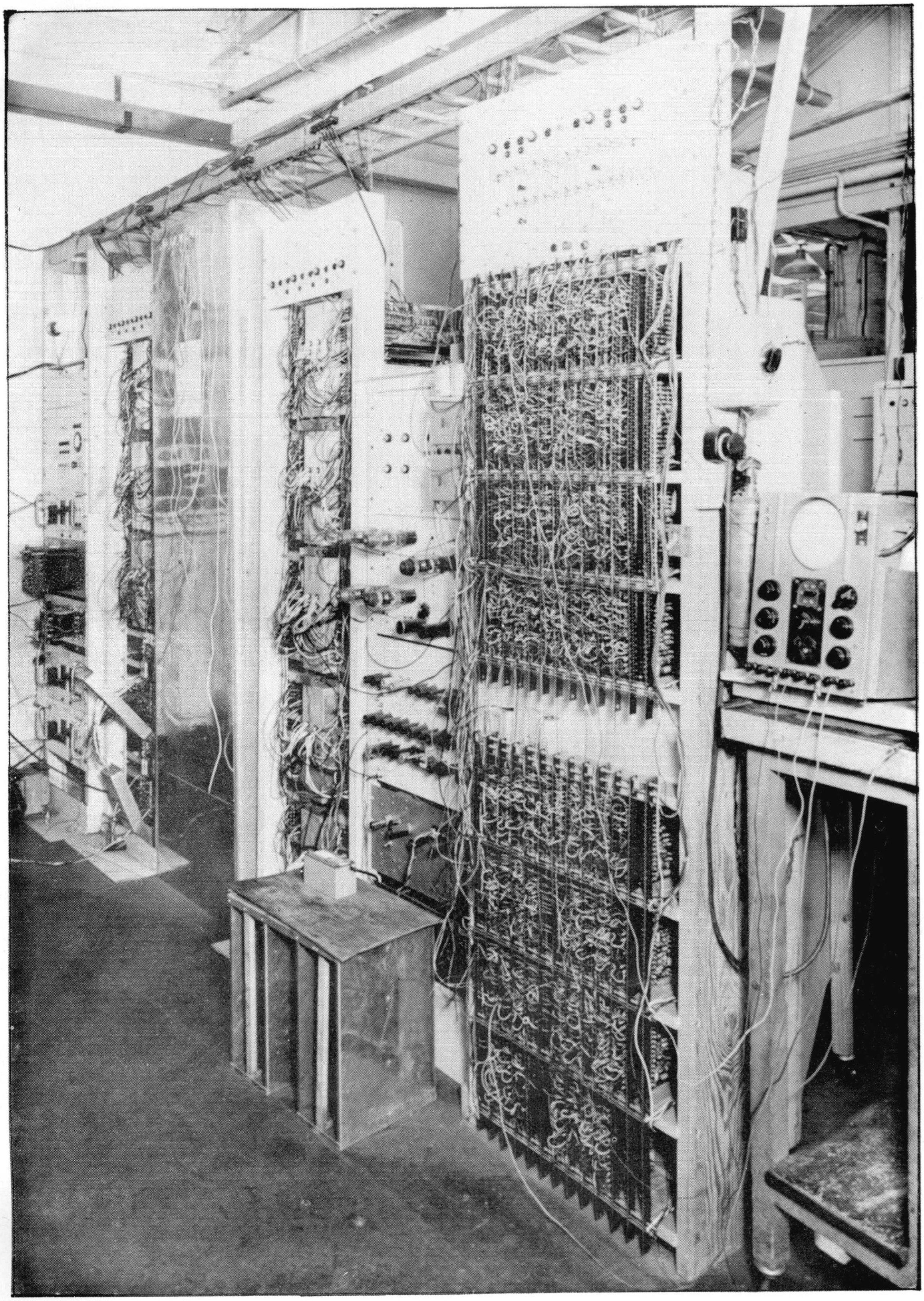
The second problem is that which inevitably arises in any high-speed machine of large size, namely, the loss of synchronism due to the finite velocity of pulses down the cables. In the case of "MOSAIC", this delay can amount to nearly ½ µS (100 yd of cable) which is one third of a pulse period.

Since a delay of up to ½ µS may exist, all signals in transit from one stage of switching to the next are arranged to be delayed a total of one third of a pulse period. Thus signals from different parts of the machine, which may be out of synchronism, are pulled back into synchronism before switching operations are performed upon them. At the same time, a re-shaping operation is performed, so that there is no progressive deterioration either of shape or of timing as the various switching and combining operations go forward. All possible paths of signals are arranged to involve similar numbers of switching stages, so that the same total delay is imposed, and this total delay is compensated for in the delay line lengths - which are in fact all exactly one pulse short, allowing 3 switching stages in the recirculation path, whatever it may be.

The padding-out to a constant delay is achieved by using three separate sets of synchronising pulses, which are available throughout the machine. Each set has a recurrence frequency equal to that of the master clock, but the sets are staggered by 1/3 pulse period relative to each other. They are called the "K1, K2 and K3" pulses. Any one signal pulse is arranged to last from K1 to K3, or K2 to K1, or K3 to K2, and is thus of 2/3 pulse period length. If a particular switching or combining circuit has to deal with pulses - from another part of the machine - which were originally of K1 → K3 type, then it is assumed that these pulses will certainly have arrived at the time K2; therefore they are read off or gated by a K2 pulse on to a trigger which is automatically restored by a K1. The output of this trigger is the pulse to be passed to the next stage of switching, and it is thus re-shaped. re-synchronised, and delayed by 1/3 pulse period relative to its original time (it is in fact a K2 → K1 pulse).

It is essential that the K pulses themselves shall be absolutely synchronised throughout the machine, otherwise delay in these pulses may be additive to the delays in the signal pulses. It is arranged that the K pulses in their peaked form are generated on each panel separately, from a 3-phase system of sine waves distributed around the machine on ring mains at master clock frequency, and to ensure absolute simultaneity, these sinusoids are in fact arranged to be in the form of standing waves. There is thus no delay as between one part of the machine and another.

Layout principles. The panels are all hinged at one side, this giving easy access to the circuits for test and maintenance purposes. The major problem in layout was that of pick-up, both at the low-level ends of the delay lines, and on the H.T. lines, where it had the effect of causing the triggers, which are very high speed, to turn over spontaneously. All H.T. leads are passed through multi-range filters, and thereafter completely screened in hollow aluminium bus-bar chambers or screened lead, and the delay line carrier amplifiers are mounted near the lines, using double screened cable for the low level inputs, soldered into the amplifier boxes. All interconnexions are in coaxial cable.



PAPER 6

Fig. 1. NICHOLAS.

6. Nicholas

by

N. D. Hill

Elliott Bros.

NICHOLAS is the name given to a high-speed digital computer which has been made at Elliott Bros. Research Laboratories at Borehamwood. The decision to build it was made in March 1952, the primary object of its construction being to furnish the mathematicians with a tool with which to gain experience in the programming and solution of a diversity of problems. Also it was desired to obtain practical experience on a full working scale of certain techniques which had been developed in the Laboratories.

A detailed description of the construction and functional logic of the machine will be given in a paper which is at present being prepared but some of its salient features may be of interest here. The arithmetic element, routing and control units are contained in one rack and consist of combinations of three basic circuit plates in which all logical operations are carried out. These logical operations are performed by coincidence gates and mixers using germanium crystals together with digit delays and inverters; the three basic circuit plates also include cathode followers. Magnetostriction nickel delay lines are used exclusively throughout the machine for storage; there are five short lines for the accumulator, multiplicand, order, order number and address counter registers and 65 long lines in the main working store, of which one is used to control the master oscillator in order to minimise the effect of temperature variations on the effective lengths of the lines. The total storage capacity of the main store is 1024 words to which the average access time is 6¼ ms. Both the circuit plates and nickel line store have been described in recent colloquia at Cambridge and the development of the store is also being described by R. C. Robbins in a later paper.

Input is from 5-hole punched paper tape and output on standard teleprinter equipment giving either printed and punched paper tape or printed page.

Words are 32 digits in length and consist either of numbers or of two 16 digit instructions. A single-address code is used in which the first 10 digits designate the location of the word to which the instruction pertains and the remaining 6 digits denote the function, e.g. add, collate. The average computing speed is approximately 100 operations per second, an operation consisting of an addition, multiplication, transfer, etc.

By the beginning of August, five months after the decision to build the machine, the majority of the hardware had come together, including sufficient of the store to enable small computing problems to be solved. The first major problem, consisting of trajectory calculations requiring about 1400 orders, was begun on January 1st and is proceeding. Other problems are being programmed at the moment.

The compilation of a library of subroutines is in preparation. The trajectory calculations were programmed directly in machine code but initial orders have now been evolved which enable programme tapes to be punched in a very simple code and numbers to be punched directly as decimal numbers. The latest problems are all being programmed in the new simplified code.

---

7. Advance Notes on RASCAL

by

E. J. Petherick

Royal Aircraft Establishment, Farnborough

Introduction

High-speed binary computers operate to best advantage on analytical problems involving lengthy computation with but a modest amount of input of data and output of answers. However, much of the computing in an experimental organization such as the RAE is not of this type; it is concerned with the reduction of experimental data, garnered from wind tunnels, structural and flight tests, missile development, etc. A vast amount of such data is collected and an increasing proportion of it is being recorded automatically both for visual scrutiny and for feeding to automatic computers. The computing needed, however, frequently falls below five arithmetic operations per item of data, though the sequences may run to several hundred operations. Such problems are therefore suitable neither for conventional business machines nor for high-speed binary computers. Business machines lack the required programming capacity; and the high-speed binary machines are limited both by their speeds of data input and answer extraction and also because the double conversion they demand, to and from the binary notation, would swamp the actual computing required (in most cases the need for scrutiny of data by experimenters renders recording of data in binary form undesirable).

In any machine for data reduction, therefore, the input facilities must be particularly good; much of the data will be read once only and in a prescribed sequence, and so attention should be paid to rapid access 'on demand' to the next number on a chosen input medium- tape or card stack. Output facilities are required which enable results to be disseminated quickly to the experimenters and later to a wider public in report form. By contrast, computing speed can be sacrificed to economy, simplicity, reliability and serviceability. Moreover, any machine built primarily for data reduction should be sufficiently simple and easy to programme that most customers can load their own problems. This has two advantages in any establishment like RAE; it emphasizes both the capabilities and - perhaps more important - the limitations of automatic computing direct to the staff concerned in the design of experiments, with - it is hoped - consequent économy in subsequent work.

Effort at RAE was first concentrated on a slow relay computer, but with the advent of reliable electronic decimal elements, the emphasis was shifted to design of a medium-speed electronic machine, RASCAL (Royal Aircraft Establishment Sequence Calculator) and development of this design is now being handled by the Plessey Company. It should be stressed that, unlike most of the machines dealt with at this session of the Symposium, RASCAL is still under development and no operating performance is available. Emphasis on data reduction has, however, resulted in some unusual features which may be of interest, particularly as this design slant resembles that given by many commercial applications.

RASCAL will use the floating decimal, parallel counting mode of operation, with fast, serial, coded input from punched tapes or cards; and up to three tape punches, card punches or typewriters will record its answers simultaneously while computing proceeds. The two - address control code - detailed in Table I - will provide facilities for modifying orders similar to those afforded by the B-tube of the Manchester machine. RASCAL should be portable and should need about 450 valves. It will use GC 10 D dekatrons and G1/370K cold-cathode trigger tubes as its main arithmetic elements and a magnetic drum as its main internal store. The rhythm of the machine follows from the dekatrons' limit of 20000 steps per second and also from the decision that each operation should be complete in one drum turn, *i.e.* in 9 ms. Some such work turns will be preceded by a turn for reading an operand to the arithmetic unit and/or writing the answer of the preceding stage on the drum; and the order for each operation will be read and modified during the preceding work turn.

Design hinges on the methods adopted for choosing and forming multiples - and their complements - in short-cut multiplication. These methods incidentally permit virtually simultaneous carry in the Accumulator and employ two decimal codes thought to be novel.

Components

In the GC 10 D dekatron (*ref.* 1) a discharge can be arranged between a central anode and one of ten surrounding cathodes. This discharge can be stepped to successive cathodes by feeding to certain transfer electrodes pulses of 25 μs duration and spaced at least 25 μs apart. Nine of the cathodes are strapped internally and the base pins give access to that group and to the remaining cathode - the valve is therefore ideal for counting and the pulse available as it passes its lone cathode (used as its ninth) can initiate inter-digit carry.

The G1/370K trigger tube (*ref.* 2) is a similar binary device. A low current discharge between a subsidiary anode and cathode can be extended to a main anode cathode gap in a few μs, by suitably pulsing a transfer electrode. Once established in the main gap the discharge takes longer to extinguish, a 25 μs pulse being required to de-ionize the gas in that gap.

Both these components have the advantage that their discharges enable the contents of the various registers, etc. to be seen immediately the machine is stopped, without further test gear. Their use in a parallel arithmetic unit permits 8 x 8 decimal multiplication to be performed in 5.8 ms, using short-cut methods and a basic pulse repetition rate no higher than 20 Kc/s.

Input

Data and orders - intermingled if needed - will be fed to RASCAL on punched tape or cards. Numbers will be supplied in the form $p \times 10^j$, where $-1 < p < 1$ and $-19 \leqslant j \leqslant 29$; the exponent j will be punched first, followed by the sign of the significant figures (coded as 0 if positive or as 9) and up to eight significant figures. Most orders will comprise two function digits, a modification digit and two four-digit addresses, which will occupy positions corresponding to the exponent, sign and significant figures of numbers, respectively. Each digit will be represented either in a column of a card by one hole, or on a line across a tape by a code in which the four channels indicate the digits 1, 2, 3 or 9 and are combined according to Table II. Each word will be preceded by an X punching on cards or by a three-hole combination C on tape; but any number of zero exponent can have that exponent and its X or C code replaced by a Y punching or another three-hole code D, respectively. This method of data input reverses the procedure on fixed point machines, of programming a floating point when required, and can only be justified if the built-in floating point does not slow computing appreciably. In RASCAL, most operation times will only exceed those in the comparable (hypothetical) fixed point design in the ratio of 7 : 6.

The readers have been designed to reduce the delay between the arithmetic unit calling for and receiving the next item of data from any chosen tape or card stack. In each tape-reader, for instance, the required number and its adjacent C or D codes will be left aligned roughly in front of four strip photocells (one for each channel) and will be scanned repetitively by a beam of light, deflected along the tape past the number required in the direction of decreasing significance. To read that number, the arithmetic unit will be connected to the photocells at the start of a scan and will record all the digits reaching it after the first and before the second C or D code is sensed. The input delay is thus determined by the speed and frequency of scan and is limited only by the resolving power of the photocells; in RASCAL, no advantage is gained by reducing the delay below 7 ms, though it could be cut to 200 μs. The tape will be accelerated independently of the scanning system, when a number is demanded, and will move till a C or D code passes a subsidiary optical system and the next number on the tape is aligned in front of the photocells. Thus only the average rate of input is limited by the tape transit mechanism, which can complete its movement while the computer utilises the data read during one scan. It will be possible to plug up to six tape or similar card readers to the computer at any time and a keyboard will facilitate manual insertion of data or orders.

Storage

On the drum the same 1, 2, 3, 9 code will be used. Each group of four tracks will store 100 numbers. The tens digits of the exponents of all 100 numbers will be stored together on one sector, followed on the next sector by all 100 units exponent digits; sector 3 will record the sign of the significant figures, sector 4 the first significant figure, etc. till the twelfth sector round the drum contains the ninth significant figures o all 100 numbers-retained to minimize loss of accuracy. Ten digits will be stored in each o the first ten of thirteen sub-sectors within each sector to pass the reading heads, the rest of each sector and two further sectors being reserved for certain control functions. Order will be stored on the drum like numbers and will be available both for control purposes and also for alteration in the arithmetic unit.

Using a circumferential packing of 75 digits/in. the drum's diameter will be approximately 8 in. and its speed, to fit in with dekatron limits, will approximate to 6700 r.p.m. This will give an addition time of 9 and a multiplication time of 18 ms including access to operands and disposal of answers, for a digit writing and reading rate no higher than 1600/s.

For 9000 addresses, only 36 magnetic heads will be used, each capable of writing number and orders on the drum and moveable to cover any chosen one of ten tracks. Four more writing heads will extend the storage to 9900 numbers and four static heads will give access to a further 100 stores, primarily for modification and output purposes. At any time, therefore, 1100 stores will be immediately available without moving any heads. Associated with and spaced one sector from each writing head will be a reading head, the separation of the writing and the reading functions simplifying switching of digits to and from appropriate heads.

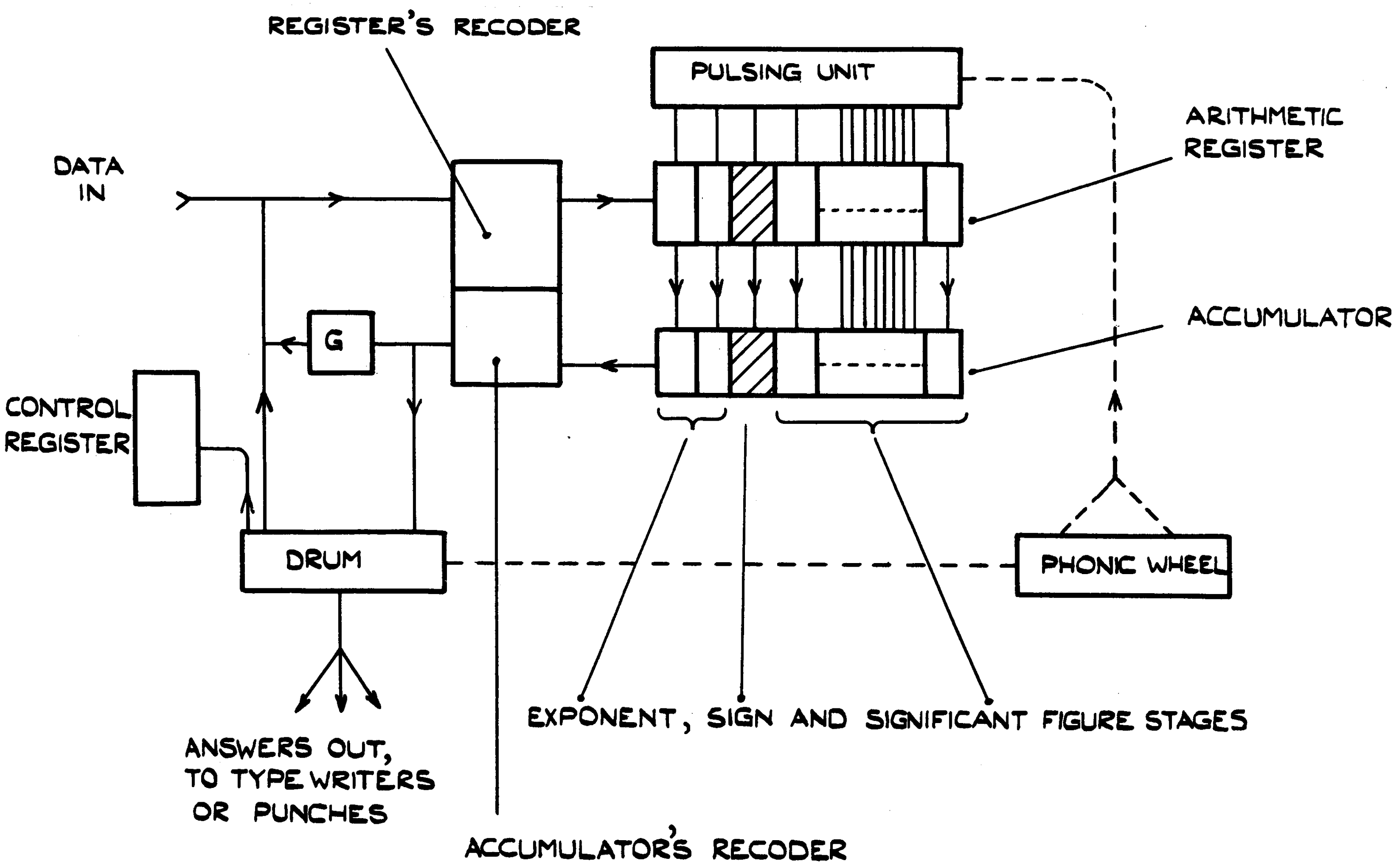
Of the four digits of each address, the second will select one group of eight heads; the first digit will control movement of the chosen heads to cover the appropriate tracks; and the third and fourth digits will set up coincidence circuits to open writing or reading gates at times dictated by a phonic wheel (see below). If the first and second digits of a address are both nine, however, the octet of fixed heads will be used.

The phonic wheel

Attached to the drum will be a phonic wheel, the 17 channels of which will define completely each digit storage location round the drum, in a two-out-of-six code resembling the 1, 2, 3, 9 code and detailed in Table II. Six channels will define which sector, a further six which sub-sector and five more channels which digit within a sub-sector is passing the reading heads at any instant. This phonic wheel may be formed from a stack of disk each divided notionally round its edge into 1820 equal divisions. Some divisions would the be left blank and others would be marked with scratches in accordance with the six-channel code of Table II; and as the wheel turns, those scratches would generate pulses in 17 magnetic heads similar to those reading digits from the drum. Alternatively the phonic wheel may consist of a single disk photographically etched with a similar code. Either scheme enables appropriate digits to be read from the drum when required, at times defined b coincidence between the outputs from the sub-sector and digit channels of the wheel and corresponding channels specifying the third and fourth digits of the address of the required number. Further coincidence circuits examining the output from the wheel will provide pulses - at intervals of five µs. for controlling the machine during each drum turn.

The arithmetic unit

In the arithmetic unit there will be only four main assemblies, termed the Recoders, the Register, the Accumulator and the Pulsing Unit (*fig.* 1). Digits of a number read serially from the drum, tapes, cards or the keyboard, through one Recoder, will be staticised on quintets of trigger tubes forming the Register, for transfer in parallel from there to the Accumulator, each digit N as a train of N pulses under control of the Pulsing Unit. For subtraction there will be available complements of numbers on the Register, while for short-cut multiplication there will be needed the positive and negative multiples 2-5 of the multiplicand on the Register. The Register will be able to shift the significant



PAPER 7

FIG. I. SCHEMATIC DIAGRAM OF RASCAL

figures of its contents one place in the direction of decreasing significance for every pulse applied to certain control lines.

The pure decimal Accumulator will employ 12 dekatrons, of which the nine for significant figures will be arranged in a ring and will carry virtually simultaneously through nines. Addition or subtraction of a number or of one of its multiples, to or from the Accumulator, will take 650 μs. Answers formed on the Accumulator in pure decimal notation will read out serially, through and under control of the Accumulator's Recoder, to the drum or back to the Register in 1, 2, 3, 9 code.

The Accumulator's Recoder will also deal with any overspill from the Accumulator during addition or subtraction; will partially standardize products; and will provide a check against the processing of numbers with exponents outside the permissible range.

Formation of multiples

Table III lists the products resulting from multiplying each possible multiplicand digit by each of the factors 1-5. Table III also lists the sum of the digits in, and hence the number of pulses needed to form, each two-digit product. But for any multiple the tens digit of one product, of a multiplicand digit by a factor, may have to be added to the units digit of another product, of the next more significant multiplicand digit by the same factor. Multiples 2-5 may therefore need up to 1+8, 2+9, 3+8, or 4+5 pulses respectively. Since for multiples 3 and 4 eleven pulses may reach any dekatron, two carries may result and the basic cycle of the machine - available as each drum sector passes the reading heads - therefore comprises thirteen pulses, one in each sub-sector.

Table III also indicates that for each factor, the pulses needed to form the two-digit products for the multiplicand digits N and (9-N) sum to nine, except for the cases boxed. This suggests that each digit N < 5 on the Register should time the opening, and each digit (9-N) the closing, of a gate feeding pulses from a common supply of nine to the appropriate dekatron in the Accumulator. Each gate will be controlled with the aid of a flipflop, which will be in its gate-closed state before each cycle of nine pulses.

To transfer an operand, for instance, each digit 1 therein will step its flipflop to open its gate after pulse 8 of the group of nine, to allow pulse 9 to reach the Accumulator. Each digit 8(=9-1) will step its flipflop and open its gate before pulse 1, to step its dekatron until the gate is closed after pulse 8 by the same control used for complementary digits 1. Any gate controlled by one of the other digit pairs 2/7, 3/6 or 4/5 will be opened before the cycle if its digit exceeds 4 and will change state after pulses 7, 6 or 5 respectively. This system enables the complement of an operand to be formed easily, by stepping each flipflop again before each subtraction cycle.

To produce each of the multiples 2-5 and their complements, the above procedure will be extended to form the 13-pulse cycle demanded by Table III. In the first part of each such cycle the tens digits, and in a second part the corresponding units digits, of the two-digit products of the multiplicand digits by a chosen factor, will reach the Accumulator. At the end of the first sub-cycle, the Register's contents will be shifted one place in the direction of decreasing significance, and inter-digit carry will be arranged before the start of the second sub-cycle. The two sub-cycles and the carry period will together occupy the first ten pulses of the cycle; pulses 11 and 12 will be needed to complete multiples 3 and 4; and pulse 13 will provide the second inter-digit carry period.

The lower half of Table III indicates the shift and carry periods for each of the factors 1-5. Before each sub-cycle, all the Accumulator's flipflops will be stepped from their gate-shut positions if a complement is needed and each flipflop may be stepped again if its controlling digit exceeds four. The flipflops will be further stepped during either or both sub-cycles according to the lower half of Table III and all will be returned to their gate-shut positions before and after each carry period and also after pulse 10.

For the boxed cases of Table III, the relevant flipflops will be stepped specially, to shut their gates after pulses 1 or 2, for factors 3 or 4 respectively, and to open their gates before pulse 5. There will thus be formed in the first ten pulses of the cycle the relevant two-digit product 07, 16, 06 or 26, to which will be added two further correcting pulses - Nos. 11 and 12 - by stepping the flipflops again before pulse 11.

Five-channel code easing choice of control pulses

Choice of the appropriate control pulse(s) by each multiplicand digit - from the set of pulses provided for each factor by the Pulsing Unit - will be simplified by a simple change in coding, from the four-channel code used on tapes and drum, where space is limited, to a somewhat similar five-channel code on the Register. The latter code, listed in Table II, indicates directly into which complementary pair each multiplicand digit falls and whether that digit exceeds four. In each stage of the Register, four trigger tubes A, B, C and D will be associated with the digit pairs 1/8, 2/7, 3/6 and 4/5, respectively, and a fifth tube E will indicate whether the digit on the quartet exceeds four. Zero will be indicated by no tubes being flashed and nine by tube E only in the relevant quintet.

Choice of multiples in short-cut multiplication

In short-cut multiplication, the multiplicand will be set on the register during one drum turn and in the next turn the multiplier will select appropriate multiples of that multiplicand, to add to or subtract from the accumulator to form the product thereon. The required multiples can be chosen correctly and easily by inspecting no more than two adjacent multiplier digits at a time, expressed in the same 5-channel code. Multiples higher than the fifth can be avoided, by subtracting a lower multiple from the tenth - the multiplicand itself shifted one place to the right; and any tenth multiple for the less significant multiplier digit of a pair suspected can be added to the lower multiple for the more significant of those digits. The most and least significant multiplier digits can be treated similarly if they are considered to be preceded and followed respectively by zeros.

This procedure is tabulated in Table IV and can be further summarized as follows. When both the multiplier digits inspected are less than 5 and the more significant is N, then the Nth multiple of the multiplicand must be added to the Accumulator. The same multiple must be subtracted if both the multiplier digits exceed 4 and the more significant is (9 - N); but if only one of the multiplier digits exceeds 4 then the multiple so chosen must be replaced by the next larger multiple.

If the multiplier digits to be thus inspected are set onto trigger tube quintets in the 5-channel code of Table II, tubes A to D in the more significant quintet determine the value of N directly; tube E therein determines the sign of the multiple and the multiple so chosen must be replaced by the next larger if only one of the E tubes in those quintets is flashed. Such choice of a multiple and its sign can be arranged with no more than two flipflops and ten gates over and above the two quintets of trigger tubes described above, and due to the arrangement of multiplier digits on the drum no further multiplier register is required; in fact the register quintets required for the exponent digits can fulfil the function of a multiplier register.

Reading of answers - the Accumulator's Recoder

For each answer formed on the Accumulator, the exponent dekatrons will hold either a positive exponent or the complement on 99 of the modulus of a negative exponent; the sign dekatron will record a zero if the answer is positive, or a nine if it is negative; each dekatron in the ring will hold either a digit of a positive answer or the complement on 9 of a digit of a negative answer; and the decimal point will effectively lie to the left of the dekatron chosen as the most significant. Some answers will be left on the Accumulator to be augmented in subsequent operations; others will be read out and cleared, digit by digit, to the drum for storage or to the Register for use as operands; while a few will be both read out and also left on the Accumulator (after operations 04, 05, 24 and 25).

In reading the Accumulator serially, the tens and units exponent dekatrons will be dealt with first, followed by the sign dekatron and then those for significant figures; that is, the digits of each answer will be read in the same order as the digits of each operand are accepted from the tapes, cards, or keyboard. The digit on the more significant exponent dekatron, after setting the Accumulators' Recoder, will determine whether to read the digit on the less significant exponent dekatron or that digit's complement on 9; and the digit read from the sign dekatron to the Recoder will similarly determine whether the digits on the significant figure dekatrons or their complements on 9 should read. This reading process will usually occur while the first operand of the following stage is drawn from the drum, cards or tape to the Register.

One dekatron will be read to the Accumulator's recoder as each of the drum sectors 1-12 pass the reading heads. As the end of each drum sector N passes the reading heads, the digit set on the Accumulator's Recoder earlier in that sector will be transferred to the storage unit, for writing on drum sector N through the appropriate recording head which, it will be remembered, will be displaced one sector from its corresponding reading head. Alternatively, at the end of each sector the contents of the Accumulator's Recoder may be transferred to the Register's Recoder and on to the Register's quintets.

Control

Successive orders will normally be drawn from adjacent storage locations round the drum and the addresses specified by each order will be modified while the preceding order is being obeyed. For this modification, each order's modification digit will select one of ten ordinary numbers on the drum; and the first four significant figures of that number will be added to the first address and the second four to the second address specified by the order, on a Control Register. The flexibility afforded by this procedure will be further extended by the ability to alter not only the modifier numbers in the arithmetic unit, but also the orders themselves; the function digits, modification digit and addresses of an order will then be treated as exponents, sign and significant figures respectively.

Output

The output system will extract up to three answers at a time from the drum and will type and/or punch them out while the computer continues calculating. This simple time-multiplexed system will be driven from the four static reading heads and will yield a typed display suitable for photostatic reproduction and up to 20 carbon copies for immediate distribution. Its maximum output speed will be 36 digits/s, which will allow one answer to be fed out for approximately every 25 arithmetic operations completed.

References

1. ACTON, J. R. The single-pulse dekatron. *Electron. Engng,* 1952. 24, 48.

2. HOUGH, G. H. AND RIDLER, D. S. Some recently developed cold cathode discharge tubes and associated circuits. *Electron, Engng,* 1952, 24, 152, 230 and 272.

TABLE I

The Order Code

Note: (A) means the contents of location A on the drum, the next number on tape or cards in reader A or the contents of the Accumulator

03 06	Add (A) to) Subtract (A) from)	Accumulator, clear result from Acc. to destination B
04 05	Add (A) to) Subtract (A) from)	Accumulator, leave result on Accumulator and also read it out to destination B
13 16	Add unity to) Subtract unity from)	4th sig. Fig. of (A), send result to B
14 15	Add unity to) Subtract unity from)	8th sig. Fig. of (A), send result to B
17	Add 5 to the Mth sig. Fig. of (A), where M is the modifier digit; send rounded up number to destination B and prepare to read it out to a punch or typewriter	
22 27	(A) X (B)) (A) X-(B))	and add product to the Accumulator
23 26	(Acc) X (A)) (Acc) X-(A))	and clear product from Accumulator to destination B
24 25	(Acc) X (A)) (Acc) X-(A))	leave product on the Accumulator and also read it out to B
80	Set trigger	according to the modifier digit M, so that when attempts are made to read an 'outsize' number from Acc: if M = 0 - 8, wait for operator to clear Acc. If M = 9, clear Acc. and proceed to next order but one
81 88	Dummy stop. Full stop	If control panel so demands, halt machine In any case, halt machine
82 87	Tabulate Return	carriage on typewriter M to next hand set tabulate stop carriage on typewriter M to start of line and line space
89	Initiate output cycle for punching or typing any numbers placed on drum by up to three operations 17	
91	Strip Acc.	For each zero in (A) clear corresponding digit on Acc. to 0 and send stripped number to destination B
92	Transfer	(A)'s exponent as a standardized number to B
93	Discriminate	If (A) is negative or zero, proceed normally, otherwise skip to order in location B
97	(Acc) x $10^{[A]}$	and clear product to B. [A] means the integral part of (A), which must lie between ±99.
98	Standardize (A)	by eliminating "leading" zeros in (A)'s modulus and correspondingly reducing (A)'s exponent

Note: each operation occupies one drum turn except

23 - 25, which take 2 drum turns; and

22 & 27, which take 3 drum turns if the Accumulator holds a number or 2 drum turns if the Accumulator is empty.

TABLE II

Codes used on tapes, drum, phonic wheel and arithmetic register

		Four-channel code for tapes & drum				Digit	Five-channel code for arithmetic register				
		Channels used					Channels used				
		1	2	3	9		1/8	2/7	3/6	4/5	>4
		✓	✓			0					
	✓	✓				1	✓				
	✓		✓			2		✓			
	✓			✓		3			✓		
		✓		✓		4				✓	
			✓	✓		5				✓	✓
				✓	✓	6			✓		✓
			✓		✓	7		✓			✓
		✓			✓	8	✓				✓
	✓				✓	9					✓
✓	✓					10					
✓		✓				11					
✓			✓			12					
✓				✓		13					
Y	X	1	2	3	9						
Channels used in 2 - out - of - 6 code on the phonic wheel											

TABLE III

Products resulting from multiplying each possible multiplicand digit by each of the factors 1-5;
The sum of the digits in each such product (in brackets); and
Method of forming each such product in a 13-pulse cycle.

Multiplicand digit \ Multiplying factor			1	2	3	4	5
0			00 (0)	00 (0)	00 (0)	00 (0)	00 (0)
1			01 (1)	02 (2)	03 (3)	04 (4)	05 (5)
2			02 (2)	04 (4)	06 (6)	08 (8)	10 (1)
3			03 (3)	06 (6)	09 (9)	12 (3)	15 (6)
4			04 (4)	08 (8)	12 (3)	16 (7)	20 (2)
5			05 (5)	10 (1)	15 (6)	20 (2)	25 (7)
6			06 (6)	12 (3)	18 (9)	24 (6)	30 (3)
7			07 (7)	14 (5)	21 (3)	28 (10)	35 (8)
8			08 (8)	16 (7)	24 (6)	32 (5)	40 (4)
9			09 (9)	18 (9)	27 (9)	36 (9)	45 (9)
Register's contents shift right and 1st carry period occurs	during pulses		1	2	3	4	5
Multiplicand digits 1	open, and digits (8)	shut gates after pulses	9	8	7	6	5
Multiplicand digits 2	open, and digits 7	shut gates after pulses	8	6	4	No. 4	3
Multiplicand digits 3	open, and digits 6	shut gates after pulses	7	4	No. 3	2 + 8	3 + 5
Multiplicand digits 4	open, and digits 5	shut gates after pulses	6	2	1 + 8	2 + 4	2
Multiplicand digits 0 and 9			do not alter gate positions during cycle				

For both boxed cases, gates are opened before pulses 5 and 11.

For boxed case No. 3, gates are shut after pulse 1 and their state is altered after pulse 3.

For boxed case No. 4, gates are shut after pulse 2.

All gates are controlled by flipflops and are shut after pulses 10, 12 and 13 and also before and after each moveable carry period.

TABLE IV

Choice of multiple at each stage of short-cut multiplication from inspection of two adjacent multiplier digits

More significant multiplier digit		0	1	2	3	4	5	6	7	8	9
Multiple required if less significant multiplier digit of the pair inspected is	< 5	0	1	2	3	4	-5	-4	-3	-2	-1
	> 4	1	2	3	4	5	-4	-3	-2	-1	-0

8. The TRE High-Speed Digital Computer

by

R. H. A. Carter

Telecommunications Research Establishment

Introduction

The TRE High-Speed Electronic Computer has been designed to work in the parallel mode, with a high-speed electrostatic store (using c.r.t.'s) with a capacity for 512 words, each of 24 digits. It has a single address code, punched tape input, and either punched tape or electric typewriter output.

An auxiliary magnetic drum store has been built with a total capacity of over 1½ million binary digits.

The computer works in a pure binary scale, with a fixed binary point, conversion from the binary coded decimal input being controlled by the input subroutine. A corresponding output routine controls the conversion from a pure binary answer into the output code.

Few restrictions are imposed by the computer upon the mathematician; *e.g.* the contents of any given address in the high-speed store may be read as often as required, and all addresses are equally accessible at all times.

As far as possible, both '0's and '1's are represented by active states.

The power supplies for the computer are provided by four 400-cycle alternators so arranged that all four alternators normally share the load, one supplying the L.T. power, and the remainder the H.T. power. Under abnormal conditions, the computer may derive all its power from only two alternators, the change from four to three, and from three to two, being effected without interruption in the computation.

General Design

As far as possible, the binary digits '0' and '1' are each represented by an active state; the voltage appearing on a digit line representing either a digit or no digit, the '0' being distinguished from the '1' by the use of separate lines. Hence, the absence or presence of *both* '0' *and* '1' is easily detected.

The circuits have been designed in such a manner that wherever it is possible, the failure of a component will not provide the active state on a digit line.

Separate trigger circuits are used for high-speed temporary storage of '0's and '1's (in registers). It is thus possible to have three states for each digit in a register, viz. 'cleared' '0' and '1'.

The equipment is mounted on both sides of Post Office racks, four digits to a rack. On the lower half are mounted the c.r.t. store units, the c.r.t.'s themselves being at the bottom. Each video amplifier is mounted on the end of the screening can of the associated c.r.t. Above the c.r.t.'s are the deflection amplifiers each of which provides the required voltage for one dimension ('x' or 'y') for four c.r.t.'s in parallel. The address is fed into the amplifier in digital form, so that losses between the address registers and the amplifiers do not affect the accuracy of the deflection on the c.r.t. screen.

Above the deflection amplifiers are mounted the Gate Units, by means of which access is gained to the c.r.t. store for both 'reading' and 'writing', and the unit also forms the link between the output of, and the input to, the store during the regeneration periods.

Above the Gate Units are the various Arithmetic Units, with the power supply controls at the top of the rack.

The 'back' and the 'front' of the digit racks are similar in appearance, having similar units mounted on them up to the power supply panel which is not duplicated on any one rack.

All the units above the c.r.t.'s have been designed for easy access without dismantling from the rack, to enable servicing to be done *in situ*.

Cooling has also been considered. Valves are held in resistor clips, so that heat may be transferred to the chassis, and the valves are arranged to protrude into a vertical duct. This duct forms a flue through which the air, heated by the equipment, is allowed to rise and is extracted from the room by fans fitted in the roof. Cooled filtered air is provided through ducts under the floor of the computer room and this air is allowed to pass freely over the units of the equipment. The convection currents thus created are found to provide adequate cooling of the equipment.

The arithmetic units, being serviced *in situ*, have no plugs and sockets; all connexions are made by means of soldered joints, and all valves are soldered in. The result of this has been that no troubles have been found to be due to poor connexions. The desirability of soldering all joints has been emphasised by the fact that plugs and sockets and associated connexions on the c.r.t. units have given some trouble. When the equipment has been working for a year or so this particular problem will be examined and the frequency of faults due to this cause will be determined, together with faults from other causes.

Power for the computer is being provided from four 400 c/s 3-phase alternators each driven by a 50 c/s, 3-phase motor. The system has been designed to provide uninterrupted power to the computer during a complete break of the power supply to the motors of up to one second. Also, should any one alternator fail while under load, automatic switch gear will take that alternator out of service and the full load will be shared by the remaining three alternators, again without affecting the computation.

The Function Design

Fourteen instructions have, so far, been catered for in the design of the Arithmetic Control section of the computer. They are:

1. P.n. Clear the Accumulator Register and then add the contents of store, address n.
2. N.n Clear the Accumulator Register and then subtract the contents of store, address n.
3. A.n Add the contents of Store, address n, to the contents of the Accumulator Register, and place the result in the Accumulator Register.
4. S.n Subtract the contents of Store, address n, from the contents of the Accumulator Register, and place the result in the Accumulator Register.
5. T.n Transfer the contents of the Accumulator Register to Store address n, leaving the contents of the Accumulator Register unaltered.
6. R.(n) Shift the contents of the Accumulator Register one place to the right (*i.e.* divide by 2). The address n has no effect.
7. E.n Compare the digits of the Accumulator Register and of the store address n, digit by digit. If the digits are both one, put one into the Accumulator Register otherwise nought.

8. O.n Compare the digits of the Accumulator Register and of the store address n, digit by digit. If the digits are both nought, put nought into the accumulator Register, otherwise one.

9. D.n Compare the digits of the Accumulator Register and of the store address n, digit by digit. If they differ put one in the Accumulator Register, otherwise nought.

10. B.n If the number in the Accumulator Register is negative, do next the instruction stored at address n. Otherwise proceed serially.

11. J.n Do next the instruction stored at address n.

12. I.n Read the row of tape into the five least significant digits of store, address n, setting all the remaining digits of that storage location to noughts.

13. W.n Write out the contents of the five most significant digits in store address n.

14. Z.(n) Stop. (n has no effect).

When the magnetic drum store is connected up to the computer, the necessary instructions can be added to the control system in a simple manner. It should be noted that reading the information from the Stores (Instructions 1,2,3,4,7,8,9 and 13) always leaves the contents of the Stores unaltered.

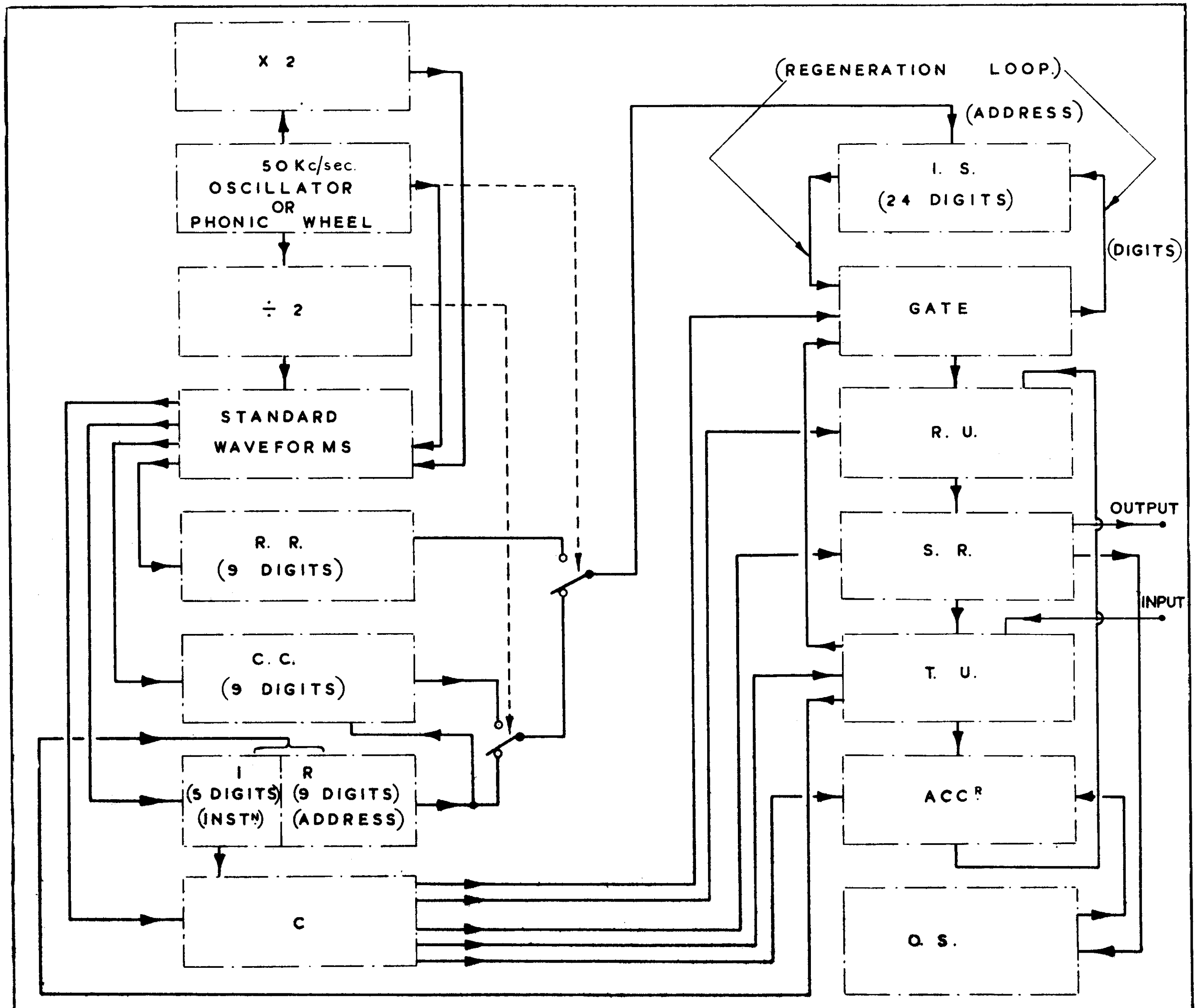
The primary timing of the operation of the computer is derived from a 50 kc/s basic waveform. This will be a phonic wheel rigidly fastened to the magnetic drum, when the latter is in use. Otherwise, an electronic generator of 50 kc/s is used. By adopting this system, the problem of synchronising a drum to the basic waveform is avoided, and is replaced by the simpler one of designing the electronics of the computer to be aperiodic in nature to a large extent. This has to be so, in any case, owing to the random nature of the sequence of instructions, some of which (*e.g.* printing, and reading input tape) are relativel slow, and require that the computer should stop while the slow mechanical operations are performed.

The phonic wheel consists of a disc of steel, four inches in diameter, with 2048 teeth cut into its perimeter. This is then rotated at the drum speed of 1500 revolutions per minute, giving an output frequency of approximately 50 kc/s.

Fig.8 is a block schematic diagram of the computer. The output of the 50 kc/s source is fed into a unit which provides a square waveform with a 50% duty cycle, together with a second output of opposite phase. These voltages are then used to provide the remaining basi waveforms of the computer, which continue uninterrupted, irrespective of the mathematics bein carried out.

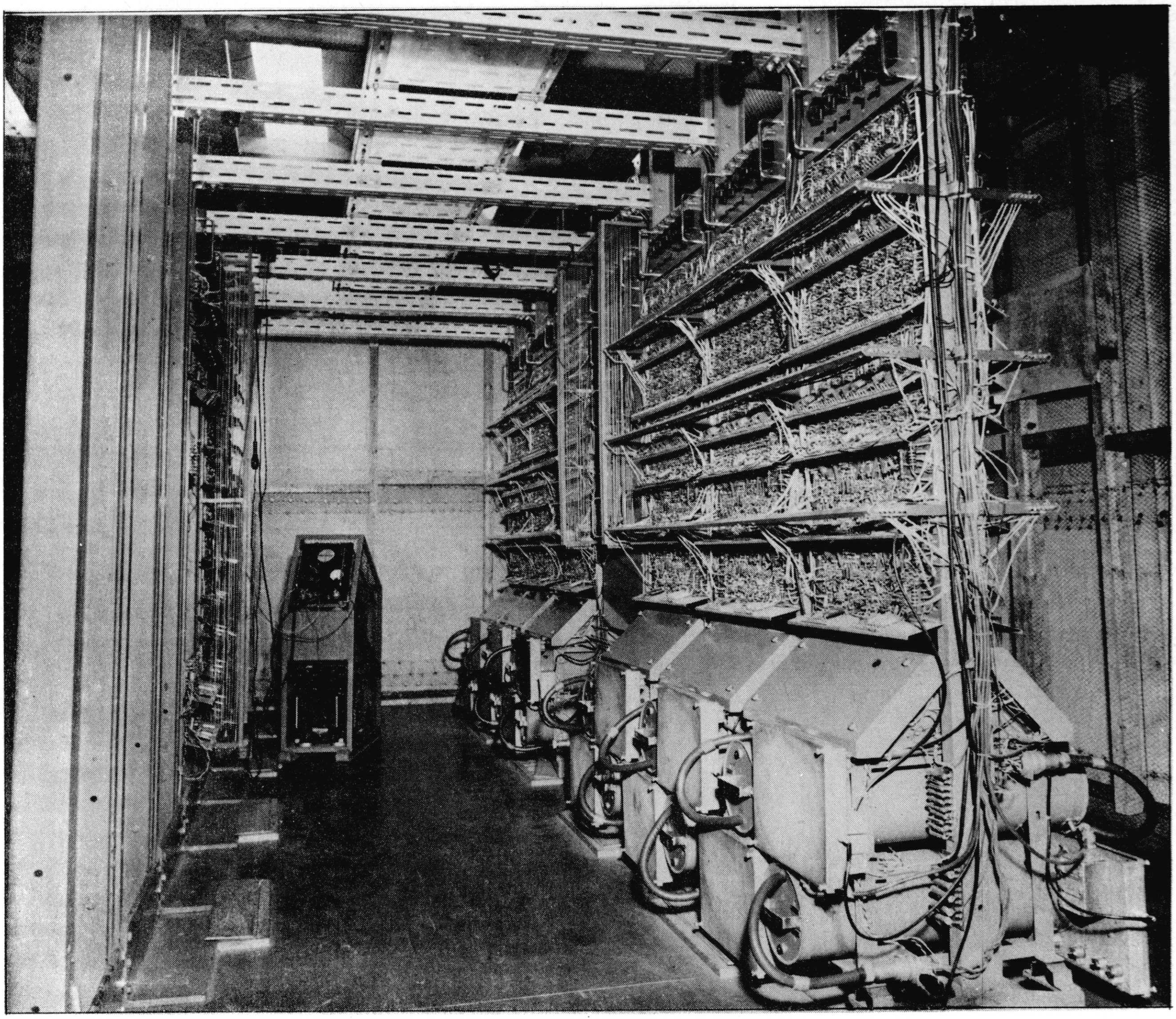
The appropriate waveforms are gated to the arithmetic and storage units, under the control of the Arithmetic Control (C) section of the computer, each of four 'beats' of the computer being 10µs in duration. Thus, a period of 40µs is normally required to read an instruction and to obey it. The time taken to transfer information between the magnetic and electrostatic stores will depend on the initial period required to reach the required address in the magnetic drum, plus the actual transfer period of 20 µs per word. Transfers will take place in blocks of 64 or 128 words.

Multiplication is controlled by a subroutine, as no built-in multiplier exists at present.



PAPER 8

FIG.1. BLOCK DIAGRAM OF COMPLETE COMPUTER



PAPER 8

Fig.2. Interior view of the TRE computer.

The Control Counter (CC), which contains the address at which to find the 'present' instruction, is of such a design that the configuration of its contents can be modified in two ways. It is a binary counter in the ordinary sense, but 'words' may be set into the counter as though it were a register, thereafter counting may continue in the ordinary way. This facility is required for instructions 10 and 11, enumerated earlier.

Except for instructions 12 and 13, each instruction takes the same time. The basic cycle which is used during computation consists of two operations, to read the next instruction, and then to obey it. The first operation is automatic except after instructions 12, 13 and 14, when a waiting period is required, and the computer is stopped. Instructions 12 and 13 involve the operation of mechanical devices, and each gives a 'go' signal on completion. The electronic part of the computer then resumes its normal operational rhythm.

Each operation consists of two parts, so that the basic cycle produces a four beat rhythm. For the sake of illustration, assume that the instruction is to add the contents of a given address in the Store (I.S. in *fig. 1)* to the contents of the Accumulator Register (ACCr).

To do this, a Shift Register (SR), is used; in design it is identical to the Accumulator Register. The result of the addition is stored temporarily in the Shift Register, and when the process of adding is complete, the Accumulator Register is cleared, and then set up to the same configuration as the Shift Register. The actual process of addition and the resetting of the Accumulator Register constitute the two 'beats' of the second half of the complete rhythm.

It was considered desirable to extract instructions from the store by means of the Shift Register and the same control system, and then to transfer the instruction from the Shift Register to an Instruction Register (IR), instead of into the Accumulator Register. This suggests that a similar two-beat cycle should be used during the first part of the complete rhythm, giving a total of four beats to a complete period of reading an instruction, and obeying that instruction.

The cycle of operations, is therefore as follows:

1. Under the control of the Control Counter, read the next instruction from the Store into the Shift Register.

2. Put the instruction into the Instruction Register.

3. Obey the instruction, putting the result, in general, into the Shift Register.

4. Complete the instruction by putting the result into the Accumulator Register or elsewhere as necessary.

It will be seen that access to the c.r.t. store is required only in beats 1 and 3. Beats 2 and 4 are then available for the necessary process of regenerating the information held in the c.r.t. store. It is not necessary to gain access to the store during beat 4, in order to write a number into the store, for example, because the computer has been designed to work with a single address code. Therefore, if an addition has been done, the result always goes into the Accumulator Register, as the store address was used to specify one of the addends. A second instruction is necessary in order to write the result into stores, and this process of writing is then carried out during the third beat of the next instruction.

By interweaving the arithmetic process in this manner it is possible to bar access to the c.r.t. during regeneration periods without serious interference with the arithmetical operation of the computer.

As the Control Counter contains the address of the next instruction, it is used only during the first beat of the cycle of four beats; it is therefore possible to modify the configuration of the Counter in any of the remaining three beats. It has been arranged to add '1' to the contents of the Control Counter during beat two. If the instruction then read out of store is *B,n* or *J,n* the configuration of the Control Counter may again be modified in beat four, ready for use in beat one of the next cycle of operations. By this means, it is unnecessary to allow for an extra '1' being added to the address contained in the instruction *B,n* or *J,n* due to the normal operation of the counter.

The Regeneration Register is a binary counter the function of which is to provide the addresses during the regeneration periods or beats, and this regeneration process is allowed to progress without interruption, irrespective of the programme.

Now, as this Register has output stages, to provide the c.r.t. store deflection amplifiers with the requisite signals (in digital form) to gain access to any address in the electrostatic stores, use is made of these stages during *Action* periods (as distinct from *Regeneration* periods) during the course of the programme. Gates are therefore provided within the Regeneration Register, and the outputs from the Control Counter are fed into them. These gates are shown in *fig. 1* as a single pole change-over switch operated by a signal from the 50 kc/s source. During beat *one* of the four beat cycle, the required address is obtained from the Control Counter, but during beat *three*, it is given by the Instruction Register. In order to meet this requirement, another gate system is provided, and is indicated in *fig. 1* by a single pole changeover switch operated by a signal from a source of 25 kc/s (50 kc/s ÷ 2), and is built into the Control Counter for convenience.

It is therefore possible to select a Regeneration address alternatively with an address derived either from the Control Counter itself or from the Instruction Register, according to the conditions set up on the gates within the Control Counter. The four beats of the cycle derive the appropriate address in the sequence:-

Beat one - from the Control Counter
Beat two - from the Regeneration Register
Beat three - from the Instruction Register
Beat four - from the Regeneration Register

The amplitude deflection voltages for the Electrostatic Stores are derived from the binary signals by means of 'current addition'. Preset fixed currents are diverted via diode gates into a common feed back resistor in each amplifier. This avoids errors in the deflection system, due to small variations in the amplitudes of the binary signals between the Regeneration Register and the Deflection Amplifier. In this way, sufficient accuracy of locating the various addresses with the c.r.t. stores has been attained, and can easily be maintained.

The system of distinction, between '0' and '1', which has been adopted in the c.r.t. Stores is the defocus-focus system. *(ref. 1)* The reader is referred particularly to a letter by G. H. Perry in *"Nature"* for the reasons for adopting this system *(ref. 2)*. One great advantage is that no restriction on 'Read-around Ratio' is placed by the system upon the mathematician, even though the machine is operating as a parallel computer with 512 digits in each c.r.t. It is expected that satisfactory operation will be obtained with 1024 digits per c.r.t. without serious modifications to the computer or cathode ray tubes.

The arithmetical processes of the Computer are carried out in a Relation unit (RU). This consists of a matrix of triodes into which are fed signals from four sources. These are

1. The Electrostatic Stores
2. The Accumulator
3. The Carry Register
4. The control section of the computer in the form of an Instruction pulse.

The Carry Register, is identical in form, to the Shift Register and the Accumulator Register, and is set up by the Carry digit from the 'carry output' of the next less significant digit Relation Unit.

No output can be derived from any Relation Unit until all the necessary input signals are present. In this way, no erroneous transitory signals are obtained. Should the function to be performed require only one digit input, as well as the *Instruction*, to the Relation Unit, then additional (erroneous) signals in the other inputs will not affect the result. The failure of a valve within the Unit will cause only the absence of a signal, not the production of a wrong digit.

The Instructions, held in turn in the Instruction Register, are interpreted by a diode matrix, and set up conditions such that the incidence of appropriate timing signals cause the instruction to be carried out.

The input of the computer is provided on punched tape, using a five digit code, pure binary in form, the letters having binary equivalent values. The initial input routine is obtained from permanent wiring on a high-speed uniselector switch, and a simple key changeover provides a check routine using only four of the fourteen instructions.

The output code differs from the input code, in that every *figure* in the decimal output requires three '1's and two '0's. All other combinations will print a character which is not a figure. The machine can identify the code being fed into it, so that output tapes may be used as input tapes.

The magnetic drum itself, four inches in diameter, is constructed of brass, and is coated with a magnetic oxide. The 24 heads used for recording and reading, are mounted on a bridge which may be fixed in position, so permitting a single turn track for each head, with a capacity of 2048 digits per head, and a mean access time of 20 ms. For the purpose of storing greater numbers of words, the bridge is released; it then oscillates as the drum rotates, giving a track length of 32 turns, with a total storage capacity of 64000 words, but at a mean access time of 1 1/4 s. For any given head, the two tracks (of one turn and of 32 turns) are completely separate, so that mutual interference is avoided; the drum thus has a capacity for storing over 1½ million 'bits'.

The power supplies are provided by four 400-cycle 3-phase alternators, each driven by its own 50-cycle, 3-phase motor. Under normal conditions, one alternator provides the L.T. power, and the remaining three alternators are connected into the H.T. system. The three alternators are separately excited, and are provided with a load sharing system, the whole computer being connected to all three alternators in Parallel. The connexion is made after rectification to avoid synchronization problems. Also, the D.C. is provided in such a manner that the total supply of 600 volts is earthed at a centre tap. In this way, any variations in voltage which occur are made to change both positive and negative supplies by the same percentage. This feature enables one to design the electronic circuits in such a manner that relatively large variations in the H.T. voltages have little or no effect on the working of the computer.

Should the L.T. alternator fail for any reason, one of the other three alternators is disconnected from the H.T. system and replaces the faulty alternator, the changeover taking 20 to 30 ms. to complete. The extra H.T. load thereby imposed on the remaining two alternators is automatically shared equally. The condition at any time, so far as the alternators are concerned, is indicated by a system of lamps.

The no-voltage release system on the power supply to the motors is so arranged that a temporary break in supply of up to one second will not cause them to trip. In order fully to maintain the power supply to the computer during such a break in the 50-cycle supply, flywheels have been fitted between the motors and the alternators.

References

1. F. C. WILLIAMS and T. KILBURN. *J. Instn. elect. Engrs.*, 1949 96, Part III, 81.

2. G. H. PERRY. *Nature*, 1951, **168**, 372.

PROGRAMMING

Chairman: Dr. E. T. Goodwin

9. Optimum Coding

by

G. G. Alway

National Physical Laboratory

Introduction: Access Time

With an automatic serial digital computer using a delay line type of memory the words (numbers or instructions) stored in the machine are immediately available only if each delay line holds just one word. For such short tanks the delay time is equivalent to one word length and the time taken for the digits of any remembered word to be transferred from the storage location in which they are stored to any other part of the machine is the time taken for these digits to run out serially. For various reasons, chiefly of economy, a machine of this kind would need to have the bulk of its memory in the form of long delay lines, each one of which would hold several words. For each long delay line only one of the words stored in it would be immediately available at any given time, and in general the operation of the machine would have to await the appearance of the required word at the end of the tank. Thus there is a so-called "access time" associated with this form of memory. It is, however, possible to design a machine so that the effect of this access time in slowing down the speed of operation is much reduced. It is the purpose of this paper to explain how this has been achieved on the Pilot Model of ACE.

The principle of optimum coding

The idea of optimum coding can best be introduced by considering a simple example, a subroutine for the computation of a square root, successive digits being obtained by the use of the ordinary Horner process. At any stage, the partial answer, the current trial digit and the remainder will have to be stored and these will be used to find the next digit of the root. If this is to be done quickly these three numbers must be immediately available to the arithmetical organs of the machine and are therefore best stored in delay lines of one word length. The machine therefore must have a certain number of short tanks. This number need, however, only be small and most of the memory can be in the form of long tanks. The best number of short delay lines will be considered later. For the square root process considered above three short tanks are enough to give short access time.

The subroutine will contain a sequence of instructions which will at each stage determine the next digit of the root, use it to form the next partial answer, determine the next trial digit, find the new remainder and finally determine when this sequence of instruction has been repeated the required number of times. The control of the machine will accept each instruction in turn and set up the necessary parts of the machine to carry out the instruction. If no time is to be lost due to the access time of the instructions each one must be immediately available to the control when it is required at the end of the previous instruction. For this it is not necessary to have the instructions stored in short delay lines, but it can be achieved by placing the instructions in such positions in the long delay lines that they are running out at the time when the control is first ready to accept them. This is the principle of optimum coding. To take advantage of this principle the control of the machine must be suitably designed.

Design of the Pilot ACE Control

We now consider the use of this principle in the square-root subroutine with reference to the Pilot Model of the ACE. A serial machine of this kind carries out three separate operations before an instruction is finally obeyed. These are:

1. The transference of the instruction from store to control. Time occupied: one word length.

2. The interpreting by control of the instruction and the setting up of the machine t obey the instruction. Time occupied on the Pilot Model; one word length.

3. The carrying out of the instruction. Time occupied: an integral number of word lengths.

The principle of optimum coding suggests that part (1) of the next instruction should occur as soon as possible after part (3) of the previous instruction. We can however, do better than this; part (1) of the next instruction can actually occur during the last word length of part (3). Thus in a sequence of instructions, for each of which part (3) occupies one word length, the machine will in alternate word time be

a. Preparing to obey an instruction.

b. Carrying out the instruction and transferring the next instruction to the control.

The instructions will therefore be obeyed at the rate of one every two word times and will therefore need to be spaced out in alternate word positions in a long delay line.

If the sequence has to be repeated, as in the square root subroutine, the first instruction will not in general be in the right position relative to the last instruction for optim coding. It is apparent, however, that if the number of instructions in the sequence is not more than half the number of words in a long tank the machine will be able to carry out the sequence in a period equal to the delay time of a long tank. Further, if the number of instructions is not more than a quarter of the number of words in a long tank we can double this rate by repeating the instructions again. This is a device which has sometimes been found useful.

Optimum coding in general

When we consider the coding of a large problem for the machine we find that most of the time taken is accounted for by the sequences of instructions which are obeyed many times. These will consist in general of the whole or parts of library subroutines for forming standard functions or performing standard processes. If we take the trouble to code only these sequences in an optimum fashion we will not be falling far short of obtaining the fastest operation of the machine. The extra labour involved in rearranging the instructions to giv optimum coding for library subroutines is thus well worth the effort.

It is evident, too, that it is necessary only for those numbers on which such sequences of instructions are operating to be immediately available, and thus to be in short tanks. In general, few such numbers are involved, and this is the reason why we can take full advantage of optimum coding with few short tanks. When this is not so, as for instance in operations on a vector of numbers which are stored in long delay lines we can still make excellen use of the optimum coding facility. The numbers can be picked out, singly or more at a tim if necessary, placed in short tanks while the required operations are performed on them, and then replaced. The instruction which does the picking out can usually be stored in a short tank so that it can be modified at the optimum rate.

On the Pilot Model there are five short tanks one word long and two delay lines holding two words only. The remainder of the store consists of eleven long delay lines of length 32 words. On the first copy (deuce) of the Pilot Model to be made by English Electric Co. Ltd., there will be four short delay lines one word long, three delay lines holding two word and one quadruple length delay line. The number of short delay lines required to make reasonably full use of the optimum coding facility will depend to some extent on the problem under investigation, but in the light of experience with the variety of problems already tackled on the Pilot Model the arrangement for the English Electric version is judged to be somewhere near the best compromise.

A further feature which affects the use made of optimum coding is the distribution of the arithmetical and logical facilities on the short delay lines. Naturally some ways of distribution are better than others. The pilot model has a two-address code, but this does not by any means imply that the number of instructions required is halved as compared with a corresponding one-address code machine. This factor, however, can be approached in many problems if there is good arrangement of arithmetical and logical facilities. The Pilot Model, for instance, has facilities which, in the root extracting sequence considered above, enable the sum of the current trial digit and the partial answer to be subtracted from the remainder, so as to produce in one instruction the number whose sign determines the next digit of the root. To make the best use of the facilities some trial and effort is often required to find the best storage locations for each number concerned, but it is worth the effort for all subroutines.

In general it seems that it is best to have additive and subtractive facilities associated with one short tank and one double length delay line. The latter can then have the multiplicative facility without a great deal of extra equipment. The logical facilities of & and $\not\equiv$ should then be associated with two more short tanks. One of these would have facilities for shifting to the right or to the left. By having the discriminatory facility associated with a special address the Pilot Model can discriminate on the contents of any store. It has been found most helpful to have two such addresses, one to determine the sign of numbers, the other to test whether they are zero or not. Lastly certain fixed words which are often required are always immediately available at certain addresses.

All this is important for making the best use of optimum coding. The number of instructions in a repeated sequence determines, as discussed above, the rate of repetition of the sequence, and in general the fewer the instructions required for the operation the quicker will the operation be performed by the machine.

With regard to multiplication there is a further important point. As in most serial machines, the Pilot Model takes a comparatively long time for multiplication, the equivalent of 2 long tanks. It is best therefore to arrange for multiplication to be carried out in parallel with other operations of the machine. It is then often possible in the coding to take advantage of the multiplication time, to do other processes which do not depend on the product awaited.

Rewards of optimum coding

Perhaps the best reward of optimum coding is that it takes advantage of fast methods of input and output. The Pilot Model uses Hollerith punched cards, reading at the rate of 200 cards per minute and punching at the rate of 100 cards per minute, only the first 32 columns of a card being available to the machine. All printing and arrangement of results is then done on a slower typewriter, at leisure, away from the machine. When the cards are used as an intermediate binary store we can put twelve words on the twelve rows of each card and thus read at the rate of 2400 words per minute, and punch at the rate of 1200 words per minute. When the cards are punched in decimal form, we can make full used of the 32 columns available only by reading 6400 decimal digits per minute or punching 3200 decimal digits per minute, the two rows of the card left over being used for indication of signs etc.

It would not be possible to make use of these speeds without optimum coding, for only then can the speed of operation of the machine enable information to be assimilated at this rate. The time between the reading passage of successive rows is about 480 word lengths and for punching about 1152 word lengths. Together with slightly longer gaps between the last row of the card and the first row of the next, this is all the time available for the assimilation of each row of information. When this time is not enough the punch and reader have to be declutched after the card, and where a large amount of information is to be taken in or given out by the machine this declutching is to be avoided as much as possible. Optimum coding enables fully punched decimal cards to be read or punched at full speed, all the conversion to and from the binary scale being effected between rows.

With matrix work, where large matrices are fed to the machine and punched out in binary form optimum coding enables all the simpler processes of linear algebra to be carried out between rows. As a simple example consider the premultiplication of a vector by a matrix too large to go into the store. The vector is stored and then the matrix is read at speed, row by row in binary form. Optimum coding enables us to use each element of the matrix as it is read, forming the scalar product of each row with the vector and storing away the elements of the resulting vector as they are formed, all between the rows of the card. The speed of the Hollerith reader is thus used to maximum advantage.

Discussion

DR. BOWDEN (Ferranti Ltd.) asked what is the effective increase in speed obtained by the use of optimum programming. He had been told in America that on a serial machine using magnetic drum storage the ratio actually obtained was the square root of that expected, and asked whether the factor 4 was a good working average.

MR. ALWAY replied that the factor of improvement depends largely upon the kind of problem; but that if 16 instructions per ms could be used the maximum factor 16 would be obtained. He agreed that 4 was what might be expected.

DR. TOCHER (Imperial College, London) said that a number of quick-access stores must evidently be provided for in the design of a machine, but that even so the machine would still have to wait occasionally for the next word from the store. The question arises whether having more short stores would cut down this waiting. Since using more short stores, carries with it the necessity of transferring words to them from the main store more frequently, there was a tendency to lose time with increasing number of short stores. A third effect would arise in the extreme example where most of the store was in the short form, as a result of having to store the components of a vector in a number of different addresses and no longer being able to perform such operations as adding the components by a single long transfer to the accumulator.

He drew a graph of time waste against the number of short stores and indicated the minimum of the sum of the three effects as the best number of short stores to use, and asked whether the number proposed for the DEUCE, being a departure from that used on the Pilot ACE, had been arrived at by objective reasoning. He also asked whether experience suggested that the minimum time waste occurred over a wide range of numbers of short stores.

MR. ALWAY replied that the answer to the first question was: Yes; some programming with a four-word tank had been done. As to the second, the optimum number of short stores was less than expected, and it should be noted that a number of two-word tanks are equivalent to twice as many one-word ones. Much more experience in programming was necessary to decide the range of values of the ratio of short to long tanks for which the minimum time waste occurred; this ratio has actually been increased in the DEUCE above that for the ACE Pilot Model.

DR. TURING (Manchester University) pointed out that the multiplier speed was of vital importance to the amount of advantage gained by optimum programming.

Two ms is many times the time taken by the frequent small operations so that if many multiplications occur, the factor 16 cannot even be approached. He asked whether any consideration had been given to the possibility of having two staticisers for the instructions, one of which would be in process of setting up while the instruction held by the other was being carried out, thus eliminating the wasted set-up minor cycle period.

MR. WILKINSON (NPL) replied that this had been considered. It would enable 32 instructions to be carried out per major cycle, did not need much more equipment, but made the problem of timing a bit tiresome. He said that the Pilot Model ACE in fact represented optimum coding at its least effective, and that the time saving factors achieved could have been much higher if the arithmetic and other facilities had been better chosen. However, multiplication time was not wasted with an automatic multiplier. If signed automatic multiplication had been provided it could have reduced the total time by doing instructions during the multiplication. He had reduced the time for some calculations in this way to the total multiplication time alone. He indicated that optimum programming gives speed just where it is wanted: during input and output. By its use five six-decimal numbers can be punched simultaneously on Hollerith cards, which could not otherwise be done. In this case a factor of 20 would be lost by not using it.

Having the arithmetic facilities such as accumulators and shifting distributed among the short tanks is a help in this respect. He did not favour the two-address code of the Pilot ACE. A serial, optimally-coded machine would preferably have a three-address code; if so, there would be half the number of instructions in this case.

MR. BLUNDELL (Ministry of Supply, R.R.D.E.) said that, speaking from 'paper' experience only in programming for a machine (MOSAIC) which was not yet working fully, he found optimum coding to be reasonably easy to work with. He regarded this as a purely mechanical process which could be done automatically by the machine itself: the instructions are put on the flow diagram in the reverse order to that in which the machine obeys them. He stated that he had a system by which this could be done by the machine in a fairly short time, provided that storage space did not run out. The use of such a system avoids wasting the programmer's time.

DR. FRIEDMAN (Cambridge University Mathematical Laboratory) referred to the use of orders of the form F n m , meaning 'operate on the word in storage location n and take next order from location m', as a 1 + 1 address system, while if this order were to mean 'add the number in location n to the number in location m and store the sum in the accumulator' it would be a genuine two-address system. He distinguished three stages in the introduction of optimum programming on the Pilot Model, (a) having short stores, (b) having instructions other than for multiplication staggered a fixed interval apart in the long stores, and (c) allowing a variable gap between the instructions. He calculated that when a variety of operations were being performed, or also the processing of large amounts of data as in the inversion of matrices, the use of (b) alone saved about 40% of waiting time otherwise wasted,

	(c) alone	"	"	65%
(a) and (b) together		"	"	75%
and (a), (b) and (c)		"	"	92%.

The latter did not exactly apply to the Pilot ACE. Only when (c) was involved was extra work required in the coding.

A one-address instruction accommodated 16 binary digits, giving two orders per word, and 700 instead of 350 on the Pilot ACE, while a 1 + 1 address instruction took 25 digits.

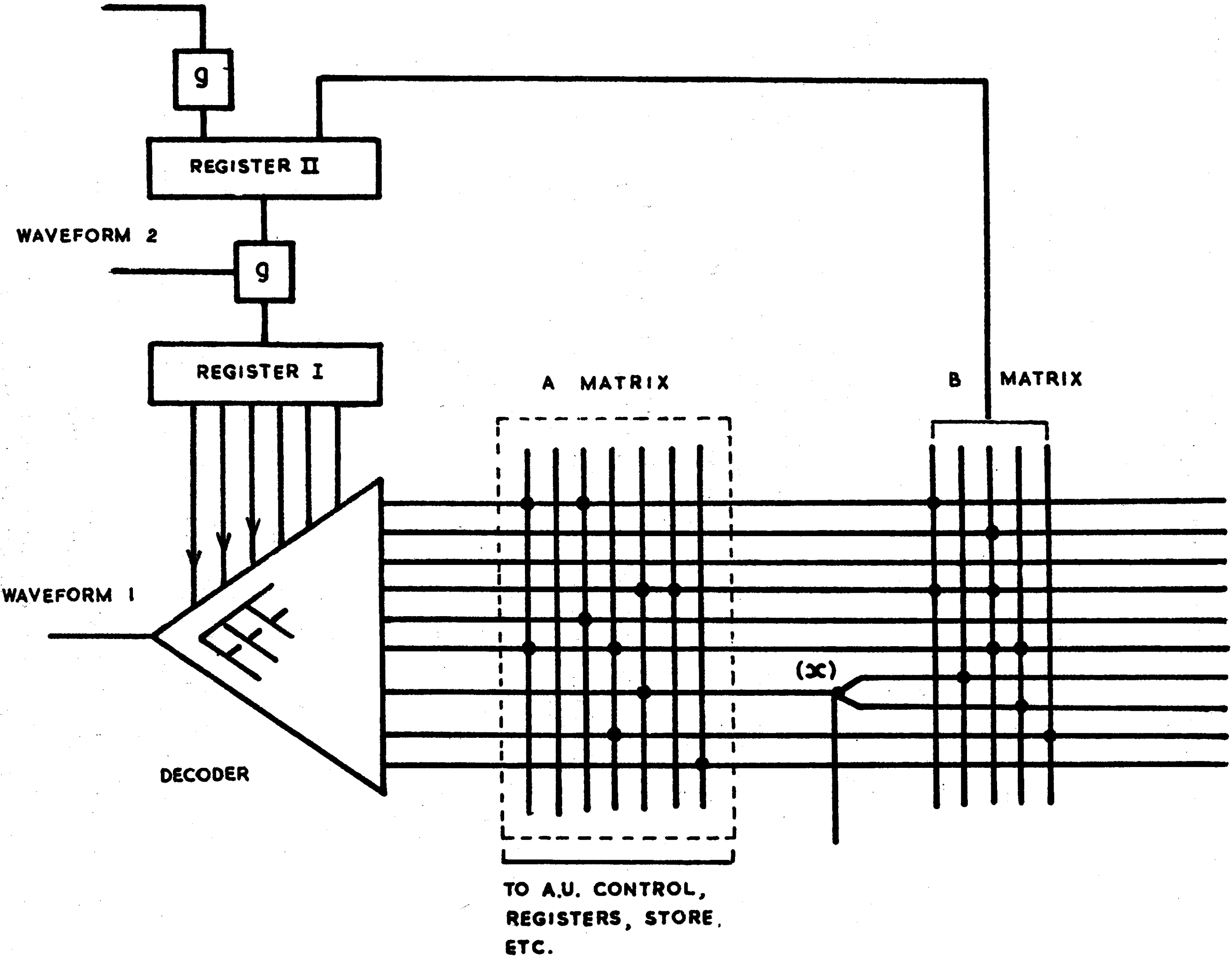
MR. WILKINSON agreed with Dr. Friedman's remarks about the ACE Pilot Model and explained that the next instruction source number only used three digits. He would not advocate the use of two instructions per word. Since the operations which were most often required to be carried out quickly related to short stores and took the form A + B → C, for example, he preferred a three-address code. One three-address instruction was worth more than two one-address instructions in subroutines although the reverse was true as regards the main programme.

MR. NEWMAN (NPL) referred to the utility of optimum programming for business machines and stated that by its use the Pilot Model ACE could convert 32 digits in a variable radix notation during the passage of a card through the reader.

DR. BENNETT (Ferranti Ltd.) indicated that Mr. Alway's remark that the time taken in coding a problem is insignificant compared with the time taken in choosing a suitable computation process, did not apply with the Manchester group. As a result of using standard processes less than 50% of the total time was spent thinking up the programme, and most of the remaining time in getting it right.

MR. ALWAY believed that his remark was true more for scientific problems than others and estimated that about 75% of the time was taken in choosing the procedure.

MR. WILKINSON said that for standard processes neither of these factors predominated, but that most of the time was spent after the coding of the problem in actually using the machine.



PAPER 10
FIG.1.

10. Microprogramming and the choice of order code

by

J. Stringer

Cambridge University Mathematical Laboratory

An automatic computer is commonly regarded as being made up of a number of more or less self-contained units.

The store holds numbers and instructions which may be read in the arithmetic unit and control sections of the machine. The arithmetic unit performs the arithmetic operations on these numbers, and transmits the results back to the store. These operations are organized by signals from the control, which is itself influenced by the instructions in the store and occasionally by the contents of the arithmetic unit.

Let us now consider the arithmetic unit in rather more detail. It may be regarded as consisting of a number of registers (storage for numbers), and equipment for shifting, adding, etc., the numbers contained in these registers. This unit will, on stimulation, perform an elementary operation such as transferring a number from one register to another, adding two numbers together, and so on. An analogy may be drawn between the small scale operations (or *micro-operations*) performed by the arithmetic unit and the large scale operation of the complete machine. Thus, in order to perform a complete calculation on the computer, a sequence of machine orders (or routine) must be performed. In the case of what may be termed the *micro-machine*, a sequence of micro-operations or *micro-routine* must be performed in order to effect a complete arithmetic operation, such as adding a number from the store to the accumulator, multiplication and so on. Pursuing this analogy a little further (without, I hope, straining it to breaking point), the computer may have several routines stored in it at one time, the calculation which is to be performed being selected by some external stimulus. In the same way, several micro-routines may be built into the machine, the arithmetic operation to be performed being selected by an external stimulus; external, in this case, to the micro-machine. Such a stimulus is, of course, the machine order itself.

There are various ways in which the micro-operations may be sequenced. One way (which is being used in the new machine being constructed at Cambridge) is shown in *fig. 1.* The arrangement consists of a decoder which routes the input pulse to one of the 2^n output lines according to the number in the n digit register I. These lines are passed into a rectifier matrix (or coder), the outputs of which are fed to the gates in various parts of the machine. Each of the matrix output lines effects a given micro-operation. Thus one or more micro-operations are performed when the decoder output line is stimulated. Each of these lines may be said to correspond to a micro-order, which calls for a selection of micro-operations. The number which must be placed in register I to select a given micro-order is called the address of the micro-order. The micro-order lines are fed also into a second matrix, called matrix B, the outputs of which are fed into a second register II. Gates are provided between registers I and II so that the contents of register II can be passed to register I. The connextions in the B matrix are arranged so that for each micro-order the address of the next micro-order in the sequence is passed into register II. Thus by applying pulses alternately to the decoder and to the gate between the registers, the machine is stepped through a sequence of micro-orders. The operations called for by these micro-orders may be arranged to give a complete arithmetic operation. If this is so, the sequence becomes a micro-routine. In general, there will be several micro-routines wired into different parts of the matrices. The whole system of micro-routines is called the microprogramme and the process of designing a *microprogramme* is, not unnaturally, known as microprogramming. As in programming on the large scale, some means must be provided for modifying the sequence of micro-orders according to the state of parts of the machine outside the microprogramme. There are two cases to be considered. Firstly, how to enter the correct micro-routine for each machine order. This is done by arranging that when the order has been extracted from the store the digits representing the function required are fed into register II instead of the matrix B outputs. They then become the address of the first micro-order in the required micro-routine. The second means of modifying the micro-order sequence is a device known as the conditional micro-order. This is used whenever an operation has to be made conditional on the state of a digit in one of the registers of the arithmetic unit, or other registers of the machine; for instance, in the micro-routine for a conditional machine order, to decide whether or not control is to be transferred, according to the sign of the accumulator. This is achieved by causing the micro-order line to branch before entering matrix B. The branch to be energised is selected by the digit being sensed. The two branches may have different addresses wired on them, so that the next micro-order in sequence depends on the required digit.

The process of microprogramming is very similar to that of the now well developed process of programming a problem for the machine. By the use of some such method of sequencing as is described above the engineering of the microprogramme is a comparatively simple matter; in fact, once the basic design of the decoder and register components has been accomplished the only remaining design is the actual position of the rectifiers in the matrix. Similarly, the redesign of the microprogramme involves very little more in the way of engineering modifications than a redistribution of the rectifier connextions. Since designing the microprogramme is equivalent to choosing the order code of the machine, this implies that the choice of order code may be left until a late stage in the design of the machine; further changes may be made to the order code at any time after the machine has been put into operation with very little change to the engineering.

The use of microprogramming has another important effect on the choice of order code. This depends on the comparative ease with which complicated operations may be synthesised from the elementary micro-operations. As a consequence of this fact, most of the engineer's objections to the user's requests for bigger and better order codes are removed. There are several kinds of instruction which programmers would find convenient, but which are not usually provided because of either prohibitive cost of the equipment involved, or fear of the unreliability of excessively complex control circuitry needed. Such facilities are of two kinds, arithmetical and organizational (or "red tape") instructions.

The class of arithmetical instructions usually consists of addition, subtraction, multiplication and shifting to right and left. In a few machines division is also provided, but I know of no electronic machine in which numbers are dealt with except with a fixed decimal or binary point. Floating point arithmetic has many advantages to the user, and is often performed by means of subroutines, in spite of the great reduction in speed which this involves. With microprogramming it is a comparatively simple matter to provide instructions which deal with numbers directly in a floating form. Other arithmetical facilities, such as calculating the square root, cosine, etc. of a number, could also find their way into the order code if it was felt that sufficient demand existed among the users of the machine.

Just as important, however, as the arithmetic operations, are the red tape parts of programmes. These often present greater difficulty to the programmer, and may take up much of the time needed to complete a problem on the machine. Most machines are provided with simple conditional and unconditional control transfer facilities, the Ferranti machine has its "B" box, and apart from this, little is provided, to my knowledge, in any existing machine. One of the most frequent kind of organizational operation is the counting of cycles of orders. This mainly involves two kinds of operation, increasing the count, and examining the result to see if sufficient repetitions have been made. A single order could be made to perform both of these functions; it may be of the form of a conditional transfer of control, the condition being the number of times it has been encountered. Another operation which is frequently needed in programmes is the calling in of closed subroutines. Here, the problem is to plant in the subroutine on entering it information which enables control to return to the main programme at the point from which it left. Again, it is possible to construct an order which will automatically fulfil these requirements. The list of facilities which would ease the organizational problems of programmers is practically endless, but the microprogrammer can nearly always find a way of providing them, provided that a sufficient variety of micro-operations are available.

It will be seen that almost the whole of the equipment, the form of which depends on the order code chosen, is concentrated in the decoder and matrix system. This means that, if the matrix is made a detachable unit, then several different order codes could be microprogrammed, the individual user choosing the one most convenient to himself by plugging in the appropriate matrix.

There is yet one more proposal which arises in the mind. Why not, instead of soldered connextions on the matrix, have connextions which can be made by the machine itself. One would then have a machine with no fixed order code. This machine would be a little difficult to use, but at least its behaviour would be interesting.

Discussion

PROF. VAN WIJNGAARDEN (Mathematisch Centrum, Amsterdam) said that in Amsterdam they have been thinking of similar schemes but from a different point of view. Their idea was to have a comparatively long instruction word and a very simple control unit which interprets the instruction by means of a looped programme.

There would be no fixed order code, but the meaning of the various digits of the instruction must be such that the normal arithmetic operations can be performed. About 20 to 30 of the normal kind of operations would be available to the programmer, but by knowing the construction of the control, he could make use of unusual operations out of the 2^{30} provided by the 30 digits of the function part of the instruction. These operations would contain repetitive cycles within them.

The effect of this scheme had been tried in certain subroutines, where it had been found that an increase in speed of 20 to 40 times could be obtained.

MR. NEWMAN (NPL) said that he could see no essential distinction between a microprogramme and an ordinary programme. He wished this to be clarified, particularly for the benefit of the engineers present. Some of the control of a computer was done by instructions and some was 'wired in'. Microprogramming was an ingenious way to achieve a 'wired in' programme.

He asked whether the micro-orders were divided into sequences so that the total of all the micro-orders in the sequences was, for example, 128 for a 7 digit matrix. In closed microprogrammes, such as square root operations, he supposed that a conditional order was necessary, and that each programme had a different conditional micro-order. Microprogramming was most useful in a machine in which access time was limiting the speed of operation. The matrix system could be regarded as a form of interpretive subroutine.

MR. STRINGER said that a good definition of microprogramming had been given by Dr. Friedmann, as acting on the realization that machine operations were composed of a number of more elementary basic operations.

The size of the matrix was not a limitation. In the future Cambridge machine there will be an 8-digit matrix giving 256 micro-orders, which proves to be adequate. The equipment required will be about 15% of the whole machine.

Even if the access time is short, microprogramming can be useful in simplifying the operation of the arithmetic unit. It also improves the speed by allowing overlapping operations.

DR. TOCHER (Imperial College, London) said that microprogramming was not such an innovation as was generally supposed, and that it may help understanding to look for connexions with the past.

Dr. Tocher then described an arithmetic unit designed in 1946 for Von Neumann's parallel computer. It contained five registers and a number of gates for transferring numbers from one register to the other. To perform a given arithmetic operation it was necessary to open the gates in the correct sequence. This was done at the time by recognizing some pattern in the sequence of operation and mechanizing this in an *ad hoc* manner. In microprogramming, the mechanization was systematic, and would allow an arbitrary pattern of gate-operations.

However, the use of microprogramming would probably result in a more complicated arithmetic unit, and this must be weighed against its advantages. Perhaps it is not necessary to extend the range of operations beyond those usually provided.

Microprogramming would be most successful in parallel machines and in any machine in which the comparatively long access time allowed complex operations to be done with extravagant inefficiency.

With regard to the suggestion that each programmer should have his own set of orders embodied in a plug-in microprogramme matrix, this, Dr. Tocher said, was an excess of scientific liberalism and might lead to serious consequences. The discipline of a fixed order code was a good thing for a group of programmers because it enabled them to share their programmes.

DR. FRIEDMANN (Cambridge University) wished to reply to Mr. Newman's implication that the matrix device was merely incidental to microprogramming. The point of microprogramming was to make the design of the arithmetic unit systematic. For any operation required, one analysed it into its basic steps and made sure the minimum equipment was available to carry out these steps. The design was then almost finished. If more complicated operations made out of the basic ones were later required by the mathematicians they could easily be added.

In reply to Dr. Tocher, he said that microprogramming was useful for a serial as well as for a parallel machine and that by microprogramming in a serial arithmetic unit using the same amount of equipment the speed of multiplication could be doubled, since one step was done per minor cycle, and other operations were four or five times as fast.

MR. WILLIS (Cambridge University Mathematical Laboratory) pointed out that microprogramming was welcomed by maintenance engineers because it made it unnecessary to follow the working of all kinds of operation. Only the various basic steps need be examined.

PROFESSOR HARTREE (Cambridge University) said that in having order codes easily modifiable it was not proposed that each programmer should have his own order code. However for different kinds of calculation a different order code would be desirable. Investigation in group theory, for example, could be facilitated by an order code different from that employed in arithmetic work.

MR. BEALE (Admiralty Research Laboratory) asked to what extent any single micro-order is used in a number of different microprogrammes. MR. STRINGER replied that it was to about the same extent as in ordinary programming.

MR. WOODGER (NPL) suggested that a fixed order code should be used during initial input, which could be modified at once by the programme to suit the problem being solved.

MR. DOUGLAS (Cambridge University Mathematical Laboratory) said he wished to make explicit some ideas that had been implicit in most of what had been said. While arithmetic operations are usually considered as basic by programmers, to the engineer it is simpler operations, namely shifts and transfers, that are basic. Microprogramming is the same process as programming, but with engineering operations as the basis. The distinction between microprogramming and programming is a question of drawing a line between the machine's control and the programmer's control.

II. Conversion Routines

by

E. N. Mutch and S. Gill

Cambridge University Mathematical Laboratory

Introduction

This paper describes methods which are being studied at Cambridge for making the best possible use of the magnetic tape auxiliary store which is now being completed for use with the EDSAC. The central feature of the proposals is a 'conversion routine' for converting programmes into precise machine orders from a form which is more convenient for the programmer to use when writing out his programmes. This routine follows similar lines to the one now being used on Whirlwind I at the Massachusetts Institute of Technology, the principles of which are discussed by Adams and by Carr *(ref. 1 and 2)*. Because of certain differences between the speeds and facilities of the two machines the EDSAC conversion routine will be somewhat simpler than that of the Whirlwind; it will also be less comprehensive.

The Cambridge University Mathematical Laboratory pioneered the automatic processing of programmes when in the autumn of 1949 an initial input routine for reading programmes into the machine was adopted which, besides converting addresses from decimal to binary form and assembling together the function and address parts of each order, also inserted given numerical values for *(see ref. 3)* additive parameters. An essential feature of the new scheme is the extension of this idea to provide the 'floating address' facility *(ref. 4 and 5)*.

The scheme will also owe some inspiration to the work of A. E. Glennie who has constructed a 'translation routine' for the Manchester University computer which allows programmes to be prepared to a large extent in a normal mathematical notation.

Magnetic auxiliary stores

Before discussing programme conversion the characteristics of a magnetic auxiliary store will be studied from the programmer's viewpoint. Magnetic forms of storage can provide large storage capacities at a reasonable cost, and they are virtually permanent, *i.e.* they do not require repeated processing to preserve the information. By virtue of the large storage capacity, problems can be solved which require the simultaneous retention within the machine of large amounts of numerical data. Moreover large programmes can be stored, and several programmes may be held in the store in addition to the one being executed. The fact that magnetically recorded information has a high degree of permanence means that programmes, subroutines and partly processed numerical data can be left in the machine for several days independently of whether or not the machine is kept switched on and in working order. It would not be appropriate to discuss here the relative merits of magnetic drums and tapes, but it should be mentioned that tape has the advantage of large capacity but the disadvantage of long access time. The capacity may be extended indefinitely by providing more reels of tape. This has the further advantage that programmers may be provided with reels of tape for their individual use thus providing a safeguard against accidental destruction of one programmer's material by another. In the case of a magnetic drum such privacy can only be obtained by rather more elaborate precautions.

The permanence of magnetic storage is of particular importance in a laboratory where a large number of different computing projects are in progress at the same time. Usually each project entails a series of computations with the same programme; this programme may be retained in the magnetic store over a period of days or weeks while the machine is busy on other problems, and may be brought back into action comparatively quickly when required. Standard test programmes may also be kept in the magnetic store, so that the machine may be put through any test at a moment's notice without the operator having to search for the right piece of punched tape or set of cards. The EDSAC will select and obey any routine in its magnetic tape store if the serial number of the routine is dialled on an ordinary telephone dial attached to the machine (the selection being carried out by a special initial input routine which can be obtained automatically by pressing a button).

For several reasons the use of an elaborate 'conversion routine' is made much more attractive by the provision of a large permanent store. Firstly, a programme which is required for frequent use need only be converted once and then kept in the permanent store in its converted form, so that it is not a serious disadvantage if the conversion process itself takes some time. Secondly, common library subroutines can be kept in the permanent store so that they can be incorporated automatically in any programme without having to be fed to the machine each time they are required. The conversion routine itself can also be kept in the permanent store. Thirdly, a considerable amount of storage space is required in order to carry out a conversion process including all the facilities which might now be considered desirable. Since the conversion routine envisaged may easily occupy more than twice as much storage space as that provided in the high speed store it will be necessary to carry out the conversion in several stages. At least two stages of conversion would be necessary in any case owing to the nature of the floating address facility which will now be described more fully.

Floating address system

A small experimental conversion routine providing the floating address facility within one part (the master routine) of a programme has been described by Wilkes *(ref. 5)*. It is now proposed to extend this facility to all parts of a programme. Briefly, it enables the programmer to refer to any word in a programme by means of a label or tag attached to it arbitrarily by the programmer, instead of by its address in the store. Thus, for example, a number appearing in the calculation might be labelled 'a_3'. The programmer could then write simply 'A a_3' to denote the operation of adding this number into the accumulator, without having to specify just where the number is located in the machine.

This is very similar to the preset parameter facility provided by the initial orders now in use in the EDSAC *(ref. 3)*, but with one basic difference. The value of a preset parameter must always be set in advance, that is it must be explicitly specified at a point on the input tape preceding all points at which it is used. This restriction is removed in the case of floating addresses. The programmer will be able to indicate to which word a floating address label belongs simply by writing the label immediately before the word concerned. Orders making reference to that floating address may occur in any part of the programme. For example, part of a programme might by written thus:

```
     A   a3

     A   2

     T   a3

     H   24

a3)  C   50

     T   a3
```

Here the label 'a_3' is attached to the fifth order, hence the first, third and sixth orders are taken to refer to the fifth order. After conversion the above orders might appear as follows:

Location in store	Order
100	A 104
101	A 2
102	T 104
103	H 24
104	C 50
105	T 104

The restriction is made that each label must consist of a letter followed by a decimal number (written for convenience as a lower case letter and a numerical suffix). This restriction could be removed so long as precautions were taken to avoid any possible ambiguity (*e.g.* confusion between labels and actual addresses in the store), but there seems no real need for any other type of label. A total of several thousand different labels will be possible of which a maximum of about 200 may be used in any one programme. This represents a considerable increase over the thirteen preset parameters at present available in the EDSAC.

Storage and assembly of subroutines

Another feature of the proposed scheme is that a library subroutine stored on the magnetic tape will be referred to in the written form of the programme by means of special words chosen to describe the operation carried out by the subroutine. The conversion programme will have to recognize these words and insert the appropriate subroutine. For example, part of a programme as written might be:

A a_5
A d_1
T a_5
sin a_5 to s_5
H s_5
V b_2

During conversion, a subroutine would be inserted for calculating a sine, and would be so adjusted as to take the argument from the location labelled a_5 and place the result in that labelled s_5. It is anticipated that most subroutines would, as here, be simply inserted into the programme where required instead of being placed in another part of the store (*i.e.* they would be 'open' rather than 'closed' - *(ref. 6)*). Frequently the only advantage of a closed subroutine compared with an open one is that the former leaves the master routine in one compact sequence and thus simplifies cross-reference between various parts of the master routine. With floating addresses all cross references are simple and this point, therefore, does not arise. The only other advantage of a closed subroutine is that where necessary it may be used at several stages in a calculation without having to be copied out several times in the store; storage space is thereby saved. However, with a magnetic store the available storage space is no longer of critical importance and it will often do no harm to copy a subroutine two or three times. To deal with cases where a closed subroutine is still desirable, it is proposed to make it possible to convert an open subroutine into the closed form if suitable indications are given when the programme is written.

Synthetic orders

Present EDSAC library subroutines tend to be fairly large. The use of very small subroutines has so far been discouraged by the fact that the proportion of orders required to form the link order is appreciable, and by the fact that it is almost as tedious to copy a short piece of punched tape as a long one. Both of these disadvantages can be removed in the scheme now proposed, and it is, therefore, to be expected that a number of very short subroutines will appear.

The use of such subroutines in a way very similar to that now proposed had been suggested by Wilkes *(ref. 5)* before the auxiliary store was available. He proposed that copies of these subroutines should be held in the high speed store during the input of the programme; being small there would be little objection to allocating the necessary storage space to them. They could then be copied into the programme automatically, where required, by means of a small specialized conversion routine. He gave the name 'synthetic orders' to the tape entries which called for the insertion of these subroutines.

It is likely that, even with a magnetic tape attached to the machine, it will be worth while keeping some small commonly used subroutines in the high-speed store during the conversion process to avoid the necessity of frequent access to the magnetic tape. It will be convenient to continue to refer to the relevant tape entries as 'synthetic orders'. It is envisaged that the operation of division, for example, would be specified by a synthetic order. Other facilities such as the carrying out of simple counting operations and possibly the extraction of square roots may also be provided by synthetic orders.

Conclusion

In 1945 the first large automatic electronic digital computer, the ENIAC, had been successfully constructed and had proved the practicability of building such large machines. There followed a period of some years, during which designs were being rapidly moulded and remoulded, before another machine actually appeared.

Programming today seems to be in a similar stage. It has been proved that programmes *can* be constructed; it has also become apparent that a sufficiently comprehensive conversion routine with access to an exhaustive library of subroutines can revolutionize the subject of programming. However, a great deal of capital (in the form of programming time) must be invested in such a system before it can be made to work, and the revolution is, therefore, bound to be a slow one.

With systems of such complexity there is bound to be a delay between the discovery of a new principle and its ultimate exploitation, and hence there is a danger that a system which has taken many months to complete may become rapidly outdated. However, a programme is easier to alter than a machine, and if sufficient care is taken in the design of a conversion routine it should not be necessary to scrap the whole routine merely because parts of it are out of date. This point should be borne in mind when the routine is planned. One facility which it is hoped to include in the EDSAC conversion routine as soon as possible is the ability to specify numbers in a variety of ways - *e.g.* as .9375, 15/16 or $3.10.2^{-5}$. Later, attention will be given to the inclusion of operations on numbers in floating-point and multi-length forms.

References

1. ADAMS, C. W. Small problems on large computers. *Proceedings of the Association for Computing Machinery*, Pittsburgh, May, 1952, 99. Published by Richard Rimbach Associates, Pittsburgh.
2. CARR, J. W. Progress of the Whirlwind computer towards an automatic programming procedure. *Proceedings of the Association for Computing Machinery*, Pittsburgh, May, 1952, 237. Published by Richard Rimbach Associates, Pittsburgh.
3. WHEELER, D. J. Programme organization and initial orders for the EDSAC. *Proc. roy. Soc., A*, 1950, 202, 573.
4. WILKES, M. V. Review of electronic digital computers. *Joint AIEE-IRE Computer Conference, AIEE, New York*, Dec., 1951, 114. (Report of a discussion).
5. WILKES, M. V. The use of a 'floating address' system for orders in an automatic digital computer. *Proc. Camb. phil. Soc.* 1953, 49, 84.
6. WILKES, M. V., WHEELER, D. J. and GILL, S. *The preparation of programs for an electronic digital computer.* Addison-Wesley Press, Cambridge, Mass., 1951.

Discussion

MR. GLENNIE (Ministry of Supply, Fort Halstead) described a conversion routine he devised for use on the Manchester computer. Algebraic symbols are used for the manipulation of numbers and an equation describes the arithmetical operation and transfer to be performed, *e.g.* $+ a + b + ab \rightarrow c$.

This constitutes a multi-address code with any number of terms on the left hand side. Organization can be specified by English, *e.g.* SUBROUTINE 3. Transfers of control are effected by writing CONTROL A and at some other point in the programme this is designated by writing ENTRY A. The machine will then work out the transfer of control required. Input time is reduced since the number of orders is reduced by the more compact method of writing. This factor is offset by the time taken to determine the meaning of the symbols.

DR. BENNETT (Ferranti Ltd.) gave his experience of the application of an interpretive routine for matrix operations on MADAM. Coding is made much easier by its use but in order to carry out a set of instructions, about ten times as many instructions are required to interpret and carry out the orders as would be required for the normal code. The process of interpretation is also time-consuming so it has been found useful to have a mixture of codes. Those orders in particular which require a lot of interpretation can be stored in the machine, while others such as organizational orders can be interpreted.

DR. TURING (Manchester University) defined a conversion routine as one in which interpretation is done once and for all at the beginning of a programme, *e.g.* floating binary or floating decimal. One of the difficulties about these routines is the need for introduction of a scale factor in the arithmetical operations.

MR. WILKINSON (NPL) referred to the well-known shortage of programmers. The reason was now quite clear: as soon as a programmer gained any experience he began programming programmes.

MR. DAVIES (NPL) put forward a plea for the term 'translation routine'. DR. BENNETT pointed out that this term was already in use for a particular class of routines, *e.g.* those applied to the translation of teleprinter tape. MR. TOOTHILL (Military College of Science, Shrivenham) thereupon suggested that Dr. Bennett use the term 'transliteration routine'.

DR. GILL (Cambridge University Mathematical Laboratory) pointed out that changes in a programme are necessary after the initial writing out. These changes can take place before the programmer does the writing, or before punching, and before reading into the machine. Changes can take place when the programme is encountered during use, or the control unit can make the changes. The convenience of the stage at which changes are made depends on several factors. On the one hand the converted orders occupy more space whereas the orders requiring interpretation take longer to carry out and require instructions for this process.

It seems advisable to concentrate less on the ability to write, say

$$+ a + b + ab \longrightarrow c$$

as it is relatively easy for the programmer to write

```
A a
A b
H a
V b
T c.
```

What does require more attention is the placing of numbers in the store. A conversion routine used at Cambridge allows algebraic symbols to be used and the routine then allocates these to suitable positions. Another facility is that symbols can be used for transfer of control or for planting instructions.

DR. BROOKER (Manchester University) divided the process of putting a problem on an automatic machine into the stages: mathematics and numerical method, the layout of store and the instructions, the coding, and finally the machine operation. Conversion and interpretive routines influence the last two items. This is generally the least part of the work. The symbols +, - etc. are not very useful unless floated. It is quite often the case that the use of interpretive routines takes as long as doing the job. DR. BROOKER said there is too much programming of programmes and that problems should be tackled by mathematics and reduced to simple processes such as occur in linear algebra to which a great many problems at Manchester had been found to lead, and in the solution of differential equations where it was best to reduce them to a set of first order equations and simplify the formulation.

DR. GOODWIN (NPL) said he was interested to hear DR. BROOKER'S remark that linear algebra constituted such a large part of their work. This confirmed NPL's experience. Nevertheless he did not agree with him on the best method of solving differential equations.

DR. WILKES (Cambridge University Mathematical Laboratory) said that all machines used conversion routines in some form or other and to various extents. The designing of a conversion routine is analogous to designing a computing machine system, and a conversion routine is as much a part of a machine as any other part. In this respect we must avoid changing it too frequently. The amount of conversion which we do depends on the number and type of programmers using the machine.

MR. BEALE (Admiralty Research Laboratory) asked whether conversion routines would be more complicated on a machine on which optimum coding was possible and MR. WILKINSON replied that this depended to a large extent on the size and nature of the conversion routine.

12. Getting Programmes Right

by

S. Gill

Cambridge University Mathematical Laboratory

This paper is concerned not with the detection of errors, nor with the treatment of machine faults, nor with the problem of distinguishing between machine faults and mistakes in programmes, but with the diagnosis of a mistake in a programme when it is known to exist.

The situation confronting the programmer is that the machine has exhibited a behaviour which, according to his reasoning, is not in accordance with the instructions he gave. Assuming that the machine is working correctly, and that the punching of the programme has been thoroughly checked, this means that there is a flaw in his reasoning; the problem is to find it. Without making further use of the machine, there are two methods of approach to this problem: to review the chain of argument leading from the statement of the programme to the expected result, looking for possible flaws, or to work backwards from the actual result looking for points at which the error might have arisen out of the intended behaviour. The former is often unprofitable because the programmer is blinded by his own conviction that the programme should work, and the latter is often limited by the fact that insufficient information was given in the behaviour actually displayed by the machine to form a stating point for any kind of reasoning. In spite of these difficulties, a mistake can often be diagnosed by re-examination of the programme in the light of the evidence provided by the machine's actual behaviour.

Occasionally, however, the chain of reasoning is too tortuous to be checked without considerable effort, and no flaw is apparent. In such cases the search is made very much easier if information can be obtained about the machine's actual behaviour which will assist in reconstructing the course of events and, if possible, provide checks at intermediate points in the calculation. For this purpose the machine itself must be used.

Several methods have now been established; *ref.1* describes how they are applied at Cambridge. There are two main lines of attack: either the programme may be inserted into the machine unchanged and the mode of operation of the machine changed to enable the desired information to be obtained, or the machine may be run normally and applied to a different programme or to a modified form of the original programme.

Changing the mode of operation of the machine usually means slowing or stopping it while the operator examines its contents. This replaces an electronic time-scale by a human one; moreover the operator may be handicapped by having to make use of monitoring devices which present the contents of the machine in an unfamiliar or inconvenient form. If the operator is alert and thoroughly conversant with the programme, he may be able to track down a mistake quite rapidly by this process. There is a considerable danger, however, that he may succeed only in consuming a great deal of machine time without finding the mistake.

Programmed devices for locating mistakes are of two kinds: the so-called 'post mortem' routines, which are used after a programme has come to a halt and which cause the contents of relevant parts of the store to be punched or printed out for inspection, and the 'checking routines' which can be attached to a programme to cause the output of extra information during the execution of the programme to assist in error diagnosis. 'Post mortem' routines are simple and convenient to use. At Cambridge they are kept available (on punched tape) for immediate use and are a standard treatment in cases of programme failure. Their limitation is that since they only indicate the state of the machine when the original programme was halted, vital information about the cause of trouble may have been destroyed by the last few operations of the programme. In the case of the EDSAC, and also of the ACE Pilot Model, this limitation is minimised by the fact that not all possible words can be obeyed as instructions and allow the machine to continue operating; in the event of serious dislocation of a programme it is highly probable that a word will soon be encountered which will bring the machine to a halt.

Checking routines can provide a record of the course of events from the beginning to the end of the calculation, and are therefore more powerful than post mortems which provide only a static picture at the end of the calculation. A variety of different checking routines is desirable, to provide different types of information about the progress of a calculation. One type used at Cambridge records the function letter of each order as it is executed, so enabling the programmer to check the sequence of execution of orders. Another provides information about the state of the accumulator at suitable moments. Refined versions provide for a suspension of checking during certain parts of a programme. Such routines can be extremely valuable in difficult cases.

The amount of information obtained by programmed methods of checking is limited mainly by the rate at which the machine can punch or print. This is much greater than the rate at which readings can be obtained by eye. Programmed methods can result in wasted machine time if used indiscriminately, but this danger is not so serious when using programmed methods as it is when the programme is examined by stopping and looking at the machine. If the programmer is already on the track of a mistake and requires only one small item of information to clinch the matter, he may well obtain it by looking at the machine. Otherwise, the surest line of attack is to employ programmed methods. These have a great advantage in that they provide a tidy, readable, permanent record which can be studied at leisure.

However, the choice depends partly on the design of the machine, in particular on the facilities available for stopping the machine and for running it in 'slow motion', and on the extent to which the instruction code lends itself to the construction of checking routines. The EDSAC, as it was first constructed, permitted the use of particularly simple and convenient checking routines, but did not have very versatile means for controlling the machine by hand.

The simplicity of a checking routine depends on the simplicity with which the operation of the control unit of the machine can be specified in terms of its own instructions. This depends primarily on the simplicity of the rules by which the control unit proceeds from one instruction to the next. Recently some additional orders were provided in the EDSAC code for producing 'transfers of control'; one effect of these was to necessitate new checking routines which were about twice as big as the old ones.

Various devices exist for facilitating the task of studying the operation of the machine while controlling it by hand. One of the most successful is the employment of special instructions for stopping the machine. The Ferranti Mark I Computer at Manchester University has two instructions, either of which will stop the machine if a certain switch is thrown; otherwise the programme proceeds at full speed. This instruction is inserted at points in the programme at which an examination of the contents of the machine might be required. In the Pilot ACE, any instruction may be made to stop the machine by making a certain digit zero.

Other devices include means of inserting words into the machine manually, of causing the machine to obey any required part of the programme, and of causing it to obey a small number of instructions at high speed. Devices of this kind, even if they are not often used in finding mistakes in programmes, can be extremely useful in tracing programme-sensitive machine faults. It would be possible to modify the machine physically to cause it to provide automatically the information given by certain checking routines, without the necessity of modifying the programme. However, besides calling for extra equipment, such an arrangement would be inflexible compared with the use of checking routines.

The choice of programme-checking methods depends on so many factors that it is difficult to see how it will be affected by future developments in programming and in machine design.

The introduction of a large auxiliary store opens up several possibilities for checking routines. It enables checking routines and post mortem routines to be kept permanently in the machine so that they may be brought into action at a moment's notice. It also makes possible the use of a new and extremely useful kind of post mortem routine, devised by Professor C. W. Adams and his group working at the Massachusetts Institute of Technology. This routine provides information about those words in a programme which have changed during its execution, and about those words only; the programmer thus has his attention directed immediately to the very points which interest him most. This is made possible only by the fact that a copy of the whole programme, in its original state, can be accommodated within the auxiliary store.

On the other hand, large storage capacities will lead in the course of time to very elaborate conversion routines. Programmes in the future may be written in a very different language from that in which the machine executes them. It is to be hoped (and it is indeed one of the objects of using a conversion routine) that many of the tiresome blunders that occur in present-day programmes will be avoided when programmes can be written in a language in which the programmer feels more at home. However, any mistake that the programmer does commit may prove more difficult to find because it will not be detected until the programme has been converted into machine language. Before he can find the cause of the trouble the programmer will then either have to investigate the conversion process or enlist the aid of checking routines which will 'reconvert' and provide him with diagnostic information in his own language.

References

1. GILL, S. *The diagnosis of mistakes in programmes on the EDSAC.* *Proc. roy. Soc. A.* 1951, 206, 538.

2. WILKES, M. V., WHEELER, D. J. and GILL, S. *The preparation of programs for an electronic digital computer.* Addison-Wesley Press, Cambridge, Mass. 1951.

Discussion

MR. CAMINER, (J. Lyons & Co. Ltd.) said he was not in agreement with the suggestion made by the author that programmes should be put on the machine as soon as possible. Programmes should be tested by another user to save wasting machine time which, to a commercial user was important. On LEO considerable use had been made of post-mortem tapes and more recently this had been supplemented by photographing the cathode ray tube presentation of the contents of the store. In dealing with the large programmes which arise in commercial work they had found it essential to plan the programmes and test them in pieces.

DR. MILLER, (Cambridge University Mathematical Laboratory) said that he was a desk computer by training and this probably influenced his attitude. He found that he was always quite sure of the method he wanted to use but made many slips in coding it. He was in agreement with putting programmes on the machine as soon as possible but tried to do it when he had the machine to himself. He made use of post-mortems but preferred to examine the contents of the store (peeping).

MR. WILKINSON, (NPL) said he had made use of test programmes but had found them very disappointing. On the other hand he found post-mortems surprisingly valuable. This was because the latter on the ACE Pilot Model produced a limited amount of data in compact form, while the product of the former was a little overwhelming. He preferred to run through programmes again very carefully before putting them on the machine, because immediately after producing a programme he was more completely in mastery of its details than later on. It might be several hours before the machine was available. When testing a programme on the machine he found it convenient to have a programme broken into a number of small sections by means of stoppers. By this means errors in the programme could quickly be traced to a section and this could then be studied one step at a time. On the ACE Pilot this was a fairly pleasant process because the next instruction to be obeyed was clearly displayed on a set of lights. Provided the display was good enough "peeping" was the most useful aid to getting programmes right. When a fault had been found, it was better to leave the machine and to put it right at leisure, but here human nature was the main enemy. It was tempting to a person to avail himself of the full time he had been allocated on the machine.

MR. STRACHEY, (National Research Development Corporation) said that errors in programming were of three types. First there were slips which meant that location and function numbers were wrong in instructions, due to an error in copying or punching or a similar lapse. These were best found on the machine and it was convenient to be able to alter the instructions on the machine. Facilities for doing this were very good on the Ferranti computer but as they involved a formidable battery of switches the operator needed to be expert in their use. Then there were mistakes where the user had intended to do the right thing but the programmed instructions fell short of the intentions. These were tiresome because they usually could not be cleared up without adding instructions. Instructions were in blocks containing a fixed number on the Ferranti machine and, as each block had usually been filled completely, this made mistakes a good deal more bother than they would otherwise have been. Mistakes should be rectified away from the machine because clear thinking was essential. Finally there were errors in design. Nothing much could be done about these but with experience they should decrease in number. He had found the most economical use of the machine for programme testing was possible if several programmes were tested at one time so that when a mistake was encountered in one this could be noted and another proceeded with.

MR. BENNET (Ferranti Ltd.) said that in his experience slips caused the most trouble. On the Ferranti machine they originally used a subroutine for printing out instructions as they were obeyed. They had found entering the subroutine was inconvenient and now had a switch on the machine which caused the code letter of successive instructions to be printed automatically. Referring to the issue of redundancy in the code he said that at Manchester there was no redundancy and he had found this a great nuisance.

DR. SLUTZ, (National Bureau of Standards) said that originally SEAC had been equipped with a switch by means of which the printing of the next instruction and its results could be forced. This was used so much that it was succeeded by a routine to print out instructions and results. The routine could be made to print all instructions or only those with a special digit. Its use of storage space was found inconvenient and as a result hardware was added which made this automatic. It could print out either all instructions or these with a coded digit only. Originally instructions were printed out but because this was so slow magnetic wire output was later used instead. In the new machine this will be omitted and a one valve facility has been added which will print out the content of any selected storage, or any group of 8, storage location every time it is changed.

DR. TOCHER (Imperial College, London) said that it took time to do anything on a relay machine, so time was important. This ruled out the use of test programmes. If the error was due to instructions in the wrong order on his relay machine then this must happen at discrimination points. At all branch points he suggested therefore that the branch taken should be indicated by printing a plus or minus sign. This may not be quite sufficient and it might be necessary to print numbers from some of the stores as well.

DESIGN

Chairman: Mr. J. H. Wilkinson

13. Special Requirements for Commercial or Administrative Applications

by

T. R. Thompson

J. Lyons & Company Limited

Introduction

This paper is concerned with the applications of automatic computers to what in the commercial and industrial world is called "office" or "clerical" work and is limited to procedures related mainly to numerical information. In this kind of work a prescribed "procedure" is carried out on given numerical information - the "data" - to produce a specified statement or statements of "results" which are required for later use. The precise steps to be taken at any particular stage of the procedure may depend on circumstances, *e.g.* the nature of the data or of intermediate results obtained during an earlier stage.

FUNCTIONAL ANALYSIS OF CLERICAL WORK

Examples of clerical work

For those who are not familiar with clerical work a few simple examples are given in appendices A to F of some well known aspects of it which will be used throughout this paper for illustrating the points made.

Appendix A gives a payroll statement and shows how the wages for time worked by an employee are calculated by multiplying his hours by the rate of pay; after adding a bonus the gross pay for the week is obtained; this is added to the wages earned in previous weeks of the income tax year to give the gross pay to date; deducting the amount of tax free pay to date, the taxable pay to date is obtained and on this the tax to date is assessed; deducting tax previously deducted from the tax due to date, gives the tax to be deducted for the current week; a deduction must also be made from the gross pay in respect of the National Insurance contribution and there may be other deductions for National Savings, Social Club Contributions and the like; the gross pay less all the deductions gives the net pay.

Appendix B depicts a sales invoice by means of which a supplier tells the customer his indebtedness in respect of goods that have been sold to him; it is what is familiarly known as a "bill".

Appendix C shows how a running statement of a customer's indebtedness is maintained by a supplier; it is usually called a sales ledger account.

Appendix D is a sales analysis statement showing the total value of goods of various categories sold to customers in various districts compared with the corresponding figures for the previous year; this kind of statement is used by a sales manager to study the effects of his sales policy.

Appendix E is a form of account which may be used for keeping control of stocks; for each number of items of stocks it shows the quantity held at the beginning of a period, the quantity received into stock during the period, and the quantity that remains at the end; the sum of the opening stock and the receipts less the quantity remaining is what is called the consumption of the commodity; this is compared with the quantity known to have been issued during the period to give the net difference over or short in stock.

Appendix F provides a simple statement of the expenditure, revenue and profitability of a manufacturing and selling department of a concern; this statement, known as a "Trading Analysis", is used by Management for controlling operations of the department; the statement results from what is known as "cost accounting" which is a procedure by which expenditure is related to the purpose for which it is incurred.

It will be clear that most of this kind of clerical work occurs at regular intervals; thus the payroll may be prepared either weekly or monthly, the sales invoices either daily or weekly; and the trading analysis may be prepared weekly, monthly or quarterly. Each time a job is done the same procedure is followed but much of the data is different.

Fundamental types of operation

Doing Arithmetic

As would be expected this is a very important operation in clerical procedures which are mainly concerned with numerical information; but it is by no means the only important one.

Selecting a particular item of information

A very important type of operation is the selection of the particular item that is required at any step in the procedure. For instance, before the total of an invoice can be posted to a customer's sales ledger account the particular account has to be selected from amongst the accounts for a large number of customers.

Sorting information

Often it is convenient to sort documents containing information into a different sequence before the next step in the procedure is carried out. Thus to calculate the gross wages for a series of employees both the hours worked and the rate of pay are required for each employee. The rates of pay for the various employees, which remain fairly constant, may be given in one statement, and the hours worked, which may vary from week to week, may be on "clock cards", one for each employee. In such a case it is clearly desirable to sort the clock cards so that the employees occur in the same sequence as they do on the statement of the rates of pay.

Comparing items of information

Another kind of operation which often occurs is the comparison of corresponding items of information to ensure that they agree. For instance, in the example of the payroll quoted above, it is necessary to check that the employee's code number on each clock card agrees with the next code number on the statement of rates of pay.

Assembling and arranging the results

The results produced by the clerical procedure must be arranged in a form suitable for the action which is subsequently to be taken.

Storing information

Much information has to be stored for comparitively long periods between the carrying out of a job in one period and the carrying out of the next; or it may only be required for future reference. Although this, the filing system, may not appear to be an actual part of a clerical procedure it is an important consideration for on it depends how quickly reference can be made to information when required.

The frequency with which these different fundamental operations occur varies greatly from job to job; for instance in a payroll job the emphasis is on arithmetical work whereas in the sales ledger work the most important consideration is that of selecting the right customer's account before an entry is made.

Nature of results produced

Some of the results produced as part of one clerical job may be required as data for another, which may be a repetition of the same job for a subsequent period using the same procedure, or it may be quite a different one using a different procedure. The running totals for the income tax year of the gross pay and income tax for each employee produced as part of the results of the payroll job and required as data in the payroll job next week are an example of the former. An example of the latter is the amount of the gross pay earned by all the employees working in a factory for a week, produced as part of the payroll job, which may be required as data in the "cost accounting" job for the factory. Such results may be conveniently termed "carried forward" results.

Where the action arising from results necessitates their being read by some person or persons, they must be produced in readable form: manuscript, typewritten, or printed; as for instance a payslip for an employee, an invoice for a customer, an "analysis of sales" for a sales manager, and a "trading analysis" for Management.

Both "carried forward" and "printed" results may be required from the same job; in fact more than one separate result of either kind may be required. For example from the payroll job the following separate results might be required:

1. a printed payslip, to give to the employee;
2. a printed payroll statement giving particulars for all employees, to be used for office reference and for audit purposes;
3. a statement of the gross pay earned and the income tax due for the income tax year to date, to be carried forward for the payroll job to the following week;
4. a total of the gross pay earned by all the employees in a factory, to be carried forward to the "cost accounting" job for that factory.

Sometimes, by producing more than one copy, the same form of statement may be used for more than one purpose.

Nature of data used

Much of the data used in clerical jobs arises from the "carried forward" results of previous clerical jobs, This data is therefore termed "brought forward" data.

A great deal of other data is of a semi-permanent character; examples of this are rates of pay of employees and prices of items sold. This type of data may be called "permanent" data as it usually remains constant over a period. Means must, however, be provided for amending permanent data when a change is required; this involves the use of "amendment" data which, when applied to the permanent data, produces new permanent data to be used until further notice.

A fourth class of data is that which applies only to the current job in hand, as for instance the hours worked in the current week for the payroll or the goods to be supplied to a customer's order. This type of data may be termed "current" data. The information for this class of data very often originates outside the office; it may arise in the factory, for instance the time worked by an employee may be recorded either automatically by a time recording machine on a clock card or by a foreman on an attendance sheet or, to take another example, an order may be made out by a customer on a preprinted sheet. The form used for an initial record must therefore be convenient, not only for the way it is used in the clerical procedure, but also for the person who has to make the original record and the circumstances in which he makes it. Since this form of data is used once only it is important that the method of recording it should be an economical one.

Completion of Jobs to a Tight Schedule

The results from some clerical jobs are a vital factor in the practical running of the concern and often circumstances make it necessary for the results to be available according to a tight schedule. Thus in order to pay the wages on a Friday, the payroll job must be finished by say Thursday at the latest; when sales invoices accompany goods to be supplied they must be completed promptly if the delivery is not to be delayed.

USE OF AN AUTOMATIC CALCULATOR TO DO CLERICAL JOBS

Theoretically any job can be done by an automatic calculator, provided the procedure can be fully prescribed. Whether it can be done economically must, however, depend on circumstances.

Circumstances in which clerical work is suitable for an automatic calculator

This question cannot be dealt with adequately in a short paper but some important considerations may be mentioned.

> The total volume of work which needs to be done using a particular procedure must be appreciable; it will clearly not be economical to prepare a programme for a job if it is used very little.
>
> Also the amount of work to be carried out at any one time must be appreciable for it will not be worthwhile feeding the programme into the calculator if the amount of work to be done is small. To make this possible there must be an adequate time lag between the time of the receipt of the data and the time when the results are required. Thus if orders are received over the telephone from a customer and have to be executed almost at once, an automatic calculator could not suitably be used for invoicing the customer. It is only if a large batch of data can be accumulated and fed as a sizeable job to the calculator that a job can be economical.
>
> Again the volume of calculation (or other manipulation) that has to be carried out on a given batch of data must be significant. It will clearly not be economical to feed a large batch of information into the calculator if very little is done with it when it gets there.

Advantage of doing clerical work by automatic calculator

In large offices there is a great deal of clerical work which satisfies these conditions and much of this work has to be done by semi-skilled people who carry out repetitive operations on varying data all day long, possibly using small calculating machines and other devices for the purpose. A great deal of effort is expended in reading information from one place and recording it in another, as for instance, reading from documents in order to press the keys of a calculator, and then reading the result shown on the dials in order to write it elsewhere.

Documents also often have to be passed from hand to hand or from machine to machine throughout the procedure involving a great deal of handling and carrying of papers. Often many of the intermediate results, recorded as part of the procedure, are not actually needed in the required final results.

The ideal way of using an automatic calculator is to carry out complete clerical procedures from beginning to end, starting with data as initially recorded outside the office and producing results which are either required for use outside the office or which are to be used in some other procedure, making permanent records of any information only if it will be required for some such purpose. In this way no unnecessary records are kept of intermediate results, handling of the documents holding intermediate results is eliminated, and all routine processes are done automatically.

One may regard the part the calculator plays in such a scheme as similar to that of a single clerk who carries out the whole of a clerical procedure by reading the data as he requires it, by carrying out mentally the arithmetic and logical processes, and by writing down the results required for future use. Acting in this way the calculator is performing the function of a routine office and it is for this reason that the Lyons project was named "LEO" - Lyons Electronic Office.

It is believed that the greatest gain in such a system will be not so much in the speed of doing arithmetical operations but in the speed with which the figures are selected and transferred, and in the elimination of unnecessary recording and handling of intermediate results.

The organization of clerical work according to groups of results

The result numbers produced during most clerical jobs form themselves into natural groups. Thus, in the sales invoicing job, the invoice for each customer can be regarded as containing a separate group of result numbers. Similarly, in the payroll job, the payslip for any employee contains a group of result numbers. Each group of result numbers is produced by a given burst of calculations, using the related group of data numbers. The fact that clerical data naturally tends to divide itself into natural groups can be exploited with advantage when clerical work is done by means of an automatic calculator. Before any burst of calculations takes place the data specifically required for that burst can be fed in and as soon as the calculations are finished the results produced can be recorded. The data fed in for the next set of calculations can be put into the calculator in the storage compartments formerly used for holding the data for the previous set, and the results as they are produced can be stored in the compartments used to hold the results of the previous set of calculations.

Not all the data is, of course, specific to a set of calculations; some of it may be common to all or very many of the calculations and must therefore be fed in before the first set of calculations takes place. Thus in the sales invoicing job the price of any commodity may be required as data in order to make the calculations for any one invoice, so a list of prices for all commodities must be fed in before making the calculations for the first invoice.

Similarly some of the results may not be related to any particular set of calculations but to all or to a large part of them. In the sales invoicing job the totals of invoices for all customers in different areas and for the whole country may be required; running totals can be maintained by adding, as part of the procedure for any invoice, the total of the invoice to any required totals to date.

Clerical Jobs involving higher mathematical processes

There is an increasing tendency in offices of large organizations to do clerical work involving higher mathematical processes, as for instance, a statistical treatment of the data. A special example is the process which has come to be known as "linear programming" in connexion with Production Planning in factories. From a knowledge of the extent to which a number of components of material, labour, machine time, etc. are required to produce a unit of each of a number of items of production and of any factors limiting the total amount of each component available, a calculation is required of the number of units which should be produced of each item in order to yield, say, the maximum profit that can be attained. This kind of job only differs from the clerical job in the amount of mathematics involved in programming the job. Once the problem has been fully visualized and the required mathematical procedure devised, the general problem of carrying out the job on an automatic calculator is no different from that for other clerical jobs.

COMPARISON BETWEEN CLERICAL JOBS DONE BY A CALCULATOR AND MATHEMATICAL JOBS

Ratio of the volume of calculations to the volume of input and output

The fundamental difference between mathematical and clerical jobs is the ratio of the volume of calculations to the volume of input and output. For very many mathematical jobs there is only a very small amount of data and a very small amount of results but a great deal of calculation. On a clerical job there is almost always a very large volume of input and output and a relatively small amount of calculation on each batch of data. This has a great bearing on the amount of equipment needed and on the way the job is organized and programmed.

Size of Programmes

Programmes for doing clerical jobs tend to be very much bigger than those for mathematical jobs. There are two quite separate reasons for this.

The first reason is that once the data has been taken into the store it is economical to carry out as many operations as possible on it; there are often a number of distinct processes which can be carried out on the same data or on intermediate results arising from the data. For example when preparing sales invoices, it may also be convenient to post sales ledger accounts and to analyse sales according to kinds of commodities, districts, and so forth. The programme should as far as possible be made to cover not one procedure, but a whole series of inter-related procedures, and the programmes become, therefore, correspondingly longer.

Another reason why programmes are longer is that commercial and administrative requirements are less rational than mathematical ones; the clerical procedures of commerce have often evolved over a long period as a result of a variety of influences. The determining factor may sometimes be a matter of commercial custom, or it may be the need to comply with some Government regulation. In devising programmes to do a clerical job on an automatic calculator many anomalies and apparently unnecessary elaborations are brought to light, and should as far as possible be eliminated.

In spite of all efforts, however, there will always be exceptional circumstances which must be taken care of in the programme; if an exception occurs only rarely the programme may perhaps best deal with it by recording the circumstances of the exception and leaving the precise procedure to be carried out subsequently; but where an exception is relatively common a sequence of orders must be included in the programme to deal with the special procedure required. For instance in a large payroll job there are always some employees who are taxed on the gross pay in the current week instead of on the total gross pay to date. Provision must therefore be made for both alternative and normal procedure. Out of a total of 1710 orders in the programme for doing the payroll job on LEO, 780 orders are contained in sequences which deal with more or less exceptional circumstances.

The size of programmes naturally varies considerably from job to job but a programme of 1500 orders is not exceptional.

Equipment needed to do clerical jobs

The size of the main store (immediately accessible during any set of calculations) must be large enough to hold:

1. the programme of orders;
2. the data common to all sets of calculations;
3. the data specific to any one set of calculations;
4. the intermediate results;
5. the results for any set of calculations;
6. the results common to all the sets of calculations.

Since the number of each of these items is very often large in comparison with the usual run of mathematical jobs, the main store of a calculator requires to be of a greater capacity when it is to do clerical jobs. The number of store compartments required for these various purposes differs from job to job. As an illustration, in doing the payroll job with LEO the number of compartments required for the different purposes listed is as follows:

1. 1710, from which 86 are subsequently used for purpose 6
2. 49
3. 52
4. 50
5. 92
6. 138

2091
less 86 used for two purposes
2005 Net Capacity required

The means whereby data is fed into the calculator must be such that the whole of the data for any set of calculations may be presented to it in a time comparable with the time taken to do any other operation.

Similarly, the means of recording results must be such that the whole of the result numbers may be cleared from the calculator in a time comparable with the time taken to do any other operation.

Reliability of Calculators

Since clerical work must often be completed to a tight schedule it is important that a calculator should not remain out of action for more than say two or three hours at any one time. Also, in any one week, the total available working time should never fall significantly below the amount of time required to do the scheduled work, so that the amount of time spent on fault finding should be not much greater than 5%.

Preparation of programmes for clerical jobs

Because the methods whereby an automatic calculator does the job are so different from the methods usually employed in an office it is important to express the requirements for any job in the simplest possible terms as though it were to be done by a single clerk without recourse to any special office devices. From the statement of requirements the general plan of doing the job by an automatic calculator is drawn up.

The person who prepares this general plan need not be a mathematician with academic qualifications, though he probably needs a mathematical type of brain. He should have a thorough knowledge of the nature of clerical work and he should also be able to visualise it in terms of an automatic calculator. The plan should be set down in the form of a flow chart, showing in what sequence the main steps of the procedure must be carried out. It should also show at what points the different kinds of data are to be used and at what points the different results are to be produced.

When the general plan is agreed a more detailed analysis of each step must be made so as to obtain an exact specification of the calculations and other manipulations required. The specification should show the nature, source, and volume of each kind of data and the nature, destination, and volume of the different kinds of results. Particulars should be obtained of all exceptional circumstances and the frequency with which they occur. Again a detailed chart showing the sequence of operations for each step is a useful way of expressing the requirements. Draft forms should be prepared, showing the way in which each type of data can most conveniently be recorded and presented to the calculator, and also draft forms showing the way results require to be recorded.

It is only when these detailed requirements have been settled that the coding of the programme can proceed. Again the coder does not require any high degree of mathematical skill but he does need to be able to think in terms of symbols. Someone who has reached School Certificate standard in mathematics and enjoyed doing mathematics in school should be able to do the coding satisfactorily.

Each step in the procedure may be coded as an independent stage in the programme in the same way as open and closed subroutines are coded separately in mathematical programmes. A closed subroutine may be useful where the same kind of procedure is required in different stages of the programme, for instance in taking in data or in carrying out division by means of a programme. Normally, however, it does not pay to use without alteration "library" subroutines, either open or closed, as part of the programme. It is better to adapt the "library" subroutines to meet the special circumstances of the particular programme so as to get the best use of the store or to shorten the time taken. This is well worth while since the programme will be used many times. The fitting together of the different stages and the entering of subroutines is, of course, precisely similar to what is needed for a mathematical programme.

As in mathematical programmes, the sequences of orders are largely repetitive but many fewer iterative sequences are used. Another difference is that, whereas in mathematical programmes the results of one main cycle form the basis for the calculations for the next cycle, in the clerical programme the successive cycles are largely independent, each new cycle having for its basis a new batch of data.

The nature of the sequences that occur is not very different from mathematical ones and the operations that the calculator is required to carry out - addition, multiplication, transfer, change of sequence, etc. - are precisely the same. In fact, with the exceptions of some additional orders in LEO, EDSAC and LEO have precisely the same order code.

SPECIAL REQUIREMENTS FOR DEALING WITH DATA

In organising clerical work by means of an automatic calculator, the way data is handled is a very important factor, since the volume is so large. The different types of data referred to in the section headed: "Nature of Data used" originate from different sources, and even for one type the data for a job may originate from more than one source. Ideally therefore batches of data of different types and from different sources should be fed into the calculator along separate input channels. Where there are more than three or four distinct types and/or sources then, to keep the number of channels to a minimum, it may be better to blend data from two (or more) sources, before feeding it to the calculator, so that both batches can be fed together through a single channel.

Because the data is derived from many sources the same treatment cannot be applied to all forms of data indiscriminately; the different types will therefore be considered under the three headings of current and amendment data, permanent data, and carry forward data.

Current and amendment data

As has been mentioned before both of these kinds of data are originated outside the automatic office as such. Sometimes it may be possible to have a means whereby the original record can be made mechanically; for example, attendance can be recorded with a time clock, or a production record may be kept by an automatic counter attached to a machine. Generally, however, such records are made by hand. When this is so it is necessary for the original record to be transcribed to some medium that can be read automatically before it can be fed to a calculator; the transcription must also be checked. The cost of transcribing and checking data in this way is expensive, besides introducing an extra possibility of error.

In order to achieve full economies in doing clerical work by means of automatic calculators some way must be found of eliminating the need for transcription. One way would be to provide those who make the initial record with a device which will produce a record which can not only be read visually, but also automatically, as for instance a device which produces a punched card as well as a typed record. It is likely, however, that such devices would prove too expensive to be provided to everyone who makes records, for example to all retail customers. The method which seems really economic is to devise a means of writing on paper, that can be read both in the ordinary way and by some scanning device which will either feed the information directly into the calculator or alternatively automatically create an intermediate record.

Permanent Data

Initially this information also originates outside the automatic office, but the problem of transcribing it to a medium which can be automatically read is not so important because, once it has been transcribed and checked, it is fed to the calculator many times without further transcription. In many types of clerical job, for instance sales records, a very large volume of permanent data may have to be held for a long period and for any run of the programme only part of it will be required. The main problem with this data is to find a suitable medium for storing. It must be a medium -

1. from which required information can be quickly selected as required;
2. which can be amended as required;
3. from which information cannot be accidentally erased;
4. which does not involve much physical handling;
5. which does not take up too much space;
6. which is not too expensive.

One particularly difficult question is the danger that, when information is being inserted in the store of permanent data or being copied from it, other items of information may accidentally and unknowingly be interfered with.

The most suitable medium may not be the same in all circumstances. It will depend on the volume of information to be kept, the way it is originated, and the way it is required in the course of the programme. Various media are being developed, *e.g.* punched cards, punched tapes, photographic films and magnetic tapes, wires and drums. All offer possibilities but as yet none appears to have provided a fully satisfactory solution. Theoretically a magnetic drum offers the best solution because of ease of access to any item of information and because it requires less physical handling, but the number of drums required for permanent storage in even a modest commercial installation might make this form of storage extremely costly.

Brought forward data

In this case the problem is one of recording the carried forward results as they are produced by the calculator as part of a previous job. Magnetic tape, magnetic wire, punched tape, or punched cards, all provide a suitable medium. In this case, however, the storage is only required for a relatively short period and the information may only be used once.

For this type of data a recording medium such as magnetic tape or wire which can be used over and over again should eventually prove the most suitable. For the time being, however, punched cards have the advantage that the equipment which handles them is already well established and reliable. Moreover equipment exists for sorting the cards into any order required and particular cards can be selected from the rest when all are not required. It is also readily possible to make amendments before the data is used in the job by punching more holes or by inserting a new card.

Sorting and Selecting of Data

Although automatic calculators as already constructed have the facility of sorting and selecting information, that facility is restricted to where the number of items is reasonably small, *i.e.* hundreds, or with a magnetic drum, tens of thousands. Since in clerical work the number of items of data is often very large, additional facilities are needed for sorting and selecting. It may not be desirable to control these processes inside the main calculator itself, for were this to be done it might mean that for a large proportion of the total working time the equipment which carried out other functions would be idle while the sorting and selection was taking place. It may therefore be better to develop auxiliary equipment for the special purpose of sorting and selecting data as required so that it is ready for a subsequent job to be carried out on an automatic calculator.

SPECIAL REQUIREMENTS FOR DEALING WITH RESULTS

The most important matter to be considered here is, as already stated, to clear from the calculator a group of results as soon as it is complete. We are concerned firstly with the results which are actually to be read and must therefore be printed in some way, and secondly with the results which are to be carried forward in some form for a subsequent job on the calculator.

Printed results

There are two possibilities: the first is to print directly by means of a high-speed printer; the second is to make at high speed an intermediate record of the results as they are produced and subsequently to read this record and feed the results more slowly to a battery of slow-speed printers. The advantage of direct printing is that it saves "double-handling" information and it permits an immediate scrutiny of the results.

Printers, whether high- or low-speed, may print either one character at a time or the whole (or part) of a line at a time. The printer which prints individual characters has the advantage that the equipment is only required to accept signals for one character at any one instant and is therefore much cheaper. On the other hand, since a line-a-time printer can print many characters at once, the mechanical action can be relatively slower. This is an important consideration since the limiting factor in printing results direct from the calculator is likely to be speeds of mechanical action. For this reason it would appear that line-a-time printers are likely to prove best for this purpose.

In clerical work the layout of result numbers is a very important consideration, especially when the documents containing the information are sent to people outside the organization operating the computer, as for instance when sales invoices are sent to customers. The printing arrangements must therefore provide the necessary diversity of layout and punctuation. Not only does the form of results differ from one job to another, but often for a single job it is necessary to record result numbers on two or more different kinds of form. For instance, in sales invoicing the layouts of the invoice for the customers and of the sales dissection statement for the sales department would be totally different.

In order to draw attention to exceptional circumstances which have been found during the execution of the job it will also be necessary to print slips giving details of these abnormal circumstances.

Where results have to be printed on several different kinds of form it may be necessary to use two separate output channels. If a direct printer is to be used for printing on pre-printed forms this is the only method possible. But if the volume of results to be printed on two or more kinds of form is not large enough to justify a separate channel it will be better to record them together on an intermediate record using one output channel; the information from the intermediate record can subsequently be split and fed to two or more separate printers each supplied with the appropriate pre-printed forms. Thus there might be one channel connected to a direct printer and another channel recording miscellaneous results on an intermediate record to be separated as they are fed to separate slow-speed printers.

Carried forward results

Results to be carried forward may also be required for more than one purpose, for example in the factory payroll the gross pay and the tax due to date will be required for the same job the following week; at the same time the gross pay for the current week may have to be used in a cost accounting job for charging against the factory cost account. Again, it may be desirable to use a separate output channel for the results for each of the two purposes or it may be preferable to use only one channel and separate the results subsequently.

The medium on which the carry forward results are to be recorded must be one which is suitable and convenient not only for the recording process but also for feeding the information into the calculator as brought forward data for the subsequent job.

Again magnetic tape or wire and punched tape or cards provide reasonably suitable means. Magnetic tape and wire are probably most convenient since they can be used over and over again.

CONCLUSIONS

Provided suitable and consistently reliable equipment can be provided there is no reason why commercial clerical work should not be dealt with effectively by means of automatic calculators. The degree of reliability must be very much greater than is required for doing mathematical jobs owing to the need for carrying out commercial jobs to a tight schedule. Existing machines fall a long way below what is wanted here. As far as the arithmetical and manipulative functions of an automatic calculator are concerned the present types of calculator are quite satisfactory. They should however be associated more or less intimately with other apparatus capable of carrying out the following functions. Some of this equipment already exists but it is not in all cases entirely suitable for the purpose, being either too slow, too expensive, or too unreliable.

1. To save transcribing and checking current and amendment data, a means must be devised for recording it in the first place either mechanically or in manuscript in such a way that it can be read automatically as well as visually; reading equipment must be designed and built for feeding the data so recorded to the calculator.

2. A cheap method of storing very large quantities of data for long periods is required; it must be such that information can easily and quickly be selected from it and inserted in it, but there must be adequate safeguards to prevent it being altered unintentionally.

3. Some device is needed which will efficiently sort and select data prior to its being fed to an automatic calculator; this may be combined advantageously with the store mentioned in 2 above.

4. It should be possible to feed the data into the calculator at a speed comparable with other operations that can be carried out inside it.

5. Similarly it should be possible to extract results from the calculator at a speed comparable with that at which other operations are performed inside it.

6. High-speed printers are required which will print information reliably and satisfactorily at higher speeds than existing tabulators and yet be equally flexible in regard to layout, use of special stationery, etc.

APPENDIX A.

PAYROLL

Department :- PACKING

Week ending 5TH DECEMBER 52

CODE NO	NAME	HOURLY RATE	HOURS WORKED	TIME WAGES	BONUS	GROSS WAGES THIS WEEK	GROSS WAGES EARNED IN PREVIOUS WEEKS	GROSS WAGES TO DATE	TAX FREE PAY TO DATE	TAXABLE PAY TO DATE	TAX DUE TO DATE	TAX DEDUCTED IN PREVIOUS WEEKS	TAX TO BE DEDUCTED THIS WEEK	NATIONAL INSURANCE DEDUCTION	OTHER DEDUCTIONS	TOTAL DEDUCTIONS	NET PAY
1	ADAMS. J.	2/7	56	7. 4. 8	1. 16. 4	9. 1. 0	284. 0. 8	293. 1. 8	113. 15. 0	179. 6. 8	29. 18. 0	28. 18. 0	1. 0. 0	5. 9	5	1. 6. 2	7. 14. 10
2	WILSON. M	2/9½	50	6. 19. 7	1. 12. 0	8. 11. 7	241. 14. 0	250. 5. 7	208. 5. 0	42. 0. 7	4. 18. 0	4. 11. 0	7. 0	5. 9	4. 11	17. 8	7. 13. 11
3	HENSON. P	2/8	56	7. 9. 4	1. 12. 6	9. 1. 10	322. 5. 7	331. 7. 5	110. 5. 0	221. 2. 5	39. 4. 0	38. 3. 0	1. 1. 0	5. 9	-	1. 6. 9	7. 15. 1
4	BARRETT. W.	2/10	58	8. 4. 4	1. 15. 6	9. 19. 10	335. 6. 1	345. 5. 11	143. 10. 0	201. 15. 11	34. 14. 0	33. 13. 0	1. 1. 0	5. 9	5. 0	1. 11. 9	8. 8. 1
5	ROBSON L.	2/11	59½	8. 13. 7	1. 15. 3	10. 8. 10	339. 1. 2	349. 10. 0	346. 10. 0	3. 0. 0	7. 0	5. 0	2. 0	5. 9	17. 0	1. 4. 9	9. 4. 1
6	ROSS E.	2/7	42¼	5. 9. 2	13. 9	6. 2. 11	244. 13. 10	250. 16. 9	201. 5. 0	49. 11. 9	5. 15. 0	5. 14. 0	1. 0	5. 9	5	7. 2	5. 15. 9

APPENDIX B

INVOICE

B 2005

WOODWORKERS LTD.
MAIN ROAD,
BRISTOL.

DR. TO A. G. BROWN LTD., LIVERPOOL

Order No. C105

Date: 28.3.53.

	20 Gr. No. 6 Woodscrews @ 9d. Gr.		15	0
	24 " " 8 " " 1/- "	1	4	0
	24 " " 9 " " 1/3 "	1	10	0
	36 " " 11 " " 1/9 "	3	3	0
	36 " " 12 " " 2/- "	3	12	0
	24 " " 15 " " 2/6 "	3	0	0
	12 " " 18 " " 3/- "	1	16	0
		15	0	0
	Less 5% Discount		15	0
		14	5	
	Add Carriage		7	6
		£14	12	6

APPENDIX C

LEDGER ACCOUNT

Dr. A. G. BROWN LTD. Cr.

Date		£.	s.	d.	Date		£.	s.	d.
1953					1953				
Apr. 1	To Balance B/Fwd.	12	10	0					
8	" Goods	15	0	0	Apr. 5	By Cash	12	10	0
17	" Goods	12	10	0	14	" Goods Returned	3	15	0
24	" Goods	17	15	0	21	" Cash	15	0	0
					25	" Cash	12	10	0
					30	" Balance C/Fwd.	14	0	0
		57	15	0			57	15	0
May 1	To Balance B/Fwd.	14	0	0					

APPENDIX D

SALES ANALYSIS Q/E. 31st March, 1953

DISTRICT	TEA		COFFEE		COCOA		TOTAL SALES	
	THIS YEAR	LAST YEAR	THIS YEAR	LAST YEAR	THIS YEAR	LAST YEAR	THIS YEAR	LAST YEAR
	£	£	£	£	£	£	£	£
Southern Counties	15 500	15 300	9 300	8 900	9 800	9 750	34 600	33 950
Midland "	14 700	15 200	8 000	7 700	10 300	10 150	33 000	33 050
Northern "	17 200	16 500	7 500	7 350	7 050	6 800	31 750	30 650
Scotland	10 300	9 900	5 600	5 500	3 250	3 350	19 150	18 750
Wales	8 000	8 100	3 500	3 200	2 950	2 820	14 450	14 120
TOTAL	65 700	65 000	33 900	32 650	33 350	32 870	132 950	130 520

APPENDIX E

STOCK CONTROL

BREAD DESPATCH 14th March, 1953

Description	Unit	Opening Stock	Received			Sub Total	Closing Stock	Consumption	Total issues	Stock Balance	
										over	short
3½lb Sandwich	ea.	100	150	200	150	600	75	525	523	-	2
3½lb Sandwich - Sliced 30	"	75	100	125	75	375	50	325	325	-	-
3½lb Sandwich - Sliced 40	"	50	100	150	100	400	40	360	364	4	-
Sandwich Hovis - Sliced 50	"	25	75	100	50	250	35	215	215	-	-
1¾lb Bread & Butter	"	50	120	150	90	410	60	350	348	-	2
1¾lb Tin - Sliced	"	75	130	150	125	480	45	435	430	-	5
Long Bread & Butter	"	25	50	75	40	190	30	160	160	-	-
Long Bread & Butter - Sliced 72	"	25	80	120	60	285	25	260	260	-	-
Long Toast - Sliced 34	"	20	75	100	50	245	30	215	212	-	3
4½lb Sandwich Hovis	"	20	40	80	40	180	10	170	170	-	-
4½lb Sandwich Hovis - Sliced 26	"	15	50	50	50	165	10	155	155	-	-
Vibro Hovis Loaves	"	15	30	40	30	115	10	105	105	-	-
Breadcrumbs - Coarse	lbs.	10	-	40	-	50	5	45	45	-	-
Breadcrumbs - Fine	lbs.	5	-	20	-	25	5	20	20	-	-
Yeast	ozs.	10	10	-	-	20	7	13	13	-	-

TRADING ANALYSIS

APPENDIX F
No.

ELECTRIC MOTOR DEPT.	Summ. Ref.	Q/E March 1953

Ref.		MARCH 1953	MARCH 1952				
		£	£	X	X	X	X
				% on Sales			
	NET PROFIT SUMMARY						
	Sales	156 483	147 291	100.0	100.0		
	Prime Costs						
	Factory Cost of Sales	- 39 434	- 42 273	- 25.2	- 28.7		
	Distribution	- 16 744	- 16 202	- 10.7	- 11.0		
	Selling	- 26 289	- 23 861	- 16.8	- 16.2		
	Gross Profit	74 016	64 955	47.3	44.1		
	Overheads						
	Factory	- 21 721	- 19 462	- 13.9	- 13.2		
	Selling	- 32 697	- 31 278	- 20.9	- 21.2		
	General	- 16 216	- 15 896	- 10.4	- 10.8		
	Net Profit (or Loss)	3 382	- 1 681	2.1	- 1.1		
	SALES DISSECTION	£	£	% on Total Sales			
	Government Contracts	39 747	39 576	25.4	26.9		
	Manufacturing Concerns	43 979	41 298	28.1	28.0		
	Dealers	72 757	66 417	46.5	45.1		
		156 483	147 291	100.0	100.0		
	BALANCE OF FINISHED STOCK (At Selling Value)	£	£				
	Production	162 627	146 483				
	Add Opening Stock	44 652	26 291				
		207 279	172 774				
	Less Closing Stock	50 796	25 483				
	Sales	156 483	147 291				
	FACTORY PRIME COST DISSECTION	£	£	% on Selling Val. of Prod.			
	Wages	28 216	28 313	17.3	19.3		
	Materials	9 723	9 083	6.0	6.2		
	Sundries	4 614	4 510	2.8	3.1		
	Total Cost of Production	42 553	41 906	26.1	28.6		
	DISTRIBUTION PRIME COST DISSECTION						
	etc.						
	etc.						

Discussion

MR. KITZ (English Electric Co. Ltd.) asked about speed. The author had said that a high input and output speed was necessary. How high should it be and how many decimal digits per second were needed? The author had stressed reliability. What sort of checks were applied?

MR. THOMPSON said he was worried about dependability more than about reliability. The accuracy obtained to the present time was as good as or better than that of clerical workers. They were satisfied there was no difficulty about obtaining accuracy. They did not favour built-in checks and thought they were not worth the cost. Instead they relied on sum checks.

Answering the first question he said that Appendix A of the paper indicated the number of digits of output in a typical piece of work and said that the results for each man should be punched out in a small fraction of a second.

MR. HAHN (Bristol Aeroplane Co. Ltd.) asked if pre-sorting of data was considered essential for the input system. Punched cards could be sorted but tape could not. Use of the latter, therefore, led to a need for large storage space. He also wanted to know how much storage space was required for commercial purposes and whether it was not better in commerce to use a number of small machines rather than one very large machine.

MR. THOMPSON replied that a lot of data was already sorted or sorted itself to some extent, but he thought there should be a means of sorting so as to feed the machine with sorted data.

LEO had 2048 stores of 18 binary digits. This was considered large enough. The number of machines was largely a question of economics. He did not favour a battery of machines in parallel because of the resulting mass of intermediate results. It would be good to have a battery of machines such as the one machine they had, but these machines were very costly.

DR. VAJDA (Admiralty Research Laboratory) asked whether the records produced by the machine were adequate for legal requirements.

Mr. THOMPSON replied that 20 years ago the question could not be answered with any certainty. At that time the courts did not accept, without question, records printed by Burroughs machines. Nowadays they were much more reasonable. They were fairly confident on this point, but the matter had not yet been tested in the courts.

DR. BOWDEN (Ferranti Ltd.) mentioned a large piece of work that had been considered by Ferranti. The amount of data was astonishingly large. The data amounted to 10^{11} binary digits doled out at the rate of 10^8 digits per month or 20 miles of tape per month. It was also a very difficult problem to programme. A typical piece of clerical work was more difficult to programme than, for example, a set of differential equations with some non-linear terms. In programming clerical work for a machine it was first essential to consider how it was done at present. It was, in fact, very difficult indeed to find out precisely what functions every clerical worker performed.

DR. GOODWIN (NPL) said that the author stated there was very little difference between this clerical work and linear programming. It appeared to him (Dr. Goodwin) that the requirements were different. In clerical work the primary need was for high input and output speed while in linear programming the need was for high computing speed.

MR. THOMPSON replied that he had no experience of very large linear systems, but he was thinking more of the sort of work done by him during the war. Sugar and fats were rationed then, but his company had to keep up that volume of supply of cakes etc. which yielded the maximum profit.

MR. NEWMAN (NPL) said that in this country there were about two million clerical workers, the cost of whom totalled £1 000 000 000 per annum. At the Ministry of National Insurance, punched card machines were now used. If the computing could be done even one hundred times faster, the saving that resulted would still only be a fraction. The problems were largely those arising from document handling, not from computing only.

It was interesting to consider the things that were done more efficiently by men than by machines. Man was efficient in judgment and in recognition. These were parallel operations. The things that machines did more efficiently were all serial operations. This might provide a criterion for deciding between men and machines for particular jobs.

MR. MICHEL (NPL) thought the author had glossed over the question of sorting. Permanently stored data could be pre-sorted, but there were sorting problems inside the machine which were not solved by pre-sorting.

MR. THOMPSON said that the need for a large high-speed store must be avoided. This limited the practicable amount of sorting. Wage rates were an example of where pre-sorting was possible. Such sorting as was necessary inside the machine was done there, but as much as possible was sorted outside. There was a definite need for a sorting, selecting, and counting machine.

PROF. HARTREE (Cambridge University) said that transliteration of data should be avoided. This was of great importance in other fields, not only in clerical work. In large-scale experimental work the data should be produced in the form required by the machine which was going to process that data.

MR. WRIGHT (NPL) remarked on the really enormous stores required for large-scale clerical work. He understood that a page of a newspaper held some 10^9 digits. Should we in the future have inputs from enormous rolls of very wide tape?

MR. D. O. CLAYDEN (NPL) asked whether a multiplier was really necessary and whether in clerical routines each instruction was not used once only (for example, for each person in a wages routine). He believed that there were about 2000 instructions in the Lyons' routine. Could these be put on magnetic tapes etc., and not in the high-speed store.

MR. THOMPSON replied that it was worth while having a multipler even though it was not used for a very large part of the time. They were, in fact, contemplating having an automatic divider which would be used even less than the multiplier was used; it would save space now used in storing the dividing routines.

It was perfectly true that each instruction was used in general only once per man. However, instructions must be held in the high-speed store. It was not practicable to read them in afresh for each man, as the data had to be read in as well, and speed was essential. Furthermore, some instructions were modified by the machine just as in scientific computing programmes.

MR. HARTLEY (University College, London) suggested that it was wasteful to use a machine for programme testing and that it would be better for programmers to check each other's work. He proposed that the checker should be given only the coding and no other information, but this scheme was not supported by other speakers.

MR. PETHERICK (Royal Aircraft Establishment) recommended the Analyx Synchroprinter as an output mechanism. It had a cylinder which rotated at 30 r.p.s. and 40 identical disks, each carrying the 10 numerals. There was a timing mechanism which hammered the paper against the required numeral in the disk. The machine, which produced surprisingly good copy, was simpler than the Potter flying typewriter, was inexpensive compared with the Bull machines or any of the hirable apparatus, and occupied two feet of rack space on a nineteen-inch rack.

MR. BIRD (British Tabulating Machine Co.) asked whether letters as well as numbers were required in the output from mathematical computations. If they were, he thought that the letters might be inserted later by hand onto the copy that carried the automatically printed numerical data.

14. INPUT AND OUTPUT

by

D. W. Davies

National Physical Laboratory

Introduction

This paper discusses the principles governing the design of an input-output system. The subject is a difficult one, because the input-output requirements of a computer depend on what problems it is called on to solve and on the staff that will programme and operate it. At this stage of development, much of what is said must be a matter of opinion. We shall naturally have in mind a computer such as the ACE Pilot Model (though not necessarily a machine so restricted in storage capacity) which is working on miscellaneous problems arising in scientific research. The main object of the paper is to provide a starting point for discussion. It was felt that a mere description of existing systems would not have served this purpose.

It is essential that input-output systems should not be so slow as to extend considerably the time taken for computation. An equally important requirement is that they should be convenient for the machine operators. A system poorly designed in this respect may waste a lot of time because of operators' errors. A further important requirement is the convenience of the programmer. In a computer used for a variety of problems, considerable time is spent in getting programmes right.

General Remarks

The simplest form of input to a computer is a number of keys like those of a typewriter, on which both data and instructions can be 'typed'. The simplest form of output is an electrically operated typewriter. These are adequate where the computer is slow and where the programme is not changed often and is retained in the store when the machine is switched off. Many present day computers have begun life with a teleprinter as input and output, because it was the simplest thing to try first.

To make full use of a fast computer it is necessary to have an intermediate store or record of very low cost between the keyboard on which the data and instructions are originally 'typed' and the fast store of the computer. Examples of this store are teletape, punched cards and magnetic tape. Many computers use, (or used in the past) teletape for this purpose because it was readily available. Similarly, Hollerith cards were pressed into service because the system already existed.

The intermediate record arrangement has the following advantages:

1. The information put on the record can be verified and corrected before it goes into the computer.

2. The rate at which numbers and instructions can be 'typed' on a keyboard is about 5 per second. The input to the computer from the intermediate record can be faster. This is an advantage because several input keyboards can be employed with one computer, and computing is not held up while the keyboards are in use. Also, the same intermediate record can be used as computer input on as many occasions as required.

It is also common to have an intermediate store between the computer and the final printing machine for the following reasons:

1. The printer may not be fast enough to match the output from the computer. In this case, many printers may be employed with one computer.
2. The layout of the printed pages can be decided after the computation is finished, and changes in layout can be made before producing the final copy.
3. Failure of the printer will not waste computer time.

If intermediate records of the same kind are used in both input and output, there is the possibility of reading in records which have been written by the computer. This can be used:

1. to perform checks on records intended for printing. This does not check the process of reading the record in the printer and printing it.
2. to enable programmes to be processed and re-recorded so that the altered programme is available for future use.
3. to carry out further computations on data where the need for further computation could only be seen by examining the results.

We shall call the keyboard operated device which writes the intermediate record the 'inscriber' and the device which reads the record and prints the final results the 'outscriber'. These terms are used in the SEAC system.

From one point of view, the inscriber and outscriber are the true input and output and the intermediate record is part of the machine store.

In some systems the input and output organs of the computer are the same thing. The UNIVAC and RAYDAC, for example, use a magnetic tape intermediate record. The tape reader and recorder are combined in a single device which can run the tape in either direction under the control of the computer and can read what it previously recorded. The input/output units are therefore stores of the computer in the fullest sense.

The primary purpose of the intermediate record is to convey data from the inscriber to the computer and from the computer to the outscriber. This data will be in decimal notation.

When the intermediate record is used as an auxiliary store of the computer, and the computer uses binary code internally, it is desirable to be able to record and read full length binary numbers as well as decimal characters.

Before discussing the parts of an input/output system in more detail it should be mentioned that there is a school of thought which opposes the use of an intermediate record for output. Although the record itself may be checked, there is a danger of errors in the outscriber. If the output printers give signals which indicate what they have printed, a computer connected directly to the printers can receive these signals and make a final check. If an intermediate record is used, a similar degree of checking can be obtained only by making a further record from the signals produced by the printers and checking this record. Even so, the possibility of an undetected error is slightly higher than in the case of the directly connected printer.

Rates of Input and Output

Computers can be divided into two classes: data processing machines and general purpose machines. The data processing computer does a small amount of computation on a large amount of data. Business computers are an example of data processing machines.

It is obvious that a data processing machine will require a high rate of input and output. The input and output systems must be designed to fit in with an organization to collect and distribute data. For maximum efficiency the input and output systems will be special to each application, and even the central computer may have special features. This paper will not concern itself with these machines.

We shall consider general purpose computers which are designed to cope with scientific and engineering calculations. In these machines also, the form of input and output required will depend on the kind of calculation usually performed. In attempting to give figures for desirable rates of input and output we shall be thinking of the sort of problems for which the ACE Pilot Model is at present used.

The maximum average rate at which decimal results are produced in the ACE Pilot Model is estimated at 100 decimal digits per second. One can imagine simple calculations that produce more output than this. However, a certain amount of inefficiency can be tolerated in these types of calculation, provided they do not form more than a small part of the work the computer is called on to do.

An output speed of 200 decimal digits per second would be desirable. In a calculation producing results at 100 digits per second only a third of the time will then be spent on output. A similar input speed would probably be adequate, but if the computer reads in instructions in decimal form a rather higher speed might be desired.

Speeds up to 1000 decimal digits per second would be useful on the ACE Pilot Model. Above this speed the conversion of decimal to binary and the corresponding reconversion, would require more instructions. This would be true in all programmes (unless the speed were variable) and it would outweigh the advantage of higher speed, which only occurs rarely.

The useful binary input and output speed depends on the extent to which the intermediate record is used as an auxiliary store. It is desirable to be able to empty or fill the complete internal store in a reasonable time. To fill a store of 30 000 bits in 30 seconds requires 1000 bits per second. At present the binary output rate of the ACE Pilot Model is 640 bits/second and its input rate 1280 bits per second. This makes a useful auxiliary store for some problems. When considering a magnetic tape store, however, the time taken to run the tape to the required item may determine its useful speed.

The Intermediate Record

The systems most worthy of consideration for intermediate records are magnetic tape, punched tape, printed tape and punched cards. Photographic methods are rejected because of the inconvenience of the developing process and the cost of material.

Magnetic tape differs from the other media in that its records are erasible. For input/output purposes this is not important, and a non-erasible record is quite satisfactory if the medium is cheap enough.

Magnetic tape allows higher packing densities than the other media, with consequent saving of space in the storage of intermediate results and data on which further computation may be needed in the future. The density which can be achieved is 2000 bits per square inch. Punched paper tape (teletape) gives 100/square inch and printed tape may give rather more. Punched cards (Hollerith) give 50/square inch but use a thicker material. Decimally punched cards in normal Hollerith code have only 17 bits per square inch, effectively.

The packing density on punched cards is low, but owing to the convenience of stacking records (of varying length) in drawers which waste little space, the storage space may not be so much greater.

At present, magnetic tape systems using the highest obtainable densities of packing are troubled by flaws in the tape. These, when all precautions have been taken in manufacture, can be kept down to a level of one per 200 feet at present. It should be possible to eliminate these flaws in time. Systems which use a wide track are not troubled by these flaws, because the flaws are of the order of 0.01 inch in diameter. A typical case is the SEAC auxiliary store which has only one track on a ¼" wide tape and writes 100 digits per inch along the tape. The packing density is effectively 400/square inch in this case.

To consider the ACE Pilot Model, which uses Hollerith cards, it is doubtful whether an average output from the machine over a day would ever exceed 10 decimal digits per second. Assuming that this output would remain one day before being printed and disposed of, about 15 drawers would be needed to hold the cards. In fact, our storage capacity of 240 drawers is mainly used for holding intermediate results because, owing to the smallness of the computer's high speed store, cards are used as an auxiliary. Programmes occupy a negligible space. The fact that cards associated with a job have to be kept until we are certain that the job is finished accounts for the size of the store.

Nevertheless, this store of cards is not excessive in size. Since cards are the least closely packed of all the media mentioned, it appears that packing density is not a limiting factor. The limit of packing which it is worth attempting in an intermediate record for output is the packing density of the printed page, which is about 200 bits per square inch.

The input and output speed of these various media depends on the number of channels. Magnetic tapes have been used with from one to eight channels. (In the eight channel UNIVAC tape, one is a 'clock' or 'sprocket' tract and one is a check digit, leaving 6 effective channels). Among punched paper tapes the most common are 5 or 7 channel teletapes. A wide paper tape was used in ASCC with 24 channels, but it was not required to move fast. Punched cards give 80 channels when fed broadside, which is the normal way.

The simple way to use any of these recording media is to record or read one line at a time. The number of recording elements in the line gives the number of channels. However, it is possible to deal with several lines at a time. A group of lines recorded or read simultaneously may be called a 'frame'. The number of channels is now the number of recording elements in a frame. At the other extreme, one may read or record only one digit of a line at a time, giving a single channel system. This is done in some teleprinter mechanisms. Magnetic tape will probably be limited to the line at a time arrangement.

The limiting rate of recording per channel with magnetic tape is, at present, about 10 000 bits per second, which is far more than we judge to be needed, even if only one channel is employed. This is in sharp contrast with the limiting rate of punching mechanisms which is about 25 bits per second per channel. The limitation is due to the energy needed to accelerate punch knives and the necessity of stopping the paper completely during the punch stroke.

It appears, therefore, that if punched tape and card systems are to reach the speed we consider desirable, it can only be by parallel operation of about 40 channels. The reproducer made by Machines Bull of Paris punches an entire Hollerith card of 80 x 12 holes at once, but since the setting up of the mechanical store is done serially, a corresponding increase of speed is not obtained.

The rate of recording per channel by printing on a paper tape is not limited in the same way as punching. In the Eastman Kodak printer, a stylus pushes a carbon paper ribbon against a moving strip of paper, the travel required by the stylus being very small. It is claimed that 2000 bits per second per channel can be achieved.

There is no significant limitation in the input speeds of any of the media we are considering. In the case of magnetic tape, the input can be as fast as the output. In the case of printed and punched media, reading photoelectrically at 10^6 bits per second might be possible, but the transport of the tape may be limited to about 100 inches per second, giving about 1000 bits per second per channel.

We therefore see that all the systems considered, magnetic tape, punched cards and tape and printed tape are acceptable as regards packing density and input/output speed provided that the punching systems have about 40 channels. Which is used will depend on other factors. We now consider in more detail the magnetic tape, punched card and printed tape systems.

Magnetic Tape

It is usual to code the decimal digits into four binary digits and to add a fifth digit to give an odd count. This check digit allows detection of any single error when the tape is read. The spare 6 values of the four bit code are used in input for instructions such as 'end of number', 'cancel the last number' and 'end of record'. In output they may be used for instructions such as 'space', 'carriage return' and 'tabulate' and for signs.

If an error occurs whereby a digit is lost during reading, there is a rather high probability that a neighbouring digit will also be lost. It would seem better, therefore, to use a code in which 3 out of 6 digits are ones and three are zeros. To have an undetected error, digits must be both gained and lost, which is unlikely. The code gives 20 values, allowing rather more scope in instructions to the printer. It will be called the '3 out of 6 code'.

There are many variations in the way in which the magnetic recording is done. Consider first a single channel system which the magnetic recording is using the 3 out of 6 code. We will suppose that ones are represented by longitudinal magnetization in one sense and zeros by longitudinal magnetisation in the opposite sense.

If a space is left between successive magnetized regions, the output for a one would be positive followed by a negative pulse and for a zero a negative followed by a positive pulse. In the recorder, the magnetizing pulse can be very short. Provided the flux has been built up in the head, a region given by the gap field will be magnetized. This is the system used in SEAC. The advantage is that no clock track is needed, the disadvantage is that records must be erased before being written over. The need for erasure may be inconvenient in an auxiliary store, but does not matter so much in input/output use.

If no space is left between successive magnetized regions and the head remains energized in the 0 direction between records we have the so called 'non return to zero' system in which erasure is unnecessary. However, if a succession of similarly magnetized areas occurs, no pulse is produced in reading. Hence, in general, a clock track is needed to define position. This clock track may have 01010 etc. recorded on it, which produces alternating pulses.

A method of avoiding the clock track is to begin each character with a recognisable symbol such as 01 and synchronise the following 6 pulses from this. It requires accurate control of the tape speed to succeed. This is analogous to the way teleprinter signals are sent. A magnetic tape system on the teleprinter principle has been developed by Standard Telephone Laboratories for LEO. A second method of avoiding the clock track is to use a tape with sprocket holes, but this limits speed and packing density.

Now consider a 6-channel system using non-return to zero recording. By recording a 1 by a *change* of magnetization and a 0 by no *change* of magnetization we produce a pulse from the reading head where 1 was recorded and no pulse where 0 was recorded. The read pulses are of alternating polarity and must be rectified. In the decimal codes we have mentioned, at least one of the digits in each line is a 1. Since a pulse appears in the reading head at each line, no clock track is needed. This system is used in the IBM type 701 computer.

This suggests an alternative to recording in one channel with a clock track. It is to record a 1 by a change of magnetization on one track and 0 by a change of magnetization of the other track. A check can be made while reading to ensure that there is an output pulse sufficiently often, but never pulses on both channels. In this way the immunity to tape flaws will be increased.

The 6-track system has the advantage over the 2-track system in that the tape speed is much less for a given rate of decimal digits. Present day systems using 7 or 8 tracks and employing a $\frac{1}{2}$-in. tape have been susceptible to flaws which, though detected, are time-wasting. Since systems with one track on a ½-in. tape have not been troubled by flaws, there seems to be some case for a wider tape for multi-channel systems, say 1-in.

Interaction of Magnetic Tape with the Computer

Under this heading we consider the synchronization of the tape recording with the operation of the computer. We are concerned now not so much with the average rate at which decimal and binary numbers are called in and put out by the computer as with the size of group called for or sent out at a time and the arrangements to make the input/output operations convenient for the programmer. This is the most difficult aspect of the subject.

The limiting factor here is the time taken to accelerate the tape from rest and to stop it, and the space wasted on the tape in the process. By accelerating only a small section of tape and making the tape reels follow slowly, the time from rest to full speed and vice-versa has been reduced to a few milli-seconds. The time required seems to increase with speed of the tape. Systems with the tape held on a capstan seem to be capable of 5 ms. at about 100 inches per second. The Ferranti tape drive can reach a 20 inches per second running speed in 2 ms and stop in the same time. There are systems in which the capstan rotates continuously and the tape is clamped to it by an electromagnetically operated pinch roller or a vacuum controlled by an electromagnetic valve.

Stopping and starting in a few milliseconds is instantaneous as regards the control of the reels. Hence the problem of the reels increases with increasing speed. The UNIVAC reel servos are very elaborate not only because of the high tape speed of 100 inches per second but also because the reels are heavy. The best technique at this speed is that used on the IBM 701 machine. The buffer loops of tape are held in tanks with a vacuum applied at the bottom to tension the tape. Vacuum operated switches detect the loop length and control magnetic clutches which move the reels.

At speeds of 10 inches per second the tape can be made to control the reels with a simple mechanical brake arrangement.

A low tape speed gives advantages in the simplicity of the mechanism, lower wear of tape and heads and, generally speaking, greater ease of threading the tape. This points to the use of several recording channels rather than one, and as close packing along the tape as possible. However, packing density along the tape may be limited in the inscriber and outscriber.

The simple arrangement for interaction between the computer and the output magnetic tape is as follows. When the computer wishes to send out some data it sends the first decimal digit to a certain one-digit decimal store and then proceeds to compute the next digit. The tape starts to move when a digit is in this store, and when the full speed is reached the digit is recorded. If the computer attempts to send the next digit to the store before the first has been recorded, it is held up automatically. After the first digit has been recorded, successive digits are recorded at regular intervals as required by the spacing on the tape. The computer must be capable of filling the digit store between the time of recording one digit and the next. When the computer comes to the end of the set of digits it wishes to record it sends no more to the digit store. At the time that a new digit is due to be recorded, if nothing has been sent to the digit store the tape is stopped automatically. By a digit is meant either a decimal digit or a signal to the outscriber such as "tabulate". The set of digits sent out at a time can constitute several numbers.

The input arrangements for the simple system are similar. The computer attempts to read a digit from the one-digit store, but is held up automatically while the tape is started and the digit read. As soon as the first digit is in the store the computer is allowed to have it and proceed with the conversion to binary. The tape continues to move and the next digit is read. The computer must be ready for this digit early enough to dispose of it before the following digit has to be read. The process can be stopped at the end of a record either by a special character or simply by the absence of any recording. A gap will have been left by the inscriber to allow for the stopping and starting of the tape. The programme of the computer must be diverted at this time. This can best be done by a forced discrimination. For example, instead of obeying the input instruction which normally takes a digit from the one-digit store, the computer will ignore the instruction and go to the instruction following the next one it would have obeyed.

This simple system employs a buffer store of one decimal digit. This store holds the digit only for the access time of the store of the computer. The system will work if the decimal digits are read and recorded at a time interval greater than the access time of the computer.

Access time here may mean the access time of instructions rather than of the store to which the decimal digit is being sent. In the ACE Pilot Model for example, the digit may be sent to a short tank with access time of 32 µs. However, when the computer is held up automatically for the line of the tape to arrive, owing to the 1024 µs recirculation time of the long tanks in which instructions are kept, it must be held up a multiple of 1024 µs to keep in step. The simple system we have described will read and record rates just less than one decimal digit per 1024 µs.

The read and record rates should be chosen rather slower than the rate at which digits can be absorbed by the conversion programme or produced by the reconversion programme. In non-optimally programmed machines this will always be several access times per digit. In the ACE it can conveniently be one access time (1024µs) per digit.

A system for binary input and output should, for the convenience of the programmer, carry out automatically the splitting of the binary number into the several lines of record and the converse process. In a parallel computer there may be a shifting register which can be used for this purpose, if computation does not continue during input/output. In a serial delay-line computer the shifting may be done by inverting delays into the recirculatio path of a delay line. For example, to assemble a number in a minor cycle length tank, six digits at a time, an alternative recirculation path with six unit delay, is provided. The normal recirculation continues until the 6 digits are ready for insertion. Then, for one minor cycle, the alternative path is used, and the six digits inserted simultaneously at the various delays. The previous contents of the tank are delayed 6 to make room for the six new digits on the beginning of the number. This arrangement will be called a "precessing tank" a term used in SEAC.

Using such a precessing tank, or a shift register, the simple input/output system described above will deal with a binary number in the time taken to deal with the corresponding number of lines of decimal digits. Thus a 36-digit binary number could be written in rather more than 6 access times. In general, the lines of six digits could be produced more rapidly than this, but the refilling of the precessing tank or shift register with a new number would take one access time. High speed could be used by leaving a gap at the end of each group of six lines or by having a second register or tank to store the new number ready for filling the shift register or precessing tank.

Since decimal digits will not be produced more rapidly than one per access time, the simple system is adequate, except possibly for one feature. Often the programmer will wish to send out numbers one at a time, as they are produced in the course of computation. A number of six digits will occupy only a short length of tape, and this will be followed by wasted tape due to stopping and starting. If the stopping time is comparable with the time to record six digits, the waste of space on the tape may be considerable.

In the case of the ACE, which conveniently produces a decimal digit per ms this wastage is not severe because stopping times can be a few ms. In any case, we consider that one digit per 5 ms would be an adequate speed for decimal input and output.

The problem is more acute when a tape is used for other purposes such as dumping the contents of the whole store or where it is used as an intermediate store by the computer. In these cases higher speeds will be needed, and to avoid excessive waste of space on the tape it is necessary to record the data in large blocks. This is sometimes done by making the input/output instruction always refer to a block of data of fixed size, such a delay line full. A special store is used to hold this block while it is being transferred to (or from) the tape. An additional buffer store is needed to hold the amount of data which can be transferred to (or from) the tape in one access time.

As an example, suppose that numbers are to be written on the tape from the ACE at 5000 lines per second and 6 digits per line, or 30 000 digits per second. Then one short tank full (32 digits) must be stored to provide the output for one access time (1024 microseconds). This short tank can precess and produce 6 digits in 32 μs, and will be required to do so every 200 μs, when a line is recorded. After 6 precessions a number will have been recorded and the precessing tank must be refilled in less than 200 μs if recording is to be continuous. Therefore a second short tank is needed which will take the next number from the store when it is available and present it to the precessing tank when required, that is at 1200 μs intervals.

Although, for tape economy in fast outputs, it is convenient to have a transfer of a large block, if what the programmer wants to transfer is considerably less than this amount the economy is lost. The programmer can arrange to store the output data until a reasonable block is ready. Alternatively, the output order may be more flexible, and may indicate the block of consecutive addresses to be recorded, allowing the block to be of variable length. This arrangement, with interlocks which enable computing to continue during input/output without danger, is to be used in DYSEAC, the successor to SEAC.

In scientific computers, as distinct from data processing machines, all the numbers read will have many arithmetic operations performed on them, and all those recorded for output will have had many operations performed on them. Therefore input and output of binary numbers need not be at a greater speed than one number per access time, even for optimum programmed computers. There are therefore the following alternatives: (All remarks apply to input as well as output).

1. Input and output in blocks of fixed length. Where much smaller groups of output are needed the programmer will have to assemble the numbers and send them out when the block is nearly full.

2. Input and output word by word. In this case the output must be programmed, whatever the block length, but numbers will often be transferred from the places in which they were formed, without any assembly process.

3. Input and output in blocks of variable length. This gives the best of both methods, provided successive output of single numbers is allowed without stopping the tape.

If the word by word method is provided, blocks of variable length can be dealt with by using a subroutine. This probably takes longer and is less convenient to the programmer than alternative 3 above, but the extra equipment is saved.

Punched Cards

Though the ultimate possibilities of punched cards in speed and packing density will not compare with magnetic tape, they have advantages, and the great merit that reliable machinery for an entire input/output system already exists, as a result of development over many years.

Punched cards are convenient to store, and records of all sizes can be put in drawers, with separating cards, so that any record can be extracted easily. The records can be read visually and altered by either punching a hole or filling a hole in. Holes can be filled by pushing back a punching, and the result will stand up to many readings, though it is advisable to reproduce it when convenient. This facility of easy alteration is particularly valuable in getting programmes right. In spite of correct punching of the programme cards from the written sheets, alterations are often necessary while the programme is being tested on the computer. These, if they are small alterations, can be made on the spot without holding up the computer. The alternative where magnetic tape is the input medium is to make the modifications in the copy of the programme in the computer store and later record the corrected programme. To do this easily, elaborate monitoring and modifying facilities are needed, and these may be valuable in themselves. There is a danger that the modified programme will be lost, due to a further programme error, before it has been recorded.

We propose to describe the punched card input/output arrangements of the ACE Pilot Model, and their possible further development.

The card is fed broadside, and 32 columns of the card are used. These correspond to the 32 digits in the binary word of the computer. In reading or punching, the 12 rows move, in sequence, past the brushes or punches. The contents of each row form a binary number of 32 digits.

The reader feeds cards at 200 per minute, and the successive numbers are read at intervals of 19 ms. The punch feeds cards at 100 per minute, and the numbers are punched at intervals of 43 ms. The rate of input is therefore 1280 bits per second, and of output, 640 bits per second.

Only complete card feeds are allowed. The computer can start either the reader or punch, which proceeds to feed one card, then stops, unless it is called on by the computer to feed a further card, in which case the card movement will continue. The computer must set up an instruction in time for each row, to read or punch the number. The obeying of the instruction is held up until a signal from the 8-card machine indicates that a row of the card is in position. Twelve of these signals occur during the passage of one card, and the last of them is distinguished by an additional signal, for the convenience of the programmer.

The punching of records whose length is not known until the last item has been punched, and the punching of records involving only a few rows of the card can be a little awkward. In the DEUCE, the successor to the ACE Pilot Model, there will be changes which, it is hoped, will help the programmer. Separate orders will start and stop the card machines. Once started, they will continue until told to stop. When a stop order is given, the signals signifying that the card row is in position will be suppressed, though the card will continue to move until a complete card cycle is ended. Even if a start order is given before the end of the cycle, the signals will not be resumed until the start of the next card.

The binary numbers are sent to and from the card machines in parallel form on 32 channels. From the card reader, the 32 channels go to a dynamiciser which produces the serial signal continuously while the brushes are reading a row. A reliable signal is obtained for 6 ms. Signals go to the punch from a 32-stage register on which the serial signal is staticised.

Both reading and punching of binary numbers could be doubled in speed by using two fields of 32 columns of the card. This is done in the IBM 701 computer and in ORDVAC.

The treatment of decimal numbers on the ACE Pilot Model is to punch them in the ordinary hollerith code of one hole per column. This has the advantage that a tabulator can be employed as outscriber, but is wasteful of both space and speed. The card is read and punched row by row for decimal numbers as for binary. There is therefore an unusual conversion and reconversion process whereby all the zeros are dealt with first, then all the ones, then the twos and so on. The programmes which do this are more complicated than digit by digit conversion and reconversion, by nevertheless can operate during the passage of the card.

If one were to employ a more compact decimal code, one would wish to simplify the conversion/reconversion processes so that they could still be done during the passage of the card. If a redundant code were employed for the decimal digits, there would have to be a built in code converter and checker, for example because programming this process is not possible in the time available. A shift register might well be provided for assembly of the signal which is to be punched and for splitting up the signal which has been read. The same shift register could take the place of the input dynamiciser and output staticiser.

The difficulty in employing such a compact code is in the inscriber and outscriber. There is no broadside step by step card feed available, and the endwise feeds are too slow to use on the computer.

A possible solution, which would enable 160 decimal digits to be punched on a card in a 6-bit code, 2 digits per column, is as follows. A special 1024 digit tank is provided. As decimal digits are produced by the programme they are stored in the tank, which precesses 6 bits each time. Decimal digits can be sent to the tank once per major cycle. When the tank is full enough, its contents are punched out. The first row of the card will need to receive every twelfth bit in the storage tank. These can be picked out and sent to an 80-bit shift register for punching. The tank is then precessed one bit and the next row punched, and so on.

The scheme requires a lot of equipment, but it is possible that some of it would have other uses, for example in input/output of binary numbers. There need not be objections from the programmer's point of view to the punching of large blocks of numbers since, except for the emptying of the delay line store at the end of a run, the programmer need never consider the way that his output is divided among the cards.

On the ACE Pilot Model, punched card equipment has given very good service with very little development. It can be argued, however, that a new input/output system developed for the purpose will be better, in the long run.

Printed Paper Tape

The possibility of printed paper tape as an intermediate record was demonstrated by the Eastman Kodak Company at the New York conference on input and output (December 1952).

The object of the Eastman Kodak machine was to print address labels on the matrix system, that is representing alphabetic characters by a dot-pattern. We are not interested in this aspect, but in the means by which it was achieved.

The characters were printed in a 5 x 7 array by passing under seven styli a white paper tape, with a carbon paper tape above it. The styli pressed the carbon paper onto the white paper. The carbon paper ran more slowly than the white paper, thereby saving material and achieving a rubbing action. It was claimed that the styli would record at 2000 bits per second. Certainly there is not the same speed limitation as in punching. A system with six channels, 20 bits per inch, 10 inches per second would achieve the speed we consider necessary for a scientific computer.

The inerasibility is no disadvantage since the material is cheap. Records of any length can be torn off as required, the record can be read visually and probably a simple system could be devised for correction. Furthermore a very small hand printer could be produced.

As the paper leaves the printing styli it passes under a radiation heater which comes on only while the tape is moving. This heater fixes the print so that it will not smudge.

No attempts have been made yet to read this tape photoelectrically but the dots are regular in shape, size and position and quite black, and there should be no insuperable difficulty.

The Inscriber

Essentially, an inscriber consists of a keyboard attached to a machine for recording, but there are usually additional refinements.

If the record cannot be read easily by the operator, it is usual for the inscriber to have a typewriter which prints an account of all that has been put on the keyboard, including corrections. This is not used for checking the record, but for the operator to be certain about errors he thinks he has made and to check that he has corrected them properly.

For correction of errors on the spot it is convenient for the operator to back space a certain way. This is difficult to achieve on high density magnetic tape. For this reason, the UNIVAC, which normally has 128 bits per inch per channel, reduces this to 20 per inch in the inscriber. In the SEAC and RAYDAC systems the problem is bypassed by recording first on teletape and transferring from this to magnetic wire or tape at constant recording speed.

The latest UNIVAC inscriber overcomes the back spacing problem ingeniously and achieves a density of 50 per inch. The tape is linked mechanically to the carriage of the typewriter making the printed record. Thus back spacing for correction is possible within the current line on the typewriter. The line is of fixed length, having 10 numbers of 12 alphanumeric characters each.

Having written the record it is usual to verify it. This can be done by making a duplicate record and comparing. A better way is to have a verifier machine which reads the first record and compares it with what the operator types on the keyboard the second time. This is done with Hollerith cards. The operator is given a second chance if there is a disagreement.

In the case of continuous tape records, a new tape will be needed except in the rate case of no errors. It is usual here for the verifying machine to make a duplicate record, as long as there is agreement. When there is disagreement the operator must choose which version is correct and put this on the new tape.

In the inscriber of the magnetic tape system developed for LEO, the correct second version of the tape is recorded on the tape that has been used for the first version. This tape passes from the reader to the recorder, with a slack loop to allow insertion of extra characters that were omitted in the first version, should this be necessary.

The Outscriber

We have said that an output of 100 decimal digits per second is the peak rate to be expected in a computer like the ACE. However it will be rarely that the peak rate is reached Owing to the fact that the ACE Pilot Model is used on a number of problems throughout the day, spending an hour or two on each one, and that each programmer has access to it for a half hour, if he wishes, for testing programmes, the output averages itself out over a day.

The worst that could happen is that the results of a problem involving a great deal of output were wanted so urgently that the computer was held on that problem all day. One must consider, however, how urgently the results could actually be needed. No scientific establishment is likely to be able to make full use of 40 000 numbers of ten digits in a day, yet this output could be produced at 10 digits per second in twelve hours.

We do not consider, therefore, that more than 10 digits per second can be expected for any length of time. This does not mean that the outscriber should be limited to this capacity, but it need not be much faster.

Fast printers exist, giving rates of about 500 characters per second. They will have applications in business machines, but at present most of them do not seem well adapted for this.

The highest rates can be obtained by non-mechanical printers. Characters can be formed on the screen of a cathode ray tube by various methods. To avoid subsequent photographic operations it has been proposed that the results should be printed by xerography, in which a dust is electrostatically attracted to a screen in the parts which have illuminated, then the dust is transferred to paper and fixed by heat. This has not been achieved to the author's knowledge in a fully operational machine.

The General Electric Company of the USA is developing 'magnetography' in which characters are first written on a magnetic material in magnetization patterns and then this is used to pick up dust which is transferred to paper. This method, if it can produce high quality, would be a useful printing process, since any number of copies can be made. The possible correction of the original magnetic record is an attractive feature. High recording speeds are possible by using the field between a fixed flat pole and a rotating disk with raised type on it.

Mechanical printers attain high speeds by parallel operation and by hitting 'on the fly'. An example of an 'on the fly' printer is the well known Potter machine. A second version of this 'Flying Typewriter' prints at 10 lines per second and saves some of its multitude of valves by using a magnetic drum. Similar printers are made by Wheaton Engineering Laboratories, Shepard Laboratories, and Anderson-Nichols in the USA. A fast printer specifically for business applications is being developed by Eckert-Mauchly for UNIVAC. These fast mechanical printers are usually expensive and do not produce high quality results.

Hollerith printers attain a considerable capacity by parallel operation of about 80 characters on a line, a method well suited to the Hollerith card. Speeds vary from 75 lines per minute to 150 per minute in a wheel-printer made by Bull Machines. In common with the other fast parallel printers, considerable storage is needed when printing from a record on a tape.

Electric typewriters are limited to about 10 characters per second. If a typewriter were made for numerical work alone a higher speed might be possible. The cost of these typewriters is low in proportion to their speed, making it possible to have a battery of them instead of one fast printer. This would have the advantage that a mechanical fault would not immobilise the whole printing capacity.

An advantage of typewriters is their adaptability to printing from a tape record, without any store. Also, the quality of printing is higher than that of any other method. For a scientific computer producing results at the rate we have estimated, two or three typewriter outscribers should be adequate.

The operation of a typewriter from high-density magnetic tape recording without an intermediate store presents certain problems. It is necessary to read the tape line by line, occasionally waiting to allow for 'tabulate' and 'carriage return' operations. In the UNIVAC system the problem is overcome by recording all data intended for the outscriber at 20 lines per inch. At this density it is possible to scan one line at a time and stop between lines. In the SEAC system, the record on magnetic wire is transferred to teletape. The characters are written at regular intervals on the wire and,with the wire running at constant speed, can be punched at a regular rate. The signals from the wire are, however, only 200 microvolts because of the low speed.

In the RAYDAC system there is an intermediate store for an 8-digit decimal number. The recording on the tape is in blocks of 32 such decimal numbers with a space between blocks. In printing, each block is scanned 32 times, and each time a different number is transferred to the store. The typewriter operates from the number in the store. Scanning a block of data takes only 1/10 second.

Apart from the decimal digits, space, decimal point and signs, the computer should be able to give the printer signals to determine the layout of the page, such as tabulate, line feed and carriage return. By means of the programme it is possible to deal with such questions as suppressing insignificant zeros and signs, inserting decimal points, extra line spaces, and headings.

In scientific computers, however, programming occupies a great deal of time and any possible way of simplifying it must be considered. Therefore the printer should have an extra range of facilities that will enable all questions of page layout to be settled after the computation is programmed. This will enable changes of layout to be made after a finished page has been seen without interfering with the course of the calculations. For this purpose, only such signals as 'end of number' and 'end of block' would appear on the tape from the computer. The layout would be decided by a control tape, for example.

Since the typewriter conveniently writes from left to right and down the page, the data should be in the correct order on the tape as it comes from the computer. This restricts the control over the layout. If the record is printed on a paper tape, editing on the tape is possible before printing, or after a first copy has been made.

Accuracy of Input/Output Equipment

Errors in going from the inscriber to the computer and from the computer to the outscriber can be reduced by a redundant code in the intermediate record. The characters are checked when reading by the verifier, computer and outscriber, or at each stage if two intermediate records are employed. Error correcting codes are not usually employed, because they are too redundant.

These error detecting systems do not avoid the waste of time if errors do occur, so a very high reliability of the equipment is still required.

The accuracy needed for some work can only be achieved by applying checking procedures such as differencing from the printed results.

If the input of data to the computer must be checked (which is not always the case) this can be done by putting it out onto the intermediate record again and checking this on the verifier.

The output of data can be checked if an intermediate record can be produced which corresponds to what has been printed. This should be possible if the printer is a typewriter, since each type bar can be connected mechanically to a contact, and the contacts can operate a recorder. The only errors which might pass through are an 8 being turned into a 3 or a + into a - by incomplete printing.

The fullest checking can be done only by a proof reading machine which will read the final characters and recognise them. A simpler device of a similar nature is one which would compare two pages for identity. This would enable a new copy made after an error had been corrected to be compared with the old one to see that no new errors had been introduced.

Discussion

MR. BIRD (British Tabulating Machine Co.) summarized a paper of his, on input and output equipment, which was to appear shortly in the Journal of Electronic Engineering. He said that MR. DAVIES's paper had provided a most searching review of the various methods of input and output to computing machines. He (Mr. Bird) would like to consider the business field in particular, where input and output are of prime importance.

From a study of the type of problems which may be tackled by computers in the field of business it appeared that:

1. A large number of similar problems requiring the same type of calculation, but on different data, must be tackled consecutively. (For example: the production of a pay roll).

2. The length of calculation per problem would be short compared with the scientific field, though the amount of input data and results to be printed might be quite large. The computer would therefore have to spend a much higher proportion of its time accepting and transmitting numbers.

3. Numbers would not only be in decimal, but in other scales; for instance, pounds, shillings and pence frequently have to be dealt with.

To be suitable for adoption by the business world a machine must be economical, reliable, and easy to maintain without the services of a large highly trained staff. Simplicity, and a small number of parts would therefore seem to be desirable.

Mr. Bird continued. Assuming that a punched card input and output would be used with a binary computer it was necessary to examine methods for conversion from any scale of notation (decimal; pounds, shillings and pence; tons, hundredweights etc.), into the binary scale, and its reconversion at the output.

Conversion Methods

A digital computer, if logically complete, could of course convert the input data from any scale of notation to pure binary and reconvert the binary answer to a scale suitable for printing or punching out. This process, whilst needing no extra equipment might be too time-consuming, as the steps in the programme are many.

In business calculations where the number of numerical operations in the main programme would be few, since the problems were usually simple, the numerical operations in each of the conversions (in and out) might exceed in length the main calculation. These numerical operations for conversion would be used many times during the calculation, once for every input number and once for every answer. The position then would be that the computer spent only a small portion of its time actually producing answers from data and most of its time changing scales. It would therefore be worthwhile to consider the construction of special *apparatus* for input and output conversion.

Mr. Bird said he would try to describe a method of conversion to binary, of the significance of the holes in a conventional punched card of the Hollerith type, whilst the card is passing through the reading apparatus, and a method of reconversion of binary to decimal or to any other scale for printing or punching the output.

Considering the Input first

The 80 columns punched in the card were read in parallel by 80 contacts, the 9's being read first, then the 8's, 7's etc. There was a binary equivalent of a "1" hole for every column of the card. These equivalents were stored in serial form, for example, round one track of a magnetic drum. Suppose there were 75 163 in decimal punched on the card. Then if 7 times the binary equivalent of 10000, 5 times the binary equivalent of 1000, 1 times the binary equivalent of 100 and sc on could be accumulated the binary equivalent of the decimal punching would be available at the end of the card passage. Mr. Bird said that an electronic circuit described in his paper selected the required equivalents from the store and allowed them to pass the correct number of times into the computer's adder.

Several numbers could be converted simultaneously, either in a single long accumulator or in several accumulators in parallel.

To change the input notation from, say, tons, hundredweights, etc., to pounds, shillings and pence, it would only be necessary to change to a fresh batch of constants on another drum track. Thus, input conversion had been dealt with.

The computer, having prepared its answers in the binary scale had to convert them to a form suitable to operate a punched card printer or punch. These pieces of apparatus would accept an input in which each digit is coded in binary. For instance to print 11, 4 relays could be set up with 1011.

Conversion from binary to these groups of binary coded digits could be performed by the inverse of the input conversion. For example to convert 1001010100 which is binary for 596 to 0101/1001/0110 we could sum the binary coded decimal equivalents of the 1's in the binary number to be converted. This needed a special type of binary adder of which more details would be given by MR. TOWNSEND in Paper 16.

Mr. Bird continued. The binary number to be converted to "binary coded decimal" was fed to a storage register and the "binary coded decimal" equivalent of a 1 in each possible digit position of the binary number was recorded serially round a drum track.

These were allowed to add into the accumulation through the special adder, if there was a 1 in the corresponding binary digit of the number to be converted. So that in a serial machine, after a number of word times equal to the number of digits in a word, the binary number had been converted to a string of binary coded decimal groups each group representing one of the digits it was desired to print or punch out.

These could then be read out in parallel to the printer or punch. For pounds, shillings and pence conversion, the equivalents were altered accordingly.

It would seem that the changes could be rung on the basic idea on both input and output to obtain more speed at the cost of more equipment. This could be done either by accumulating several numbers in parallel or by translating several digits within the number simultaneously.

It was Mr. Bird's opinion that if the provision of extra equipment necessary for input output conversion in the computer was to be avoided then more equipment had to be used elsewhere in one of the following ways.

1. By building a faster, larger storage binary machine and doing the conversion by programme.

2. By building a machine that worked in a scale other than binary.

3. By converting the input and output from and to binary in ancillary equipment.

The choice of which method was adopted would depend on the purpose for which the computer was required.

The discussion was continued, by:

MR. POLLARD (Ferranti Ltd.) who described the high-speed printer now connected to the Ferranti machine. It consisted of a wheel printer made by the Bull Co. of France. There were 92 print wheels and the printing speed was 150 lines per minute.

The machine had an extra c.r.t. store, which could be treated just as one of the main storage units except during printing, when the situation was that all connexions to the computer were broken and each short line of 10 binary digits represented two decimal digits. Each such line was scanned at each "point" of the printing cycle. This scanning was not synchronized in any way, the speed rendering this unnecessary; in fact there was sufficient time between "points" of the printing cycle to repeat the scan.

DR. WILKES (Cambridge University Mathematical Laboratory) said the design of commercial machines started with the form of the data, and the machines were built around this, whereas mathematical machines were first designed and then the designer proceeded to his input and output. In his opinion, what was really required was not a fast input and output but a large auxiliary store.

Auxiliary stores could be loaded when the machine was working on some other problem. Input and output were connected to the auxiliary store and the latter connected to the machine. The auxiliary store should be in binary, the input and output in decimal.

DR. TOCHER (Imperial College, London) described a method of decimal-binary conversion which was an alternative to that described by MR. BIRD, suitable for a slow machine such as a relay machine.

DR. BENNETT (Ferranti Ltd.) said DR. TOCHER had emphasized a point he would like to bring out in a different form. The use of cards would be easier if they were fed endwise instead of broadside on.

MR. BIRD (Brit. Tab. Co.) said there were end-wise card feeds but they were slow; however they were cheap. He also remarked that DR. TOCHER's scheme would have to be changed for sterling calculations.

MR. DAVIES (NPL) said a fast endwise feed had been made for UNIVAC. It had the drawback that there was too much data on each card, and the card could not be stopped in the middle. UNIVAC also had a good auxiliary store but the input and output were rather awkward. He was disappointed that the discussion had concentrated on punched cards and that other forms of input and output had been neglected.

15. ECHELON STORAGE SYSTEMS

by

D. O. Clayden

National Physical Laboratory

In all forms of storage for high-speed computing machines there are, to some extent, problems of access time. The store of a computer must be designed so that the time lost due to access time is kept small, but at the same time the amount of equipment must be kept to a minimum. This can be achieved by providing the machine with storage units having a variety of access times. These stores can be arranged in "tiers" corresponding to their access times in much the same way that an army in the field can be organized into echelons.

This paper is mostly about delay-line type machines, but applies also to machines with cathode ray tube storage.

Access Time

A general purpose computer has to deal with numbers and instructions (words) whose frequency of use is likely to vary over a very large range. Some words, for instance, may be required several thousand times a second whereas others may be required only once every few minutes.

The words used most frequently should become available as soon as they are required. Any delay in their availability is likely to add substantially to the time taken to do the programme. On the other hand the computer can take a relatively long time to obey the instructions which occur least frequently without unduly increasing the time taken by the programme as a whole. The words used least frequently can therefore be kept in storage units having a long access time without much loss of machine speed. For instance, if a block of instructions or data is required only once per second, it could be kept, not in the high-speed store, but on a magnetic drum with an access time of 10 ms (*i.e.* revolving at 6000 r.p.m.), and transferred to the high-speed store when required. The time wasted due to access time would in this case be 1% of the total time. Similarly, if a block of numbers is kept in, say, a long delay line, these numbers can be transferred to a number of short delay lines to speed up a calculation. The numbers used most frequently should, of course, be located in storage units with a short access time, but the fixed instructions using these numbers can be kept in locations having a long maximum access time provided that the instructions are duplicated where necessary and become available when required by means of optimum coding.

Since, in general, storage units having a longer access time have a larger capacity and a lower cost per digit, it is likely that equipment and cost can be saved, or storage capacity can be increased by providing a machine with a variety of storage units with access times covering a considerable range.

High-Speed Storage

The main high-speed stores of present computers are made up of units having a capacity of usually not more than 40 words. This size has usually been due to technical limitations, and in the case of delay line machines, may not otherwise be the most economic. Of course there is no point in making a delay line so long that nearly all the instructions in it have to be duplicated in order to conserve time, but it is probably worth making it so long that a quarter or even a half of its instructions are duplicated, provided of course, that its cost is not significantly increased. In fact, from an economic point of view, it is probably best to have instruction delay lines of several different lengths, although this may be impracticable in machines using absolute timing.

Storage locations associated with arithmetic units must be of the appropriate capacity, usually holding one or two words. On the ACE Pilot Model it is necessary to provide at least one single-word store to enable numbers to be added to the accumulator several times in a long transfer, and also for transfering information to and from long delay lines. (This does not apply in the case of cathode ray tube storage where any location can be selected at any time). One of the one-word stores associated with an arithmetic unit can be used for this purpose.

However, apart from these special cases, numbers usually need not be located in one-word stores. On the Pilot Model the access time can often without harm as long as four-word times; this is the time taken when, for instance, a number is transferred to the accumulator, operated upon, and transferred back again. Numbers can often have an access time considerably greater than this without increasing the time taken by the loop of instructions using them. It appears, therefore, that the storage units used for numbers can, with advantage, be provided in a variety of sizes.

Lower-Speed Storage

A similar argument applies to magnetic drum storage. If a machine is provided with two drums, it is probably advantageous to make one drum with a larger capacity per track and a longer access time than the other.

Several different types of input mechanism are in common use. Punched cards and punched tape are most convenient for data which have to be used in mechanical equipment such as a printer. Time can, however, be saved by using magnetic tape, which is faster, for output data that will be used in the machine again.

Application

These principles have been applied to some extent in the design of the ACE Pilot Model and to a greater extent in its successor the DEUCE, which is provided with mercury delay lines that store 1, 2, 4 or 32 words, and an auxiliary magnetic drum store of (initially) 1024 words. The input and output is on Hollerith cards.

At the present state of development it is likely that mercury delay lines can be made satisfactorily with a capacity of 2000 and possibly 4000 digits. (The delay lines used on the Pilot Model hold 1024 digits).

A machine of the Pilot Model type with its store designed on the foregoing principles, and with delay lines holding 64 words for its main high-speed store, might have a distribution of storage capacity as shown in *fig.* 1. This store uses the same number of delay lines as the DEUCE but has 50% more high-speed capacity. Such a machine would probably be as fast as the DEUCE and might on some problems be faster due to the increased high-speed storage capacity and the redistribution of short delay lines.

Discussion

DR. KILBURN (Manchester University) said that the principles advanced in this paper could usefully be applied only to small-scale machines. Essentially, they gave a method of minimizing the disadvantages of a slow-access store, whereas a perfectly satisfactory fast-access store was already available for use in any large machine. A small computer using a magnetic drum was being developed at Manchester University: for this and similar projects, the speaker agreed with the application both of optimum programming and of the principles of echelon storage.

Any machine with slow-access storage should have automatic floating-point arithmetic, so that adequate return could be obtained in one operation for the time taken in getting the operand out of the store. He was surprised that automatic floating operations were not included in the ACE Pilot Model.

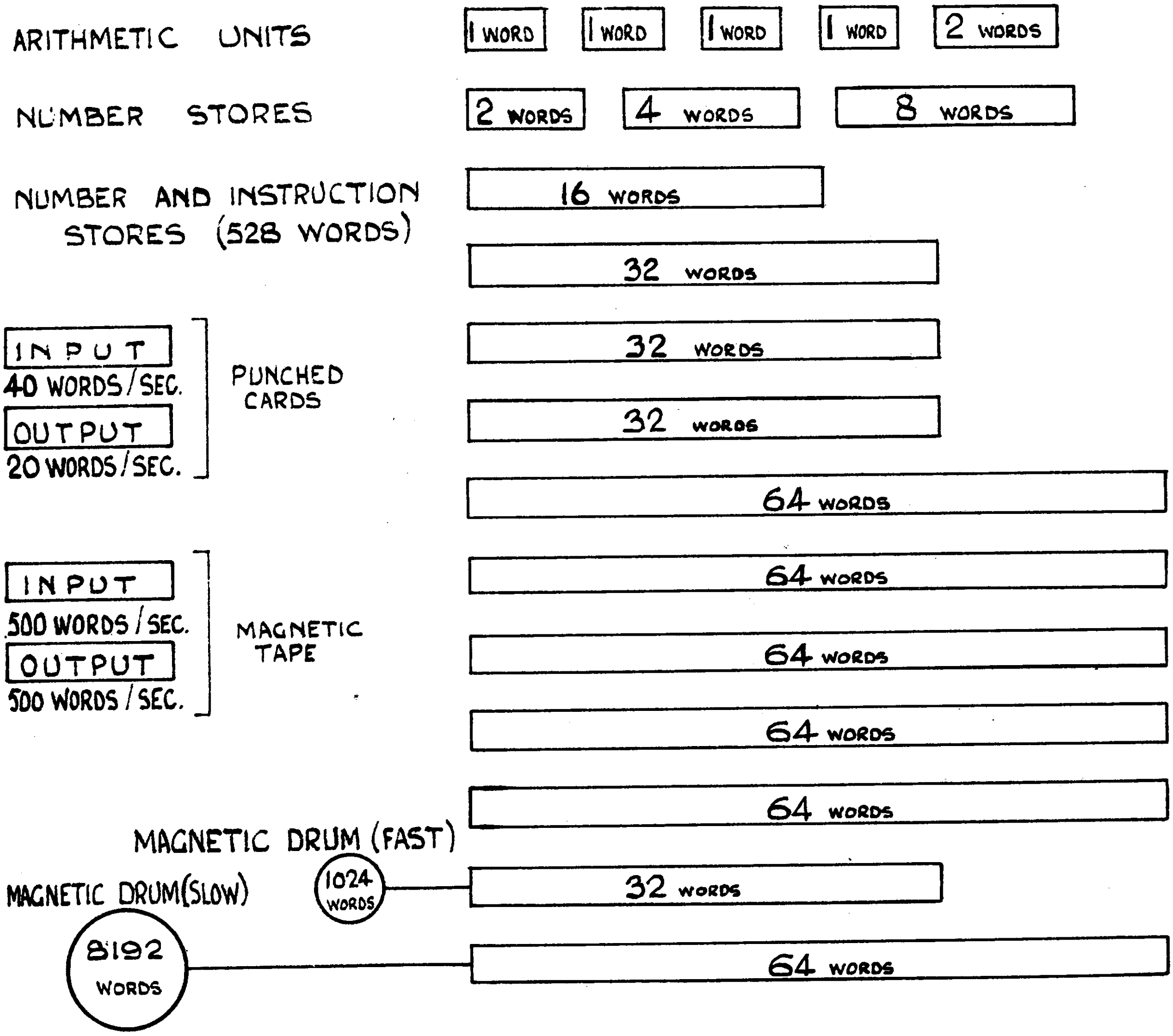


FIG. 1

A DELAY LINE STORE FOR AN OPTIMUM CODED COMPUTER

MR. BLUNDELL (Ministry of Supply, RRDE) said that the use of a 16-word store to hold a subroutine was unnecessary, since it was possible to insert all programmes in the order of the flow diagram, and to use a conversion routine to prepare the optimum coding. Library subroutines would then be available only in flow-diagram order.

DR. FRIEDMAN (Cambridge University Mathematical Laboratory) said that it was possible to have echelon storage without optimum programming. A magneto-strictive delay line storage system, for instance, could be arranged to give out information faster than it could be taken in by inserting several reading heads on the same line. Such a store could be made to eject the next few instructions to be obeyed, into a buffer store, where they would be available as soon as required.

MR. WILKINSON, (NPL) replying to Dr. Kilburn's last remark said that the ACE Pilot Model, in common with other early machines, was made as simple as possible. In particular, it had no automatic floating point facilities. However, such facilities were really less necessary in a machine with optimum coding and several temporary stores, where floating-point operations by a subroutine would still be comparable in time with the access of individual numbers.

MR. MacMULLAN, (De Havilland Propellers, Ltd.) said that the use of a programme to convert from a flow diagram into optimum coding form would considerably increase the difficulty of checking programmes; in the present state of reliability of machines, it would also increase the risk of confusion between machine faults and programme errors.

MR. NEWMAN, (NPL) said, in reply to Dr. Kilburn, that there might be circumstances in which a delay line store was preferable to a c.r.t. store, in spite of the disadvantage in access time. In such circumstances, tricks to minimize this disadvantage were well worth while, in that they could reduce the time of operation by a factor of eight in the most favourable cases with little extra equipment. Optimum programming was not difficult.

MR. BIRCHALL, (RAE) said that a conversion routine for optimum coding would need to be very complicated, since the most frequently repeated loops of instructions would have to be coded first.

In reply to MR. COHEN, (RAE), MR. CLAYDEN said that in his proposed scheme the major cycle would consist of 64 minor cycles.

16. Serial Digital Adders for a Variable Radix of Notation

by

R. Townsend

The British Tabulating Machine Co. Ltd.

Most of the computers so far constructed have operated in binary notation to take advantage of the engineering simplicity which this confers. A computer for business application must be able to handle a large quantity of numerical data upon which comparatively simple calculations are to be performed, and since much of this data must be in decimal or sterling, it is necessary to convert it to and from binary notation. This can be quite satisfactorily performed by programme in the case of scientific applications where there is only a small input and output of data in comparison with the time of calculation. But for business purposes it is found that a disproportionately long time is taken in conversion, except on the fastest machines.

One method of overcoming this is to build special conversion units, and methods of automatic conversion have been discussed by Mr. Bird in a paper referred to in the discussion of paper 14. One method of conversion from binary to binary coded decimal or sterling makes use of an adder which can add directly in serial binary coded decimal or sterling. An alternative solution is to perform the whole calculation in coded groups, which has been done for decimals in some American machines. In Britain it is important to work in sterling or other non-uniform scales of notation. A completely universal adder would work in any scale of notation either uniform or non-uniform with binary as a special case. This paper is concerned with adders of this form in which numbers are represented in coded groups in which each group can have the same or different radices. These have been named programmed adders, as the scale of notation can be changed as desired during the course of a programme.

When a number is represented in programmed notation, each group of 4 digits can represent any number up to 16. If two such numbers are added it must be arranged that the correct sum is produced in each group with the appropriate carry of 1 into the next group if required. This can be done by the addition of 16-r, which is called the "filler", and r is the radix of notation, if the sum in any group exceeds the radix r. The generalized rules are:

$A + B < r$	no "filler" added
$r \leq A + B < 16$	"filler" added
$16 \leq A + B < 32$	"filler" added

Some examples of numbers represented in programmed notation are shown innTable I, and of their addition in Table II.

Four methods have been devised and are described, by which numbers in programmed notation can be added.

Programmed Adder Type 1

A binary adder has been designed which can accept at its input three serial binary numbers, the stored number from the accumulator A, the incident number from the main store B and the "filler" F, which is held in an associated temporary store. It provides two outputs, the binary sums A + B, and A + B + F, that is the unfilled number and the filled number respectively.

The two binary sums pass simultaneously into two 4-digit delays. At the end of each group of 4 binary digits, a test is made to see if a carry has occurred in either the sum A + B or A + B + F. If there has been a carry in either case a flip-flop is set to the filled condition which opens a gate and allows the filled number, delayed 4 digit periods, to enter the main accumulator.

If on the other hand, there is no carry from A + B or from A + B + F at the end of the group, the flip-flop is set to the unfilled condition and a gate is opened allowing the unfilled number, delayed 4 digits, to enter the accumulator register. This process is repeated for each group of 4 digits in the number.

Programmed Adder Type II

In this method, the binary sum of the number in the accumulator and the incident number from the memory is added in a two input binary adder, and passes into a 4-digit delay. At the same time the sum from this adder A + B is compared with the radix of notation minus one (r-1) for that particular group in a comparison circuit. It so happens that r-1 may be easily obtained by negating the "filler".

IIf $A + B > r-1$ *or* if there has been a carry in the first binary adder at the fourth digit in the group, then a gate is opened allowing the next filler group which has been delayed 4 digits to be added to the sum A + B delayed in a second two input binary adder, forming the sum A + B + F which passes into the main accumulator register.

If $A + B \leq r-1$ *and* there has been no carry at the fourth digit in the first binary adder, then the gate is closed and no filler is added in the second binary adder, allowing the sum A + B delayed to pass unchanged into the main accumulator register.

Programmed Adder Type III

In this type of adder, the number in the accumulator and the incident number from memory are added as before in a two input binary adder, the sum passing into a 4-digit delay unit. At the same time the sum A + B is added to the filler in a rudimentary binary adder in which only the parts necessary to form the carry are retained, which is called a carry generator.

The first binary adder and the carry generator are tested at every fourth digit for carries. If there has been a carry at the fourth digit in either the first binary adder *or* the carry generator a gate is opened allowing the filler delayed four digits to be added to the delayed sum $A + B$ in a second binary adder, the sum $A + B + F$ passing into the main accumulator register.

If there has been no carry in both the first binary adder *and* the carry generator at the fourth digit, the filler is not added in the second binary adder and the sum $A + B$ is allowed to pass into the main accumulator register unchanged.

Programmed Adder Type IV

In this last method, a 3-input binary adder is again used, the binary sum of the stored digit in the accumulator, the incident digit from the memory and the filler digit all being added together, and the binary sum $A + B + F$ passing into a 4-digit delay. The 3-input binary adder has two carry stores and these are tested at the end of each group as usual. If there is a carry from either the sum $A + B$ *or* the sum $A + B + F$ at the end of a group, a flip-flop is set to the filled state which closes a gate and allows the delayed sum $A + B + F$ to pass into the main accumulator register unchanged.

When, on the contrary, there is no carry in either of the carry stores in the 3-input adder at the fourth digit, a flip-flop is set to the filled state, which opens a gate and allows the complement of the filler to be added to the sum $A + B + F$ in the second binary adder, having the effect of subtracting the filler from $A + B + F$ and allowing $A + B$ to enter the main accumulator register. The unwanted carry then produced at the end of each group of 4 digits in the second binary adder is suppressed.

Programmed Excess Notation Adder

It is difficult to subtract in ordinary programmed notation and this has led to the suggestion of Programmed Excess Notation in which subtraction is easier.

The use of excess three notation for decimals is well known. In this code, 3 is added to each decimal digit and the result is expressed in binary form. The advantage of this is that the complement to 9 needed in subtraction can be obtained by simply transposing 0's and 1's. This scheme can be extended to any even non-uniform radix of notation, so that for scale of 12 the excess will be 2 or in the general case $16 - r/2$.

The rules for addition can be summarized as:

1. If the sum of the two groups produces a carry at the end of the group then $16-r/2$ must be added.
2. If the sum of the two groups does not produce a carry at the end of a group then $16-{}^{r}/_{2}$ must be subtracted.

For addition in this notation, the number in the accumulator and the incident number from the memory are first added in a two-input binary adder and the sum passes into a 4-digit delay. At the end of each group of 4 digits the carry store of the first adder is tested. If it contains a 1, a gate is opened which allows the excess to be added to the delayed sum in a second binary adder. If on the other hand there is no carry, a gate is opened which allows the complement of the excess to be added in the second binary adder, thus having the effect of subtracting the excess.

The complete paper (from which this summary has been taken) will appear in "Electronic Engineering" and the author's thanks are due to the Editor for permission to publish this paper at the Symposium.

TABLE I

Binary Coded Decimal

Denomination	10 thousands	thousands	hundreds	tens	units
Numerical example	9	7	6	4	5
Binary Code Decimal equivalent	1001	0111	0110	0100	0101
Radix	10	10	10	10	10
Filler	6	6	6	6	6
Filler in binary	0110	0110	0110	0110	0110

Programmed Sterling

Denomination	ten pounds	pounds	ten shillings	shillings	pence
Numerical example	2	9	1	7	11
Programmed Notation equivalent	0010	1001	0001	0111	1011
Radix	10	10	2	10	12
Filler	6	6	14	6	4
Filler in binary	0110	0110	1110	0110	0100

Programmed Hours, Minutes and Seconds

Denomination	hours	ten minutes	minutes	ten seconds	seconds
Numerical example	2	4	3	3	7
Programmed Notation equivalent	0010	0100	0011	0011	0111
Radix	10	6	10	6	10
Filler	6	10	6	10	6
Filler in binary	0110	1010	0110	1010	0110

TABLE II

Addition in Binary Coded Decimal

Decimal	Ten thousands	thousands	hundreds	tens	units	Remarks
39828	0011	1001	1000	0010	1000	
49137	0100	1001	0001	0011	0111	
	1000	0010	1001	0101	1111	Binary sum.
		0110			0110	Underlined groups filled
88965	1000	1000	1001	0110	0101	

TABLE II (contd.)

Addition in Programmed Sterling

£ s d	10 £	£	10/-	s	d	Remarks
38 13 8	0011	1000	0001	0011	1000	
29 14 7	0010	1001	0001	0100	0111	
	0110	$\underline{0001}$	$\underline{0010}$	0111	$\underline{1111}$	Binary sum.
		0110	1110		0100	Underlined groups filled.
68 83	0110	1000	0000	1000	0011	

Discussion

DR. VAN WIJNGAARDEN (Mathematisch Centrum, Amsterdam) proposed a system of notation in which numbers were represented by their residues when divided by each of a fixed set of primes. This would define uniquely all numbers less than the product of the chosen primes. Addition, subtraction and multiplication could be carried out by applying these operations separately to the individual residues. No carry would be required from one residue to the next, and the system would be particularly convenient for a parallel machine.

In reply to MR. NEWMAN (NPL), DR. VAN WIJNGAARDEN said that the problem of reconversion was not too difficult, but that he had yet to discover a satisfactory process for division in this notation.

MR. TOWNSEND replying to MR. HARTLEY (University College, London) said that normal multiplication was difficult, using his adder, because of the problem of shift. It might be possible, however, to use the well-known "halving and doubling" method of multiplication. Halving could be achieved by shifting the number down by one binary digit and subtracting half the filler where appropriate. If the filler were made zero, the adder would perform normally in the binary scale.

MR. WILKINSON (NPL) said that routines for reading decimally punched cards into the ACE Pilot Model incorporated a check. This ensured that just one hole was punched in each column of the field. The check was originally designed to detect faulty operation of the reader, but in practice had served more to check that the cards had been correctly punched.

THE UTILIZATION OF COMPUTING MACHINES-I

Chairman: Dr. E. T. Goodwin

17. Mathematics and Computing

by

A. van Wijngaarden

Mathematisch Centrum, Amsterdam

A peculiar interaction takes place nowadays between mathematics in general and its remarkable offshoot that forms the topic of this conference, viz. highspeed computing. Of course, a good deal of inter-connexion already exists between mathematics and ordinary computing but high-speed computing asks for new mathematical methods and also makes it possible to use already existing methods that were declared obsolete because they are impractical for ordinary calculations. High-speed computing acts therefore on one side as a stimulus to mathematics whereas on the other side the computer may profit from the large reservoir of mathematical knowledge already in existence.

It is not my intention to discuss fully all consequences of the impact of high-speed computing on mathematics or vice versa but I shall restrict myself to some special topics in this field, mostly in connexion with work that has been done at the Mathematical Centre at Amsterdam.

First of all, I should like to emphasize that mathematics is more than analysis and algebra and that in principle other fields of mathematics might be of comparable use to computing. But, of course, the computer who does other work than sheer arithmetic and elementary algebra is more likely to come into contact with analysis than with anything else. Already many interesting points arise in the application of analysis to computing. Perhaps the most important tool that analysis provides the computer with is the calculus of finite differences. At first sight this looks less highbrow than the infinitesimal calculus, dealing with infinitely small differences but in reality it lies much deeper and forms actually a rather advanced topic in the theory of complex functions. The fact that it is so difficult is the reason that not very much is known in comparison to the situation in ordinary calculus. This provides an interesting source of uncertainty about our results.

One of the most prominent applications of the calculus of finite differences to computing is the theory of numerical interpolation, integration, and so on. An interpolated or integrated value of a function is represented as a linear combination of tabular entries, either directly, in which case we speak of Lagrangean type formulas, or in two steps in that first, certain simple linear combinations of the entries are formed, so called differences of various orders, after which process the result is formed as a linear combination of one or two entries and a number of differences of increasing orders. The coefficients by which these differences have to be multiplied tend to zero rather rapidly in general. As long as only a finite number of terms are used both methods are equivalent, apart perhaps from rounding-off errors. The computer can only use, of course, a finite number of terms, and the question arises: "How many terms?" This is a very interesting question with a lot of aspects. First of all, it might not be superfluous perhaps to emphasize the fact that in general an arbitrary accuracy cannot be obtained by taking more terms into account, if at the same time the "step", *i.e.* the increment of the argument of the entries is kept constant. In general, the series diverges, and if it happens to converge it is an extreme coincidence if the sum is the result that one wants to obtain. Therefore, one has to be modest in the number of terms. Under certain rather weak restrictions, the error committed by stopping after n terms in all these types of processes is equal to the product of three factors. The first factor is a rather irrelevant with increasing n, decreasing function of the spot at which one interpolates and so on, the second is the n-th power of the step and the third is the derivative of order n+a (a being some constant) of the manipulated function somewhere in the interval determined by the arguments used in the process. From this observation, one can prove the following general statements. The greater the step the more necessary, dangerous and inefficient is the use of higher differences. The smaller the step, the more superfluous, harmless and efficient is the use of higher differences.

If smaller steps are used in integration, say, obviously a large number of them have to be taken in order to reach the aim. If high-speed computers are used, there are no direct objections to the use of very small intervals. This is a very nice situation. All remainder terms decrease enormously in size. Either one can increase the accuracy considerably in this way or one can abstain from the use of differences of high order. In the last case one gets instead of higher accuracy something else of high value, viz. ease of programming. The tendency towards ease of programming may here and in similar cases easily go so far that one accepts only the crudest possible methods. In order to keep the accuracy constant one has then to decrease the step extremely. Not only does this decrease the efficiency of the high-speed computer but it also may introduce errors of another type, viz. those due to rounding off. Moreover in cases where all entries have to be stored *e.g.* in the solution of elliptic partial differential equations, the required storage space may easily become prohibitive.

Both these remarks point our attention to another branch of mathematics, viz. statistics. The study of the phenomena due to rounding off errors follows lines closely related to those followed in mathematical statistics. It is of vital importance for modern computing and here a direct stimulus to mathematics comes from computing. This study is moreover rather interesting from a mathematical point of view also. Here again, apparently simple problems need already powerful tools of analysis and more than that, one fruitfully introduces geometrical and numbertheoretical concepts. The first results of general character reveal unexpected and peculiar phenomena.

The second difficulty mentioned, viz. that of the storage of many function values in the solution of partial differential equations, is overcome at least in principle in a remarkable way by the introduction of the diffusion analogy, analyzed years ago by Courant, together with the application of the Monte Carlo method. Here the close co-operation with the statistician is evident and the theory of the random walk is rapidly extending, due to the stimulus of computing.

Another interesting point arises in connexion with the Monte Carol method connecting computing with the theory of numbers, viz. the generation of random digits and random numbers. Of course, one can make those by means of special electronic devices, the electronic coins, but it is much more interesting and also more practical in relation to the reproducibility of the gambling process to generate them by computing. The question arises then, how to generate very long sequences of numbers or digits, "very long" meaning, with a very long period of repetition of the same pattern. Moreover the numbers or digits must pass successfully, statistical tests for randomness. A proper method has been indicated by Lehmer, who defines the sequence by the congruence $u_n = a\ u_{n-1}$ (mod N)., $0 < u_n < N$. Each number is completely determined by its predecessor, and as there are only N numbers different modulo N, the sequence is periodic and its period is less than N. If one chooses a arbitrarily with respect to N, the period may be only a small fraction of N but, corresponding to a given N, there exists a maximum period and a can be chosen in several ways so that one obtains that maximum period irrespective of the value u_0. This is a pure numbertheoretical problem. I suggested some time ago the use of recurrent sequences of second or higher order. Indeed, if one defines the sequence *e.g.* by $u_n = u_{n-k+1} - u_{n-k}$, a term is defined by its predecessors, and therefore the first observation teaches that the period is less than N^k. Now a number which we know is less than N^k has a good chance to be considerably larger than a number that is less than N. The formation is moreover extremely simple, the simplest example being the Fibonacci sequence $u_n = u_{n-1} + u_{n-2}$, so that there was a good reason to investigate the matter in considerable detail. Extensive studies have since been made by several research workers at the Mathematical Centre and I think it a good example of real contact between mathematics and computing. The computer here yields the stimulus and some provisory theorems most likely to be true as a first working basis. The pure mathematicians have not only provided the correct proofs but they have done much more; they added to pure mathematics a new chapter that was at the same time of practical value to the computer. Moreover, each new practical application of number theory is of interest as such, as there are still people who think that numbertheory is impractical.

At this moment it is perhaps worth while to ask whether high-speed computing can be of any value to pure mathematics, because number theory is then one of the most likely candidates to profit. The field certainly lies open here. For instance, the search for particular numbers can be extended to a somewhat higher level and can suggest conjectures that may be proved or disproved, but usually remain open. For instance, the search for new Mersenne prime numbers has been a welcome prey for the fast computers and the computations performed on the SWAC have yielded interesting results. It would be of real theoretical interest, for example, to go a step further than could be handled by the SWAC in its present state, and to investigate the number $-1+2^{8191}$, for if this proved to be prime then a large amount of theoretical work should be justified in order to prove or to disprove the conjecture: If $m_2 = 2^{m_1}-1$ is a prime number, then $m_3 = 2^{m_2}-1$ is also prime. The validity of this conjecture would for instance imply a constructive proof of the existence of a prime-generating function.

Another helping hand can of course be lent by tabulating functions. There are people who do not appreciate tablemaking, and therefore I shall particularise somewhat by taking a specific example. Quite recently I developed a rather general transformation that enables us to compute functions from their heavily diverging asymptotic expansions. This method is in itself not directly meant for high-speed computers but it has the peculiarity that for application one needs extensive tables of very peculiar functions (unrelated to the special function that one wishes to compute, of course). The computation of these tables is an enormous task, and here the high-speed computer can render welcome help.

Quite apart from any applications of the computers, their construction alone yields problems in many fields of mathematics. It is well known that the design of circuitry is to a high degree equivalent to problems in formal logic. Boolean algebra or Aiken algebra form nowadays tools of the computer engineer. In general also all questions with respect to binary representation of digits or numbers suggest problems. For instance, the following is a problem connected with error detecting and correcting codes: how many configurations of n binary digits can be constructed that differ from each other in at least d digits? The solution is not known. It may be regarded as a problem in combinatorial algebra, but it is also a geometrical problem in n dimensions: How many solid hyperspheres of diameter $d^{\frac{1}{2}}$ can be attached with their centres to the corners of a hypercube of unit size? In this form it is closely related to a well-known subject in modern geometry, viz. that of the closet packing of spheres.

Closely connected is the theory of switching functions. Many results in Aiken's tables of switching functions are really clarified by regarding them from a geometrical point of view. If one defines a polytope, formed by corners of an n-dimensional hypercube of unit-size oriented along the coordinate axes, to be at its inside if the corners have not a coordinate in common and at its outside if they have, then a reflection on the table of switching functions reveals the fact that for $n \geqslant 4$ there are regular isotopes that fit into the inside as well as in the outside, whereas for $n < 4$ they do not exist.

A third problem arises from a question of Aiken. If the decimal system is to be used for a computer, and k parallel lines of binary digits are put available to represent decimal digits, then k is obviously at least equal to 4. Is it possible to derive a coding scheme, if necessary at the cost of a greater value of k, such that the sum digit and carry digit in the addition or multiplication of two decimal digits can be obtained, say, by simple permutations or more generally by circuits of given simplicity? Duparc has analysed this and similar problems successfully, but his proofs require concepts of grouptheory.

Van der Pol has dealt recently with problems related to the sum of the digits in any scale of the integral part of functions of x. The first results already are of a certain interest to computing and perhaps an interesting field is opened here. In general one might say that mathematics is quite well armed to answer successfully the peculiar questions arising in high-speed computing. Of course, there are certainly fields in which a lot has to be done, *e.g.* in the theory of repetitive processes.

Discussion

DR. FOX (NPL) said that a further important aspect of the effect of machines on computing was due to the fact that the people who had become proficient at desk computing had not, in general, adapted themselves to programming, for which somewhat different skills were needed and consequently a large number of people using machines had not had the benefit of a thorough experience of the difficulties of computing. Skilled computers knew of the many ways in which errors could arise, and hence the need for checking all work must be strongly emphasized. It is not sufficient to verify that the programmes are correct, *i.e.* that they do what the programmer wants them to do. There must be adequate checks to verify that the solution satisfies the original mathematical problem. Also, since the numbers occurring in the course of computations in machines are rarely seen, more thought is necessary about the mathematical analysis of the problem before it is programmed than is required before a desk computation. There is a tendency for this to be neglected because some clients now expect results far too quickly. There is also a tendency to skimp this preliminary mathematical work because programming is novel and hence more interesting.

With regard to Prof. Wijngaarden's paper, Dr. Fox said that he took a practical attitude in the use of finite difference methods. If, in forming the difference table, it is found that the sign of the n^{th} difference oscillates and its absolute magnitude is less than 2^{n-1} then the interpolation polynomial of the n^{th} order could be used for all applications. For interpolation, small intervals, making n small, are more convenient, but for differentiation of a function, larger intervals will give more correct significant figures. For electronic machines smaller intervals may mean more serious build up of rounding-off errors and often a larger store.

MR. OLVER (NPL) agreed with the last remarks of the previous speaker and considered the other error arising in the use of finite difference formulae, viz. the truncation error. For an interpolation formula $f(a + ph) = A_0 + A_1 \delta + A_2 \delta^2 + \dots$ this error is represented by a remainder term of the form $A_n h^n f^n (\xi)$. Now $h^n f^n (\xi)$ is, in practice, of the order of the n^{th} difference and therefore the remainder term is of the same order of magnitude as the first neglected term.

DR. TURING (Manchester University) preferred a different expression for the remainder term

$$f(z) - P(z) = \int_C \frac{f(t)(z-z_1)(z-z_2)\dots}{(t-z)(t-z_1)(t-z_2)\dots} \, dt$$

where $P(z)$ is the interpolation polynomial coinciding with the values of the function at z_1, z_2, ... and the contour integral includes z, z_1, z_2... but not any other singularities of the integrand. This formula is useful for evaluation of the remainder and showed that special consideration must be given to interpolation near singular points of the functions.

He agreed with Dr. Fox that extra care is needed in the analysis of problems because the computer is not able to see the numbers arising in the machine in the course of computation. On the other hand, the complete analysis of the methods, necessary in programming, enabled an upper limit to the error to be more easily worked out.

DR. BULLARD (NPL) considered that, so far, automatic computing machines had for the most part been rather trivially used on problems for which the solution was easily obtained to a limited accuracy, but for which somebody had been willing to pay for finding a more accurate solution. Now the important thing for physicists was intuition, and in many fields they were lacking this. This was especially true where non-linear equations were concerned. The detailed consideration of the behaviour of certain such equations, using computing machines, would be a way in which a fund of physical intuition could be built up, so that physicists could get the feel of such equations and, scorning the detailed mathematics, could say it was obvious by intuition!

PROF. HARTREE (Cambridge University) quoted Jeffrey's remark that in order that a series should be useful for calculation it was neither necessary nor sufficient that it should converge. As an example of the care needed in inferring the numerical order of magnitude of a quantity from the order of magnitude of the corresponding expression, he referred to a remainder term of the type $Ah^n f^n(\xi)$. If Richardson's process is applied to find an estimate of the value of this, there are cases where the result obtained would be grossly out since $f^n(\xi)$ for the different values of h used might differ by several powers of 10.

18. Linear Algebra on the Pilot ACE

by

J. H. Wilkinson

National Physical Laboratory

Introduction

The basic problems arising in the field of linear algebra are from a mathematical standpoint comparatively simple, but in practice they often prove intractable because large two-dimensional arrays of numbers are involved and consequently the number of arithmetic operations required for their solution is almost prohibitive on desk machines. It is to be expected then, that linear algebra should be a suitable field of operation for high-speed automatic computers, but it appears at first sight, that, to be satisfactory, such a computer should possess a large high-speed store, or at least a large intermediate store. Although the Pilot ACE is particularly deficient in both these requirements it is probably true that in the last year it has been more successful in dealing with problems in this field than in any other. Its sucess has been due to the fact that the auxiliary Hollerith equipment used on the machine has proved very reliable, and the method of optimum coding has made it practicable to use Hollerith cards as an intermediate store for binary numbers, speed being achieved by carrying out nearly all computation between rows while reading from or punching on them. In this paper I shall discuss the use of the machine for:

1. Solution of linear simultaneous algebraic equations.
2. Matrix multiplication.
3. The extraction of the latent roots and vectors of matrices.

Solution of Linear Equations

The problem of solving large sets of linear equations by direct methods has received considerable attention in the last few years. This work indicates that for desk machines the most satisfactory method is that in which the matrix A of the set of equations

$$Ax = b \tag{1}$$

is expressed as the product of a lower triangular matrix L and an upper triangular matrix U. Thus the solution of (1) is reduced to the solution of

$$LUx = b \tag{2}$$

This is expressed in the form

$$Ly = b \tag{3}$$

$$Ux = y \tag{4}$$

The sets of equations (3) and (4) are both triangular in form and can easily be solved by what is usually called a back substitution process.

This method has two advantages over most other methods. It gives a somewhat greater accuracy and it involves recording many fewer numbers on paper. The second of these advantages does not carry so much weight on an automatic computer because when a number arises as the result of an arithmetic operation on such a machine it is, so to speak, already written down. The first would appear to be equally commendable on any equipment, but on automatic machines there is a complicating factor. It is not easy to foresee how large the largest element of L and U is likely to be and it is found in practice that even for sets which are only moderately ill-conditioned quite large numbers do arise. Consequently on a high-speed computer it is necessary either to allow a fair number of spare digital positions to accommodate possible increases in size of numbers (this usually means using double-length arithmetic) or alternatively to resort to floating arithmetic. Either has the effect of decreasing the accuracy obtained from a given storage capacity, and at the same time increasing the time taken for a solution. For this reason the method has not so far been much used on the Pilot ACE. Instead the method of successive elimination of variables or pivotal condensation has been adopted because it is easy to arrange the computation in a form in which the use of floating arithmetic is avoided. The variant of pivotal condensation which is used is that in which the variables are eliminated successively in the natural order, *i.e.* beginning with x_1 and ending with x_{n-1}, but the first row is not chosen automatically at each stage as the pivotal row, but instead that row is chosen which has the largest coefficient in the variable which is about to be eliminated. Thus in the reduction from order n to n-1 the row with the largest coefficient in x_1 is chosen as the pivot and x_1 is eliminated from each of the other (n-1) equations by adding a suitable multiple of the pivotal row to it. Because of the way in which the pivotal row is chosen the multiplying factors cannot exceed unity. After a single reduction the largest coefficient in the reduced matrix cannot be greater in absolute value than twice the largest coefficient in the original matrix. After m reductions the largest element is at most 2^m times as large as the largest original coefficient. It is possible to construct sets in which this factor is achieved but in practice an increase seldom takes place; more frequently the coefficients become progressively smaller, particularly if the equations are ill-conditioned. It was not considered necessary to make an automatic provision for dividing rows of the reduced matrix by a factor in the event of their growing too large, which would have used valuable storage space, but instead it was regarded as adequate to have an arithmetical check to detect if any coefficient exceeded 2^{30} in absolute value on a 32-digit machine, to stop if this occurred, and divide by a factor using a separate programme. If the coefficients are less than 2^{30} before reduction then they must be less than 2^{31} after reduction and therefore the numbers cannot grow out of range in a single reduction. It is interesting that in thousands of applications of this programme, this arithmetic check has never shown a division to be necessary although the original matrix of the coefficients is usually scaled up by a power of ten until the largest elements have 9 decimal digits. Included in the programme for reduction are the usual arithmetic checks based on the sum of the elements of each row and an additional check that the multiplying factors are not greater than unity, *i.e.* that the correct row has been chosen as pivot. In order to prevent trouble from row sums exceeding capacity, sums divided by a power of 2 are used instead of true sums.

The completion of the problem requires the solution of the triangular set formed by the n successive pivotal rows. It is not possible to use a fixed binary point in this part of the process. Instead a reasonable position for the binary point is chosen, based on physical considerations or otherwise, and the back substitution proceeds until the value of a variable obtained exceeds capacity. When this occurs the binary point is adjusted in the values of the variables already found, and the process is continued with the adjusted position of the binary point. *i.e.* a continuous adjustment of the binary point is adopted instead of a true floating binary point because it is faster and needs fewer instructions.

A number of programmes based on the above process have been prepared. The simplest of these is known as the 'fast' programme. This deals with the largest sets which can be solved while storing all the coefficients in the memory. The programme consists of four parts A, B, C and D. The decimal data is placed between A and B and the complete pack is put in the Hollerith reader. Programme A is read in first and immediately calls in the decimal data, stores it in the memory and then calls in Programme B. This carries out the reduction and then calls in Programme C. Programme C effects the back substitution and then calls in Programme D which converts the final answers and punches out decimal results. It also punches a card with -1.0 on it to determine the decimal point in the solution. Thus a floating decimal notation is avoided at the end. The programme will deal for example with a set of order 13 with two right-hand sides or a set of order 10 with 10 right-hand sides. (The arithmetical operations take only a few seconds, less, in fact, than the time taken to read the data). Although one or two important problems have been done which involved the solution of a large number of sets of small order, most of the sets to be solved have contained a large number of variables and for these, other programmes exist which use cards as an intermediate store. The original data is first converted by a programme into binary form and punched on cards, one number on each row of a card, an integral number of cards being allocated to each row of the equation. Since 12 binary numbers may be punched on a card, a set of order 30, for example, will have 3 cards for each row of the equation, that is, 90 cards in all. An arbitrary number of right-hand sides may be dealt with, and associated with each row is a row sum. The main programme, which carries out the reduction, operates as follows: The pivotal row is read in and kept in the memory and then each of the other rows of the equations is read in in turn and the corresponding reduced row punched out. Computation is carried out between rows of cards while punching. The machine determines the next pivotal row during this process and retains it. The reduced set of equations is then transferred from the punch hopper and reduced one stage further. Throughout the course of the programme the machine is reading or punching continuously. The time taken is comparable with that which could be achieved if it were possible to store all numbers in the high-speed store on a machine not using optimum coding. The latest programme of this type deals with equations up to order 192.

Back substitution is carried out by a separate programme in which the pivotal rows are read in one by one starting with the last. For sets of order 32 and less, back substitution can be done on up to four right-hand sides simultaneously so that the machine is then working at about 80% of the speed it could achieve if all numbers were in the high-speed store.

The greatest weakness of the above method is that no advantage can be taken of the existence of zeros in the coefficients except in the punching and conversion of the original decimal data. Sets in which a very high percentage of the coefficients are zero are of common occurrence. For these an iterative method has been used based on the Gauss-Seidel process. The equations are first converted into binary form using a programme and each non-zero coefficient is punched on one line of a card together with a binary number giving the number of the variable with which it is associated. As a first approximation to the solution an arbitrary vector is fed into the machine (or a better approximation if one is available) and then a complete iteration is carried out by reading the binary matrix through the reader, carrying out the computation between rows. Thus for a set of order 120, say, with 2000 non-zero coefficients, the matrix would be punched on 167 cards (12 per card) and the time taken for a single iteration would be $\frac{167}{200}$ minutes. Unfortunately sets of this type usually involve a large number of variables and there are (in general) no zeros in the solution vector. If therefore the set of equations has a number of right-hand sides it is not possible to store a number of iterating vectors simultaneously. For sets up to order 64, three right-hand sides may be stored at one time and there is time between rows to deal with all three. Sets up to order 200 may be solved, but for the higher orders only one solution at a time can be dealt with. This is a great weakness if the programme is being used to invert a matrix, because n separate solutions are necessary in this case.

For sets in which the non-zero coefficients consist of a band of width three centred on the diagonal, the method described at the beginning of this section has been used. The upper and lower triangular matrices are then also of special form, each of them consisting of a band of coefficients of width two. A programme based on this method using single-length arithmetic is trivial but not very useful, because either very low accuracy must be used or else the programme will frequently fail because of numbers growing out of range. A programme using double-length arithmetic in which the original data may contain 14 decimals or less has been made. This permits the numbers to grow considerably in the course of computation without giving trouble.

Matrix multiplication

Although no great diversity of methods exist for matrix multiplication it is an operation of such frequent occurrence that it is important to have efficient programmes for dealing with it. Since the Pilot ACE has such a small memory, subroutines for matrix multiplication in which all matrices are stored is of little value. Matrices have almost always been of such a magnitude that card storage has been necessary.

The simplest of the programmes for forming the product,

$$AB = C,$$

stores A in the memory, reads in B one column, x, at a time, forms Ax in the store at high speed and punches it out. A row of column sums is kept in A and the machine performs the distributive check. This programme may be used for mxn matrices A where both m and n may take values not greater than 14; matrix B may have any number of columns. For a 14 x 14 by a 14 x 14, the time taken is $1\frac{1}{4}$ seconds for the computation, plus the time of reading and punching the binary cards. No computation is done between rows. For matrices of higher order there are a number of programmes in which the computation is carried out between rows. Typical of these is the following which will do matrix multiplications up to order 52. Three column vectors of B, x_1, x_2 and x_3 are read into the store and then matrix A is read. Ax_1 Ax_2 and Ax_3 are formed simultaneously during reading and then punched out. The distributive check is also carried out. The machine is occupied for about 80% of the reading time which is about as high a percentage as may safely be used. Although reading and punching is taking place continuously the machine is achieving about 70% of the speed which would be possible if all members were in the memory. For a set of order 47 for instance, which needs 4 cards per row and has 48 rows because the row of column sums is included, the time taken for the multiplication is

$$\frac{4 \times 48}{200} \times \left(\text{Integer greater than } \frac{47}{3}\right)$$

since 3 columns are dealt with at a time. This is rather less than 15 minutes. For sets of higher order it is possible to do only 2 columns at a time or one column. The limit for one column work is over 200. For large matrices it would clearly be economic to partition in order to use the three-column programme.

Latent Roots of Matrices

The methods used on the Pilot ACE for finding the latent roots and vectors of a matrix have so far all been based on iterative processes. The simplest of these, for finding the latent root λ and latent vector x such that

$$Ax = \lambda x,$$

is that in which an arbitrary vector x_0 is chosen and from it a sequence of vectors x_n is found, such that

$$y_{n+1} = Ax_n$$

$$x_{n+1} = y_{n+1}/N,$$

where N is a normalizing factor.

Then x_n tends to the latent vector corresponding to the largest latent root λ_1. The speed of convergence is comparable with the speed with which

$$\left(\frac{\lambda_2}{\lambda_1}\right)^r \to 0 \text{ where } \lambda_2 \text{ is the next root.}$$

If λ_1 and λ_2 are nearly equal this rate may be quite slow. In this case, the speed of convergence may be greatly accelerated by the use of Aitken's process for obtaining an improved vector from three equally spaced approximations x_r, x_{r+s}, x_{r+2s}. The *ith* component of the improved vector X is obtained from the ith components of the three approximations by the formula

$$X^1 = \frac{x^1_r \; x^1_{r+2s} - (x^1_{r+s})^2}{x^1_r - 2x^1_{r+s} + x^1_{r+2s}}$$

Where the separation of roots is poor, the speed of convergence may be increased by iterating with $(A - pI)$, where I is the unit matrix and p is constant, instead of with A. This matrix has the same latent vectors and its latent roots are all p less than those of A. The separation of roots thus becomes $(\lambda_1 - p)$ to $(\lambda_2 - p)$ instead of λ_1 to λ_2. By a suitable choice of p it may be possible to effect a considerable improvement. In particular we can deal with the case $\lambda_2 = -\lambda_1$ which otherwise gives rise to a non convergent sequence of vectors.

When a root has been found it may be removed from the matrix. The process of removal used for a symmetric matrix is different from that used for an unsymmetric matrix.

In the symmetric case the vector which has been found is normalised so that the sum of the squares of its components is unity. The matrix A_1 is then formed, defined by

$$A_1 = A - \lambda x x'.$$

This matrix has the same latent roots and vectors as the original matrix except that the root corresponding to λ has become zero. The roots and vectors may be found and removed successively. The reduced matrices are all of order 'n' and when the last root has been removed the final matrix should have zero components everywhere apart from rounding-off errors.

For an unsymmetric matrix the method of removal is as follows. The vector x is first normalized so that is largest component is unity. Suppose this is the *rth* component x_r of x. Then the reduced matrix A' has its (i, j) component a'_{ij} given by

$$a'_{ij} = a_{ij} - a_{rj} \, x_i$$

This is an (n x n) matrix with its rth row consisting of zero elements. Suppose y is a second latent vector of A corresponding to a root μ. Choose α and β so that the vector z defined by

$$z = \alpha x + \beta y \tag{1}$$

has zero for its *rth* component so that

$$\alpha + \beta y_r = 0 \tag{2}$$

then

$$\begin{aligned} A'z &= Az - (a_{rj} \, x_i) \, z \\ &= A(\alpha x + \beta y) - (a_{rj} \, x_i) \quad (\alpha x + \beta y) \\ &= \alpha\lambda x + \beta \mu y - \alpha\lambda x - \mu\beta y_r \, x \\ &= \mu(\beta y + \alpha x) \\ &= \mu z \end{aligned} \tag{3}$$

This means that the vector z is a latent vector of the matrix 'A' corresponding to a latent root μ. The latent roots of A' are thus the remaining roots of A, but the latent vectors are linear combinations of those of the original matrix A and the vector which has been removed. Equation (2) shows that the latent vectors of A' all have zeros for the rth component. We may therefore omit the *rth* row of A' which is zero anyway and the *rth* column since it is to multiply a zero component in z and thus work with a square matrix of order (n-1). The latent roots may be found by a successive application of the above process but the set of vectors which are found will not be the true latent vectors.

These may be found as follows. Suppose the first two vectors found are x and z. x will be a true latent vector but z will contain only (n - 1) components. Extend it to n components by putting a zero as its *rth* component where the *rth* is the largest element of x. Then from (1) the true latent vector y corresponding to z is given by

$$y = px + qz \tag{4}$$

If x_r is the *rth* row of the original matrix then

$$x_r' y = p x_r' x + q x_r' z$$

$$i.e.\ p\mu = p\lambda + q(x_r' z)$$

$$y = (x_r' z)x + (\mu - \lambda)z, \text{ ignoring a multiplying factor.}$$

The true latent vector corresponding to the *ith* vector found by the above process may be found by (i - 1) applications of this simple algorithm

The above processes have been programmed for the Pilot ACE. For the symmetric case there is a programme which does not use cards as an intermediate store which will deal with matrices of orders less than 20. In the iteration programme advantage is taken of symmetry, a triangular matrix only being stored. The time for a iteration on a set of order 19 is 2½ seconds.

For the unsymmetric case the largest matrix which can be dealt with without the use of cards is of order 15. The time for an iteration of order 15 is 2 seconds. For the unsymmetric case there is a programme for sets up to order 64, using cards as an intermediate store. The matrix A is produced in binary on cards and one iteration is performed each time the matrix is read. The time for an iteration of order 60 is 1½ minutes. A programme for the Aitken process has been used in connection with this latter programme and gives an enormous saving in time for matrices with poorly separated roots. The Aitken programme is applicable to vectors of any order.

A second iterative method which has been used is that based on successive matrix squaring. If the sequence A, A^2, A^4, A^8 ...... is produced then the successive matrices ultimately consist of a set of n columns, each of which is parallel to the largest latent vector. The columns tend to this vector as fast as

$$\left(\frac{\lambda_2}{\lambda_1}\right)^{2^r} \rightarrow 0.$$

A high rate of convergence is achieved even for quite poor separation, but since there is n times as much work in a single iteration than in the previous process it will not pay to use this method unless poor separation is expected.

A programme based on this method was made at an early stage of development before the machine had its present storage capacity. This could cope with a 12x12 matrix. An interesting feature of the programme is the method used to avoid true floating arithmetic. A power of 2 is associated with each of the matrices as a multiplying factor, this factor being chosen so that the largest element lies between 2^{28} and 2^{29}. A trial matrix multiplication is performed in which only the relative size of the elements of the products is of interest. The largest one so far formed is kept at each stage of this preliminary multiplication. From it a shift is determined which will make this element lie in the range 2^{28} to 2^{29}. A second multiplication is then performed in which all the elements of the product are recorded each being given the appropriate shift. The appropriate power of 2 to associate with the derived matrix is easily determined. For a symmetric matrix, the largest element in any of the squares will be on the diagonal and therefore the preliminary multiplication need only form the n diagonal coefficients. This method of organizing the programme gives the highest permissible accuracy for a given storage space and is still faster than floating arithmetic, even for the unsymmetric case. The use of the Aitken process in connection with the simpler iterative process probably means that matrix squaring will hardly ever be used.

A third iterative process, for use on symmetric matrices only, is that due to Hestens and Karush. In this method, starting with an arbitrary vector x_0, a sequence of values λ_r and x_r tending to a latent root and vector are formed from the relations

$$\lambda_r = \frac{x_r' \, A \, x_r}{x_r' \, x_r}$$

$$x_{r+1} = x_r + \alpha(Ax_r - \lambda x_r)$$

where α is a suitably chosen constant. The value of λ_r associated with x_r is usually called the Rayleigh value for that x_r. In general λ_r has twice as many figures correct as the vector x_r after a certain stage. If one value of λ only is required and the latent vector is not needed accurately, this is a good method. The speed of convergence of the vector is no better than can be obtained by iterating in the simpler manner with $(A - pI)$, the choice of a suitable value of p being as simple as that of a suitable α. If the vectors are needed accurately the third method thus compares unfavourably with a simple modification of the first method and its programme is more extensive. Fast programmes based on it have been made and also a programme using cards as an intermediate store for sets up to order 32. The programmes are now little used.

Discussion

DR. HARTLEY (University College, London) remarked on some features of the method of solution of linear equations on the Pilot ACE which were useful when the equations arose in statistical work. The reduced equations were punched out and a back substitution could be done on each set thus finding the solution in terms of 2, 3, 4, ... variables. The statistician often required these solutions before he knew how many variables to take and hence did not want the order of the variables to be disturbed.

MR. LIVESEY (Manchester University) said that he had used the Manchester machine for linear algebra and found the large auxiliary store was very convenient since a matrix of order up to 100 could be held in it. Often where the latent root of a matrix was required, this matrix was derived from a small amount of data concerning some vibrating system and it would save a large amount of punching data if the machine could handle the whole problem. In general, the programming of this first part was more work than the programming for the finding of the latent roots, and what was required was a programme sufficiently flexible to set up the matrix in for instance all critical speed problems.

DR. FOX (NPL) said the one disadvantage of the method used up to the present on the Pilot ACE for finding latent roots was that no use could be made of a known good approximation to a latent vector, for which the corresponding latent root was neither the smallest nor the largest, when all that was required was a more accurate value of this particular latent vector. Here Rayleigh's principle provided a useful method for estimating the latent root, but there is no method for improving its accuracy.

MR. WILKINSON said that there was no technique on the ACE Pilot Model for improving an approximation to a middle root and vector analogous to that sometimes used on desk machines. If p were the approximation to the required root, then iteration with $(A-pI)^{-1}$ would give rapid convergence in general, but the labour of finding $(A-pI)^{-1}$ was itself far from negligible. This approach was particularly valuable when the root in question was bounded on each side by an unwanted complex root. He said also that he agreed with remarks of Mr. Livesey and found that the first part of the problem often consisted of matrix multiplications.

MR. DAHLQUIST (Swedish Board for Computing Machinery) mentioned that a method of solution of partial differential equations involved the inversion of matrices of which the only elements that were not zero were those on or near the main diagonal and gave the reference "Karlquist, Tellus, November, 1952".

PROF. HARTREE (Cambridge University) asked about the method used on the Pilot ACE for finding latent roots. Does not the removal of the latent vector by the expression $(a - \lambda x\, x')$ involve at each stage the loss of more and more significant figures, and what happens if the vector chosen initially is orthogonal to the vector corresponding to the largest latent root?

MR. WILKINSON replied that a satisfactory estimate of the errors in successive vectors was a theoretical problem of some difficulty. At the moment an upper bound, which was clearly a gross overestimate, was all that he had been able to find. Whenever it had been possible to determine the accuracy of a solution, the errors had been astonishingly small. Moreover, in the symmetric case, the matrix which had been left when all the roots had been removed had, in all cases in his experience, elements none of which were greater than three in the least significant place. He emphasized that iteration was always continued until the latent vector had settled or was going through a cycle of values very close together, so that the vectors which were removed were always very accurate. Where the smallest roots required were small compared with the largest, the percentage error was of course greater in these roots, but such inaccuracy was inherent in the problem.

PROF. HARTREE disagreed with the last remark, pointing out that the matrix may have arisen from theoretical considerations in which case it would be exact.

MR. WILKINSON added that the vector chosen initially was zero and the whole process started because, in the division routine as programmed, the result of dividing zero by zero was equal to 31 successive ones. On no occasion so far had this vector proved to be orthogonal to the latent vector corresponding to the largest latent root, though it was clear from the behaviour of the iterates that it had sometimes been an unfortunate choice so that the largest latent root had taken longer to assert itself. Unless the matrix was of a rather special form, absolute orthogonality was extremely improbable.

MR. HEALEY (Rothamsted Experimental Station) said that if a set of linear equations arising in statistical work were ill-conditioned this had a significant meaning. Some solutions were much better determined than others and this was clear when the Gauss-Seidel process was used. Was there anything corresponding in the process of pivotal condensation?

MR. WILKINSON said that in the hundreds of sets of equations which had been solved on the Pilot ACE the worst case of ill-conditioning was on a set of order 40 where the original matrix elements had 9 significant decimal digits and the reduced matrix had a diagonal element of only 4 significant digits. In this case the back substitution gave a solution with 4 correct significant figures which was the best one could get from such a reduced matrix.

Dr. GOODWIN (NPL) remarked that on several occasions when a customer had sent in a very ill-conditioned set, the customer's theory had been wrong and the equations should have been singular! He suggested that extremely ill-conditioned sets should be viewed with grave suspicion.

19. THE NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS

by

L. Fox and H. H. Robertson

National Physical Laboratory

Introduction

Differential equations arise in many contexts. They may be involved in the investigation of the behaviour of some physical system, or they may be derived as the easiest method of evaluating particular functions, and in this sense they are of considerable assistance in the work of tabulating the functions of mathematical physics. We may accordingly require only a few figures in the solution, sometimes merely a graph, or we may need ten or more accurate decimal places in a mathematical table. The equations may be linear on non-linear in varying degrees of complexity, and their associated boundary conditions may be given at a single point, or may be shared among two or more points in the range of integration.

These considerations suggest that no single method will be satisfactory for all problems. A well-equipped computational centre should have at its disposal a variety of methods, from which the most appropriate can be selected for any given problem. There is a considerable literature on methods for use with desk computing machines, each new method aimed at improving either the convenience or accuracy of the calculation, or both. In the present state of development of high-speed computers we find that emphasis is naturally focussed on methods which are easy to code and which make the best use of the limited storage space. Speed and efficiency of input and output is another important consideration.

Though these questions are clearly of immediate importance, it is probable that, as coding becomes easier, an obvious requirement, and machines acquire more storage, the accuracy of desk machine methods will attain more importance than their inconvenience, and some of them will find favour also on high-speed computing machines.

For numerical purposes ordinary differential equations are best classified according to the position of their associated boundary conditions. If all these conditions are specified at the same point of the range of integration, we shall call the problem an <u>initial-value</u> problem: if boundary conditions are shared between two points in the range, we shall speak of a <u>boundary-value</u> problem. Our methods of solution will depend primarily on the class to which the given problem belongs.

Initial-value problems

Here a differential equation of order n has all its associated n boundary conditions given at the same point of the range. The techniques of solution all come under the general heading of "step-by-step" processes. These techniques can be subdivided further as follows.

1. Methods not involving finite-differences, of which the two most common are (a) Taylor-series method and (b) the Runge-Kutta method.

2. Methods which involve finite differences, but in which the differences are never formed. Lagrangian formulae are used throughout. The methods of W. E. Milne provide typical examples of this class.

3. Methods involving finite differences, making use of the difference tables. These methods again divide into two classes, of which the first attempts to obtain the correct answer at each step (prediction and almost immediate verification), while the second first obtains an approximate answer at every point, making subsequent correction, for example, by the so-called "difference-correction" method: the latter we shall call "indirect" methods.

4. Other methods, such as those of Rayleigh-Ritz, Galerkin, and "collocation", take a series of specially chosen functions which satisfy either the differential equation or the boundary conditions, and determine the constants involved in the series by solving a set of simultaneous algebraic equations.

We shall consider briefly the adaptation of most of these methods to high-speed computing machines. Only section (4) above receives no further mention.

The Taylor-Series method

Given the value of the function and its first n-1 derivatives, we can calculate the n^{th} derivative y from the differential equation, and successive derivatives y^{n+1}, y^{n+2} ... by successive differentiation of the given equation. We then calculate the function and its derivatives at the next pivotal point from the Taylor series

$$y(x_0 + h) = y_0 + hy'_0 + \frac{h^2}{2!} y''_0 + \dots \qquad (1)$$

$$y'(x_0 + h) = y'_0 + hy''_0 + \frac{h^2}{2!} y'''_0 + \dots$$

and so on. By this means we advance one step, and by changing the sign of h in (1) we have a powerful check on a previous value.

This method, which so far as we know has not been used on high-speed computers, should often be quite practicable. These machines are generally well adapted for summing series like (1), and in many cases successive derivatives are conveniently obtainable from recurrence relations. Full accuracy is achieved at each step, and there is no difficulty about changing the interval. The method is not so convenient when successive derivatives can be obtained only from very complicated expressions.

The Runge-Kutta method

This method can be applied only to an equation of first-order, or to a set of simultaneous equations of first order: it can therefore be used for any initial value problem by taking dependent variables y, y', y''.... There are several variants of this method, but a typical formula of reasonable accuracy for the production of the value $y(x_0 + h)$ is given by

$$y(x_0 + h) = y(x_0) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \qquad (2)$$

where

$$\left.\begin{aligned} k_1 &= hf(x_0, y_0) \\ k_2 &= hf(x_0 + \tfrac{1}{2}h,\ y_0 + \tfrac{1}{2}k_1) \\ k_3 &= hf(x_0 + \tfrac{1}{2}h,\ y_0 + \tfrac{1}{2}k_2) \\ k_4 &= hf(x_0 + h,\ y_0 + k_3) \end{aligned}\right\} \qquad (3)$$

and the differential equation is given by

$$y' = f(x, y) \qquad (4)$$

It can be verified that (2) agrees with the Taylor series as far as the term in h^4.

The convenience of this method is obvious. The main part of the work is the computation of $f(x,y)$ for various values of x and y, so that only one process is involved. There is no difficulty about changing the interval, and no special device is needed at the start. Few quantities have to be stored in the machine. On the other hand it is not easy to determine the truncation error at any point, or the accumulation of such error, and the usual check is a repetition at a smaller interval. This method has been described in detail by S. Gill (ref.1) and is now a standard process on most machines.

Methods using Lagrangian formulae

We turn now to methods which use finite-difference formulae in their Lagrangian form. Typical of these methods is that of W. E. Milne (ref.2). For the first-order equation (4) he finds a tentative value of y_{n+1} from the formula

$$y_{n+1} = y_{n-3} + \frac{4h}{3} (2y'_n + 2y'_{n-2} - y'_{n-1}), \tag{5}$$

calculates the corresponding value of y'_{n+1} from the differential equation and then gets a better value of y_{n+1} from the formula

$$y_{n+1} = y_{n-1} + \frac{h}{3} (y'_{n+1} + y'_{n-1} + 4y'_n) \tag{6}$$

Again the value of y_{n+1} so obtained agrees with the Taylor series as far as the term in h^4.

This method needs special treatment at the start, does not lend itself to easy change of interval and needs to store more quantities at each stage. Apart from these drawbacks it would seem to involve less computation per step than the Runge-Kutta method. The extension of both these methods to the solution of simultaneous first-order equations is obvious.

Special "predictor" and "corrector" formulae have been suggested by Milne for a second-order equation which involves no first derivative. We quote the predictor

$$y_{n+1} = y_n + y_{n-2} - y_{n-3} + \frac{h^2}{4} (5y''_n + 2y''_{n-1} + 5y''_{n-2}) \tag{7}$$

and the corrector

$$y_n = 2y_{n-1} - y_{n-2} + \frac{h^2}{12} (y''_n + 10y''_{n-1} + y''_{n-2}) \tag{8}$$

The methods of the last two paragraphs can give only limited accuracy. If fifth and higher differences of y are significant the results are in error, and can be improved only by repeating the processes at a smaller interval. We turn now to methods which use finite-differences, but as with the Taylor-series method can take account of all significant terms in the relevant formulae. First we consider methods whereby extrapolation is followed by almost immediate correction.

The Adams-Bashforth method

This method also applies directly only to first-order equations, and is again a matter of prediction followed by verification. The basic formula for prediction is given by

$$y_{n+1} = y_n + (1 + \frac{1}{2}\nabla + \frac{5}{12}\nabla^2 + \frac{3}{8}\nabla^3 + \ldots)hy'_n . \tag{9}$$

From this we can calculate a tentative y'_{n+1} from the differential equation, and obtain a better value of y_{n+1} from the formula

$$y_{n+1} = y_n + (1 - \frac{1}{2}\nabla - \frac{1}{12}\nabla^2 - \frac{1}{24}\nabla^3 \ldots)hy'_{n+1}. \tag{10}$$

Here less direct computation involving the differential equation is needed, but the method has the serious disadvantage of requiring to store more quantities, and often many more, at each step. The coefficients of successive differences, moreover, decrease only slowly, and rounding errors may accumulate rapidly.

Central-difference methods

The large coefficients in (9) and (10) can be avoided by the use of central-difference formulae, but central differences at a particular point are not available until several more steps have been performed. The general procedure used on desk machines for an equation of order n is to record the function, its various derivatives up to order n, and all the differences. To advance it is necessary to estimate some central differences, and the check is carried out at a later stage, dependent on the number of significant differences.

These methods do not appear to provide useful techniques for high-speed computing machines. Many quantities are stored and the cyclic process may need several repetitions if the original estimates are not sound. These difficulties, indeed, are apparent to desk computers, and the tendency is to use small enough intervals to ensure that no differences of very high order are significant, and to make estimation accurate. There is therefore here a practical limit, rather than a theoretical one, to the size of interval which can be used.

Indirect methods

Indirect methods try to remove these difficulties in two ways. First, all derivatives are expressed in terms of pivotal values, and "prediction" involves only pivotal values. Second, "correction" is performed only after a complete approximate solution has been obtained with the use of simple "predictors". Consider, for example, a second-order equation

$$y'' + y' f(x) + g(x,y) = 0 \tag{11}$$

Using the central-difference formulae

$$\left.\begin{aligned} h^2 y''_r &= (y_{r+1} + y_{r-1} - 2y_r) - \frac{1}{12}\delta^4 y_r + \frac{1}{90}\delta^6 y_r \ldots \\ h y'_r &= \frac{1}{2}(y_{r+1} - y_{r-1}) \quad \frac{1}{6}\mu\delta^3 y_r + \frac{1}{30}\mu\delta^5 y_r \ldots \end{aligned}\right\} \tag{12}$$

we can replace (11) by the equation

$$\left.\begin{aligned} & y_{r+1}\left(1 + \frac{h}{2} f_r\right) = -y_{r-1}\left(1 - \frac{h}{2} f_r\right) + 2y_r - h^2 g(x_r, y_r) - \Delta(y_r) \\ \text{where}\quad & \Delta = \left(-\frac{1}{12}\delta^4 + \frac{1}{90}\delta^6 \ldots\right) + hf\left(-\frac{1}{6}\mu\delta^3 + \frac{1}{30}\mu\delta^5 \ldots\right) \end{aligned}\right\} \tag{13}$$

In the first approximation we neglect Δ and calculate successive y_r by recurrence: two initial values are required which can be calculated from the given initial conditions. We then compute $\Delta(y_r)$ from the central difference of y_r, and obtain a correction η to y from the equations

$$\eta_{r+1}\left(1 + \frac{h}{2} f_r\right) = -\eta_{r-1}\left(1 - \frac{h}{2} f_r\right) + \eta_r\left\{2 - \frac{\partial g}{\partial y_r}(x_r, y_r)\right\} + \Delta(y_r) \tag{14}$$

with the two initial values of η equal to zero. Here we have neglected only the terms $\Delta(\eta_r)$, and further improvements can be effected if these are significant.

Though the equations seem formidable, there are several advantageous features of this method. First, the only quantities stored are sufficient pivotal values to enable all significant central differences to be obtained, and the difference corrections Δ calculated from the differences. Second, we can work at a large interval, thus reducing the accumulation of rounding and truncation errors. Third, the corrections η indicate the number of correct figures in the solution.

For linear equations this method is fast and accurate, and for non-linear equations of the form (11) there is little extra difficulty. A non-linear equation of the form

$$y'' = f(x,y,y'), \tag{15}$$

on the other hand, involves at each step the solution of a non-linear algebraic or transcendental equation. The difficulty depends on the form of the non-linearity, and Newton's method is often effective. The corrections η can nearly always be obtained from a linear recurrence relation, as in (14).

A second-order equation which involves no first derivative can be expressed in a form which involves a very small correction. The equation

$$y'' = f(x,y) \tag{16}$$

has the central-difference equivalent

$$\left\{y_{r+1} - \frac{1}{12} h^2 f(x_{r+1}, y_{r+1})\right\} + \left\{y_{r-1} - \frac{1}{12} h^2 f(x_{r-1}\ y_{r-1})\right\} - \left\{2y_r + \frac{5}{6} h^2 f(x_r, y_r)\right\} + \Delta\,(y_r) = 0 \tag{17}$$

where $\Delta = \frac{1}{240}\delta^6 - \frac{13}{15120}\delta^8 \ldots,$

an equation analogous to (8), used in Milne's method. Again this method is quick and powerful for linear differential equations, non-linear equations necessitating the solution at each step of a non-linear algebraic equation.

These techniques, and others involving first-order equations, have been discussed by Fox and Goodwin for use with desk calculating machines (ref.3) Their adaptation to high-speed computers appears practicable and often desirable.

Boundary-value problems

When we have a second-order equation for which a boundary condition is given at each end of the range of integration, two possible procedures are available. First, we can use arbitrary second conditions at one point, integrate by step-by-step methods, and combine solutions obtained in this way so that the other boundary condition is satisfied. This method is attractive but often laborious. In particular, whenever the complementary function increases at a rate faster than the required particular integral, rounding errors increase and may become so great that the step-by-step solutions require the retention of a large number of guarding figures to absorb the building-up error. In some cases the number of guarding figures is prohibitively large. Second, we can use other techniques in which both boundary conditions are satisfied simultaneously. We now consider the latter method, applied to second-order equations.

We take first a linear equation of the form

$$y'' + f(x)y' + g(x)y = k(x), \tag{18}$$

for which the finite-difference equation is given by

$$\begin{gathered} y_{r+1}\left(1 + \frac{h}{2} f_r\right) + y_{r-1}\left(1 - \frac{h}{2} f_r\right) - (2 - h^2 g_r) y_r + \Delta(y_r) = h^2 k_r \\ \Delta = -\frac{1}{12}\delta^4 + \frac{1}{90}\delta^6 \ldots + hf\left(-\frac{1}{6}\mu\delta^3 + \frac{1}{30}\mu\delta^5\right) \end{gathered} \tag{19}$$

Using the "Indirect Methods" previously explained we obtain a first approximation by solving (19) with Δ neglected. The simplest boundary conditions specify y_0 and y_n, the two end values, so that the set of equations (19) are now soluble as a set of simultaneous algebraic equations. For this a nice technique is available, described by Thomas (ref.4) and based on the theorem that a square matrix A can be expressed as the product LU of lower and upper triangular matrices. If the right-hand side of the equations is represented by a vector b, then we solve Ay = b by first carrying out the triangular decomposition, then solving for an auxiliary vector z from the equation Lz = b, finally obtaining y from Uy = z. The matrix A has a simple form, and this is reflected in the forms of L and U, and all the computations can be expressed by simple formulae. The matrices are shown below, gaps denoting zero elements.

$$\begin{array}{ccccc} & & A & & \\ a_{11} & a_{12} & & & \\ a_{21} & a_{22} & a_{23} & & \\ & a_{32} & a_{33} & a_{34} & \\ & & a_{43} & a_{44} & a_{45} \end{array} \qquad \begin{array}{cccc} & & L & \\ 1 & & & \\ l_{21} & 1 & & \\ & l_{32} & 1 & \\ & & l_{43} & 1 \end{array} \qquad \begin{array}{ccccc} & & U & & \\ u_{11} & u_{12} & & & \\ & u_{22} & u_{23} & & \\ & & u_{33} & u_{34} & \\ & & & u_{44} & u_{45} \end{array}$$

The computation is summarized by the following equations:

(i) determination of L and U

$$\left.\begin{array}{ll} \text{(special)} & u_{11} = a_{11} \\ \text{(general)} & \left\{\begin{array}{l} u_{r,r+1} = a_{r,r+1} \\ l_{r+1,r}\, u_{rr} = a_{r+1,r} \\ l_{r+1,r}\, u_{r,r+1} + u_{r+1,r+1} = a_{r+1,r+1} \end{array}\right. \end{array}\right\} \tag{20}$$

(ii) determination of z

$$\left.\begin{array}{ll} \text{(special)} & z_r = b_1 \\ \text{(general)} & l_{r+1,r}\, z_r + z_{r+1} = b_{r+1} \end{array}\right\} \tag{21}$$

(iii) determination of y

$$\left.\begin{array}{ll} \text{(special)} & u_{nn}\, y_n = z_n \\ \text{(general)} & u_{r,r}\, y_r + u_{r,r+1}\, y_{r+1} = z_{r+1} \end{array}\right\} \tag{22}$$

This process is easy to code and quick to perform. When the first approximation has been obtained the difference corrections can be calculated and the process repeated.

For non-linear equations the matter is more complicated. The simultaneous algebraic equations are non-linear, and no very satisfactory technique exists for their solution. They can often, however, be solved by an iterative procedure involving alterations in the coefficients but not the form of A after each iteration. For example, if y is an approximate solution of the non-linear equations (17) (with Δ neglected), now regarded as simultaneous equations, a correction η can be found from the linear equations typified by

$$\eta_{r+1}\left\{1 - \frac{1}{12}h^2\left(\frac{\partial f}{\partial y}\right)_{r+1}\right\} + \eta_{r-1}\left\{1 - \frac{1}{12}h^2\left(\frac{\partial f}{\partial y}\right)_{r-1}\right\} - \eta_r\left\{2 + \frac{5}{6}h^2\left(\frac{\partial f}{\partial y}\right)_r\right\} = 0 \tag{23}$$

Eigen-value problems.

Eigen-value problems are of boundary-value type and homogeneous. If we use the same finite-difference equations as in the previous section, a linear second-order equation reduces to the determination of the eigen-values and vectors of the matrix equation

$$(A - I\lambda)y = 0 \tag{24}$$

for which standard techniques are available on most high-speed computers. This method has decided advantages over the corresponding step-by-step method. No initial knowledge of an approximation to the required solution is needed, and several solutions, if required, can be calculated almost simultaneously from (24).

Programming on the ACE Pilot Model at NPL

We now give a brief description of the programmes so far produced for the solution of ordinary differential equations using the ACE Pilot Model at NPL.

1. *The Runge-Kutta method*

We use Gill's adaptation of this method, which takes equations somewhat different from the set (3), reducing the number of storage locations for each equation from four to three. The programme has 74 instructions, including a multiplication routine, solves up to 10 simultaneous first-order equations, and can be extended to 32 with no more instructions. The time taken for a single step of one equation is (70 + 4T) ms where T ms is the time required to compute the function f(x,y) in (4). It is convenient to use, and the facility with which the interval of integration can be halved, while keeping down the rounding error, lends itself to an easy determination of the truncation and building-up errors. This is done effectively by retaining a few guarding figures, and it has been found useful, when integrating over a long range, to punch out this extra word as well as the function, in order to restart the process at any point and obtain identical results.

2. *Methods using Lagrangian Formulae*

We have not programmed any of Milne's methods, but a programme for "prediction followed by immediate verification", using simple Lagrangian formulae, has been produced for the equation

$$y'' = f(x)y, \tag{25}$$

and can be extended almost trivially to the case

$$y'' = f(x,y). \tag{26}$$

The simple finite-difference recurrence equation is given by

$$\left.\begin{aligned} y_1 &= (2 - h^2 f_0)y_0 - y_{-1} + \Delta(y_0) \\ \Delta &= \frac{1}{12}\delta^4 - \frac{1}{90}\delta^6 \text{ ——} \end{aligned}\right\} \tag{27}$$

We assume $\delta^6 y_0$ to be zero and use the differential equation to write

$$\delta^4 y_0 = -\delta^2 h^2 f_0 \, y_0 \tag{28}$$

In general, a small interval will have to be used, so there are less significant figures in $h^2 f_0 y_0$ than in the function itself. By computing $f_0 y_0$ to the same number of significant figures as in y, we are able to carry a few extra figures in the function which is used to absorb the rounding error due to taking a large number of steps.

We write

$$\left.\begin{aligned} y &= Y + w \\ F &= -2^p h^2 f \end{aligned}\right\} \tag{29}$$

where Y is a single-length word and w is a word of p binary digits designed to absorb the rounding error, when the formula (27) becomes

$$\left.\begin{aligned} \bar{Y}_1 + \bar{W}_1 &= 2Y_0 - Y_{-1} + 2W_0 - W_{-1} + 2^{-p} F_0 Y_0 \\ Y_1 + w_1 &= \bar{Y}_1 + \bar{w}_1 - \frac{2^{-p}}{12}(-F_{-1}\bar{Y}_{-1} + 2F_0\bar{Y}_0 - F_1\bar{Y}_1) \end{aligned}\right\} \tag{30}$$

At any stage $\bar{Y}$ and $\bar{w}$ are obtained from the first of (30), in which the Δ of (27) is neglected (prediction), and immediately corrected by the inclusion of the second term on the right of the second of (30), corresponding to the inclusion of $\Delta(\bar{Y} + \bar{w})$. 55 instructions, including a multiplication subroutine, are required, and 11 storage locations for the functions Y, F, FY and $\delta^2 FY$. The facility for shifting the contents of a delay line is used to ensure that the quantities required at one step are in the correct positions for their application in the next step. The time taken per step is 15 ms plus the time required for computing f(x).

A second programme exists, similar to the above, in which the fourth difference is taken to be zero. 38 instructions, including a multiplication subroutine, are required and 6 storage locations for the function. The time for one step is 10 ms plus the time for calculating f(x).

There is also a third programme for solving (25), which takes advantage of the absence of the first derivative, using the linear form of (17) given by

$$\left.\begin{aligned} &(1 - \frac{h^2}{12}f_1)y_1 = (2 + \frac{10h^2}{12}f_0)y_0 - (1 - \frac{h^2}{12}f_{-1})y_{-1} + \Delta(y_0) \\ &\text{where} \qquad \Delta = -\frac{1}{240}\delta^6 y_0 \text{ —} \end{aligned}\right\} \tag{31}$$

Starting with given values y_{-1} and y_0 we recur three steps forwards and two backwards, then compute $\delta^6 y_0$ by the Lagrangian formula. The pivotal value y_1 is then improved and we repeat the process with y_0 and the corrected y_1 as new starting values. The procedure is clumsy and lengthy but may sometimes be useful when y is rapidly increasing, for we need left-shift only two numbers when they reach a predetermined size.

3. *Indirect methods.*

A programme exists for the application of the indirect central-difference method for solving the more general equation

$$\frac{d^2y}{dx^2} = f(x)y + g(x), \tag{32}$$

which again lacks its first derivative. The recurrence relation corresponding to (31) is now

$$(1 - \frac{h^2}{12}f_1)y_1 = (2 + \frac{10h^2}{12}f_0)y_0 - (1 - \frac{h^2}{12}f_1)y_1 + \frac{h^2}{12}(g_1 + 10g_0 + g_{-1}) + \Delta(y_1) \tag{33}$$

and we ignore in Δ differences of orders greater than 12. There are two parts of the programme. From the first we calculate an approximation using (33) with $\Delta = 0$, difference y, calculate Δ and punch out the values of y, $\frac{h^2}{12}$ f and Δ at all points of the range. The second programme reads in these numbers in large batches and we can obtain the correction η from the equation

$$(1 - \frac{h^2}{12}f_1)\eta_1 = (2 + \frac{10h^2}{12} f_0)\eta_0 - (1 - \frac{h^2}{12}f_{-1})\eta_{-1} + \Delta(y) \tag{34}$$

which neglects $\frac{-1}{240}\delta^6\eta$. The value of η is added to the corresponding value of y at each stage and punched out.

For the calculation of Δ we store the appropriate number of central differences, a new line of backward differences being formed as soon as a new pivotal value becomes available. Twelfth differences are retained in Δ, and a warning is automatically given if these go beyond a certain size, serving as a check against accidental errors. In the second part of the programme a similar warning is given if $\frac{1}{240}\delta^6\eta$ is not negligible.

To obtain the required Δ at the first point of the range the recurrence relation (33) is used to compute several pivotal values in the reverse direction. This is performed at the start of the first programme and the instructions are then overwritten with those for producing the differences and the corrections Δ.

The programme was designed to fit in with the characteristics of the Hollerith punch. If it is desired to punch out numbers, then allow the machine to do some computing before calling again for a punch cycle, the clutch may have disengaged and will have to complete the cycle before engaging again. This fact may affect the time taken for a problem by as much as a factor of two, so it is desirable where possible to have the punch running continuously or for the duration of several cards. The programme does all the computing between rows of punching and 4, 2 or 1 values of y can be punched per card provided there is sufficient time to calculate the required number of values of f and g between cards. Otherwise f and g can be precomputed and fed into the machine in large numbers.

The main programme takes 160 instructions including subroutines for multiplication and division and uses 64 storage locations for the functions, leaving 96 instructions and 32 storage locations in which to form or store the functions f and g. The time taken when running continuously to punch out 100 values of the un-corrected function is about 15 seconds. These values can then be corrected and punched out in decimal by the second part of the programme in something under a minute. A programme is in course of preparation for the solution of the general linear equation

$$\frac{d^2y}{dx^2} + f(x)\ y' + g(x)y + w(x) = 0, \tag{35}$$

using the same type of indirect method.

In this case the first programme must punch out the four quantities y, $\frac{h^2}{12}$ f, $\frac{h^2}{12}$ g and Δ, for each step, so it has been designed to punch out a maximum of two values of y per card, and tenth differences are taken into account.

In all the programmes of this kind it is a trivial matter to ignore differences of orders less than ten or twelve when these are insignificant, which might happen if the chosen interval is too small, or at a point at which it is really desirable to increase the interval.

4. *Matrix Methods*

The programme in use for solving boundary value problems of the type (19) has already been described by J. H. Wilkinson. It can use a maximum of 64 pivotal points, working with the equivalent of about fourteen decimal figures. It is a simple matter to form and to repeat the solution with this correction as the right-hand side of the equations.

Checking

Many problems solved on automatic high-speed computers, for example the evaluation of a single algebraic or transcendental expression, can be checked only by repetition: such a repetition would with advantage involve, if possible, different orders of the computational steps. In the solution of differential equations, however, differencing and spot checking should guard completely against the possibility of accidental error. Systematic building-up error can best be detected by a previous mathematical investigation: the latter is perhaps even more desirable for work with these machines than for work with desk machines.

References

1. S. GILL. *Proc. Camb. phil. Soc. 1950*, 47, 96.
2. W. E. MILNE. Numerical Calculus. Published by *Princeton University Press, 1949*.
3. L. FOX and E. T. GOODWIN. *Proc. Camb. phil. Soc. 1949*, 45, 373.
4. L. H. THOMAS. Elliptic problems in linear difference equations over a network. Watson Scientific Computing Laboratory Note.

Discussion

DR. MILLER (Cambridge University Mathematical Laboratory) agreed that, compared with difference methods, the Taylor series method often has the disadvantage of necessitating the evaluation of complicated derivatives. Frequently however it is easy to obtain a recurrence relation for the derivatives and this fact makes the class of equations which are amenable to the Taylor series method larger than is often supposed. The method has been coded on EDSAC by R. Brooker for the evaluation of the error function in the complex plane by integrating the relevant differential equation along lines parallel to the imaginary axis.

With regard to central difference methods he thought that in spite of the drawbacks mentioned in the paper it should be possible to produce a practicable programme on some high-speed computing machines. Estimation of the differences could be made by the machine itself, taking as a first guess, for example, some fixed multiples of the last three differences of the same order.

DR. FOX mentioned that the use of recurrence relations for obtaining successive derivatives was in fact mentioned in his paper.

DR. BLANCH (National Bureau of Standards) remarked that if the homogeneous part of an ordinary differential equation is linear in form then it is sometimes advantageous to use an integral representation of the equation on account of the greater stability of the solution. She had recently written a paper on this aspect. She enquired whether the Taylor series method needed the use of a smaller interval than other methods.

DR. MILLER replied that just the opposite was true; the Taylor series method could be used at very wide intervals. As an extreme example, he had carried out the integration of Bessel's equation at a unit interval keeping 25 decimals; the number of terms required varied between 25 and 80. There is no theoretical difficulty in the use of wide intervals as there may be with differences; the Taylor series always converges within its circle of convergence.

DR. TOCHER (Imperial College, London) said that Milne's methods were being used on the relay computer at Imperial College. A difficulty had been encountered in the solution of the radial equation.

$$\frac{d^2y}{dr^2} = f(r)y,$$

where, for small values of r

$$f(r) = k(k-1)r^{-2} + O(r^{-1}).$$

Near $r = 0$ the solution behaves like constant $x\ r^k$ and for large k a prohibitively small interval would be required. The usual transformation $y = r^k u$ produced an equation in which a term in the first derivative is present and so was not in the standard form of Milne. However, R. S. Lee had found some recurrence relations of Milne's type for dealing with this kind of equation.

Some work was also going on concerning predictor formulae of Milne's type for the first order equation

$$\frac{dy}{dx} = f(x, y).$$

The general form of such a formula is given by

$$y_1 = a_0 y_0 + a_1 y_{-1} + \dots + a_n y_{-n} + b_0 q_0 + b_1 q_{-1} + \dots + b_m q_{-m},$$

where $q = h dy/dx$. This uses $n + m - 2$ storage positions in the machine. They were finding out at Imperial College which formula has the smallest truncation error and which leads to the smallest rounding error for fixed values of $n+m$ ranging from one to ten.

Prof. HARTREE (Cambridge University) said that Dr. Fox had mentioned two ways in which higher differences could be taken into account when using difference formulae. These were the central difference processes in which a correction is made at each step and the after-correction or difference-correction processes. A procedure intermediate to these which is worth investigation consists of integrating at two intervals simultaneously. Four steps are taken at an interval h and two at an interval $2h$ and then Richardson's h^2 extrapolation formula is used to eliminate the leading term in the truncation error.

The method described in the paper for solving the approximate finite difference equation by the process of matrix inversion was readily applicable to the solution of partial differential equations of the elliptic kind. He wondered if this procedure was essentially the same as that referred to by Dahlquist in the discussion of the preceding paper.

Mr. DAHLQUIST (Swedish Board for Computing Machinery) replied that he believed this was so. He also remarked that he had been carrying out an investigation of predictor formulae similar to that mentioned by Tocher. One feature of such a formula was that if it was based on k previous points and had an error term of order h^{2p} then it was only stable if $p > k + 2$.

PROF. WIJNGAARDEN (Mathematisch Centrum Amsterdam) remarked that numerical methods for solving second-order differential equations generally gave less accuracy in the derivative than the function, but the solution of two simultaneous first-order equations determined both quantities to the same accuracy.

20. The Solution of Partial Differential Equations by Automatic Calculating Machines

by

N. E. Hoskin

Manchester University

Any method for performing a calculation with the use of desk machines can be programmed for an automatic computer but, owing to the inherent limitations of such calculators the best method for hand computation is not necessarily the most practical when used on automatic machines. This difference arises because human intuition can no longer play a part, the storage space is limited, and also because provision has to be made to deal with all possible conditions which may occur.

The differential equations specifically mentioned below are those of second order in two independent variables, but the methods are in general capable of extension to a larger number of dimensions. The storage requirements will increase with the number of independent variables and the organization will become more complex. For most partial differential equations the high-speed store of an automatic machine will be much too small to meet the storage requirements and an auxiliary store will be necessary. This may be one of several types such as a magnetic drum or magnetic tape, or the results may even be taken out on punched cards or tape and fed back into the machine when required. Since transfers to or from the auxiliary store are relatively slow processes, the number of such transfers should be kept as low as possible. More discussion of this point will be given when considering elliptic differential equations.

The methods of solution will depend on the type of differential equation considered and it will be simplest to consider each type separately. We may mention first the Monte Carlo method since it may be adapted to solve these problems where the value of the function is specified on the boundary of the region over which the differential equation is valid. However, the convergence is extremely poor and the number of runs necessary to achieve a desired accuracy is usually excessive. It may be a practical method if the behaviour of the function is only required in a small section of the domain of integration.

It is assumed throughout that any method chosen will be stable, so that errors do not build up excessively. This would have to be considered even if the solution were being obtained by hand but if an unstable method is being used on an automatic machine the effects will probably go unnoticed until the error becomes so great that the programme breaks down.

Parabolic Equations

We will discuss the methods of solution by reference to the diffusion equation in one dimension

$$\frac{\partial f}{\partial t} = p(x)\frac{\partial^2 f}{\partial x^2}, \tag{1}$$

which governs conditions in the region

$$0 \leqslant x \leqslant 1, \qquad 0 \leqslant t \leqslant \infty,$$

and with boundary conditions given on $x = 0$, $x = 1$, and $t = 0$. These boundary conditions are those which often occur in practice, and since the conditions are open in time it is usually easiest to integrate in the direction of increasing t.

A finite difference equation may be constructed by replacing the derivatives by finite differences between function values on a rectangular mesh with intervals Δt and Δx. The difference equation will depend on the way in which the derivatives have been replaced (here the question of stability will have arisen) but in all cases an equation is obtained which relates the values of the functions at time $(t + \Delta t)$ with those at time t (and perhaps those at time $(t - \Delta t)$). The resulting equations may be solved for the set of function values lying along the mesh line $(t + \Delta t)$ and the same programme may be used to solve for the functions at $(t + 2\Delta t)$ by changing t to $t + \Delta t$ and replacing $f(x, t)$ by $f(x, t + \Delta t)$ etc. The boundary conditions on $x = 0$ and $x = 1$ can be taken into account fairly simply. Even if normal derivatives are involved it will add only two equations to the total number of simultaneous equations to be solved.

An alternative method is to replace the derivatives with respect to only one of the independent variables by corresponding finite differences. In this way ordinary differential equations are obtained. In equation (1) this may be done in two ways: (a) by replacing the derivative with respect to x and thus obtaining a set of simultaneous first order differential equations which may be integrated along the lines $x = n\Delta x$, or (b) by replacing the time derivative and obtaining a second order differential equation which has boundary conditions on $x = 0$ and $x = 1$. The disadvantage of having closed boundary conditions makes (b) less amenable to machine use than (a) and in addition the process (a) is immediately extensible to more than one space variable and to non-linear equations.

Hyperbolic Equations

The methods mentioned above for parabolic differential equations are also suitable for solving hyperbolic equations and method (a) may be used without modification if the equation possesses open boundary conditions in time.

However, since the characteristic curves of a hyperbolic differential equation are real, it is possible to programme a machine to obtain the function values at the points formed by the intersection of the two families of characteristics. This may be done even if the equation is not linear, although it may involve an iterative process at each step.

Elliptic Equations

When a solution to a hyperbolic or parabolic equation is sought by the method of finite differences it is at worst only necessary to solve a set of simultaneous equations for the values of the function along one mesh line t = constant since the function depends on previous and not on future values. However for an elliptic differential equation the value of the function at any point in a region depends on conditions holding on a curve totally enclosing that region. Therefore, when the equivalent finite difference problem is formed it is necessary to solve a set of simultaneous equations for the values of the function at each mesh point, including points on the boundary which are included by expressing the boundary conditions in finite difference form. Therefore the number of equations is much greater in the case of an elliptic equation than for a parabolic or hyperbolic equation of corresponding complexity. The problem of storage is now much more important and is a dominating factor in the formation of a programme to solve the equation on an automatic computer. An auxiliary store is a necessity and, since the access time to this store is relatively lengthy, processes which require a large number of transfers to or from the auxiliary store are to be avoided if at all possible. However it is often possible to transfer information in blocks of numbers (rather than in single numbers) and this fact may be put to good effect. Consider the problem of solving the equation

$$\Delta^2 f = \frac{4\pi\rho}{D} \tag{2}$$

to be solved on a rectangular net. We may form the corresponding difference equation

$$\sum_{i=0}^{n} a_i f_i = \frac{4\pi\rho_0}{D} \tag{3}$$

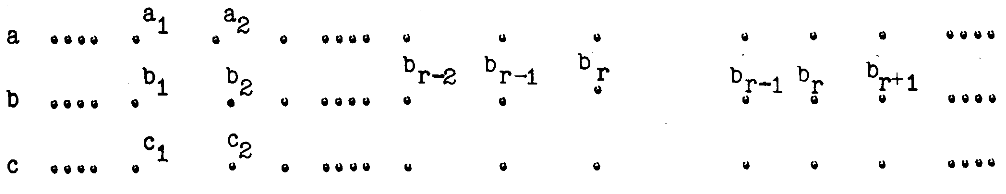
connecting the value of the function at a point with the values of its neighbours. One may choose the simplest form, using a square mesh of length h,

$$\left(\begin{array}{ccc} & 1 & \\ 1 & -4 & 1 \\ & 1 & \end{array}\right) f + 0\,(h^4) = \frac{4\pi\rho_0}{D} \tag{4}$$

or the more accurate form

$$\left(\begin{array}{ccc} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{array}\right) f + 0(h^8) = \frac{4\pi\rho_0}{D} \tag{5}$$

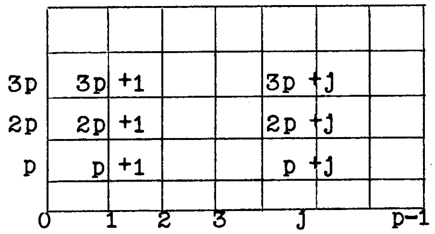
and if any point is being considered three rows (or colums) are involved. Therefore a saving in time is effected if it is possible to transfer a row (or column) of function values in a single operation. Also if it is possible to hold three rows of function values simultaneously in the working store the function values along the whole row may be adjusted before another transfer is nesessary. If the number of points in the row is too large to be transferred at once, or if the space available in the working store is insufficient to hold the contents of three complete rows the data must be transferred in sections.



Set 1 Set 2

The contents of rows a, b and c containing the numbers 1 to r (i.e. transfers are made in blocks of r numbers) are brought down as Set 1. This enables the values of the functions b_1 to b_{r-1} to be adjusted. The function b_r cannot be adjusted since b_{r+1} (perhaps a_{r+1} and c_{r+1}) is not available in the working store. Therefore some duplication has to be made as shown. *i.e.* b_{r-1} and b_r must also be present in Set 2. If more complicated difference equations are chosen the problem of storage will become progressively more complex, *e.g.* if the equation has non-constant coefficients it may be necessary to store these, although it will sometimes be more practical to calculate them when required. However, if more complicated equations are considered in which the error term is much smaller, less points need be considered.

The task of devising a programme to deal with all types of boundary is a formidable one but in many cases the specific problem will be of sufficient importance to justify time being spent in devising individual methods. However, some possibilities suggest themselves when constructing a general routine. Let us first consider a rectangular net covering a rectangular region. In this case, the problem of recognising a boundary point is very simple. The points on the mesh may be numbered in order; each number being of the form $m = pk + j$, where p is the number of points in each row and j specifies the distance along each row.



The boundary points are then given by the following conditions:

$$k = 0, \; k = \text{maximum value of } k, \; j = 0, \; j = p - 1.$$

All other points are interior points.

This system is so compact and suitable for a machine that it is desirable to transform the domain of applicability of the differential equation to a rectangular region, if such a transformation is possible. This results in the difference equation having non-constant coefficients, a serious disadvantage in hand computation, but when using an automatic machine this is outweighed by the ease with which boundary conditions can be taken into account. This is especially so since the normal to the boundary lies along a mesh line.

In some cases it will be impossible to transform the region to a rectangular region and in this case the problem becomes more complicated. However one may label each type of point by means of a tag-number and this can be done by using the least significant digits of the value of the function itself. In this way, by an analysis of the tag the programme can distinguish between interior and boundary points and use the appropriate difference equation to adjust the function value. The details of the boundary conditions will have to be stored together with the value of the function at the boundary point. These may be stored in storage locations adjacent to the function value or some other location which may be found by further analysis of the tag-number.

When the elliptic differential equation is replaced by a finite difference equation, a set of a large number of equations must be solved simultaneously. When the problem is done by hand, the obvious choice would be the relaxation process but this has severe disadvantages when applied to machine use, the main one being that the choice of the next mesh point to be adjusted depends on a condition satisfied by the present value of the residual at that point. This means that the value of the residual at every point of the mesh must be inspected; this is extremely wasteful of machine time. Also one of the advantages of the relaxation process when used by hand is that all calculations are performed with small numbers (the residuals) but this advantage disappears when the calculations are performed by an automatic computer since it is no shorter or easier to calculate with numbers of few digits than it is with numbers just within the range of the machine.

Therefore a straightforward iteration process seems most appropriate for machine use, but the form of the process is arbitrary. We again consider points on a rectangular net of square mesh, using the same numbering as before, and denote by $\phi^{n}_{j,k}$ the value after n iterations of the function at the point $m = pk + j$. The simplest iteration process is that known as the Richardson process which for Laplace's equation applies a correction to each point, thus

$$\phi^{n+1}_{j,k} = \phi^{n}_{j,k} + \alpha\left[\phi^{n}_{j-1,k} + \phi^{n}_{j+1,k} + \phi^{n}_{j,k-1} + \phi^{n}_{j,k+1} - 4\phi^{n}_{j,k}\right]$$

Usually $\alpha = \frac{1}{4}$ since this value gives optimum convergence and also results in the disappearance of the term $\phi^{n}_{j,k}$.

A modification of this process is the Liebmann process which applies this correction in succession using corrected values already obtained. If in the net considered here the process is started at $m = 0$ and continues with increasing m, the equation becomes

$$\phi^{n+1}_{j,k} = \phi^{n}_{j,k} + \alpha\left[\phi^{n+1}_{j-1,k} + \phi^{n}_{j+1,k} + \phi^{n+1}_{j,k-1} + \phi^{n}_{j,k+1} - 4\phi^{n}_{j,k}\right]$$

and again α is usually taken to be $\frac{1}{4}$; the residual at the pivotal point is then temporarily reduced to zero with each application. The convergence of this process is better than that of the Richardson process but it has the disadvantage of having to retain the corrected value in the working store for use in adjusting subsequent points.

When the relaxation process is used in computing by hand, greater convergence may be achieved by "over relaxation" and this is also true when an automatic computer is used. The convergence of the Liebmann process is improved by choosing $\alpha > \frac{1}{4}$. For Laplace's equation on a mesh of pq points (where p and q are large), Frankel has shown that the optimum value of α is

$$\alpha \approx \frac{1}{2} - \frac{\sqrt{2}\pi}{4}\left(\frac{1}{p^2} + \frac{1}{q^2}\right)^{\frac{1}{2}}$$

For more complicated equations it may be difficult to calculate the actual optimum value of α but a value may be found empirically. In a problem actually considered solutions were found for various values of α and the optimum value was found to be $\alpha = \frac{3}{8}$. This value should be as good as any other arbitrary choice unless the actual optimum value can be calculated.

The convergence may also be improved by retaining previous solutions (if this is possible without overtaxing the auxiliary store). In order to improve the accuracy of the solution it is possible to use the difference correction technique and thus take higher differences into account. Or again the solution may be found on a net of different mesh length and an improved solution found by Richardson's h^2 - extrapolation process.

Throughout the planning of a programme one has to consider the factors of speed, accuracy and space. The three are interdependent and the programmer has to attempt to obtain the maximum efficiency by considering all three. The main point to remember is that much time will be wasted if the programme has to deal with many possibilities. It is better to have a complicated equation where all complications occur in an ordered manner (*e.g.* non-constant coefficients) than to have a simple equation which has to be applied in a non-routine manner.

Discussion

PROF. HARTREE (Cambridge University) opened the discussion with two allied warnings. Just as it was dangerous to stipulate general principles for programming when concerned only with a projected machine, so also broad principles relevant to the numerical solution of both parabolic and hyperbolic equations could be misleading. For example, although, as in parabolic equations, most problems involving hyperbolic equations had boundary conditions of the open type, cases did exist in problems with periodic solutions in the time variable which had, effectively, closed boundary conditions.

PROF. HARTREE'S next point concerned parabolic equations. Suppose the time derivative has been replaced by a finite difference approximation. Then the step-by-step integration of the second-order equation in the space direction is unstable, and it is necessary to use relaxation, or equivalently the process of matrix inversion of the approximate finite difference equations described in the previous paper.

Thirdly, he queried whether it was wise to consider parabolic and hyperbolic equations together; he himself regarded hyperbolic as one stage more difficult than parabolic equations.

His fourth point was to remind his audience of the fact (implied in Hoskin's paper) that if the coefficients of the second derivatives in a hyperbolic equation were functions only of the independent variables, then it was possible to determine initially the whole mesh of characteristic curves; but if these coefficients involved the dependent variable it was necessary to obtain both the characteristics and the function values simultaneously at each step. An obvious example of this latter case was the equations of supersonic flow, where, incidentally, there were effectively four simultaneous equations, not necessarily algebraic, to solve at each step.

Fifthly, he admitted that he used to think that the possibility of relaxing a partial differential equation depended, merely, on whether it was elliptic or not. But consideration of the simplest relaxation pattern for the elliptic equation $\nabla^2\phi = k\phi$ shewed that if k were positive, relaxation was easier the larger k was, but for negative k, relaxation was impracticable. Conversely, although on first sight the simplest finite difference approximation to the Wave equation

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2 u}{\partial t^2}$$

in the x, ct plane was the worst possible from the relaxation point of view, it was immediately soluble. Moreover, since the general solution is $F(x + ct) + G(x - ct)$, the finite difference equations and the differential equation have a common solution; so that there is no truncation error.

Sixthly, he quoted an example of an elliptic equation from a wave guide calculation where a purely mechanical process of relaxation did not in fact converge to the correct solution.

He concluded by an appeal for more research with desk machines on all types of partial differential equations before too much energy was devoted to solving them on automatic computers.

MR. HOSKIN replied that the greatest difficulty of using relaxation methods on automatic computers was the considerable amount of organization required.

DR. HARTREE replied that this was far worse for non-linear equations.

MR. HOSKIN then gave as an example the non-linear equation $\nabla^2\phi = \sinh\phi$, which he had actually solved on the Manchester machine. He had transformed the independent variables so that the boundary conditions became rectilinear, but the equation became even more complicated. Dr. Hartree commented that the original equations could be written as

$$\nabla^2\phi = \left(\frac{\sinh\phi}{\phi}\right)\phi ,$$

and the coefficient of ϕ on the right hand side was positive, so that this elliptic equation belonged to the class for which relaxation was easily used.

DR. FOX (NPL) agreed that the matrix inversion process for the space-wise equations was very satisfactory when solving parabolic equations step by step in time. With regard to these elliptic equations which are not suitable for relaxation he had encountered this difficulty in two kinds of problem. Firstly, problems of forced vibration demand the solution of the equation

$$\nabla^2\phi + k^2\phi = f(x,y).$$

Unrefined relaxation was not necessarily a convergent process, but the finite difference equations could quite readily be solved directly, by the Gauss - elimination process for example. Secondly, eigen-value problems may lead to the equation

$$\nabla^2\phi + k^2\phi = 0,$$

where the value or values of k which provided non-trivial solutions satisfying the boundary conditions were required, in addition to the solution ϕ. Here the difficulty was not easily overcome, but general experience and a process of using changes in k to liquidate the residuals had enabled him to obtain solutions. He concluded by remarking that he regarded elliptic equations as harder to deal with than hyperbolic or parabolic especially when they involved conditions on curved boundaries. More research was required.

DR. LEE (Admiralty Research Laboratory) suggested the possibility of speeding the convergence by letting the quantity mentioned in Hoskin's paper in connexion with the Liebmann process, vary from point to point. He outlined a process for choosing α as the solution progressed.

Finally MR. WILKINSON observed that when using the method of characteristics, it was usually possible to get a very good initial guess from the differences to the solution by iteration of the four simultaneous equations mentioned by Prof. Hartree.

THE UTILIZATION OF COMPUTING MACHINES - II

Chairman: Dr. L. Fox

21. Mathematical Tables

by

E. T. Goodwin

National Physical Laboratory

A. THE PRODUCTION OF MATHEMATICAL TABLES BY A HIGH-SPEED DIGITAL COMPUTER

Introduction

The effect of the advent of high-speed digital computers on the preparation of mathematical tables has two distinct aspects. We may be concerned with the preparation by the machine of tables for general use and this I shall discuss in the first part of this paper; in part B, I shall consider the preparation of mathematical tables for direct use within the machine in the course of its work.

In discussing these two topics we shall need to consider four broad classes of mathematical tables. These are tables of elementary functions, such as the exponential or trigonometric functions; tables of higher mathematical functions, such as the exponential integral or Bessel functions; tables of "physical functions", usually simple functions containing experimentally determined constants; and truly "physical tables" which have been obtained from the results of experiments, the drag law for projectiles being a good example of this type of table. Of these, it is the tables of higher mathematical functions whose production will be most profoundly affected by the use of electronic machines and all the points I wish to make will be brought out by reference to this class only.

The Immediate effects of Electronic Computation

The three main stages in the production of a mathematical table are planning, computation and publication. For many years computation and publication have afforded problems of roughly equal complexity, though recently publication difficulties have tended to increase. The immediate effect of using high-speed computers is to reduce the labour of computation by a factor of the order of a hundred. That it will be impossible to continue publication of tables produced in this way by letterpress is clear for two reasons; the printers could not possibly keep pace with the output of tables; also it is now possible to compute many more large tables and the cost of publishing them all in this way would be prohibitive.

It has been suggested that the solution to this problem is to dispense with publication and to make the table available on, for example, punched cards, or the table might be planned and programmed but the computation postponed until a specific need for numerical values arose, the existence of the programme being advertised meanwhile. I do not believe that either of these suggestions can have more than a limited success. They can be very helpful as far as the world's main centres of computation are concerned but for small organizations and individuals there will be no substitute for a published table.

Alternative Methods of Publication

Of the various possible alternatives to printing by letterpress, the one that appears likely to be most convenient and that can at the same time yield results of good quality is photographic reproduction from mechanically-prepared copy. There seems no reason why tables prepared in this way should fall very far short of the high standard of presentation established in the past by such bodies as the British Association Mathematical Tables

Committee. But really good results can only be achieved by the exercise of great care, and then only with the right sort of equipment.

The copy will be prepared on some form of mechanical typewriter, - I am using "typewriter" here in a very general sense. This typewriter may be linked directly to the computer or may be an independent unit whose operation is controlled by punched cards, magnetic tape or whatever may be the normal form of output of the computer.

The second alternative has been chosen in our Mathematics Division, where we have an IBM electromatic typewriter controlled by a card-reader supplied by British Tabulating Machine Co. The usual objection to the direct linkage of typewriter and computer lies in the relative slowness of the typewriter compared with the speed at which the computer can produce results. But even if a sufficiently speedy typewriter were available, a high standard of presentation can only be achieved by independent operation. A well set-out table cannot be planned in complete detail until the results have been calculated and examined.

In this connexion it is important that the control of the typewriter should be as flexible as possible. The effective presentation of a table depends very much on such typographical details as the ability to suppress leading zeros, or signs, the inclusion of the sign at the head of blocks of five entries and when sign-changes occur and so on. The choice of a suitable form of type is also very important, of course, as is the choice of the paper on which the table is typed.

Apart from the question of speed, photographic reproduction has the advantage of eliminating the necessity for a great deal of proof-reading, hitherto made necessary by the errors occurring during typesetting. Unfortunately proof-reading of the typewritten copy is still necessary as we cannot be certain that the correct figures have always been typed even though the punched cards, for example, are known to be correct. Various checking devices can be built into the control of the typewriter but I do not think that any experienced table-maker would ever be prepared to assume its infallibility. Proof-reading is a very time-consuming operation and this raises seriously the question of whether an automatic proof-reader can be constructed. Such an instrument, if moderately priced, would form part of the equipment of any computing organisation of even moderate size.

Planning the Future Mathematical Tables

The completion of a number of existing projects for mathematical tables will be hastened by the use of electronic computers; the form of such tables is unlikely to be altered appreciably, however. Where electronic computation will have its biggest effect in table-making is in the preparation of very much more extensive tables than has hitherto been possible. In referring to more extensive tables I am thinking of their scope rather than their physical size; it is, indeed, essential that the latter should not be allowed to increase proportionally to the former.

From the human computer's point of view the ideal table gives him precisely the function he wants and is linearly interpolable. It has been quite justifiable to give him such tables in the past, tables usually of elementary functions which could be kept within a reasonable compass. But such a policy cannot be pursued in the future; not even a large organization would have the shelf space to store, or the money to buy, the volumes that would ensue.

Two methods of restricting the size of tables of higher mathematical functions have already become common practice; these are the use of auxiliary functions and of some form of modified differences. A third method which is more recent is the use of auxiliary functions of the argument as well as the function. As an example of the successful use of all three methods I may mention a table just completed in the Mathematics Division of NPL from which one may obtain 8 or 9 decimal values of any of the Bessel functions J_n, Y_n, I_n and K_n for integer orders of n up to 20 and a range of argument from 20 to infinity, the whole table occupying 20 pages. The user of such a table will have to do a small amount of computation to obtain the values he requires but this inconvenience is completely outweighed by the advantage of having a small compact table readily available for use.

Where we are concerned with the tabulation of a function of one variable, even if it be over an infinite range, for a moderate number of values of any parameters involved, we may expect such methods as these to enable us to present the tables in a reasonably compact form. The satisfactory tabulation of a function of two or more variables, however, presents a problem of a different order of magnitude. No single complete answer to this problem is likely to be found, but recent work by F. W. J. Olver of our Mathematics Division suggests a line of approach that may be very profitable in many cases.

Olver has shown that the Bessel function $J_\nu(\nu z)$ has an extremely powerful asymptotic expansion in descending powers of ν^2. Though the coefficients involved are complicated functions of z they can be tabulated and the series used to compute $J_\nu(\nu z)$. In this way the tabulation of a function of two variables can, for a considerable fraction of their total range, be reduced to the tabulation of a number of functions of one variable only. It is by such methods as this, in which analytical methods are used to break down the tabulation to one of a smaller order of magnitude, that we may hope to produce usable tables of higher mathematical functions.

It is, I think, worth labouring the point that, if it is not to be misapplied, the use of high-speed computers to produce tables of this nature will not lead to a decrease in the necessity for preliminary planning and analytical work. Indeed when we consider the rate at which tables can be computed on these machines it is clear that there will be a demand for a considerable increase in the analytical investigation of the functions' properties. It will not be necessary to discover such erudite methods of avoiding computational labour as in the past, but this gain will be more than offset by the difficulty of working out really compact methods of presentation.

B. THE USE OF MATHEMATICAL TABLES IN A HIGH-SPEED DIGITAL COMPUTER

Introduction

In the first section of this paper, I referred to four broad classes of mathematical tables. Tables of each of these four types are needed for use in computations carried out in high-speed digital computers. The use of elementary functions or of the simple physical functions in this way generally presents no great difficulty. They are usually easily calculable, a simple programme for their computation can be incorporated within the main programme, and in the case of elementary functions this will usually be a standard subroutine. When electronic digital computers were first being developed, there was a tendency to assume that the use of other forms of mathematical tables would present no greater difficulty; the assumption was that the table could always be stored in the machine, either in its main high-speed store or on punched cards, magnetic tape or whatever the auxiliary store might be. In practice the problem cannot be dismissed quite so summarily.

The high-speed store of a machine is usually of a very limited capacity. In all but the simplest of problems it is all too rapidly filled up with instructions and subroutines for carrying out the computation. Though an intermediate store offers rather more scope and though it or the high-speed store may well include subroutines for the determination of elementary functions, a programmer would be unwilling and usually unable to include a complicated programme for the determination of a higher mathematical function. Thus the table would have to be stored in the auxiliary store and one is immediately brought up against the problem of the time of search in a table stored in this way. This is particularly true for an extensive table so that, even though the auxiliary store may be of unlimited size, a considerable advantage can be gained by making the table more compact.

If one regards a programme for the computation of a function as being a particular form of table of that function then it will be seen that the main problem before us is the production of the most compact table possible for use in the machine. I now consider some ways in which this may be attempted.

Elementary Functions

Though I have said that the use of elementary functions and simple physical functions presents no real difficulty, space in the high-speed store is so valuable that there is every incentive to produce the most compact possible programme even for these functions.

Many functions are conveniently calculated from series expansions; this is particularly so if several such functions are required in one problem when the subroutine for summing a series is common to all computations. The effectiveness of such a method depends on the number of terms that have to be retained to obtain any required degree of precision and this may be drastically reduced by the use of Chebyshev polynomials. A series of such polynomials provides the lowest degree polynomial that approximates to a given function with a certain precision over a prescribed range of the argument x. For use in the machine, the series may then be rearranged as a polynomial in x. A collection of such approximations is being prepared in the Mathematics Division by C. W. Clenshaw. Typical examples of the saving achieved in this way are given in Table I. It is worth noting that the use of Chebyshev polynomials is closely allied to the use of modified differences referred to earlier in this paper under "Planning the Future Mathematical Tables".

TABLE I

Function	Range	Accuracy	Number of terms in	
			Taylor expansion	Chebyshev expansion
e^x	0 to 1	10D	14	9
cos x	$-\frac{\pi}{2}$ to $\frac{\pi}{2}$	10D	8	7
$\sin^{-1} x$	$-\frac{\pi}{\sqrt{2}}$ to $\frac{\pi}{\sqrt{2}}$	10D	25	10
ln(1+x)	0 to 1	10D	10^{10}	14

A closely allied method of producing more compact representations of functions is the Rational Approximation method of C. R. Hastings Jnr. of the Rand Corporation. His method appears to be somewhat empirical. Where he produces polynomial representations these strongly resemble or are identical with those obtained using Chebyshev polynomials. However, as the name implies, his method consists in general of producing rational approximations to functions and so is more general than a method which can only produce polynomial approximations. Some of the formulae he obtains are remarkably compact and his results are likely to be of considerable value to digital computers. Unfortunately such accounts as I have read of Hastings' methods suggest that they owe a great deal to his particular flair for the work. A systematic mathematical attack on these lines would be of great interest.

Higher Mathematical Functions

Both the methods just referred to can be used in a similar way for some higher mathematical functions. When this is possible a simple programme can be obtained for the computation of the function and, once again, the most convenient "table" of the function is this programme. But it cannot be expected that this will always be the case, particularly when the function contains variable parameters or has two or more independent variables. In such cases Olver's method, referred to in an earlier section, should again be helpful. Even if it is used, however, it appears very unlikely that it would be possible to avoid storing some form of table in the machine.

As I have said the main requirement for any table stored in the machine is that it should be compact. We can, therefore, expect that the methods mentioned in the section on "Planning Future Mathematical Tables", as enabling tables to be considerably reduced in size, will again be useful. Indeed such methods as these may be used to an even greater extent in an electronic machine, as the added labour of computing the function from a highly condensed table is of less importance than when the work is undertaken by a human computer.

There is a distinction between the human and the mechanised computer that should be made at this point. With the human computer a table can very largely be judged on its own merits; for example the necessity of having a table of Everett coefficients for use with a table giving high-order differences for interpolation is not regarded as weighing heavily against the convenience of such a table. With an electronic machine such considerations may be of great importance. Since the programme for many problems contains a subroutine for summing power series, it may well be that the most convenient table for machine use will give coefficients in such series rather than the more usual modified differences.

Physical Tables

I will conclude with a few remarks concerning the use of tables derived from experimental work. Again, wherever possible, the table will not be stored but will be fitted by, for example, a sequence of polynomials whose coefficients are stored. It is very important indeed that the table should not be endowed with a spurious degree of precision, particularly when this seriously complicates the problem of fitting it satisfactorily with simple functions.

In some cases a strong distinction is made between real accuracy and nominal accuracy. Though the experimental work on which a table is based may be of very limited precision, the table may have been deliberately smoothed to a much greater precision in order to obtain greater consistency between results of computations based on its use. In such a case it may be advantageous to go right back to the experimental results in order to fit them by simple functions which may nevertheless be designed to give the required degree of nominal accuracy.

Discussion

MR. SADLER, (H.M. Nautical Almanac Office) said that Dr. Goodwin had listed three stages in the making of tables - planning, calculation and production. The importance of good presentation of printed tables should also be stressed. To obtain the best presentation, tables could not be produced directly from an automatic calculator.

Thirty years ago mathematical tables and particularly tables of logarithms were used as instruments of calculation as well as books of reference. In the future only casual users would need printed tables. Systematic work on automatic calculating machines would use tables punched on cards or from the machine store. To cater for these casual users, tables must be produced at a minimum cost with the maximum amount of data per page. They could with advantage be reproduced photographically but good legibility must be preserved. Proof reading would always present a problem, for even if the print unit could be made completely reliable photographic copying is never perfect. An efficient automatic proof reading machine would be a great boon.

DR. BLANCH (National Bureau of Standards) said that high-speed automatic calculators could produce vast numbers of tables, but that publication would be difficult. She raised the question as to whether, according to the theory of the "Survival of the Fittest", the bigger or the better tables would survive.

Table making has tended to be treated as a background problem for some high-speed machines, and for this purpose tables must be uniform and therefore often cumbersome.

The preparation of planned tables of special interest to physicists or mathematicians has never been profitable. However, if their requirements could be specified and programmes produced, it should be possible in the immediate future to provide by high-speed machines a cheap service of this kind.

Much could be done to produce tables of five, six or even eight significant figures by methods such as those of C. R. Hastings.

The users of electronic machines tend to be over-enthusiastic and rather inexperienced computers. It is most essential to be able to state the accuracy of the final results, and to obtain a knowledge of the properties of a function before computation.

MR. OLVER (NPL), said that in the planning of future tables standard interpolation facilities are desirable. In addition to the complication of changing variables etc., in existing compact tables there is a choice of interpolation facilities. One can use modified differences, reduced derivatives or Lagrangian formulae. Chebyshev polynomials might provide a better method for interpolation. It would be necessary to tabulate functional values, together with the coefficients of the interpolation polynomial of lowest degree.

Such tables would have the advantages that no tables of interpolation coefficients are required, a larger interval of tabulation is possible than when using differences, and derivatives can be calculated fairly easily from the same tables. Preliminary tests have indicated that this method of interpolation is quicker than standard methods.

MR. SADLER pointed out that such tables do exist on punched cards, but for use in interpolation the functional values of a printed table would have to be faked.

MR. OLVER agreed that faked functional values of the form $f(x) + \delta^5/61{,}440$ could be used, but said that it is possible to tabulate true functional values by using a modified polynomial with only slight loss in accuracy.

MR. ROUTLEDGE (RAE), stressed the importance of considering the errors arising from the use of Chebyshev polynomials. Any $f(x) = \sum_0^\infty a_n T_n(x)$, truncated to N terms, is not the polynomial which deviates least from f(x) over the range (-1, 1) although it is a good fit: the theorem is strictly true only when f(x) is a polynomial of degree $N + 1$. Chebyshev polynomials are Fourier series squashed into polynomials, and curtailed Fourier series are not necessarily the best fit to a series of functional values.

Considerable errors arise in derivatives calculated from Chebyshev polynomials. The differentiated series based on the polynomial of five terms may produce a derivative with an error five times as great as that of the functional values. The Taylor series on the other hand can be differentiated without loss of accuracy.

Mr. Routledge derived an expression for the remainder in terms of functional values, from the contour integral formula quoted earlier by Dr. Turing for the remainder term in a finite difference expansion.

DR. MILLER (Cambridge University Mathematical Laboratory), compared the errors arising in interpolated functional and derivative values produced by the use of n terms of a Taylor series and by a Chebyshev polynomial. Errors in the derivative are not so serious as Mr. Routledge had suggested. The gain in accuracy when using n terms of a Chebyshev polynomial in preference to n terms of a Taylor series to calculate functional values is 2^n, while the loss in the derivative is only n.

Interpolation facilities are an essential part of a table, but users require as small a table as possible. A table with modified second differences is better than a linear table, and the use of Miller's "modified Everett Chebyshev formula" reduces the number of tabulations required for a linear table by a factor of approximately fifty.

22. Applications of Electronic Machines in Pure Mathematics

by

J. C. P. Miller

Cambridge University Mathematical Laboratory

Many pure mathematicians can see no way in which their studies can be assisted by electronic computers. There are, nevertheless, many processes which, though not completely numerical, can be standardized and codified. Some of these can involve long and tedious work, and it may be possible to use a machine.

In addition to such applications there are cases where numerical tables can shed new light on certain questions of interest, or where they may suggest new problems for study.

I shall consider under the following headings a few cases known to myself or which have interested me. I am conscious that I have probably missed many other applications.

ALGEBRA
1. Elementary
2. Group-theory
3. Matrices

TABULATION
NUMBER-THEORY
1. Diophantine Problems
2. Factorization and Listing primes
3. Other problems

ALGEBRA

Elementary

Many investigations involve algebraic or similar processes. It may, for instance, be useful to manipulate a power series. It is relatively simple to code up multiplication, division and powering (integral or fractional) of a power series. It also is easy to deal with differentiation or integration term by term.

Such processes have been considerably developed and used by theoretical chemists under Dr. S. F. Boys in Cambridge; in particular, J. L. Turner has made EDSAC routines for applying recurrence relations to or differentiating series of the form

$$\Sigma e^{-\alpha x} y^{\beta} z^{\gamma} u^{\delta} \quad \text{or} \quad \Sigma \mathrm{Ei}(-\alpha x) y^{\beta} z^{\gamma} u^{\delta}, \text{ and so on.}$$

Turner's routines involve obtaining new coefficients from old ones, read in on a tape, then combining and re-arranging the terms, and presenting them to the user on an output tape, ready to be printed or used again.

Group-Theory

(a) C. B. Haselgrove, of Kings College, Cambridge has investigated routines which, from a small number of defining relations for the generators of a group, seeks out all possible further relations, useful in reducing a given combination of multiplications to its simplest form. These relations can then be used, by means of another routine, to determine all the elements of the group. This pioneer programme is not fast, nor can it deal with large groups, but it demonstrates the possibility of using a machine for such problems.

(b) Haselgrove has made up another programme which also starts from a number of defining relations connecting the generators of the group. It makes a list, however, of elements of the group and the product of each by each generator. Elements are numbered in succession and coincidences eliminated. In this way the order of the group is determined, and a 'linear' multiplication table formed. This is much more rapid than the earlier programme - 30 to 50 times faster. It can deal with very large groups by means of sub-groups of manageable order. The same treatment is then applied to a sub-group and its co-sets; the order of a large group may thus be found as the product of manageable factors.

The investigation of group-theory is still at a very early stage of development.

Matrices

The use of machines in connexion with matrices - the evaluation of determinants, the inversion of a matrix, the determination of latent roots, etc. comes rather in the realm of applied mathematics. Nevertheless, the need to tackle these problems on electronic machines has stimulated a very great deal of research and development of methods; one may mention particularly the work of C. Lanczos at the Institute of Numerical Analysis at Los Angeles, on his method of minimized iterations.

TABULATION

Several tabulations have been undertaken in attempts to prove, disprove or throw light on various mathematical conjectures. In any case, I maintain that a full understanding of any mathematical function is incomplete without at least a skeleton tabulation, which may well bring out qualities that might otherwise be overlooked.

Other tables have been made purely from interest (a very proper pure mathematical outlook!) and, when made, have suggested new properties or relationships for further study.

C. B. Haselgrove has tabulated the Riemann Zeta-function $\zeta(\frac{1}{2} + it)$ and $\zeta(1 + it)$ for $t = 0(0.1)100$, in connexion with an attempt to disprove Polya's conjecture that $\Lambda(n) = \sum_{1}^{n} \lambda(r)$ never becomes positive for $r > 1$. Here $\lambda(r) = (-1)^k$, where k is the number of prime factors of r, repetitions being counted. Thus $\lambda(1) = 1$, $\lambda(5) = -1$, $\lambda(15) = +1$, $\lambda(120) = -1$, etc. This also allows study of the zeros of $\zeta(\frac{1}{2} + it)$, famous in connexion with the Riemann hypothesis, that all complex zeros of $\zeta(s)$ lie on the line $s = \frac{1}{2} + it$.

Other investigations by A. M. Turing, Manchester, and by J. B. Rosser and D. H. Lehmer in Los Angeles have been more concerned with the mere location of many zeros, rather than with accurate determination of a few early ones.

Another table, produced on hand machines rather than electronically, will illustrate the point that new investigations may be suggested. This was computed by D. F. Ferguson of Manchester, and gives coefficients in powers $[F(x)]^k$ of the product

$$F(x) = (1 - x)(1 - x^2)(1 - x^3) \ldots (1 - x^r) \ldots$$

The power series for k = 1 and k = 3 are very simple

$$F(x) = 1 - x - x^2 + x^5 + x^7 - x^{12} - x^{15} + x^{22} + x^{26}$$

(with differences 1, $\frac{2}{2}$, 3, $\frac{4}{2}$, 5, $\frac{6}{2}$, etc. between successive exponents)

$$[F(x)]^3 = 1 - 3x + 5x^3 - 7x^6 + 9x^{10} \ldots.$$

(the exponents being triangular numbers). The power k = 24 gives Ramanujan's well-known function $\tau(n)$. Other cases where k is a submultiple of 24 have also been investigated by L. J. Mordell, R. A. Rankin and others. Ferguson's tables, which extend to 300 coefficients for each k = 1(1)24, and some coefficients for k = 26, 28, 30, have lead to interesting new relations involving k = 5, 7 and 11, and have demonstrated many interesting cases with zero coefficients for k = 4, 6, 8, 14, 22, 26. In particular, for k = 15, just one zero coefficient - that of x^{53} - has been found; this fact was found of great interest by D. H. Lehmer.

Examples could be multiplied. But this must suffice.

NUMBER THEORY

Diophantine Problems

A problem suggested by Prof. L. J. Mordell for investigation by EDSAC will indicate the usefulness of machines here.

The relation $a^3 + b^3 + c^3 = 3$, has two known solutions (1, 1, 1) (−5, 4, 4).
(a) Are there others? (b) Is there a parametric form?

There is a parametric form for $a^3 + b^3 + c^3 = 1$, namely

$$a = 9t^4 \qquad b = 3t - 9t^4 \qquad c = 1 - 9t^3$$

There is also a form for $a^3 + b^3 + c^3 = 2$, namely

$$a = 6t^3 + 1 \qquad b = 1 - 6t^3 \qquad c = -6t^2$$

(c) Are there other cases where a parametric form is possible (not counting obviously related forms for, *e.g.* $k = 8$, etc.)? (d) Are there other solutions for $k = 1$ and $k = 2$ which are not given by the formula?

There were very few known sets of solutions for other values of k - just a few isolated ones with small (a, b, c).

A search for solutions with $k \leqslant 100$ has been made on EDSAC by M. F. C. Woollett and myself, resulting in some 433 solutions, nearly all new and the majority primitive, which is probably complete to $a = 3000$. (a) (b) (c) (d) $k = 2$ remain unanswered, *i.e.* none found.
(d) $k = 1$, $k = 16$, yes.

Although the original questions have not been answered, yet the large body of solutions obtained gives more material for study, and raises new questions. Why are there so many solutions (17) for $k = 83$? Why are there so many "blanks" among values of k of form $9n \pm 3$? (No solutions have been found for k = 30, 33, 39, 42). Will the values k = 240, 264, 312, 336 - 8 times as large - be equally barren?

A further search is to be made for values of a, b, c (+ or −) each of the form $3\lambda + 1$. This reduces the trials in the ratio $\frac{2}{27}$ and still gives all solutions in the interesting case $k = 9n \pm 3$. The value of k permitted will be raised to 2000 or more.

Other like problems have been suggested by Prof. Mordell, for example, a search for solutions of

$$x^2 + y^2 + z^2 + 2xyz = 49$$

other than the obvious (7, 0, 0). If there is one more, then there is an infinity of others; the problem is to find just one.

Factorization and listing of primes

Problems of this nature have interested many machine users. This is probably because they are simple to understand, and because many of the methods are quite straightforward. The interest continues, but now it is more concentrated on the attempt to use more sophisticated and highly developed methods to see how they work with very large numbers.

Factorization routines have been constructed by several machines; all work by testing a series of divisors that includes all primes up to the square root of the number tested. Some very fast routines have resulted, which can factorize numbers of order 10^{12} or larger in a matter of minutes.

One interesting new feature has resulted from the mechanization of the process; this is a method for developing remainders after division of N by successive odd numbers, in the form of a recurrence relation, apparently unsuspected before its development by G. G. Alway.

Factorization routines may be used for listing least factors of a succession of numbers, or for listing primes. It is however, much better to design fresh programmes giving a sieve (as originally done by Eratosthenes) and to cast out in succession multiples of 2, 3, 5, 7, 11, etc. up to $\sqrt{N}$, N being the largest number tested. It does not seem very suitable to make factor tables in this way, since the listing of least factors requires so much output, and the saving over hand methods is less pronounced than with many problems.

However, the listing of primes can be done economically so far as output is concerned, and a succession of improved programmes has been prepared for EDSAC, and a prime-sieve programme for the Pilot ACE. The ACE Pilot Model programme yields sieves, on cards, for a million numbers between 10×10^6 and 20×10^6 in about 3½ hours. The latest EDSAC programme lists primes in a million in the same range in about 13 hours, in the form of 5 leading digits and 10 sets of 3 final digits per line of the printed page. Another EDSAC tape gives differences Δp between successive primes in a simple coded form on teleprinter tape at the rate of 125000 range per hour. With Pilot ACE results used for checking, nearly 3 millions have been covered in this way, largely as a background job for EDSAC.

Refinements of the programme are possible and it is hoped to improve speed still further.

Other Problems

(a) *Primitive roots.* A. E. Western is interested in finding primes with very large least primitive roots, *e.g.* one is known with 73 as least. Machines can help in this search. John Todd has prepared a programme on SEAC and dealt with primes to 50 000.

(b) *Quadratic partitions.* Several problems make use of partitions $kp = a^2 + Db^2$ for various fixed D and special small k (*e.g.* 1, 2 or 4) depending on D, where p is any prime in certain arithmetical progressions (about half of all primes for each D). Also of interest is the least solution of the congruence $kp = n^2 + D$. EDSAC programmes have been prepared for the cases $D = 1$ and $D = 7$. Solution with $D = 1$, p near 150000, takes about 1½ seconds.

(c) *Tests for primality.* Fermat's theorem has been used for testing primality of numbers of the form $kp + 1$, and $kp^2 + 1$ with $p = 2^{127} - 1$; the converse of Fermat's theorem is true for $k < p$ in these cases. In this way, primes were found on EDSAC, larger than $2^{127} - 1$, proved prime by Lucas about 77 years ago and for 75 years the largest known prime. In fact $180p^2 + 1$ held the record for some months.

Later Lehmer used Lucas's test on Mersenne numbers of form $2^p - 1$, and on SWAC on January 30th 1952, raised the record again to $2^{521} - 1$ and $2^{607} - 1$. Since then, still using SWAC, the largest known prime has been successively $2^{1279} - 1$, then $2^{2203} - 1$, and now stands at $2^{2281} - 1$ (*ref. 1*)

To summarize, electronic machines are playing a part in pure mathematics by suggesting new problems with the help of tables prepared by the machines, by carrying out masses of computational, algebraic or other processes which can be codified, and by enabling processes that have been developed to be used and tested to a greater extent than was previously possibl

References

1. Two New Mersenne Primes. *Math. Tab., Wash.* 1953, **7**, 72.

Discussion

DR. TURING (Manchester University) said it was really quite difficult to find useful things to do on a machine, partly because in pure mathematics there was no urgency about getting a solution, and partly because some of the fun of it was lost when a machine was used. He suggested two problems of interest:

1. To find the Betti numbers and torsion coefficients of a topological manifold. For example, into how many separate spaces does it break up? The manifold could be described as a totality of simplexes (such as triangles) under prescribed incidence relations. Topologists have a definite procedure for finding these numbers but it is so tedious that it has hardly ever been tried.

2. To tabulate the Riemann-Zeta function with a view to possible disproof of the Riemann hypothesis that all zeros of the function have real part $\frac{1}{2}$. The labour of calculating even a single value of the function is such that one should calculate it at only a few points near each zero expected, and then interpolate. Titchmarsh has verified the hypothesis up to $t \leqslant 1468$, where t is the imaginary part of zero. The range from $t = 24\,000$ to 25 000 has been checked on the old Manchester University machine.

DR. TURING stated that he now had a method of investigation suitable for large t, in which the work was so arranged that the calculation of one value leads quickly to the results at all other points in a chosen range. For $10^{11} < t < 10^{11} + 100$ one might use this method and expect to find about 1000 zeros.

DR. WIJNGAARDEN (Mathematisch Centrum Amsterdam) suggested that the assertion that a computing machine cannot help pure mathematics must be made with caution. For example if one found a zero of the Riemann-Zeta function off the line $s = \frac{1}{2}$, or found an exception to Fermat's last Theorem that $a^n + b^n = c^n$ is not possible for integers if $n > 2$, one would certainly have contributed to pure mathematics, and would in fact upset several theories which had assumed the truth of these hypotheses.

The real fun of mathematics lies in making statements that one can prove oneself and that others can prove to their satisfaction. If a fairly large number is stated to have a certain factor this can at once be verified. But no such test is available for the statement that $2^{2281} - 1$ is prime, which to this extent is not a proper contribution to mathematics, unless someone has a theorem influenced by this result.

He went on to describe a geometrical representation of a combinatorial problem which had been suggested by a study of switching circuits. The vertices of a hypercube in four dimensions may be represented as four-digit words formed by their coordinates, taking one vertex at the origin and side of unit length. A set of such words represents a set of these vertices which will define a polytope, either interior to or on a face of the hypercube according as the set of words have not or have a digit "in common". For example
0000 represents an equilateral tetrahedron within the hypercube, but 0000, in which all words
0011 0011
0101 0101
1001 0110
have the same first digit zero, represents an equilateral tetrahedron in one face of the hypercube.

MR. ROUTLEDGE (RAE) told the story of the argument between Hilbert and Brouwer early this century concerning existence proofs; Hilbert had said he would produce a rule of thumb procedure for establishing the truth or falsehood of any statement in any branch of mathematics. Turing had proved in his paper "on Computable Numbers" that no such mechanical procedure exists, and that there exist numbers which no machine could ever calculate. The logic of propositions however admitted of such a procedure, as did also elementary School Algebra and Geometry. For the general calculus of properties and relations a decision - method was not possible, but became possible if properties only and not relations were used.

MR. ROUTLEDGE said he had programmed for the ACE Pilot Model a decision method for the propositional calculus based upon the Disjunctive Normal Form rather than Truth Tables. This was entirely non-arithmetical. Each 32-digit word represented five six-digit symbols of a statement, and 100 symbols accommodated in a long tank formed the largest expression with which he could deal.

23. The Application of Automatic Computing Machines to Statistics

by

K. D. Tocher

Imperial College, London

Introduction

The work of statisticians covers a very wide range and in each of the various branches a great deal of tedious work can be involved. It has been suggested that automatic computing machines could be used to advantage to eliminate much of this drudgery and it is the intention of this paper to make a brief survey of the various branches of statistics and the value of automatic machines to the work of these branches.

Without claiming that the list is exhaustive, the more important branches of statistical work can be enumerated as

1. Census work.
2. Sample surveys.
3. The Analysis of Designed Experiments.
4. The Analysis of Data.
5. Quality Control and Sampling Inspection.
6. The Design of Experiments.
7. The Calculation of Sampling Distributions.
8. Sampling Experiments.

Census Work

Most mathematical statisticians, and quite a few economic ones too, would deny that census enumeration was a part of statistics and would dismiss it as mere counting.

Notwithstanding this, the popular conception of statistics is so intimately connected with census returns that some remarks on this activity are not out of place. In spite of the simplicity of the task undertaken, of counting and recording the number of items in a set of mutually exclusive classes, the volume of data involved is so enormous that problems of organization become very serious.

I should like to stress that I do not think that our present level of development of machines is sufficient to undertake a complete census enumeration. The number storage involved becomes far greater than anything available yet.

Consider the National Census involving some 6×10^7 individuals. If the details of that could be recorded on a single Hollerith card, then using a machine with a single Hollerith input the passage of the data into the machine would take about 8 months! Since it would take 50 girls about that time to punch the cards there would be little point in speeding this process up. However it is clear that the data can only be used in this form once or twice if present day rate of progress is to be exceeded. This difficulty arises because present methods allow whole batteries of machines to perform parts of the job simultaneously. If the work is to be done automatically it must all be done by one machine.

The coding used in punching the cards will be one convenient for the punch operators and is hardly likely to be suitable for use inside the machine, where a binary classification scheme is clearly most suitable. The translation from one code to another might be done by the machine and this would only take a little longer than the 8 months of input time.

Whatever standards of reliability we finally achieve with these machines, it will be an optimist who thinks his machine will be faultless for 8 months continuous operation. Faultless operation is necessary, for granted the value of a complete enumeration accurate at one past instant of time (a value most statisticians would strongly deny) then complete accuracy is essential.

The machine can be regarded as on production - the production of translations - and as with all production the quality of the work will vary. To give a check on the machine I suggest that a system of sampling inspection of the translations should be instituted. The translation of items, chosen at random by the machine and constituting some small proportion of the total, should be printed out and compared with a manual translation. If faults are found, the machine must be overhauled and recent production repeated.

Such a checking system would probably be supplemented by a daily check on the machine's functioning using test programmes, but to the layman the correctness of every result tested is far more convincing than any other test.

The machine would almost certainly be unable to hold all the translations within itself and even if this were possible it would be undesirable since a power failure or other major fault could lead to the complete loss of the results. A system of punching the translations on cards or tape would be the safest process. This could be transferred by a set of machines to magnetic tape for faster refeeding to the machine for the census proper.

The machine code would consist of a binary number which gives the class to which the individual belongs. To illustrate the further processes, consider a simple example where the data recorded for each individual consists of (i) its sex (ii) its age (iii) its annual income (iv) its marital status. Then the number of classes possible inside each heading is

(i) 2 male or female.
(ii) 108 age to nearest year.
(iii) 13 income groups would be sufficient.
(iv) 7 single, married, widowed, divorced n times $n = 1, 2, 3, 4.$

Thus the total number of binary digits required to describe an individual is $1 + 7 + 4 + 3 = 15$. There would be an average of $\frac{6 \times 10^7}{2^{15}} \simeq 2^{11}$ individuals in each group. Suppose we allow 20 times that number for the maximum. Experience in previous censuses could fix a more realistic safety factor. Then 15 binary digits are required to store the possible number of individuals in any classification.

Whatever the actual subdivision of the storage into numbers may be it is always possible to programme the machine to use it as if it were composed of 15 digit stores. We should require about 30 000 such stores and far more for any real census problem.

As each piece of data is read in, it (or a part of it) is read as the name of a store and 1 is added to that store. When all the data has been classified, the numbers in the stores must be combined together suitably to translate back into the normal classification and the numbers printed out.

The theory of information indicates that it will be impossible to compress the data into a smaller number of bits than it takes to write out the details of each individual. The method of translation used is chosen so that the work is most easily organized within the machine. Until a machine of sufficient size exists nothing much can be done and when it does the method of coding suggested above leads to a very simple organization of the problem.

There is no doubt that, one day, it will be possible to build a machine with a sufficiently enormous store but I have every hope that statisticians will be able to convince those concerned of the superiority of sample censuses before that day arrives.

Sample Censuses and Surveys

The results of a complete census can be obtained with sufficient reliability from a properly chosen sample of the population. Similarly if information is required concerning the distribution of some quantitatively measured quality in a population a survey of the population based on samples is all that is required and has distinct advantages in time and cost over a complete survey. It is also possible to estimate the error of the sample which gives an objective measure of its reliability - a thing completely missing in a complete census, which obtains the true population state for a single instant of time.

Such surveys are not restricted to human populations but are extensively used in agriculture to give estimates and forecasts of crops of various kinds.

The work of a sample census or survey breaks into 4 distinct phases:

1. The survey is planned. This settles (a) the methods to be used to select the sample;

   (b) the procedure to be adopted to collect the sample (and counter measures if these procedures fail, *e.g.* what to do if the caller in a house survey can obtain no answer at a chosen house) and

   (c) the method of analysis to be used.

2. The selection of the sample.

3. The collection of the sample.

4. The analysis of the sample.

Clearly, phases 1 and 2 require no aid from any machine.

The selection of a sample always involves, in any properly conducted survey, some random selection to ensure the validity of the errors determined by the analysis. Although the methods in use at the present time are exceedingly tedious they are rather difficult to organize on a machine. There is little difficulty if a population can be defined precisely by, say, a map reference or a serial number. National Identity Card numbers are not an example of this latter class since certain numbers like AIAU4173/4 do not correspond to any individual (this would be my younger brother if I had one). In most cases the only complete and accurate description of the population is an enumeration and in these cases the translation of this list into a form suitable for the machine to use is more work, even more tedious than that involved in our present manual methods.

The analysis of sample survey data is very similar to that of complete census work, except that the scale of the work is reduced and that counts are replaced by weighted counts and averages. The central difficulty, that the work includes the sorting of a large volume of data, remains.

The details differ to such an extent from one survey to another that it is difficult to believe that a common programme could be evolved which would solve all the organizational problems of different surveys. The labour of constructing these different programmes is another factor mitigating against the use of automatic calculating machines in sample surveys.

However, much survey work today is mechanized with punched card machines, and there is a great future for the development of machines which are faster, more flexible and have a greater internal storage capacity than the present punched card machines, but still remain only partially automatic.

In particular, a machine with a function similar to that of the sorter which rearranges a large quantity of data on, say, a magnetic tape at about 20-50 times the present rate would be an enormous aid in both complete census and sample survey work. With such an adjunct to an automatic computing machine many of the problems discussed above might become soluble.

The Analysis of Designed Experiments

Experiments divide naturally into two classes according to our amount of knowledge concerning the phenomenon under investigation

1. When there is little understanding of the underlying laws, experiments are performed 'to find out what happens'. In such experiments there is very little room for

planning. Several possible outcomes are considered, and arrangements made for observing which occurs. The type of analysis involved is a logical one.

2. When the basic laws of behaviour of the system under study are understood and there is a mathematical theory or a working pragmatic procedure already known. The unknown quantities are merely the values of certain parameters in the mathematical theory. The observations are subject to errors, and the problem involved is one of estimating these parameters with the minimum possible error. In this situation, it is possible to plan the experiment so that the effect of the errors on the optimum estimates is minimized.

The study of methods of estimation is a central problem of statistics and most work has been directed at the case where the unknown parameters enter the problem linearly. Suppose our theory asserts that our observations, if not perturbed by error, would be linear functions with known coefficients of the parameters we wish to estimate. In matrix symbolism, if $\underline{y}$ is a vector of observations, $\underline{\theta}$ the vector of the parameters and the coefficients are given by a matrix $\underline{a}$, then

$$\varepsilon(\underline{y}) = \underline{a}\,\underline{\theta}$$

where $\mathcal{E}(X)$ denotes the expected or mean value of X. The errors associated with $\underline{y}$ are most naturally measured by their variances and correlations. In the most important case, where errors are of equal magnitude and independent, $v(\underline{y}) = I\sigma^2$ where σ^2 is a scale factor.

The estimates of minimum variance (least error) are given by the method of least squares (a theorem due to Gauss). Symbolically

$$\begin{aligned} \hat{\underline{\theta}} &= (\underline{a}'\underline{a})^{-1}\,\underline{a}'\underline{y} \\ V\underline{\theta} &= (\underline{a}'\underline{a})^{-1}\sigma^2 \end{aligned} \qquad (1)$$

Hence the design problem is to choose the coefficients of $\underline{a}$ which control the conditions of the experiment so that the diagonal elements of $(\underline{a}'\underline{a})^{-1}$ are minimized.

One of the most important branches of experimentation amenable to this treatment, known as a linear set-up, is that of agriculture. Here a large tract of land is divided into blocks and then each block into plots. Within each block different treatments are assigned to the different plots and then, if the differences used are confined to those between plots in the same block, fertility differences arising between plots in different blocks are eliminated. Since the natural assumption, that the fertility of neighbouring plots is more nearly equal than that of distant plots, is usually true, this method of analysis gives great accuracy. This analysis assumes that the difference between two treatments in the same block is independent of the block chosen, which implies that the response on a given plot can be written in the form $t_i + b_j$ where t_i corresponds to the treatment applied and b_j to the block in question. If there are t treatments and b blocks we require to find $t_1, t_2, \ldots, t_t$, $b_1, b_2, \ldots, b_b$. These are subject to one linear restraint but we will not emphasise this complication in what follows.

This is a simple example of a linear set-up in which the elements of $\underline{a}$ are either 0 or 1. Thus the analysis to determine the best estimates of $\underline{t}$ and $\underline{b}$ is merely a matter of forming $\underline{a}'\underline{a}$, inverting and applying (1). However, it is not uncommon for t and b to be as large as 100 and then the formation and inversion of a 199 x 199 matrix cannot be undertaken light-heartedly even with automatic computers. A little simple algebraic analysis reduces the inversion necessary to one of a t x t matrix instead of (t+b-1) x (t+b-1).

The problem of the design of block experiments of this kind consists in attempting to minimise the diagonal elements of $(\underline{a}'\underline{a})^{-1}$. Success in this enterprise usually results in $\underline{a}'\underline{a}$ having a comparatively simple form with a small set of, say, p different values arranged systematically and symmetrically.

In a large class of designs, known as partially balanced designs, the inversion of this matrix can be reduced to the solution of p equations for the p values arranged in the same systematic and symmetrical arrangement as in the original matrix ($\underline{a}'\underline{a}$). Typical values of p are 2, 3, or 4 while p = 8 is a rare case.

The wide variety of different experimental arrangements possible leads in current practice to a wide variety of different techniques of analysis. It will be advantageous if it is possible to provide a single programme which will analyse almost all, if not all, block experiments. This comprehensive set of instructions must be designed so that the complications necessary for the more complex designs are by-passed for the simpler designs which do not require them. Unless this is achieved the speed of calculation for the simpler and more commonly occurring designs will be severely reduced, resulting in uneconomical operation of the machine.

A common form of analysis which would form the natural basis for the master programme proceeds as follows. Linear compounds of the observations are formed from the block and treatment totals. Suppose these are called Q_i $i = 1, 2, \ldots, t$. The estimates of $\underline{t}$ are formed as linear compounds of the Q's. The coefficients of these compounds are the p values mentioned above. The linear compounds are formed by sorting the Q's into p sets, accumulating the Q's in each set as they are sorted and then taking a weighted sum of these totals.

The sorting process can be expedited by preparing partition matrices which can be fed into the machine at the same time as the data and can be formed in the first place by the machine using another programme. This programme could be a common one for finding the partition matrix for any experiment.

By a suitable arrangement of the order of input the time of analysis can be reduced to very little more than that for the input. The storage required is only about 2t numbers.

A more complicated analysis, known as an inter-block analysis, is possible and uses the information available for comparisons between blocks and allows for the decreased precision of such comparisons. In this analysis $\underline{a}'\underline{a}$ is replaced by a matrix Ω which depends on the ratio of the error variances within and between blocks. This has to be estimated from the data. The construction and inversion of Ω can be avoided by an iterative procedure based on the normal analysis. For further details see *ref.* 2.

In some experiments the various treatments consist of combinations of various factors at different levels and then a so-called factorial analysis of the treatment effects is required. This consists of making a special set of comparisons of the treatment constants estimated from the experiment and identifying these with a factor or the interaction of pairs of factors, and so on.

This may be performed best on an automatic calculator as a simple analysis for block effect carried out not only on the original observations but on linear compounds formed from observations in the same block.

Thus as an extremely simple example consider four treatments 0, 1, 2, 3, formed as the combination of 2 factors each at two levels with each block containing each treatment exactly once. Writing t_0, t_1, t_2, t_3 as the responses from the four treatments in a typical block the quantities A, B, C, D are formed

$$\begin{aligned} A &= t_0 + t_1 + t_2 + t_3 \\ B &= t_0 - t_1 + t_2 - t_3 &&= A - 2(t_1 + t_3) \\ C &= t_0 + t_1 - t_2 - t_3 &&= A - 2(t_2 + t_3) \\ D &= t_0 - t_1 - t_2 + t_3 &&= A - 2(t_1 + t_2) \end{aligned} \quad \ldots\ldots(2)$$

An analysis of A gives a test if there is a block effect, one of B if there is an average (or main) effect of one factor (averaged over the two levels of the other factor). C gives a similar test for the other factor averaged over the two levels of the first one, while D gives a test of the effect of one factor changes as the level of the second factor is altered - in the statistician's terminology a test for interaction.

If the names 0, 1, 2, 3 of the treatments are written in binary form 00, 01, 10, 11, the first digit gives the level of one factor, while the second digit gives that of the second factor. Then using the identities on the right hand sides of (2) the required compounds can be built up from the partial sums (i) t_1+t_3, (ii) t_2+t_3 and (iii) t_1+t_2. The terms included in these sums are found as (i) those with 1 in the least digit of their binary name (ii) those with 1 in the leading digit (iii) those whose digits sum to 1.

This method of selecting the terms can be generalized to cases with more factors and to cases with more than two levels. More details can be found in Kempthorne (*ref.* 1).

If the observations for a plot are stored with the store number corresponding to the treatment name then simple modular arithmetic on the names in turn can decide if the term should be added or not into a particular compound. Alternatively, the names of the admissible treatments can be generated successively.

The main application of these treatments is to agricultural problems when the effort of analysis by present day methods is still only a small fraction of the total effort in performing the experiment. The time scale is so long that rapid analysis is not necessary. However, in industrial application the time and effort to perform the experiment is much less, and rapid analysis might be essential for the effective use of these experiments. These might then form the basis of routine industrial quality control schemes.

Among the most important experimental arrangements which are not linear set-ups are those known as quantal response experiments. The result of a test on some object may be positive or negative; there are no shades of response. Of a group of objects, some will react positively and the rest negatively and we can regard a randomly chosen object as having a probability P of reacting positively. Now suppose that the test depends on some parameter θ so that as θ is varied P will vary. In many cases the relation between P and θ can be expressed as

$$P = \int_{-\infty}^{\theta} \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-(x-\mu)^2/2\sigma^2\right\} dx = \int_{-\infty}^{\frac{\theta-\mu}{\sigma}} \frac{1}{\sqrt{2\pi}} \exp\left(-\tfrac{1}{2}x^2\right) dx \qquad \text{.........(3)}$$
$$= \int_{-\infty}^{\frac{\theta-\mu}{\sigma}} Z \; dx$$

A typical experiment consists of testing a set of n objects at a level θ_1, another set of n_2 objects at a level θ_2.... and finally a set of n_m objects at a level θ_m. In these sets r_1, r_2.... r_m give positive results. The problem is to estimate μ and σ of equation (3).

A well-known estimation procedure with several desirable features is the method of maximum likelihood and an application of this gives the estimates as the solution of the equations

$$\sum_{r=1}^{m} \left(\frac{r_i}{P_i} - \frac{n_i-r_i}{1-P_i}\right) Z_i = 0$$
$$\sum_{r=1}^{m} \theta_i \left(\frac{r_i}{P_i} - \frac{n_i-r_i}{1-P_i}\right) Z_i = 0 \qquad \text{.........(4)}$$

Present day procedure consists of an iteration, each step of which consists of a weighted regression involving weights $\frac{n_1 Z_1^2}{P_1(1-P_1)}$ ($= \frac{nZ^2}{PQ}$). Tables of $\frac{Z}{P}$, $\frac{Z^2}{PQ}$ have been constructed and are used in this technique.

For mechanization of this process a method of evaluating $\frac{Z}{P}$ is required. This is given by the continued fraction extension due to Laplace

$$1 - P = Z\left(\frac{1}{x+}\;\frac{1}{x+}\;\frac{2}{x+}\;\cdots\cdots\right) \qquad (5)$$

This does not converge well near the origin and a modification by Thiele's method would be used in practice to improve the convergence in that range. The iterative procedure now used could be abandoned and the equations (4) solved by bivariate inverse interpolation.

These analyses are performed in great quantities throughout the country, but at any establishment only a few are required. The organization between the consumer establishments and a machine would be quite difficult. A more practical solution might be for experiments to be standardized (this is so to a large extent already) with fixed values for m and n, and equal intervals of θ say, and to use a machine to construct tables which give the solutions for all possible results. Such tables would be very bulky but various symmetries could be utilised to reduce their size.

Certain other laws relating P to the parameter θ are sometimes used and there is a certain controversy about the best laws to use. (This is an example of an experimental situation of type (b) with a working pragmatic procedure known, rather than a real theory). An extensive programme of numerical work is necessary to throw light on this problem and a machine could be of great assistance in such a project.

One of the main applications of this quantal response analysis is in insecticide research. In some complicated situations, the simple model outlined above is unsatisfactory and more complex models have been studied theoretically. Methods of analysis still need developing, for these models are considerably hampered by the volume of computing which may be eased by automatic machines.

Analysis of Data

Most of a statistician's work is concerned with the analysis of results collected from various sources and not from planned experiments.

Much work of this kind falls into a branch of statistical analysis known as multivariate analysis. The basic assumption made is that groups of n observations can be regarded as having a joint n-dimensional normal distribution. Such a distribution is completely specified by a set of n means and an n x n symmetrical variance matrix. The object of the analyses is to determine these parameters and to test for certain relations between the parameters of different distributions.

The first step in all such analyses is to form a sample variance matrix. If a typical n-dimensional observation is $y_{1r}, y_{2r}, \ldots\ldots, y_{nr}$ then the matrix required is $\left[\sum_{r=1}^{N} y_{ir}\, y_{jr}\right]$ where N is the total number of observations. The building up of the scalar products simultaneously is quite easily effected providing there are at least $\frac{n(n+1)}{2}$ stores available for numbers.

The general difficulties of scalar products do not arise in this case since the observations only have 3 significant decimal digits at most and scale factors can be adjusted so that products are accumulated at the back end of the arithmetic unit, eliminating problems of loss of accuracy due to cancellation. The contribution to each scalar product from an observation can be added at the same stage in the calculation, removing the need to store the whole of the data.

The further stages of the analyses involve constructing other matrices from this matrix and then either solving a set of simultaneous equations or finding the latent roots and vectors of a matrix. All these are standard numerical problems and the origin of the matrices usually ensures that the various forms of ill-conditioning sometimes encountered will be absent.

Other fruitful sources of heavy computing in statistical work are periodogram and correlogram construction in the analysis of time series. This is also very suitable for mechanization.

The theory of time series has attracted considerable attention of late years and there is a growing literature on the analysis of models and on the prediction of the consequences of these models. Much less work has been done on methods of analysis of actual time series to distinguish between alternative models. Such work as has been done has suffered from lack of adequate numerical verification. This is another research project which could benefit from the use of automatic computing machines.

The remaining sources of numerical work in statistical analysis spread over a wide range, but the outstanding omissions from the discussion so far are ranking methods and the calculation of χ^2 goodness of fit criteria. This work appears to be rather difficult to organize on automatic computing machines.

Sampling Inspection

One of the chief advances in British production technique during the last war was the wholesale introduction and acceptance of sampling inspection and quality control. The methods used then and now are extraordinarily simple but very effective. If the methods could be more complicated, then further increases in efficiency might arise.

The basic assumption of sampling inspection is that when production is under control a certain small proportion p of the product will fail some relevant test. If the quality deteriorates this proportion will rise. Batches of product are assumed homogeneous, a random sample is taken and tested. The proportion p is estimated and, if it is reasonably established that the true p for the batch is higher than is tolerable, then the whole batch is rejected or tested in its entirety to eliminate defective items (rectifying schemes). The test must be a function of the number of items tested and the number found defective. Some schemes of inspection fix the number tested per batch beforehand (fixed sample schemes) while others have rules of selection which allow the sample size to increase in the more doubtful cases (sequential sampling). If at any stage of the sampling the number of items tested is n with r defective and n-r effective this can be plotted on a sampling inspection diagram or lattice as the point (r, n-r). The succession of points form a path and sampling ceases when this path meets a boundary on the lattice. This boundary is in two parts, an acceptance boundary and a rejection boundary. By varying the boundary different tests are derived. Most of the boundaries in use are simple ones for which the properties of the corresponding tests can be derived theoretically. Except for these simple boundaries the calculation of the performance of a test can only be discovered by numerical analysis. This is exceptionally simple and could be organized on an automatic computing machine. A programme of research on these lines is being considered at Imperial College.

In certain cases the assumption of homogeneity within each batch is justified, *e.g.* when the batch is a natural unit of production. In other cases the production is continuous and then the batching is arbitrary and breakdown in quality can occur in the middle of a batch, introducing heterogeneity. The random selection within the batch re-establishes homogeneity but disturbs the assumed binomial distribution and reduces the power of the test procedure.

Theories of sampling for continuous production are now being developed and it is likely that many numerical problems will arise. When these have been solved it is almost certain that practical schemes for sampling continuous production will be more complex than those used in batch production and then machines might be useful in the factory itself.

In quality control qualitative tests are replaced by measurements and then the continuous production does not lead to quite the same difficulties as sampling inspection. The problems of quality control are similar to those of sampling inspection and the following remarks will apply equally to both forms of production control.

Once the idea of using the machine to perform the necessary calculations involved in the execution of the tests is admitted, much more complicated sampling schemes can be envisaged. A product can be sample inspected at several points of its production process and the final decision made on the totality of the test results. Present day practice regards each stage of production as unrelated to all others and so loses power.

It will be possible to replace the present-day two sentence schemes (accept or reject) by many-sentence schemes. As a simple example suppose the production is batched and each batch is sampled at two points of the production line A and B, then the sentence at A can be (i) reject batch (ii) accept batch but stiffen sampling test at B (iii) accept batch (iv) accept batch and relax test at B. It might also be possible to arrange that significant discrepancies in the tests at A and B could alter the future sentencing at A *i.e.* that the scheme may be made to adjust itself to the actual conditions of production.

Clearly to justify an automatic machine in a factory all the inspection results of the factory must be processed by it and it only needs a little imagination to see that special input and output devices attached to the machine could be arranged so that it chooses its own samples, collects the results of the tests and indicates the final sentences all automatically. These sentences could be framed in terms of corrective action to the machines as well as a sentence on the product. This gives a form of automatic factory control.

It might be argued that it would be better to arrange that the machine can send instructions to take the corrective action directly and so eliminate the waste almost entirely. This is more difficult than is usually realized. The results of tests, especially quantitative tests as used in quality control, are themselves subject to error. In a sampling scheme these errors are averaged out and their effect allowed for, and have a very limited effect on the final result. If machines were being constantly adjusted because of errors in the tests, then instability of the machine processes could easily be set up with disastrous results. If an averaging scheme were used to prevent this instability, then real changes of quality could occur without immediate corrective action and so production would still be wasted. If the errors of tests were eliminated the extra cost of providing and maintaining the additional expensive apparatus might exceed the saving on waste production.

The speculation in the last paragraphs may prove to be false but I think it is certain that in one way or another automatic computing machines will in time play a revolutionary role in our factories.

The Design of Experiments

We have seen that block experiments can be arranged so that the amount of work involved in their analysis is substantially reduced. The conditions necessary for the reduction are known, but all possible experiments meeting these conditions have yet to be discovered and present day methods of discovering them are tedious.

The conditions give rise to a series of diophantine equations whose solutions generate a class containing all the possible solutions to the arrangement problem. This class must be further reduced by testing if the members of that class also satisfy a combinatorial requirement. There are very few methods known as yet of solving these problems and resort has to be made to trial and error methods which are extremely slow.

This raises the question of using machines for solving combinatorial problems and shows that this is a branch of pure mathematics which will have useful applications. At the moment there does not seem to be an adequate arithmetical theory for this subject. Macmahan's work when tried in practice is rather disappointing. For example, the function he proposes for enumerating the number of Latin Squares consists of a sum of terms each 1 or 0 according as a given square corresponding to the term is Latin or not. This amounts to the trial and error method of trying each possible square in turn. It is extremely dangerous to assume that the great speed of these machines will enable such crude methods to work. For the enumeration of the 8 x 8 Latin Squares, by testing every 8 x 8 square consisting of

8 replicates of 8 distinct symbols, if the machine enumerates test squares at the rate of 10 a second, the enumeration will take 10^{40} centuries.

A possible starting point for a numerico-analytic theory of combinations is the work on balanced incomplete block designs of W. S. Conner who uses the properties of Hessé and other invariants of algebraic quadratic forms.

The theories of design relating to factorial experiments involve Abelian group theory, and have now been reduced to routine enumeration which could undoubtedly be performed on automatic machines.

Problems of rank correlation and other non-parametric tests can also lead to problems in diophantine analysis and problems of enumeration.

The Calculation of Sampling Distributions

Given a set of random variables denoted by $\underline{x}$ and their joint differential distribution $p(\underline{x})d\underline{x}$, a function y of $\underline{x}$ has a cumulative distribution which is defined by

$$F(y) = \int_{x \varepsilon R_y} p(\underline{x})d\underline{x}$$

where R_y is a region of the n-dimensional space $\{\underline{x}\}$ dependent on the value of y. If $\underline{x}$ denotes a set of observations (usually independent so that $p(x) = \pi_1\ p(x_1)$) and y is an estimate of some parameter of the distribution, then F(y) is the cumulative sampling distribution of this estimate.

In many cases the dimensionality of the problem can be reduced by analysis to 1 or 2. Often the function F(y) can then be determined by

(a) the solution of a differential equation, ordinary or partial.

(b) a recurrence relation

or (c) an expansion (i) power series,

(ii) asymptotic series,

(iii) continued fraction,

(iv) in terms of some complete set of orthogonal functions, *e.g.* Fourier expansion.

Thus the evaluation of sampling distributions involves problems which are perfectly familiar in numerical analysis. The fact that all the quantities dealt with are probabilities and therefore non-negative often simplifies the investigation of errors. Many of these functions behave like $\int_{-\infty}^{x} e^{-\frac{1}{2}x^2}dx$ for large x, and the extremely slow growth of this function is sometimes a nuisance but can usually be turned to advantage.

For some of the less frequently used functions the construction of tables is adequate, but for certain functions (already extensively tabulated) the need is for methods of calculating spot values needed in other more advanced distributional theories. The most important functions of this type are the χ^2, t, Z and F distributions used in the analysis of variance. Since this aspect of the use of machines has been dealt with by E. T. Goodwin in another paper of this Symposium, nothing further need be said here.

The theory of stochastic processes and time series also gives rise to similar distributional problems and these often involve the solution of integral equations. The existence of yet another field of application of these may stimulate some worker in numerical analysis to consider this rather neglected branch of his art.

Sampling Experiments

Certain sampling distributions defy reduction by analysis to one of the forms considered in the last section. In that case the multiple integral may be evaluated numerically, but if the boundary of R_y is at all complicated (as it will be in these cases) the problems of error determination are exceedingly difficult and it is difficult to programme the calculation for an automatic machine.

A well-established technique used by statisticians in this case is that of a sampling experiment. A model of the parent population is made and samples are taken from it repeatedly. From these samples a histogram or sample frequency curve is constructed and sampling repeated until the chance fluctuations have been averaged out to acceptable limits.

Performed manually this is extremely slow and is only used when all else fails. The rapid calculating rate of automatic machines widens the possibility of using this method and in certain cases it can be shown that sampling experiments can be more efficient than deterministic methods.

This arises from the fact that the truncation, even in the deterministic method, is fixed by the number of points N which is considered and is proportional to $\sqrt[n]{\frac{1}{N}}$ where n is the dimension of the space, whereas in the sampling method the truncation error can be made negligible by using sufficient figures in the random numbers. The sampling error is proportional to $\frac{1}{\sqrt{N}}$ independent of the dimensionality of the space.

There is considerable controversy concerning the source of random number that should be used. The alternatives seem to be

1. to use sets of previously prepared random numbers (taken from tables such as those of Tippett or Kendall and Babington-Smith).

2. the generation of the numbers by some physical process, *e.g.* the random decay of radio-active material.

3. the generation of pseudo random numbers by some arithmetic process, *e.g.* taking as the $(n+1)^{th}$ number the middle digits of the square of the n^{th} number.

The first of these methods suffers from the disadvantage that the quantity of such numbers now available is limited and the large amount of input will slow the machine down. The second has the disadvantage that the numbers and therefore the calculations are not reproducible. The numbers generated could be recorded but then external or auxiliary storage would be needed for this and the repeat calculations would be of the first type.

The objection to the last alternative is that the numbers are not truly random in the sense of Von Mises, since a rule exists for predicting a member of the series from previous members. Without entering into the philosophical questions concerning the meaning of random, for practical purposes the numbers are required to be generated by a process which has no relation to the purpose for which they are to be used. Hence, if an experimenter can convince himself that he can convince his critics that there is no relation, then this method can be used and has considerable computational advantages.

The next problem is how to convert the random numbers into random variables. All n-decimal numbers constructed as a set of n random numbers are equally likely, and for large n can be regarded as random variables uniformly distributed over the range $[0, 1]$. If a random variable has a distribution $p(x)dx$ then $\int_{-\infty}^{x} p(x)dx$ takes values in the range $[0, 1]$. Solving, for y, the equation

$$F(y) = \int_{-\infty}^{y} p(x)\,dx = Z \quad \ldots\ldots\ldots , \qquad (6)$$

where Z is a random variable uniformly distributed in the range $[0, 1]$, gives y distributed as $p(x)$. This method is completely general, but involves the calculation of the inverse function to $F(y)$ which may be rather difficult.

For certain special variates, other methods can be used. For example, if a normal variate is required the average of n uniform variables is nearly normal for n about 10, and this extravagant use of random numbers will be much quicker than the use of the particular case of equation (6).

Random normal deviates are so much in demand that, if the first alternative for the generation of numbers is used, then these can be normal deviates with advantage. If uniform variables are required these can be found by the formula $U = \left(\sum_{i=1} z_i\right)$ mod. 1. For moderate $n (\geq 10)$ this formula gives sufficiently accurate results.

Sampling experiments are not confined to the solution of sampling distribution problems. Any insoluble problem mathematically formulated involving a stochastic element can be simulated on an automatic machine, and the results from the model used to indicate the behaviour in the real problem. Examples of this technique already used include problems concerning queues of aircraft at airports, and of cars at traffic lights, life and death processes, the storage required in factories, and the optimum renewal procedure for electric lamps.

There is developing a school of thought which advocates solving deterministic problems by these methods. The deterministic problem is replaced by a stochastic one for which some distribution involved can be shown to be related to the solution required. For example, Laplace's equation can be solved by showing that the distributions involved in a suitable random walk are related to the solution of this equation. These methods are too new for any final judgement on their merit to be given.

Summary

This brief survey has served to show that automatic computers can be of considerable value to statisticians. There are, however, certain logically simple problems for which these machines are not very suitable. This should give an incentive to our engineers to build, not bigger and better machines for the same job but different machines for a different job.

References

1. KEMPTHORNE, O. *Biometrika*, 1947, **34**, 255.

2. TOCHER, K. D. *J. roy. stat. Soc.*, 1952, **14B**, 45.

Discussion

MR. STRACHEY (National Research Development Corporation) mentioned some of the difficulties involved in Census calculations. Though there was no multiplication or division, the amount of data and the number of sub-totals required were so great that the process was not economic on these complicated machines. There is much sorting and many card passages are needed and even assuming that 32 000 totals could be formed in each run the time taken on the Manchester computer is about 5000 years - slow, even for the Census!

DR. HARTLEY (University College, London) agreed with this and suggested that for such problems the first essential was high-speed input and output, allied to sorting equipment.

The technique of experimental sampling, using the Monte Carlo method, can be used to obtain a frequency distribution, for example, of the range, that is the difference between the largest and smallest values in random samples of five, say. For this purpose tables of random numbers are required, but if these are generated in the machine there is no input problem, and high-speed computers are then very useful. Mr. Fieller had developed a method which reduced the number of samples required. He computed also a second statistic, the control variable, whose frequency distribution is known exactly and is similar to that of the range. The resulting calculations are arranged in a double-entry frequency table in which the row and column sums give respectively the frequency distributions of the control variable and range in the sample. Adjustment by multiplication to the known distribution of control provides the required range distribution. If the two variables are perfectly related the matrix of entries in the frequency table is diagonal. The accuracy of the result is determined by the magnitude of the off-diagonal elements.

MR. HEALEY (Rothamsted Experimental Station) said that the advent of high-speed computers would have as large an effect in statistics as had that of desk computers. In particular, they will be of great use in the analysis of time series and in problems such as crop forecasting, when answers were required quickly. They have less application in the analysis of experimental data because of the complexity of the assumed model hypothesis. Programming might take longer than desk computation.

DR. TOCHER disagreed with the last point. Mr. Healey's methods for agricultural problems have widespread application, and a large number of the rather less complicated problems can be put in a systematic form and a common programme made for use with high-speed machines.

24. General Discussion on Machine Utilization

opened by

D. H. Sadler,

H.M. Nautical Almanac Office

I think you will agree with me that it is rather inappropriate for me to open the discussion, because I have never used an electronic machine and the subject of this discussion is primarily the utilization of these machines. You may find it interesting that a desk computer should listen to your discussions today and pass some remarks on them. But first of all I should like to say that I am surprised at how much I agree, since I expected to find a lot of points of disagreement. I agree particularly with Prof. Hartree's remarks, both in his opening address and those he has made on machine utilization, and with those which Dr. Fox made. It is very tempting for those computers who have been brought up in the hard school of desk computing to regard it as essential for the computing to have a background of these methods, and while I am beginning to doubt it myself I still think that one of the things that is perhaps lacking in the usage of electronic machines at the present time is a sufficient number of people with adequate background of ordinary computing methods.

I hear (I know nothing about these things) talk about microseconds and machines working at very high speeds; then we have an enormous amount of effort being put into speeding up these machines which are already very fast. To speak of modernizing is to use such words as optimum coding, microprogramming and so on, but I would ask something about how much time is spent on planning the jobs for the machines so that the maximum use is obtained from the results. In statistics you plan experiments, but how many physical problems are planned in this sense, so that the data derived from the solution of your equations on the electronic machines can be interpreted?

I have very little contact with the world of electronic machines but even I know of three problems that have been solved, two in this country and one in America, in which numerical results have been useless simply because two or three hours consideration was not spent in the early stages, thinking what use could be made of the results when they had been obtained.

Therefore I would like to support very strongly the suggestions that have been made earlier that the people programming problems on electronic machines should get right down to the root of the problem with its originators in order to ensure that they are solving the correct problem, and if possible that the solution when obtained will be of direct use. There is another reason. It seems to me that among the methods we have heard there have been very few new computing methods. But of course there are new techniques, techniques which perhaps have been forced upon the users of electronic machines by storage capacity limitations which circumvent, for instance, the use of differences, but which make it more economical of storage capacity to use the Lagrangian type formulae rather than formulae involving simple differences. It seems to me that by going to the root of the problem it should be possible to derive much more direct methods of computation than those which have previously been used on desk machines.

The paper that has been of particular interest to me was that on the commercial application of electronic machines as exemplified by LEO. The particular interest to me is that our production of navigational astronomical almanacs has very much in common with this commercial work in that we have a very large output of relatively small amounts of simple arithmetic. Yet it seems to me that the successful application of electronic machines to commercial problems does open up the possibility of their successful application to problems over a much wider range. I am thinking now about survey problems and similar problems which at first sight do not look as if they are suitable for very fast electronic machines. In a purely astronomical field - and here I am distinguishing between astronomy and astro-physics, for in the astro-physical field you have got the same type of problem as occurs in the ordinary physical problem - we have several problems dealing with celestial mechanics of the solar system which are suitable for solution by electronic machines. In the problem of determining the future of the solar system we shall have something like 30 second-order non-linear differential-equations to solve simultaneously, and since we want the solution not for all time but at least for a very long time, we shall have to carry anything up to 10 or 20 extra decimal digits in order to guard against building-up error, and I think the storage problem will be very considerable.

I must say that I have listened to this discussion today with very great interest, and I feel quite sure that the large number of people who attended this Symposium and the enormous amount of enthusiasm shown will result not only in the application of the old methods of desk computing but will also in a very short time produce new methods or techniques. In conclusion, I think it is a great pity that these electronic machines do not yet seem able to improve on desk computers as far as relaxation methods are concerned.

Discussion

DR. VAJDA (Admiralty Research Laboratory) gave some details of the computations involved in problems of linear programming and the theory of games.

DR. MILLER (Cambridge University Mathematical Laboratory) amplified his earlier remarks about remainder terms in finite-difference formulae. The remainder term gives an upper bound, often much too large, and other curtailing criteria are more practicable. They must, however, be used with care: "stopping at the smallest term" is not satisfactory unless the terms are decreasing rapidly. The theory of the replacement of a function by a polynomial had been treated by Whittaker in his discussion of "cardinal functions".

MR. BENNETT (Ferranti Ltd.) described experiments at Manchester in the solution of differential equations, involving an automatic adjustment of the finite-difference interval. A Milne predictor formula checks that the interval is the right size, the final integrations being performed with a smaller interval.

DR. TURING (Manchester University), reverting to the question of truncation of infinite series, said that it was best to determine the required number of terms by analysis, since rules of thumb could break down in certain circumstances. This happens, for example, in the use of Euler's transformation for accelerating the convergence of infinite series, when the n^{th} term has a pole for a non-integral value of n.

DR. GOODWIN (NPL) supported this comment. In that particular example, the correct result is obtained if the Euler summation is carried out to the end, but this would be done only by an intelligent computer who noticed the peculiarity of the function: a machine, using a standard routine, would probably fail.

In further support of the need for careful planning he quoted some examples of the solution of ordinary differential equations by finite-difference methods. The equation $y' + 20y = 19e^{-x}$, with $y(0) = 1$, has the solution e^{-x} and with some methods can be solved at an interval $\delta x = \frac{1}{2}$. Other methods break down when the complementary function e^{-20x} is so badly represented as to be exponentially increasing. This fact can be used to advantage in solving so-called "stiff" equations, in which the wanted solution is well-behaved but all other solutions increase very rapidly. It may be possible to choose a representation of the equation which gives so bad an approximation to the unwanted solutions that they decrease, instead of increasing, and the wanted solution is obtained quite accurately.

CIRCUITRY AND HARDWARE

Chairman: Mr. E. A. Newman

25. Gates and Trigger Circuits

by

W. W. Chandler

Post Office Research Station, London

Introduction

This is a very large subject - too large to be treated adequately in the time available. It is proposed therefore to give a very brief summary of our findings on MOSAIC and other electronic machines, leaving if possible some time for discussion.

Gate Circuits

The generalized gate circuit is a device which, supplied with signals from "N" sources, will itself provide a signal if "X" or more of the input leads are simultaneously stimulated, where X may have values from 1 to N. If X = 1, the device may be called an "OR" gate; if X = N, it may be called an "AND" gate.

The obvious way to provide a "gate" facility is by a multiple potentiometer system. The drawbacks of such an arrangement are:

1. Reduction of level in output signal relative to input-directly proportional to N.

2. Variation and instability with time, of available resistors.

3. The fact that such an arrangement draws current from the original sources, and thus cannot be repeated often on any one signal bus-bar.

The first of these objections is usually decisive in very high-speed machines, where available voltage swings are in any case low due to necessity for low time-constants.

Improvement may be effected by the use of diodes (thermionic or crystal) provided that the gate is an "OR" or an "AND" gate. Voltage swings are then transmitted with very little reduction, but a load, albeit a small one, is still thrown on the sources of signals. To avoid even this trouble, it is necessary to use some scheme whereby each input lead is terminated on a separate valve, or at least on a separate electrode. Thus there are the techniques using the control and suppressor grids of a pentode (satisfactory if a signal sufficiently large to swing the suppressor is available), feeds on several valve grids, the anodes of the valves being commoned (in which case amplification, phase reversal and change of absolute level results) and feeds on several valves connected as cathode followers with common cathode (in which case level and phase are retained but amplitude may be slightly reduced). In all types of circuit, it is usually necessary and always desirable that the gate shall be either of "AND" or "OR" type, and the circuit should be re-cast to make this so. The point is illustrated by consideration of the MOSAIC Adder circuit below.

An "Adder" circuit is supplied with three signals, the A and B digits (from the two numbers A and B to be added) and the C or carry digit. The output is two-fold, one line providing the required sum A + B and the other deducing the carry digit to be fed back to C for the next A + B digits. The answer "1" is required on the "Sum" line if the sum of A B and C is 1 or 3; the answer "1" is required on the "Carry" line if the sum of A B and C is 2 or 3. All other answers are zero.

The logical circuit shown has the required facilities, and at low speeds is satisfactorily transformed into the networks also shown. It has however one gate circuit of the type to be avoided, namely the circuit provided with 3 inputs, ($N = 3$) any 2 of which acting together are to be effective ($X = 2$). It is desirable to find another circuit which does not rely on such gates.

Such a circuit may be derived (using circuit algebra) as follows:

If $f_1(A,B,C)$ represents line 1 output
and $f_2(A,B,C)$ " " 2 output (carry digit)

then $f_1(A.B.C.) = A.B.C. + (A+B+C).\ f_2\,[(A,B,C)]^1$

and $f_2(A.B.C.) = A.B.C. + A.B.C.^1 + A.B^1.C + A^1.B.C.$

the dashes indicating negation.

$\therefore$ $f_2(A,B,C) = A.B.(C + C^1) + A.C.(B + B^1) + B.C.(A + A^1)$ by adding A.B.C. twice.

$$= A.B. + A.C. + B.C.$$

If, instead of using the variables A and B directly, we first derive the two new variables $A.B. = G$ and $A+B = F$, using simple "AND" and "OR" gates, then:-

$$f_1(A.B.C.) = (A.B.).C + [(A+B)+C].\ [f_2(ABC)]' = G.C + [F+C].\ [f_2(A,B,C)]'$$

and $f_2(A,B,C) = A.B + (A+B).C = G + F.C$

These latter two equations give a logical solution involving only simple "AND" and "OR" gates in F, G and C. The logical diagram and the equivalent circuit diagram are shown, the switching in the latter case being by diodes and cathode followers. This is the MOSAIC adder circuit.

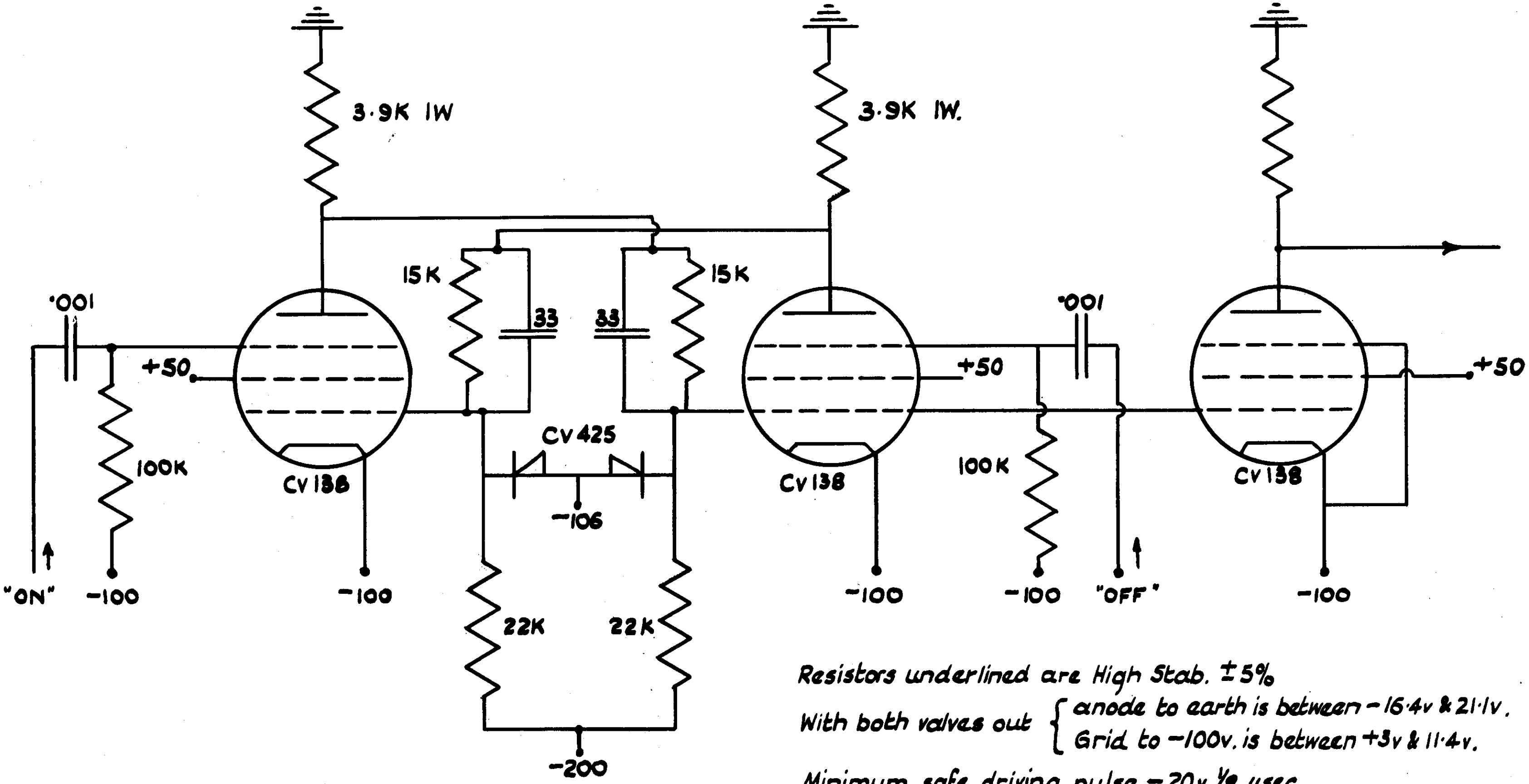
Trigger Circuits

Many types of binary trigger circuit are available, and there is time here only to indicate qualities held to be desirable, which have been incorporated into the standard MOSAIC trigger. In general, the symmetrical type of circuit, in which each anode feeds the opposite grid, the two cross-feed circuits being identical is preferred to other types - for instance, that where feed one way is by anode coupling and the other by cathode coupling - for a variety of reasons. The "ON" and "OFF" signals are of exactly the same kind, the two half-circuits are independent and can therefore be separately tested, the circuits themselves are very easy to design and appreciate, and the input signals can be specified with respect to fixed voltages - not for instance with reference to a cathode which is itself at a varying voltage level. It is admitted however that the above does not cover the subject, and the preference may be merely a personal one. The remaining points however are regarded as fundamental.

1. It is desirable to separate the operating connexions from the internal (or stabilizing) connexions. The obvious solution is to use pentodes, stabilizing on the control grids and switching on the suppressors - an application of the "One electrode-one job" technique already referred to by Dr. Coombs in Paper 5. The highly-favoured double-triode valve is eliminated by this method, it is true, and of course the number of separate valves is increased, but the advantages are felt to justify the increase.

2. The upward swing of the control grids is limited by grid current. If the downward swing is also limited - by diode clamps connected to a voltage just below cut-off - a trigger is obtained which is sensitive to small and short duration input signals, and is furthermore unaffected in speed of operation by wide variation in the resistors constituting the cross-feed networks (provided of course that these networks give stable D.C. conditions). An essential part of the circuit is an additional condenser feed from each anode to the opposite grid giving an A.C. coupling effect. Without the diode

Standard "Mosaic" Trigger Circuit



Resistors underlined are High Stab. ± 5%

With both valves out {*anode to earth is between -16·4v & 21·1v.*
Grid to -100v. is between +3v & 11·4v.

Minimum safe driving pulse -20v 1/8 μsec.

FIG. 1
PAPER 25

clamps, these condensers tend to increase the total resolving time, since the control grids are depressed too far; with the clamps, this effect does not arise. Triggers of this type can be made to respond in a few milli-microseconds.

3. The output should be taken from a buffer valve, so that the output circuit may be designed independently of the internal circuit parameters.

Fig.1 depicts a trigger circuit, designed in accordance with the above principles, which is the standard binary trigger used throughout MOSAIC.

Discussion

MR. R. T. CLAYDEN (English Electric Co. Ltd.) in referring to this paper and to the one presented by Dr. Coombs said that the policy adopted in the design of MOSAIC seemed to be to use a profusion of valves, combined with "one electrode - one job". In MOSAIC some 6000 valves are used whereas, using ACE Pilot Model circuitry, only 2800 would be required. For example, the diagram (*Fig.2*) shows a trigger circuit using fewer electrodes and valves than the MOSAIC trigger. The common cathode coupling is used thus making full use of the cathode. The input is applied to Grid A which is used *only* to drive the trigger. Buffer stages are eliminated and the anode is used only for the output. One D.C. coupling is used as against two in MOSAIC and even this can cause trouble if the resistors change in value. The pentode shown in the output stage of the MOSAIC circuit is not essential. Further the Pilot ACE circuitry does not make use of notoriously unreliable suppressor characteristics. The CVD specification for CV138 allows a very wide tolerance on these.

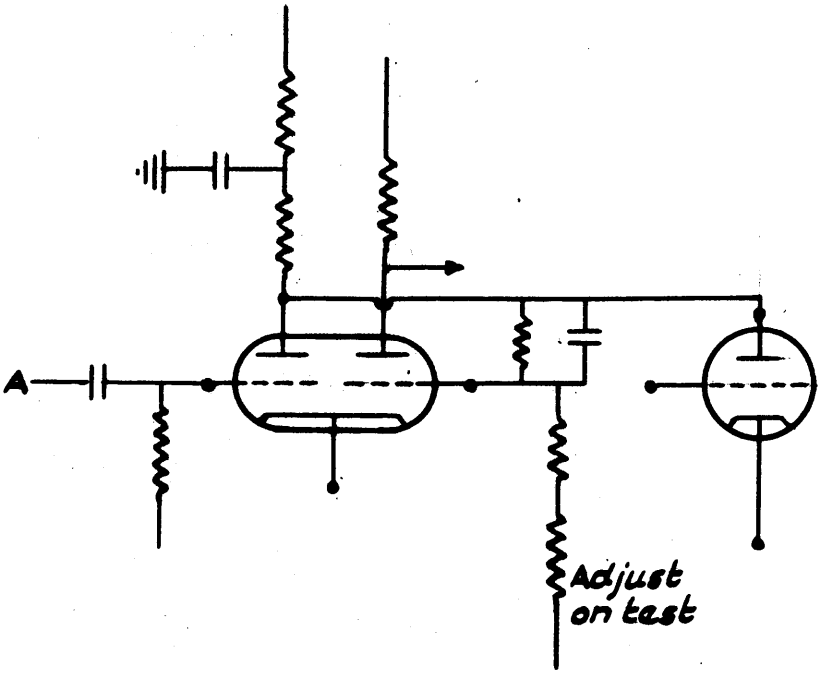


Fig.2

MR. D. O. CLAYDEN (NPL) said that the original design for the ACE Pilot Model specified that the output of a trigger or gate should be known and independent of valve characteristics. *Fig.3* shows a simplified gate circuit on these lines. The voltage across the 10K cathode resistance is large in comparison with the grid bias of any valve used so that for any reasonable grid swing the cathode current is not altered appreciably.

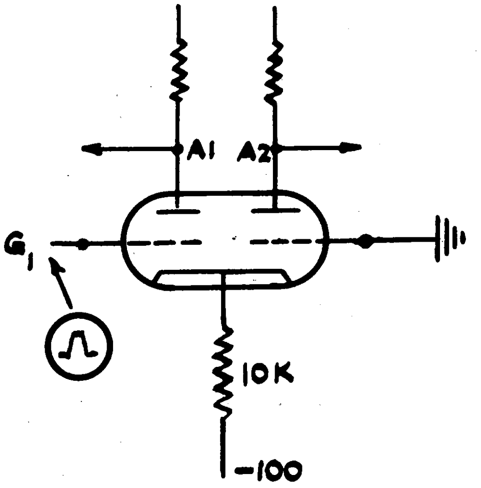


Fig. 3

Double-triodes are used for this circuit throughout. The signal is applied to G1. If G1 is at, say, + 10V all the cathode current passes through A1 and if it is at, say -10V it passes through A2 so that at either anode quite a definite voltage equal to the product of cathode current and anode load is obtained. More elaborate circuits are obtained by connecting valves in cascade as shown in *Fig. 4*.

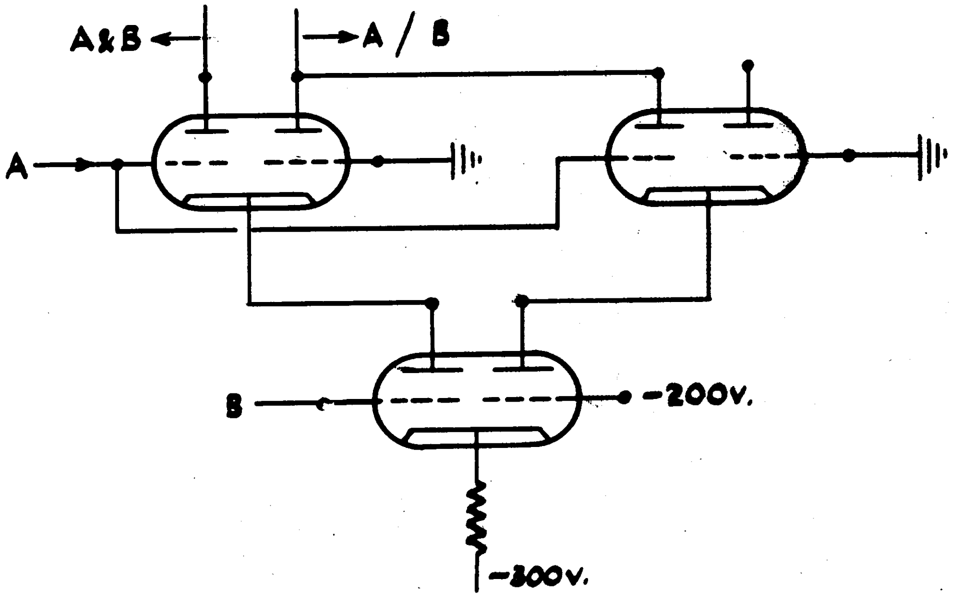


Fig. 4

Out of the four anodes, any combination of voltages corresponding to the input signals A and B may be obtained. Furthermore by connecting the two anodes as shown the logical operations A and B and A ╪ B can be performed.

DR. COOMBS (Post Office Research Station, London) in reply said that the "one electrode one job" policy seemed to have been misunderstood. It means that one electrode should not do *more* than one job and not that every electrode must do *something*.

Pentodes are used as output valves in order to maintain the uniformity of valve type. This entails no sacrifice of efficiency or increase in cost and has the advantage of easier servicing. The Pilot ACE type trigger saves one grid at the expense of troublesome manual adjustments, whereas the double-pentode trigger can be installed as designed without further adjustment. Regarding the use of suppressor grids the valve specification quoted is not relevant to the typical application of CV138 valves in MOSAIC. D.C. switching is never used on suppressors. Triggers are switched by an easily obtained very high A.C. signal on the suppressors and the valves are not necessarily cut off. The circuits are so arranged that only a small voltage movement on the anode is required.

MR. NEWMAN (NPL), from the chair, remarked that in the triggers used in the ACE Pilot Model the tolerance on either grid is about 10 volts. They, also, functioned as designed.

25a. Parallel Ferroresonant Triggers

by

Prof. J. Garcia Santesmases

Consejo Superior de Investigaciones Cientificas, Spain

Introduction

In a circuit made up of a resistor, an air-core inductance and a capacitor in series or parallel, currents are completely determined if the applied voltage is sinusoidal.

If an iron-core inductor is substituted, the aspect of the problem is changed, and the inductance becomes a function of the current. Whenever this element is present in a circuit, an accurate study of the circuit becomes difficult because of the distortion which the core introduces in the voltage or current waveforms. Under certain conditions, two stable states can be obtained. This phenomenon is referred to as ferroresonance. Ferroresonance can occur in the series circuit (different current for the same voltage) or in the parallel circuit (different voltages for the same current).

A systematic investigation of the parallel ferroresonant circuit has been made by the writer *(ref. 1 and 2)*.

At the beginning of last year, while I was in the Computation Laboratory of Harvard University, I started working on a trigger circuit which is based on parallel ferroresonance *(ref. 3, 4, 5)*. Afterwards I continued this work in Madrid University.

Parallel ferroresonant circuit

We shall consider a parallel circuit consisting of a resistance, an iron-core inductor and a condenser *(fig. 1)*. Assume the applied voltage V in the circuit is sinusoidal, and the source has a very low internal impedance. To simplify studying the problem in a general way, we can also assume that the current through the circuit is sinusoidal, and the losses in the iron and the resistance of the coil are negligible.

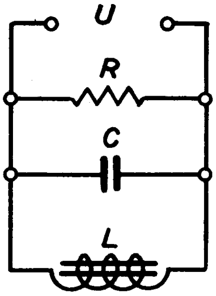


Fig. 1

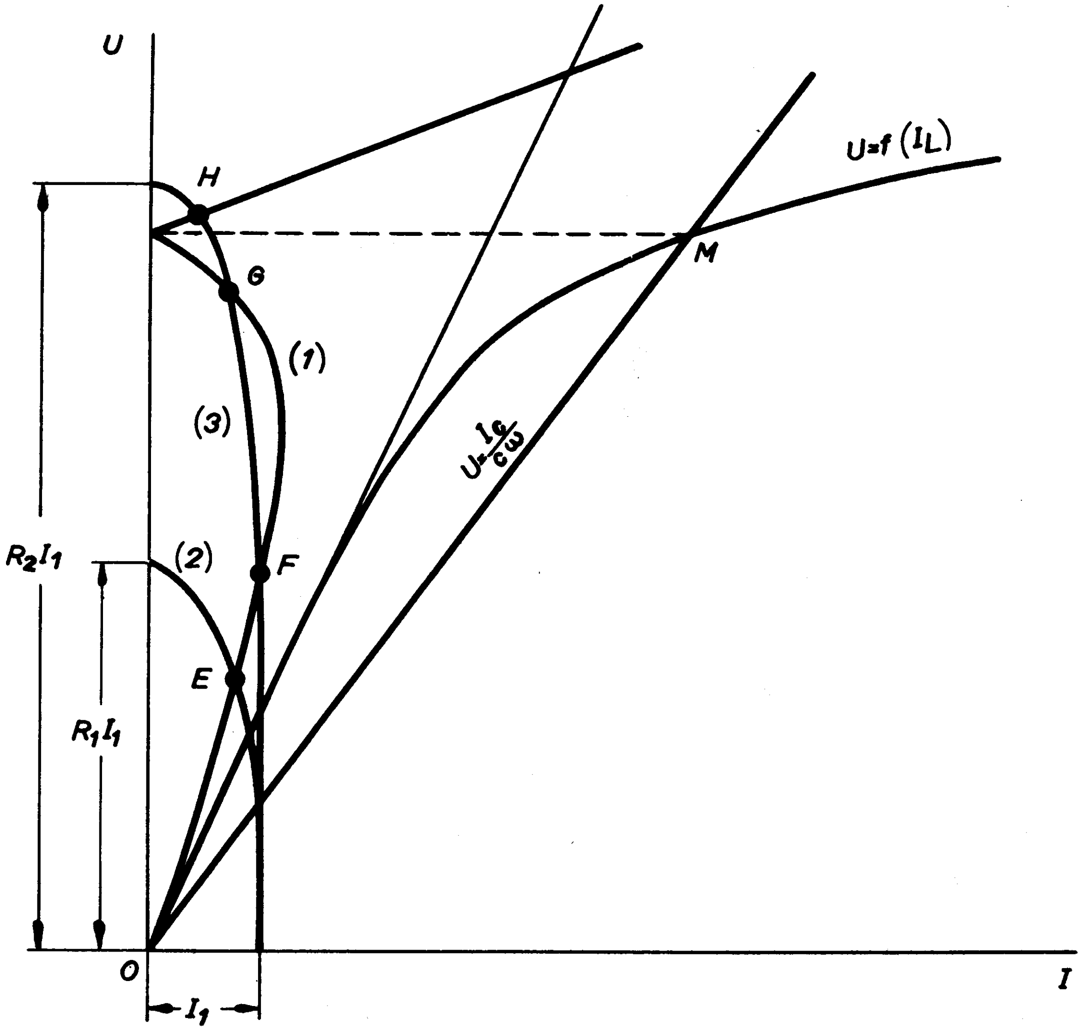


Fig. 2

Under these conditions, if we consider effective values, we can write:

$$I^2 = \left(\frac{V}{R}\right)^2 + (I_L - CV\omega)^2 \qquad (1)$$

Equation (1) can be written as follows:

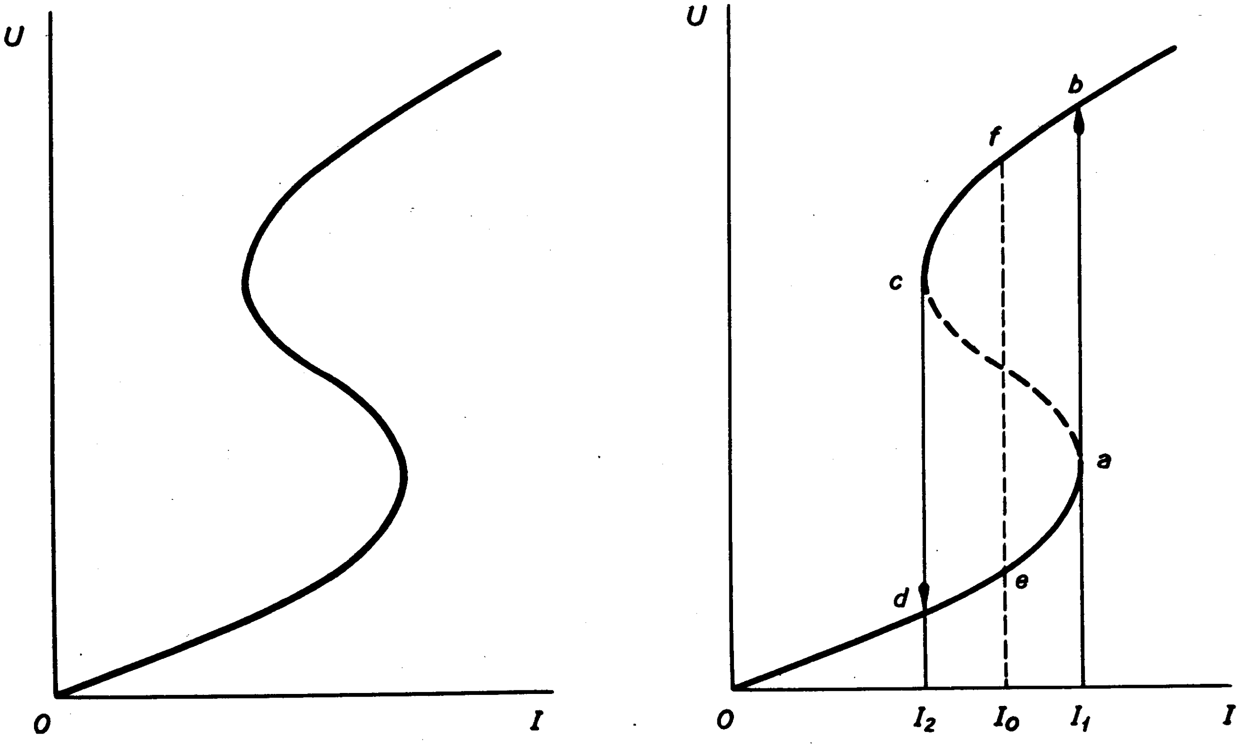
$$I_L - CV\omega = I_0 \qquad (2) \qquad\qquad \frac{I_0^2}{I^2} + \left(\frac{V}{RI}\right)^2 = 1 \qquad (3)$$

where I_0 is the current through the condenser and the coil in parallel. In *fig. 2* the equation (2) is represented by the curve (1) and the equation (3), which is an ellipse, by the curve (2). The points of intersection of these two curves are the operating points of the circuit for the assumed current. In the figure two ellipses have been indicated corresponding to the same value of the current I_1, with two resistances, R_1 and R_2, ($R_1 < R_2$). The influence of the parallel resistance in the behaviour of the circuit may be observed. By drawing the corresponding ellipses for different current values, and the same resistance, the ferroresonant curves may be obtained.

It is possible to obtain the ferroresonant curves in an easier way by using a network of circles instead of a network of ellipses *(ref. 4)*.

Using one of these constructions and assuming the same ideal conditions it is possible to study graphically the behaviour of this circuit *(ref. 2 and 4)*.

In *fig. 3a* the shape of a ferroresonant curve is shown. Assuming a constant voltage source is used, this curve is stable in all its branches.



(a)

CONSTANT VOLTAGE SOURCE

(b)

CONSTANT CURRENT SOURCE

Fig.3

It should be pointed out from the experimental point of view, that the losses on the iron-core have a very great influence on the shape of the ferroresonant curve. As these losses increase, the curve becomes less accentuated until a limit is reached where there is no ferroresonance.

Constant current source. Ferroresonant triggers

If we apply a constant current source to the parallel circuit, the ferroresonant curves are not stable in all branches (*fig. 3b)*. The branch *a c* is unstable.

Let us assume that *e* is the operating point, which corresponds to a current I_0, and that we increase the current. When I_1 is reached, the voltage jumps abruptly to the operating point b. If, now, we decrease the current to the same original value I_0, the voltage will follow the branch *b c* until the point f is reached. Thus, in this operation the voltage has jumped from *e* to *f*. If we perform the same operation but in the contrary direction, decreasing the current first, the operating point will drop from *f* to *e*.

Obviously this ability to change from one stable state to another meets the requirements of a trigger circuit. It is not necessary to use a change in the alternating current for triggering the circuit. We can use current pulses as well. For this purpose, we can assume that in the iron-core inductor direct and alternating magnetizing forces act simultaneously. If, in these conditions, we change the value of the direct magnetizing force, the shape of the ferroresonant curve changes too. A family of ferroresonance curves for different values of the direct current *i* is shown in *fig. 4*. The explanation of this family of curves is based on the fact that incremental permeability decreases when the constant field is increasing.

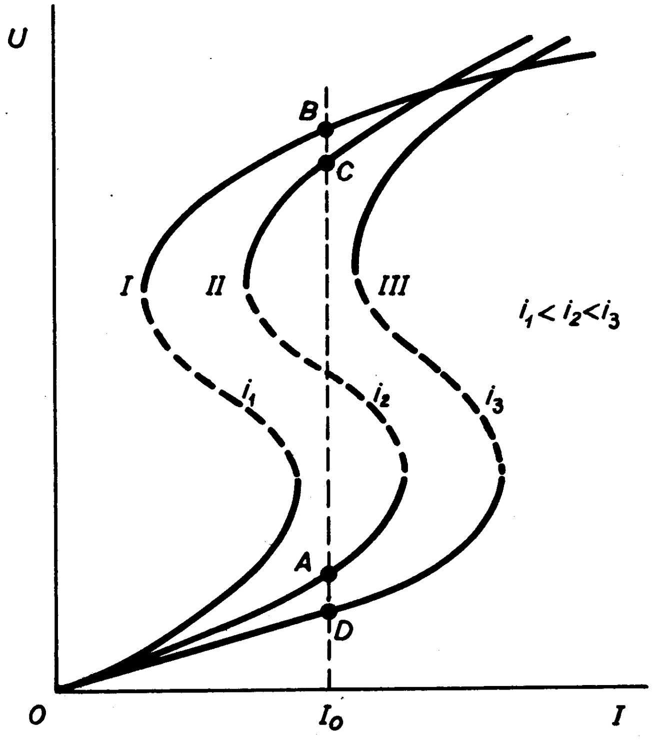


Fig. 4

Assume the alternating current is I_0, the direct current i_2, and the operating point A. If the direct current changes from i_2 to i_1, the ferroresonant curve changes from II to I, and, as the alternating current I_0, remains unchanged, the operating point jumps up to B. If now the current i_1 increases to the initial value i_2, the operating point drops to C. So that a decreasing current pulse i_2-i_1, jumps the operating point from A to C. Similarly an increasing current pulse (i_3-i_2) drops the operating point from C to A.

In fact, instead of using a source of direct current and another of alternating current, our tests have been carried out using a single source, the plate circuit of a pentode *(fig. 5)*.

For a certain bias on the grid, and a certain voltage from the oscillator, the current through the coil should be the superposition of a direct current and an alternating current. By changing the voltage of the oscillator or the grid bias, the alternating or direct current is changed.

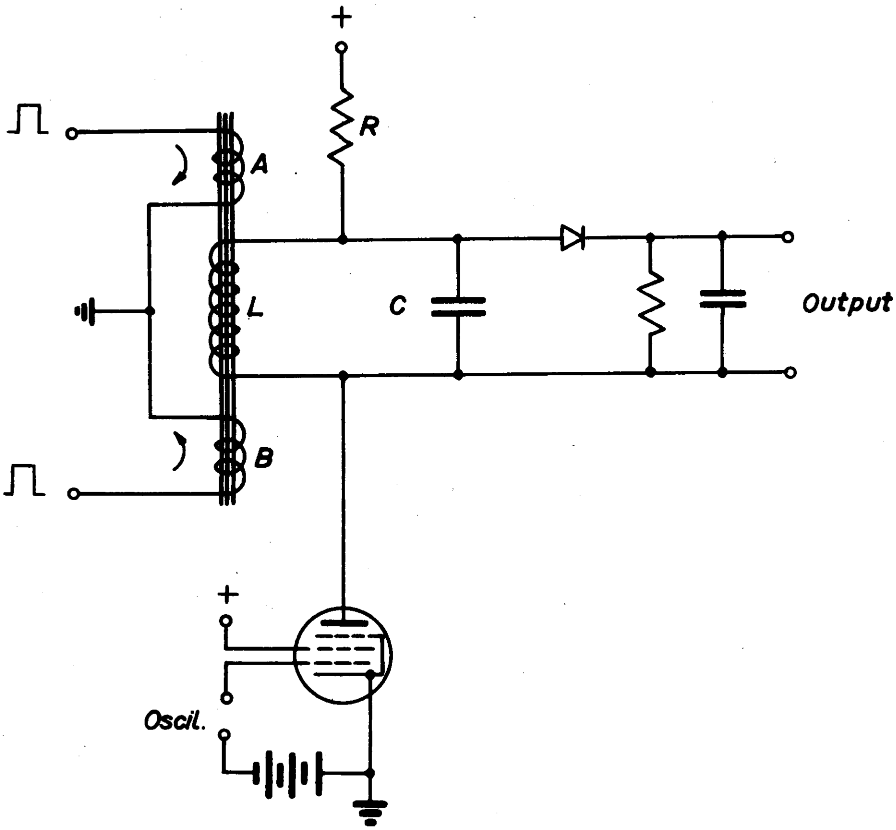


Fig. 5

A similar effect to that of a change in grid bias can be obtained as follows. If we apply two secondary windings to the core of the coil *(fig. 5)* and we send a pulse to the winding A in the direction indicated by the arrow, which decreases the constant flux through the core, the system will change from the low to the high state. Inversely, a pulse in the opposite direction through the coil B will re-establish the low state. The output is obtained after detection of the modulated wave.

In order to get an output not connected to the plate circuit (which may be of interest if it is desired to use various units in series with the object of constituting a counter), the system indicated in *fig. 6* may be employed. This diagram also shows the procedure for obtaining two outputs opposite in phase.

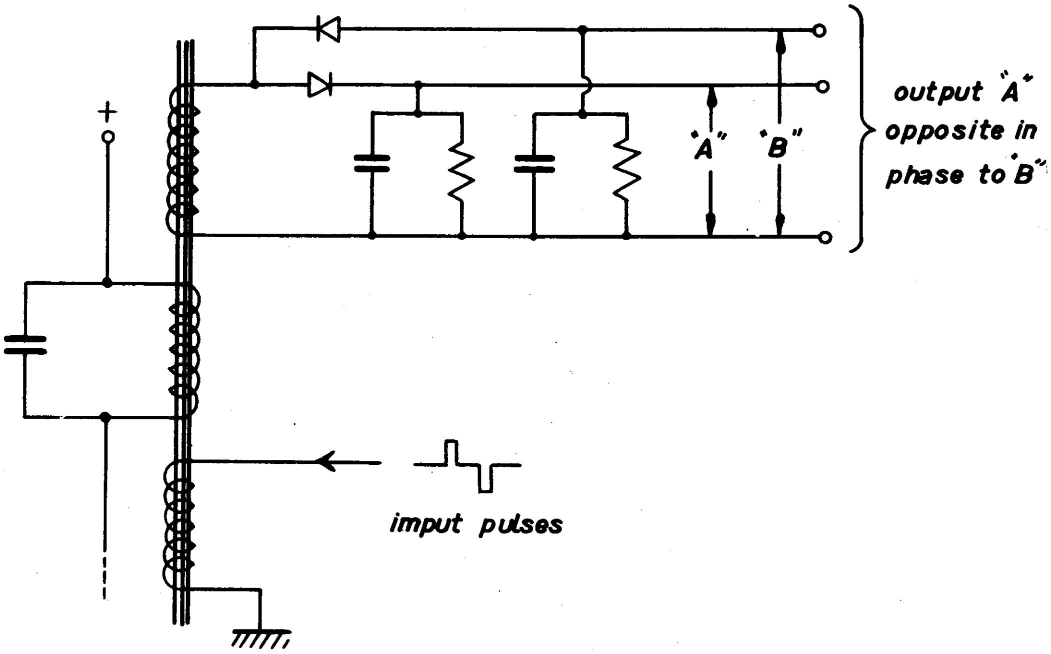


Fig. 6

With these basic principles some trigger circuits have been designed and the experimental tests have been successful.

The first tests *(ref. 4)* were carried out using a core made of Mu-metal. The carrier frequency was 70 kc/s. About 1.8 D.C. ampere-turns per cm, and 1 A.C. ampere-turns per cm. were required. In these conditions the system was triggering with pulses applied successively to the one or the other secondary, with a rate up to 12 kc/s. The ampere-turns per cm corresponding to the current pulses were about 0.6. The pulse width should be equivalent to about two or three waves of the carrier frequency.

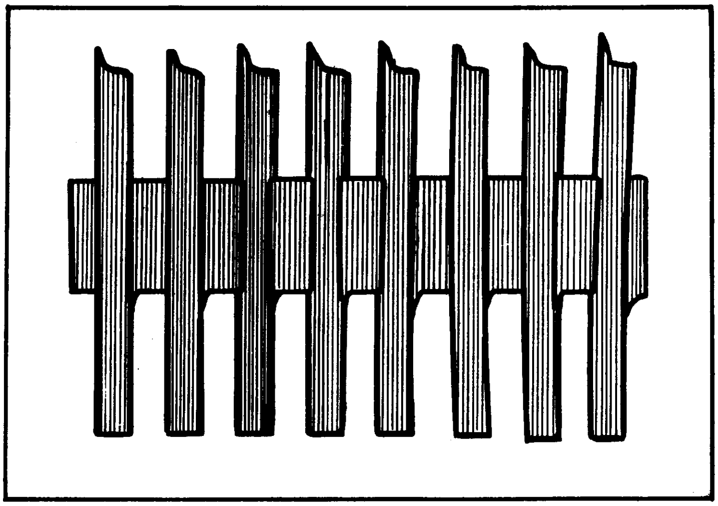
A second series of tests has been carried out at Madrid, using cores made of ferrites. We have increased the carrier frequency up to 1 Mc/s, and the rate of pulses up to 150 kc/s. About 4 D.C. ampere-turns per cm, 1.3 A.C. ampere-turns per cm, and 0.5 ampere-turns per cm of current pulses are used.

A picture of the modulated wave, and the corresponding output after detection, as displayed in the oscilloscope, is shown in *fig.7*. The carrier frequency is 1 Mc/s and the rate of pulses is 20 kc/s. The output and the modulated wave for the same carrier frequency at a rate of pulses of 65 kc/s is shown in *fig.8*. The modulated wave at a rate of pulses of 150 kc/s and for a carrier frequency of 1.5 Mc/s is shown in *fig.9*.

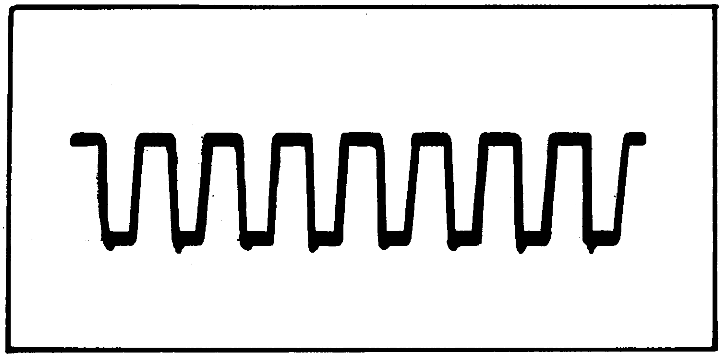
It should be pointed out that this trigger needs no more than one core for performing the same operation as the equivalent vacuum-tube trigger.

The fundamental ideas and experiments of the parallel ferro-resonant triggers have been described. At present, we are making experiments in order to put several units in series (which will give the advantage of using the same current for all of them) and in such a way to make a counter.

I wish to express my gratitude to Professor Howard H. Aiken, the Director of the Harvard Computation Laboratory, and to Sres. Rodríguez Vidal and Sánchez Rodríguez for the help they have given me in the University of Madrid.



(a)



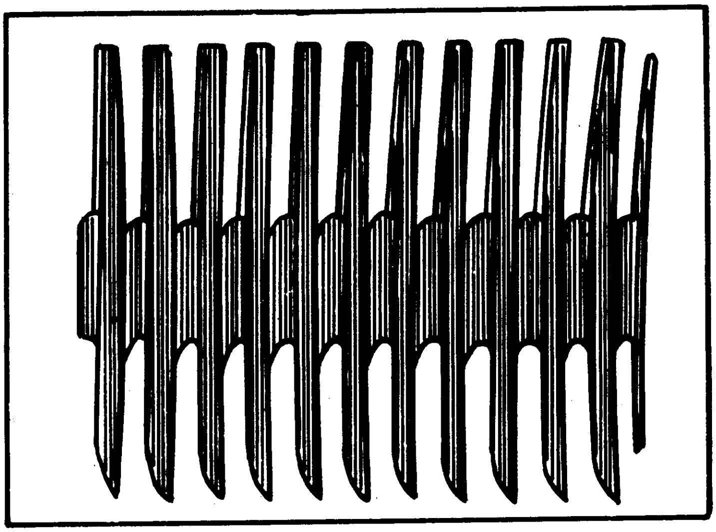
(b)

FIG. 7

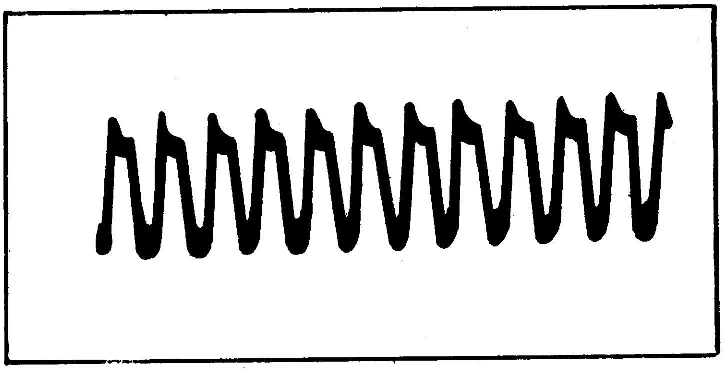
PAPER 25A

(a) Modulated wave (b) Detected wave

Carrier frequency: 1 Mc/s.
Rate of pulses: 20 kc/s.



(a)



(b)

FIG. 8.

PAPER 25a.

(a) Modulated wave

(b) Detected wave

Carrier frequency: 1 Mc/s.
Rate of pulses: 65 kc/s.

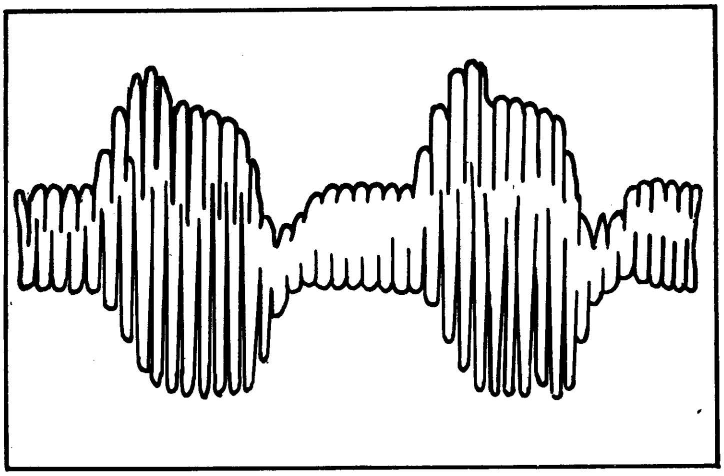


FIG. 9.

Carrier frequency: 1.5 Mc/s.
Rate of pulses: 150 kc/s.

References

1. J. GARCIA SANTESMASES. Sobre ciertos aspectos de la ferro-resonancia de un circuito. *Anales de Física y Química.* 1942, V. XXXVIII, 21.

2. J. GARCIA SANTESMASES. Contribución al estudio de la ferro-resonancia y de la autoinducción. Consejo Superior de Investigaciones Científicas, Madrid, 1943.

3. J. GARCIA SANTESMASES. *Anales de Física y Química.* 1952, V. XLVIII-A, 171.

4. J. GARCIA SANTESMASES. Laboratory Computation. Harvard University Progress Report, 1952, 22.

5. J. GARCIA SANTESMASES. *Anales de Fisica y Quimica,* 1952, V. XLVIII-A, 355.

26. Mercury Delay Line Storage

by

M. Wright

National Physical Laboratory

Introduction

Computing machines using mercury delay line stores have enjoyed considerable success for over five years. The keynote of this success is reliability and since the store is the largest part of any machine the reliability of m.d.l. storage can be taken as established.

A storage system can be assessed in respect of three major features:

1. Cost
2. Reliability
3. Speed and access time

The initial cost of m.d.l. storage is fairly high in comparison with that of some other storage systems but this is relatively unimportant. In order to assess the economics of any storage system a comparison with the cost of maintenance and the cost of the rest of the machine must be made. The initial cost is, therefore, less important than it is sometimes stated to be.

The easiest way of analysing the reliability of any storage system is to consider its possible faults. In general these can be divided into four classes:

1. Complete and continuous breakdown
2. Breakdown for certain pulse patterns only
3. Random pickup
4. Intermittent breakdown of the types (1) or (2)

Mercury delay lines are free from (2) and can be made free from (3). The associated circuitry may of course exhibit any of these faults, but that is true of most of the circuitry of a computer.

It is convenient to base the assessment of m.d.l. storage on our experience of the designs used in the ACE Pilot Model. Many published descriptions of delay lines are available *(ref. 1,2,3)* and the assistance derived from them is fully acknowledged.

The Mechanical Design of the Pilot Model Mercury Delay Lines

Three lengths of line are used, with nominal delays of 32μ s, 64μs and 1024μ s. The two shorter types are similar in design and all have a mechanical length adjustment.

Each short line is built from a straight length of mild steel barrel, ¾ in. bore, 1/8 in. wall, the ends of which are threaded to take the crystal assemblies. The barrel is mounted vertically with a fixed crystal holder at the bottom and an adjustable one at the top. The crystal holders on all the delay lines are similar but the methods of mounting are different.

The bottom holder is fixed onto a steel disk which screws onto the end of the barrel, whereas the top holder is fixed onto a thin tube. A micrometer type adjustment is fitted to this tube, which is free to move only up or down so that the electrode connexion can be taken from the centre via a hole in the side.

The holder contains a perspex insulating disk, ½ in. in diameter, which is clamped in position by a steel ring. A brass electrode, 3/8 in. in diameter, is set in the perspex insulator with its surface recessed 0.001 in. below the rim of the perspex. The crystal, which is ½ in. in diameter, rests on this rim and is retained in that position by a light wire spring. The spring is mounted on four pillars fixed to the steel clamping ring.

The crystal is made of X cut quartz 0.007 in. thick. The surface finish on the face which sits on the perspex rim is matt but the other, which is in contact with mercury, is polished.

Mercury appears to wet the surface of polished quartz so that a reliable and intimate contact is maintained. Originally the crystals used were matt on both faces. At that time the lines were filled first with methylated spirit, which was then displaced by filling with mercury. It is believed that these lines worked by virtue of a thin film of spirit between crystal and mercury which wetted both surfaces. They tended to fail after a few months - possibly due to the gradual disappearance of this film. The lines are now filled directly with clean mercury. There have been no line failures of the type originally experienced since this technique has been used. The method of polishing thin quartz crystals is simple and few breakages occur during the process.

There is an air gap of 0.001 in. between the crystal and the brass electrode. The mechanical impedance of this air gap is effectively zero and no energy is transmitted into it by the transmitting crystal. Most of the energy which appears at the receiving crystal is reflected at this back interface and is sent back towards the transmitting crystal. The attenuation of the signal in a long mercury delay line path is 15 dB and this reduces the reflected signal to a level 30 dB below that of the primary signal. An artificial attenuator is inserted in the short lines so that the reflected signals are 20dB below the primary signal. This attenuator is a fine-mesh grid of steel wire which is mounted in the barrel at an angle of 45° to the axis.

The long lines make use of a double reflection path. They are built from two stainless steel tubes, ½ in. bore, 1/8 in. wall both mounted vertically. The tubes are held parallel and brazed into two blocks, one at each end, after which the projecting ends of the tubes are ground parallel.

A block housing the transducers is fitted to the top of the tubes and another, with 45° reflecting surfaces, is fitted to the bottom, all joints consisting of fine-ground flat surfaces held in contact. The wave path is down one tube across the reflecting block and up the other tube.

The reflecting block, which is a right-angled triangle in section, is bored to extend the tube holes to the two perpendicular faces and the holes in these faces are joined by a horizontal hole. Polished stainless steel reflecting disks are fixed to these faces, which are accurately ground so that the angle between them is 90°. The angle of incidence of the beam on the reflecting disks is thus 45°. In this connexion it should be noted that total reflexion occurs for angles of incidence greater than 30°.

The transducer block is also bored to extend the tube holes, and to accommodate the crystal mounting. A hole for filling is drilled in the side of this block. The mounting for the crystal holder is kinematically designed to provide two rotations about mutually perpendicular axes, and a vertical translation for path length adjustment. The overall length of the mounting is about five inches.

Assembly

The materials in contact with the mercury are steel, perspex and quartz. All the steel parts are cleaned thoroughly by wiping with benzene and finally rinsing in clean benzene or acetone. Care is taken to ensure that no brazing material gets into the tubes during construction. The quartz is cleaned in benzene or acetone but the perspex is cleaned only with a dry cloth. The parts are left to dry and are then assembled. Fresh mercury is used to fill each line but it is only a commercial grade, of 99.5% purity.

The signal obtainable from the line is very small immediately after filling but after 24 hours the output signal is normal. This effect is probably caused by small bubbles of air which form during the filling process and which gradually disperse. The angular adjusters are set to give a maximum signal and the line is then ready for use.

Working Tolerances

The bandwidth of the line and its transducers at the crystal frequency (15 Mc/s) is about 7 Mc/s. The input consists of 0.3 µs clock pulses, widened to 1 µs, at 1 Mc/s repetition rate. The bandwidth required to transmit these is about 2 Mc/s. The driving and receiving circuits are designed to give a total bandwidth of about 3 Mc/s so that the pulse rise-time at the output is 0.1 to 0.2 µs. The output pulse is reshaped to give a rise time of less than 0.1 µs, and is then used to gate clock pulses into the input of the line. Since the clock pulses are only 0.3 µs wide and the gating pulses are 1 µs wide, a timing tolerance of ±0.35 µs is available. The length of the delay line can be set to the optimum and controlled to +0.1µs which represents a temperature control of ±¼°C. Allowing for the pulse rise-time, the final working tolerance is ±0.15 µs. Each delay line is inspected daily but alterations are made much less frequently.

The total attenuation of the long delay lines (1024 µs) is about 60 dB at 15 Mc/s. By careful assembly and design the variations between one line and another are reduced but, as an extra precaution, a ±10 dB gain control is provided in the transmitter. The gain of the receiving amplifiers is not independent of temperature and of valve characteristics. Also, the long and short lines differ in attenuation. The gain control provides means of compensating for all these variations.

Normally a signal of about 70 volts D.A.P. (double amplitude peak) is sent to the transmitting crystal and the output from the receiving crystal is amplified to produce 30V pulses after rectification.

The long lines used in the ACE Pilot Model are housed in a large box 10 x 5' x 7 ft. high, which is made of 3/16 in. thick hardboard. There is a door at one end so that the lines can be inspected easily. A single fan stirs the air in the box, which is thermostatically controlled to 30 ±¼°C.

Cost

The cost of a complete mercury delay store such as that in the ACE Pilot Model, *i.e.* using long and short delay lines, is estimated to be about £10 per word of 32 digits or about 7/- per digit. Two thirds of this cost is for circuitry and the other third for the delay lines themselves.

Reliability

Four types of fault were mentioned at the beginning of this paper.

1. Complete and continuous breakdown
2. Continuous breakdown for certain patterns
3. Random pickup
4. Intermittent breakdown

Complete or intermittent breakdown can be caused by several types of failure

1. Incorrect length
2. Broken crystals
3. Misalignment of the crystals
4. Mercury creeping behind the crystals, and shortcircuiting them
5. Poor contact between the mercury and the crystal
6. A temperature gradient across the mercury

The required delay time is dependent on the delay in the circuitry but the lines can be set to the optimum delay by the length adjustments.

Faults due to broken or misaligned crystals do not occur except by accident or carelessness. Mercury has been known to get behind a crystal but if the perspex rim is machined flat and not mistreated, this should not occur.

The contact of the mercury and quartz can be impaired by contamination of either. Under normal conditions no such trouble has been found, but on one occasion during experiments solder got into the mercury and ended the life of the line with a disconcerting suddenness.

A temperature gradient across the mercury column can cause a corresponding velocity gradient and a consequent bending of the beam. This is never sufficient to cause trouble in the Pilot ACE unless a source of heat is deliberately applied to one side of the tube.

As far as the lines themselves are concerned, there are no features which can give rise to faults dependent upon pulse patterns, and the cause of any such faults, if they occur, must be sought in the circuitry.

There remain only faults represented by the loss or pickup of pulses. In this respect, mercury delay lines offer the exceedingly important advantage that they can readily be operated at signal levels such as to render them, practically speaking, immune from such faults. (The Pilot Model lines are normally used so as to give pulses of about 70 mV D.A.P. at the receiving crystal.) It is our common experience that in the absence of any specific and readily identifiable maladjustment, or any mechanical defect such as a faulty plug-and-socket connection or similar break in screen continuity, the storage system neither gains nor loses a single digit even in the longest runs.

Finally, it may be pointed out that m.d.l. storage is intrinsically free from any limitation on the number of times that any item of stored information can be read.

The working life of a mercury delay line is not yet known with certainty. In the Pilot Model, four long lines and nine short ones of the latest type have been working for about a year without any replacement, and other users can no doubt claim even longer runs. For all we know to the contrary, their life may be taken as indefinitely long.

An Experimental Multi-Reflexion Type

American workers have produced some very ingenious and sophisticated designs in which compactness is achieved by multi-reflexion paths. We also have experimented on these with some success. One of the lines now in use is a metal tank 9 inches by 7 inches by 1¼ inches. Eleven adjustable polished steel reflecting plates are distributed round the four narrow walls and the ultrasonic beam is sent round and round in the enclosure by 45° reflexions from these. The orientations of the surfaces were adjusted optically and the line worked without further adjustment when filled. The total attenuation in this line is about the same as in the comparable double-reflexion line already described, indicating that there is no appreciable loss in reflexion under these conditions.

An Apparent Anomalous Velocity Effect

The earliest models of the short lines were of fixed length, and it was observed that the delay was 0.3 μs longer than the calculated 32 μs. The discrepancy was confirmed by further experiment and it is understood that the effect has been noted elsewhere. There has not been time to elucidate it.

Speed, and Access Time

Mercury delay line storage can be used at very high pulse repetition rates, *i.e.* at very high digit frequencies. A practicable upper limit is set by a number of factors, *e.g.* crystal strength, temperature control requirements, attenuation, and some features of associated circuitry, in particular circuit delays. It appears that 5 Mc/s is about the highest digit frequency that could be realized in the present state of the art. This would require a carrier frequency of about 30 Mc/s.

As far as access time is concerned - the delay line store does not offer the possibility of direct access to required words which is such a valuable feature of some kinds of electronic store. On the other hand, as previous speakers have shown, a suitable logical design and the use of optimum coding enables the effective access time of a delay-line cum magnetic-drum system to be reduced to that of a single-word delay line.

References

1. MASON, W. P. Electrical and mechanical transducers and wave filters. Published by Van Nostrand, 1948.

2. WILKES and RENWICK. Ultrasonic memory unit for the EDSAC. *Electron. Engng,* 1948, **20**, 208.

3. HUNTINGTON, H. B., EMSLIE, A. G. and HUGHES, V. W. Ultrasonic delay lines. *J. Franklin Inst,* 1948, 245, 1, 101.

Discussion

MR. WILLIS (Cambridge University Mathematical Laboratory) asked if it was possible to transmit two or more pulse trains simultaneously using different carrier frequencies. The author said this was certainly possibly but had not been tried at NPL.

DR. PINKERTON (J. Lyons & Co. Ltd.) mentioned some difficulties that had been encountered in the operation of the LEO delay lines and in particular, failure due to the mercury not making a "wet" contact with quartz. A wetting medium of glycerine and water (with an anti-rust material added) had proved satisfactory. Polishing the crystal faces was a more expensive solution. The modulation system used was the same as at the NPL. He gave 5 dB as the attenuation in the mercury column as against the 20 dB in the NPL lines. The author in reply said that the polishing of a crystal face cost only five shillings. The higher attenuation in the NPL lines was probably due to the different carrier frequency as well as to the narrower bore.

In reply to a question about the cost of mercury delay line storage, the author gave an estimate of £130 for a 1000-digit line and its circuitry.

Mr. STRINGER (Cambridge University Mathematical Laboratory) confirmed the efficacy of the polishing of the crystal face to secure "wetting" by the mercury, and described a new type of line now being made at Cambridge, in the form of solid rectangular steel blocks bored with holes parallel to the upright sides and the base, with 45° reflecting plates at the junctions of the holes. An accurately machined plate, resting on the machined top surface of the block, carries the crystals.

DR. COOMBS (Post Office Research Station, London) said that they had had some trouble at Dollis Hill due to stray pick up at the low-level end of the lines, and advised that amplifiers should be mounted as close as possible to the low-level ends and that leads should be well screened and earthed. The author replied that there had been little or no pick-up trouble at the NPL and that operation with the largest practicable signals was a good safeguard against it.

27. Applications of Magnetostriction Delay Lines

by

R. C. Robbins and R. Millership

Elliott Bros. Ltd.

SUMMARY

Mercury delay lines have been widely and successfully employed. In comparison with the magnetostriction line they have certain disadvantages, chiefly their weight, size, complexity and temperature instability. The magnetostriction line is simple, cheap and

robust and may be constructed to give a zero temperature variation. The latter, however, may not be used over as great a frequency range. Other forms of solid delay line tend to be too complex.

It is felt that for use in digital computers the magnetostriction lines of the type described have considerable advantages over the other forms of delay line which have been used.

INTRODUCTION

The magnetostriction delay line is an acoustic line, in which the input electrical energy is converted into an elastic wave which then travels along the line and is reconverted into electrical energy at the output. A short variable delay line of this type which operated with a rectangular pulse input of the order of 5 µs in width has been described by Bradburd. It is the purpose of this paper to describe certain forms of magnetostriction delay line which have been developed and some of the ways in which they have been applied or in which their use is envisaged in electronic digital computing equipment.

DESCRIPTION OF DELAY LINE

General

The line in its simplest form consists of a length of thin wire or tape of a magnetostrictive material threaded through a coil near each end *(fig.1)*.

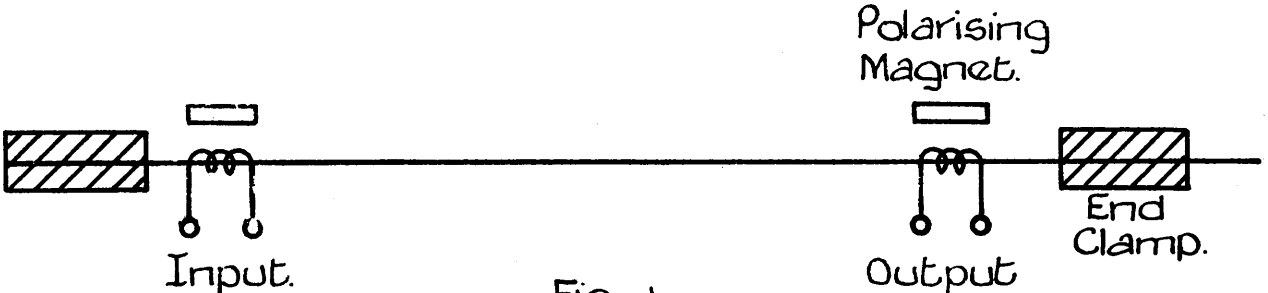


Fig. 1.

Simple Form of Magnetostrictive Line.

If an electric current pulse is passed through one coil the portion of line affected by the resulting magnetic field will undergo a change in dimensions. As a result, stress pulses will be transmitted along the line at the velocity of sound in the material. The stress pulse arriving at the second coil produces a change in the magnetic induction of the delay line material by virtue of the reverse magnetostriction effect and consequently an electric pulse is induced in that coil.

Output Pulse Shape

A simplified explanation of the production of the output pulse is given in *fig.2*. Consider a step waveform of current supplied to a delay line having one short coil and one long coil. As the magnetic induction at the input coil builds up, a stress proportional to the induction is established, the extent of which is determined by the length of the coil, and by the time required for the magnetic field to penetrate into the material. This stress pulse is propagated down the line in the direction of the second coil. As the stress pulse passes through this coil, the value of the magnetic induction integrated over the coil length alters in the manner shown. Consequently a voltage is induced in the output coil proportional to the differential of this waveform.

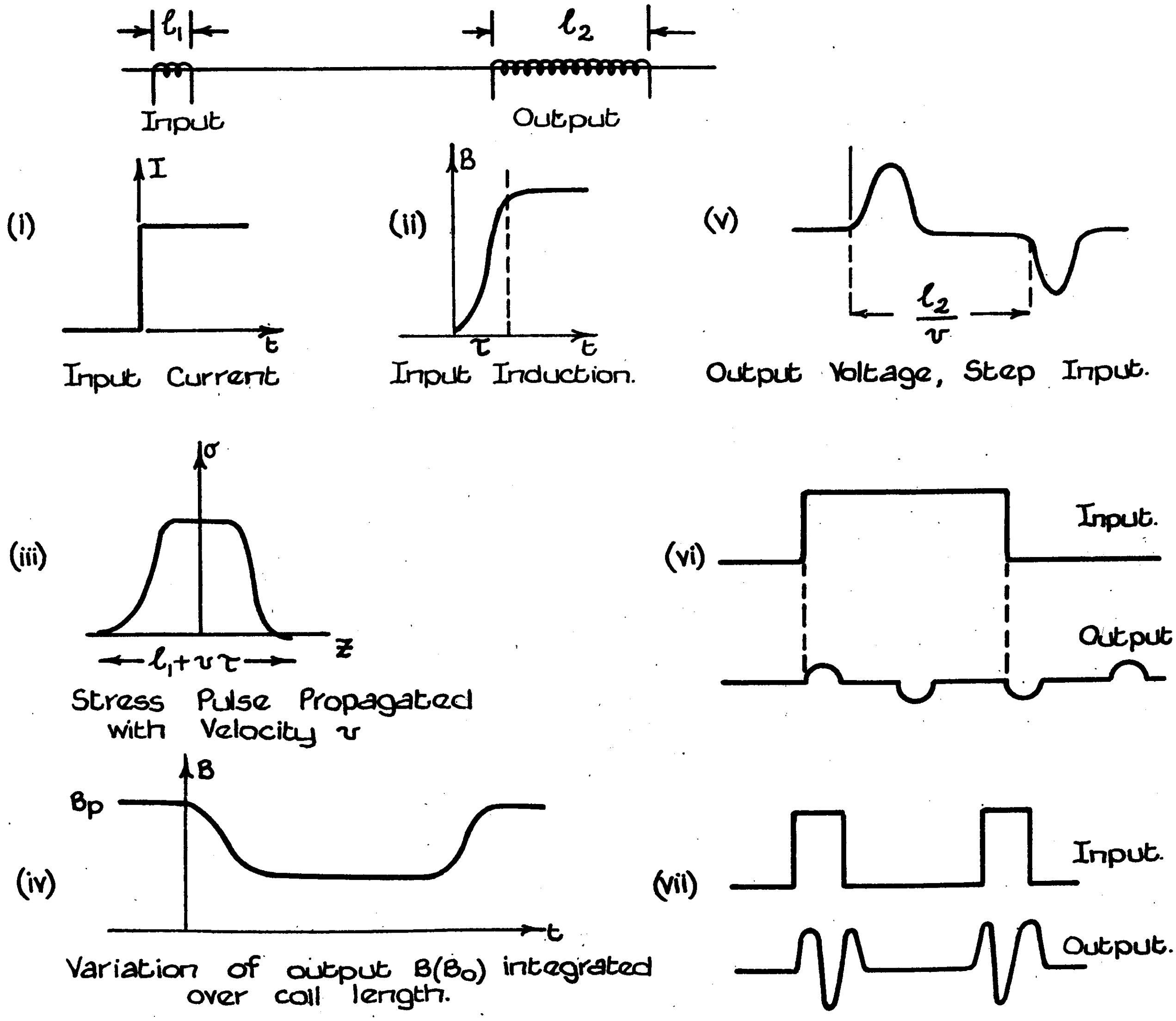


Fig. 2.

Production of Output Pulse.

From the input step wave, two output pulses will be obtained of width determined by the length of the shorter coil and the time taken to establish the stress pulse, and with a separation determined by the length of the longer coil. The same output is obtained if the coils are reversed so that the longer coil is in the input position. A negative going step will produce two similar pulses of the opposite sense so that a wide square wave will produce four output pulses. If the length of the long coil is reduced these may be combined so that two output pulses are obtained, each with a positive and negative going portion. With a narrow rectangular input pulse these also may be combined to give the characteristic output form shown in the diagram.

A number of factors have been neglected such as attenuation, possible mode conversion of the acoustic pulse, effect of input and output circuits, but the explanation serves to indicate the necessity for utilising coils of an appropriate length and of material sufficiently thin, if an adequate output waveform is to be obtained with a narrow input pulse. Examination of the type of output obtained with a very short coil and a very long coil has proved most useful in the investigation of this type of delay line and in the prediction of its performance.

End terminations

It is necessary to terminate the delay line at each end in some form of damping pad in order to prevent reflexion of acoustic pulses from the free ends of the line. This pad may consist of rubber or neoprene clamped to the end of the line, or the end may be immersed in grease or a termination of suitable shape and material may be moulded on the end. The size and shape of the termination will depend on the material used but suitable pads can be easily and simply constructed to occupy a very small space and yet reduce the reflexions to 1% or less of the required pulse.

Need for Polarization

The stress pulse in the line produces a change in the easy direction of magnetization of the magnetic domains. In the case of a material with positive magnetostriction a tension will produce a rotation towards the direction of the applied tension whereas for a material with negative magnetostriction the rotation is towards the transverse direction. If the material is in the unmagnetized condition the direction of spontaneous magnetization of the individual domains will be randomly oriented throughout the material. A stress will therefore produce a number of equal and opposite rotations, the net change of induction will be zero and consequently no output will be obtained. It is necessary, therefore, to apply a static polarizing field at the output position in order that a change of induction, and hence an output, should be obtained. The nature of the output will depend upon the direction and magnitude of the polarizing field so that for a unidirectional input pulse, a positive going, negative going, or zero output can be obtained. This effect has been utilized in the construction of a static magnetostriction system which will be described later. Also the shape of the output will depend on the direction and uniformity of this field and this fact has been used in the construction of some of the lines to produce a good output pulse shape.

The amplitude of the output pulse is a function of the magnitude of the polarizing field at the output. With a longitudinal field the output rises to a maximum as the field is increased and then diminishes as saturation is approached. For nickel the optimum value is of the order of 60 oersteds which can readily be realized by using a small permanent magnet which is adjusted for maximum output from the line.

It may also be of advantage to have a polarizing magnet at the input position, although this is not necessary when large input pulses are used.

Losses

The losses in the delay line are such that amplification of the output is necessary if a usable pulse is to be obtained. These losses are largely due to inadequacy of the electromechanical coupling at the input and output and very little is due to attenuation of the acoustic wave. The value for nickel of the latter quantity is approximately 6 dB/ms. It is possible to construct delay lines of considerable length since the attenuation forms such a small part of the overall loss and lines of about 6 ms. delay could be operated with a reasonably simple amplifier. It appears likely that distortion of the pulse rather than amplitude considerations sets the limit to the delay times which can be obtained.

For a line constructed of nickel the attenuation is smaller for the material in the hard drawn state than when annealed but the magnetic properties are better in the annealed condition. It is, therefore, advantageous in such a line to construct the line of hard drawn material and to anneal this at the coil positions.

Delay Time

The delay time introduced is, of course, dependent upon the path length of the acoustic pulse and on its velocity. The velocity of longitudinal waves is given, approximately, by $v = \sqrt{E/\rho}$ where E is Young's Modulus and ρ is the density. Perhaps the most suitable magnetostrictive material for the simple form of line is nickel, which has a high magnetostriction constant and is readily available in a suitable form. The velocity of sound in nickel is such that the delay time obtainable is 5.2 us/in. so that for a 1.5 ms. delay 24 ft. of line are required. The line, however, may be coiled and this length can easily be accommodated in a space about 18 in. square. A line may, in fact, be coiled, within limitations, to any convenient shape and could, for example, be wound round its associated amplifier.

Temperature Stability

Since the delay time is a function of the elastic constants and the length it will, in general, be temperature dependent. The variation is largely due to the variation of Young's Modulus, the thermoelastic coefficient being much greater than the coefficient of linear expansion. For nickel the thermal coefficient of the delay time is $1.4 \times 10^{-4}/ {}^{o}C$ or 0.14 μs/ms/°C which is less than half that of a mercury delay line.

This variation, however, although small may still be too great for certain applications, particularly if very long delays are required. Using the simple type of construction, there is no material which is sufficiently magnetostrictive and yet shows a sufficient decrease in temperature dependence. However, with very little increase in the complexity of the system a line may be constructed in which the main body of the line consists of non-magnetostrictive material and the ends only are plated with nickel or have magnetostrictive material attached. For a line constructed in this manner with Elinvar as the line material, the thermal coefficient of delay was -7.2×10^{-6} °C or less than 1/100 μs/ms delay/°C. This value could in fact be improved by suitable heat treatment but for most practical purposes such a line may be considered as being independent of temperature. It should be said that this type of line may be operated without plating the ends - in fact in the same manner as a nickel line but there is some decrease in amplitude.

Apart from its simplicity of construction and space saving considerations, an attractive feature of this type of line is that the delay can easily be varied. Also an output can be obtained from any point on the line and may be obtained from several positions. This feature has been used in the construction of pulse generators and the static store mentioned previously.

EXAMPLES OF THE USE OF MAGNETOSTRICTION DELAY LINES IN TWO TYPES OF SERIAL STORAGE SYSTEM AT 330 Kc/s

Two examples will be given to illustrate the application of magnetostriction delay lines as storage elements. The first example is that of a 1.5 ms. delay line used as a 16-word (512 digit) storage loop. It represents our first serious application of this type of storage. The second example is taken from a recent development on which work is still in progress, and is that of a single package 1-word store. Between these two we have other applications which are in current use.

A. 1.5 ms. Line used as a 16-word Storage Loop (512 digits at 330 kc/s)

Type of Line

The delay line consists of about 24 ft. of 42 SWG nickel wire wound in a loose flat spiral and resting lightly on its supports. The sending and receiving coils are alike and both consist of 500 turns of 50 SWG wire wound on a glass former about 0.05 in. long. With the nickel wire threaded through the coil, the inductance is about 150 μH and the D.C. resistance is about 50 ohms. A driving pulse of 40-50 mA peak gives an output of about 2 mV.

End reflexions are reduced to a negligible level by means of terminating pads consisting of short lengths of rubber tube filled with a suitable grease.

Circuit and Wave-forms (Figs. 3 and 4)

Input pulses from the computer are of greater width and amplitude than clock pulses. They are applied to a gate which lets clock pulses through to a pentode driving stage

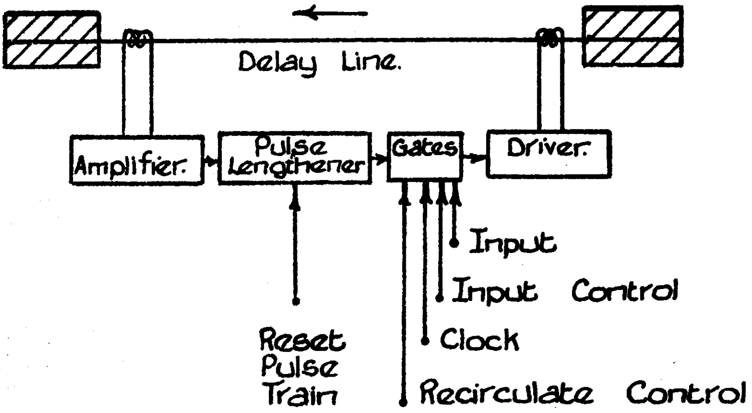


Fig. 3. One Form of Magnetostriction Delay Storage Loop.

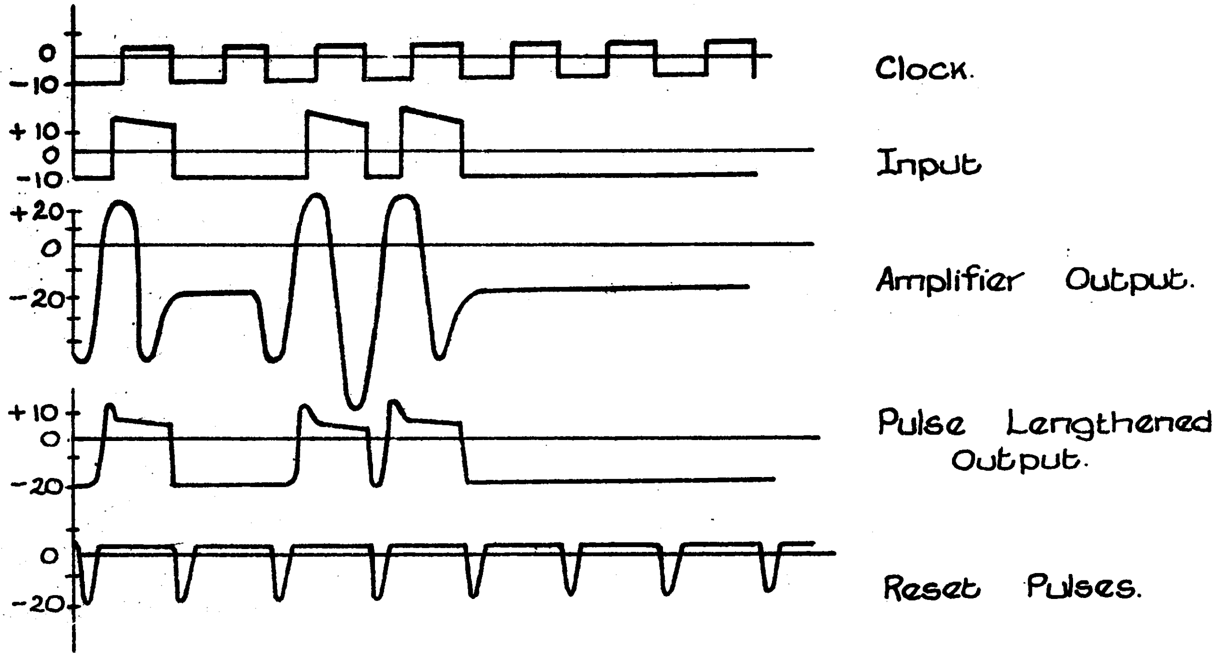


Fig 4. Diagrammatic Wave-Forms in 1·5 m.sec Delay Store.

delivering about 8 mA pulses to the primary of a current step-up transformer. This in turn delivers the required drive current to the sending coil on the delay line, a polarising magnet being used to preserve a unidirectional exciting field.

Output signals from the delay line are amplified by means of a 3-stage pentode amplifier to an amplitude suitable for clipping and pulse-lengthening. The overall voltage amplification can be set to any value between 10000 and 40000 approximately. As the signal waveform is substantially balanced over a digit period, a good low-frequency response is not required and the amplifier is deliberately given a band-pass characteristic extending from about 75 Kc/s to 750 Kc/s. The positive portion of each digit signal is used to charge a pulse-lengthening condenser which is discharged at the end of each clock pulse period by an externally generated reset pulse.

A more or less conventional crystal diode array is used to control input and recirculation of stored words.

The complete circuit uses four E.F.91 pentodes, a 12AT7 double triode (as a pair of cathode followers) and an E.B.91 double diode (for the pulse lengthener). The power consumption is about 13 Wh D.C. and 12 Wh A.C. for valve heaters.

Composite Storage Loops and Performance

Blocks of eight such 1.5 ms. lines have been connected in cascade, with a regenerating circuit between each pair, to form 128-word storage loops each with a total circulation time of 13 ms. This arrangement has no particular advantage other than economy in address circuits and for this reason was used in a particular computer for which the correspondingly long access time was acceptable. (It should be mentioned that with improved techniques and low temperature coefficient line material, it now seems practicable to use delay lines of up to 3 ms delay.)

As in this application nickel lines are used, the effect of temperature change amounts to a delay change of about 1/12 digit period/^{o}C. Means have been taken to compensate for this by using a clock whose frequency is accurately controlled by the delay in a similar line, housed in the same enclosure with the storage loops. With Elinvar lines this should not be necessary.

This particular application has worked sufficiently well to demonstrate a number of advantages of this method of storage, but has also proved useful in demonstrating a number of weak points in the original design. Some of these have already been corrected on the machine itself, others have been avoided in later applications.

The chief troubles have been:

1. Signal distortion due to excessive support constraint *(fig.5)*. This effect was markedly microphonic and could be much reduced by keeping the lines in a state of low frequency vibration. Improved supports have almost entirely removed this trouble.

2. Change of pulse width with duty cycle *(fig.5(iii))*. This fault though not serious has the effect of reducing delay tolerances. It results from trying to get too much advantage from a transformer drive. The effects of current reversals due to A.C. coupling can be reduced but not removed by the use of a polarizing magnet, and this reduction becomes less useful the greater the driving field compared with that necessary to saturate the nickel. It is now considered that there is little to be gained by using a current step-up transformer, and so D.C. drives are being used.

3. Interference. This was largely due to bad earthing and inadequate filtering of supply lines. The worst of this was cured by the insertion of a small metallic shield between the pick-up coil and the base plate of the delay line.

4. Inadequate gain stability.

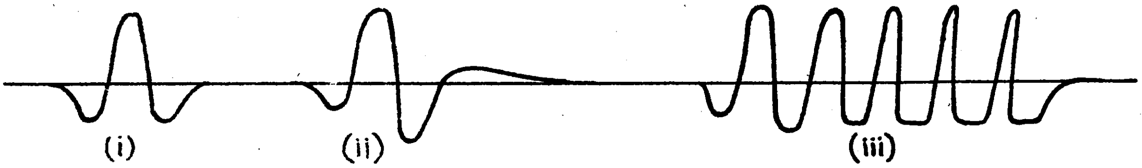


Fig. 5. Two Forms of Delayed Signal Distortion.
(i) Undistorted (ii) Constraint Distortion
(iii) Effect on Pulse train of Overdrive using transformer input.

A Single Package 1-word 34-digit Storage Loop

Type of Line

In this case the delay line consists of a single loop of about 2 ft. of nickel tube of one of the standard sizes used in the manufacture of valve cathodes. The outside diameter is 0.036 in. and the inside diameter 0.031 in. It is slotted and annealed in the region of the input and output transducers. The coils have 900 turns of 49 S.W.G. wire and are about 0.15 in. long. Under working conditions they have an inductance of about 1½ mH. With input pulses of 20 mA peak, a useful output from the line of 1/3 to 1/2 V is obtained. The two ends of the line are terminated with neoprene pressure pads.

Circuit and Wave-forms

The circuit is considerably simpler than that described for the 16-word store. It employs a 6CH6 pentode to provide a direct drive of 20-mA pulses. Cathode degeneration is used to define this current within reasonably close limits so that ageing of the valve (which, however, is very much under-run) should not cause rapid deterioration of drive.

The line output amplifier is a 12AT7 double triode with a small amount of overall current feedback. The positive portion of its output is strobed by means of a narrow pulse (about 0.4 µs wide) and a suitable diode gate.

The output from this diode gate is then lengthened by a circuit using a 12AT7 double triode and the lengthened pulse is terminated by means of an externally generated reset pulse of the kind used throughout the computer. The resulting lengthened pulses are of the standard form required by the logical circuits of the computer.

Gating of input and regenerated signals is carried out by means of a crystal diode array of much the same type as that used in the 16-word storage loop.

In all, the circuit uses 3 valve envelopes (two double triodes and a pentode) and 14 crystal diodes. It consumes about 9½W D.C. and 8½W A.C. for valve heaters.

The whole circuit including the delay line goes into a plug-in package 6¼ in. x 9½ in. x 1¾ in. thick.

So far only preliminary bench tests have been carried out but these have been encouraging. Lying open on the bench it has stored for about 7 hours without error, this being the longest period over which we have tested it. It is very non microphonic and can be hit quite hard while in operation, without adverse affects.

COMPARISON OF DELAY LINES

Requirements

The major requirements for a delay line are that:

1. A reasonable, though not necessarily faithful, reproduction of the input should be obtained.

2. The overall energy loss should be small.

3. The delay should be reasonably constant with variations of temperature etc.

4. The size should not be excessive nor the construction over-elaborate or fragile.

These requirements may be realized by making use of an acoustic line. Although a gas would appear to be the most suitable material since the delay for a given length will be great, the bandwidth requirements for pulse operation are such that a solid or liquid medium is preferable. The attenuation of the higher frequencies is not so great in these media and an efficient electromechanical transfer can be obtained. The most widely used transducing mechanism is the piezo-electric crystal and of these X-cut quartz has been most frequently employed.

Mercury Delay Lines

There are certain apparent advantages in using a liquid as the transmission medium. Shear wave propagation is excluded and in general the velocity is lower than for solids.

Mercury is very suitable since the mechanical impedance is high and a wider bandwidth can be obtained with a quartz crystal than can be obtained, for example with water. Furthermore mercury has a very low attenuation for sound waves (1.9 dB/ft. for a 0.25 - 0.5 μs pulse). A good match may be obtained with a quartz crystal so that a reflexion coefficient of only 12% or less may be obtained.

The velocity of sound in mercury is reasonably low (17.25 μs/in.), about the same as in water, so that a substantial delay may be obtained in a reasonable space and the variation of delay with temperature is fairly low ($-340 \times 10^{-6}/{}^{\circ}C$) which is considerably lower than for most liquids.

The problem of delay variation with temperature may be overcome with a liquid line by utilizing the fact that water has a temperature coefficient of the opposite sign to that of most liquids, so that by mixing it with other fluids (*e.g.* ethanol, methanol) in the correct proportions a small or zero coefficient can be obtained. In this type of line, however, the mechanical impedance of the fluid is still low and the attenuation is higher than for mercury.

Mercury lines have been widely and successfully used but have certain disadvantages when compared with the magnetostriction lines described. The major disadvantages are their weight and size, even with a folded line, and the complexity of input and output systems which require considerable accuracy in machining and alignment. The delay can only be varied over a small range and even then a fairly complex system is required. The signal is sensitive to temperature gradient and there is the necessity for containing the liquid in a robust sealed container and the attendant difficulties of transporting the line.

The temperature variation although small is still so large that some form of temperature compensation is necessary for a long delay.

Magnetostriction Delay Lines

In comparison the magnetostriction lines are extremely simple, being cheaply and easily constructed and yet are extremely robust. The delay can easily be varied so that a wide tolerance in manufacture is allowable since the line can be rapidly and accurately adjusted to the required delay. Consequently no accurate machining is required. A delay time of considerable length can be made to occupy a very small space and the line is light and easily transportable. Moreover the lines may be made independent of temperature without detracting from their properties and without any substantial increase in complexity. The loss is much the same as that of a mercury line and in fact magnetostriction delay lines may be constructed to give a much greater output.

The major disadvantage of the magnetostriction line lies in its frequency limitation. Lines have been operated with 1 Mc/s pulse input but it appears that in practice the upper limit of this type of line would be about 2 Mc/s, whereas a mercury line may be operated with an input pulse frequency well above that figure.

In the case of the simple magnetostriction line the problem of bonding the transducer to the medium and obtaining a match does not arise since the medium is the same throughout. It is necessary to use wire or material sufficiently thin to allow rapid penetration of the applied magnetic field but the use of material in this form does closely define the acoustic path due to the gross mismatch with the surrounding air. Suitable materials are available in this form. Again the delay in the magnetostriction line is easily varied and the output is available at any point along the line. This latter fact is true even in lines with zero temperature coefficient when these are constructed in the simplest form, although there is some loss of output because of the poor magnetic properties.

Other solid delay lines

Other forms of solid delay line, with piezo-electric transducers, have been used, although not, it is believed, in computing equipment. The major apparent disadvantages are that multiple mode propagation is possible and it is in fact somewhat difficult to eradicate secondary vibrations and the complex pulses which arise due to poor contact between the crystal and the medium. The attenuation is higher, in general, than for mercury and the velocity greater although this disadvantage may be counter-balanced by the fact that a more compact form of line can be used.

Fused quartz has been used to construct some delay lines, largely because the attenuation is small and a good accoustic match is possible with the transducing crystal. An X-cut quartz crystal bonded to a fused quartz "line" shows a power loss 7 dB better than for mercury. Short delays have been obtained using a straight through path in a rod of quartz and longer delays have been obtained by Arenberg using multiple reflexions in a solid block. The chief disadvantages of this latter system were that multiple pulses were liable to occur, very accurate machining of the block was necessary and difficulty was experienced in obtaining quartz blanks of the required size. Solid delay lines have also been made using magnesium alloys which have a sufficiently small attenuation. In this case a Y-cut quartz crystal was used to propagate the shear mode, since the attenuation for this mode is smaller than for the longitudinal mode and also the velocity of propagation is smaller. The technical problems of bonding shearing crystals to solids are considerable and consequently this type of line suffers from some of the disadvantages of a quartz line.

Electromagnetic delay lines

For very small delays it is probably more convenient to use an electromagnetic delay line and it is possible that a combination of short electromagnetic delay lines with transistor coupling would produce a long and yet economical delay line.

OTHER APPLICATIONS OF MAGNETOSTRICTION DELAY LINES

Frequency Dividing Circuit (Fig. 6)

This system is certainly not novel, but it is one which works out in a very convenient and simple form using a magnetostriction line.

A single-shot pulse generator is used to launch an acoustic pulse along a line, the delay of which is adjusted to the reciprocal of the lower frequency required. The output pulse from the delay line is then used to gate a pulse from the given pulse train and so to re-fire

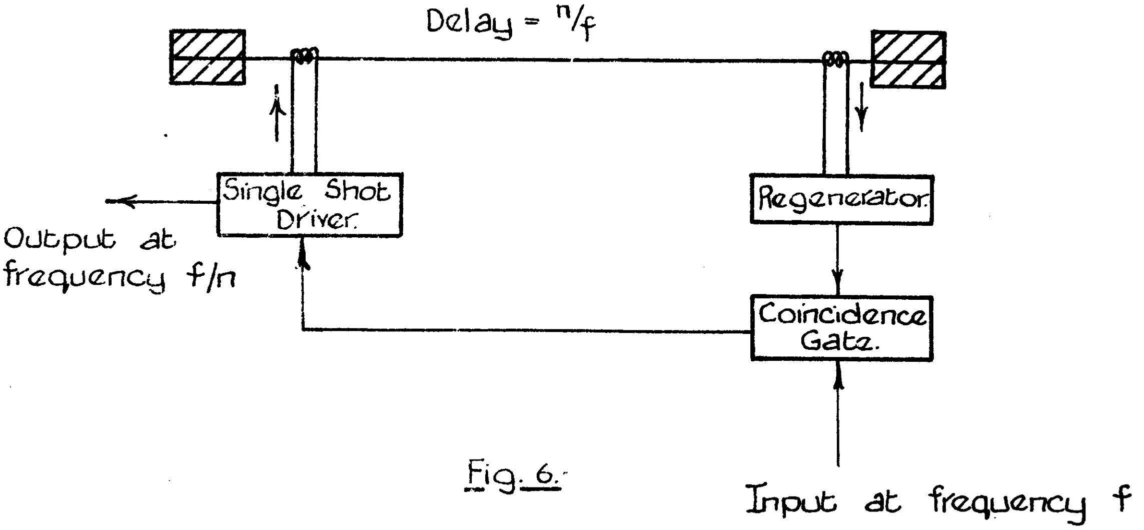


Fig. 6.

Frequency Divider

the single shot generator. Means should be taken to prevent the possibility of more than one pulse circulating round the loop, but with this arrangement it is possible to obtain division by any integer within reason. For high division ratios, high stability of delay is necessary so that a low temperature coefficient delay line is desirable. This feature, together with the general ruggedness and the ease with which the delay can be varied, makes the magnetostriction delay line very useful for this application.

Serial Word Generator (Fig. 7)

One advantage of the magnetostriction line over other types of acoustic delay line lies in the fact that one can arrange coils at intermediate points along the line without serious effect on the transmission.

It is quite possible to construct a magnetostriction delay line for 330 Kc/s with 30 or more coils spaced at digit intervals. If the line is suitably polarized, a single pulse propagated along it will induce a response in each coil in turn so that by connecting the appropriate combination of coils to a common pulse regenerator circuit, any required serial number of about 30 digits can be set up.

The system can be worked either way round, and, in fact, it may be advantageous to have only one coil connected to the regenerator and to do the switching at a high level by directing the driving current pulse to the appropriate combination of coils.

If for practical reasons coil packing becomes excessive, two or more lines can be used with suitable interleaving of the coil positions.

Static Storage on Magnetostriction Delay Line (Fig. 8)

The polarity of the output from any given coil in the word generator just described will depend on the direction of magnetic polarization of the line within that coil. Such polarization can in fact be provided by the remanent magnetization of the line itself.

If therefore a positive response is used to indicate a "1" and negative response to indicate a zero, a number may be stored in static form on the line by locally magnetizing it in the appropriate direction at each pick-up coil position. If all the pick-up coils are taken to a common regenerator circuit, the stored number can be read off, without erasure, by transmitting an acoustic pulse along the line.

This system has been demonstrated in principle but has not yet been applied.

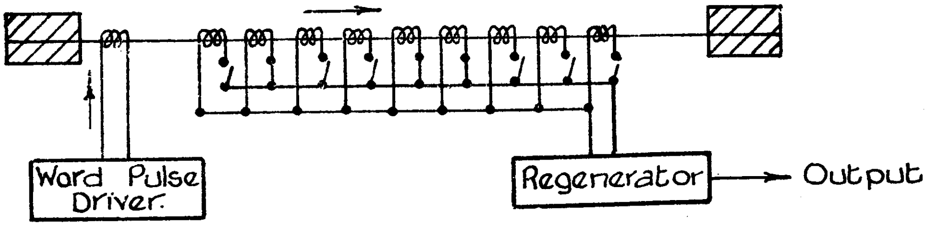


Fig. 7.

Serial Word Generator.

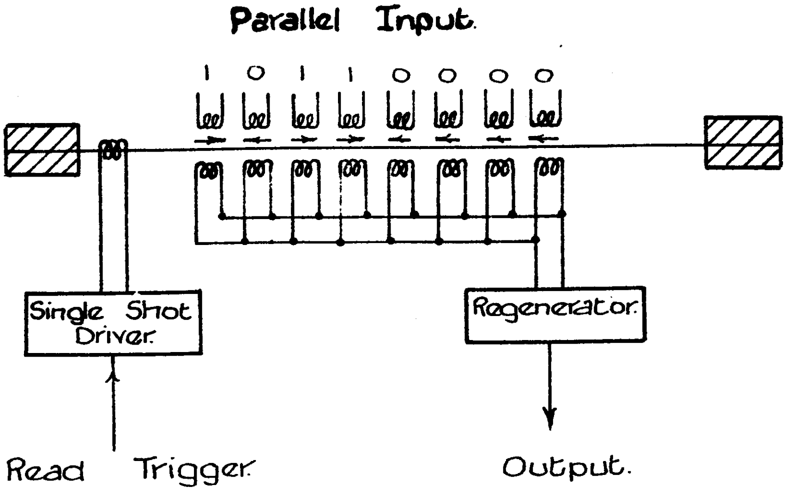


Fig. 8.

Static Storage.

COMPARISON BETWEEN DELAY LINE AND OTHER HIGH-SPEED METHODS OF STORAGE

In general the choice of storage system to be used in any given set of circumstances involves too many considerations to be discussed here. Instead we shall try to consider some ways in which the development of magnetostriction delay stores may modify previous conclusions.

Mercury delay lines are probably the most likely choice for digit frequencies higher than 1 or 2 Mc/s, though for small capacity storage the electromagnetic delay line used in conjunction with transistors, seems a probable rival.

Similarly the Magnetic drum is probably the inevitable choice for non-volatile storage capacity greater than about 100000 digits. (Less likely would be the use of two-state toroids.)

Again, c.r.t. storage meets the requirements for rapid access storage of medium capacity (a few tens of thousands of digits).

It might seem therefore that the field is well covered by well-proved techniques and that there is no need to introduce a new one which has limitations all of its own. Against this, magnetostriction line storage is relatively cheap and simple to construct, the circuits are simple and it can be almost unaffected by temperature. It is suitable both for small capacity rapid access storage (up to one or two thousand digits), and for medium capacity long access storage (a few tens of thousands of digits) and can be used in conjunction with large capacity magnetic drum storage without any special means for digit rate conversion.

With mercury delay line storage, one can synchronise the drum to a line, but it is necessary then to keep all lines of the delay storage system at nearly the same temperature as each other. Using magnetostriction storage, the drum can be synchronised either to a crystal or to a line, and temperature changes need have little effect.

Where only medium capacity long access storage (say, 50000 digits) is required, long line magnetostriction storage may well appeal to those who revolt at the thought of rotating machinery. It certainly seems to compare favourably with long line mercury storage for digit rates below about 1 Mc/s, on grounds of simplicity, cost and temperature insensitivity. Though a medium capacity store using magnetostriction lines would probably need more equipment than would an equivalent drum, such equipment would consist mainly of a number of similar circuits of a simpler and less specialized type than would be used with the drum. On the other hand, the large extent to which the capacity of a drum can be increased with little increase in equipment and cost, and the non-volatility of this form of storage, are advantages which will often outweigh those of the delay line.

The argument in favour of using the type of storage system possessing the simplest and least number of types of circuit, probably weighs more heavily in the case of the small capacity rapid access store. A rapid access store for about 30 words, using magnetostriction store packages of the kind described in the second example, would require a total of about 200 valves, rather over half of which would be associated with address finding. This is 4 or 5 times the number required by the equivalent c.r.t. store, but it would use only the one type of package differing from those used elsewhere in the computer. Servicing should therefore be considerably simplified without likelihood of reducing reliability.

Two further considerations are worth mention:

1. The rapid-access magnetostriction store can probably work up to 2 or 3 times the digit frequency practicable with a c.r.t. store but not much higher. It has no "read-around-ratio" limitations and no time is lost by need for regeneration.

2. The useful signal level at the output of a short magnetostriction delay line at 330 kc/s is between 40 and 50 dB higher than that appearing at the signal plate of a c.r.t. store at the same frequency.

Discussion

MR. COOKE-YARBOROUGH (Atomic Energy Research Establishment, Harwell) asked about the best wire diameter, annealing, and method of support. The author replied that the wire should be thin enough to allow complete flux penetration. Two types had been used at Elliotts - nickel tube of 0.0025 in. wall and 0.036 in. outside diameter for short lines, and 42 SWG nickel wire for long lines. Nickel lines are improved by annealing, though the increased permeability leads to increased flux-penetration time. The essential feature of the wire support is small constraint.

MR. BRADFIELD (NPL) said that the magnetostrictive delay line was first used in 1947 in the USA. In 1948 it was used at the Telecommunications Research Establishment and had been further developed by Ferranti's. It was used at the NPL for the precise measurement of the elastic constants of alloys. The method of support adopted at the NPL was to thread the wire through a grease-filled tube. Greater energy could be obtained by the use of torsional vibrations instead of longitudinal ones. A suitable material for the wire was a cobalt-iron-vanadium alloy, if properly made and annealed. Barium titanate tubes had also been used. MR. WRIGHT (NPL) asked Mr. Bradfield what was the maximum frequency that could be used with magnetostrictive delay lines.

MR. BRADFIELD replied that he estimated the limit with 42 SWG wire to be 0.5 Mc/s. He said that it had been found with mercury delay lines that by fixing to the crystals a backing of plastic, loaded with tungsten powder, a very good impedance match was obtained with consequent good dissipation of energy and no reflexions. The bandwidth was also improved.

MR. ELLIOTT (Elliott Bros. Ltd.) emphasized the cheapness of magnetostriction stores.

28. Cathode Ray Tube Storage

by

T. Kilburn

Manchester University

Introduction

Since the theoretical aspects of cathode ray tube storage are considered in a paper to be published elsewhere *(ref. 1)* and the reliability of the c.r.t. store in the present Manchester machine is discussed in companion papers at this Symposium *(ref. 2, 3)*, this paper is confined to the practical application of c.r.t. storage to a machine now being built at Manchester University. It is perhaps of some interest to consider first the general shape of the machine itself, before confining the discussion to the c.r.t. store.

The machines built in recent years at Manchester have operated completely in the serial mode. Though it is realized that in some applications speed is the dominating factor and leads in 'fixed-time' machines to parallel operation with little regard to cost, it is still felt that serial operation is in general correct for 'scientific' machines, since less total equipment is required. The statement that parallel machines are preferable because they have more identical units is not thought to be of much importance in view of the fact that the total number of units is greater. Further, it is thought that parallel machines tend to be somewhat inflexible, since the high cost of parallel registers leads to the provision of only a limited number of them. This puts an unpleasant restriction on both the machine designer and user, especially since the arrival of the B-tube.

The bias which this general line of argument gives towards serial operation has been unfortunate for c.r.t. storage, since the rapid-access feature of this type of store is only partially exploited in a serial machine. However, experience has shown that the minimum capacity required of a rapid-access store is about 10000 digits, if flexibility of programming is to be retained and if the time for transfers from intermediate to rapid-access stores is not to become too large. Since 10000 digits, or, about 10 c.r.t's are essential it is no longer a question of saving equipment but of arranging essential equipment, namely 10 c.r.t. stores to the best advantage. Undoubtedly, considered from the point of view of the c.r.t. store alone, a parallel arrangement is to be preferred, since speed is achieved without extra cost.

With a parallel store postulated, should the remainder of the machine be operated in the parallel mode? For the reasons given above this question was answered in the negative. It was then desirable to operate serially at such a speed that the increased speed obtained from the parallel c.r.t. store was not squandered. For this to be so, the digit frequency had to be 1 Mc/s. This frequency, suitable for the computing circuitry, was too high for registers to be provided in terms of serially operated cathode ray tubes, which in any case are somewhat expensive. The answer to this problem is, of course, - especially within these four walls - delay lines.

With this background of a machine containing a parallel c.r.t. store and delay line registers established and the reasons for such a choice given, the organization of the c.r.t. store in relation to the basic machine rhythm is now described.

Parallel-Series: Series-Parallel Conversion

Fig. 1 shows schematically a 10-tube parallel c.r.t. store. This holds 1024 ten-digit words and operates at the standard period of 10 μs/digit. The read output of the store which is present for 5 μs is gated by a read pulse 1 μs in width, and inserted into a 10 μs electrical delay line, at points separated by 1 μs along the line. The timing of the front edge of the read pulse, which is shown by the vertical arrows in *fig. 2(i)*, is set to give the optimum tolerance for gating the 5 μs read output, and occurs at a time t_r (say 2 μs) before the end of an action period. By inserting a delay t_r at the end of the read output delay line, the serial output from the store is made to emerge exactly within the 10 μs scan period *(ref. 4, 5)* following the action period in which that word was scanned in the store. Thus, the waveforms generated to control the scan and action beats of the parallel store can also be used to control the serial computation.

A similar procedure is used for writing information into the store. Here, words arrayed in the write input delay line during a scan beat, are gated, or 'flashed' into the c.r.t. store during an action beat.

Basic Rhythm

From the previous section, it will be apparent that the normal arrangement of scan beats for regeneration alternating with action beats has been retained, except that here, of course, scan and action beats are only of 10 μs duration.

Fig. 2 shows the basic rhythm of 6 beats to a bar. The instruction, which is 20 digits or two words in length, is scanned during the action beats A1 and A2, and appears in serial form at the store read output during S2 and S3. During S2, the 'b,f' part of the instruction is taken to select the B-tube line and the function involved. Since the function part of the instruction is mainly concerned with computing as opposed to c.r.t. storage, nothing further will be said about this phase of the operation: except that the b digits cause addition to the 's', or address, part of the instruction during S3, if required. At the end of S3, 's' (or 's + b') selects the 10-digit number scanned during A3. Because only 1024 addresses exist in the store, only 10 digits are required for control transfers and B-tube operations, so that the "short number" operations are complete when the 10-digit number has emerged during S4. S4 is overlapped with the S1 beat of the next bar, and thus a short number operation takes 60 μs to perform. Numbers are written into the c.r.t. store by arraying them in the write input delay line during S3, and flashing them into the store at the beginning of A3.

For long number (40 digit) operation, the bar is extended to 120 μs to allow time for the selection of three further ten-digit words. It is natural to locate the four words required in addresses s, s + 1, s + 2 and s + 3 and the two words required for the instruction in c and c + 1, where c is the number in the control register. During action beats then, the addresses required are either c or s, or addresses obtained by adding 1 to these numbers. In a similar manner, addition of 1 to a ten-digit register, gives the addresses required for sequential scanning, for regeneration purposes, during scan beats. The way in which this normally unchanging addition of '1' during both scan and action beats is used to generate the c.r.t. raster and select raster addresses is now briefly described.

The Raster Circulator

Two 10 μs delay lines are arranged, as shown in *fig. 3*, in a closed loop in company with an adding circuit. One line, L, is tapped at ten points, five points being fed to the

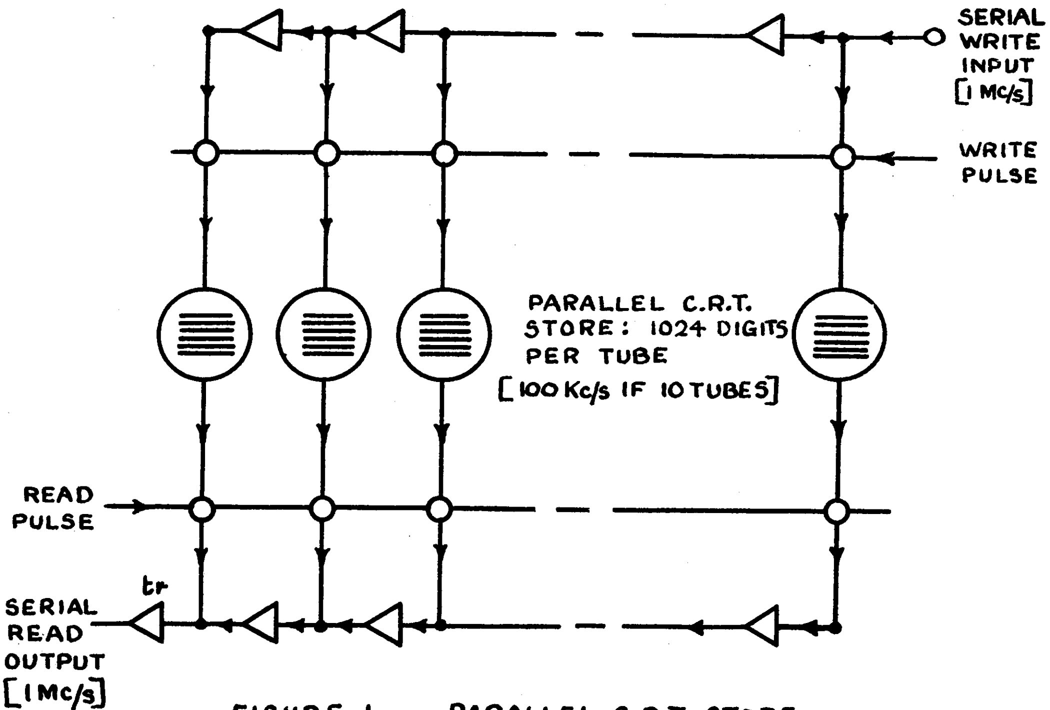


FIGURE 1 PARALLEL C.R.T. STORE - SERIAL COMPUTATION

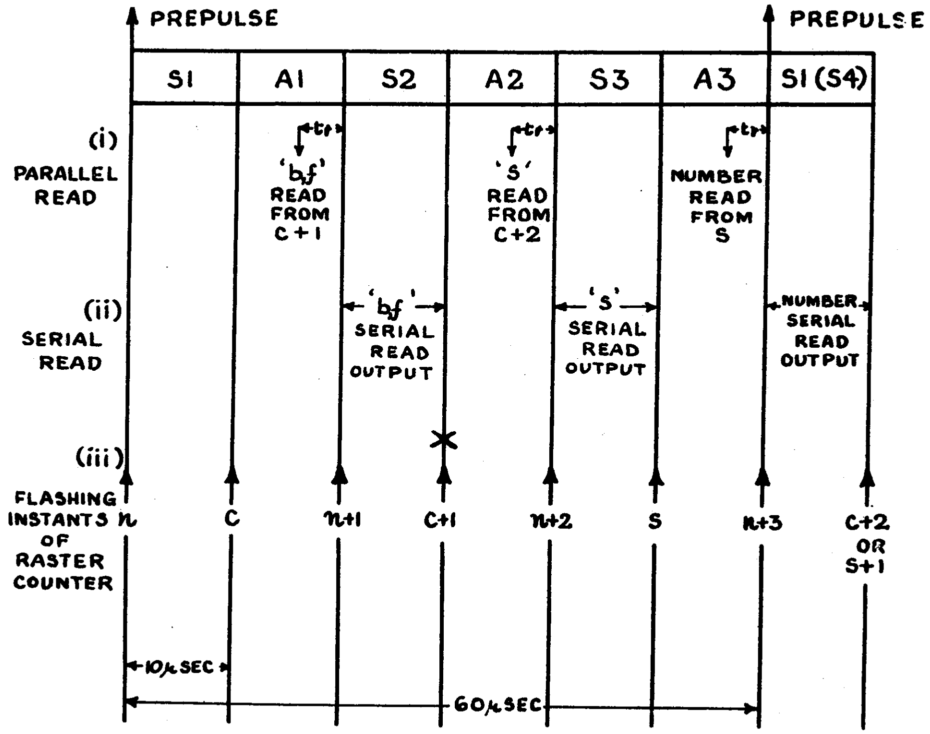


FIGURE 2 BASIC RHYTHM

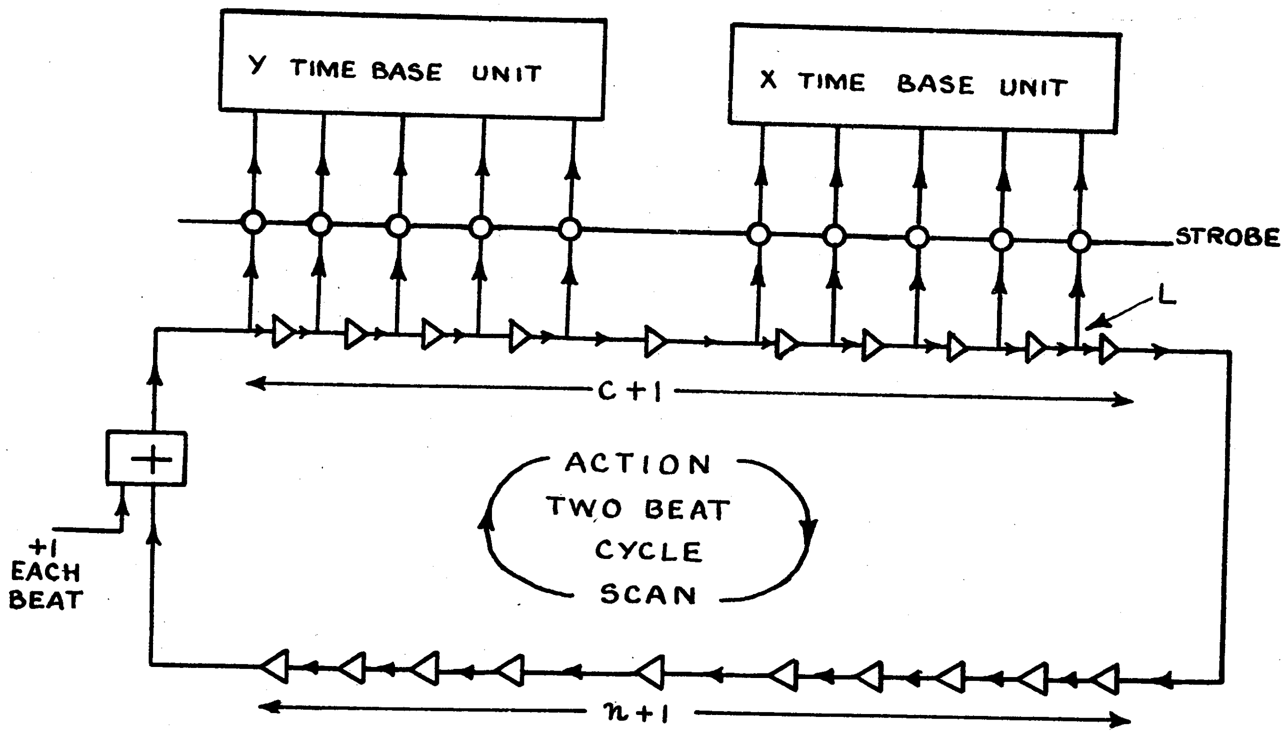


FIGURE 3 THE RASTER CIRCULATOR – AT INSTANT S2/A2

X time-base unit and five to the Y time-base unit. These connexions are made via gates so that L can be flashed by a microsecond strobe, exactly in the same way as that used for the write input to the c.r.t. store. The flashing instants for L are shown in *fig.2(iii)* by the arrows, the numbers arrayed along the line at these instants being given below the arrows. At a particular instant, say that between S2 and A2, c + 1 is arrayed along L *(fig.3)* and is flashed into the time base units causing the address part 's' of the instruction to be selected during A2. At the same instant, n + 1, the regeneration count is arrayed along the second delay line, but it is not used until the end of A2 when, increased by unity by the adder, it is arrayed as n + 2 along L. It will be seen that:

1. the regeneration count is increased by one every alternate beat, causing sequential regeneration of the raster.

and

2. appropriate replacing of c by s, and s by c causes the correct selection of active addresses, for example

c; c + 1; s; s + 1; s + 2; s + 3; c + 2; etc.

This method of raster generation and selection is much more economical than methods using thermionic valve binary counting.

Results

The methods described are operating successfully though the most stringent testing will occur only when the machine itself is complete.

In some parallel c.r.t. stores, serious trouble has arisen because only small read-around ratios have been achieved, whilst the machine design was such as to demand large ratios. In the present case, the maximum theoretical ratio of 1024 has been obtained experimentally, 512 has been obtained with some degree of certainty, and 256 with what is thought to be complete certainty, though as stated above only the machine itself can give a safe answer this question. However, since the maximum read-around ratio which will ever arise in practice is 228, no difficulty is expected.

Conclusion

The association of cathode ray tubes and electrical delay lines has met with considerable success. It seems reasonably certain that the speed of the final machine based on this principle will be ten times that of the present machine, and that the final machine will only require, when other factors are also taken into consideration, less than half the number of valves used in the present machine.

References

1. WILLIAMS, F. C., KILBURN, T., LITTING, C. N. W., EDWARDS, D. B. G. and HOFFMAN, G. R. Recent advances in cathode ray tube storage. To be published in the *J. Instn. elect. Engrs.*

2. WILLIAMS, F. C. Report on MADAM. (*See* Paper 4).

3. ROBINSON, A. A. Component reliability. (*See* Paper 3).

4. WILLIAMS, F. C. and KILBURN, T. A storage system for use with binary digital computing machines. *J. Instn. elect. Engrs.*, 1949, 96, Part III, 81.

5. WILLIAMS, F. C., KILBURN, T. and TOOTILL, G. C. Universal high-speed digital computers: a small-scale machine. *J. Instn. elect. Engrs.*, 1951, 98, Part II, 13.

Discussion

In reply to Dr. Pinkerton, the author said that it was not necessary to regenerate after each delay line. Three valves sufficed for regeneration after 16 lines.

MR. COLEBROOK (NPL) said that the respective merits and demerits of c.r.t. and delay line storage were very nicely balanced and it would be difficult to guess which was likely to be favoured in future, quite apart from the fact that both might be superseded.

MR. McPHERSON (International Business Machines, World Headquarters, New York) referred to I.B.M.'s development of a truly parallel 2000-word c.r.t. store using 3 in. tubes. Regeneration is carried out during the action period. Special tubes which impose no programme limitation have been successfully developed.

MR. NEWMAN (NPL) remarked, that it looked as if a combination of c.r.t. and delay-line storage could result in a machine almost as fast as a delay line machine.

DR. BOWDEN (Ferranti Ltd.) pointed out that increase of computation speed gave a diminishing increase in total speed of solution of complete problems. He had calculated that on average, increasing the Manchester machine speed to infinity would only give a 50% decrease in the total time for a problem.

DR. HULSIZER (Illinois University) mentioned the two machines of the Von Neumann type they had made— one being retained at the University for research. The store is fully parallel, with 40 ordinary commercial tubes, of which one in five proved to be serviceable. The read-around ratio had been raised to 120 by some recent improvements. Access time is 18 μs and addition takes three access times. The normal method of input is a Ferranti tape reader.

29. Memory Studies and other developments at the National Bureau of Standards

by

Ralph J. Slutz

National Bureau of Standards

INTRODUCTION

Since 1946, the National Bureau of Standards has been engaged in a comprehensive program of component development for electronic digital computers. While much of the early effort was directed toward input and output equipment, special electro-static storage devices, based on the Haeff type of holding beam tube, were investigated. This is the kind of storage where you have a holding gun of slow electrons in addition to the writing gun, and so attempt to have continuous stability in the tube. This was found to be beyond the state of tube construction art in those days, and it did not seem sound at that time to go for extensive construction of so complicated a special vacuum tube. Incorporation of storage devices into complete memory systems for use in a computer began with the design of SEAC* *(ref. 1)* in late 1948. Originally the SEAC was intended to be an "interim" computer ---- to be completed quickly and simply and used in the interim caused by the delayed delivery of commercial equipment. The circuits were to be copied from existing techniques and the logical structure was to be minimal.

It soon became evident, however, that the then "existing techniques" left much to be desired, and they required extensive engineering to make them satisfactory. While carrying this out, it also proved possible to generalize the minimal logical structure, making provision for future additions adequate to complete a full-scale machine *(ref. 2)*. In spite of the extensive engineering needed, the installation was put into regular operation a little over a year and a half from the start. Thus in the SEAC design the pressure was on immediate completion of equipment. The basis of the memory design was the mercury memory planned for the EDVAC, and changes were introduced only where they appeared significant for improved reliability or circuit uniformity.

Subsequent to the completion of the SEAC, work was started on the design of specialized equipment for the Office of the Air Comptroller, United States Department of the Air Force. This equipment is to be applied to the solution of economic equations of very high order arising from the Project SCOOP (Scientific Computation of Optimum Programs). For this application, the individual computations are relatively simple, but a very large number of them are required. The general logical design of a machine suitable for this use was worked out by 1951 *(ref. 3)* and since that time work on this program has been concentrated on the necessary input-output devices and the random-access high-speed memory needed.

Shortly after work was started on the SEAC in the Washington laboratories of the Bureau, a development program was also started in the Los Angeles laboratories. This work was aimed at the design and construction of a machine to provide computation services in its area. It resulted in the building of the SWAC.† The program is now continuing with the primary aim of the extension of that facility. I may say, in referring to SWAC how interested I was to see the Pilot ACE, for I realized from seeing it the influence his stay at NPL had on Dr. Harry Huskey, and on the initial planning of SWAC.

The subject of discussion in this paper is a very broad one. In order to keep the paper to a manageable size I have tried to duplicate as little as possible of what is in already-available reports. Thus the amount of detail in each of my descriptions is definitely not intended as any indication of the importance of the subject matter ---- it is instead merely an inverse function of the amount of information already distributed.

---

* SEAC stands for "National Bureau of Standards Eastern Automatic Computer".

† SWAC stands for National Bureau of Standards Western Automatic Computer.

ACOUSTIC MEMORIES

In designing the acoustic memory for SEAC, the physical structure of the EDVAC memory was used almost without modification. The circuitry of the EDVAC memory, however, had not yet been completed, so it appeared wise to develop circuits which would fit well with the others in the SEAC. This resulted in a complete ab initio circuit design. The results have been highly satisfying. The memory has been found to have a wide temperature tolerance, to have no difficulty with mercury contamination, and in general to give highly reliable performance. It has been in operation for just about three years now and has had only minor circuit modifications. It has been so satisfactory that a version modified only slightly is being installed in the new computer now being assembled --- the DYSEAC *(ref.4)* This new computer is being installed in a truck trailer and the ruggedness of an acoustic memory is attractive for this service. Interestingly enough, the ruggedness of this particular memory was not at all recognized at first. It was found, though, that once precautions had been taken to prevent the mercury from spilling, the line could withstand accelerations of 5g with no difficulty. DYSEAC, started late in 1952, is primarily for classified military purposes. In essence the computer is intended to handle a quite flexible connexion between the computer and its auxiliary devices - storage devices to display results, and input devices not under the computer's control. DYSEAC has certain features provided in it to make it capable of running in joint operation with similar computers so that the computers can share portions of each other's memory in carrying out work which is too large for any one of them to handle. In the interest of getting this second SEAC going rather rapidly, it is quite similar circuit-wise to SEAC, and actually the mercury memory of SEAC has been taken over almost entirely in view of the very good experiences there has been with that memory. But DYSEAC does differ quite a bit from SEAC in logic. Power has been put on its completed units, but the machine is not quite complete yet.

The general design features of acoustic memories are sufficiently familiar to make a review of them unnecessary. Instead I will concentrate on some of the less common parts of the design. In doing so I will have to be rather technical, since it is just such points that are inadequately handled by the existing general descriptions. I hope you will bear with me.

Contamination

One of the most frequently asked technical questions is about contamination of the mercury. At the time the SEAC memory was being designed there was a great deal of uncertainty about this question. Since we planned to use individual glass tanks for the mercury, we were interested in the effect of the two on each other. On the one hand it was reported that in time the mercury would leach contaminant out of the glass and form a surface film which would eventually spoil the contact of the mercury with the quartz crystal transducers or with the signal leads. On the other hand at least one chemist claimed to have seen clean mercury stored in Pyrex glass for several years with almost no visible contamination. The situation was far from clear. Many varied suggestions were offered for helping matters. On the one hand it was proposed that a strong solvent be used to leach the contaminants out of the glass walls. On the other hand it was thought that such strong measures might be unnecessary, if not actually harmful.

Luckily it was planned to operate the computer for some time under the direct supervision of its design engineers. Thus it seemed appropriate to take a modestly experimental approach to the problem, and observe the results of experience. The memory was divided into six groups of mercury tanks. The first group made up about half of the total number (28 out of the 64 tanks), and its treatment was complicated enough to give a feeling of being on the safe side. The remaining five groups (of about 8 tanks each) had varying, and simpler, treatments. The treatment of the large group was as follows:

1. Scrub the internal surfaces of the tank with a cotton swab soaked in carbon tetrachloride,

2. Immerse the entire tank in the vapor of boiling trichlorethylene for 15 minutes,

3. Air dry.

4. Soak for 4 hours in 1:3 nitric acid at 70°C,

5. Flush many times with distilled water, and

6. Bake at 200°C for 2 hours.

Perhaps the simplest of the treatments was the following:

1. Scrub with carbon tetrachloride as above,

2. Air dry,

3. Rinse with distilled water three times, and

4. Air dry.

Unfortunately this rather elaborate experiment must be reported as a failure ---- unfortunately, that is, as an experiment, but fortunately enough for SEAC performance. No difference has been observed among the different types of cleaning, since no difficulty at all has been encountered from contamination. In the three years since the memory was assembled there have been only two tanks removed from position. One was quickly removed because its length differed from the others by more than 1/10 μs delay. The other tank was unnecessarily removed before the difficulty was discovered to be a faulty plug-in connexion.

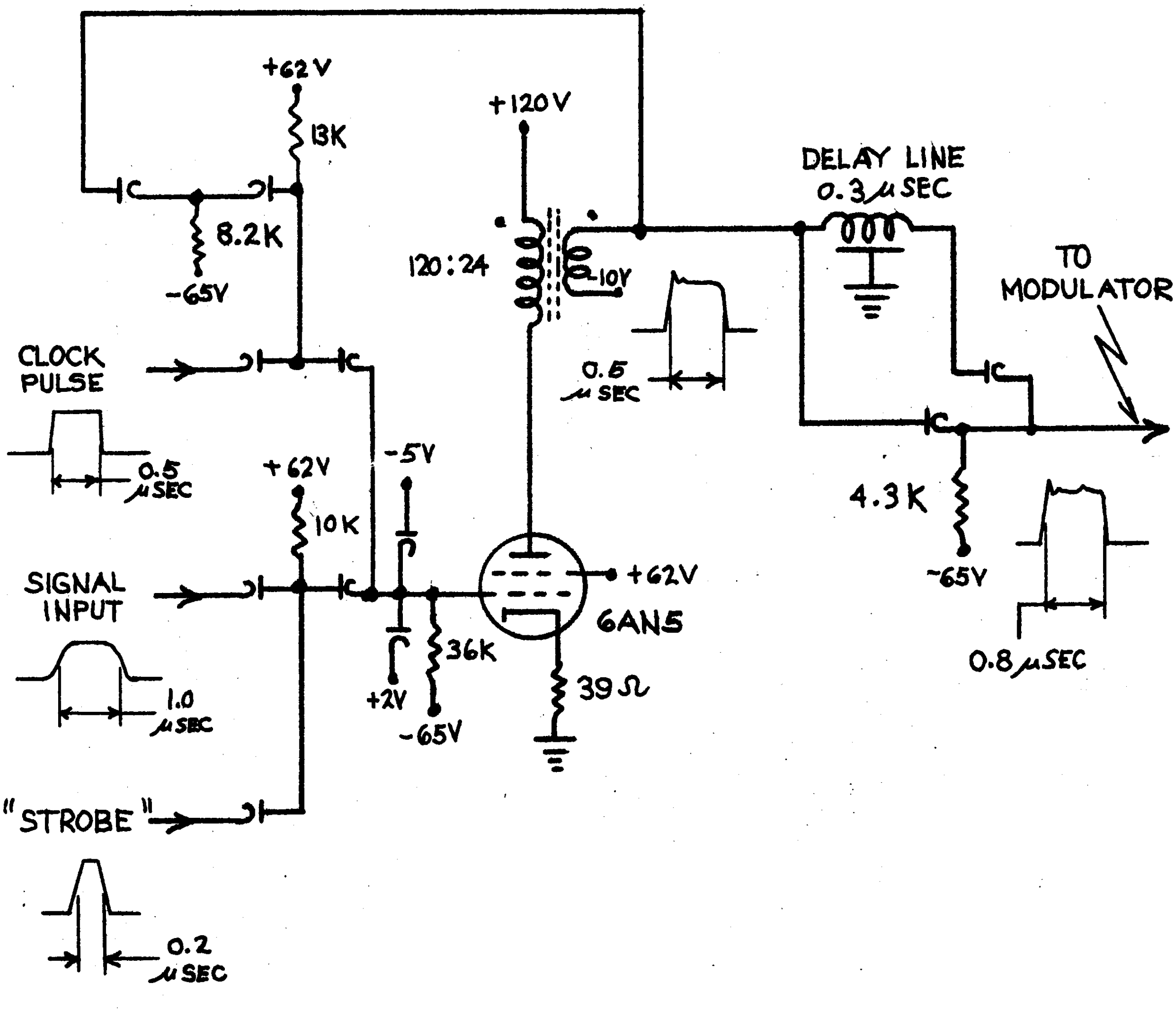
So much for contamination of the crystal surface. With respect to high-resistance signal leads, though, it must be mentioned that another design feature of the memory makes its performance very insensitive to high impedance at this point. This feature is the circuit used to prevent electrical "feed-through" on these tanks, and will be described next.

Electrical "feed-through"

Because of the individual glass tanks used for the mercury lines, a circuit difficulty is present which would be greatly reduced in installations using metal containers. In the glass tanks it is difficult to get a ground connexion of sufficiently low impedance. Tests on the circuit showed that without such a good ground, spurious electrical signals reached the receiving transducer with significant amplitudes. A ground lead of two or three inches of wire produced an electrical signal amplitude of 20% of the acoustic signal. Very slight amounts of increased contact resistance between the mercury and the ground lead increased the electrical signal until it equalled the acoustic one. Since the spurious signal is not appreciably delayed while the signal within the acoustic line is, it could produce serious interference. It was thought for a short time that there would be very serious maintenance difficulty in keeping the ground resistance low enough for satisfactory operation, but we were able to include in the design satisfactory protection from this difficulty. Instead of using the customary single-ended circuit, the driving circuit was isolated from ground by a transformer. The transformer secondary return and an associated shield provided a separate "ground" return to the mercury column, adjacent to the regular ground lead. Thus the circulating current of the driving signal is kept out of the ground path common to both driving and receiving transducers. Using this circuit on both ends of the mercury line reduced the electrical feed-through signals to a point where they were entirely negligible in comparison with the acoustic signals. No trouble has been observed from these feed-through signals, even though observations have been made with ground-lead resistances of the order of tens of ohms.

Temperature tolerance

The temperature of the mercury lines is controlled by electrical heaters; two heaters are used with separate thermostats to reduce temperature differences from one part of the block to another, and are able to keep the temperature of all of the mercury lines always within a range of about ±0.8°C. The design of the recirculation amplifier gating is such as to give a much wider temperature tolerance than this. This tolerance is achieved by using design principles introduced by Samuel Lubkin. The output of the rf amplifier and detector is a pulse of approximately 0.8 µs duration (the pulse repetition-time is one µs. This pulse is sensed by a narrow "strobe" pulse and the result triggers a transformer-coupled blocking oscillator. (The circuit is shown in *Fig. 1*.) For maximum temperature tolerance, it would be desirable to have the pulse transmitted through the acoustic line capable of proper strobing over a duration of just one µs, but the transformer-coupled blocking oscillator requires about half of each microsecond to recover. This reduces the



RECIRCULATION AMPLIFIER DETAILS
MERCURY ACOUSTIC MEMORY

FIG. 1

permissible output duration to less than desired, so the pulse width is increased by using a delay-line broadening circuit. The broadened pulse then gates an 8 Mc/s carrier, and the final stage amplifies the result for driving the acoustic line. The net effect of this procedure is to sense the incoming pulse with a strobe of the order of 0.2 µs duration, and then to generate a resultant pulse nearly one µs long. Thus temperature variations which shift the total delay of the mercury column may shift the timing of the stored pulse over a total range nearly equal to the pulse repetition time before any trouble is experienced with synchronization.

For the length of line used, 384 µs, a temperature change of about 8.8°C should shift the timing by a full microsecond. Measurements were made on one acoustic line, and gave a timing tolerance which corresponds to a total temperature range of 7.9°C or 14°F. This particular measurement of course made no allowance for length differences among the various lines, but these have been found to be small. Obviously there is no difficulty whatever in maintaining temperature within a range which is of the order of 10°F.

Self-checking

Recently it was found to be easy to add to the SEAC circuits which automatically check the correctness of memory operation. This is done in the simplest possible manner. An already-available extra pulse position in each word is used to make the total pulse count of that word even (or odd) when the word is transmitted to the memory. When each word is read out it is automatically checked to verify that the total count still fits this gross check.

This simple addition is mentioned here because it has made much more of a difference than was anticipated when it was planned. The procedure for correcting difficulties in the memory is now much different from what it was when the memory was non-checking. Formerly the difficulty might not be apparent until considerable false computation had proceeded. Now the moment that any tank in the memory becomes marginal it shows up and indicates which tank is affected. This permits prompt correction.

Quite properly the earliest electronic computers were made non-checking to keep them simple ---- it was enough of a feat to get the simplest possible assembly to operate. Now that the basic techniques are pretty well in hand, it is my opinion that much can be gained by attention to the relationship between the machine and the people who operate it. Relatively simple amounts of checking can very materially ease the burden of both the maintenance engineer and the mathematical coder and programmer. Other machine features which warrant detailed study are the order code used, automatically-scaled number representation, the organization of input-output media, and the efficient use of the still relatively-expensive high-speed memory.

Solid acoustic lines

A very small amount of work has been done on solid acoustic lines. It is mentioned here mostly as an indication of present interest.

Some two years ago a start was made toward using solid materials instead of the mercury lines. A magnesium alloy line was obtained which had delay and frequency characteristics similar to those of the SEAC lines. An interesting feature of this line was its use of reflexion of the acoustic signals. The delay per unit length of the line was less than for mercury, so the same delay required a longer line. In order to test it conveniently in the same equipment used with SEAC, the magnesium alloy line was folded: two corners at 45° were used to reflect the acoustic wave. This line was entirely acceptable although it did have significantly lower band width than is possible with mercury. The project was abandoned, however, because of difficulties in obtaining additional material with uniform characteristics.

At present the construction of the DYSEAC is stimulating our interest in rugged memories, and additional studies are being undertaken with quartz and glass lines. It now begins to appear that satisfactory supplies of these materials may become available.

ELECTROSTATIC MEMORIES

Since this subject is much more generally familiar than the others being discussed, it will be handled with less detail.

SEAC memory

With the completion of the SEAC, work was begun on a program aimed at equipment more directly designed to handle the economic calculations of the Office of the Air Comptroller. It took only a little study to show that the major engineering effort needed to lie in the fields of input-output devices and of memories. Thus active work was started therein. The SEAC itself made a good engineering tool for this work, since it had been designed with provision for the ready attachment of a variety of both input-output devices and memories *(ref.2)* In fact, from the early logical design of the machine, provision was made for operation with either a serial or a parallel memory ...... or with both simultaneously. The memory inter-connexions included both the simple serial connexion and also a complete shift register for conversion from parallel to serial or from serial to parallel. The control circuits also were designed to provide either the fully-space selection needed for a random-access memory or the partially-time selection needed for a sequential-access memory. With both types of memory simultaneously available, the control is capable of considering them as parts of a common memory, and so can refer arbitrarily to one or the other as the program might indicate.

This feature of the SEAC was used to facilitate work on a prototype electrostatic memory similar to that described by Williams and Kilburn *(ref.5)*. This memory was designed with a capacity of up to 1024 words of 45 bits each, although it has most frequently been run at a capacity of 512 words. It is fully parallel, the 45 bits of a word being stored on each of 45 cathode ray tubes, and the 512 words being obtained by having 512 storage positions in each of the tubes. The access cycle is 12 µs. Since the SEAC cannot ask for words more often than once every 48 µs, this gives time for at least three regenerations for every machine reference. Details of the construction of this prototype memory are given in other reports and so will not be repeated here.

In trial runs with the SEAC this memory has completed a total of some 400 hours of useful computation, but it has never reached the long-term reliability of the SEAC mercury memory. This appears to be caused by no inherent fault of the storage tubes themselves, but to result from the extraordinary sensitivity of the deflexion system and the signal sensing system. In comparison, the signal from the SEAC acoustic memory has something like 100 times as much voltage and at much lower impedance. Also its selection system is fully digital rather than analogue. We were, perhaps, "spoiled" by the SEAC circuitry, which has been deliberately designed for low-impedance, high-power operation. No-where does it have anything like the sensitivity of the electrostatic memory. The difficulties introduced by that sensitivity were not adequately realized in the original design, and it is now difficult to overcome the weaknesses without completely redoing the system. Nevertheless, this prototype has provided much valuable information in assessing the capabilities of such a system.

SWAC memory

An independent effort was the construction of the electrostatic memory for the SWAC at the Los Angeles branch of the Bureau. This also is a fully parallel memory, with a capacity of 256 words of 36 bits each, and an access cycle of 8 µs. At one time during the development of this memory the available cathode ray tubes had so many flaws that serious consideration was given to using the defocus-focus storage scheme. This did indeed decrease the flaw problem, but it so adversely affected the read-around-ratio that it was dropped.

This memory is now in regular operation with the SWAC.

Refill sensing

Recently R. Thorensen has suggested a new method of operating an electrostatic memory of the general Williams-Kilburn type *(ref.6)*. This consists of sensing the amplifier output at a later time than has been usual (and inverting the output in applying it to the customary control circuits). In the dot-dash method of storage the output when a dash is sensed is customarily a short positive pulse followed by a strong negative pulse. It has been customary to sense the presence of the dash from the positive pulse, but this new scheme is to sense the presence (or absence) of the subsequent negative pulse. This negative pulse is inferred to arise from secondary electron refill of potential "wells" adjacent to the beam. Thus I have called it "refill sensing".

This method has not yet been tried in a full memory system, but tests on individual tubes seem to indicate that this subsequent pulse is less affected by tube blemishes and by read-around-ratio than the customary method. That is, it is possible for the machine to make a greater number of references in the vicinity of any given spot without spoiling its information.* The tests have shown improvements in this read-around-ratio by factors of two and three times.

This question of read-around-ratio is particularly important to the American developments because these memories are being designed to be fully parallel and to work with computing circuitry having pulse rates around Mc/s. This question hardly arises when the memory is used in serial fashion, or with lower computing rates. Then it is impossible for the machine to make frequent enough memory references to be troublesome.

Tube improvement

In addition to the circuitry development work, a program is underway aimed at the improvement of the storage tube itself. In fact, most of the American laboratories have at last swung over to the viewpoint commonly expressed in England some time ago ---- that commercial cathode ray tubes have so many weaknesses for this use that significant effort is warranted toward getting special tubes. At the National Bureau of Standards, this work has been carried out largely in co-operation with commercial cathode ray tube manufacturers. Several series of tubes have been made and tested, with the aim of improving both the electron gun and the storage surface *(ref. 7,8)*. The gun can be improved in the direction of producing a beam that is narrower, a beam with sharper edges, and one having reduced deflexion defocusing. These are of course objectives in the design of commercial cathode ray tubes, but for storage work there is very little requirement for light output from the tube, so the beam can be narrowed down without worrying about the reduced light output. Laboratory tubes already received indicate that the read-around-ratio can be better by at least a factor of two than it is in commercial tubes ---- in some instances factors much higher have been observed. The electrostatic memories currently operating have maximum permissible read-around-ratios in the range of 20 to 100. Tests on the SEAC memory indicate that it should be feasible with commercial tubes to work in the range of 100 to 200, but individual special tubes have tested up to 500 to 2000. It remains to be seen, though, whether such high values would be achievable in the production of significantly large lots of tubes and in their simultaneous operation in a full computer memory. The operation of a full memory is very different from tests on a single tube.

Surprisingly, the series of tubes in which the storage surface has been studied has shown nothing significantly better than a standard P1 phosphor *prepared with great cleanliness*. It was thought that surfaces of mica, or phosphor surfaces having no binder would be significantly more uniform, but this has not been shown in the tubes tested so far. On the other hand, extreme cleanliness in preparation seems adequate to produce satisfactorily clean surfaces using the standard phosphor.

DIODE-CAPACITOR MEMORY

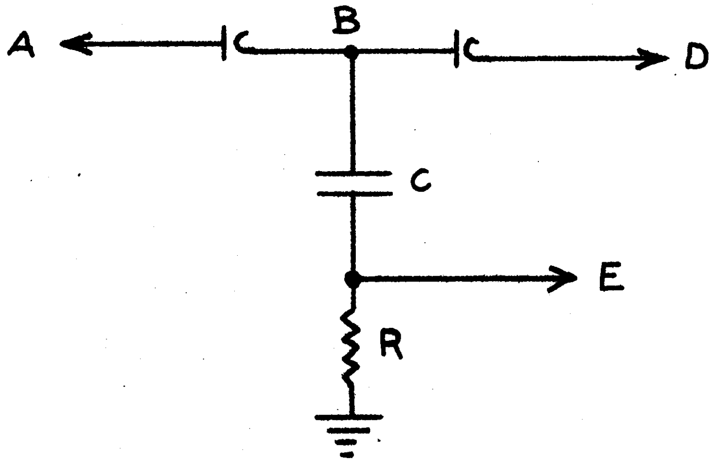
A recent development in the search for rapid-access memories is the diode-capacitor memory scheme *(ref. 9)*. This was suggested by A. W. Holt, and is being developed by a group under his direction. It is an excellent example of the inadequately-recognized situation, that the difficult part of a rapid-access memory is not the memory but the access. This scheme uses the simplest of storage devices, an ordinary capacitor and gets its importance from the efficient access scheme used.

---

* Read-around-ratio is defined as the ratio of the number of machine references to spots adjacent to a given spot, to the number of machine references to that spot. The permissible maximum for this ratio is increased as the interaction between adjacent spots is decreased.

Circuit

The basic storage element of this memory is shown in *fig.2*. The connexion E is used for both reading and writing, while the two diodes between A and D are used as a "squeezer" to connect the capacitor to the reading-writing circuits, as will be seen. During holding, both diodes are biased in their back direction. For example, A might be held at -4V with respect to ground, and D held at +4V. Then if the capacitor has a charge of, say, 2V both diodes will be biased in their back direction and only small currents will flow into or out of the capacitor. Now for reading, suppose that both points A and D are forced to ground



BASIC STORAGE ELEMENT
DIODE CAPACITOR MEMORY

FIG.2

potential ("squeezed"). This will cause one or the other diode to conduct and a voltage will appear across the resistor R. If C was charged with 2V of such polarity as to make its lower terminal (in the figure) more negative than its upper terminal, then when the squeeze occurs there will appear at E a pulse of -2V, which then dies out with the time constant RC. This would be recognized by the reading circuits at E as the binary digit "zero". If the polarity of the charge on C had been in the opposite direction, the squeeze would have produced a positive pulse instead of negative, and would be recognized as the binary digit "one". Thus the content of the storage element has been read, but in doing so it has been (at least partially) discharged and the information lost from the storage element. The information must be rewritten to continue the storage beyond the reading operation.

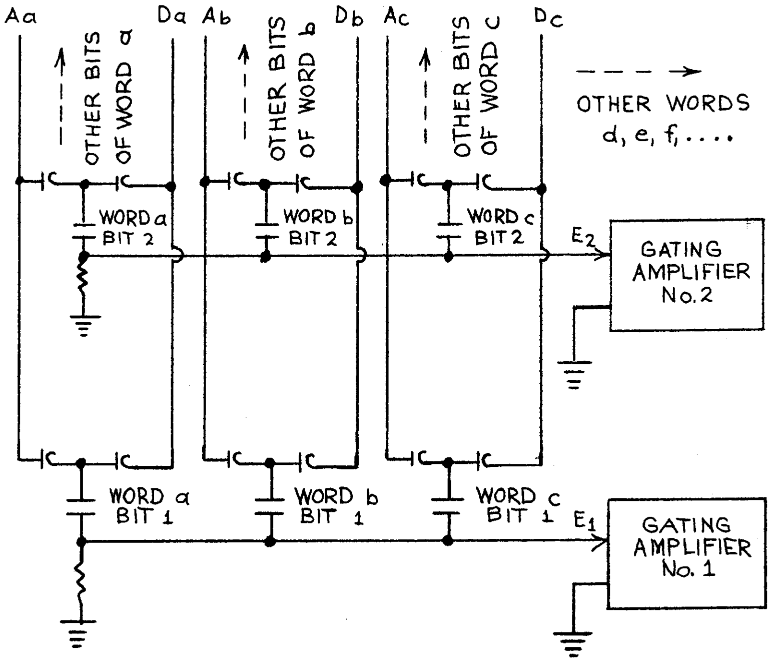
In order to write (or rewrite) information it is merely necessary to force the lead E to the desired state during the squeeze, and hold it there until the squeeze is over. Thus while A and D are at zero volts, suppose that E is forced to +2V and held there at least until A and D are returned to their normal voltages of -4 and +4 respectively. Then the capacitor is left with a charge of 2V and upon the next squeeze it will produce a positive pulse at E. That is, we have written a "one". Obviously the opposite is equally possible: forcing E negative until the end of the squeeze will write a "zero". Note that once A and D have returned to their normal voltages, the charge on the capacitor will be undisturbed by later changes of E, provided the magnitude of E's voltage never exceeds 2V. Thus E can have other pulses on it, either positive or negative, and the charge stored on C will remain unaffected because both diodes will remain with backward bias. This is important for organizing many basic storage elements into an efficient memory assembly, and is the reason for charging the capacitor to only ± 2V while biasing the diodes twice as much.

In the description so far the diodes have been implicitly assumed to be ideal, having practically infinite forward conductance and practically zero backward conductance. The effect of finite forward conductance is modest; it will reduce somewhat the output pulse amplitude, and it will determine how long a writing pulse must last to charge the capacitor adequately. The effect of finite backward conductance, however, is critical. During the holding operation relatively long times will elapse, and even minute currents through the diodes will disturb the capacitor charge. The unit would gradually leak toward a condition of no charge on the capacitor, or even a condition in which the sign of the charge is reversed. Thus the permissible duration of the holding operation is determined by the rate at which the capacitor charge leaks through the diodes' back current. Arbitrarily long storage of information is achieved through regeneration: before the capacitor charge can change to a point where there is danger of losing the information, the memory control circuits as a routine read the content of each cell and rewrite it accordingly.

Gating amplifier

What is needed at point E, then, is an amplifier which will sense the polarity of E during the early part of the squeeze period, together with a gate structure which will force E to the desired polarity during the latter part of the squeeze period. For reading or regeneration, E is forced to the same polarity that was read; for writing new information the polarity to which E is forced is independent of what was read, but is determined by the new information being written. Such a gating amplifier is easy to construct. The amplification required is very modest, since its input is a pulse whose amplitude is of the order of one or two volts. The gating can be accomplished with standard techniques. The whole thing can be done with two or three vacuum tubes and several diodes, and need not be described further here.

An interesting point is that the memory in this form permits the ready incorporation of powerful self-checking features. The input to the gating amplifier is expected to be bipolar. That is, a definite pulse should be received every time a storage element is read. This pulse may be either positive or negative, depending on the information content of the storage element, but it should not be zero. If a signal approaching zero amplitude is received it is a direct indication that the operation of that particular storage element is marginal. Thus at the expense of some complication of the gating amplifier it can be made to recognize three different input levels: acceptably positive, unacceptable, and acceptably negative. An unacceptable input need not, of course, be restricted to being very close to zero. A pulse of anything less than, say, one third of the normal amplitude might be sensed as being unacceptable. This would give a very prompt indication of incipient failure.



ASSEMBLY OF WORDS
DIODE CAPACITOR MEMORY

FIG.3

In order to achieve acceptable efficiency, it is essential that one such gating amplifier serve many basic storage elements. *Fig.3* shows how this is done. The busses A and D are made common to all of the bits of a particular computer word, and a particular gating amplifier serves the same bit on each of many words. Thus for 256 words of 40 bits each we might have 256 pairs of leads A and D, and 40 gating amplifiers. For reference to word b, the busses A_b and D_b would be squeezed to zero voltage, while all of the other pairs would be held at their normal values of -4 and +4V. In this way each gating amplifier receives a pulse from its bit of the selected word, so the word is available in parallel at the gating amplifiers. These amplifiers can then write into this word, or rewrite it, without affecting the other words, since all diodes in the other words remain with backward bias as already described. After the squeezing, busses on word b are returned to normal, any other word may be referred to in the same way. Thus we have a fully parallel, random-access memory. Regeneration is of course handled by having the memory control intersperse regeneration cycles between the computer access cycles. For the regeneration cycles, the words are read one after the other, and rewritten to their former state.

Regeneration

At present the quantitative aspects of the regeneration problem appear to be the greatest limitation on this entire memory scheme. As a rough approximation consider the following argument: the rate of discharge of the capacitor during holding is proportional to I_b, where I_b is the back current of the diode at a voltage of about 4 to 6V. Similarly, the rate of charging during writing and re-writing is proportional to I_f, where I_f, is the forward current at something like 0.5 to 1V. The safe holding time and writing time are inversely proportional to these rates, so the ratio of the permissible holding time to the writing time is just I_f/I_b. This ratio of permissible holding time to writing time indicates how many writing operations can be done before it is necessary to come back and rewrite a particular bit. Thus it is an approximate measure of the number of bits that can be served by one gating amplifier. When we include safety factors, reading time, and possible selection times, this figure comes in the range of about

$$\frac{1}{10}(I_f/I_b) \text{ to } \frac{1}{100}(I_f/I_b).$$

For actual diodes, this means that with the customary germanium whisker diodes, only some 32 to 64 bits can be served by each gating amplifier. It is of course possible to have multiple sets of gating amplifiers, but having many such sets would seriously increase the cost of the system. On the other hand, miniature selenium diodes give a much better figure of merit: it would appear possible to operate safely with 256 to 512 bits per amplifier. These diodes have much greater capacitance than the germanium, but the balanced construction of the squeeze circuit overcomes much of the difficulty caused. An ideal diode for this application is the new silicon junction diode, which has a simple thermally-diffused junction. Laboratory models of these diodes have been able to withstand rather less back voltage than the germanium or selenium, but they have a fantastic I_f/I_b ratio. The back voltage these units will stand is of the order of 20V, but that is entirely acceptable in this memory circuit. On the other hand the ratio I_f/I_b is greater even than some thermionic vacuum diodes. Only two such diodes have so far been available to us for test,* but they formed a basic storage element with writing times of a few microseconds and holding times of two to three seconds. The diodes tested appear capable of operating in a memory with 10 000 words per amplifier, with safety factors of 10 in the forward direction and 100 in the backward direction. Right now these units are rarities, but there appears good hope that they will be available in quantity and at reasonable cost in a few years.

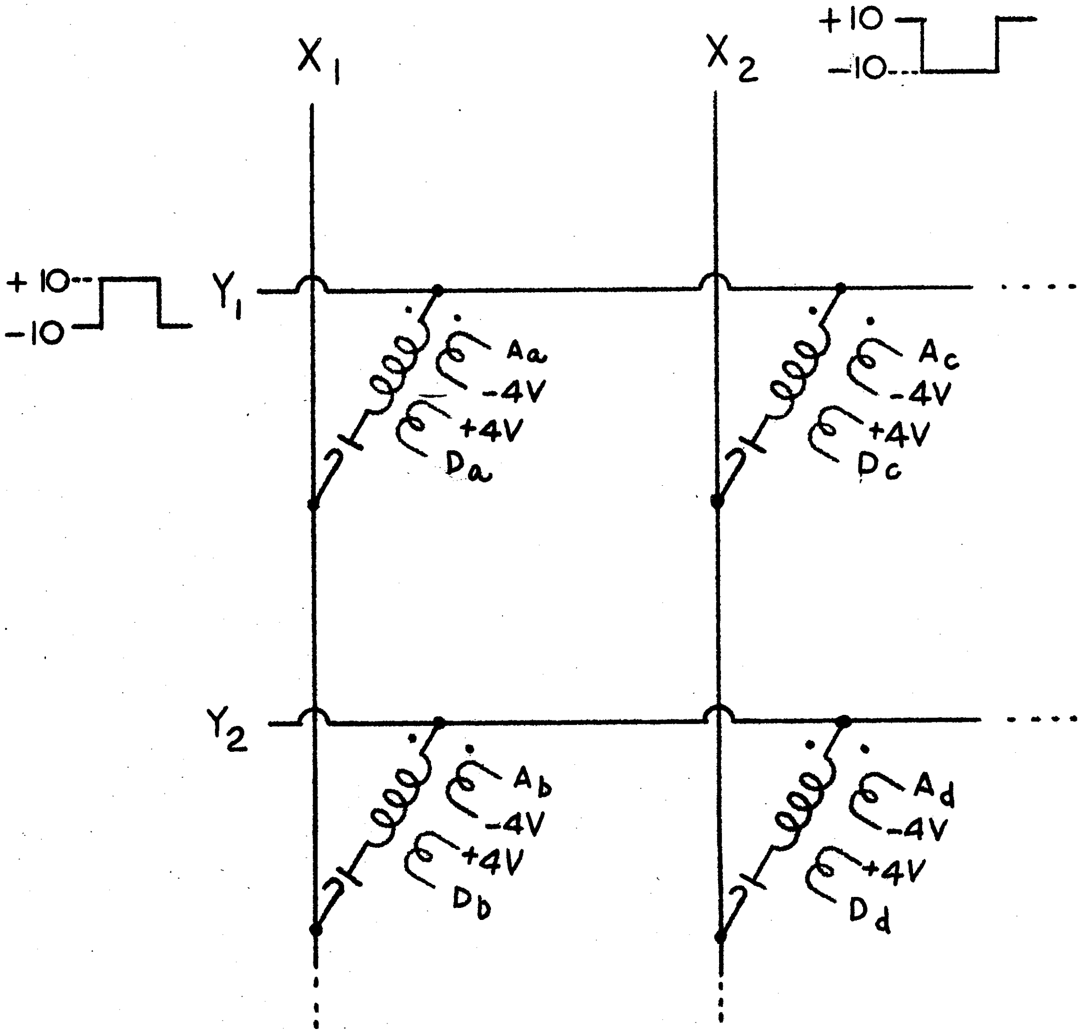
A possibility that should be mentioned for the future is the use of capacitors which exhibit strong voltage-charge hysteresis. Such capacitors could be used in this system without requiring tight limits on their characteristics. This sytem would permit much looser specifications for the capacitors than present alternative proposals for their use. Using them in this system would eliminate the I_f/I_b restriction on the number of memory elements served by each gating amplifier. The specifications for the diodes would also be greatly relaxed, but there would be no decrease in the number of diodes needed.

Selection matrix

The system described so far achieves reasonable efficiency for the gating amplifiers, but requires a selection circuit capable of squeezing the appropriate pair of busses for a particular word. This could be accomplished by the customary diode matrix, but the customary form of such a matrix has large standby currents. In this memory the squeezing busses require relatively large currents; the resultant selection matrix is feasible, but draws large amounts of standby power. To avoid this, a selection matrix using transformers and diodes is used as shown in *fig. 4*. This gives a matrix which has no standby power requirement, although it does require more input drivers than would be necessary with a multi-dimensional diode matrix. For the transformer-diode matrix, 2n inputs are required to select from among n^2 words. The matrix is made up of two sets of crossing busses (X and Y in *fig. 4*). At each crossing a diode and transformer are connected as shown. Normally all of the X busses are held at, say, +10V, and all of the Y busses are held at -10V. This puts backward bias on the diodes associated with each transformer, so no current flows through any transformer. If one X bus is dropped to -10V, still no current will flow; but if simultaneously one of the Y busses is raised to +10V, then just the one transformer at the crossing of these two busses will receive a signal. Thus if, in the figure, X_2 is lowered to -10V and Y_1 raised to +10V, the transformer secondary connected to busses A_c and D_c will squeeze the voltage on these two busses together. This will select the desired word.

---

* These diodes were loaned us through the courtesy of the Bell Telephone Laboratories.



TRANSFORMER SELECTION MATRIX.
DIODE CAPACITOR MEMORY

FIG. 4

Experimental program

After testing individually the elements of the system, a laboratory model was built containing 16 words of 4 bits each. With this model in its final form several successful lengthy tests of storage were carried out. On five occasions the unit was left running for three-day periods and was found to have the correct information at the end of that time.

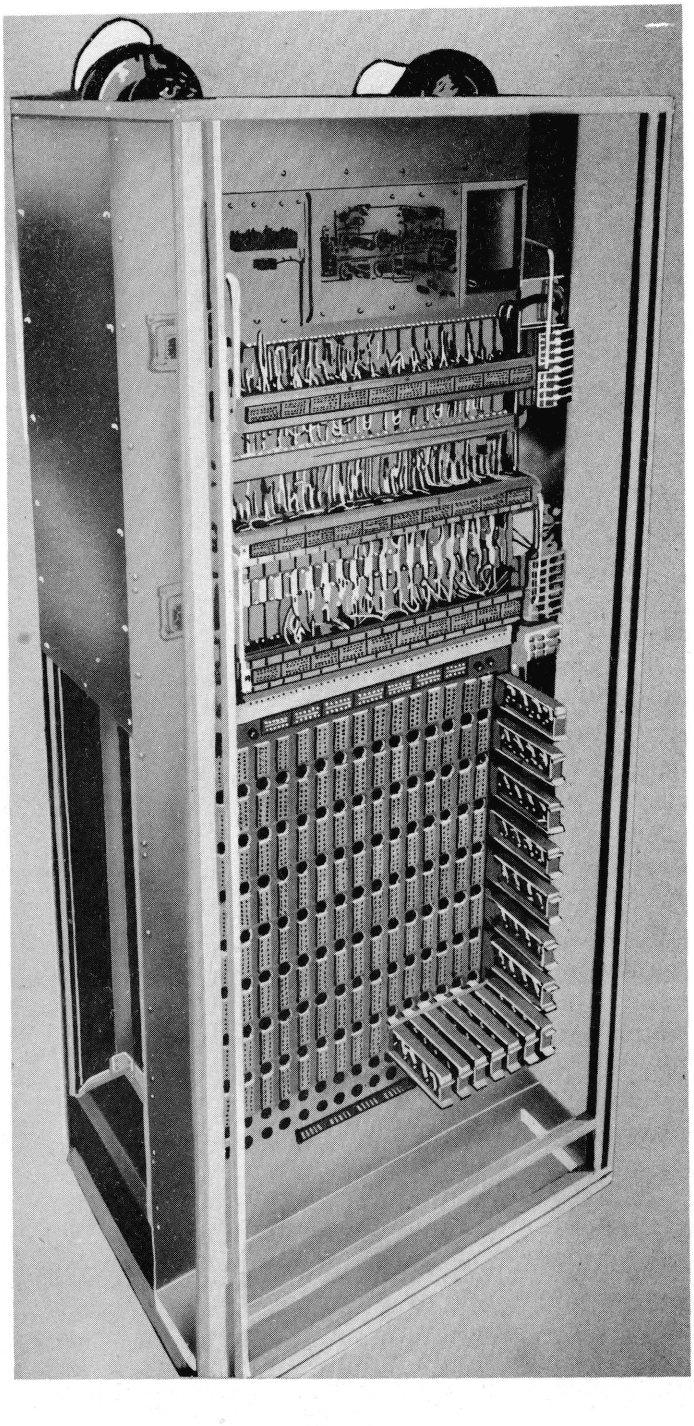


Fig. 5

Front View of Diode-Capacitor Memory Rack

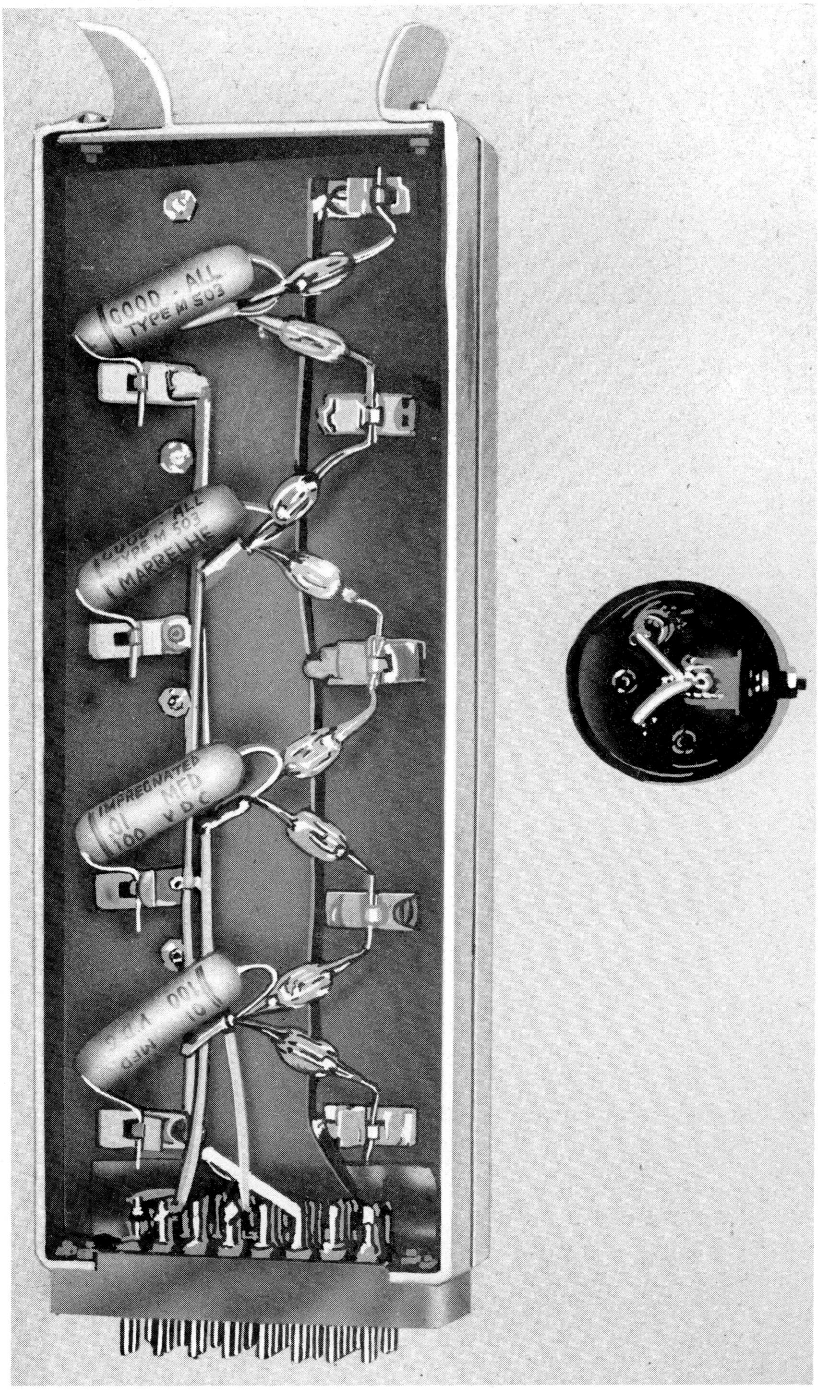


Fig. 6

8 Bit Memory Package & Transformer "AND" Gate

Results with the laboratory model have been sufficiently promising to make it desirable to test something more nearly approaching a full memory. Thus, a prototype is being built now which will be attached to the SEAC and so tested in the same way as the electrostatic memory prototype has been tested. For the diode-capacitor memory, the unit is designed for a capacity of 256 words of 45 bits each, but only 128 words of 8 bits each are being built as a start. Since the words have only 8 of the customary 45 bits, it will not be possible at first to operate the SEAC exclusively from this trial memory unit. However, since the SEAC will operate from both the acoustic and the diode-capacitor memories in integrated fashion, it is possible to do extensive testing of the new memory by using test routines stored in the acoustic memory. If all goes well with these tests, the memory will undoubtedly be expanded to a useful size.

The photographs show this unit as it is now being assembled, together with a view of the 8-bit memory package. It will be noticed that very little attention has been paid to compactness in this construction. Quite the opposite, the units have been deliberately separated to permit access to them during experimental runs. Bracketing estimates on a full-scale memory assembly indicate that a thousand words could be packaged in 20 to 50 cubic feet.

In describing the operation of the memory no mention was made of the access rate that can be achieved. That is because this is primarily limited not by the memory elements but by the external circuitry. The characteristics of the diodes in the memory unit determine the ratios that were discussed, but within wide limits the operating rate can be selected by selecting the capacitor size. This generalization becomes more limited if diodes are used which have large capacitance themselves ---- such as the selenium diodes, but for low capacitance diodes the generalization is reasonable. In the experimental equipment being built the active part of the basic cycle will be a 3-μs period during which reading and writing occur. This cycle is repeated every six microseconds, the remaining three microseconds being used for recovery of the transformers in the selection matrix.

Conclusion

The diode-capacitor memory has several advantages. One especially nice one is that there are no very weak signals or sensitive leads. The minimum signal is of the order of a volt across a few hundred ohms, and all of the selection is truly digital ---- there are no analogue voltages to be derived, and all characteristics of the materials are bounded on only one side. That is, there is no limit on the upper end of the diode characteristic: it does not have to be matched to the other diodes in the circuit. The memory is very rapid ---- access to random information is possible at well over 100000 words per second ---- and it can be very rugged in construction where this is an important attribute.

On the other hand, it has several disadvantages too. A large number of diodes are required for a large memory. A memory of 25 000 bits requires 50 000 diodes, and it still is an open question whether 50 000 diodes will give reliable operation even when the design allows them wide tolerances. Still, three years' experience with some 15 000 diodes in SEAC indicates that such an operation is not entirely out of line. Also, schemes have been worked out for rapid maintenance testing of such a large memory, a form of marginal checking which should permit replacement of drifting units before they cause trouble in computing. Yet, so many individual elements to be assembled will of necessity make for higher fabrication cost; present estimates indicate that for the same capacity the cost would be approximately twice that of a mercury acoustic memory or a Williams-type electrostatic memory (the units we have constructed indicate roughly equal cost for these latter two types).

It would appear at present as though the proper balance among cost, performance, and serviceability is something that only more experience can indicate. We hope that the prototype construction will give us the experience necessary to determine this balance.

References

1. SEAC, The National Bureau of Standards Eastern Automatic Computer; NBS *Technical News Bulletin*, 1950, 34, No.9.

2. Leiner, A. L. Provision for Expansion in the SEAC. *National Bureau of Standards Report No. 12*.3-4R, 1950.

3. Leiner, A. L. Preliminary Notes on the System Design of the STATAC-SCOOP Computer, *National Bureau of Standards Report* No. 1036, 1951.

4. Leiner, A. L. System Specifications for the DYSEAC; *National Bureau of Standards Report* No. 1951, 1952.

5, Davis, W. W. The SEAC Memory using Williams' Storage; *National Bureau of Standards Report* No. 12.3-7R, 1951.

6. Thorensen, R. An Improved Cathode Ray Tube Storage System; *Proceedings of the Joint AIEE-IRE-ACM Computer Conference*, Los Angeles, Feb., 1953.

7. First Quarterly Progress Report of the Research Program for Improving Cathode Ray Storage for Computers; *National Bureau of Standards Report* No. 2052, 1952.

8. Second Quarterly Progress Report of the Research Program for Improving Cathode Ray Storage for Computers; *National Bureau of Standards Report* No. 2263, 1953.

9. Holt, A. W. An Experimental Rapid Access Memory Using Diodes and Capacitors; *Proceedings of the Association for Computing Machinery*, Toronto Meeting, Sept., 1952.

(In presenting his paper, Dr. Slutz added to it considerably. The following is an almost verbatim report of the additional matter.)

THE STATAC SCOOP

A third computer was actually started before the second. It was intended to meet the requirements of the Office of the Air Comptroller, which demand, mathematically, various manipulations on large matrices. The formulation of the Office's economic equations require the solution of many large underdetermined sets of linear inequalities. To give you some idea of the size of the problem we have already done problems for this work in which we handled simultaneously 36 sets of 80 x 80 matrices - that is 80 equations in 80 unknowns. The Air Comptroller has used SEAC regularly on this work, and is now using a UNIVAC, installed at the Pentagon. Our third machine is intended to extend this work, and to handle 36 or more 2000 x 2000 matrices simultaneously. Consequently this is a problem which entails tremendous input and output. The actual computations are relatively simple - this is very nearly a data handling problem - and I agree firmly with Dr. Bowden when he says, for many such problems, that if you speeded up the rate of computation you would still hardly affect the total output. This was true when the SEAC handled this 80 x 80 x 36 problem because about 96% of the time was then used in feeding in successive values from magnetic tapes and getting partial answers back on magnetic tapes, and only 4% of the machine time was being spent doing any computation inside. Consequently, in our plans for this next STATAC SCOOP machine, we have put considerable emphasis on achieving a more satisfactory use of machine time. We believe this balance definitely requires a sizeable improvement in access to the high-speed memories of the machine, so that we have been working to that end. I will digress and describe the general input-output scheme proposed for the STATAC SCOOP.

We have devised what we call a concurrent input-output scheme, which means that more than one operation is taking place inside the machine at one time. For a single channel you can think of it largely in this fashion. If you consider the entire memory of the machine, a typical input order might be to fill a certain portion of this with data, and to bring new data into another portion of it. Now if you know you are not going to operate with this other portion of the memory until all the new data is in, you have a very simple situation. You can let the rest of the computer keep on going, quite independently of the input process, and in fact it is quite easy to arrange this. To prevent rather stringent timing requirements for the coders we have felt that they should not have to allow time for the new data to get in. For optimum efficiency, and optimum timing, a coder may wish to know how long it is going to take and to code accordingly, but we thought it was essential that the machine should not make an error if he comes back and starts referring to some of these words in the computation program before they actually arrive in the machine. So for this concurrent control we have set up interlocks in the circuitry such that the computer can continue to compute as long as it is referring anywhere in the memory other than to that portion which has not yet been filled. If for instance the computer tries to refer to a word which has not yet been stored, the machine then will block out until that particular word has been received, and then will carry on with its program. Though this does increase input speed, the best that can be achieved with a single concurrent input arrangement is just a 2 to 1 saving on problem solution time, and that only occurs when the solution time and the input time are exactly equal. We therefore extended the system to include not just one channel of concurrent control but sixteen such channels, so that if one channel is reading into one portion of the machine another channel may be reading into another portion of the memory and the same interlock system is applied. To make this possible for sixteen channels so that the coder cannot cause the machine to make errors we have provided additional interlocks to prevent overlapping of the channels. This, by the way, is described at length in the NBS report of June, 1951 by A. L. Rheiners. With this kind of input you need something that is rather faster in access than the serial delay lines, so in preparation for the actual construction of this machine, we have been working with memories of very short access time. We made tentative plans for an electrostatic memory of the Williams type with a cycle of about 3 - 4 µs. As described above, we have assembled a prototype of that with the SEAC and have been experimenting with it quite extensively. Also in parallel with that we have been working on the diode capacitor memory (DICAP) also described above.

SELF-CORRECTING CODES

I will now go a little further into the problems associated with self-correcting codes. Mr. Davies dismisses them rather shortly in his paper with the statement that their use in a system adds much complexity to it. But the added complexity may be relatively small.

Mr. Hamming of the Bell Telephone Laboratories has done some very significant work on error correcting codes of maximum possible efficiency. He shows that an error correcting code for each word in a 48-digit-word machine would only need another 6 check digits per word to permit the detection and correction of any one error in the 54 digits. Seven check digits suffice to detect and correct one error, or to detect and halt the machine for any two errors in the 55 digits.

If you assume that errors will be less likely than one in each word, you can construct an even more efficient code. A total of ten check digits is all you need to detect and correct one, or detect, without correcting, two errors in an assembly of 8 words, provided you assume that not more than two errors will ever occur in this assembly.

However I feel that the mathematical beauty of Hamming's work has hindered the practical application of self-correcting codes. Those codes which are mathematically most efficient turn out to be quite inefficient equipment-wise, and the complexity of the check for the correcting process is very great.

I would like to mention some thoughts we have had about the use of self-correcting codes in machine memories. The engineering design of these should be in accord with two principles, both of which seem trivial when expressed, but are usually overlooked. The first is the need to fit the code to the equipment, and the second is the need to balance the efficiencies of the mathematical checking process, of the equipment and of the human operators.

Hamming's typical highly efficient codes are based on the assumption that there will be a limited number of errors in a given number of digits and that an error will consist of an incorrect reading of a digit. This is true of mercury delay lines, but not of output printing, nor of pickup from single channel magnetic tapes. In these the most typical error is not an incorrect reading of a digit but the complete missing of a digit. Hamming's type of code cannot handle such errors, because the count goes wrong, and the codes will not tell which digit is missing. This shows the need to fit the code to the equipment.

The second principle - that of balancing mathematical equipment and human efficiencies - leads to the use of codes which, though mathematically not so pretty as those due to Hamming, are very simple and require very little equipment. As an example, if you have an electrostatic store of 48-digit words a simple system which permits you to have full correctibility for a single error in those 48 digits is to group them electrically into a 6 x 8 rectangle, and to carry out odd-even checks on the sum of each row and column. This needs 14 extra digits, an increase of 30%, but calls for very simple correcting equipment. The system observes about 40% of all double errors, but some give false information and are not caught.

The application of these two principles to a mercury memory leads to interesting possibilities. Suppse we wish to check a series of delay line tanks by a self-correcting code. The most typical type of error is not one in which a digit is omitted, but one where a digit is read incorrectly. Usually a whole tank is affected and several digits are wrong simultaneously. A really reliable self-correcting code for such a mercury memory can be constructed by adding to each tank enough check digits to permit you to recognize the correctness of that tank to the required degree of reliability. If you assume that only one error occurs at a time one check digit will suffice. To allow for two or three errors you need two or three check digits. You can attempt this on every tank - making them extra long - so that you can sense errors in individual tanks. Or you can have just one self-correcting tank, just like all the others, in a complete array. You arrange that this carries the sum modulo 2, i.e. the digit-by-digit sum of all the others. You maintain this in operation as follows. Feed the outputs of the output access circuitry and of the special tank into a serial subtractor, then add this to the output of the input access circuitry by a serial adder, finally passing the result back to the check tank. Every time you feed a new word into the memory, the old one is subtracted and the new added to the check tank, so that the check tank always contains the sum of all the others. A simple operation inside the machine to sense the presence of an incorrect tank permits you to read the check tank and then to subtract from this reading all the other tanks except the one that went wrong. The result is the word which should be in the bad tank.

There are other similar schemes you can apply to other systems. One has been worked out for application to electrostatic memories. You might have two such memories, and have the choice of the full capacity, just error checked, or of operating with half capacity but in a fully self-correcting way. Such schemes applied to electrostatic memories allow many of the tubes in the system to fail completely, even to be pulled out of their sockets, without the machine knowing that anything has happened.

This is the sort of thing you may need in machines designed for control applications, such as the control of chemical processes. The machine simply must not fail in the middle of a chemical process.

Discussion

MR. TOOTHILL (Military College of Science, Shrivenham) said that they had tried a capacitor memory similar to that described by Dr. Slutz but had concluded that the need to use large capacitors with large charging current made it too expensive.

DR. SLUTZ, in reply, said that the diode transformer matrix he described had been developed particularly to avoid the need for large standing currents. Printed circuit units had also been developed to reduce the cost.

MR. COOKE-YARBOROUGH (Atomic Energy Research Establishment, Harwell) was interested to note that Dr. Slutz was favouring selenium diodes instead of germanium. His own experience had led to the opposite conclusion; selenium tended to develop a high forward resistance.

DR. SLUTZ replied that they had not decided about selenium but were considering it. The important characteristic is the ratio of the resistances, rather than the actual forward resistance, and this is larger in selenium. A preventive maintenance procedure on SEAC led to the replacement of about 25% of the total number of germanium diodes each year.

SERVICING AND MAINTENANCE

Chairman: Mr. F. M. Colebrook

30. Preventive or Curative Maintenance

by

E. A. Newman

National Physical Laboratory

An electronic computer differs from most electronic equipment in the complexity of its organization and the simplicity of its basic circuitry. To trace a fault to a particular functional unit will usually be difficult, but the pin-pointing of it, once this has been done, relatively easy. For this reason a maintenance procedure which is effective for other electronic equipment may be unsuitable for computers. We must bear this in mind when comparing maintenance techniques. There are many possible ways of servicing the usual kind of equipment, where the complexity, if any, lies in the circuitry, and even more when the equipment is organically complex, as in the computer. It takes a long time to assess the value of any method, and since no computer of the modern kind has so far been in full use for very long, no final conclusions can yet be drawn.

We cannot compare the merits of different maintenance techniques until we have some criterion by which their effectiveness can be assessed. This must clearly be related to the normal use of the computer. This can be illustrated by reference to other kinds of mechanism. For example, it is essential that an aircraft should never fail in flight, even if it has to be grounded most of the time to ensure this. On the other hand, a production machine tool will be judged by the total quantity of its good output. A computer, as normally used, is to be compared with the machine tool rather than the aircraft, for it is usually its total output that matters - though there may be exceptional instances where the other criterion applies, as for example in the forecasting of the result of the American Presidential election. Even neglecting such special cases, however, the criterion will still be complex, for at least three separate factors must be considered,

1. the reliability of the computer, which may be defined as the time required for a given computation with faultless operation, divided by the time actually taken, including all servicing and repair time;

2. the size and grading of the servicing team;

3. the cost rate of replacements.

The relative weightings of these factors is very difficult to judge. A small increase in reliability in a very fast machine is worth more than a similar increase in a slow one, and would justify a correspondingly larger servicing team.

In either case, the economic value of the extra output must be balanced against the cost of the extra labour. Similarly, if the replacement of all valves at regular intervals made it possible to reduce the servicing team by one, the cost of the probable unexpired life of the valves would have to be balanced against the saving of labour cost.

Two main kinds of maintenance procedure can be distinguished: preventive maintenance, in which incipient faults are located and prevented before they occur; and curative maintenance, in which faults are located and cured as they occur. Obviously, the maintenance of aircraft must be preventive. That of computers need not be, and in practice existing schemes of maintenance are likely to be variously compounded of preventive and curative processes.

Advantages of curative maintenance are:

1. the machine is unavailable only when faulty;
2. an actual fault will, in general, be easier to locate than an incipient fault;
3. it should minimize replacement costs.

On the face of it, these advantages seem so considerable as to be decisive; but there are two counter considerations.

In the first place, computers, like human beings, can be off colour without being definitely ill. That is to say, because of their organic complexity, they do not necessarily move sharply from a state in which they make no mistake into one in which they do nothing right. More often they drift into a condition in which they will do some programmes correctly and make errors in others. With a machine in this state it can sometimes take a very long time indeed to locate the defective functional unit, even though the fault, when pin-pointed, proves to be a very definite electronic defect. It is not always thus, for many faults develop sharply and are quickly found; but it happens often enough to make a case for some degree of preventive maintenance.

The second and even less obvious consideration which affects the balance between preventive and curative maintenance is the quite disproportionate effect of even a small breakdown on the total output. This is due to the dislocation of the operator's routine caused by the breakdown. Most operators can get far more out of a faultless 8 hour run than out of the same total working time broken by faults into a number of shorter periods.

But even allowing for these two considerations, preventive maintenance will still fall short of justification if it absorbs far more servicing man-hours than curative maintenance. To be really effective, preventive maintenance must pre-detect nearly all potential faults - or at least nearly all the troublesome ones. This calls for the inspection of every part of the machine several times in the average interval between the faults that would otherwise occur.

Two main preventive procedures can be distinguished:

1. The whole machine is given a thorough and periodic electronic check - either as a whole in a single major operation or by parts according to a regular rota. Neither way is obviously better than the other, and as far as we know at the NPL no systematic comparison by trial has ever been carried out.

   It is clear that this kind of procedure, if really thorough, will take a long time, and it is open to question whether it will save as much time as it wastes. The process can, of course, be shortened by limiting its scope. If, for example, most faults are due to valve failure, all the valves can be taken out and checked at intervals. But, unfortunately, this handling might itself be a cause of faults that would not otherwise have occurred. Operational evidence on this point is inconclusive.

2. The alternative preventive maintenance procedure is known as marginal checking. The idea is that by varying certain suitable operating conditions, grid or anode voltages, heater currents, or even frequency, any circuits which have incipient faults can be made to fail while those free from such faults, and therefore more tolerant of the marginal check, will continue to function. The method is not applicable to all kinds of circuit, but where it can be used it would seem to have great advantages. Programmes can readily be devised to locate a functional unit in which there is a definite fault. Such a programme, in conjunction with a suitable marginal adjustment, should therefore locate an incipient fault. The precision of such fault location can, of course, be increased by reducing the size of the sections of the machine to which the marginal variation is applied.

> The method has the further advantage that it does not involve any mechanical disturbance of the machine. It may also have some disadvantages, and there is a need for operational evidence on it. We cannot unfortunately provide any such evidence at the NPL because the ACE Pilot Model has very little built-in marginal checking. It is being provided throughout in the DEUCE but this will not give us any valid comparative data because the DEUCE will also be an improvement on the Pilot Model in many other respects. For example, it will run much cooler. Also the Pilot Model has some good and some not so good circuits, but the DEUCE will have only those which have been found to be good.

These two machines illustrate another fairly obvious point about maintenance procedures, namely, that they must be adapted to the construction of the machine. The ACE Pilot Model is built up of replaceable plug-in units, but the individual components are not normally accessible during operation. In the DEUCE, on the other hand, the units will not be so readily replaceable, but all components will be accessible.

The replaceable unit construction has some obvious advantages. For example, if replacement of a unit clears a fault, this must be in the abstracted unit, which can be repaired away from the machine without loss of machine time. Routine preventive inspection is similarly facilitated by this type of construction, the units being examined in succession on special test set-ups, again without loss of machine time. On the other hand, this implies extensive spares and complex test gear, both expensive in initial cost and in upkeep.

There is also another and less obvious drawback. Taking full advantage of the plug-in construction implies fairly frequent substitution of units; but this continual mechanical disturbance may itself contribute materially to the incidence of faults.

Finally, plug-in units imply a multiplicity of plug and socket contacts and all the potential failures associated therewith.

This last consideration raises the whole question of the best size of detachable unit. In current practice there is a wide range in this respect, from the valve itself at one end, through "packages" consisting of one or two valves and their associated components, up to units such as those in the ACE Pilot Model which carry 20 or 30 valves and their circuitry, sufficient, say, for a complete adder.

Our experience with the ACE Pilot Model leads us to favour the larger units, provided the valves themselves are very readily detachable. We find that once a fault has been located in a functional group, it is fairly easily pin-pointed. On the other hand, we find it very desirable that the valves themselves should be plugged in. We have in fact a special range of test valves - double triodes with one or other of the triode pins removed (known locally as port or starboard johnnies because of their red and green markings) - which we find exceedingly useful.

Now, having surveyed the arguments about preventive and curative maintenance, fixed or detachable units, and so on, what conclusion can be drawn? Briefly, the conclusion that at present no general conclusion can be drawn. This is largely because of the interdependence of maintenance and design. We can decide on a maintenance technique and engineer the machine to suit, or engineer to meet some other requirement and make the maintenance procedure conform to this design. Either way, detailed and conscientious recording of performance and servicing, for prolonged periods, is the only way to get the evidence we need. We have ourselves been keeping very comprehensive fault records. The analysis of these is difficult, and there is much still to be done. It is hoped that the discussion on this paper will clear some of the fog away and that the frank interchange of experiences will lead to a clearer view of the way ahead.

Discussion

MR. PHISTER (Cambridge University Mathematical Laboratory) distinguished between *maintenance efficiency* (detection and repair of faults) and reliability, which depends on the fault rate. The figure of merit for reliability will depend on (a) the speed of computation which determines the amount of work done between faults and (b) the efficiency of *preventive* maintenance. The relative importance of these elements will depend on the use to which the machine is put. Briefly, therefore, reliability is determined by the number of useful unit operations per fault.

MR. COOKE-YARBOROUGH (Atomic Energy Research Establishment, Harwell) said that machines are switched off outside working hours for fear of break-downs and such time should therefore be counted as unserviceable.

MR. WRIGHT (NPL) said that a reliability figure should also take account of the number of useful operations lost in repair time.

The author observed that the overall measure of reliability is the ratio of the useful work actually done in a given time to the amount which would have been done by the machine if in perfect condition.

MR. R. T. CLAYDEN (English Electric Co. Ltd.), drawing a comparison with a broadcast service, advocated the duplication of computers for regular service.

MR. WILLIAMS (NPL) said that if machines were duplicated, users would call for full output from both! Maintenance efficiency is determined by total fault repair time rather than by the number of faults. In the ACE Pilot Model many faults had been due to resistor drift. Plug and socket faults had been unimportant. Test programmes are helpful but cannot easily be made to detect every fault.

DR. PINKERTON (J. Lyons & Co. Ltd.) maintained that the important thing is to find what causes breakdowns and how to prevent them. LEO is tested and inspected and readjusted as a regular routine. Suspected valves are removed and tested and if satisfactory, replaced, but not in the same socket. They are regarded as downgraded and put in circuits known to be tolerant.

MR. DAVIS (English Electric Co. Ltd.) said that intermittent faults in the ACE Pilot Model could often be located by tapping parts of the machine. Suspected valves were tested in circuits which simulated the working conditions. Routine maintenance included monthly inspection of all mechanical parts and random removal and checking of units. Check programmes were used to locate a fault in a large block, which is then more closely scrutinized by inspection of wave forms at suitable points. In addition to diagnostic programmes, problems known to be specially exacting are used to test the machine. Such problems are analysed to determine the points at which the programme is liable to cause the machine to fail.

MR. RUBACH (All Power Transformers Ltd.) recalled that wartime experience with radar equipment showed that preventive maintenance could be overdone and could cause more faults than it prevented. This may have been due to low grade maintenance personnel.

The author, summarizing, said that evidently, different groups have built up different diagnostic and maintenance techniques according to the needs of the particular machine, *e.g.* some groups favour an analytical approach whereas others use an objective method. In general results tend to be the same. The total time wasted is the product of the number of faults and the average time taken to find a fault. Preventive maintenance probably discloses more faults but enables them to be more easily detected and repaired, whereas otherwise a lesser number of more obscure faults are observed. Marginal checking might very well be a means of reducing the "fault x time" product.

31. Experience with Marginal Checking and Automatic Routining of the EDSAC

by

M. V. Wilkes, M. Phister Junr., and S. A. Barton

Cambridge University Mathematical Laboratory

Introduction

This paper contains an account of certain experimental marginal checking facilities which have recently been fitted to the EDSAC. The experiments are still in progress but it is thought that a short account of the results which have been obtained so far will be of interest.

Ideally, marginal checking should be included in the design of a computing machine from the beginning and it is not very satisfactory to add it to an existing machine. This was particularly so in the case of the EDSAC which had undergone many modifications since it was completed in 1949, with the result that both the logical design and the physical layout were not quite as straightforward as they might have been. At the same time, this situation made the provision of a system which would facilitate fault location and maintenance particularly attractive. The main object of the work, however, was to obtain experience which it was hoped would be useful in the design of future machines. The EDSAC is a serial machine and what is said in this paper relates primarily to machines of that type.

There are two main benefits which may be hoped to result from a system of marginal checking. In the first place, marginal checking should enable loss of operating margin due to deterioration of valves, or drift of component values, to be detected and rectified before becoming serious enough to cause errors under operating conditions. This should result in a material reduction in the chance of a machine failure occurring during operating hours. In the second place marginal checking should be of assistance in locating those elusive faults which occur in some programmes and not in others, or which occur apparently at random. These faults are frequently due to a particular circuit being "on the edge" and failing occasionally when the pulse pattern presented to it is a particularly unfavourable one. Such faults should be accentuated and made easier to locate when marginal checking is applied, especially if the marginal checks are arranged so that they can be applied to the various units of the machine independently. While marginal checking may be expected to assist in the location of apparently random faults of the type just described, it is unlikely to be of assistance in locating intermittent faults due to dry joints or mechanical failure of components.

Attenuation of pulses

As a first step in the fitting of marginal checking to the EDSAC it was decided to insert switchable attenuators in a number of leads carrying pulses or control waveforms. During marginal checking these attenuators, which are normally inoperative, are switched into circuit and the action of the machine tested by means of a test programme especially designed to be as searching as possible. If any of the pulses or control waveforms have become low in amplitude as a result of component deterioration, errors are likely to occur when this test is applied. It will be noted that during marginal checking the machine operates in the normal manner under the control of a programme and that no specially introduced test pulses are used.

A simple form of resistance attenuator is used and in order to avoid undue lengthening of pulse leads all attenuators are controlled by relays operated from a central panel. The circuit used is shown in *fig. 1;* it will be observed that during normal operation the relay contacts are open so that there is no danger of the marginal checking system giving rise to extra faults due to faulty relay contacts. Since the attenuators are all connected between the output of a cathode follower and the input of another cathode follower or amplifying

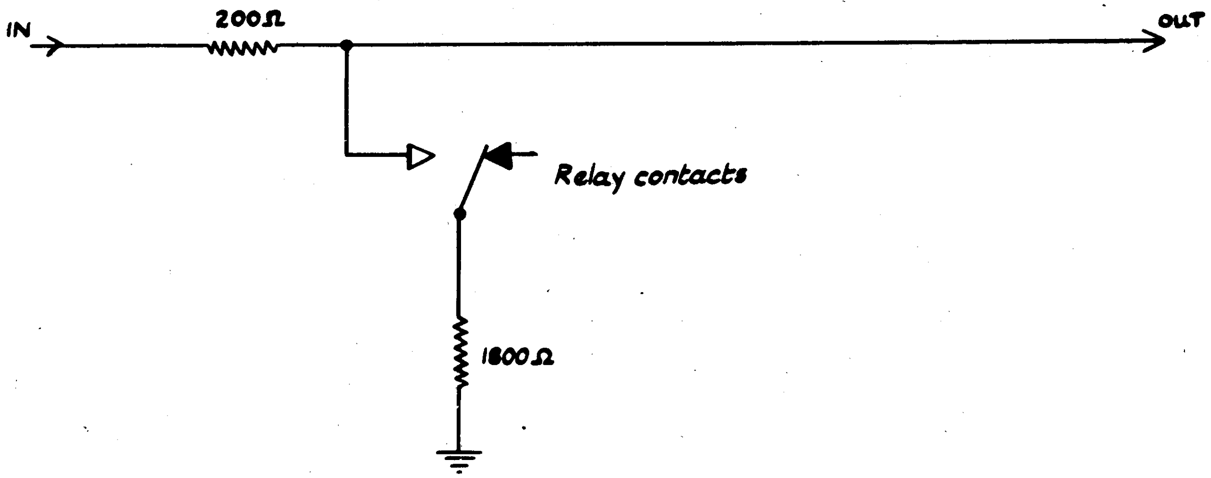


FIG. 1

valve the presence of the 200-ohm resistance in the pulse lead has no appreciable effect during normal operation. The attenuators are mounted on small plates clamped to the machine framework at convenient points.

In all, 45 attenuators were fitted, 15 in the store and associated access circuits, 11 in the arithmetic unit, and 19 in the main control. Careful consideration was given to the amount of attenuation to be introduced. It must be greater than the greatest reduction in pulse amplitude which can occur between successive marginal tests as a result of component deterioration. On the other hand, if the attenuation is too great there is a danger of misleading results being obtained by testing the machine under conditions widely different from those of normal operation. Moreover the use of too much attenuation will result in adjustments being made to the machine, and components being replaced, before this is necessary; not only is this wasteful of material but it is also wasteful of time, since the components must be located before they can be replaced. With these considerations in mind an attenuation of about 10% was decided on for the attenuators fitted to the EDSAC, although the circuits would have stood rather more than this. It was hoped that 10% attenuation would be enough to turn infrequent and apparently random errors into consistent ones. In addition to reducing amplitude the attenuators have the effect of reducing slightly the effective width of the pulses passing through them. Care was taken when choosing the points at which the attenuators were introduced to avoid any situation in which pulses passed through more than one attenuator before being amplified and regenerated. The object of this was to make it possible for the machine to be tested with all attenuators in circuit without pulses in any part of the machine being unduly attenuated.

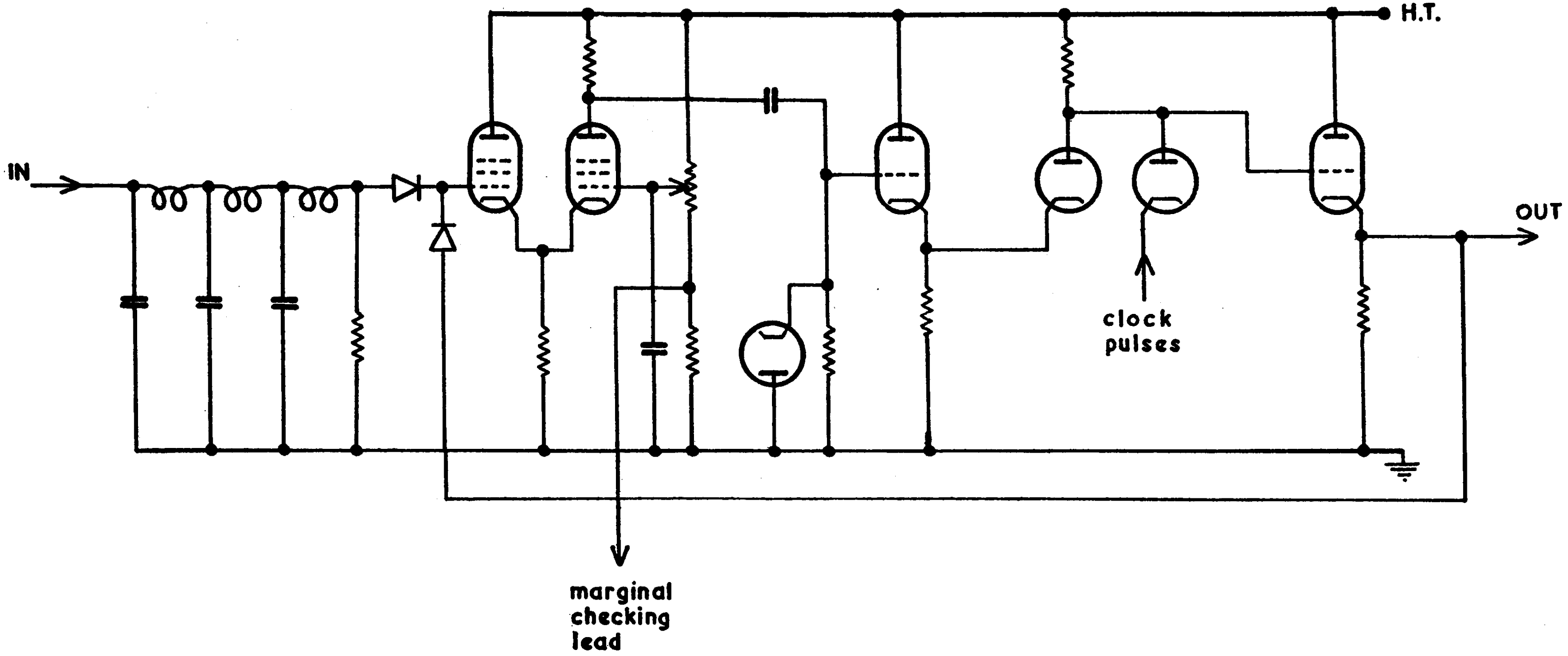
One immediate result of fitting attenuators was the discovery that some parts of the machine had hardly any operating margin at all. A number of modifications were therefore made, the last and most striking being the complete reorganisation of the clock pulse distributing system which had, up to that time, been overloaded. Only when these modifications had been made was it possible to operate the machine with all the attenuators switched into circuit. We feel that the bringing to light of unsatisfactory features in the design of a machine is not the least advantage which accrues from the introduction of marginal checking; if marginal checking is included in the design of a machine right from the beginning many weaknesses in the design will be avoided altogether.

It is always difficult to evaluate the effect of any modification made to a machine by comparing the records of maintenance before and after, since the time intervals are rarely long enough for the records to have any statistical significance. This difficulty is especially acute in the case of the EDSAC, since a number of modifications of various kinds have been made to the machine in the past year and others are in progress. The remarks which follow are based therefore, partly on general impressions. During the two years immediately before the introduction of the attenuators we had had an average of about 25 faults per month. About 9 of these were cured by adjusting amplifiers, 6 by the replacement of valves, resistors or condensers, and the rest by miscellaneous adjustments mostly to mechanical equipment. We expected that after the initial difficulties associated with the introduction of marginal checking had been overcome, the machine would settle down with about the same number of failures per month - perhaps very slightly more since components would be replaced before they had deteriorated quite as far as formerly. We hoped, however, that most, if not all, of the 15 failures a month cured by replacements and amplifier adjustments would be forestalled by marginal checking, and would not cause breakdown of the machine during operating hours.

The results were somewhat disappointing. On an average of only 3 occasions a month were replacements or adjustments made as a result of marginal tests with attenuators switched into circuit. Not more than this number of failures during running hours can, therefore, have been forestalled. These results presumably indicate that the assumption upon which marginal checking with attenuators is based - namely that the primary effect of ageing of valves and components is to cause a deterioration in pulse amplitude - is false. On consideration this conclusion does not seem unreasonable and it might perhaps have been anticipated. A change in the value of a resistor is just as likely to increase the gain of an amplifier and hence to increase the amplitude of a pulse as to decrease it, and this can lead to trouble in various ways. For example, it may cause spurious pulses which should be below a fixed threshold to exceed that threshold; alternatively, it can cause pulses to become lengthened and run into one another. Once these facts have been appreciated it will be realised that the use of attenuators for marginal checking is logically unsatisfactory since the variation introduced is one-sided. It would be more satisfactory if switchable attenuators allowing for (say) 0%, 10% and 20% attenuation were fitted and the machine run normally with 10% attenuation, marginal checking being performed first by switching to zero attenuation and then, in a second test, to 20% attenuation. However, this idea has not been followed up, since a more attractive scheme, described in the next section, presented itself.

Marginal checking of amplifier adjustments

Pulses which have become reduced in amplitude, or which are slightly late compared with the clock pulses - for example, as a result of having passed through a number of gates in cascade - are regenerated at various points in the EDSAC by means of the circuit shown in *fig. 2*. This includes a delay line which retards the incoming pulses by slightly less than 2 μs (the pulse interval), an amplifier and a gate fed with clock pulses. If the circuit is correctly adjusted, the output pulses are cleanly-gated clock pulses. It will be noticed that positive feed-back is included in the circuit so that, even if the waveform emerging from the delay line falls before the end of the clock pulse being gated, the output is nevertheless held up until the end of that pulse. The amplifier is of a non-linear variety, and the potentiometer is adjusted so that the first valve is normally cut off. In this way spurious pulses generated by internal or end reflection in the delay line are eliminated. For satisfactory operation the potentiometer must be so adjusted that there is sufficient supression of spurious pulses and, at the same time, sufficient gain in the amplifier. There is a small range of adjustment over which both these conditions are satisfied and when initially adjusted the potentiometer is set somewhere in this range. The method of marginal checking now being described is designed to verify that no drift sufficient to cause the adjustment to become marginal has taken place. It consists in applying small voltages, first positive and then negative, to the lead marked "marginal checking lead" in the diagram. This has the same effect as moving the slider of a potentiometer a few degrees, first in one direction and then in the other. The marginal checking leads from the various amplifiers were taken to three-position switches mounted on a central panel. About 80 amplifiers required treatment in the manner just described; of these 20 were in circuits identical with that shown in *fig.2*, and 60 were in similar circuits carrying control waveforms. The



PAPER 31

FIG. 2

voltage to be applied to the marginal checking leads was chosen with considerations in mind similar to those mentioned above in connexion with the choice of the amount of attenuation to be introduced by the variable attenuators. After some preliminary trials it was decided that a suitable voltage was ±1.5 volts.

The system of marginal checking just described has been most successful. Since it was installed at the end of January 1953, a number of adjustments have been made to amplifiers as a result of indications obtained during marginal checking, but no failure of the machine which could be corrected by the adjustment of an amplifier has occurred during operating hours. There is no doubt that the system has enabled substantial improvement to be made in the standard of serviceability of the machine.

The main reason for the success of marginal checking applied to the amplifiers is, no doubt, that changes occurring either in components or valves are most serious when they are directly associated with amplifiers. In addition, changes which take place in components not directly associated with amplifiers can often be compensated for by making an adjustment to an amplifier; these variations are, therefore, brought under control by marginal checking applied to the amplifiers. It would seem to be a sound practice, whenever an adjustable control is provided in a circuit, to provide also a means of marginal checking by which it can be ascertained that the control is set well in the middle of its range of satisfactory operation. The marginal checking switch should, if possible, have exactly the same effect on the circuit as moving slightly the preset control, first in one direction and then in the other. The existence of a large number of preset controls is a weakness in a machine which has no system of marginal checking but, if marginal checking along the lines just indicated is provided, it becomes, on the contrary, a source of strength.

At the present time both pulse attenuators and marginal checking switches for the amplifiers are fitted to the EDSAC. Some of the attenuators have been rendered superfluous by fitting marginal checking to the amplifiers and these will probably be removed. Others - such as those in leads carrying pulses which set or reset flip-flops - still serve a useful purpose and will be retained.

Marginal testing of the store

Mercury tanks are used in the high-speed store of the EDSAC and each tank has associated with it an amplifier and clipping circuit which must be set up by adjusting a potentiometer. Marginal checking has been applied to this adjustment in a manner similar to that described in the last section except that, instead of switching positive and negative D.C. voltages in turn, a 50-cycle A.C. voltage is applied. This means that the test can be carried out in one operation instead of two. Apart from its use in routine marginal checking, the method is particularly convenient when the amplifiers are being set up with the help of an oscilloscope, since the effect of the marginal variation up and down is clearly visible on the screen.

The application of a low-frequency alternating voltage for marginal checking purposes can only be adopted where there is no possibility of alternating voltages spreading to parts of the machine other than those being tested. The circuit used in the EDSAC store has an A.C. coupling with a short time-constant immediately after the point at which the alternating voltage is introduced and this effectively blocks the 50-cycle signal. 50-cycle A.C. has also been used in Cambridge for marginal checking in a piece of equipment which is D.C. coupled throughout; here the 50-cycle signals are applied to the grids of valves which, in normal operation, are sufficiently cut off to prevent any current flowing in their anode circuits. The use of low-frequency A.C. is not to be recommended for testing circuits which carry pulses only rarely, since the testing voltage is at its peak for only a small fraction of each cycle and, unless a very long time is allowed for the test, there is a danger of the circuit not being adequately tested. Where the method is applicable, however, there is no doubt that it is highly convenient.

Marginal checking was fitted to the store amplifiers in March 1952. Shortly before this, however, the store had undergone extensive modification which had improved its general performance. It is, therefore, difficult to assess the value of the marginal checks and there is some difference of opinion in the laboratory. Some people think that the use of marginal checks during setting up and occasionally during routine maintenance has contributed substantially to the trouble-free operation of the store; others think that a comparable performance would have been obtained without it. It is true that few adjustments have been carried out as a result of indications provided by marginal checking but this may merely indicate that marginal checking enables an optimum adjustment to be achieved in the first place.

Automatic routining

When marginal checking is fitted to a machine, routine checking becomes a matter of some complexity, since various test programmes have to be run with, and without, margins. Shortly after the attenuators were fitted it was decided to make the entire process automatic. This was done by controlling the attenuator relays from a uniselector (rotary stepping switch) which could be stepped from one position to the next under the control of the programme. All that was necessary was to put the correct input tape in the photoelectric tape reader and to start the machine. The various test programmes were then carried out, with or without margins, according to the wiring of the uniselector. The system has recently been modified so as to include marginal checking of the amplifiers.

Routine testing procedure

The complete testing procedure for the machine is now as follows. First the store is tested and the clock pulse frequency adjusted to the centre of its range. The tape reader is then tested, use being made of a marginal check which consists in switching a resistance in series with the lamp so that the amount of light falling on the photocells is reduced. These tests are carried out manually using suitable test programmes. Next, the automatic routining equipment is switched on and a comprehensive test lasting about six minutes is carried out automatically. All machine orders are tested and two different tests are applied to the entire store. Each test is carried out three times; first without any margins, secondly with the attenuators switched in and a positive voltage applied to the marginal checking leads of all the amplifiers, and finally with the attenuators switched in and a negative voltage applied to all the amplifiers. After each test, symbols are printed identifying the test and indicating whether an error occurred. When the test is finished the machine stops and the operator examines the record. If there has been a fault the operator can endeavour to locate it by stepping the uniselector to positions, not used in the routine tests, in which marginal checking is applied only to certain sections of the machine. If necessary he may then make use of the switches controlling the attenuators and amplifiers individually. This feature of the marginal checking system fitted to the EDSAC is of great value. Sometimes faults which the system would not, at first sight, be expected to cover can be successfully located. This is because a fault in a particular part of the machine is likely to be sensitive to *any* change in the operating conditions in that part of the machine. Even a very slight change in the behaviour of the machine when a marginal switch is operated may be sufficient to suggest where the trouble is to be found.

Small faults which show up during testing of the machine under marginal conditions are corrected if this can be done without the expenditure of a great deal of time. Otherwise the fault is tolerated. The reason for this policy is that one of the main objects of testing the machine under marginal conditions is to reduce the time taken to locate the elusive faults that occur under normal operating conditions by turning them into consistent ones. This object would be defeated if a corresponding number of elusive faults occurring under marginal conditions had still to be located. Needless to say a rare fault which occurs under marginal conditions is carefully noted and if it persists for any length of time efforts are made to locate it.

The automatic routining system has proved very successful. It is quicker and much less trouble to use the automatic routiner than to do the same tests by hand. Exactly the same tests are carried out on every occasion and a printed record of the results is provided. There is no danger of tests which are found to show an error very rarely being gradually dropped without anyone being conscious of it. Further, the automatic routiner can be operated by an unskilled person who need not have a detailed knowledge of the testing procedures in use. Although the automatic routiner takes care of the normal daily testing of the machine, it is advantageous to run other test programmes from time to time, since it is found that one programme will bring to light faults missed by others; the greater the variety of test programmes in use the higher the standard of serviceability which can be maintained.

Conclusions

We are at present still feeling our way towards a full understanding of the factors affecting the occurrence of faults in an electronic computing machine. Until this has been obtained it will not be possible to lay down the logical principles on which a system of maintenance should be based. We will attempt to summarize in this section the general conclusions at which we have arrived so far.

In evaluating any system of marginal checking it is important to keep clear the two fundamentally distinct benefits which such a system might give; these are, (1) a reduction in the number of faults which occur in scheduled operating time, and (2) more rapid location of such faults as do occur. Our experience shows that a reduction in the incidence during operating hours of faults due to the steady deterioration of valves, or to changes in the values of components, can certainly be achieved by an efficient system of marginal checking. This is of great value, since the majority, although not an overwhelming majority, of faults in the EDSAC are of this character. There are also a large number of miscellaneous faults, mostly of a mechanical nature - dry joints, bad contacts, faulty valve holders, and such like. Individually, one is inclined to put these down to bad luck and to say that they are unlikely to recur. However, we find that fresh faults of the same kind appear at a steady rate; in a particular month they may even account for half the total number of faults occuring. The number of these faults is being gradually reduced as a result of improvements made to the machine and in a new machine it would undoubtedly be less than in the EDSAC. However, we do not think it will ever be an entirely negligible factor governing machine serviceability.

It is sometimes supposed that when valves approach the end of their useful lives they begin to deteriorate at an increasing rate. If this were so valves about to fail would be located by marginal checking. Our experience, however, is that deterioration continues slowly throughout the life of the valve until a sudden failure occurs, usually by the valve becoming soft or losing its emission completely. Marginal checking serves to control the steady deterioration but it does not give warning of when the valve is about to fail. The behaviour we observe may be accentuated by the fact that many valves in the EDSAC are used in circuits which have a good deal of negative feed-back so that the effects of cathode deterioration are masked. Marginal checking of the amplifier adjustments may also be regarded as providing negative feed-back, the maintenance engineer forming part of the feed-back loop.

A system of marginal checking intended to help in fault location must cover the whole machine with few gaps. Given this we are in no doubt that marginal checking reduces the amount of time taken to locate a fault and makes the accurate diagnosis of its cause more certain. The fault can then be cured once for all. If the diagnosis is not accurate, there is a danger that the machine may be made to work by adjusting an amplifier, or by replacing a component, either of which is only remotely connected with the faulty circuit; this will not give a permanent cure and sooner or later the true cause of the fault will have to be found. Thus one fault may be responsible for several breakdowns of the machine.

Much more extended trials will be necessary before it is possible to give any numerical estimate of the extent to which a system of marginal checking can lead to improved serviceability of a machine. All we can say is that our experience so far has confirmed us in our original optimism. A positive recommendation we can make is that whenever an adjustable preset control is provided in a machine there should also be provided a means of marginal checking by which it may be verified that the control is set well in the middle of its working range.

Discussion

DR. PINKERTON (J. Lyons & Co. Ltd.) said he had largely followed EDSAC practice in marginal testing, but had sought to make the procedure quantitative, *e.g.* a known A.C. bias voltage is applied to the amplifiers and the tolerated range is recorded and taken as a measure of their state. Marginal test adjustments have also been fitted to the store, not only on pulse amplitude but also on pulse-repetition frequency and the tolerated frequency range is noted for each delay line. It has also been applied to the photoelectric reader as a ± 5% variation of illumination.

DR. COOMBS (Post Office Research Station, London) asked whether a gradual reduction of valve heater voltage on a machine would give useful indication of "edginess" of circuits arising, possibly, from drift of resistor values.

MR. NEWMAN (NPL) strongly supported marginal testing, its great advantage being that it could be applied without any mechanical disturbance of the equipment. It had only been used in a limited way on the ACE Pilot Model but was being incorporated wherever possible in the DEUCE.

MR. DAVIS (English Electric Co. Ltd.) asked whether marginal testing had proved useful in pin-pointing the location of a fault.

MR. PHISTER said that at Cambridge, marginal testing had proved as valuable in pin-pointing the location of a fault as in revealing incipient faults. He questioned the value of DR. PINKERTON'S quantitative procedure. It took longer, and quantitative values would not be characteristic of the circuit but only of the particular components.

DR. PINKERTON agreed that the measurement and recording of margins took longer but thought this was justified by the additional information. An item which was observed to have a less-than-normal tolerance range on test could be noted for early detailed inspection.

MR. WRIGHT (NPL) said that the Pilot ACE circuits were designed for definite tolerances of component values and operating voltages and these theoretical ranges can be checked by the marginal test procedure.

DR. GRIMSDALE (Manchester University) said that the Manchester machine had no built-in marginal tests, but the facility for slow application of the heater voltage was used as an overall marginal test by running with reduced heater voltage.

32. Diagnostic Programmes

by

R. L. Grimsdale

Ferranti Ltd.

Introduction

Diagnostic programmes or test programmes are an essential part of the servicing scheme of a large electronic computing machine.

The paper deals with some of the general aspects of these programmes, Details of the test programmes designed for the Manchester machine are given in the Appendix.

Types of fault

Faults can be classified in two groups, firstly according to their outward effect or appearance as

1. Engineering faults.
2. Logical faults.

and secondly according to duration as

1. Permanent faults.
2. Intermittent faults.

An engineering fault may be defined as a condition in which a part of the machine is not working as it was designed. A logical fault is one in which the machine fails to produce the correct result to an operation. An engineering fault may result in a permanent logical fault or an intermittent logical fault or in no fault at all. A programme is only capable, by its very nature, of testing for logical faults.

As a general rule, the cause of a permanent fault is easy to find. Diagnostic programmes are useful in this case for two reasons. The location of the fault can be isolated to within a small part of the machine in a very short time and secondly a logical system for searching for the fault is automatically carried out.

Intermittent faults present a very difficult problem. If the occurence of the fault is not too frequent, for example on average once every 10 minutes, it may be possible to do useful computing. For this to be possible it is necessary that programmes put to the machine have incorporated checks at frequent intervals so that in the event of a failure not too much time is wasted. If the fault is fairly frequent then it may be possible to find it by using a diagnostic programme which is continuously repeating. Now the programme may consist of a large number of instructions, on only one of which will the machine fail. In order to increase the probability of this instruction being obeyed, whilst the fault is present, the number of instructions should be kept to a minimum.

It is therefore desirable to get an approximate idea of the location of the fault and then a shorter, less general programme can be used.

If the frequency of the occurrence of the fault is such that it is not possible either to find the fault or to proceed with the computation, for example if the fault occurs once every two minutes, then the supply voltages are varied and this may result in making the fault more frequent in occurrence.

The Diagnostic Programme as part of the General Servicing Scheme

Faults in the electronic circuits are almost invariably finally located by using an oscilloscope to observe the waveforms. Even if the fault is completely diagnosed without the need for an oscilloscope it is desirable to check the waveforms after the fault has been cleared.

It is a good plan to perform a daily inspection so that faults may be anticipated. That is to say there may be "engineering" faults which have not yet developed into logical faults.

Marginal checking (that is the alteration of operating conditions of the valves *e.g.* variation in supply voltages) used in conjunction with diagnostic programmes is a very valuable tool.

The daily maintenance at Manchester is arranged as follows. All the test programmes are run and, if there is no failure, the heater voltage is slowly reduced by about 5%. If there is still no failure, the heater voltage is returned to normal. Then as part of a general inspection of the whole machine a small section of it is examined in detail. Simple tests are made to check the emission of the valves in situ. A different part of the machine is taken each day and during the course of a few weeks the whole machine is checked.

The Scope of Test Programmes

There is a certain section of the machine in which there must be no fault for a programme to run and it follows that it is not possible to use test programmes to find faults in this section. This part of the machine is associated with the generation of the basic waveforms. However, a fault in this section is generally very obvious. For example, it may manifest itself on the displays on the operating console as the absence of part or all of the raster.

There is another section of the machine in which the location of the fault can be indicated by a programme but not precisely specified. In the remainder of the machine the fault can be pin-pointed to one or two valves. The relative sizes of these sections in the Manchester machine has been estimated by counting the number of pentode valves in each. They are

no indication	27%
approximate indication	29%
precise indication	44%

The second purpose of Test Programmes

In addition to locating faults, test programmes are useful in indicating that the whole machine is working correctly. The "overall test" was designed with this aim in view. In conjunction with the storage tests this routine gives a substantially complete check of the working of the machine. Furthermore, in the event of failure, the overall test gives the approximate location of the fault.

Approaches to the design of Test Programmes

The essential problem is to find the instruction (and possibly the number on which the instruction performs) which the machine has failed to do correctly. When this is known the instruction can either be set up on a row of switches and the instruction repeated continuously or a very simple programme can be constructed using only two or three instructions including the defective one. In addition, a list of probable valve failures which would produce the fault can be provided with the test programme.

One way of deciding the way in which a test programme can be constructed to test a particular unit is to consider all the possible failures of the component parts of the unit. Whilst it would be desirable to make the programme include tests for all these failures, a more direct approach is to consider the ways in which the instruction associated with the unit may fail and to determine all the different digit combinations of the numbers on which these instructions act.

A principle which has been adopted in the design of test programmes for the Manchester machine is to build up each programme from a simple starting point. It is first necessary to assume that certain elementary functions are working, for example, the subtractor and the test facility; then using these, further instructions can be checked and these instructions can then be used in testing other instructions.

Conclusions

Experience on the Manchester University Computing Machine has shown that test programmes are essential for efficient and rapid servicing.

It has been observed that, when the test programmes were first used on the Manchester machine, failures were not uncommon, but as the faults diagnosed by programmes were cured the failure rate diminished considerably. In other words, the machine "learns" to do the test programmes. This is, of course, an indication of the value of the programmes.

In view of the value of these programmes, it is suggested that machine designers should keep them in mind, and design so as to reduce to a minimum the part of the machine - quoted as 27% for the Manchester machine in a previous part of the paper - which cannot be tested by a programme.

One deficiency in the scheme originally adopted for test programme design resulted from the assumption that the most difficult numbers which a circuit had to handle could be stated by the designer. This has proved to be untrue and far more stringent testing is obtained by using random numbers (for example, see Random Multiplier Test-Appendix 2.)

The programmes which have been used most in the particular case of the Manchester machine are the overall test, the multiplier test, the c.r.t. storage test and the magnetics storage test. These and other tests are described briefly in Appendix 2.

Appendix 1

The Manchester Mark II Machine

The machine has 8 c.r.t. stores each of which can hold 64 twenty-digit lines of information. An instruction occupies one twenty-digit line and a number generally occupies two consecutive twenty-digit lines. The Cathode ray tube stores are backed with a magnetic drum store, which is divided into tracks, each track being equivalent in capacity to two c.r.t. stores.

There are four subsidiary c.r.t. stores. The control store holds a number representing the position of the instruction next to be obeyed. A second line on the control store holds the digits of the instruction being obeyed.

The B-store has 8 twenty-digit lines, and is used, with its associated facilities to modify instructions. It can also be used for other purposes including keeping a count of the number of times a loop of instructions has been obeyed.

The accumulator store is double length so that it may accommodate the product of two forty-digit numbers. Special provision is made in the arithmetic instructions to give access to the more and less significant halves.

The multiplier store holds two 40-digit numbers - the multiplicand and the multiplier.

The more important facilities of the machine are as follows:- addition, subtraction, the logical operations "and" "or" "not equivalent", and units to find the position of the most significant digit of a number, and to find the number of "ones" in a number (the sideways adder). A fast multiplier is also provided.

Appendix 2

The Library of Test Programmes for the Manchester Machine

In general in the event of a failure the routine stops, and information about the failure is displayed on one of the monitor tubes on the operating console.

The Overall test

With the exception of the c.r.t. and magnetic storage this is a complete check of the whole machine.

This routine is stored permanently on the magnetic drum. The routine is arranged so that after every transfer of a section of the programme from the magnetic drum store to the c.r.t. store, the contents of the c.r.t. store are checked by a summation of all but the last line and the sum compared with the check sum on the last line.

The first section deals with the accumulator and part of the arithmetic unit - the adder, subtractor and logical operations.

The individual answers to a series of operations are recorded and at the end all the sub-answers are summed and the final result tested. In the event of failure, information can be obtained by examining the sub-answers on the c.r.t. store monitor tube on the console.

Test programmes should be divided into two categories (a) verification programmes to check whether the machine is in a reasonable working condition (b) analytical programmes designed to give an indication of the location of a fault.

It is possible to use many mathematical programmes as verification test programmes, as, for instance, when a computation is frequently repeated with different data. Runs can be repeated at intervals, with marginal test voltages applied, and the results compared with those obtained originally.

MR. CAMINER (J. Lyons & Co. Ltd.) was surprised that any mathematician would risk using a machine with a fault occurring as frequently as once in ten minutes, as was suggested in the paper. This, he said, would involve frequent arithmetic checks and might cost twice as much time for checking as for useful computing. A computing machine, in his opinion, should only be used when it was in perfect working order as far as could be ascertained.

DR. FRIEDMAN (Cambridge University Mathematical Laboratory) agreed that computing machines appear to learn to do test programmes but thought this was due to a weakness in the test programmes. He supported Mr. Phister's test-programme rota proposal.

MR. WILLIAMS (NPL) pointed out that the impression that computing machines learn to do test programmes arises from the fact that the service engineer learns from them how to detect particular machine faults which are quickly rectified when they occur or are permanently cured by slight local redesign.

MR. WRIGHT (NPL) thought that by studying the fault record of a machine and its design it should be possible to design good test programmes for verification and diagnostic purposes. He suggested that faults dependant on pulse pattern should be classified as conditional permanent faults.

MR. D. O. CLAYDEN (NPL) agreed that diagnostic and test programmes should be based on experience of machine faults. Full details of the programmes should be written up for the benefit of all machine users and particularly for the service engineers.

The author, in reply, agreed that machines appear to learn to do test programmes. He preferred to keep the number of test programmes as small as possible so that users could be familiar with them.

Although generally agreeing with MR. PINKERTON on terminology he pointed out that engineering faults do not necessarily cause logical faults.

The use of mathematical programmes as test programmes was a useful technique but could not always be applied.

33. Component Reliability in the Computing Machine at Manchester University

by

A. A. Robinson

Ferranti Ltd.

The computing machine at Manchester University has now completed more than 7000 hours of total running time, and it is interesting to examine the records of component failures to compare rates of failure of different components and to analyse the causes. The list of recorded failures from 15 July 1951 to 21 February 1953, a period of nineteen months covering 7420 hours of total running time, is as follows:

	Number used in machine	*Failures*
Resistors	8500 (estimated)	19
Capacitors	1500 (")	10
Valves	4053	304
Storage tubes	12	28

This is intended to cover all kinds of components of which more than five have failed. It excludes the power supply equipment, and also a quantity of resistors which did not fail but were replaced as a result of modifications, or, in the case of some RMA type 1 resistors, because they appeared to be over-rated.

RMA type 1, 2, 8, and 9 resistors are often referred to as 2, 1, ½, and ¼-watt resistors, but reference to the curves of recommended maximum rating against ambient temperature shows that at these ratings type 1 has a margin of only 50% at 40°C ambient temperature, whereas types 2, 8, and 9 have 100% margin. This may account for type 1 resistors showing signs of deterioration. Our policy for some time has been to replace type 1 resistors by vitreous enamelled wire-wound types.

The causes of failure in resistors were as follows:

Open circuit	12	(9 of these were wire-wound)
Resistance decreased	5	
Unrecorded	2	
Total	19	

and for capacitors

Short circuited or low resistance	6
Other causes or unrecorded	4
Total	10

Thus the most common cause of resistance failure is open circuit, and also, seeing that only about 10% of resistors in the machine are wire-wound, a disproportionate number of failures are in wire-wound resistors. In capacitors the most common cause of failure is loss of insulation.

We now come to valve failure which is by far the most common form of component failure that we have met with. The rule is that wherever possible each valve failure shall be recorded in the machine log book together with a diagnosis made of the cause of failure. The faulty valves are kept for examination, and recently a batch of 244 of these valves was examined in the laboratory to check the working of this system. The results were as follows:

Valves examined	244
Valves for which a diagnosis had been given	144
Valves diagnosed correctly according to laboratory tests	90
Valves found faulty, but wrong or no diagnosis given	98
Valves in which no fault was found by laboratory tests	56

It will be seen that the diagnosis, when given, were right in nearly two-thirds of the cases. The proportion of valves in which no fault could be found does not seem unduly high, especially as this number will certainly include valves with obscure intermittent faults which are not easily detected in the laboratory.

The numbers of failures of the different valve types are:

EA 50 (CV 1092)	106 failures out of 2408 valves (4.4% in 7420 hours)
EF 50 (CV 1091)	109 failures out of 870 valves (12.5% in 7420 hours)
EF 55 (CV 173)	45 failures out of 525 valves (8.6% in 7420 hours)
Other types (largely CV 138)	44 failures out of 250 valves (17.6% in 7420 hours)
All types together	304 failures out of 4053 valves (7.5% in 7420 hours)

The following table classifies the causes of failure in the valves whose faults were identified in the laboratory.

	EA 50 (CV 1092)	*EF 50 (CV 1091)*	*EF 55 (CV 173)*	*Other types*
Low or no emission	27	72	22	10
Cracked envelope or soft	5	-	2	6 (All but one were CV 138)
Heater open-circuit	3	-	3	2
Interelectrode leaks or shorts.	30	2	2	1
Other cause	-	1	-	-

The EA 50 appears the best type from the point of view of percentage failures, but accounts for a large proportion of the total failures because of the large number of valves used in the computer. Many failures are due to interelectrode leaks and short-circuits, which points to a weakness in the insulation of the valve. This taken together with the difficulty in finding a satisfactory valve-holder and the fact that the EA 50 is nearly obsolete, has determined us to replace the EA 50 by the CV 140 in future machines. At present we have no first-hand quantitative information about the reliability of the CV 140, but we believe that it will be at least as reliable as the EA 50.

In the cases of the EF 50 and the EF 55 the majority of failures are due to loss of emission, which does not necessarily point to any weakness in the construction of the valve. The fact that the EF 55 has a lower percentage failure than the EF 50 may be accounted for by the EF 55 having been on the average more conservatively rated than the EF 50. Recently we have tended to rate the EF 50 more lightly, but a result of this policy has been to make the average rating of EF 55's still more conservative.

Among the miscellaneous types an outstanding feature is failure of five CV 138's where the valve has become soft. Four of these were cracked near the pip, which rules out the suggestion that the seals were strained during insertion.

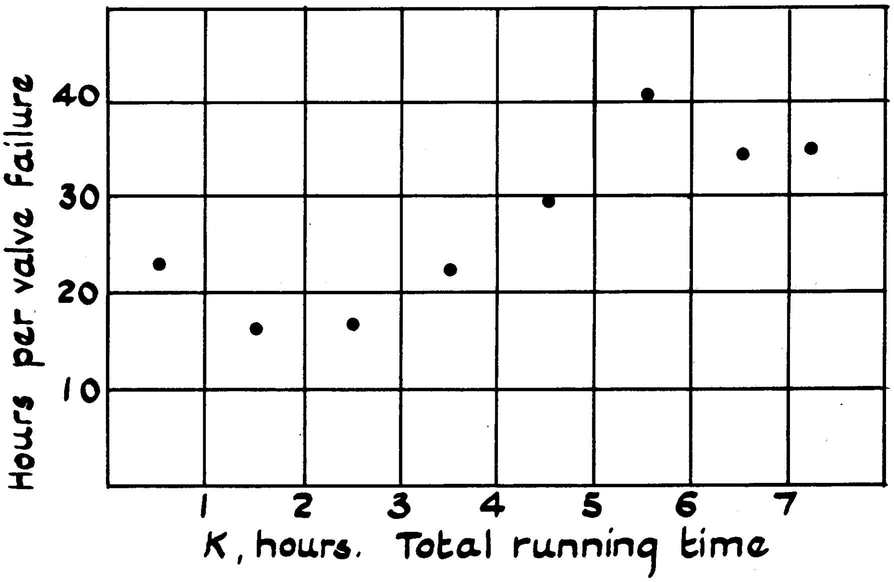


FIG. 1

Fig. 1 shows the mean hours per valve failure averaged over successive periods of one thousand hours plotted against time. (There are only small differences in the modes of variation for the different types.) There is a decrease in hours per valve failure to about 15 at 2000 hours, and after that an increase, which has recently shown signs of a check. At present there are on the average about 35 hours of total operation time per valve failure. We attribute the change of trend at 2000 hours to the introduction of a twenty-four hour working day which took place at this time.

With regard to storage tubes, the computer uses twelve of these, all of type VCRX 266. There have been 28 failures in the 19 months under review, corresponding to an average working life of nearly 3200 hours, or, on a basis of calendar time, of just over eight months. Of these tubes, 18 have suffered from a manufacturing fault which seriously shortens their life. Fortunately the cause has been found by the manufacturers, and tubes are now being produced free from this fault. It is noticeable that storage tubes show a high proportion of rejections due to open or short-circuits, leaks inside the tube, or other faults that cannot be put down to wearing out of the tube. This may be a natural consequence of the developmental nature of the tubes. The demand for storage tubes is at present increasing rapidly, and we hope that the situation will be improved when storage tubes are made in larger quantities.

Two other sources of unreliability in the machine do not concern the failure of components. They are power supply irregularities, and unsteadiness of storage tube deflection wave-forms - often an effect of power supply troubles. Between them these have had an effect on overall reliability comparable with that of valve failures. Two lines of attack are possible. The first, which has recently been carried out with outstanding success, is the reorganization of the deflection wave-form generators, so that power supply irregularities have less effect on deflection wave-forms. The second, which will be carried out as soon as possible, is improvement of the power supplies with the object of removing the irregularities at the source.

To summarize, examination of the records of faulty components in the computer at Manchester University has shown that the majority of component failures are of valves. Just over 4000 valves are used, and the average failure rate has been very nearly 1% per thousand hours of total running time. Storage tubes have shown a much shorter life than valves, but the comparatively small number used has usually prevented their high rate of failure from having an appreciable effect on the overall reliability of the machine. Failure of resistors or capacitors has been rare, but wire-wound resistors have been found significantly less reliable than carbon types. Other important causes of failure are power supply irregularities and the resulting unsteadiness of storage tube deflection wave-forms.

Methods available for improving reliability are the reduction of storage faults by the methods outlined, and an attempt to decrease the number of storage tubes subject to functional failure as opposed to cathode failure. Little improvement can be expected from improvement of resistors and capacitors. Reducing functional failures in valves could produce an appreciable but not a large reduction in total failures. Any large improvement in this direction in a machine of fixed size involves prolonging cathode life by even more conservative rating of the valves, or by making use of types having unusually long cathode life.

Discussion

(Note: Those parts of the discussion on Paper 31 which were found to be more relevant to Paper 33, have been included in *this* discussion.)

MR. CARTER (Telecommunications Research Establishment) said he was disturbed about the number of low-emission failures in c.r.t.'s particularly in view of the fact that the emission density in tubes used for storage is relatively low. They had had many failures in silvered mica condensers, but little trouble with resistors after eliminating a particular variety of carbon resistor.

MR. KITZ (Instituto Nazionale per le Applicazione del Calcolo, Italy) referring to the c.r.t. failures, said that the prolonged running of such tubes, or ordinary valves, at much less than their normal emission had been found to be very bad for them, for reasons not fully understood. The low-emission fault in a c.r.t. might perhaps be cured by "flashing" - running for a short time at high emission and high temperature.

DR. PINKERTON (J. Lyons & Co. Ltd.) said that a high proportion (some 300 out of 2000) of a certain type of double diode had shown a great increase of slope resistance. "Flashing" with 8-10 V. on the heater, effected some improvement. He noted that the author seemed to have had no trouble with plugs and sockets or valve holders, which was also his experience. He had had many failures of small ceramic condensers, and some in silvered mica condensers; also drift in high-value "high-stability" resistors. He had found that valve heater failures were very infrequent. In LEO the heaters are switched on and off very slowly - about 2 minutes being allowed for each process.

MR. NEWMAN (NPL) found that the valve reliability figures for the Pilot ACE and MADAM were not very different when fully analysed. Some of the ACE Pilot Model valve failures might have arisen from mechanical disturbance on plugging chassis in an out. Heater failures were rare. In many cases of apparent low emission, the real cause was found to be "interface" effect. The apparent higher failure rate in c.r.t.'s might be due to their more stringent operational requirements as compared with the on-off behaviour required of valves.

There had been very few condenser failures in the Pilot ACE, and little trouble with resistors, except for large drifts in the value of "high-stability" types of over 100 000 ohms. These seemed more numerous than they really were, because they gave rise to very troublesome types of fault.

DR. COOMBS (GPO Post Office Research Station, London) said that in his experience, large electronic equipments work best when they are left alone as far as possible. He was horrified by DR. PINKERTON'S procedure of taking valves out to test them - which was like digging up potatoes to see if they are growing.

There seemed to be no agreement about the relative merits of leaving heaters on continuously or switching on and off as required by the use of the machine. This was being investigated on MOSAIC by applying the different procedures to different large groups of valves. The switching on and off is done slowly - over a period of a minute.

There had been much talk of valve failures, but it was equally true that a good modern valve would stand a surprising amount of bad treatment and he gave instances of this. Work directed to very long valve life (100 000 hours or more) is making good progress and a considerable improvement in this respect could reasonably be anticipated.

DR. GRIMSDALE (Manchester University) replied to DR. COOMBS, that the Manchester machine showed fewer faults when it was run continuously.

MR. RUBACH (All Power Transformers Ltd.) suggested that components which had been subjected to drastic processes in manufacture, *e.g.* vitrification or bakelizing, were likely to be unreliable in use and should not be chosen for computers.

MR. PAGE (NPL) said that a routine testing of all valves in a chasis, found many valves which would have been rejected on initial test, though the chasis still worked. Such routine testing would tend to exaggerate apparent valve failures.

MR. GRIMSDELL (Mullard Research Laboratories) described experimental confirmation of increased failure rate in miniature valves due to frequent plugging in and out. (Other speakers reported similar observations.) The continued running of valves with reduced heater voltage could be very harmful, if there was appreciable anode current.

MEDIUM-SIZE DIGITAL COMPUTING MACHINES

Chairman: Mr. J. R. Womersley

34. The Harwell Computer

by

E. H. Cooke-Yarborough

Atomic Energy Research Establishment, Harwell

Introduction

The Harwell Computer (*Fig.* 1) differs very markedly from other automatic computers designed up to the present. It may be of interest, therefore, first to relate the reasons for adopting this unorthodox design, and then to show in what ways the design has influenced the use of the computer.

In 1950 the Harwell Computing Group were finding that an increasing amount of their work involved much computation of a simple but repetitive nature. Such work performed on a desk calculator is extremely tedious for the operator and there is a real danger of errors arising from sheer boredom. Moreover such computations divert computing effort from other work which would make better use of the flexibility and adaptability of the human intellect. Accordingly the Harwell Electronics Division began to consider the possibility of designing an automatic computer to do this work.

Only a small amount of effort could be assigned to the development and construction of a computer, so the design had to be made simple and had to use well established techniques where possible.

It appeared that a high-speed computer was not essential, since much time would in any case be spent in feeding data into and out of the machine. What was needed was something which would plough steadily and relentlessly through the mass of data presented to it. An all-relay computer was first considered, since adequate speed seemed attainable and experience with quite complex relay apparatus at Harwell had shown reliability to be excellent. Moreover, a multi-contact relay is an almost ideal component for circuit switching in a slow computer; indeed, designers of fast computers must often wish for a high-speed circuit element as convenient and adaptable. Relays are however far less convenient when used as storage elements, and in existing all-relay computers it is necessary to use a very large number of relays to make up the quite modest amount of quick-access storage provided in these machines. Consequently it was decided to use relays only for switching and, after examination of other possibilities, dekatrons (*ref.* 1) were chosen for the quick access storage, since these were already coming into wide use in scaling circuits at Harwell and their properties were well known and seemed to offer many advantages. For example, each dekatron can store one decimal digit, thus doing the work of at least four storage relays with a power consumption of less than 100 mW. The associated electronic circuits can be relatively simple, since a dekatron delivers a pulse large enough to be used directly in arithmetical circuits without amplification, while the power required to register information in a dekatron is conveniently small. Several other convenient features of the dekatron emerged as the design proceeded.

It is not proposed here to discuss the detailed circuit design of the computer which has been fully described elsewhere (*ref.* 2). The design philosophy was that circuit simplicity and reliability were all-important, speed being a secondary consideration.

It was soon found that a decimal machine presented difficulties and complexities of a logical, not a circuit nature, which might not have been encountered in designing a binary machine. After some time had been spent in investigating the principles involved in long multiplication and division, a pilot machine was built, using three decimal digits plus the sign, to help resolve these problems.

The information thus obtained greatly facilitated design and construction of the full-scale machine which worked for the first time in April, 1951, just over a year after the commencement of the project. Within a few weeks it was performing simple but useful computations connected with particle size data. A good deal of work remained to be done on the circuits and wiring, so, although the machine performed increasing amounts of useful computation during this period, it was not finally handed over to the Computing Group until May 1952.

Properties of the Computer

The machine obtains its instructions and numerical data from standard 5-hole teleprinter tape. There are seven mechanical tape-readers; some of these usually carry repetitive routines in the form of closed loops of tape. Results are printed by a modified teleprinter. A two address code is used and each arithmetical instruction contains two pairs of decimal digits identifying the two addresses involved, preceded by a digit defining the operation to be performed. The reading of each instruction takes about a second and its execution takes a further second if the operation is a simple one; multiplication or division take ten to fifteen seconds. The speed is thus comparable to that of a normal desk machine.

The computer handles numbers between +10 and -10 containing eight decimal digits; negative numbers appear as complements. There are 40 quick-access storage locations using dekatrons (with provision for adding a further 50) and a double length dekatron accumulator.

The machine is designed for unattended running, so there is a comprehensive system for electrically checking the correct functioning of the circuits. Arithmetical checks are usually included in the programme to detect the small proportion of errors which may elude the functional checks. When the machine is set for unattended operation the occurrence of an error or other difficulties causes the machine to recommence the computation, to go back a few steps in the computation or to start the next problem. If the machine fails three times without completing the current computation it shuts down. If, however, after one or two failures it successfully completes the computation it begins the next computation with all its "lives" returned. This arrangement has been found very valuable in preventing the computer being shut down by a trivial fault.

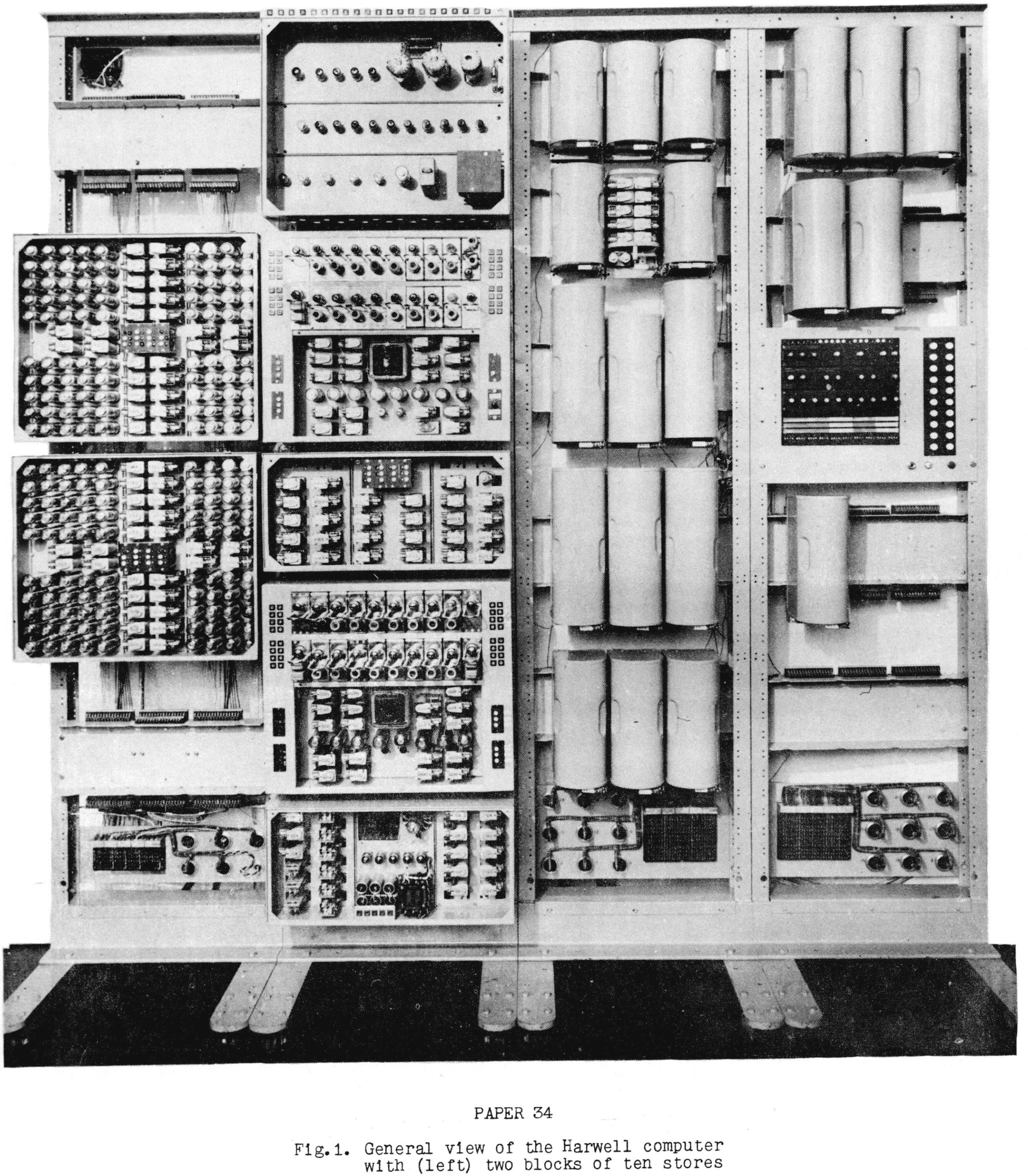
Serviceability

A slow computer can only justify its existence if it is capable of running for long periods unattended and if the time spent performing useful computations is a large proportion of the total time available. There appear to be as many ways of assessing computer serviceability as there are computers. The method used here is to take the number of hours the computer runs out of the possible maximum of 168 hours in a week. This therefore gives the total computing time plus the time spent on fault finding or maintenance with the computer switched on. The latter time seldom exceeds four hours per week and is usually much less, so it does not very significantly affect the results.

Over the period May 1952 to February 1953 the computer averaged 80 hours a week running time. This includes one week when the machine was serviceable but no programme was available and four weeks when the machine was out of action following an accident which caused severe mechanical damage. If these five weeks are excluded the average weekly running time becomes 92½ hours or 55% of the possible maximum. Most of the lost time was due to the machine failing during unattended running and having to wait until the next working morning for the fault to be corrected.

A further way to assess serviceability is to consider on what proportion of occasions when the machine was left to run overnight it worked until the following morning (15 hours later). This has been the case on 56% of the occasions. The corresponding figure for working over the weekend (63 hours) is 33%.

These percentages exclude occasions on which failure was due to errors in setting up the machine or in preparing the programme; they also exclude the occasions when the machine successfully completed the computation before the end of the period. On many of the occasions when the machine failed, the computation had reached a useful stage before failure occurred, so that the night's work was by no means wasted.



PAPER 34

Fig. 1. General view of the Harwell computer with (left) two blocks of ten stores in position.

Most of the failures tend to be of a fleeting, intermittent nature, so one fault in the computer may cause several failures before it is detected. Faults in the relays and in the gasfilled cold cathode tubes have been commendably few. Most of the faults have been due to more conventional components, such as thermionic valves, metal rectifiers etc. Clearly there is room for improvement. It is hoped that this will be achieved by replacement of components prone to failure and by arranging for regular preventive maintenance.

Even at present, however, serviceability is sufficient to make the machine a powerful computing tool. Few, if any, other computers show comparable serviceability reckoned on the same basis, especially as maintenance of the Harwell computer represents only a small part of the work of one man.

Preparation of Programmes

The Harwell computer provides almost all the facilities of a large computer except high speed, so programming is in principle very similar. In practice, however, many of the subroutines and procedures used with fast machines become excessively time-consuming and alternative methods are used whenever possible.

The main difficulty arises in reading instructions or data from the paper tape in an order which cannot be predicted when the tape is prepared. This situation arises, for example, if the tape contains a table of functions or a rather complex subroutine containing many conditional instructions. Under these conditions the machine has to search along the loop of tape for the desired address. Computation is interrupted until this address is found, so if a long search is necessary much time may be lost. Numerous long searches also tend to reduce the life of the tape.

The need for long searches is minimized by splitting subroutines into as many small loops as possible to make best use of the available tape-readers. A consequence of this is that it is difficult to prepare economical subroutines which are flexible enough to be used in many different computations.

When the computer was first put into service it had only 20 dekatron stores and this made programming very difficult. The increase of the number of dekatron stores to 40 has almost completely removed this difficulty and shortage of quick-access stores is now seldom an embarrassment. Of course, if there were enough of these stores to accommodate long subroutines or tables of functions, the difficulties mentioned in the preceding paragraphs would disappear. The cost, however, would be prohibitive, because, unlike magnetic drum storage, the cost of Dekatron storage is almost directly proportional to the storage capacity provided.

Desirable Improvements

Experience has shown that certain changes would simplify the programming or speed up the computation without adding much complication. While the changes could easily be included in a new machine they have not been incorporated in the present one as it is considered that extensive modification of an existing machine of this type would cause permanent deterioration of reliability.

The changes are indicated below.

1. Provision of a facility for clearing a storage address and feeding a number into it in one operation. At present a store can only be cleared while transferring a number out of it.

2. Elimination of many sequential relay operations when reading in orders and setting up the conditions for arithmetical operations; this would reduce the reading time to about 500 ms and speed up multiplication and division.

3. A change in the method of recognizing an address when searching a tape, which would result in the search speed being limited only by the characteristics of the tape-reader itself.

4. Provision for computation to continue while an address is being sought on a tape.

5. Use of the faster dekatrons now available. This would reduce the times for multiplication and division by about three seconds.

Conclusions

Although of relatively simple design and low speed, the Harwell Computer has shown itself capable of performing a very useful amount of computation, because it can run for long periods unattended.

The combination of dekatrons, relays and teleprinter apparatus has been found very suitable for a computer of this size and speed. Modern dekatrons are capable of much higher speeds, but their use would add little to the speed of this type of computer. To exploit their increased speed successfully, very different methods of switching and storage would be necessary. Such a computer would undoubtedly be very useful, but it is perhaps questionable whether it could be constructed, run and maintained with as little technical effort as is needed by the Harwell computer.

References

1. BACON, R. C. and POLLARD, J. R. An electronic digital computer. *Electron. Engng.* 1950, **22**, 173.

2. BARNES, R. C. M., COOKE-YARBOROUGH, E. H. and THOMAS, D. G. A. An electronic digital computer. *Electron. Engng,* 1951, **23**, 286, 341.

35. The APE(X)C - A Low Cost Electronic Calculator

by

A. D. Booth

Birkbeck College Electronic Computation Laboratory

Historical Background

During the 1939-45 war it became apparent to the author, that the extensive computations which result from any attempt to apply available mathematical techniques to problems of real physics must lead to a mass of numerical work far in excess of the available resources of almost all University departments.

At the conclusion of hostilities, work was started at the laboratories of the British Rubber Research Association on a special purpose digital calculator for the processes of Fourier analysis and synthesis. As this work proceeded, it became more and more apparent that the components needed for the special purpose machine could be more advantageously disposed in a general purpose machine whose sequence of operation was pre-set, either by means of punched tape or by physical plugging.

Whether this line of development would have lead, independently, to the modern concept of a general purpose computer with a central storage organ it is impossible to say, since a visit to the U.S.A. in 1946 brought the author into contact with von Neumann and revealed the extensive progress already made by the Princeton group.

After a period of study and stimulating discussion at Princeton work began in 1947 on the construction of a calculating machine operating with a central storage organ.

The first magnetic store was operated in November 1947 and consisted of a *disk* of oxide coated paper rotated at high speed. Because of the poor wear characteristics of this device it was soon replaced by a nickel plated cylinder.

Experiments upon the design of read/record heads followed and, after a brief period during which the Bigelow single wire head was favoured, a single-turn single lamination version was designed (*ref. 1*) which had the required characteristics. This was adopted by other computer groups and particularly by Williams at Manchester, where a substantially similar design was produced and used in the Ferranti computer.

The computing organ used with this store was a parallel operation relay device, since called ARC (Automatic Relay Calculator). It formed an excellent testing ground for logical techniques and was designed, constructed and operated within a period of 4 months - surely a record.

From this point two divergent paths were followed. The relay circuits of ARC were combined with an electro-mechanical store for numbers, and a sequencing device for instructions. The electronic circuits and magnetic drum store were used as the basis of a prototype all-electronic machine, called SEC (Simple Electronic Computer).

ARC is now housed in the Computation Laboratory of Birkbeck College and only lack of operating staff and of space prevent its continuous use. SEC was eventually completed as a student training project and is now used only for demonstration purposes. In its early stages it proved useful for circuit testing but its progress was so slow that the complete APE(R)C machine was in operation at an earlier date.

THE APE(R)C

The very limited resources of the Computer group (at no time has more than one engineer been employed) made it necessary to construct a first final model of the Birkbeck machine for a sponsoring agency, the British Rayon Research Association. The APE(R)C, or All Purpose Electronic Rayon Computer, was completed in August 1951, but, owing to staff changes was not operated on problems until July 1952, and then only with limited storage. The complete store was installed at the end of 1952 and the machine is now in full operation. (*Fig. 1*).

APE(R)C Characteristics

It has long been maintained by the author that for reliable operation an electronic computer must use the minimum possible number of electronic valves. APE(R)C, designed with this in view, has less than 450. No germanium diodes are used and the number mentioned above includes all thermionic diodes.

To simplify the design requirements the speed of operation was set at 30 kc/s. Serial working of the arithmetic and storage units was adopted and a serio-parallel control unit designed.

The word length is 32 bits and numbers normally consist of 31 binary digits and a sign, in the range $-1 \leqslant x < 1$, negative numbers being shown as complements.

By the use of a two address code the normal mean access time of 8 ms. associated with the magnetic drum store, can be reduced by optimum coding so that the operating time for addition or subtraction is only about 1 ms. Multiplication requires n addition/subtraction times, where n is the number of changes $0 \rightarrow 1$ or $1 \rightarrow 0$ in the multiplier digit sequence.

The storage capacity is 512 words of the standard length and input/output is by means of teleprinter.

The input/output organs

Both of these functions are performed by a standard Creed keyboard teleprinter. Input data is typed in the form:

(location) (word)

a printed record is produced and if this agrees with the coder's script the depression of a switch causes the word to be recorded in the required storage location.

It has been found that although the teleprinter often operates in a faulty manner, rarely, if ever, is a correct character typed in response to incorrect setting signals. The chief cause of misbehaviour in earlier teleprinter input systems seems to lie in the attempt to run the printer in synchronism with some standard frequency. In the Birkbeck system the printer emits *from its own mechanism,* shift signals which cause absorption of the actual signal, appearing on the operating magnet by the electronic register of the computer.

Output is performed similarly. A start impulse causes the drive clutch to be engaged, the printer then selects (electronically by means of a shifting register) the data for presentation to its operating magnet at the required instants of time. The use of this system has resulted in faultless operation of the printer (except for mechanical breakage).

It is worth noting that tests on solenoid operated parallel input systems of the type used on EDSAC showed that the solenoids required 50–100 ms. to settle down (out of the 150 ms. available), whereas the Creed magnet requires only 2 ms. out of the 20 available. This seems to indicate that, unless very well designed solenoids can be produced, the present system is superior.

The system adopted permits the use of *any* standard input/output equipment with the keyboard teleprinter. For example a standard tape-reader can replace the manual operation of the keys, or alternatively signals can enter via a post office line. For output the printer can be paralleled with a reperforator and a tape prepared.

The store

Mention has already been made of the original nickel-coated drum store. This has now been replaced by a much more compact design using oxide as the storage medium. The original single lamination heads have also become obsolete and the new version gives greatly improved characteristics. It was found that nickel plating was too variable in quality to make possible a reduction of track width to much less than 0.25 in., and that any decrease below this value resulted in at least some of the tracks having "phony" spots. A somewhat superficial investigation of the effect suggested that the cause was irregular grain size and possible strain production during plating but in the absence of adequate metallurgical laboratory facilities no more definite conclusion was possible.

The oxide coated drums permit the use of up to 50 tracks/in. and a packing of 100 digits/in. in the track without the need for non-restoring recording techniques. Signal/noise ratio is better than 25 to 1 (compared with 3 to 1 for equivalent track width on nickel) and the output amplitude is several times greater than that originally obtained.

The illustration (*fig. 1*) shows a drum store for 1024 words of 32 bits as manufactured by the Wharf Engineering Laboratories. Credit for the mechanical design is due to Commander S. J. Booth, O.B.E.

In the APE(R)C, track selection is performed by means of a tree of high-speed relays. This gives, despite the low head impedence (less than 1 ohm), completely reliable operation at switching rates of 300 c/s, which matches adequately the speed of the remainder of the machine. A coded interlock is included in the machine so that errors do not arise when switching to other tracks in a manner unknown initially to the programmer.

The arithmetic and control units

It is not proposed to describe in any detail the function or form of these units since descriptions have already been published (*ref.* 2, 3 and 4). The points worthy of remark are chiefly those of economy and reliability.

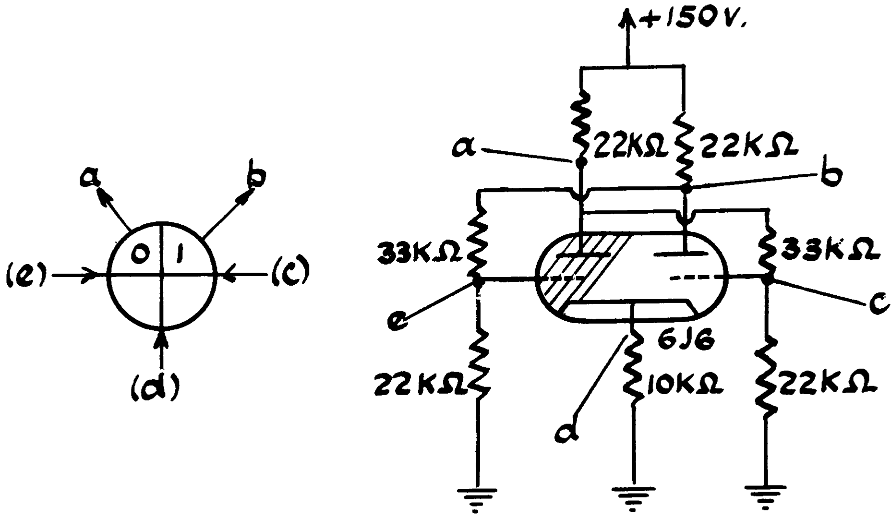


FIG. 2 APE(R)C Flip-Flop

PAPER 35

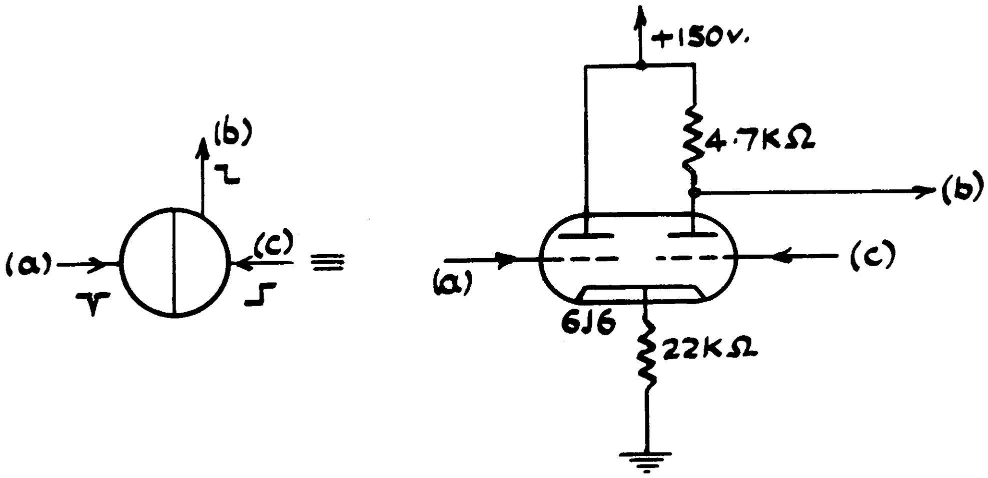
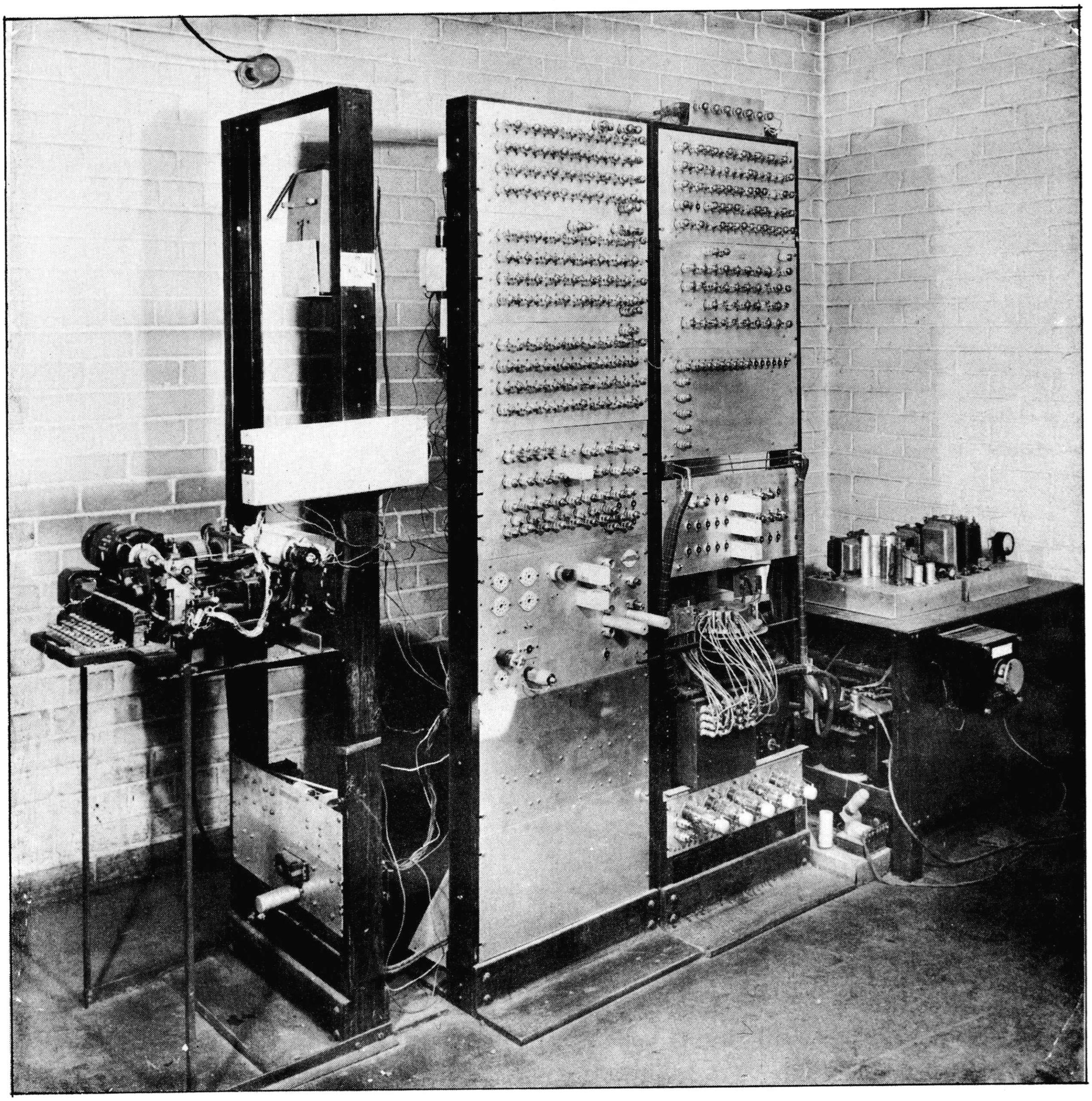


FIG. 3 APE(R)C Gate

PAPER 35



PAPER 35

Fig. 1. The APE(R)C at Birkbeck College Computation Laboratory.

Only two valve types are used, the 6J6 and the 6 AL 5. The former is made the basis of the single-digit storage binary element and of the 2 gate; the latter is used as a buffer where required and also as the delay storage element in the shifting registers.

Very extensive tests were carried out on germanium diodes (chiefly Sylvania type 1 N 38) and the conclusion was reached that these elements were too unreliable, both in characteristics and performance, to find extensive use in computers. Later tests on more recent specimens have done nothing to alter this opinion.

Some typical examples of APE(R)C circuits are shown in *Figs.* 2 and 3.

Operational experience

During the four months July-October 1952 the causes of failure on APE(R)C were:

1. 11 valve failures.
2. 2 electrolytic condenser failures.
3. Teleprinter mechanical breakdowns.

The electronic portion of the machine was serviceable for about 80% of the period mentioned, but teleprinter failure put the installation, as a whole, out of commission for about 50% of the time. It is perhaps only fair to say that the teleprinter is a surplus item which the makers state to be in poor condition, however, the faults seem to fall into two classes:

a. Adjustment drift
b. Fatigue breakage of components

Since fitting the full storage unit and relay track switch device too little time has elapsed to make any statements on reliability. A number of valve failures were apparent when the machine was restored to operation in February 1953, but these have not yet been analysed and some are undoubtedly due to mechanical damage.

THE APE(X)C

The APE(R)C is to be moved to Manchester in the near future. Meanwhile the computer designed for permanent installation at the Computation Laboratory has been built and unit tested. Operational and test experience on the APE(R)C suggested a number of desirable alterations both logical and mechanical, and the APE(X)C (All Purpose Electronic X-ray Computer) has incorporated these in its construction.

In the first place APE(R)C was "designed" by the author on odd scraps of paper and unfortunately the engineer then employed made no attempt to remove redundant elements common to various units. When the APE(X)C drawings were made, no fewer than 180 valves were eliminated!

Secondly, the original chassis layouts are such that servicing is extremely difficult. This has been remedied and although plug-in chassis proved too expensive for the available resources, the new units are all of a standard size and layout which should make maintenance easy.

Finally, considerable use has been made of selenium disk rectifiers, especially in decoding and coincidence sensing. This has lead to the general use of D.C. techniques throughout the machine and, on chassis test at least, to greatly increased tolerance and reliability.

Extended functions

It is hoped eventually to have a large battery of storage drums, and the APE(X)C has been planned so that these can be added without disturbance. Briefly, the machine is synchronized with the new drum, a process which takes 16 ms. A relay director system is used for the selection process. Special instructions allow inter-drum transfer and provide the necessary interlocks.

In addition to this, at least one extra shifting register (two exist in APE(R)C) is to be added, and this will increase the speed of multiplication by a factor of 16.

The repetition rate is increased to 50-60 kc/s so that addition and subtraction will require only 500 μs, and this increase in speed is associated with an extension of the basic storage capacity to 1024 words of 32 bits.

Input/Output

The unfavourable experience with teleprinter equipment has led to its replacement, in APE(X)C, by a Hollerith tabulator which combines the functions of input and output. Modifications have been made to this instrument which enable it to operate under the instruction of the main machine and preliminary tests suggest that it will entirely eliminate the present troubles.

Provision has been made for the input of data, either in standard punched card form, (one decimal word per card) or in the form of binary punchings of 12 words per card. It appears that, for scientific use at least, the latter is quite suitable and is rather easier on the coding operative than the decimal version.

Quite apart from this large scale equipment, Wharf Laboratories are experimenting with a high-speed magnetic tape mechanism which will accept machine output at the 50 kc/s digit rate. The experimental model appears to offer the advantages of reliability and simplicity combined with an ease and cheapness of production which is noticeably absent in available designs.

Completion date

The financial cuts imposed upon university expenditure during the current quinquennium make it impossible to predict any date for the completion of APE(X)C. At present no work is being carried out on the machine since it has not proved possible to obtain the services of an engineer on the inadequate sum allowed for this purpose.

All chassis have been tested, both statically and dynamically and the remaining work is that of interconnexion and integration with the drum and terminal organs, a process which *should* take less than 6 months.

COPIES OF APE(X)C

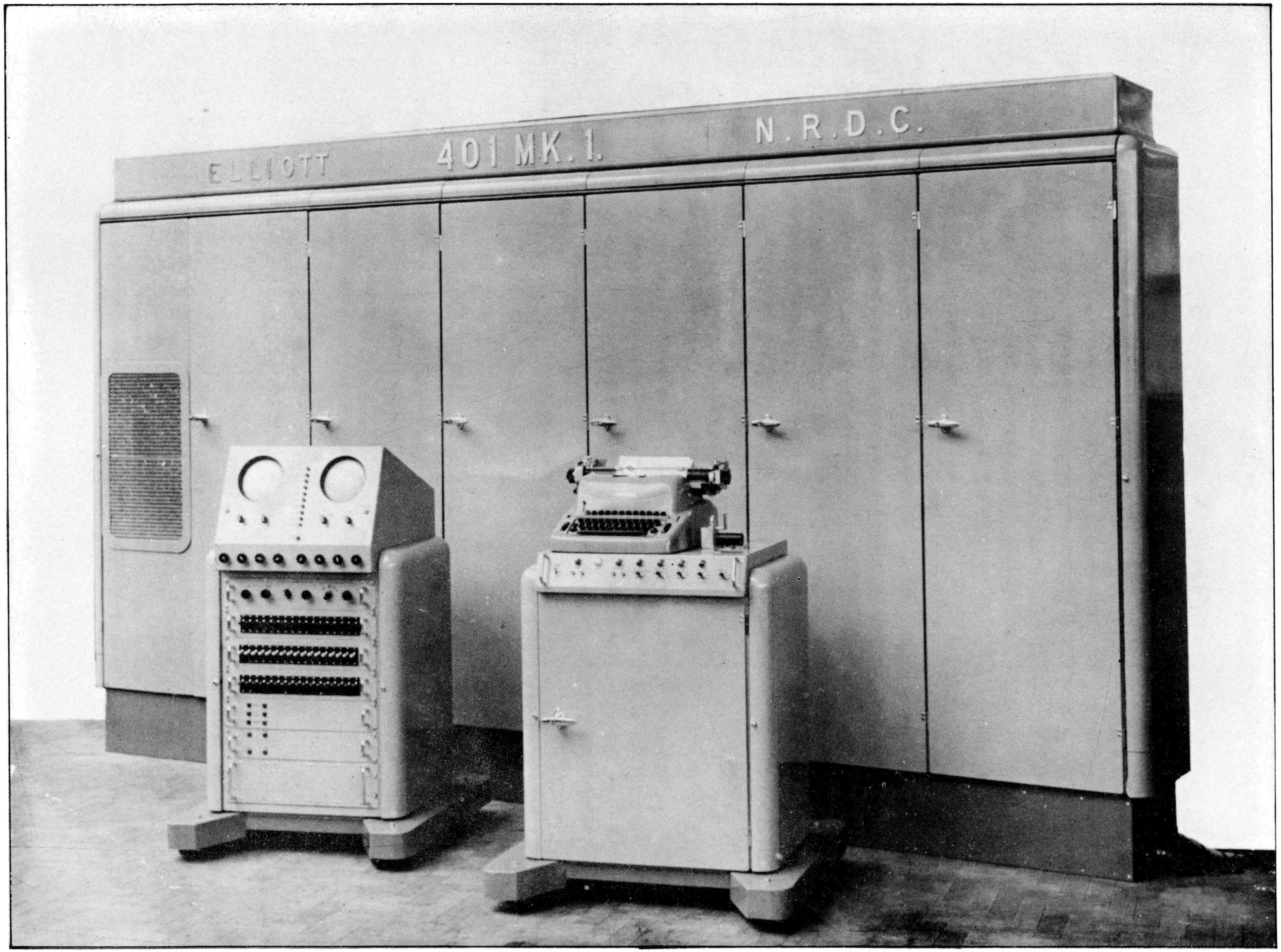
Up to the present, two copies of APE(X)C are in existence. The first, at the Norwegian Board of Computing Machines, Oslo, under the direction of Dr. T. Hysing, is undergoing tests. The second is in general use at the laboratories of the British Tabulating Machine Company.

The latter agency is in the course of developing further machines, based upon similar principles, but designed for commercial use. Considerable stress will be laid upon design for reliable operation and it is hoped to produce a machine which will equal the performance of the standard punched card machines.

It is particularly important, in a machine intended for business use, that no time should be wasted in conversion to and from binary scale, so that the rather elementary computations can be carried out during the passage of the current card through the machine. Two general methods are available for achieving this end. Either the speed of operation of the computer can be made adequate to programme conversions to and from binary scale during card passage, or special conversion equipment can be built. Investigations are proceeding along both of these channels and it appears, at present, that the latter is likely to prove most satisfactory.

References

1. BOOTH, A. D. *Electron. Engng*, 1949, **21**, 234.
2. BOOTH, A. D. *Electron. Engng*, 1950, **22**, 492.
3. BOOTH, A. D. *Electron. Engng*, 1952, **24**, 442.
4. BOOTH, A. D. *J. Brit. Instn Radio Engrs*, 1952, **12**, 587.



PAPER 36

Fig.1. The Elliott-NRDC Computer 401.

36. The Elliott-NRDC Computer 401 - A Demonstration of Computer Engineering by Packaged Unit Construction

by

W. S. Elliott, H. G. Carpenter and A. St. Johnston

Elliott Brothers (London) Ltd. Research Laboratories

Introduction

The Elliott-NRDC Computer 401 (*Fig.* 1) is an essay in computer engineering.

A contract placed in 1951 with Elliott Brothers (London) Ltd., by the National Research Development Corporation called for a design study of a large-scale computer, on the lines of the Manchester University machine, but using new circuit and construction techniques. The study report was presented in February 1952, and described a machine using magnetostriction delay lines for immediate access and intermediate access stores and magnetostriction lines or a magnetic drum for the main store. The proposed machine had multiple transfer facilities, a high-speed multiplier and special circuitry for floating point operation. The Corporation, after considering the report, requested the firm to develop and make a "Pilot Model" computer which would use some of the proposed techniques and would be a small computer in its own right. The computer was to be made so that it could be set up in a University department for a "user trial", that is, to determine what degree of dependability it might have in the hands of a user.

Design

The request for a Pilot Model was considered and it was decided to build an essentially minimal machine round a small magnetic disk store and using plug-in circuit packages including single-word magnetostriction delay line registers. The digit rate was to be 330 000/s, for which frequency a set of standard arithmetic and control circuits had already been developed. The magnetic disk store was under development at the 330 000/s digit rate and at the time, for a store of about 1000 words, seemed preferable to magnetostriction delay lines (*ref.* 3). The logical design of a small computer using these components was devised in June 1952, the principal aim being to secure that the functioning of the machine could be easily checked by waveform monitoring. This aim led to an order code for the machine in which the instruction digits are divided into four independent sub-groups; each sub-group, of three digits, is assigned to some function of control, independently of other functions. It was accepted that programming might thereby be made more difficult and that the size of the machine might be increased above the minimum; but it was felt that this control system would be easier to commission and to understand in operation and therefore easier to maintain.

A two-address code was adopted, to allow optimum programming and so, relatively fast operation.

The magnetic wheel has 8 tracks closely spaced near the outer rim of a 9 in. diameter disk, each track carrying 128 words of 34 digits each (32 active and 2 digits spacing). At a 333 000/s digit rate the rotation period is 13.1 ms.

The single-word registers are used as accumulator, multiplier register, multiplicand register, instruction register and one general purpose register. Two single-word registers are used for the accumulator which can operate single-length or double-length. One 3-digit sub-group in the order selects input to the accumulator, another selects accumulator function, a third selects accumulator output and the fourth is used for miscellaneous purposes.

All transfers are to or from the accumulator, which is used in multiplication to accumulate the double-length product.

The machine functions with a "two-beat" rhythm, that is, orders are selected and obeyed alternately.

The eight magnetic store tracks are electrically switched. The same heads are used for reading and for writing on the tracks. At the moment, four word-times must elapse after track switching or after writing before a word is read. It is hoped later to reduce this amplifier paralysis time.

Engineering Development

The outline scheme was approved by the Corporation and the project commenced in July 1952 with consideration of possible forms of construction. Development of the arithmetic and control circuits already mentioned had resulted in a set of basic circuits (digit delay, inverter, gates and cathode follower) and these had been arranged in three packages. For the previous project, the packages were to be constructed in the form of glass plates with silver-printed wiring and were carefully designed round sub-miniature valves (*ref.* 2). It was in mind that the Pilot Model should be the basis of a computer, physically small though not small in storage capacity nor slow in operation, which could be batch-produced at a low cost. The package circuits were therefore adapted to use miniatures instead of sub-miniatures, the miniatures being much cheaper per envelope as well as providing two separate triodes per envelope. A risk was taken here that valve life might not be as great.

After discussion with the Corporation it was decided that the glass plate, silver-printed type of unit would not be used, but that this Pilot Model, an engineering demonstration, would use packages conventionally wired on paxolin plates. In any later machines resulting from the Pilot Model, automatic assembly techniques might be used. The method of construction of the packages which was adopted, was one originally developed only for laboratory use without a view to manufacture, in which the paxolin plate is mounted in a light alloy surround. The surround, which carries "Jones" type plugs to plug the unit into the computer, conveniently forms a handle for carrying the unit and for plugging it in and out. The three arithmetic and control packages were designed in this form, and the magnetostriction delay registers were designed in three packages ("driver", "line" and "amplifier") of physical form identical with that of the arithmetic/control packages. Other packages for waveform generation and other special purposes were designed in the same physical form. The main circuitry of the whole computer is housed in three 6-ft Imhof cabinets with inter-package wiring at the rear. At the front up to 56 packages are plugged in to each cabinet.

One cabinet houses the magnetic wheel store and the amplifiers. These also are plug-in, though they are not uniform physically with the other packages. In the store cabinet there is also circuitry for controlling the speed of the magnetic disk. The disk carries "clock" and "address" tracks in addition to the 8-store tracks. The motor driving the wheel is speed-controlled by a frequency discriminator to which the "clock" pulses from the wheel are applied. Thus the (passive) frequency-discriminating circuit sets the computer clock frequency.

Two cabinets contain the power supply, line voltage monitoring, line fuses etc. A three-phase A.C. input to metal rectifiers is used, and the rectifier output is taken without energy-storage smoothing to series-stabilizer chassis. An isolating motor-alternator set will be used where the supply mains may be "jumpy".

A seventh cabinet houses a fan to draw air from a duct running above the cabinets. In each cabinet air is drawn up from the base plinth and is taken from each chassis in the cabinet into vertical ducts running up each side of the cabinet to the top duct.

The main computer assembly of seven cabinets, with top duct and base plinth, is of overall size 13 x 2 x 7 ft 6 in. The assembly is readily broken down into units, none of which has a dimension greater than 7 ft 6 in (connexions between cabinets are easily broken and remade). The machine can therefore be moved and set up in a new location with minimum effort.

The input tape-reader (photoelectric) and the output typewriter are mounted on a trolley from which the machine is switched on and controlled. Optional stop and single-step operating facilities are available on the control trolley. A second trolley provides monitor oscilloscopes and facilities for setting up numbers and orders by hand.

Full protective circuits are arranged so that, for example, on failure of any supply line to any cabinet, the computer is automatically switched to a safe state. The machine is also shut down by an overspeed trip on the store driving motor, by excessive temperature of air drawn by the fan from the computer or by no air from the fan (*i.e.* fan failure).

Progress of Project

The plan for development of components, manufacture of components and assembly of machine was drawn up at the end of August 1952. Some development of components had taken place earlier in parallel with the discussions on general design.

The computer was to be a "Laboratory Model", though well enough made to be taken away for use, as previously stated. The mechanical work was done in a "local workshop" adjoining the laboratory and under the direct supervision of the design staff. Wiring staff were detailed to work in the laboratory alongside the design staff. This wiring staff has numbered three for most of the duration of the project. One aim of the exercise has been to demonstrate that we have a technique whereby a dependable computer may be made in a short time using wiring and other manufacturing staff of ordinary "electronic" skill. It is of some interest to record that the whole manufacturing staff was changed half-way through the project and that the assembly and wiring of the arithmetic/control packages was by a sub-contractor.

Most of the packages, chassis and cabinets were completed by the end of January 1953 and tests of the arithmetic and control circuits, in conjunction with one track on the disk, were made in February, with success. During February and March the power supply cabinets were completed and tested and the control trolley is now being completed. It is expected that first tests with tape input to the complete machine will be made early in April 1953.

Conclusion

An experiment in straightforward engineering development has been made. From the time of agreement on a general scheme in June 1952 and the submission in July of a more detailed specification, no significant change in the scheme has been permitted. (One important addition has been made; this is provision to extend the 8-track 1024-word store to a 23-track 2968-word store. Relay switches are installed to pre-select, under programme control, any one of 16 tracks. The selected track can then be used as if it were one of the 8 electronically selected tracks.) Development and manufacture of the computer has proceeded substantially according to a plan prepared in August and having as its target the assembly of the machine by March 31st 1953, "so as to demonstrate its capabilities". It was envisaged that a "running-in" period would follow.

The degree of success or failure of the experiment will shortly be known. If it is successful we will have developed a range of "bricks" and built from them a dependable computer. Computer 401, if it is successful, will handle at fair speed a large range of the problems in mathematics, science and engineering design which are now awaiting machine time. The main achievement however will be the constructional system whereby computers of quite different logical design, for quite different applications, may be assembled from common bricks by staff not specially trained (and, if the bricks are made stock items), in a short time from logical design to commissioning. Because the bricks can be made by existing electronic production procedures and because any one computer breaks down into several items of repetition work, general purpose and special purpose computers using these bricks should be relatively low in cost. A Mk II version of Computer 401, now under consideration, with a modified or different logical design and some mechanical changes should be a very attractive low-cost general-purpose digital computer.

References

1. ELLIOTT, W. S., CARPENTER, H. G., and OWEN, C. E. "The Development of a range of digital computer components with special reference to packaged circuit units," Elliott Brothers Research Laboratory Report No. 303.

2. OWEN, C. E. Three standard circuits for serial digital computers, Elliott Brothers Research Laboratory Report No. 301.

3. ROBBINS, R. C. and HALSEY, J. R. Magnetostriction delay line storage, Elliott Brothers Research Laboratory Report No. 309.

4. ROBBINS, R. C. and MILLERSHIP, R. Applications of magnetostriction delay lines. Paper 27 read at this symposium.

37. A Medium-Size Decimal Computing Machine

by

N. Kitz

Istituto Nazionale per le Applicazioni del Calcolo, Italy

Introduction

The use of binary scale in computing machines, although widely accepted, has never met with universal support and a few pioneers in the field of automatic computation have always regarded it as a passing phase. The opinion has been considerably strengthened by the recent completion of large-scale decimal electronic computers like the UNIVAC and the Harvard Mark IV.

For many kinds of scientific computation, machines which need not perform elaborate conversion routines on each incoming or outgoing decimal digit present very substantial advantages.

The problems which come into this category are those which require few operations to be performed on vast amounts of information. Binary machines are inefficient for this type of work as virtually all the machine time is spent in performing conversions to and from decimal.

The machine described in this paper is meant to handle such computations and is to be applied to problems like: electrical networks, ballistics, stress and vibration analysis, Fourier synthesis, matrix manipulation, statistics, census, inventory records, actuarial studies, accounts, production planning, cost distribution, personnel records, PAYE, etc.

The main feature of this machine is its unusually large memory system consisting of a high-speed magnetic drum of 10 000 decimal digit capacity, normally supplemented by up to ten external stores on magnetic tapes having a total capacity of 10 000 000 decimal digits.

Though a powerful tool for computation, the machine has been designed for economy; its valve complement, diodes included, is not expected to reach 800 envelopes.

General Computational Organization

The installation would be based on a central computing unit aided by a number of auxiliary devices, the latter being chosen to render the installation either completely autonomous or to fit it into any existing computational organization.

This flexibility is obtained by making the computer itself operate exclusively with magnetic tape input/output and providing facilities for converting to this system any other form of data representation. It is envisaged that auxiliary mechanisms will be available for the following purposes:

1. Preparing magnetic tapes from a manual keyboard.
2. Preparing magnetic tapes from IBM or Remington-Rand cards.
3. Preparing magnetic tapes from punched paper tapes.
4. Preparing a typewritten record from magnetic tape.
5. Punching IBM or Remington-Rand cards from magnetic tape.
6. Duplicating magnetic tapes.

Description

The machine will operate throughout with decimal numbers treated serially. The method of representation adopted is the "excess of three" code already being successfully used in the UNIVAC.

The machine is programmed and obeys instructions of the two-address type, each order specifying the operation to be performed, the address of one operand and the memory location of the next order.

An important new feature is the inclusion of facilities for performing subroutines over and over again without any modification of their orders. As the type of storage used is permanent, it is proposed to have subroutines, particularly suited to the type of work being carried out, always available in an extension of the main store.

The operations which are "wired" into the machine are the basic arithmetical processes of addition, subtraction, and multiplication as well as the usual "logical" instructions.

Word Structure

Ten decimal digits are used to represent either a number or a coded instruction. In both cases the most significant digit denotes the sign.

Numbers (represented as absolute values) consist therefore of nine decimal digits; it is assumed by the arithmetic circuits that they lie in the range $-1 \leqslant n < +1$ and that the decimal point is fixed immediately to the right of the sign.

An instruction also consists of nine decimal digits; of these six are normally used to specify the two addresses, while the remaining three are used to identify the order.

Arithmetic Unit

The operations performed by the arithmetic unit are: addition, subtraction, multiplication, multiple left and right shifts, transfers from memory to multiplier and accumulator registers and vice versa, collation, conditional transfer, forward and reverse tape hunts, transfers to and from tapes, counting, dummy stop and normal stop.

For addition and subtraction, the computer applies an "overflow" control and refers to memory position 999 if a result exceeds the capacity of the Accumulator.

Multiplication is performed by means of a built-in "signed" multiplication routine requiring an average of 2.5 additions or subtractions per digit of the multiplier.

The use of the "excess of three" code facilitates checking against failure of the computing circuits. In this code the combinations 1111 and 0000 are not used. As the dropping or picking up of a whole series of pulses is one of the most common errors, the adder has been designed to test for these combinations and to stop the machine should they occur.

Internal Memory

The internal memory consists of a magnetic drum 12 in. in diameter and 3 in. long. Spaced at about one thousandth of an inch from the surface of the drum are twenty read/record heads, mounted in four stacks. A special screened stack houses the clock heads.

The part of the drum surface opposite each head is called a track; each track holds fifty words so that the total storage capacity is 1000 words.

The speed of the drum is 3600 r.p.m. and 2000 pulses are generated by it at each revolution giving the machine a basic frequency of 120 000 pulses/s.

External Memory

The machine may be provided with up to ten identical auxiliary memory units each storing 100 000 words of ten decimal digits.

The units handle about 1000 ft of tape. Each tape is divided into five magnetic channels of which one is used to carry a clock track while the other four are used to store words in batches of fifty. The tapes are operated in conjunction with a multiple read/record head, the pulse packing density being 100/in.

Facilities are provided for hunting on tape, this operation proceeding independently of computation. If a tape read or write order is encountered while a search is in progress, the machine loses cycles and does not proceed until the hunt has been completed.

Coding

Instructions are represented by nine digits and a sign. Generally speaking the main programme is written out with positive orders, while subroutines (if the automatic facility is required) are written out with negative orders. The machine can differentiate between the two types of instruction and automatically returns control to the main programme at the end of a subroutine. Normally if a subroutine is called in at address "n" in the main programme, the machine will, on completing the subroutine, return control to the order in memory position "n+1".

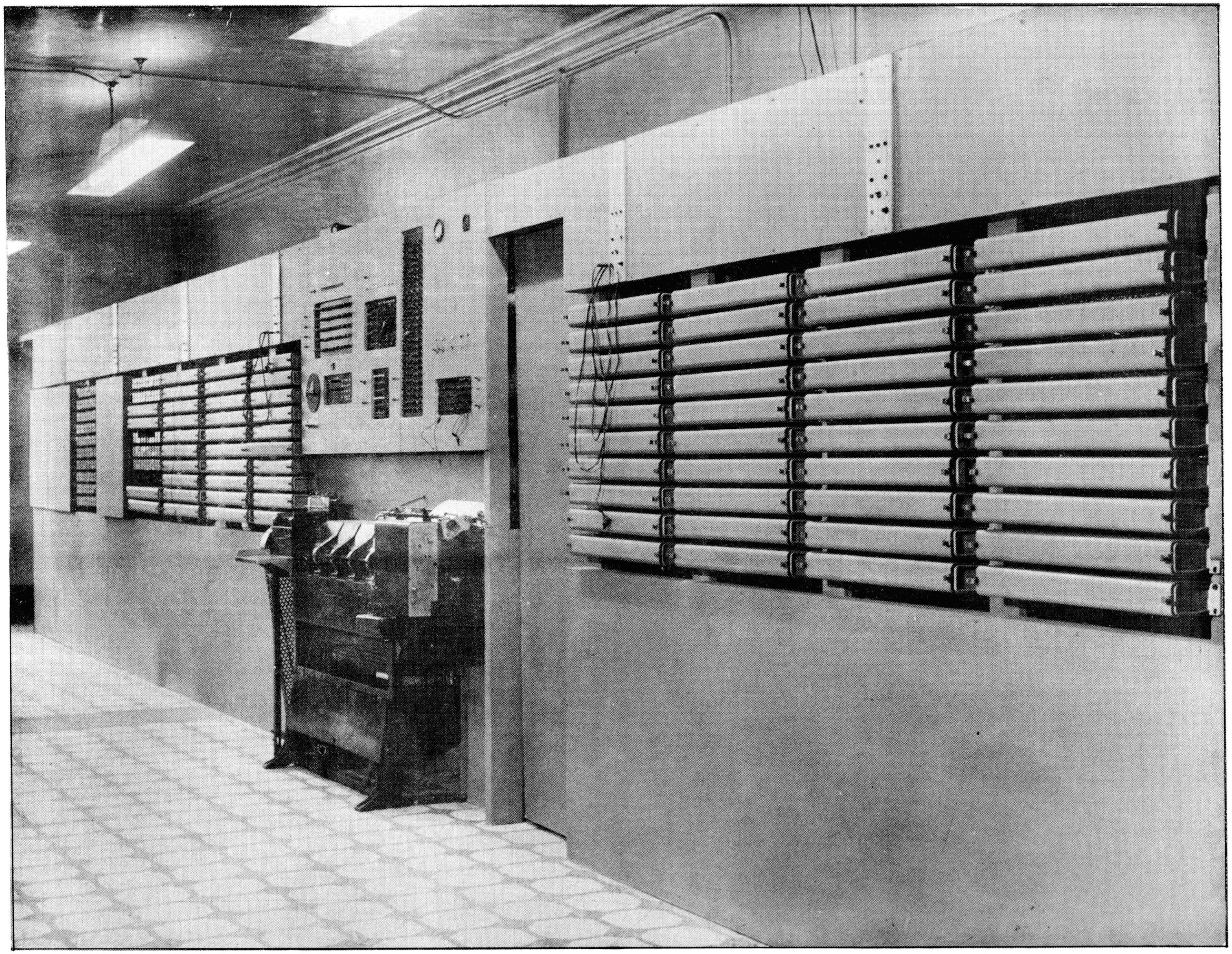
Manual Controls

The operation of the computer will be supervized from a control desk provided with sets of switches for introducing numbers into the machine and fitted with neon displays indicating the contents of registers.

These controls have the dual function of enabling the operator to supervize the work of the computer and of assisting in fault tracing, as by their use the correct functioning of the computer can be ascertained.

To facilitate replacement of faulty parts the machine will be constructed with plug-in standardized circuits and detachable panels.

It is hoped that the machine will be easy to handle from both the mathematical and engineering aspects and that it will prove useful in extending electronic computation to new fields.



PAPER 38

Fig.1. The Imperial College relay computer.

38. The design requirements of a low-cost computing machine

by

K. D. Tocher

Imperial College, London

My colleague Mr. Michaelson and I have been engaged in building a relay machine (*fig.* 1) at Imperial College. This machine (*fig.* 1) is about twice as big as the Harwell machine, about five times as fast and has certain other advantages. But I would rather spend the few minutes I have, not in describing that particular machine, since I think that relay machines are becoming rather antiquated, but in discussing the general problem of designing a machine.

The title of this section of the Symposium is 'Medium Size Machines' and our chairman has noted that this is a polite term for small machines. However, Dr. Booth has hit the nail on the head when he entitles his paper "The APE(X)C - a *low cost* computer." There are many small-scale organizations that could make very good use of a computer but are prevented from doing so by the high cost of these machines. To obtain a low-cost machine it is clear that something present in the big machines must be sacrificed, but it is not at all clear that this must be size.

Suppose the characteristics of an ideal machine are listed. It will then be possible to see what sacrifices from this standard will produce the greatest economies. The ideal machine will have

1. A very fast operating speed.
2. A huge store with fast access to all its members.
3. Fast, reliable and flexible input and output arrangements, (on the pattern described by Mr. Davies in the second session).
4. Extreme reliability.
5. Simple maintenance.
6. An easily learnt and easily used code.
7. An associated staff of programmers and engineers who are very efficient at keeping the machine busy on useful work.

Organizations which suffer from lack of money also suffer from lack of labour, and in general, labour is more difficult to obtain than lump sums of money, since labour constitutes a continuing commitment. It follows that the best sacrifices to make are those which reduce the running costs of the machine rather than those which reduce the initial building costs.

Thus our last requirement for an ideal machine must be relinquished; we must be prepared to have our machine idle for quite long periods and to have a programming staff largely recruited from people who actually want the results, ranging from geologists and physicists to biologists and economists. It follows also that requirement 6 must be emphasized; this implies that the machine should not use optimum coding. I am fully aware of the enormous advantages of this form of programming, but after coding both for machines which use it and those which do not, I consider that non-mathematically inclined people would find optimum programming more difficult than ordinary coding. I use the term optimum programming for that form of programming of a synchronous machine in which the instruction order is determined implicitly and I exclude the type of code which Dr. Wilkes calls 1 + 1. The latter gives the option of optimum coding whereas in the ACE Pilot Model consideration of the positioning of instructions in the store is essential.

If optimum coding is abandoned, then I think some consideration should be given to some engineering equivalent - some hardware we could incorporate which would produce decreases in operation time similar to those that result from optimum programming.

To cut the maintenance staff, it is clear that requirements 4 and 5 cannot be relaxed. Rather, more care must be taken over these items. It seems obvious that the slower any piece of equipment is expected to work, the more reliance can be placed on its operation, the more drift from its nominal condition can be permitted, and the less frequently corrective maintenance will be required. Hence it seems that requirement 1 should be sacrificed, but the effect of slowing down the basic rate of operation can be alleviated by parallel working. This will increase the initial cost of the machine but this increase will be more than offset by the decreased running costs. Also the cost of a parallel machine is, contrary to popular opinion, not so much greater than that of a serial machine. The cost of storage is almost unaffected since the differences are only in the mode of access to it. Any arithmetic unit with a built-in multiplier requires at least one staticizing register or short delay line, whereas a Von Neumann type arithmetic unit needs about 5 such registers.

The cost of the arithmetic unit in serial machines is such a small proportion of the total that a 5-fold increase in the cost of this item does not seriously alter the total cost. The control of a parallel machine is rather easier than that of a serial machine, since all the digits of a number are available for testing before the arithmetic begins; thus division, shifts by the contents of a store, and shifts to standard form are all quite easily organized.

We now turn to means of cheapening the initial building costs. The first step in this direction is to do without elaborate conversion equipment for input and output. Punched-card feed and print bank or reproducer punch are obviously ideal for a parallel machine and have the advantage that their speed is sufficient to eliminate the worse mis-matches of operating rates encountered with teleprinters or automatic typewriters. Disadvantages concerned with editing arrangements must be accepted as one of the prices paid for a cheap machine.

The size of the fast-access store must be kept as small as possible, and backing storage must be provided on drums or tape whichever is found most convenient. The remarks already made at this Symposium about echelon storage and hierarchies of stores are relevant here. For a very large number of problems the fast access storage is mostly required for instructions, the number of fast-access storage locations required for numbers being quite small. This raises the question whether there are any special characteristics of instruction storage that could be used to cheapen the cost of that storage. This would raise the heresy of separating the number and instruction stores. The combination of these stores arose from the flexibility given to the machine by enabling it to modify its own instructions by performing arithmetic upon them. The invention of the B-box enables these modifications to take place without altering the stored quantities and there are other *ad hoc* devices which will produce the same effects. The need for combination of the stores is now eliminated and with their separation a form of non-erasible storage could be used for instructions and the time-access characteristics can be arranged to be exactly as required.

In a relay machine, punched tape meets these requirements exactly since it is non-erasible and cheap, and access is instantaneous, followed by a dead time while the tape is moved forward one row (in general, instructions are read in order). This is different from a synchronous form of storage such as on a drum in which the successive words can be used at regularly spaced intervals. A parallel machine is most easily constructed asynchronous and then the time-access characteristics of tape are ideal.

Of course, punched tape, mechanically moved, is not practicable for an electronic computer, and Mr. Michaelson and I have searched for some equivalent with no moving parts. We believe that a cross-bar system, using pins to sense the paper tape and small low-capacity metal rectifiers to eliminate feedback, can be built at very low cost indeed. If we change the paper tape into little hard-board strips with holes punched in them, we find that we can lay our programmes out by assembling these strips in a tray and inserting it into the cross-bar system. The strips can be set up about half as fast as an unskilled operator can use a teleprinter, and although it will not be as convenient for the initial input, there will be some advantage when a programme needs to be modified to remove errors.

The store is an extreme form of the type that Professor Hartree mentioned in his introductory address, one in which the access time is fast but the input time is slow.

The use of non-erasible storage prevents the programming of programmes, the modification of programmes by programmes for checking and so on, but this is a minor inconvenience that must be tolerated in a cheap machine.

I now return to the code used in a machine. The separation of the two storage systems means that the word length for an instruction can be settled by considerations other than the accuracy required of the machine. The arguments between the supporters of the two principal codes are based on economy of storage; neither are perfect in this respect. The three-address code involves a redundancy of operations to transport numbers around the machine. An alternative which overcomes both these objections is the code used in our relay machine. In this, each basic operation, such as addition, has four versions in which either the contents of the arithmetic unit or that of a store can be used for the second operand while the result may be copied from the arithmetic unit or not. Thus, if X, Y, Z represent stores and β the appropriate register of the arithmetic unit, the four versions available are symbolised as $X + \beta$, $X + \beta \rightarrow Z$, $X + Y$, $X + Y \rightarrow Z$. All unnecessary transportations of numbers and all unnecessary operations are eliminated; we have called this a variable-address code.

In a parallel machine working a three-address code each operation will consist of setting two registers on the arithmetic unit, letting this produce the result and finally putting the result out to a store. If only a single highway between store and arithmetic unit is provided (and this would be so in a cheap machine) then the parts of the instruction giving the operation to be performed, and the names of the stores involved are not all required at once but are needed in order. If the order is $X + Y \quad Z$, the first part required is '+', then the store name X, then the name Y, and finally the name Z. This enables a serio-parallel input of the instruction to the control of the machine to be used, and reduces the amount of equipment about four-fold. This same mode of input is ideally suited to the variable-address code as it solves the problem of variable length words for the different version of each instruction. In the code each instruction consists of a group of words, the first giving the operation and the following ones such names as are required. The control of the machine draws these from the store in order as it requires them.

The code has been used for the relay machine by a group of programmers with widely varying mathematical talents and most of them have agreed that this mode of coding eases the problem of programming.

To sum up then, I would suggest that if an organization requires to build a comparatively cheap computer with low running costs, it should build a parallel machine, working at a low repetition rate, using a Von Neumann type of arithmetic unit with a limited amount of parallel number storage backed by magnetic drum or tape and with a separate non-erasible instruction store, and using Hollerith equipment for input and output. It should have a semi-parallel mode of input of instructions, making use of a variable-address code.

APPENDIX I

ALPHABETICAL LIST OF ORGANIZATIONS REPRESENTED AND THEIR DELEGATES

Acoustic Products, Ltd., Stonefield Way, South Ruislip, Middx.

- Connelly, Dr. F. C.

Admiralty Research Laboratory, Teddington, Middx.

- Beale, E. M. L.
- Hobson, A.
- Lee, Dr. E.
- Nichols, L. H. F.
- Owen, G.
- Steel, F.
- Vajda, Dr. S.
- Wilson, Dr. E. M.

All Power Transformers Ltd., Byfleet, Surrey

- Rubach, A.

Armament Research Establishment, Fort Halstead, Sevenoaks, Kent

- Dodd, K. N.
- Gawlik, H. J.
- Maccoll, Dr. J. W.
- Thornhill, C. K.

Atomic Energy Research Establishment, Harwell, Berks.

- Barnes, R. C. M.
- Cooke-Yarborough, E. H.
- Fossey, E. B.
- Howlett, Dr. J.

Australian Scientific Liaison Office, Africa House, Kingsway, London, WC.2.

- Blunden, R.

Belgium, 104 rue du Pere Devraye, Woluwe St. Pierre,

- Franckx, Prof. E.

Birkbeck College, London

- Booth, Dr. A. D.

Birmingham University

- Redshaw, Prof, S. C.

Bristol Aeroplane Co., Filton, Bristol

- Hahn, J. M.

British Tabulating Machine Co., Letchworth, Herts.

Bird, R.
Cartwright, R.
Dagnall, B.
Dickens, C. D. C.
Holland-Martin, C. G.
Michaelson, R. L.
Townsend, R.
Womersley, J. R.

Cambridge University Mathematical Laboratory

Barton, S. A.
Douglas, A. S.
Friedman, Dr.
Gill, S.
Leigh, Dr. D. C. F.
Miller, Dr. J. C. P.
Mutch, E. N.
Phister, M.
Renwick, W.
Stevens, G. J.
Stringer, J. B.
Wilkes, Dr. M. V.
Willis, D. W.

Cambridge University (Department of Applied Economics)

J. A. C. Brown
Prais, S. J.

Cavendish Laboratory, Cambridge

Hartree, Prof. D. R.

Consejo Superior de Investigaciones Cientificas, Serrano 119, Madrid, Spain

Santesmases, J. G.
Vidal, M. R.

Consiglio Nazionale delle Richerche, 7 Piazzale delle Scienze, Rome, Italy

de Finetti, Prof. B.
Rodino, Dr. G.

A. C. Cossor Ltd., Highbury Grove, London, N.5

Batty, N. G.

Courtaulds Research Institute, Maidenhead, Berks.

Crank, J.

De Havilland Aircraft Company Ltd., Hatfield, Herts.

Hunt, P.

De Havilland Propellers, Ltd., Hatfield, Herts

Chadwick, E. G.
Copland, Miss M. F.
Evans, L. G.
MacMullan, J. S.
Stonell, A. C.
Swingler, J.

Department of Scientific and Industrial Research, Charles House, 5-11, Regent St., London, SW. 1.

Greenall, P. D.

Elliott Bros. (London) Ltd., Elstree Way, Borehamwood, Herts

Carpenter, H. G.
De Barr, A. E.
Devonald, C. H.
Elliott, W. S.
Hersom, S. E.
Hill, N. D.
Owen, C. E.
Robbins, R. C.
St. Johnston, A.
Wykeham, W. A. P.

E. M. I. Engineering Development Ltd., Hayes, Middx.

Spencer, R. E.

E. M. I. Research Laboratories Ltd., Hayes, Middx.

Cork, E. C.
Harker, M. G.
Huntley, K. G.
Scantlebury, G. S.
Vanderlyn, P. B.
White, E. L. C.

English Electric Co., Stafford

Haley, A. C. D.
Robinson, C.
Scott, W. E.

Fairey Aviation Co. (Research Division), Heston, Middx.

ApSimon, Dr. H.
Quinn, Miss S.

Ferranti Ltd., Moston, Manchester, 10

Bennett, Dr. J. M.
Bowden, Dr. B. V.
Carter, J. D.
Gradwell, C. F.
Pollard, B. W.
Robinson, Dr. A. A.
Swann, B. B.

General Reinsurance Company of Amsterdam, 24-25 Fenchurch St., London, EC.3.

Monic, B.

H.M. Nautical Almanac Office, Herstmonceux Castle, Hailsham, Sussex

Sadler, D. H.

I.B.M. France, 5 Place Vendome, Paris 1, France

Ghertman, I.

I.B.M. United Kingdom, Ltd., 17 Berkeley St., London, W.1.

Hudson, T. C.
Swann, E. D.

IBM World Headquarters, 590 Madison Ave., New York 22, NY, USA.

McPherson, J. C.

Imperial College, Exhibition Rd., S. Kensington, London, SW.7.

Bickley, Prof. W. G.
Lehman, M.
Michelson, S.
Rocher, Dr. K. D.

Indian Scientific Liaison Office, Africa House, Kingsway, London, WC.2.

Pasricha, Lt. Col. C. L.

Istituto Nazionale per le Applicazione del Calcolo, Piazzale delle Scienze 7, Rome, Italy

Kitz, N.

Kings College, Strand, London, WC.2.

Barnett, Dr. M. P.

Koninklijke/Shell - Laboratorium, Badhuisweg 3, Amsterdam N, Holland

Lauwerier Dr. H. A.
Lunbeck, Dr. R. J.

Laboratoire Central de l'Armement, 1 Place Saint Thomas d'Aquin, Paris (7), France

Bosset, J.
Ceschino, F.
Jeanniot, J.
Sestier, A. L.

Liverpool University

Jones, Dr. C. W.
Rosenhead, Prof. L.

J. Lyons & Co., Cadby Hall, London, W.14

Caminer, D. T.
Kaye, E. J.
Lenaerts, E. H.
Pinkerton, Dr. J. M. M.
Thompson, T. R.

Manchester University

Brooker, Dr. R. A.
Chaplin, G. B. B.
Grimsdale, R. L.
Hoskin, N. E.
Kilburn, Dr. T.
Livesey, R. K.
Turing, Dr. A. M.
Williams, Prof. F. C.

Mathematisch Centrum, 2e Boerhaavestraat 49, Amsterdamo, Holland

Loopstra, B. J.
Wijngaarden, Dr. A. van

Military College of Science, Shrivenham, Wilts

Holt-Smith, Prof. C.
Preston, G. B.
Toothill, G. C.

Ministry of Civil Aviation, 19/29 Woburn Place, London, WC.1

Johnson, M. H.

Ministry of Supply, DWR(D), Shell Mex House, Strand, London, WC.2

Hinds, Brig, G. H.

Ministry of Supply, Fort Halstead, Sevenoaks, Kent

Corner, Dr. J.
Glennie, A. E.

Ministry of Supply, RRDE, Gt. Malvern, Worcs.

Blundell, P.
Deighton, R.

Ministry of Supply, Statistical Advisory Unit, Carlton Hotel, Haymarket, London, SW.1.

Fieller, Dr. E. C.

Mullard Research Laboratories, Cross Oak Lane, Salfords, Redhill, Surrey

Braybrook, C. H.
Dell, H. A.
Goldsmith, M. N.
Trier, P. E.

Murphy Radio, Ltd., Welwyn Garden City, Herts

Lovelock, R. T.
McMullin, T. A.

National Bureau of Standards, U.S.A.

Blanch, Dr. Gertrude
Slutz, Dr. R. J.

National Physical Laboratory, Teddington, Middx.

Allway, G. G.
Clayden, D. O.
Colebrook, F. M.
Davies, D. W.
Fox, Dr. L.
Goodwin, Dr. E. T.
Newman, E. A.
Robertson, Dr. H. H.
Wilkinson, J. H.
Wright, M. A.

National Research Development Corporation, 1 Tilney St., London, W.1.

Crawley, H. J.
Strachey, C.
Hennessey, D.

New Zealand (At present Dr. Woods is working in the Aerodynamics Division, NPL)

Woods, Dr. L. C.

Northampton Polytechnic, St. John St., London, EC. 1.

Bridger, M.
Harrison, L. H. R.

Olivetti, Ivrea, Italy

Insolera, D.

N.V. Philips' Gloeilampenfabrieken, Eindhoven, Holland

Heyn, H.
Nijenhuis, Ir. W.

The Plessey Company Ltd., Coombe Rd., Croydon, Surrey

Ringrose, J. W.
Sabel, C.

Pollak, Mercer & Tench, 134 Cheapside, London, EC.2.

Maddox, H. F.

Post Office Research Station, Dollis Hill, London, NW.2.

Chandler, W. W.
Coombs, Dr. A. W. M.

Powers-Samas Accounting Machines, Ltd., Aurelia Rd., Croydon, Surrey

Johnson, W. E.

Private Individuals

Bailey, C. E. G., 59A Kensington Mansions, London, SW. 5.
Olsson, M., Svenska Flygmotor Aktiebolaget, Trollhätan, Sweden.
Southwell, Sir Richard, The Old House, Trumpington, Cambridge.

Research Institute of National Defence, Stockholm 5, Sweden

Agerberg, J.
Bergman, C. I.
Löfgren, L.

Rolls Royce Ltd., Derby

Griffiths, L.

Rothamsted Experimental Station, Harpenden, Herts.

Healy, M. J. R.

Royal Aircraft Establishment, Farnborough, Hants.

Birchall, P. C.
Cohen, A.
Cork, G.
Hollingdale, Dr. S. H.
Palmer, E.
Petherick, E. J.
Routledge, N.
Rowley, G. C.

Royal Naval College, Greenwich

Brown, B. M.
Nicholson, Prof. G. F.

Scientific Computing Service, Ltd., 23 Bedford Sq., London, WC. 1.

Gilles, Dr. D. C.

Shell Development Co., 3737 Bellaire Boulevarde, Houston 25, Texas, USA.

Kaufman, S.

Shell Petroleum Co. Ltd., 16 Finsbury Circus, London, EC. 2.

Sears, G. W.

Société d'Electronique et d'Automatisme, 138 Bde. de Verdum, Courbevoie, Seine, France

Albin, J.
Dussine, R.
Namian, P.

Standard Telecommunication Laboratories, Ltd., Gt. Cambridge Rd., Enfield, Middx.

Harrild, P. W. S.
Rice, J.
Wright, E. P. G.

Standard Telephones & Cables, Ltd., Connaught House, London, WC.2.

Branch, M. C.
Hartley, G. C.

Swedish Board for Computing Machinery, Drottninggatan 95A, Stockholm 6, Sweden

Dahlquist, G.

Telecommunications Research Establishment, Malvern, Worcs.

Carter, R. H. A.
Perry, G. H.
Pincherle, Dr. L.
Taylor, P.
Uttley, Dr. A. M.

University College, Gower St., London, WC.1.

Buckingham, Dr. R. A.
Erskine, Dr. G. A.
Hartley, Dr. H. O.
Massey, Prof. H. S. W.
Roberts, F.
Thompson, Prof. E. H.

United States Navy, Keysign House, 429 Oxford St., London, W.1.

Weber, R. R.

University of Illinois, Urbana, Illinois, USA.

Hulsizer, Dr. R.

Vickers-Armstrongs, Ltd., Crayford, Kent

Guttridge, Major E. J.
Hitch, H. P. Y.
Lucas, J. H.
Yandell, R. P. B.

APPENDIX II

ALPHABETICAL LIST OF DELEGATES

(To find the address of any organization, refer to Appendix I)

Agerberg, J., Research Institute of National Defence, Sweden.
Albin, J., Societe d'Electronique et d'Automatisme, France.
Allway, G. G., NPL, Teddington.
ApSimon, Dr. H., Fairey Aviation Co.

Bailey, C. E. G., 59A Kensington Mansions, London, S.W.5.
Barnes, R. C. M., Armament Research Establishment, Fort Halstead.
Barnett, Dr. M. P., Kings College, London.
Barton, S. A., Cambridge University Mathematical Laboratory.
Batty, N. G., Messrs. A. C. Cossor Ltd.
Beale, E. M. L., Admiralty Research Laboratory, Teddington.
Bennett, Dr. J. M., Messrs. Ferranti Ltd.
Bergman, C. I., Research Institute of National Defence, Sweden.
Bickley, Prof. W. G., Imperial College, London.
Birchall, P. C., Royal Aircraft Establishment, Farnborough.
Bird, R., British Tabulating Machine Co.
Blanch, Dr. Gertrude, National Bureau of Standards, USA.
Blundell, P., Ministry of Supply, RRDE.
Blunden, R., Australian Scientific Liaison Office, London.
Booth, Dr. A. D., Birkbeck College, London.
Bosset, J., Laboratoire Central de l'Armement, France.
Bowden, Dr. B. V., Messrs. Ferranti Ltd.
Branch, M. C., Standard Telephones and Cables, Ltd.
Braybrook, C. H., Mullard Research Laboratories.
Bridger, M., Northampton Polytechnic, London.
Brooker, Dr. R. A., Manchester University.
Brown, B. M., Royal Naval College, Greenwich.
Brown, J. A. C., Cambridge University (Department of Applied Economics).
Buckingham, Dr. R. A., University College, London.

Caminer, D. T., Messrs. J. Lyons & Co.
Carpenter, H. G., Messrs. Elliott Bros. (London) Ltd.
Carter, J. D., Messrs. Ferranti Ltd.
Carter, R. H. A., Telecommunications Research Establishment, Malvern.
Cartwright, R., British Tabulating Machine Co.
Ceschino, F., Laboratoire Central de l'Armement, France.
Chadwick, E. G., De Havilland Propellers, Ltd.
Chandler, W. W., Post Office Research Station, London.
Chaplin, G. B. B., Manchester University.
Clayden, D. O., NPL, Teddington.
Cohen, A., Royal Aircraft Establishment, Farnborough.
Colebrook, F. M., NPL, Teddington.
Connelly, Dr. F. C., Acoustics Products, Ltd.
Cooke-Yarborough, E. H., Atomic Energy Research Establishment, Harwell.
Coombs, Dr. A. W. M., Post Office Research Station, London.
Copland, Miss M. F., De Havilland Propellers, Ltd.
Cork, E. C., E.M.I. Research Laboratories Ltd.
Cork, G., Royal Aircraft Establishment, Farnborough.
Corner, Dr. J., Ministry of Supply, Fort Halstead.
Crank, J., Courtaulds Research Institute, Maidenhead.
Crawley, H. J., National Research Development Corporation, London.

Dagnall, B., British Tabulating Machine Co.
Dahlquist, G., Swedish Board for Computing Machinery.
Davies, D. W., NPL, Teddington.
De Barr, A. E., Messrs. Elliott Bros. (London) Ltd.
de Finetti, Prof. B., Consiglio Nazionale delle Richerche, Italy.
Deighton, R., Ministry of Supply, RRDE.
Dell, H. A., Mullard Research Laboratories.
Devonald, C. H., Messrs. Elliott Bros. (London) Ltd.
Dickens, C. D. C., British Tabulating Machine Co.
Dodd, K. N., Armament Research Establishment, Fort Halstead.
Douglas, A. S., Cambridge University Mathematical Laboratory.
Dussine, R., Societe d'Electronique et d'Automatisme, France.

Elliott, W. S., Messrs. Elliott Bros. (London) Ltd.
Erskine, Dr. G. A., University College, London.
Evans, L. G., De Havilland Propellers, Ltd.

Feiller, Dr. E. C., Ministry of Supply, Statistical Advisory Unit.
Fossey, E. B., Atomic Energy Research Establishment, Harwell.
Fox, Dr. L., NPL, Teddington.
Franckx, Prof. E., Belgium.
Friedman, Dr.

Gawlik, H. J., Armament Research Establishment, Fort Halstead.
Ghertman, I., IBM, France.
Gill, S., Cambridge University Mathematical Laboratory.
Gilles, Dr. D. C., Scientific Computing Service, Ltd.
Glennie, A. E., Ministry of Supply, Fort Halstead.
Goldsmith, M. N., Mullard Research Laboratories.
Goodwin, Dr. E. T., NPL, Teddington.
Gradwell, C. F., Messrs. Ferranti, Ltd.
Greenall, P. D., DSIR.
Griffiths, L., Rolls Royce.
Grimsdale, R. L., Manchester University.
Guttridge, Major E. J., Vickers Armstrongs, Ltd.

Hahn, J. M., Bristol Aeroplane Co.
Haley, A. C. D., English Electric Co.
Harker, M. G., EMI. Research Laboratories Ltd.
Harrild, P. W. S., Standard Telecommunication Laboratories, Ltd.
Harrison, L. H. R., Northampton Polytechnic, London.
Hartley, G. C., Standard Telephones & Cables, Ltd.
Hartley, Dr. H. O., University College, London.
Hartree, Prof. D. R., Cavendish Laboratory, Cambridge.
Healy, M. J. R., Rothamsted Experimental Station, Harpenden.
Hennessey, D., National Research Development Corporation, London.
Hersom, S. E., Messrs. Elliott Bros. (London) Ltd.
Heyn, H., N.V. Philips' Gloeilampenfabrieken, Holland.
Hill, N. D., Messrs. Elliott Bros. (London) Ltd.
Hinds, Brig. G. H., Ministry of Supply, DWR(D).
Hitch, H. P. Y., Vickers Armstrongs, Ltd.
Hobson, A., Admiralty Research Laboratory, Teddington.
Holland-Martin, C. G., British Tabulating Machine Co.
Hollingdale, Dr. S. H., Royal Aircraft Establishment, Farnborough.
Holt-Smith, Prof. G., Military College of Science, Shrivenham.
Hoskin, N. E., Manchester University.
Howlett, Dr. J., Atomic Energy Research Establishment, Harwell.
Hudson, T. C., IBM, United Kingdom, Ltd.
Hulsizer, Dr. R., University of Illinois, USA.
Hunt, P., De Havilland Aircraft Co. Ltd.
Huntley, K. G., E.M.I. Research Laboratories Ltd.

Insolera, D., Messrs. Olivetti, Italy.

Jeanniot, J., Laboratoire Central de l'Armement, France.
Johnson, M. H., Ministry of Civil Aviation.
Johnson, W. E., Powers-Samas Accounting Machines, Ltd.
Jones, Dr. C. W., Liverpool University.

Kaufman, S., Shell Development Co., USA.
Kaye, E. J., Messrs. J. Lyons & Co.
Kilburn, Dr. T., Manchester University.
Kitz, N., Istituto Nazionale per le Applicazione del Calcolo, Italy.

Lauwerier, Dr. H. A., Koninklijke/Shell - Laboratorium Amsterdam.
Lee, Dr. E., Admiralty Research Laboratory, Teddington.
Lehman, M., Imperial College, London.
Leigh, Dr. D. C. F., Cambridge University Mathematical Laboratory.
Lenaerts, E. H., Messrs. J. Lyons & Co.
Livesey, R. K., Manchester University.
Löfgren, L., Research Institute of National Defence, Sweden.
Loopstra, B. J., Mathematisch Centrum, Amsterdam.
Lovelock, R. T., Murphy Radio, Ltd.
Lucas, J. H., Vickers-Armstrongs, Ltd.
Lunbeck, Dr. R. J., Koninklijke/Shell - Laboratorium, Amsterdam.

Maccoll, Dr. J. W., Armament Research Establishment, Fort Halstead.
MacMullan, J. S., De Havilland Propellers, Ltd.
Maddox, H. F., Messrs. Pollak, Mercer & Tench.
Massey, Prof. H. S. W., University College, London.
McMullin, T. A., Murphy Radio, Ltd.
McPherson, J. C., IBM World Headquarters, New York.
Michaelson, R. L., British Tabulating Machine Co.
Michelson, S., Imperial College, London.
Miller, Dr. J. C. P., Cambridge University Mathematical Laboratory.
Monic, B., General Reinsurance Company of Amsterdam.
Mutch, E. N., Cambridge University Mathematical Laboratory.

Namian, P., Societe d'Electronique et d'Automatisme, France.
Newman, E. A., NPL, Teddington.
Nichols, L. H. F., Admiralty Research Laboratory, Teddington.
Nicholson, Prof. G. F., Royal Naval College, Greenwich.
Nijenhuis, Ir. W., N.V. Philips' Gloeilampenfabrieken, Holland.

Olsson, M., Svenska Flygmotor Aktiebolaget, Tröllhatan, Sweden.
Owen, C. E., Messrs. Elliott Bros. (London) Ltd.
Owen, G., Admiralty Research Laboratory, Teddington.

Palmer, E., Royal Aircraft Establishment, Farnborough.
Pasricha, Lt. Col. C. L., Indian Scientific Liaison Office, London.
Perry, G. H., Telecommunications Research Establishment, Malvern.
Petherick, E. J., Royal Aircraft Establishment, Farnborough.
Phister, M., Cambridge University Mathematical Laboratory.
Pincherle, Dr. L., Telecommunications Research Establishment, Malvern.
Pinkerton, Dr. J. M. M., Messrs. J. Lyons & Co.
Pollard, B. W., Messrs. Ferranti Ltd.
Prais, S. J., Cambridge University (Department of Applied Economics)
Preston, G. B., Military College, Shrivenham.

Quinn, Miss S., Fairey Aviation Co.

Redshaw, Prof. S. C., Birmingham University.
Reiners, Mrs. C. A., Ministry of Supply, WRIDI (1a).
Renwick, W., Cambridge University Mathematical Laboratory.
Rice, J., Standard Telecommunication Laboratories, Ltd.
Ringrose, J. W., The Plessey Company Ltd.
Robbins, R. C., Messrs. Elliott Bros. Ltd.
Roberts, F., University College, London.
Robertson, Dr. H. H., NPL, Teddington.
Robinson, Dr. A. A., Messrs. Ferranti Ltd.
Robinson, C., English Electric Co.
Rodino, Dr. G., Consiglio Nazionale delle Richerche, Italy.
Rosenhead, Prof. L., Liverpool University.
Routledge, N., Royal Aircraft Establishment, Farnborough.
Rowley, G. C., Royal Aircraft Establishment, Farnborough.
Rubach, A., All Power Transformers Ltd.

Sabel, C., The Plessey Company Ltd.
Sadler, D. H., H.M. Nautical Almanac Office.
Santesmases, J. G., Consejo Superior de Investigaciones Cientificas, Spain.
Scantlebury, G. S., E.M.I. Research Laboratories Ltd.
Scott, W. E., English Electric Co.,
Sears, G. W., Shell Petroleum Co. Ltd.
Sestier, A. L., Laboratoire Central de l'Armement, France.
Slutz, Dr. R. J., National Bureau of Standards, USA.
Southwell, Sir Richard, The Old House, Trumpington, Cambridge.
Spencer, R. E., E.M.I. Engineering Development Ltd.
Steel, F., Admiralty Research Laboratory, Teddington.
Stevens, G. J., Cambridge University Mathematical Laboratory.
St. Johnston, A., Messrs. Elliott Bros. (London) Ltd.
Stonell, A. C., De Havilland Propellers, Ltd.
Strachey, C., National Research Development Corporation, London.
Stringer, J. B., Cambridge University Mathematical Laboratory.
Swann, B. B., Messrs. Ferranti Ltd.
Swann, E. D., IBM United Kingdom, Ltd.
Swingler, J., De Havilland Propellers, Ltd.

Taylor, P., Telecommunications Research Establishment, Malvern.
Thompson, Prof. E. H., University College, London.
Thompson, T. R., Messrs. J. Lyons & Co.
Thornhill, C. K., Armament Research Establishment, Fort Halstead.
Tocher, Dr. K. D., Imperial College, London.
Toothill, G. C., Military College of Science, Shrivenham.
Townsend, R., British Tabulating Machine Co.
Trier, P. E., Mullard Research Laboratories.
Turing, Dr. A. M., Manchester University.

Uttley, Dr. A. M., Telecommunications Research Establishment, Malvern.

Vajda, Dr. S., Admiralty Research Laboratory, Teddington.
Vanderlyn, P. B., E.M.I. Research Laboratories Ltd.
Vidal, M. R., Consejo Superior de Investigaciones Cientificas, Spain.

Weber, R. R., United States Navy.
White, E. L. C., E.M.I. Research Laboratories Ltd.
Wijngaarden, Dr. A. van, Mathematisch Centrum, Amsterdam.
Wilkes, Dr. M. V., Cambridge University Mathematical Laboratory.
Wilkinson, J. H., NPL, Teddington.
Williams, Prof. F. C., Manchester University.
Willis, D. W., Cambridge University Mathematical Laboratory.
Wilson, Dr. E. M., Admiralty Research Laboratory, Teddington.
Womersley, J. R., British Tabulating Machine Co.
Woods, Dr. L. C., New Zealand
Wright, E. P. G., Standard Telecommunication Laboratories, Ltd.
Wright, M. A., NPL, Teddington.
Wykeham, W. A. P., Messrs. Elliott Bros. (London) Ltd.

Yandell, R. B. B., Vickers-Armstrongs, Ltd.

DS 14567/1/Wt.4471 K.6 3/54 DL

PRINTED AND PUBLISHED BY HER MAJESTY'S STATIONERY OFFICE
To be purchased from
York House, Kingsway, LONDON, W.C.2 423 Oxford Street, LONDON, W.1
P.O. Box 569, LONDON, S.E.1
13a Castle Street, EDINBURGH, 2 1 St. Andrew's Crescent, CARDIFF
39 King Street, MANCHESTER, 2 Tower Lane, BRISTOL, 1
2 Edmund Street, BIRMINGHAM, 3 80 Chichester Street, BELFAST
or from any Bookseller

1954

Price £1 1s 0d net

PRINTED IN GREAT BRITAIN

S.O. Code No. 48-126*