

The Ferranti Perseus Data-Processing System

by P. M. Hunt

Summary: The Ferranti Perseus Data-Processing system is described in detail. This is a new system designed especially for large scale data-processing operations. The facilities introduced into the equipment to make it suitable for such work are explained and the reason for their inclusion stated. The method of automatic checking used in the system is also discussed.

INTRODUCTION

The Ferranti Perseus Data-Processing System consists of two distinct parts capable of independent operation. The first part has as its central item the Perseus computer. This is a new, large, electronic digital computer designed for data-processing and commercial work in general. The computer is built from a range of plug-in packages many of which are the same as those used in the Pegasus computer, and the reliability of which is known. The second part is a line-at-a-time printer capable of printing the content of a reel of Perseus magnetic tape, using a Samastronic print head.

THE PERSEUS STORES

The internal store of Perseus (usually called the *computing store*) is physically constructed of nickel delay lines of two types. Part of the computing store consists of lines holding one word of information (Fig. 1), and the remainder consists of lines holding 16 words (Fig. 2). The store is divided into blocks of 32 words; there are 5 blocks (160 words) held in single-word lines and, in the basic machine, 27 blocks (864 words) held in 16-word lines (there being two 16-word lines per block). It is possible to add another 32 blocks (1,024 words) held in 16-word lines if required. There is immediate access to information stored in the single-word lines; the maximum access time in the 16-word lines is approximately 4 milliseconds, the average access time being 2 milliseconds. Conventionally the two stores are used in a particular manner for data-processing operations; this is described later. The first block of the store consists mainly of special registers of which those with addresses 0, 1, 2, 3, 4, 5, 6, 7, have special properties and are known as *qualifiers*.

THE PERSEUS WORD

Each word contains 12 alpha-numeric characters, each character being held as 6 binary digits according to a specific code (see Table 1). Thus, for example, the decimal number 514358012933 is held in a Perseus word as,

000101	000001	000100	000011	000101	001000	000000	000001	000010	001001	000011	000011
5	1	4	3	5	8	0	1	2	9	3	3

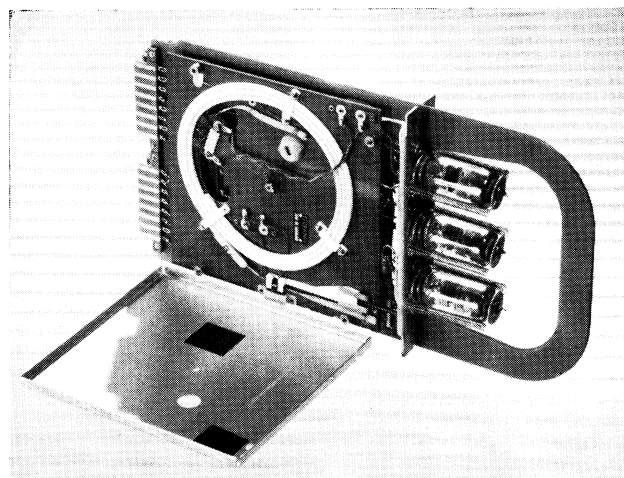


FIG. 1.—A single-word nickel delay line.

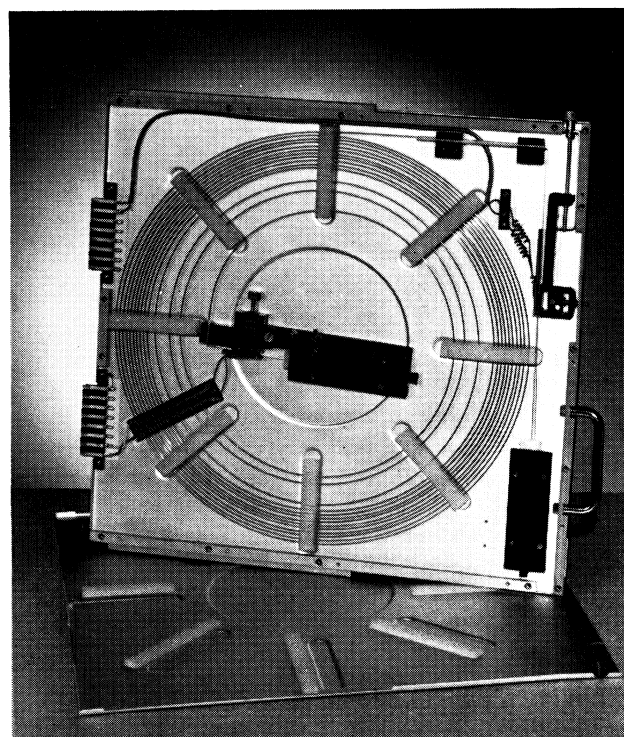


FIG. 2.—A sixteen-word nickel delay line.

and the sterling quantity £49951/19/11 as

				000100	001001	001001	000101	000001	000001	001001	001011						
				4	9	9	5	1	1	9	11						
.....				£				<<		s		<<		d		>	
(note that pence is contained in one character position).																	

The address 7 REGENT ST. is held as

000111	110000	101001	010101	010111	010101	100101	110010	110000	110001	110010	100000
7	Sp	R	E	G	E	N	T	Sp	S	T	

TABLE 1
THE PERSEUS COMPUTER AND PRINTER CODE

Six-bit character	Significance	Six-bit character	Significance
000000	0	100000	Decimal point
0001	1	0001	J
0010	2	0010	K
0011	3	0011	L
0100	4	0100	M
0101	5	0101	N
0110	6	0110	O
0111	7	0111	P
1000	8	1000	Q
1001	9	1001	R
1010	10	1010	Special characters
1011	11	1011	
1100	12	1100	
1101	Special characters	1101	
1110		1110	Space
1111		1111	
010000	Minus	110000	
0001	A	0001	S
0010	B	0010	T
0011	C	0011	U
0100	D	0100	V
0101	E	0101	W
0110	F	0110	X
0111	G	0111	Y
1000	H	1000	Z
1001	I	1001	Special characters
1010	Special characters	1010	
1011		1011	
1100		1100	
1101	Special characters	1101	Special characters
1110.		1110	
1111		1111	

There are thus $12 \times 6 = 72$ bits in a Perseus word; these, with a gap character of 6 bits, make 78 bits in all. Since the digit rate is 3 microseconds the word time is $3 \times 78 = 234$ microseconds.

In a total Perseus block it is therefore possible to store $32 \times 12 = 384$ alpha-numeric characters and, in insurance work, for example, this is used to describe one policy (Fig. 3). In other applications the item size is not necessarily one block—there may be, for example, particulars of 4 items per block. Instead of containing 12 alpha-numeric characters it is possible for a Perseus word to contain 3 orders. These orders are held in binary form and each consists of 24 bits. The significance of a typical order is shown in Fig. 4.

MIXED-RADIX ARITHMETIC

It is possible on Perseus to carry out mixed-radix arithmetic, so that conversion to and from binary is not necessary when information is read into or written out of the computer. For example, the computer can work in the decimal scale, or in sterling, or in hours, minutes, seconds, etc. Once the computer is set to work in one radix (e.g. decimal) it will continue to work in this radix until it is informed that the radix must be changed. For addition and subtraction, the radix in which the computer is to work is defined by the content of one storage location—the *radix register*—and the arithmetic is carried out in a special accumulator. To change from working in one radix to working in another it is only

	0	1	2	3	4	5	6	7	8	9	10	11													
0	POLICY NUMBER											SPLITS													
1	OFFICE CODE			MODE OF PAYMENT			TOTAL INSTALMENT PREMIUM																		
2	S.V. CODE		SPECIAL DEALINGS		DATE NEXT PREM DUE		TOTAL BONUS																		
3	DATE LAST PREM PAID				LOAN DUE DATE		EXISTING LOAN DEBT																		
4	TABLE CODE		CUR-RENCY		ANNUAL DUE DATE		ACCUMULATION OF PREMIUMS ACC BEN PREMIUM																		
5	NEW BONUS					UNPAID LOAN INTEREST																			
6	DATE OF ENTRY INC BEN		REASS		UNMATCHED INSTALMENT PREMIUM																				
7	AGE AT ENTRY BINARY		PREM PAYING TERM		ARREARS SPREAD PREMIUM EARLIER CASH OPTION - TABLE VIII																				
8	INC BEN PREM PAYING TERM		AGE AT ENTRY		SUM ASSURED																				
9	INC BEN TERM		TERM		CASH OPTION																				
10	AGE AT ENTRY FOR INC BEN		DATE OF ENTRY		OFFICE PREMIUM																				
11	NET PREMIUM					ADDITIONAL TABLE XIII PREMIUM PAID UP TABLE II OPTION																			
12	ASSUMED INCOME P.A.					EXTRA PREMIUM																			
13	SECY NET PREMIUM					SECY PREMIUM																			
14	PREMIUM FUNCTION					DISABILITY BEN PREMIUM																			
15	CASH AT DEATH	COUNT	DATE OF ADDN OF PROF.	TERRI- TORY	J-RATE	PROPOSER'S RISK PREMIUM																			
16	MEMBER'S TITLE, INITIALS, SURNAME																								
17	MEMBER'S SURNAME CONTINUED																								
18	MEMBER'S SURNAME CONTINUED					ADDRESS FOR PREMIUM NOTICES																			
19	ADDRESS FOR PREMIUM NOTICES CONTINUED																								
20	DO																								
21	DO																								
22	DO																								
23	DO																								
24	LIFE ASSURED'S SURNAME																								
25	LIFE ASSURED'S SURNAME CONTINUED						LIFE ASSURED'S INITIALS																		
26	A.C. CODE AND DATE			A.C. CODE AND DATE			A.C. CODE AND DATE																		
27	LIFE ASSURED'S DATE OF BIRTH, SEX		RACE	EARLIER C.O.CODE	MED	NFA	POLICY ADDRESS	A.C. CODE AND DATE																	
28	2ND LIFE ASSURED'S DATE OF BIRTH, SEX		RACE	ORIGINAL AMOUNT OF LOAN			ACC BEN CODE	EXTRA PR.CODE	D.B. CODE																
29	STOP ORDER CODE					TRAILER	PAID UP PORTION WAS VIII																		
30																									
31																									

FIG. 3.—An insurance policy described in a Perseus block.

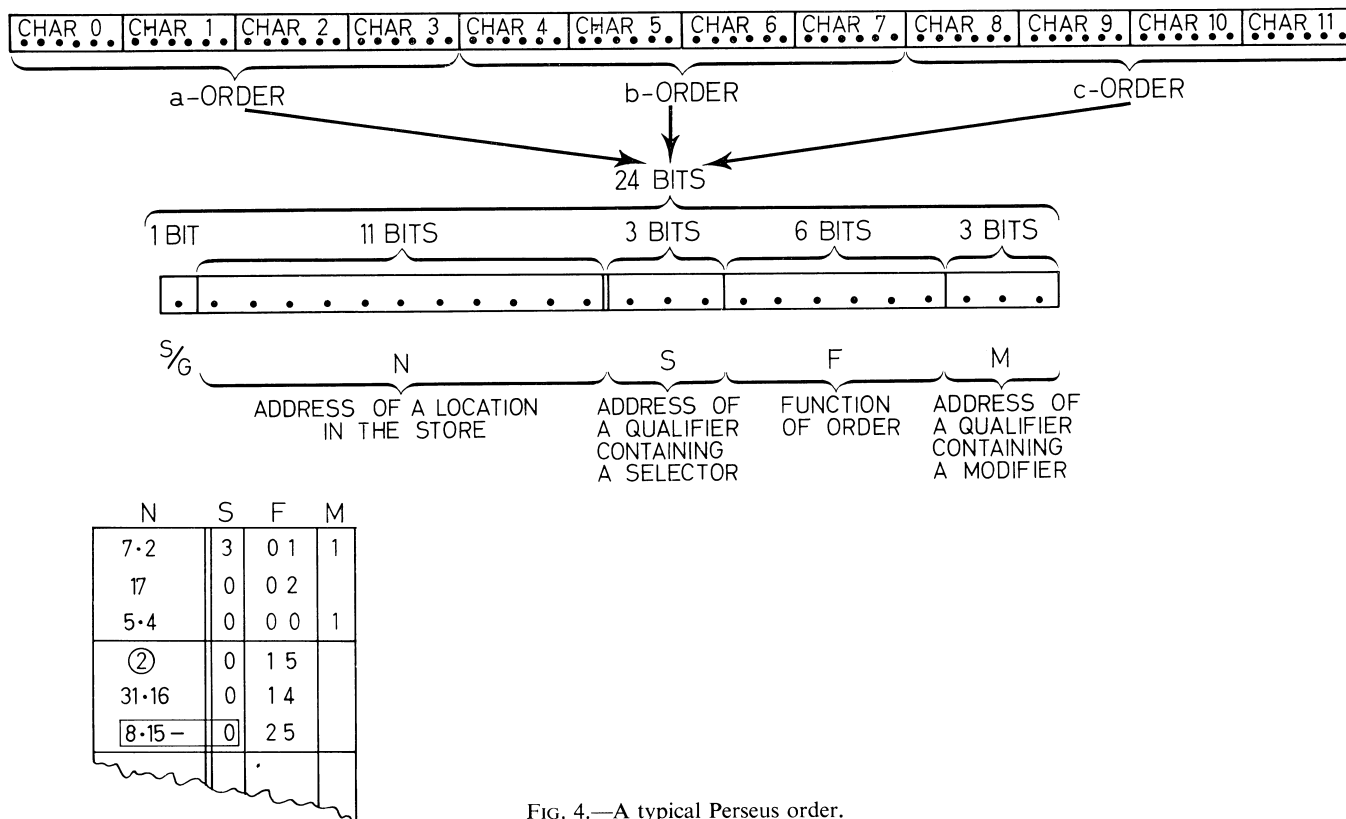


FIG. 4.—A typical Perseus order.

necessary to change the content of the radix register; this can be done under program control in a matter of microseconds. For decimal working, the content of the radix register (which is an ordinary location in the store of the computer) must be

10.10.10.10.10.10.10.10.10.10.

and for sterling working it must be

10.10.10.10.10.10.10.10. 2.10.12.

In the case of multiplication it is possible to multiply together two quantities of different units and to obtain a double-length (i.e. 24-character) product. For example, it is possible to multiply together a sterling quantity and a decimal quantity, or to multiply a quantity in hours, minutes, seconds by a decimal quantity, etc. Two radix registers are provided for holding the radix numbers of the multiplier and the multiplicand. The operation of multiplication can be defined as forming $n \times a$ where n is anywhere in the store and a is in a particular storage location. If $n = £5/3/4$ and $a = 3$ years 4 months, then $n \times a$ is £206/13/4 provided the radix registers are set appropriately, and if $n = £101/12/10$ and $a = 419$, then $n \times a = £42,587/17/2$. An accumulative multiplication order is also provided.

The operation of division forms the quotients (rounded and unrounded) and remainder obtained by dividing a double-length (24-character) dividend by a single-length (12-character) divisor. This again is a mixed-radix operation so that, for example, £4,591/19/10 can be divided by 104 to produce

an unrounded quotient of £44/3/0,

a rounded quotient of £44/3/1,

and a remainder of 7/10.

It is also possible to shift quantities contained in the mixed-radix accumulator. Shifts down are rounded, and an overflow indicator is set if overflow occurs when shifting up.

BINARY ARITHMETIC AND LOGICAL OPERATIONS

As well as the mixed-radix accumulator, another accumulator is provided, in which it is possible to perform binary additions and subtractions and various logical operations (e.g. collation, logical shifts, etc.).

PARTIAL-FIELD WORKING

Perseus has facilities for carrying out arithmetic on units of information other than whole words. It is possible to operate arithmetically on information contained in any field (i.e. portion of a word) formed by consecutive character positions of the word. For example, a word can contain several items (see Fig. 5), and it is possible in one order to select just a part (or field) of a word to be operated upon instead of the whole word. For example, the premium alone can be routed to the mixed-radix accumulator for further (sterling) work, or the policy number alone can be routed to the

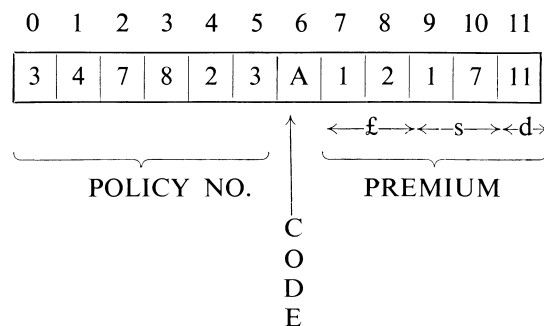


FIG. 5.—Three items stored in one Perseus word.

binary accumulator for matching purposes, etc. When the information is packed as shown in Fig. 3, this facility is very useful for obtaining just one piece of information from the item, or for changing some odd isolated pieces of information in the item.

NEGATIVE NUMBERS

Negative numbers can be represented in two ways in Perseus: either in sign and modulus form (e.g. -127) or in complement form (e.g. 999999999873). In everyday life (e.g. when punched on cards or printed on paper—negative numbers are usually represented in the first form, and it is for this reason that this form of representation is available in Perseus. Negative numbers are also held in complement form since it takes less electronic equipment to perform arithmetic on numbers represented in this way.

Facilities are available for converting from sign and modulus form to complement form and vice versa. All numbers must be in complement form if any arithmetic is to be carried out on them, and they are automatically complemented if they are in sign and modulus form when they are read into the mixed-radix accumulator. For printing purposes they are always held in sign and modulus form.

THE PERSEUS ORDER CODE

The order code is characterized by the function digits, which are written on the program sheet as two octal digits (see Table 2). Orders of group 0 (i.e. orders in which the first function digit is 0) are concerned with mixed-radix arithmetic, those of group 1 with binary arithmetic and logical operations. Groups 2 and 3 are concerned with conditional jumps, group 5 with shifts, group 6 with buffer transfers, and group 7 with magnetic-tape orders.

ZERO SUPPRESSION

In a Perseus system, printing is an off-line operation; that is, it can be carried out independently of the computer. Quantities to be printed are recorded first on a reel of magnetic tape, which is subsequently mounted on the Samastronic printer.

It is usual, when quantities are printed, to omit non-significant zeros and, in the case of sterling units, to omit any zero tens of shillings. This zero suppression

TABLE 2

THE PERSEUS ORDER CODE

00 $n' = x$	40 $m' = B \cdot P - R$	PERSEUS ORDER CODE NOTATION N First address in order X Mixed-radix accumulator Y Binary accumulator B Block in the computing store P Position of word in block R Position of order in word L Link digit H Shunting quarter block V Block of buffer store $B \cdot T$ Quarter block of main store x Word in X y Word in Y n Word in location of address N m Content of modifier c Content of counter s Content of selector d Content of punched card h Content of shunting quarter block v Content of buffer store $(b \cdot t)$ Content of quarter block of computing store (pq) Content of registers 16, 17 a Content of register 18 (multiplier) OVR Overflow indicator S/G Stop/Go bit Primes indicate values after obeying order
01 $x' = n$ (complementing)	41 $m' = m + B \cdot P - R$	
02 $x' = x + n$	42 $c' = N$	
03 $x' = x - n$	43 $c' = c + N$	
04 $x' = x$ (sign and modulus)	44 $s' = B \cdot P$	
05 $x' = B \cdot P$	45 $(pq)' = n \times a$	
06 $x' = x + B \cdot P$	46 $(pq)' = (pq) + n \times a$	
07 $n' = \text{spaces}$	47 Form $(pq)/n$	
10 $n' = y$	50 Arithmetical shift of x , N chars. up	
11 $y' = n$	51 Arithmetical shift of x , N chars. down	
12 $y' = y + n$	52 Logical shift of y , N chars. up	If N is negative the shift will be in the reverse direction
13 $y' = y - n$	53 Logical shift of y , N chars. down	
14 $y' = y$ and n	54 Logical shift of y , N bits up	
15 $y' = N$	55 Logical shift of y , N bits down	
16 $y' = y + N$	56 Double-length arith. shift, N chars. up	
17 $n' = 0$	57 Double-length arith. shift, N chars. down	
20	60 $b' = v$	PERSEUS ORDER CODE NOTATION N First address in order X Mixed-radix accumulator Y Binary accumulator B Block in the computing store P Position of word in block R Position of order in word L Link digit H Shunting quarter block V Block of buffer store $B \cdot T$ Quarter block of main store x Word in X y Word in Y n Word in location of address N m Content of modifier c Content of counter s Content of selector d Content of punched card h Content of shunting quarter block v Content of buffer store $(b \cdot t)$ Content of quarter block of computing store (pq) Content of registers 16, 17 a Content of register 18 (multiplier) OVR Overflow indicator S/G Stop/Go bit Primes indicate values after obeying order
21 Jump to	61 $v' = b$	
22 $B \cdot P - R$	62 $b' = v$; $v' = b$	
23 storing	63 $(b \cdot t)' = d$	
24 the link	64 $(b \cdot t)' = h$	
25 if	65 $h' = (b \cdot t)$	
26 $L = 1$	66 $(b \cdot t)' = h$; $h' = (b \cdot t)$	
27	67 Jump to $B \cdot P - R$, return immediately	
30	70 Read next block of tape	
31 Jump to	71 Write on next block of tape	
32 $B \cdot P - R$	72 Step back one block and read	On tape mechanism N
33 storing	73 Step back one block and write	
34 the link	74 Search for block c	
35 if	75 Rewind	
36 $L = 1$	76 Step back one block	
37	77 Stop; jump to $B \cdot P - R$	

is not done by the printer but is carried out automatically whenever the number is sent from the mixed-radix accumulator into the computing store. For example, if the decimal number

0 0 0 0 0 2 9 7 0 8 5

is sent out of the accumulator it appears in the store as

Sp Sp Sp Sp Sp Sp 2 9 7 0 8 5

and is written on the magnetic tape in this way. When this tape is eventually printed on the off-line printer the number appears on the printed page as,

2 9 7 0 8 5

If the number should be read back into the accumulator for further arithmetic to be carried out upon it, the spaces are replaced by zeros, so that an arithmetically correct answer will result if the number is used as an operand in an arithmetic operation.

Similarly, the sterling quantity

0 0 0 0 0 1 0 4 2 0 2 9

when routed out of the accumulator appears in the store, and hence on magnetic tape, as

Sp Sp Sp Sp Sp 1 0 4 2 Sp 2 9

and, when printed, appears on the page as

£1,042/ 2/9

PUNCHED CARDS ON PERSEUS

When a data-processing system is introduced into an organization which already has a large punched-card installation, it is usually necessary to transfer the information contained on the cards on to magnetic tape. With the Perseus system the computer is used to perform this operation. This is convenient since usually the information on the card has to be rearranged before placing it on magnetic tape, and the rearrangement may be different according to the type of card being read.

Card files which are already in existence may contain cards punched in various codes, and the cards may be either of the Powers-Samas, Hollerith or I.B.M. type. In order to be able to convert the information contained on existing punched cards (in various codes) into information on magnetic tape, facilities are provided with Perseus for reading any of these types of card punched in any code. Since the code in which the cards are punched varies from installation to installation, the conversion from card code to computer code is not done by built-in hardware but by program.

In any installation, when the conversion has been completed (i.e. when all file information has been put on to magnetic tape), it is possible to punch all new cards in a standard code. Since this code is fixed for all installations, it is possible to provide automatic conversion from card code to computer code on input of a card to the computer, and this conversion need no longer be done by program.

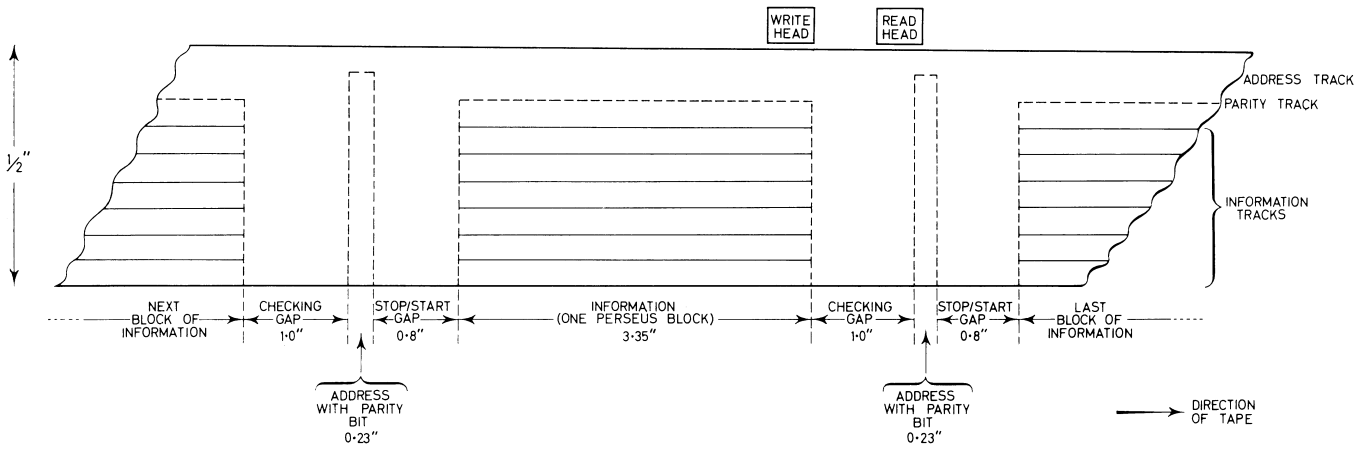


FIG. 6.—The layout of the information on Perseus magnetic tape.

The overpunching technique can be used with the Powers-Samas cards, and it is also possible to use interstage working if required. IBM and Hollerith cards are read at 200 per minute and Powers-Samas cards at 240 per minute. Plugboards are available for distributing the card columns among the characters in a quarter block when the card is read into the computer.

MAGNETIC TAPE

Each reel of magnetic tape is pre-addressed before being used with the computer. Those regions of the tape where the tape-addressing equipment has found “drop outs” to be present are not addressed, and the computer does not therefore attempt to record information on these parts of the tape. The maximum size of a reel of tape that can be used with Perseus is 3,000 ft, and such reels contain 6,050 blocks of information; each block consists of 32 words and is the same size as a block in the computing store.

In order to write information on a tape, or read information from it, the tape is mounted on a tape mechanism, and it is possible to have up to 16 tape mechanisms attached to one Perseus computer. The reading and writing of information on the tape is controlled by a tape control unit. Each of these units contains a buffer store and controls up to a maximum of four tape mechanisms.

All information is written on to or read from the tape *one block* at a time. The reading and writing is carried out via a buffer store, and all operations between the buffer store and the magnetic tape are autonomous. That is, the computer can go on and obey other orders while the information is being read from or written on the tape. The time taken to read a block from the magnetic tape into the buffer store and vice versa is approximately 75 milliseconds. The transfers from buffer store to computing store and vice versa take 8 milliseconds.

The blocks on a reel of magnetic tape have addresses

0, 1, 2, 3, . . .

and a search order is provided which causes the tape to move and position a specified block so that it will be read by the next read-tape order.

When the whole of a reel of magnetic tape has been used it has to be rewound before it can be used again. This can be done under program control and takes place at a faster speed than normal magnetic-tape operations; a 3,000-ft reel is rewound in 4 minutes. Information is recorded on the magnetic tape at a density of 123 bits per inch and the layout on the tape is shown in Fig. 6. The operations of reading from and writing on magnetic tape are checked, as described later.

PAPER TAPE INPUT

Two paper tape readers are provided. One is used to read in 5-hole paper tape containing program; the other is capable of reading in 7-hole paper tape containing data.

CONVENTIONAL USE OF PERSEUS STORE FOR DATA-PROCESSING OPERATIONS

The basic operations involved in updating a main file can be described by Fig. 7.

As stated earlier, the Perseus store is of two types. Blocks 0, 1, 2, 3, 4, consist of single-word lines to which there is immediate access. Blocks 5, 6, . . . , 31

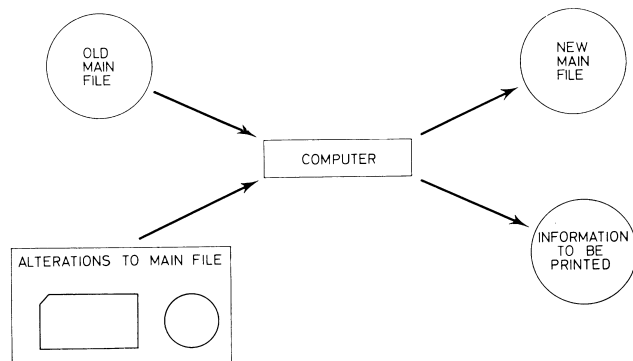


FIG. 7.—A typical data-processing operation.

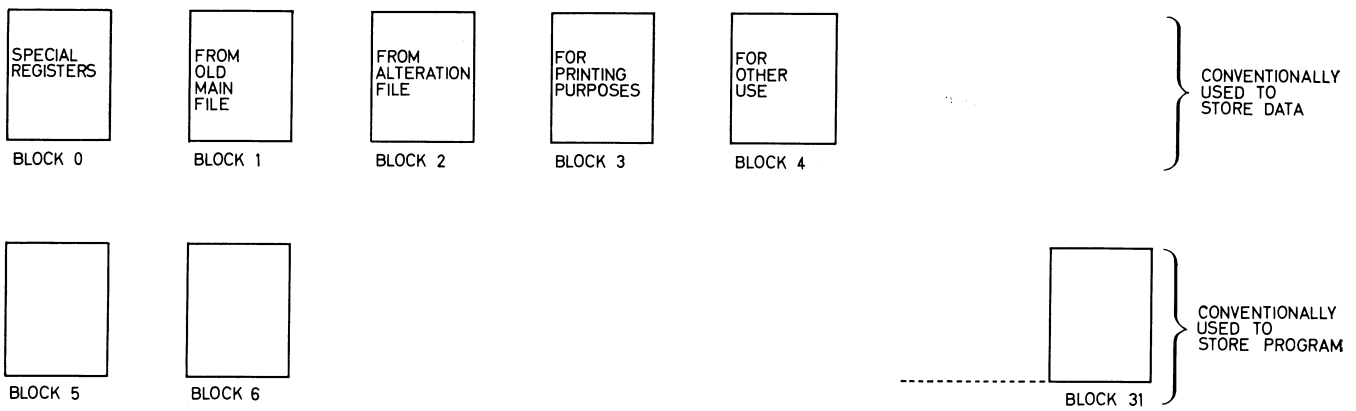


FIG. 8.—Conventional use of Perseus stores.

each consist of two 16-word lines with a variable access time of between 0 and 4 milliseconds. Conventionally the store is used so that data are stored in blocks 1, . . . , 4 and the program in the remainder of the store, as shown in Fig. 8.

Although there is an access time of between 0 and 4 milliseconds to obtain a word from blocks 5, 6, . . . , 31, the addresses in these blocks are “scrambled” as shown in Fig. 9. In this way three normal orders are obeyed in five word times. In the first word time the order triplet is routed to the order register, in the next three word times the three orders are obeyed, and there is then one word-time wait before the next order triplet is routed to the order register. As stated earlier, the word time is 234 microseconds.

THE OFF-LINE PRINTER

The printer has been designed to print the contents of a reel of Perseus magnetic tape on either preprinted stationery or plain paper. Blocks are read from the magnetic tape, and 93 characters of each quarter-block can be printed on one, two, three or four lines of print. The printer has a repertoire of 50 characters (including space) and there are 140 possible printing positions on any line, with a horizontal pitch of 10 positions to the inch. The vertical pitch of print is either 6 lines per inch or 8 lines per inch; the normal printing speed is 300 lines (printing cycles) per minute.

It is possible to have two webs of paper in the print head at any time and printing can occur on either web, but no overlapping is allowed. It is possible to throw the paper on either web up to a distance of 1 in without losing a printing cycle. On one web it is possible to throw it longer distances, up to a maximum of 16 in (taking 1, 2 or 3 printing cycles).

The equipment is also capable of selecting quarter blocks from the reel of magnetic tape for printing, so that, although information to be printed on many types of form is contained on the reel, only that information to be printed on one type of form may be selected on any passage of the tape.

The information to be printed on one, two, three or four lines of print is distributed by plugboards, of which four are available, and it is possible to print on a line, together with the 93 characters of the quarter block, a phrase previously wired on a plugboard attached to the printer. Up to seven different phrases are available.

MISCELLANEOUS FACILITIES

Various other facilities are available for performing a variety of functions. Direct teleprinter output is provided, for use when developing programs and for issuing instructions to the operator during productive work. Such instructions as the reel of magnetic tape that requires changing, the label of the new tape to be mounted and the label to be stuck on the old tape, etc., are included in this category. It is an easy matter to transfer quarter blocks around the computer so that information from punched cards can be sorted quickly.

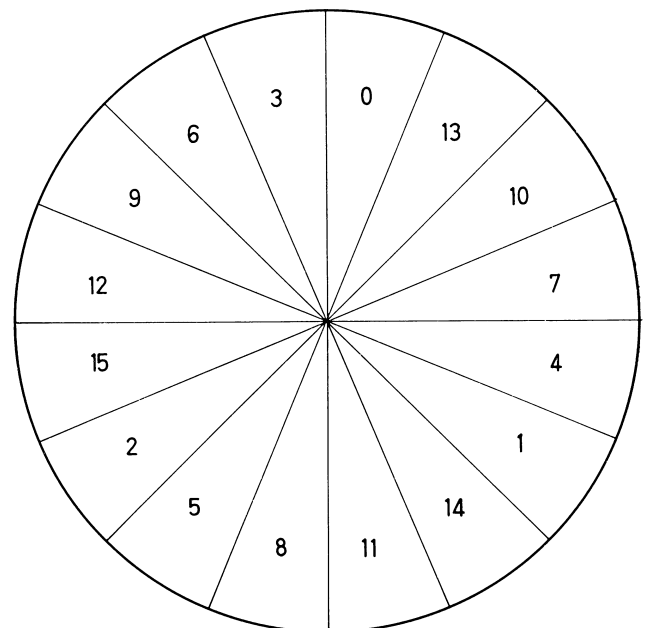


FIG. 9.—The staggering of the addresses in the sixteen-word lines.

A paper tape reader reading 7-hole paper tape is also available for inserting data into the computer. The 7-hole code consists of a parity bit for checking purposes, and 6 bits of information; these form the computer code for the character, so that no conversion is required. It is also possible to fill a block of the computer store with zeros, or space characters, by just one order.

CHECKING EQUIPMENT

It was felt that for commercial work it was absolutely essential that the results obtained were checked to ensure accuracy. Although this may be done by program it was felt desirable to relieve the programmer of this burden and to do as much checking as possible automatically.

The nickel delay-line store of Perseus is checked by holding a parity bit with each word of information. When information is recorded in the line the parity bit is inserted to make the total number of binary 1's in the word odd. Whenever the word is used (i.e. read out of the store) parity is checked. All arithmetic operations are checked by using a system of checked adders. Cards are read at two reading stations and the readings are compared to see if they are the same. Paper tape input is checked by the parity bit on the 7-hole tape and by the choice of the code on the 5-hole tape.

When information is recorded on the magnetic tape a parity bit is recorded with each 6-bit character, and the information in the block is followed by a *check sum* (Fig. 6). This check sum is the quantity obtained by adding together all the characters in the block.

When a block of information is read from the magnetic tape the parity of each 6-bit character is checked before the parity bit is discarded, and the content of the block is summed as the reading operation is carried out. When the block has been read the sum of its content is therefore known. This sum is also on the magnetic tape and, if the operation has been carried out correctly, the two quantities should agree. If the parity of any of the characters read from the tape is incorrect, or if the two check sums do not agree, the computer automatically

reads again all the information contained in the block. If the operation still fails the computer continues to re-read the block automatically, up to a maximum of eight times. If on any of these occasions the computer reads the block of information correctly, it will then continue and obey further orders in the usual way. If after re-reading the block eight times it still has not read the information correctly, the computer will stop and light a tape-failure indicator light on the control console.

The converse procedure happens when writing on magnetic tape. Characters of a block are written on the tape together with their parity bits, while the check sum of the block is formed, written on the tape, and held in the computer. A second head, situated behind the writing head, reads back the information in the block, checks the parity of each character, and forms the sum of the information read. The second head also reads back the check sum recorded on the tape. If all these three sums agree, the computer accepts that the block has been written correctly on the tape. If any of them disagree, or if the parity of any character is wrong, the computer will automatically write the information on the tape again. Unless on any occasion the operation is carried out correctly, the computer will continue to re-write the information for a maximum of eight times. If on all occasions the operation is carried out incorrectly the computer will stop.

At the beginning of their life the magnetic tapes are addressed before being used on the computer. During this process an extra bit is recorded after the address, so as to make the total number of 1's present in the address even. Whenever an address is read (which occurs in all magnetic-tape orders except the rewind order) the number of 1's in it is counted to make sure it is even. As before, if the address is continually misread, eight repeats will occur before the computer stops.

ACKNOWLEDGEMENT

The author is indebted to Ferranti Ltd. for permission to publish this paper.

REFERENCES

- The Programming Manual for the Ferranti Perseus Computer.* Ferranti publication List CS 181.
Programming Examples and Solutions for the Ferranti Perseus Computer. Ferranti publication List CS 77.