ABC*/JAA

A T L A S

The machine described in this document is the result of a collaboration between Ferranti Ltd. and Manchester University, whose installation will be known as MUSE. This document is confidential and may not be given or lent to any person other than the addressee without the authority of Ferranti Ltd. It constitutes a description of the system from the viewpoint of a knowledgeable programmer, and is intended for the use of persons concerned with the design of the system.

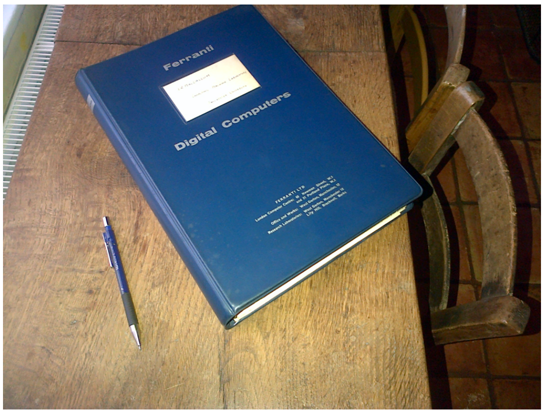
This description is initially incomplete and will be augmented from time to time. It is therefore not arranged primarily for easy reading but rather to simplify the issuing of amendments. Nor is it intended for lay reader, whose interests will be catered for when the design is nearer completion, when another description will be prepared for general circulation presenting the system in an appropriate manner for those who will be concerned solely with using it.

19/6/59

* ‘ABC’ is Ben Cooper, a Ferranti employee who was in charge of documentation for the joint University/Ferranti Atlas design team from 1959 onwards. This document was sometimes known by the engineers as *Ben’s Bible.*

Transcribers' Notes

This is a transcription of Iain MacCallum's July 1963 paper copy of the document colloquially known as the "Atlas Bible". The original artefact is a stiff blue foolscap ring-binder with *Ferranti Digital Computers* printed in white on the cover:



The original document contains a total of 335 foolscap pages, inserted into the ring binder. Each page has a page/section number at the top of the page and a date-of-issue at the bottom. Apart from the Preface page (which is dated 19/6/1959) the issue-dates for the various sections in this particular copy span the period 2/1/61 to 1/7/63. Indeed, 78% of the pages date from before the official inauguration of the Manchester Atlas on 7th December 1962.

The original idea was that replacement pages would be issued from time to time, as errors were found or equipment was added or modified. The document as a whole can thus be said to represent the evolving MUSE/Atlas project during the development stages of the hardware and systems software. As such, the information precedes the later issue of most of Ferranti's published Atlas user-manuals.

This transcription was made by Dik Leatherdale in 2013. Detailed proofreading was carried out in January 2015 by Simon Lavington. Comments are welcome and should be sent to dik@leatherdale.net or to lavis@essex.ac.uk .We have not tried too hard to create an exact replica of the foolscap original although we have respected the original layout as far as practical. The issue-dates at the bottom of each original page have been omitted from the transcription but dates can be deduced from the Contents page. An idea of the appearance of each of the original foolscap pages may be had from: http://bitsavers.informatik.uni-stuttgart.de/pdf/ict_icl/atlas/Atlas_Machine_Description_Sections_8-13_Nov64.pdf

To assist the modern reader, the transcribers have added a few annotations in red. These deal with such things as:

- minor obvious typo errors (7 instances);
- other suspected errors, deduced by reference to the January 1965 edition of the Programming Manual for Atlas Basic Language (ICT list CS 348A);
- poorly-printed original pages where certain characters are not clear (2 instances).

Contemporary minor hand-corrections to the original text (6 instances) remain in black but are indicated by strike-through characters in the transcription.

21st century readers may find some of the terminology and nomenclature unfamiliar - in particular the term “store lines” (especially “B-lines”). This should be taken to refer to a word or halfword of storage or a register, the origin of the term referring to the way in which “lines” of bits were stored on the face of a cathode ray tube in earlier computers build around CRT storage. Occasionally the term “register” is used to refer to a word of storage.

In similar vein the reader will notice that bits within a word or halfword are referred to with bit 0 being the <u>least</u> significant bit (“little-endian”) contrary to the usual mainframe parlance. Again this is believed to derive from practice with earlier systems with serial arithmetic units where the low order bits were the first to be passed to the arithmetic unit. In later Atlas documentation the opposite (“big-endian”) practice was adopted and, in particular, the manual for Atlas Basic Language (ABL) explains that programmers number bits one way and engineers the other. Still more modern usage on Minicomputers and PCs tends to revert to the earlier practice.

Finally, the original text uses the abbreviation mμsec for ‘millimicroseconds’, now more usually known as nanoseconds, nsec.

Contents (as at 1st July 1963)

[1] The section numbering in the body of the document as scanned differs slightly from that given here.

[2] Section 8.11 is not present in the 1st July 1963 contents list, but exists in the body of the document as scanned (as three foolscap pages dated 1/7/63).

[3] Sections 9 and 9.1 are missing from the body of the document as scanned.

ATLAS NOTATION

Where the contents of a location are specified, lower case letters are used. Affixes are written as lower case letters. The various possible combinations involving affixes are not included. Some letters have alternative meanings according to context. The result of an operation is denoted by a prime.

A	The full double length floating point accumulator, comprising 79-bit fractional part Ax and 8-bit exponent Ay
Al	the 48-bit floating point number comprising L, Ls and Ay
Am	the 48-bit floating point number comprising M and Ay
AO	accumulator overflow register
B	a B-register
Ba	the B-register specified by Ba digits of an instruction
Bc	the B-carry
Bd	the pair of B-registers, B98,99, used in 48-bit logic Extracodes
Bm	the B-register specified by Bm digits of an instruction
Bt	the B-test register
C	the main control register (B127)
C( )	the contents of the location specified within the brackets
DO	division overflow register: set if divisor non-standard or zero in 374,376, 377 orders
E	the Extracode control register (B126)
EO	(1) the exponent overflow register (2) the instruction can set EO and so cause an interrupt
F	the function digits
H	a half-word 24-bit register
I	the interrupt control register (B126)
II	the inhibit interrupts switch
I/ME	a control flip-flop which is set if interrupt control is being used
K	a character address within a word, i.e. the least significant 3 bits of the address
k	(affix) last octal digit of affixed quantity
L	the less significant half of Ax, 39 bits with no sign
LO	page or block lock-out
Ls	the special sign bit associated with L
M	(1) the more significant half of Ax, 40 bits, the most significant bit being the sign bit (2) the main control register
M/E	A control flip/flop which is set if main control is being used, or else if an interrupt occurred while in main control
N	(upper case only) the address part of an instruction regarded as a 24-bit signed integer
n	(lower case only) N after modification according to the type of instruction
P	(1) page in main store (2) engineer’s punch (3) (if followed by a digit) peripheral “look-at-me” in central computer V store
PAR	a page address register
PRA	a present block address (magnetic tapes)
Q	(1) word within a page

	(2) the accumulator is left standardised as the result of the instruction
Qs	sign of the quotient
R	the instruction rounds A by forcing 1 in the least significant bit of M unless the contents of L are zero (in division unless remainder is zero)
R+	the instruction rounds A by forcing 1 in the least significant bit of M if the most significant bit of L is a one
S	(not affix) the address of a store location, usually a full word
s	(affix) the sign bit
SVI	sacred violation by an instruction
SVO	sacred violation by an operand
T	the magnetic tape block marker interrupt "look-at-me" in central computer V-store
TAC	a tape address command register
TC	the tape co-ordinator
TCR	a tape command register
V	V-store (
VC	V-store (core store)
VD	V-store (drums)
VP	V-store (peripherals)
VT	V-store (magnetic tape)
Vα	(α integer) lineα of V-store (similarly for sections of V)
Vα/β	(αβ integers) bit of line of V- store (similarly for sections of V)
W	a full word address (i.e. K ignored)
w	(affix) the most significant 21 bits of the affixed quantity regarded as an integer
x	(affix) the fractional part of the affixed location
y	(affix) the exponent of the affixed location
:	(affix) the consecutive pair of locations beginning with that affixed
*	(affix) the register immediately following that affixed

Section 1 - General Description

1.1 Structure of a Word

Each word consists of 48 binary digits. An instruction and a floating point number each occupy one 48 bit word. A word may be sub-divided into two 24 bit half-words or eight 6 bit characters. Normally the three least significant bits of the modified address are ignored when referring to a store word (the two least significant for a half-word), but for certain extracode routines the individual characters in a word or half-word may be referred to. A register may thus be considered to consist of either:

(i) a 48 bit word with address r (or r.0)
or (ii) two 24 bit words with addresses r (or r.0) and r.4
or (iii) eight 6 bit characters with addresses r (or r.0) r.1; r.2; ...r.7

(When an address is specified it will be obvious from the context whether a word, a half-word or a character is referred to).
An instruction is divided as follows:

F Ba Bm S

where

1. the twenty-four least significant bits i.e. characters 4, 5, 6, 7 (S) normally refer to an address in the central store.
2. the next seven bits (Bm) refer to one of 128 twenty-four bit B registers (some of these registers which are used for special purposes do not contain all twenty-four bits. See section 2.2.2). Except when Bm = 0, S is modified by the contents of the specified B register before the instruction is obeyed. It is not however, possible to B-modify all the B-test codes as in some of then the Bm digits are used as an operand.
3. the next seven bits (Ba) refer to one of these B-registers. With some functions (A codes) a second B modification takes place; with others (B codes) Ba specifies a B register which is to be operated on.

The case Ba = 122 is an exception. The above operation is then performed, not with B122 but with the B-register whose address is stored in B121. Thus for an A code, S is modified by bm and also by the contents of the B register specified by the contents of B121. For a B code the address of the Ba used is given by the contents of B121.

4. the ten most significant bits (F) are the function bits, and the functions may be either basic functions or extracode routines. For a basic function these digits are decoded directly, but for an extracode routine control is transferred via a jump table to the address of the first instruction in the fixed store for the extracode routine.

1.2 The Central Store

The word structure allows for 2^{21} forty-eight bit word addresses but it is not necessary for so many registers to exist. This store is sub-divided as follows:

(1) Fixed Store

This is constructed from ferrite rods in a woven wire mesh. The access time for reading is 0.2μsecs and it is not possible to write into this store by programme. The three most significant digits of the addresses in this store are 1 0 0 thus allowing for a theoretical maximum of 2^{18} forty-eight bit registers.

This store contains various fixed routines (the extracode routines, the programmes for the peripheral equipments etc), the 512 jump instructions for the extracode entry jumps and some useful constants. This store is in multiples of 4,096 forty-eight bit word registers and it is anticipated that a store of 8,192 words will be sufficient for most installations. It will, however, be possible to decrease (losing some of the programmes) or increase this 8,192 word store if required.

(2) Subsidiary Store

The minimum size of this store is 1024 forty-eight bit words and it may be increased in multiples of this amount. The three most significant digits of an address in the subsidiary store are 1 1 1. This store is used for the directories and working space required by the fixed store programmes. Its size can therefore be considered to be dependent upon the routines in the fixed store.

It is not possible for either the subsidiary store or the V-store to be used by a programmer accidentally. A lockout is provided to ensure that if these stores are referred to by main control (i.e. when a programme in the main store is being obeyed) an automatic interrupt occurs. For this reason these stores are referred to as the “private store”. The use of these stores is only permitted when a programme in the fixed store is being obeyed (more precisely using either the extracode or interrupt controls).

(3) The V-store

This is the name given to a number of registers specified by the address digits but which are in practice distributed in various parts of the machine e.g. in the peripheral equipments. Access to and from these registers is by the normal Accumulator and B register type orders. The three most significant digits in the address of any word in this store are 1 1 0 and this store forms part of the private store.

This store contains:

(a) the Page Address Registers.
(b) the flip-flops associated with each magnetic tape channel.
(c) the registers noting the angular position of each drum i.e. which sector is at present under the reading head.
(d) registers holding the information to be transferred to or from the appropriate peripheral equipments e.g. paper tape reader and card reader. This information is

transferred between the main store and these special registers by normal instructions. These words may be taken from or put into a core store page which is time shared between the peripheral equipments. (For the magnetic tape and the drums, the transfer of information to and from the main store is automatic).

(e) information referring to the peripheral equipments e.g. the state of the Engage/Disengage flip-flop on each magnetic tape mechanism. A further example is the 5/7 channel tape reader which has a digit indicating to which mode the reader is switched and hence by testing this the programmer knows whether the reader is reading 5 or 7 channel paper tape.

(4) <u>Main Store</u>

The addressing system allows for a maximum of 2^{20} forty-eight bit words in this store and the most significant digit of the address of a word in this store is always zero. A "block" consists of 512 words and the main store address digits are as shown in the following diagram.

```
23   22 21             13 12  11 10             4 3  2    1 0
     <---Block address--->    <---address of 48--->       One of
                                 bit word in block        four
                                                          characters
                                                          in a half
                                                          word
                                                 24 bit
                                               half word
```

The main store may consist of core store only or core store supplemented by drums. However, due to built-in equipment and a fixed store programme the programmer will use this store as if it were a one-level store.

(a) <u>Core Store</u>

This is constructed of ferrite cores with a cycle time of not more than 2µsecs. These cores are in pairs of "stacks", the size of a stack being 4,096 words. (It is possible for an installation to have a core store consisting of stacks of 1024 words but in this description it is assumed the normal size will be 4,096 words). The minimum size of the core store is one pair of stacks, i.e. 8,192 words, and it may be increased in multiples of this amount. The size of this store is likely to be restricted by practical considerations although theoretically the limit is 2^{20} words.

There will be a selection mechanism for each stack and each pair of stacks will be considered to consist of an "even" and an "odd" stack. An even stack will contain all the registers with even addresses for that pair of stacks and the odd stack will contain the registers with odd addresses. The arrangement for a machine with two pairs of stacks can be depicted diagrammatically as:

First pair of stacks

Even Stack	Odd Stack
Page 0 Even Addresses	Page 0 Odd Addresses
Page 1 Even Addresses	Page 1 Odd Addresses
Page 15 Even Addresses	Page 15 Odd Addresses
4096 words per stack	4096 words

512 words per page

Second pair of stacks

Even Stack	Odd Stack
Page 16 Even Addresses	Page 16 Odd Addresses
Page 17 Even Addresses	Page 17 Odd Addresses
Page 31 Even Addresses	Page 31 Odd Addresses
4096 words	4096 words

512 words per page

When obeying a programme instructions are taken from the core store in pairs. A pair of instructions consists of an instruction from an even stack plus the next instruction from the odd stack in the same pair. Provision is made in the machine for the cases

(i) when the first instruction in a pair involves a control transfer.

(ii) when a control transfer to the second instruction in a pair is involved. No facility is built into the machine for detecting when the second instruction in a pair has been altered by the first instruction.

(b) <u>Magnetic Drum Store</u>

Each drum has 192 <u>information</u> tracks each of 6,144 bits, i.e. a total of 24,576 words. These tracks are divided into 8 bands of 24 and the 24 bits of each half-word are kept in the corresponding position on 24 different tracks. A transfer involves a block of 512 words and there is room for 6 blocks or sectors on each band of tracks. A drum revolves at about 5,000 r.p.m. giving a revolution time of 12 milliseconds and a time to transfer one block of 2 m.s. plus any waiting time. Transfers to and from the drum hold up the computer if reference is made to that part of the store being used in the transfer.

(c) <u>File Drum Store</u>

The use of high capacity drums of the type recently introduced by I.C.T. is being considered. Each file drum will have a capacity of about 3 x 10^5 words. Transfers between the core store and the file drums may be arranged in a similar manner to those between the core store and the magnetic drums. A block will probably consist of 4,096 words.

1.3 <u>Identification of Store Type</u>

Address digit position

23	22	21	20		
0	δ	δ		Main store (core store and drums)	
1	0	0		Fixed Store	
1	0	1			
1	1	0		V store)	
1	1	1		Subsidiary store)	Private Store

An Automatic Interrupt (see Section 14) occurs if

(a) reference is made to the Private Store when on Main Control

(b) the modified address is in the Private Store but the unmodified address is not when on Extracode or Interrupt Controls.

The information in locations in the Fixed Store may be read but cannot be overwritten by programme. If an attempt is made to write to the Fixed Store the instruction is "completed" in the normal manner but the Fixed Store information is unaltered.

Section 2 The Function Code

2.1.1. The Function Digits

The ten function digits are allocated as follows :-

<------------------------------------>

47	46	45	44	43	42	41	40	39	38	
0	0	0	0	δ	δ	δ	δ	δ	δ	Not allocated
0	0	0	1							Basic B codes
0	0	1	0							Basic test codes
0	0	1	1							Basic A codes
0	1	0	0							Not allocated
0	1	0	1							Basic B codes + extracode return
0	1	1	0							Basic test codes
0	1	1	1							Basic A codes + extracode return
1	0	δ	δ							Extracode B entry
1	1	0	0							Extracode B entry
1	1	0	1							Extracode A entry
1	1	1	δ							Extracode A entry

Function codes with either 0000 or 0100 in digits 47-44 are not allocated. Digit 24. of line 1 of the Central Computer V-store (see Section 13.2) is set and an automatic interrupt occurs if an attempt is made to obey an instruction with one of these function codes.

If a code with either 0101 or 0111 in function digits 47-44 is decoded whilst on main control the corresponding basic function with digit 46 = 0 is performed.

Codes with 0110 in function digits 47-44 are duplicates of the Basic test codes i.e. 0010 in digits 47-44.

In instructions with either 0010 or 0110 in function digits 47-44 digit 43 is not decoded and hence two codes for each of these functions are available, one with digit 43 as a 0 and the other with digit 43 as a 1.

The above allocation of the Extracode entry codes allows for 320 B-type and 192 A-type extracodes.

2.1.2. Extracode

The instructions provided are divided into two categories:

(1) basic
(2) extracode

The extracode instructions are performed by routines in the fixed store which are entered automatically when the most significant function digit in an instruction is a one. They are sub-divided into two types, A or B, according as to whether double or single modification of the address occurs. A total of 512 codes is available for extracodes of which 192 are A type and 320 are B type.

At present the first 2048 words of the fixed store are reserved for the extracode routines. The extracodes are divided into groups of 64 and each group is allocated a "column" of 256 words in the fixed store. As it takes about 1μ.s. to switch between different columns of the fixed store it is desirable, particularly for short extracodes, that all the instructions for a given extracode should be in the same column. Normal control transfers to another column may however be used when this is not possible.

It is possible for one extracode to use another as a sub-routine. Two main cases arise:

(a) normal sub-routine entry and exit by setting links. If only two exits are made, markers (e.g. a digit in a B-register) may be used in place of links. A special case is when the final setting is common, there is no need to test for the different masters and normal extracode exit can be used.

(b) it is necessary for the sub-routine, which is a complete extracode, to be as fast as possible. Three methods are used, each of which makes the “master" extracode slightly longer, but does not affect the sub-routine extracode:
   (i) if the master has complex conditions to set up initially, but has the same ending as the sub-routine, entry may be made by an extracode order merely to save space. Exit is by return to main programme.
   (ii) if the end conditions are not the same, it is necessary to first store the contents of B127 and then reset B127 to a fixed store address before entering the extracode sub-routine. The latter exits into main control and results in the remainder of the master extracode being obeyed in main control. The final exit to the main program is by restoring B127.
   (iii) similar to (ii), but the final setting up requires reference to the private store, and cannot therefore be achieved on main control. Private extracode instructions are provided which, in effect, change to extracode control, subject to some condition being satisfied. Two conditions being used are that main control contains an address with non-zero sign bit, or a subsidiary store switch is set (the switch would have been set before entering the sub-routine).

Entry to A Type Extracodes

Nearly all entries to the extracode routines are encountered when the machine is on main control. Occasional entries may be either;

(a) when on extracode control and a second extracode is used as a sub-routine or

(b) when on either extracode or interrupt control and a machine (or fixed store programme) fault specifies extracode entry.

(a) Entry when on Main Control.

The conditions for entry are $f_9 = 1$, $f_8 = 1$ and f_7 f_6 are either 01, 10 or 11 where f_9 f_8 f_7 f_6 f_5 f_4 f_3 f_2 f_1 f_0 are the function bits, I/ME = ME and M/E = M.

1) (S + bm + ba) is stored in B119. Thus, by using B119 as a modifier in the extracode routine the required address may be specified.

2) the control number in main control (B127) is advanced by 1 i.e. to the address of the next instruction.
3) Ba, i.e. the number specified by the seven Be digits, is stored in digits 8-2 of B121.
4) the function digits are placed in extracode control (B126) as follows

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	0	0	0	0	0	0	0	F_8	F_7	F_6	0	0	F_5	F_4	F_3	F_2	F_1	F_0	0	0	0

5) the M/E digit is reset to E to specify extracode control.

The effect of this is to transfer control to the appropriate line (given by F) in the table of control transfer instructions in the Fixed Store.

b) Entry when on Extracode Control

The conditions for entry are as above except that M/E = E. The effect is:

1) (S + bm + ba) is stored in B119
2) b126' = b126 + 1 (subsequently being overwritten)
3) b121' = Ba
4) b126' = F and this causes a control transfer to the extracode routine.

c) Entry when on Interrupt Control If I/ME = I then

1) (S + bm + ba) is stored in B119
2) b125' = b125 + 1
3) b121' = Ba
4) b126' = F
5) M/E = E

However, as the Interrupt control always has priority over Main and Extracode controls no immediate entry to the Extracode routine takes place. When the interrupt routine is completed and I/ME reset to ME, this switches to Extracode control and the specified routine is entered.

Entry to B Type extracodes

The conditions for entry are f_9 = 1, and either f_8 = 0 or f_8 = 1 and f_7 f_6 = 00

1) (S + bm) is stored in B119
2) the number in the current control register is increased by 1.
3) Ba, i.e. the number specified by the seven Ba digits, is stored in B121. By using B122 (i.e. the B substitution register) in the extracode routine the B register specified by the Ba digits may be used as a second operand, or by specifying B121, the seven Ba digits themselves can be used as an operand.
4) 126' = F
5) the M/E digit is reset to E

Exit from Extracodes

If a basic instruction (f_9 = 0) is decoded with f_8 = 1 whilst on extracode control the corresponding basic function is obeyed and also the [sic] if it is either a B or an A type function, (f_6=1). The instruction immediately following the one specifying entry to the Extracode routine is then obeyed.

If the corresponding basic function is a test code i.e. digits 45 and 44 are 10 then the basic test code is obeyed but the M/E digit remains set at E. An "Unassigned Function" interrupt occurs if an attempt is made to obey a code where digits 45 and 44 are 00, the M/E digit remaining set at E.

Owing to the possibility of a non-equivalence interrupt occurring the last instruction in an extracode routine must never be an order referring to an address in the main store.

Sacred Violation and Unassigned Extracodes

When on Main control any reference to the "private store", i.e. the V-store and the Subsidiary store, causes an automatic interrupt (Sacred Violation). On Extracode (and also Interrupt) control an automatic interrupt occurs if the modified address is in the private store but the unmodified address is not. This prevents the programmer referring directly to locations in the private store.

Not all the 64 available codes in each group are allocated and consequently the corresponding control transfers are not needed, making additional space for useful instructions available at the beginning of the column. Where this happens care is taken that if an unassigned extracode is specified, and an entry is made part way through an assigned extracode, the instructions obeyed do not overwrite part of the private store, (except possibly the common working space available to the extracodes). Thus obeying an unassigned extracode will not interfere with any peripheral equipment or any other programme.

2.1.3. The Control Registers and the "Pair" Flip-Flop

The Control Registers

The control registers are fast flip-flop registers used to store the address of the instruction to be obeyed. When the instruction is obtained from the address specified, the contents of the control register in use are increased by 1 to give the address of the next instruction in the programme. If the instruction is for a control transfer, the address specified in the order is placed in the control register and the next instruction taken from this address.

The organisational routines and extracode routines for Atlas are contained in the Fixed Store. In order to save having to store, and later restore, the contents of the Main programme control register (B127) when a Fixed Store routine is to be obeyed two further control registers are provided. They are Extracode (B126) and Interrupt (B125) controls.

Two "control flip-flops" (line 3, digit 25 and line 4, digit 24 of the Central Computer V-store - see section 13.2) are used to specify the control register being used. They are:

"Main/Extracode" flip-flop M/E
"Interrupt/Main or Extracode" flip-flop I/ME

The outputs of these two flip-flops specify which control register is being used, and in the case of an Interrupt which control was being used when the Interrupt occurred; the four possibilities being:

M/E	I/ME	
0	0	Extracode control
0	1	Interrupt control (interrupt occurred whilst on Extracode control)
1	0	Main control
1	1	Interrupt control (interrupt occurred whilst on main control)

Both these flip-flops can be set and re-set by fixed store programme by writing 1 or 0 to the appropriate digit of the Central Computer V-store. Control is changed automatically

(a) to Extracode on encountering an Extracode instruction when on Main Control
(b) to Main on encountering an "Extracode return" instruction (Section 2.1.2)
(c) to Interrupt if an "automatic interrupt" occurs (see Section 14)

The exit from Interrupt control must be initiated by the fixed store programme by resetting the I/ME digit in the V-store to ME.

The Pair Flip-Flop

Instructions are always taken from the main core store in pairs, the pair comprising an "even" instruction and the consecutive higher addressed "odd" one. These are placed in two flip-flop registers, the Present Instruction Even (PIE) and the Present Instruction Odd (PIO) registers.

Associated with these registers is a "Pair" flip-flop. This is set to "Pair" when main control is being used and the next instruction required from the main core store has an even address.

It is reset to "Not Pair" when:

(a) on either Extracode or Interrupt control and the next Instruction required from the main core store has an even address.

(b) a write to main control is specified.

When an instruction is required from an odd address in the main core store it is taken direct from PIO to PIE and obeyed if the Pair flip-flop is set; if the pair flip-flop is not set two instructions are first read from the main core store into PIE and PIO and then the required instruction is taken from PIO to PIE and obeyed.

In the case of a main control transfer to an even address the flip-flop is reset to "Not Pair" (as M has been written to) and is then set to "Pair" after the address has been partially decoded. For a control transfer to an odd address the flip-flop remains reset to "Not Pair" until the next "even" instruction is about to be obeyed. If this "odd" instruction causes a change of control, the register is still reset to Not Pair and reference to the core store is thus necessary to get the next instruction.

Instructions are read one at a time from the Fixed and Subsidiary stores to the PIE register; the "pair" flip-flop and the PIO register remaining unchanged unless B127 is written to.

On Extracode and Interrupt controls the pair flip-flop remains reset at Not Pair once an even instruction has been read from the main core store. Instructions are subsequently taken in pairs from the core store but in all cases the required order is transferred to PIE (via PIO in the case of the odd instruction) and obeyed. Thus if it is desired to obey instructions in the main core store on Extracode (or Interrupt) control the entry should be to an even addressed instruction so that the Pair flip-flop is reset to Not pair, otherwise the first instruction obeyed will be taken from the PIO register. (Alternatively the flip-flop could be reset to Not pair by writing to B127).

Programmers should not use an even instruction to alter the odd instruction [of] the same pair otherwise inconsistent results may be obtained on different runs of the programme. Normally the odd instruction obeyed will be the one already waiting in the PIO register but if an interrupt involving a change of programme occurs immediately after the even instruction is obeyed, on returning to the first programme the (altered) odd instruction will be taken from the core store.

2.2.1. The B-registers

B-Modification

There are 128 B-registers and these are separate from the central store. Most of the B-registers consist of twenty-four bit core store registers with a cycle time of about 0.65μsecs and the remainder consist of trigger circuits.

In all instructions, apart from the B-code instructions, the contents of Ba and Bm (i.e. ba and bm) are first added to the address to give a modified address or operand. The operand can either be

the contents of (S + ba + bm) or
the integer (S + ba + bm).

In most B-code instructions, ba is used as a second operand, the modification of S being by bm only.

In two of the B-code instructions, i.e. 164 and 165, bm is used as an operand and hence no B-modification takes place for these two codes.

For most of the B-test codes bm is used as a further operand, but in those codes where this does not happen (e.g. basic codes 0224-0227) B-modification takes place as usual.

When obeying an instruction a test is made to see if the Bm and/or the Ba digits are zero. If so then the appropriate modification of the address is omitted.

B-Carry Digit

The "B-carry" digit is a digit stored in position 24 of line 6 of the Central Computer V-store. It records the "carry or borrow" from the last stage of the B-adders during the B-add part of certain B- instructions. The rules for setting it are:-

(1) on main or extracode control, it is set to 1 or 0 on the following B-codes.
100, 110, 120, 130, 140, 150, 160, 170
102, 112, 122, 132, 142, 152, 162, 172
104, 114, 124, 134, 144, 154, 164, 174.

(2) on interrupt control it is not set for any codes but is left as it was set previously.

(3) on all other basic codes it is not set but is left as it was set previously.

2.2.2. Special Purpose B-registers

The following is a list of the B-registers which are used for special purposes.

B-register	Allocated for use by
0	Always contains zero
81-89	Library sub-routines
90	Sub-routine level
91-99	Extracode routines
100-110	Supervisor
111-118	Interrupt routines
119	Extracode address operand
120	Display
121	Substitution Register (see Section 1.1)/Ba operand (see Section 2.1.2.)
122	Does not exist. When specified by the Ba digits the contents of B121 are called up
123	Special register used when dealing with the peripheral equipments (see Section 12.2.)
124	Exponent of Accumulator
125	Interrupt control
126	Extracode control
127	Main control

B-registers 121-127 inclusive are flip-flop registers and not core store registers.

B120 is the address of 24 neon lights on the Engineer's Console. The contents of any store register can thus be displayed by writing them to B120. Reading from B120 always gives zero.

B121 consists of the 7 bits 8 to 2 only; bits 23 to 9, 1 and 0 are always zero.

B124 consists of 9 bits only, the 8 bit exponent is stored in digits 22-15 and digit 23 is used as a guard digit. Digits 14-0 are always zero.

2.3.1 Summary of Basic B Functions

	0*	1	2*	3	4*	5	6*	7
010	ba'=s-ba	ba'=s	ba'=ba-s	ba'=-s	ba'=ba+s	ba'=2^6ba+s (circular shift)	ba'=ba≢ s	ba'=ba&s
011	s'=s-ba	s'=-ba	s'=ba-s	s'=ba	s'=ba+s	(s'=ba+s)	s'= ba≢ s	s'=ba&s
012	ba'=n-ba	ba'=n	ba'=ba-n	ba'=-n	ba'=ba+n	ba'=2^6ba+n (circular shift)	ba'= ba≢ n	ba=ba&n
013	n-type code	n-type code	n-type code	n-type code	n-type code	n-type code	n-type code	n-type code
014	(ba'=s-ba)	(ba'=s)	(ba'=ba-s)	ba'=ba/2-s (circular shift)	(ba'=ba+s)	(ba'=s)	(ba'=ba v s)	ba'=ba v s
015	bt'=s-ba	s-type code	bt'=ba-s	s-type code	s-type code	s-type code	s-type code	s-type code
016	(ba'=n-ba)	(ba'=n)	(ba'=ba-n)	ba'= ba/2-n (circular shift)	ba'= ba + (bm&n)	ba'=bm&n	(ba'=bavn)	ba'=bavn
017	bt'=n-ba	n-type code	bt'=ba-n	n-type code	n-type code	n-type code	n-type code	n-type code

* B-carry is set by functions in the columns marked with an asterisk.

Instructions enclosed in brackets are duplicates of instructions elsewhere in the table

The Codes marked "s-type code" may cause either a "Non-equivalence" or a "Sacred Violation Interrupt".

2.3.2 Summary of Basic Testing Functions

	0	1	2	3	4	5	6	7
020 (024)	ba'=n bm'=bm+0.4 if bm ≠ 0	ba'=n bm'=bm+1.0 if bm ≠ 0	ba'=n bm'=bm-0.4 if bm ≠ 0	ba'=n bm'=bm-1.0 if bm ≠ 0	(ba'=n bm'=bm+0.4 if bm ≠ 0)	(ba'=n bm'=bm+1.0 if bm ≠ 0)	(ba'=n bm'=bm-0.4 if bm ≠ 0)	(ba'=n bm'=bm-1.0 if bm ≠ 0)
021 (025)	ba'=n if bm is odd	ba'=n if bm is even	(ba'=n if bm is odd)	(ba'=n if bm is even)	ba'=n if bm = 0	ba'=n if bm ≠ 0	ba'=n if bm ≥ 0	ba'=n if bm < 0
022 (026)	ba'=n bm'=bm+0.4 if bt ≠ 0	ba'=n bm'=bm+1.0 if bt ≠ 0	ba'=n bm'=bm-0.4 if bt ≠ 0	ba'=n bm'=bm-1.0 if bt ≠ 0	ba'=n if bt = 0	ba'=n if bt ≠ 0	ba'=n if bt ≥0	ba'=n if bt < 0
023 (027)	(ba'=n if xa = 0)	(ba'=n if xa ≠ 0)	(ba'=n if xa ≥ 0)	(ba'=n if xa < 0)	ba'=n if xa = 0	ba'=n if xa ≠ 0	ba'=n if xa ≥ 0	<ba'=n if xa < 0

As digit 19 is not decoded in the test codes, duplicate codes are available for each function (e.g. 0200 and 0240). In functions 200-203 and 220-223 the increment is added to, or subtracted from, bm only if the test succeeds. Functions 224-227 and 230-237 are the only ones in which normal B-modification of n takes place.

2.3.3 Summary of Basic Accumulator Functions

	0	1	2	3	4	5	6	7
030	a'=am+s Q EO	a'=am-s Q EO	a'=-am+s Q EO	(a'=-am+s) Q EO	am'=s,l'=0 Q EO	am'=-s,l'=0 Q EO	am'=s,l'=0 Q EO	(am=s,l'=0) Q EO
031	a'=am+s N Q EO	a'=am-s N Q EO	(a'=-am+s) Q EO	(a'=-am+s) Q EO	am'=s N AO	am'=-s N AO	(am'=s,l'=0) Q EO	(am'=s,l'=0) Q EO
032	am'=am+s QR EO	am'=am-s QR EO	am'=-am+s QR EO	(am'=-am+s) QR EO	(am'=s.l'=0) Q EO	(am'=-s,l'=0) Q EO	(am'=s,l'=0) Q EO	(am'=s,l'=0) Q EO
033	a'=am+s AO	a'=am-s AO	a'=-am+s AO	(a'=-am+s) AO	am'=s,l'=0 AO	am=-s,l'=0 AO	(am'=s,l'=0) Q EO	(am'=s,l'=0) Q EO
034	a'=a Q EO	a'=a EO	a'=am.s Q EO	a'=-am.s Q EO	l'=xs, ya'=ya	l'=xs,m'=sign xs ya'=ya	s'=am,a'=0	s'=al,l'=0
035	(a'=a) Q EO	(a'=a) EO	a'=am.s sign l = sign m' AO EO	a'=-am.s sign l' = sign m' AO EO	am'=a R AO	s'=am,am=l Q EO	s'=am	s'=al
036	am'=a QR EO	am'=a R EO	am'=am.s QR EO	am'=-am.s QR EO	xa'=xa.8 ya'=ya AO	xa'=xa8^{-1} ya'=ya AO	am'=|am|,l'=0 Q EO	am'=|s|,l=0 Q EO
037	(a'=a) Q EO	(a'=a) EO	a'=am.s AO EO	a'=-am.s AO EO	am'=am÷s DO QR EO	al'=a÷|s|, m'=rem a÷|s| DO not set EO	al'=am÷|s|, m'=rem a÷|s| DO EO	al'=am÷|s|, m'=rem am÷|s| DO EO

N in instructions 310,311,314 and 315 means that L is not cleared before the addition takes place.

The codes where S is not specifically mentioned may cause either a “Non-equivalence” or a “Sacred Violation Interrupt”.

2.4 Extracode Routines

The following two sub-sections give brief specifications of the extracodes. For detailed descriptions reference should be made to the Atlas Fixed Store Routines (Vol. 1), section 9 to 11.

2.4.1 Allocation of Function Numbers

There are 512 function numbers available for extracodes, 1000-1777. Of these, 1000 - 1477 are singly modified instructions i.e. B-type, and 1500 - 1777 are doubly modified i.e. A-type.

The extracodes are divided into sections as shown below:

1000 - 1077	Peripheral routines.
1100 - 1177	Organisational routines.
1200 - 1277	Test instructions and character data processing.
1300 - 1377	B-register operations.
1400 - 1477	Complex arithmetic, Vector arithmetic, and other B-type accumulator functions.
1500 - 1577	Double length arithmetic and accumulator operations using the address as an operand.
1600 - 1677	Logical accumulator operations, trigonometric routines and half word packing.
1700 - 1777	Logarithm, exponential, square root etc. And miscellaneous arithmetic operations.

Where possible the last two octal function digits correspond to those of similar basic operations.

2.4.2 Arithmetic Extracodes

The extracode function is listed at the left of the page and followed by a description. The number of basic orders obeyed is given at the right of the page. This number includes the extracode order and its entry in the jump table; where necessary a range or formula is given.

Accumulator operations are rounded floating point unless marked X when they are suitable for fixed point working.

Test Instructions

1200	ba' = n if AO set; clear AO	9
1201	ba’ = n if AO clear; clear AO	7
1206	ba' = n if most significant character in g = 0	4
1216	ba’ = n if bm>0	4-6
1217	Ba’ = n if bm≤0	4-5
1223	ba’ = n if Bc=1	4
1226	ba’ = n if bt>0	4-6

1227 ba’ = n if bt≤0 3-5

1234 c’ = c+2 if am approx. = s 11-12

1235 c' = c+2 if am not approx. = s 11-12
Approximate equality is defined by
$\frac{|am-s|}{|am|}$ < C(ba), with am standardised
If am = 0, am is not approx. = s

1236 ba' = n if ax>0 4-6

1237 ba’ = n if ax ≤0 3-5

1255 ba’ = n if m is neither zero nor all ones 9

1727 = c+1, 2 or 3 as am >, = or < s 7

1736 c' = c+2 if |am| > s 9

1737 c' = c+2 if |am|<s 7

<u>Character data processing</u>

In 1250 and 1251, S is taken as a character address

1250 ba' (digits 0-5) = s, ba’ (digits 6-23)= 0 7-10

1251 s' = ba (digits 0-5) 11-18

1252 Unpack n characters starting from character address C(ba), to half words from C(ba*) 16+int.pt.(6¾n)

1253 Pack n characters starting from halfword address C(ba*), to character address C(ba) 18+5n

<u>B-register operations</u>

1300 ba' = integral part of s, am' = fractional part of s 10

1301 ba' = integral part am, am' = fractional part am 9

1302 ba’ = ba.n 23-24

1303 ba' = -ba.n 22-23

1304 Ba’ = int. pt. (ba÷n), b97' = remainder 25-28
In 1302-1304, ba and n are 21 bit integers in digits 23-3.
Octal fractions are rounded towards zero

1312 ba' = ba.n 23-24

1313 ba' = -ba.n 22-23

1314 ba' = int. pt. (ba÷n) b97' = remainder 25-28
In 1312-1314, ba and n are 24 bit integers

1340 ba' ba.2^{-n}; unrounded arithmetic shift right 10-22

1341 ba' ba. 2^{n}; unrounded arithmetic shift left 9-21

1342 ba' = ba. circularly shifted right n places 10-19

1343 ba’ = ba. circularly shifted left n places 9-18

1344 ba' = ba. logically shifted right n places 10-21

1345	ba' = ba. logically shifted left n places	9-20
1347	h' = h v ba[4]	5
1356	bt' = ba ≢ h	7
1357	bt’ = ba≢n	5
1364	ba’ = (ba & ň) v (bm & n); b119'= (bm[5] ≢ bm) & n	4
1371	b121' = Ba, b119' = S+bm	2
1376	bt' =ba & h	5
1377	bt' = ba & n	4

Complex arithmetic

The complex accumulator, Ca, is a pair of consecutive registers, the first register having address ba. If Ba = 0, Ca is locations 0,1. s: is a number pair. Ca may coincide with S: but not overlap with it. A is spoiled.

1400	ca' = log s:	
1402	ca' = exp.s:	140
1403	ca' = conj.s:	8
1410	Ca’ = √s:	Max. 117
1411	am' = arg. s: radians	
1412	ca' = mod. s:	Max. 53
1413	ca’ = s cos s*, s sin s*	95
1414	ca' = recip. s:	15
1420	ca' = ca+s:	8
1421	ca' = ca - s:	8
1424	ca’ = s:	6
1425	ca’ = -s:	6
1456	s:' = ca	5
1462	ca' = ca. S:	18

Vector Operations

The vectors are of order n. $\underline{S}_1$ is stored in consecutive locations from ba, and $\underline{s}_2$ from ba*. A is spoiled.

1430	$\underline{s}_1' = \underline{s}_1 + \underline{s}_2$	9+4n
1431	$\underline{s}_1 = s_1 - \underline{s}_2$	9+4n
1432	$\underline{s}_1' = am \,.\, \underline{s}_2$	10+4n

[4] The ABL manual specifies as “s” the quantity written here as “h”.

[5] The bm is in the original. This is an obvious error. The ABL manual corrects this to ba.

1433 $\underline{s}_1' = \underline{s}_1 + am.\underline{s}_2$ 10+5n

1434 $\underline{s}_1' = \underline{s}_2$ (forwards or backwards) 13+3n

1436 $am' = \Sigma^{(0\text{->}n-1)}\ \underline{s}_{1i}.\underline{s}_{2i}$ 10+5n

1437 $a' = \Sigma^{(0\text{->}n-1)}\ \underline{s}_{1i}.\underline{s}_{2i}$ 10+13n

Miscellaneous B-type accumulator operations

1452 $m' = m.xs.\ 8^{ya+ys-ba}$, $ya' = ba$. (X) 19-23
1461 generate pseudo random no.
1466 $a' = C(S + bm + ba)\ .\ C(S + bm) + s$ 18
1466[6] $am' = \Sigma^{(0\text{->}r)} s_i x^i$ where $x = am$, $S_i = S + i$, $r = ba$ 6+3n
1473 $m' = (xa \div xs).8^{ay-ys-ba}$; $ya' = ba$. (X) 24-28
1474 C(ba)' = quotient (am ÷ s), am' = remainder. (X) 20-29
1475 C(ba)' = quotient (a ÷ s), = remainder (X) 19-28
1476 C(ba) = quotient([int.pt.am] ÷ s), am' = remainder 28-37
1477 Remainder and adjusted integral quotient[7] 14-31

Double length arithmetic

The double length number is s: = s + s* where ys - 13 ≥ ys*. s* and al are assumed to be positive numbers.

1500 a' = a + s: 10

1501 a' = a - s: 10

1502 a' = -a+s: 14

1504 a' = s: 4

1505 a' = -s: 3

1542 a' = a.s: 15

1543 a' -a.s: 19

1556 s:' = a 5

1565 a' = -a 5

1566 a' = |a| 4-6

1567 a' = |s:| 5

1576 a' = a÷s: 19

Arithmetic using address as Operand

The address is taken as a 21 bit integer with one octal fractional place. Fixed point operations imply an exponent of 12.

1520 am' = am + n 10

1521 am' = am - n 9

---

[6] Original shows this function code as 1466, an apparent typo. The ABL manual confirms the correct code to be 1467.

[7] This extracode is absent from the ABL manual and the fixed store listing indicates that it was disabled in later versions of the Supervisor

1524	am' = n, 1[8a] = 0	8
1525	am' -n, l' = 0	7
1534	am' = n, l' = 0 (X)	10
1535	am' = -n. l' = 0 (X)	9
1562	am' = am.n	8
1574	am' = am÷n	16
1575	am' = a÷n	15

Logical accumulator operations

The logical accumulator G is B98 and B99

1204	ba' = no. of 6 bit characters from most significant end identical in g and s	10-31
1265	g' = 2^6g + n	11
1601	g' = s	3
1604	g' = g + s	7
1605	g' = g + s with end around carry	12
1606	g' = g ≢ s	4
1607	g' = g & s	3
1611	g' = ¬g	3
1613	g' = g[8b]	3
1615	am'= g	4
1630	g' = g & ¬s	5
1635	g' = am	4
1646	g' = g v s	3
1652	bt' = g - s	7-9

Half word packing

h has an 8 bit exponent and a 16 bit argument.

1624	am' = h	6
1626	h' am, with h rounded	8

Functions and miscellaneous routines

1700	am' = log s	
1701	am' = log am	

[8a] The ‘1’ in the original is an obvious typo. The ABL Manual confirms that the correct symbol should be ‘l’.

[8b] The g’ = g in the original is an obvious typo. The ABL manual confirms that the correct version should be s’ = g

1702	am' exp		43
1703	am' = exp am		42
1704	am' = int. pt. s		5
1705	am' = int. pt. am		4
1706	am' = sign s		5-6
1707	am' = sign am		4-5
1710	am' = √s		Max 42
1711	am' = √am[9]		Max 41
1712	am' = √(am^2 + s$^{2)}$		Max 50
1713	am = am^s		
1714	am' = 1/s		4
1715	am' = 1/am		4
1720	am' = arc sin s (-π/2 ≤ s ≤ π/2)		
1721	am' = arc sin am		
1722	am' = arc cos s (0 ≤ s ≤ π)		
1723	am' = arc cos am		
1724	am' = arc tan s (-π/2 ≤ s ≤ π/2)		
1725	am' = arc tan am		
1726	am' arc tan (am/s) (-π ≤ am ≤ π)		
1730	am' = sin s		41
1731	am' = sin am		40
1732	am' = cos s		42
1733	am' = cos am		41
1734	am' = tan s		34
1735	am' = tan am		33
1752	m' = xa.8^{12} ; ya' = ya - 12	(X)	10
1753	xa' = m.8^{-12}; ya' = ya + 12	(X)	6
1754	round am by adding; standardise		6
1755	xa' = xa.8^{ya-n}; ya' = n	(X)	17
1756	s' = am, am' = s		8
1757	am' = s ÷ am		4
1760	am' = am^2		3
1762	m' = xa. 8^{12}	(X)	9
1763	xa' = m. 8^{12}	(X)	5
1764	xa' = xa.8^n	(X)	17
1765	xa' = xa.$^{8-n}$	(X)	12

[9] The ABL Manual specifies “aq” as the operand in functions 1711-3

1766	am' = |s|	(X)	4
1767	am' = |am|	(X)	3
1771	b121' = Ba, b119' = S + ba + bm		2
1772	m' = (m.xs) 8^{12}; ya' = ya + ys - 12	(X)	11
1773	m' = (xa ÷ xs) $8^{ya-ys-12}$; ya' = 12	(X)	27
1774	am' = am ÷s		10
1775	am' = s[10] ÷ s		9
1776	Remainder		13

10 The ABL Manual gives this quantity as “aq” (which makes more sense).

2.4.3 Organisational and Peripheral Extracodes

Magnetic tape

Block Transfer

1001 Search for section S on tape B

1002 Read next K sections from tape B to store blocks, P, P+1....

1003 Read previous K sections from B to P+K-1,...., P.

1004 Write P,P+1,..,P+K-1 to next K section on B

1005 Move tape B forwards K sections

1006 Move tape B backwards K sections

Organisational Instructions

1010 Mount

1011 Mount free

1012 Mount next reel of file Ba

1013 Receive tape (from another program)

1014 Write title

1015 Read title

1016 Unload

1017 Free tape

1020 Release tape (pass to another program)

1021 Release mechanisms

1022 Re-allocate

1023 How long?

1024 Where am I?

Variable Length Instructions

1030 Start reading forwards

1031 Start reading backwards

1032 Start writing forwards

1033 Select

1024[11] Start reading forwards from fixed blocks

1035 Start reading backwards from fixed blocks

1036 ba' = selected magnetic tape

1037 s' = mode of magnetic tape Ba

Transfer Instructions

1040 Transfer

---

[11] A typo. Should be 1034.

1041 Skip
1042 Mark
1043 Stop
1044 Word search
1046 Read next block on Orion tape
1047 Read previous block on Orion tape

Input

1050 Select input n
1051 Find selected input
1052 Find peripheral equipment number
1053 Test binary/internal code
1054 next character to Ba/jump to n at end of record
1055 ba' = no. of blocks read
1056 Read ba half words to S
1057 Read next record to S

Output

1060 Select ouput n
1061 Find selected output
1062 Find peripheral equipment type
1063 Delete output n
1064 Write character n
1065 End this record
1066 Write ba halfwords from S
1067 Write a record from S
1070 Rename output n as input ba
1071 Break output n
1072 Define output n

Subroutine Entry[12]

1100	Enter subroutine at s, ba' = c+1	6
1101	Enter subroutine at S, ba' = c+1	5
1102	Enter subroutine at bm, ba' = c+1	6

Branch Instructions

1103 Establish B branches

12 The ABL Manual (issue dated 1965) also specifies a further extracode in this group, namely 1362

1104 Start branch B at S
1105 Kill B. If B = 64 kill current branch
1106 Halt current branch if B is active
1107 Jump to n if B is active

Monitor

1112 Set Monitor jump to n
1113 Set Restart address to n
1114 Exit from trap
1115 Dump tape B section n if program monitored
1116 Do not dump if program monitored
1117 End program

Miscellaneous Transfers

1120 ba' = clock
1121 ba' = date
1122 ba' = local instruction counter
1123 set inst. counter = $n.2^{10}$
1124 v6' = n
1125 ba' =v6 & n
1126 v7' = n (hoot)
1127 ba' = v7 & n (read handswitches)

Searches, Traps, Miscellaneous

1131	Table search	8+6n
1132	Set trap/normal mode	
1133	ba' = trap address	
1134	Trap	
1135	Jump to n when block ≥ ba defined	
1136	am' = no. instructions obeyed	

Compiler and Supervisor

1140 Read parameter Ba to s
1142 End compiling
1143 Call System document s to be input stream ba
1144 Call System document s to store block ba onwards
1147 Call in compiler n
1150 Assign ba blocks, labels n to (n+ba-1) to overflow

1151 Set up blocks n onwards from overflow K.

1156 Enter extracode control at n if the "In Supervisor switch" is set

1157 Enter extracode control at n if the "Process switch" is set.

Store

1160 Read block P
1161 Duplicate read
1162 Read K+1 blocks
1163 Reserve band n
1164 Rename
1165 Store allocation = n blocks
1166 ba' = lowest block label $\geq$n
1167 Clear blocks
1170 Write block P
1171 Duplicate write
1172 Write K+1 blocks
1173 Release block P
1174 ba' = no. of pages available
1175 ba' = no. of blocks available
1176 Lose band n
1177 Lose block P

Section 3 Details of the Basic Function Codes

3.1 Floating Point Arithmetic

3.1.1 The Floating Point Accumulator

Representation of a Number

The Atlas word is 48 bits in length and when used to represent a number it is divided into eight bits for the exponent and 40 bits for the mantissa with, in each case the most significant digit being a sign bit. A binary point is assumed to lie between the sign bit and the next most significant digit of the mantissa. The exponent is an octal one and so the number represented is

$$x.8^y$$

Diagrammatically this may be shown

Sign + 7 bits	Sign + 39 bits
Y Exponent	X Mantissa

True complements are used for negative number so that when a non-zero number is in standardised form

$$1/8 \le x < 1 \quad \text{or} \quad -1 \le x < -1/8$$
$$\text{and} \quad -128 \le y < 128$$

A non–zero standardised number is one where the sign bit and the three next most significant bits are not all the same. Floating point zero is represented as 0×8^{-128} and this is taken to be a standardised number.

The Accumulator Mantissa

Floating point arithmetic makes use of a floating point accumulator in which the mantissa is double length (sign + 78 bits). To prevent loss of information due to “overflow” when an instruction is being obeyed two “guard” digits are provided at the more significant end of the accumulator. In the transfer of a number from a store location to the more significant half of the accumulator the sign digit of the number is also copied into the two guard digits. When a number is copied from the more significant half of the accumulator to a store register, it is the sign bit and the 39 most significant bits after the sign bit which are transferred.

In addition to the 79 bits in the mantissa of the accumulator a further digit exists to give the sign of the less significant half of the accumulator. This digit is set to zero by

a) most instructions involving addition or subtraction and where L is cleared initially (300–303, 312, 313, 320–323, 330-333)
b) instructions which copy information between a store location and either M or L (including am’ = - s) and where either L or A is specifically cleared (304–307, 316, 317, 324-327, 334-337, 346, 347)

c) six of the multiplication orders (342, 343, 362, 363, 372, 373).
d) the four division instructions (374-377).
e) the two modulus instructions (366, 367).

The digit is left unchanged by
a) the two addition and subtraction instructions where L is not cleared initially (310, 311).
b) instructions which copy a number from a store location to M without clearing L (314, 315).
c) instructions specifying a combination of rounding, standardisation and testing for exponent overflow but without other arithmetic operation having been performed (340, 341, 350, 351 354, 360, 361, 370, 371). This digit is however cleared on the 340, 350, 360 and 370 orders when A is set to floating point zero i.e. when the initial contents of M and L are zero but the sign of L is a one.
d) instructions which copy a number from either M or L to a store location without clearing A or L (355-357).
e) the two instructions specifying one octal shift of the contents of A (364, 365).

The digit is set to 0 or 1 as the sign of L by
a) instructions which copy the contents of a store location to L (344, 345).
b) two of the unstandardised and unrounded multiplication orders (352, 353).

The Accumulator Exponent

The exponent of the Accumulator is stored in digits 22–15 of B124, digit 23 being used as a guard digit to detect exponent overflow or underflow. The remaining digits of B124 are zero; digit 15 has the significance of 1 and digit 22 has the significance of “128” in the exponent. It is possible to set the guard digit of the exponent of the accumulator to an overflowed or underflowed condition by one of the B codes.

An octal exponent is used in Atlas and hence any shift of the mantissa must be by three places to change the exponent by a one. When a shift down takes place the more significant guard digit (and not the sign digit) of the mantissa is repeated in the new contents of the sign and guard digits.

Standard Numbers

In the list of basic codes many of the functions are specified as standardising the contents of the accumulator after the operation (e.g. 0300). In these cases the accumulator contents are shifted up or down (with suitable adjustment of the exponent) until a standardised number is obtained. A shift of six bits can be done in the same time as a three bit shift and if a shift of more than three bits is required, the contents of A are shifted six bits at a time until either the number is standardised or only a three bit shift is necessary and this three bit shift is then performed. For example, if the accumulator contents have to be shifted up 15 bits to be in standardised form this is done by two six bit shifts and one three bit shift. A similar breakdown into six bit and three bit shifts takes place when either the contents of the accumulator, or the number from a store location, have to be destandardised for a floating point addition or subtraction.

A special test is made to see if the accumulator is zero before standardization takes place and, if so, no shifting takes place and the exponent is changed to -128. The test as to whether A is zero is over the 42 bits of M, including the sign and guard digits, and the 39 digits of L but not the sign digit of L. The Ls digit is however cleared if A is set to floating point zero.

The accumulator is not set to floating point zero for the 340, 350, 360 and 370 orders. If the initial contents of Ya are less than -128 and the contents of M are in either standard or superstandard form. In the former case no change is made to the contents of A (apart possibly from round-off for the 360 order); in the latter case the contents of M and L are shifted down one octal digit and one is added to Ya.

The contents of the accumulator said to be in “superstandard” form if the sign digit and the two guard digits of the fractional part are not the same. For the unstandardised codes (e.g. 0330), the contents of the accumulator may be left in either superstandard or sub-standard form. In the former case the Accumulator Overflow digit in the V–store is set but if an attempt is made to copy the contents of the accumulator to the store the guard digits are not copied and hence an incorrect answer may be copied to the store. For example, if -1.0 is added to -1.0 using an unstandardised code (e.g. 330) the operation is, (assuming no exponent arithmetic is necessary)

	d_{-2}	d_{-1}	d_0	d_1	d_2	d_3 ...	d_{39}	d_{40}...	d_{78}
initial contents of A									
	1	1	1.	0	0	0	0	0	0
contents of S									
	1	1	1.	0	0	0	0	0	0
final contents of A									
	1	1	0.	0	0	0	0	0	0

Where d_{-2} and d_{-1} are the guard digits
d_0 is the sign digit
d_1, $d2_1$, d_{78}, are the accumulator digits.

On copying the more significant half of A to a store line, the exponent and digits d_0 to d_{39} are copied i.e. a number with mantissa identically zero.

Similarly, if floating point operations are carried out when the initial contents of A are superstandard the incorrect answer may be obtained.

Rounding

The method adopted is that for the floating point operations specifying rounding a one is forced i.e. “OR-ed” into the least significant digit of M if the contents of L are not zero. No forcing takes place If the contents of L are zero. For the codes specifying standardisation and rounding, the standardisation takes place first. The action of forcing a one into the least significant digit of M if the contents of L are non-zero cannot cause a standardised number to become non-standard and hence no further standardisation is necessary.

The error introduced by this method of rounding is

Last digit of M initially	Contents of L (39 digits)	Last digit of M after rounding	Error introduced
0	000........00	0	0
0	000........01	1	+111........11
0	000........10	1	+111........10
0	100........00	1	+100........00
0	111........10	1	+000........10
0	111........11	1	+000........01
1	000........00	1	0
1	000........01	1	-000........01
1	100........00	1	-100........00
1	111........10	1	-111........10
1	111........11	1	-111........11

From this table it can be seen that the average error (assuming all possible contents of L are equally likely) in rounding the final contents of the double length accumulator to a single length answer is zero and hence the result is unbiased.

A further advantage is that single–length integer arithmetic can be carried out exactly without introducing any rounding errors providing in every case the result of the operation is completely in M and does not extend into L.

An instruction (0354) is provided to give the “plus one” type of rounding when required i.e. to add one to the least significant digit of M when the most significant digit of L is a one. A table corresponding to the one above gives

Contents of L (39 digits)	Added to M	Error Introduced
000......00	0	0
000......01	0	-000......01
000......10	0	-000......10
011......11	0	-011......11
100......00	1	+100......00

100......01	1	+011......11
111......10	1	+000......10
111......11	1	+000......01

Again assuming all possible contents of L are equally likely a bias is introduced here by the case when L contains 100......00. (In decimal working the “plus one” type of arithmetic gives an unbiased error, if, in the case when ...500..0 is to be rounded, one to added to the next more significant digit to the five if that digit is odd; the five being discarded without a one added to the next digit if that digit is even).

3.1.2. Floating Point Operations

Addition and subtraction

Before two numbers which are represented in floating point form can be added/subtracted the numbers must be shifted relative to each other so that digits which have the same significance are added together. The amount of shift required is determined by the difference in the exponents of the two numbers.

In Atlas the number which has the smaller exponent is the one which is shifted and it is shifted down into Al. If an odd number of octal shifts is involved the first shift is a single one and after that the shifting is done two octal digits at a time. For rounding purposes and double length working it is required that the final contents of the accumulator (before rounding, if any) should always have their correct double length representation and it is sometimes necessary to take special action in the (am-s) and (-am+s) type codes before shifting. This special action is necessary because the accumulator adder is only- associated with Am and not with Al.

If the code is of the (am–s) type and the exponent of s is smaller than or equal to the exponent of a, xs is negated before any shifting takes place and the adder gates changed to do (am +s).

Similarly if the code is of the (-am + s) type and am has to be shifted (ys>ya) it is first negated and the adder gates changed to do (am + s).

At the same time as the shifting (if any) is being done the exponent of the accumulator is set to the larger of the two exponents.

The two numbers are then added or subtracted as required (this operation being over 42 bits) followed by standardisation and rounding as specified. In all codes the EO digit is set if appropriate just before the accumulator complete signal is given.

In the case of the standardised codes if the accumulator is zero on entering the standardisation loop the exponent is set to -128.

Numbers with exponents differing by 16 or more in the case of rounded codes and 32 or more in the case of unrounded codes are said to be out of range. Out of range numbers are treated in a special way as the accumulator contents are replaced by the number with the larger exponent, this number being negated if necessary, i.e. if the code is an (am–s) type and the numbers are out of range with the exponent of s the greater then the accumulator is set to –s. (The numbers 16 and 32 were chosen to determine out of range because they are the first binary powers greater than 13 and 26 respectively).

Multiplication

The standard process is:

1) a test is made to see if xa=-1.0. If so xa is shifted down one octal place but no alteration is made to the exponent

2) (ya + ys) is formed and tested for underflow. If underflow has occurred operation (6) below is left after the first octal digit has been dealt with and the accumulator is set to floating point zero.
3) The sign of L and the q digits are cleared. (The q digits are three digits off the bottom end of L).
4) The contents of the store line are copied into L and the sign is stored (not in the sign digit position of L).
5) am (including the two guard digits) is stored in the “Xs” register within the accumulator; 3am is also formed and stored within the accumulator and am is cleared.
6) The multiplication is done one octal digit at a time and the amount to be added to the partial products is determined from the bottom 3 bits of L and the top bit of q.
a) If the code is of the (am.s) type then the quantity which is added to the accumulator is given in the third column of the following table, (where am is the original contents of the accumulator). The corresponding quantity for the (–am.s) type code is given in the fourth column.

Bottom Octal Digit of L	Top q digit	am.s code	-am.s code
0	0	0	-0
1	0	am	-am
2	0	2am	-2am
3	0	3am	-3am
4	0	-4am	4am
5	0	-3am	3am
6	0	-2am	2am
7	0	-am	am
0	1	am	-am
1	1	2am	-2am
2	1	3am	-3am
3	1	4am	-4am
4	1	-3am	3am
5	1	-2am	2am
6	1	-am	am
7	1	0	0

b) the whole accumulator is shifted down one octal place and the addition for the next octal digit takes place
c) The accumulator registers are so arranged that the last octal digit taken (i.e. the fourteenth) is the sign bit of the store line i.e. it is 0 if xs is positive and 7 if it is negative.
7) If the initial contents of Xa were –1.0. (and hence were shifted down one place) the contents of Xa after the multiplication are shifted up one octal place without any alteration of the exponent.

The above process is the same for all multiplication codes. After this the action of the particular codes is as follows.

342-343 The standardisation loop is entered and after standardisation the exponent is tested for overflow

352–353 The sign of L is set equal to the final sign of Am, the accumulator overflow digit is set if necessary and the exponent tested for overflow.

362–363 The standardisation loop is entered and after standardisation am is rounded if necessary and the exponent tested for overflow

372-373 The accumulator overflow digit is set if necessary and the exponent tested for overflow.

Division

On all operations the sequence is as follows

1) (ya – ys) is formed [and sent to ya] and tested for overflow or underflow. If exponent overflow has occurred the EO digit in the Central Computer V–store is set and the division operation proceeds. If underflow has occurred the accumulator is cleared to floating point zero and the "accumulator complete" signal is given
2) the sign of L is cleared.
3) the signs of a and xs are compared and the sign of the quotient (0 or 1 according as to whether a and xs have the same or different signs) stored as the "Qs" digit (line 6, digit 28 of the Central Computer V–store).
4) the modulus of xs is taken and if the result is superstandard, xs is shifted down one octal place and one is subtracted from the exponent.
5) the modulus of am is formed (i.e. am' = |am|) and if the result is superstandard the complete accumulator is standardised.
6) (xa – xs) is formed in the accumulator buffer and if this is negative i.e. xs > xa, xa is shifted up one octal place, one is subtracted from the exponent and the "12/13 digit" (line 6, digit 29 of the Central Computer V–store) is set to zero (i.e.13) if it is positive the 12/13 digit is set ~~as a one~~. [The deleted words have been crossed out in ink in the original and replaced by: *to zero (ie 13), if it is positive the 12/13 digit is set to 1*].
7) The division loop is entered and cycled thirteen times. The answer is formed in the three "q" digits, which are off the less significant end of L and which are cleared initially. The division loop is as follows

a) a comparison is made between the contents of M, including the guard digits, (referred to as D) and the contents of xs (referred to as d). If (D-4d) would be positive i.e. the more significant guard digit would be zero, the subtraction (D–4d) is performed (giving the D for the next operation) and a one is put into the most significant q digit.
b) a comparison is made between D and 2d. If D is greater than or equal to 2d, the subtraction (D-2d) is performed and a one is put into the middle q digit.
c) a comparison is made between D and d. If D is greater than or equal to d the subtraction (D-d) is performed and a one is put into the least significant q digit.
d) the accumulator contents, including the q digits, are shifted up one octal place leaving the q digits clear. Finally the quotient is in L. and the "remainder" has been shifted up into M.

8) the final action on all codes before the accumulator complete signal is given is to test for exponent overflow. The additional action on the individual orders is:

the 374 order
(i) xs is tested for being sub-standard or zero, the D_0 digit being set and an interrupt occurring if it is. The instruction

is however completed normally except that steps (a), (b) and (c) of the division loop are omitted.[13]
(ii) L is cleared initially.
(iii) after am and xs have been made positive 14 is added to the exponent.
(iv) after the division loop the answer is rounded by forcing one into the least significant digit of L if the contents of M are not zero.

(v) M is cleared and the accumulator is allowed to standardise.
(vi) M is negated if the Qs digit is a one and the accumulator standardised again if necessary.

The 375 order
(i) after (5) above one is added to the contents of Ya.
(ii) after the division loop, the contents of M are shifted down one octal place, overwriting the initial contents of the least significant octal digit of M. This means that if xs $\leq$ xa the information that was in the least significant digit of L before the division commenced and which has been shifted up into M is lost. Also the final contents of M will have the wrong sign if the binary digit following the sign bit was a one before the last octal shift in the division loop, because when M is shifted down again this digit will be copied into the guard digits and the sign digit of M.
(iii) the quotient is in L with its exponent in Ya and the remainder (with a sign bit that may have to be corrected by fixed store programme) is in M where its exponent is taken to be

    ya - 12 if |xs| $\leq$ |xa| (12/13 digit set as 1)
    ya - 13 if |xs| > |xa| (12/i3 digit set as 0)
    no interrupt occurs if xs is zero or superstandard.

If xs is zero each octal digit in the final contents of L is a zero or a seven according as to whether the corresponding octal digit in the initial contents of M was greater than or equal to 4 or less than 4. (If on entering the division loop the more significant guard digit is zero i.e. the octal digit is less than 4, the "subtractions" (D-4d), (D-2d) and (D-d) are successful and each of the q digits in turn is set as one. If however the more significant guard digit is a one, the "subtractions" are unsuccessful and the q digits remain set at zero).
The final contents of M are the initial contents of the least significant octal digit of M and the twelve most significant octal digits of L. The final guard and sign digits of M are the same as the initial most significant binary digit of the

[13] This seems to be an error. Surely failure to carry out any of the subtraction, sign testing and setting the Q digits will result in the numerator being shuffled upwards and (entirely or almost entirely) lost off the top end of the mantissa. The quotient will be 0 because no bits are ever added into the bottom. Accordingly this sentence seems curious – and perhaps should be ignored? [Ed.]

least significant octal digit of M. The 12/13 digit is a 1 and the Qs digit the same as the original sign digit of M.
The correct result for floating point working is obtained if

$0 \leq xa < |xs|$

If this instruction is used for “fixed point” arithmetic i.e. when the programmer does not take any account of exponent arithmetic, the correct result is obtained only if

$0 \leq xa \leq |xs| < 1.0$ but not if - 1.0

the 376 order
xs is tested for being sub-standard or zero, the D_0 digit being set and an interrupt occurring if it is. The instruction is completed except that steps (a), (b), and (c) of the division loop are omitted.[14] [15]

the 377 order
This is the same as the 376 order except that, the initial contents of L are cleared.

---

[14] As above, the note about omitting steps a, b and c of the division loop seems to be incorrect. [Ed.]

[15] Following the division loop, 10 is subtracted from the exponent. If the 12/13 digit is not set, xa is shifted down one octal place. [ed.]

3.2 Overflow and Underflow

(a) Exponent Overflow
If as the result of an arithmetic operation, the exponent exceeds +127 i.e. the two most significant bits of B127 are 01, then an exponent Overflow flip-flop (digit 29, line 1 of the Central Computer V-store) is set and an automatic entry is made to a monitoring routine via the normal interrupt procedure. In the case of a multiplication where an intermediate result before the standardisation is non-standard and the exponent exceeds +127, then provided the final result is in range Exponent Overflow does not occur. For example, if the contents of M and Xs are 1/4 and ya = ys = 64, when multiplied the intermediate answer is (1/16, 128) but the final standardised answer is (1/2, 127). This will not cause Exponent Overflow.

It is proposed that in normal circumstances the contents of certain specified registers should be printed out and the programme then terminated. It is, however, possible for the programme to specify that an alternative routine be obeyed. This will form part of the programme being obeyed. The entry to either the automatic post-mortem or the programmer's routine will be governed either by the setting of a B-register or possibly by obeying a jump instruction stored in a particular register in the store.

(b) Accumulator Overflow (AO)
If accumulator overflow is detected (i.e. the accumulator is super-standardised) at the conclusion of an arithmetical operation in the accumulator using the unstandardised instructions then an Accumulator Overflow flip–flop is set in the V–store (digit 27, line 6 of the Central Computer V-store). This does not cause an automatic interrupt. Extracode instructions are provided so that this register can be tested and cleared. This overflow digit can only be set by accumulator operations and it must be reset by fixed store programme.
It should be noted that with these instructions if accumulator overflow does occur no standardisation takes place.

(a) Exponent Underflow
If, during the course of a multiplication, division or standardisation operation, the exponent is less than -128, i.e. the two most significant bits of B124 are 10, the contents of the accumulator are replaced by "floating point zero" i.e. 0×8^{-128} (xa = 0, ya = -128).
In the case of a multiplication where the contents of M and xs are both -1.0 and the sum of the exponents is -129, the final result is xa = 0, ya =-128 instead of xa = 1/8, ya = -128.

3.3. Basic B Functions

All arithmetic takes place over 24 binary digits. The B-carry digit is set or cleared by the instructions marked Bc. It is set by most arithmetical instructions to the value of the carry, or borrow, formed at the most significant digit position in the B-adder.

S is always a 24 bit register which may be either the more or less significant half of a 48 bit word register, the least significant two bits of S being ignored.

Some of the following codes are marked as "Not Assigned" and if specified and obeyed will not alter the contents of any of the B registers, the main store registers or the B test registers. They do, however, perform some operation in the B-adder and may consequently set or clear the B-carry digit. The codes marked "n-type" may be used as dummy instructions. The codes marked "s-type" call for the operand specified in the address digits and may therefore cause either a "Non-equivalence" or a "Sacred Violation by Operand" interrupt. An interrupt may be avoided with these codes by specifying either a fixed store address (i.e. the most significant octal digit of the address being four) or the address of a word known to be in the core store (e.g. the current instruction).

0100 ba' = s - ba Bc
Subtract ba from s and place the result in Ba. The contents of S are not altered.

0101 ba' = s
Copy the 24 bits from S into Ba.

0102 ba' = ba - s Bc
Subtract s from ba and place the result in Ba.

0103 ba' = - s
Place the negative of s in Ba.

0104 ba' = ba + s Bc
Add s to ba and place the result in Ba.

0105 ba' = ba (with circular shift up 6 bits) +s
Shift the contents of Ba up 6 bits, copying the initial most significant 6 bits into the new least significant 6 bits, then add s and place the result in Ba.

0106 ba' = ba ≢ s
Not equivalent ba and s and place the result in Ba.

0107 ba' = ba & s
Collate ba and s and place the result in Ba.

0110 s' = s - ba Bc
Subtract ba from s and place the result in S. The contents of Ba are not altered.

0111 s' = - ba
Place the negative of ba in S.

0112 s' = ba - s Bc
Subtract s from ba and place the result in S.

0113 s' = ba
Copy ba into S.
0114 s' = ba + s Bc
Add ba to s and place the result in S.
0115 As for 0114 except that Bc is unaltered.

0116 s' = ba ≢ s
Not equivalence ba and s and place the result in S.
0117 s' = ba & s
Collate ba and s and place the result in S.
0120 ba' = n - ba Bc
Subtract ba from the integer n and place the result in Ba.
0121 ba’ = n
Place the integer n in Ba.
0122 ba’ = ba - n Bc
Subtract the integer n from ba and place the result in Ba.
0123 ba' = -n
Place the negative of the integer n in Ba.
0124 ba' = ba + n Bc
Add the integer n to ba and place the result in Ba.
0125 ba' = ba (with circular shift up 6 bits) +n
Shift the contents of Ba up 6 bits, copying the initial most significant 6 bits into the new least significant 6 bits, then add the integer n and place the result in Ba.
0126 ba' = ba ≢ n
Not equivalent ba and the integer n and place the result in Ba.
0127 ba' = ba & n
Collate ba and the integer n and place the result in Ba.
0130 Not assigned, n-type code (n-ba) Bc
0131 Not assigned, n-type code (-ba)
0132 Not assigned, n-type code, (ba-n) Bc
0133 Not assigned, n -type code, (ba)
0134 Not assigned, n-type code, (ba+n) Bc
0135 Not assigned, n-type code, (ba+n)
0136 Not assigned, n-type code, (ba≢n)
0137 Not assigned, n-type code, (ba&n)

0140 As for 0100
0141 As for 0101
0142 As for 0102

0143 ba' = ba (with circular shift down one bit) - s
Shift the contents of Ba down one bit, copying the initial least significant bit into new most significant place, subtract s and place the result in Ba.

0144 As for 0104

0145 As for 0105

0146 ba' = ba v s
Perform the logical "OR" on ba and s and place the result in Ba.

0147 As for 0146.

0150 bt' = s - ba Bc
Set Bt in accordance with the result of subtracting ba from s. The contents of Ba and S are not altered.

0151 Not assigned, s-type code (s-ba)

0152 bt' = ba - s Bc
Set Bt in accordance with the result of subtracting s from ba.

0153 Not assigned, s-type code (ba-s)

0154 Not assigned, s-type code (ba+s) Bc

0155 Not assigned, s-type code (ba+s)

0156 Not assigned, s-type code (ba vs)

0157 Not assigned, s-type code (ba vs)

0160 As for 0120

0161 As for 0121

0162 As for 0122

0163 ba' = ba (with circular shift down one bit) -n
Shift the contents of Ba down one bit, copying the initial least significant bit into the new most significant place, subtract the integer n and place the result in Ba.

0164 ba' = ba + (bm & n) Bc
Collate bm and n and add the result to ba, placing this final result in Ba. If Bm=0 this code is obeyed as if it were a 124 code.

0165 ba' = bm & n
Collate bm and n, placing the result in Ba. If Bm=0 this code is obeyed as if it were a 121 code.

0166 ba' = ba v n
Perform the logical "OR" on ba and n and place the result in Ba.

0167 As for 0166

0170 bt' = n - ba Bc
Set Bt in accordance with the result of subtracting ba from the integer n.

0171 Not assigned, n-type code (n - ba)

```
0172  bt' = ba - n    Bc
            Set Bt in accordance with the result of subtracting the
            integer n from ba.

0173  Not assigned, n-type code (ba - n)
0174  Not assigned; n-type code (ba + n)   Bc
0175  Not assigned, n-type code (ba + n)
0176  Not assigned, n-type code (ba v n)
0177  Not assigned, n-type code (ba v n)
```

3.4 Basic Test Functions

As function digit 19 is not decoded for the test functions, two codes are available for each function. In some cases further duplicate codes are also available.

The contents of Ba are unaltered by these instructions if the test does not succeed. If Ba is the control register in current use, the contents are stepped on by 1 in the usual way and then if the test succeeds, replaced by n.

In instructions 0200-0207 and 0220-0223

(i) bm is altered by 0.4. or 1.0 in order that consecutive 24 or 48 bit words may be selected. This alteration takes place only if the test succeeds.

(ii) if Ba = Bm the final contents of the specified B-register are n if the test is successful i.e. the alteration by 0.4 or 1.0 is overwritten.

In instructions 0224-0237 the Bm digits are used for B modification as usual.

0200 If bm ≠ 0, ba' = n and bm' = bm + 0.4
0240 If the contents of Bm are not zero, place the integer n in Ba and add 0.4 to the contents of Bm. Otherwise leave Bm and Ba unaltered

0201 If bm ≠ 0, ba' = n and bm' = bm + 1.0
0241 If bm is not zero, place the integer n in Ba and add 1.0 to bm

0202 If bm ≠ 0, ba' = n and bm' = bm -0.4
0242 If bm is not zero, place the integer n in Ba and subtract 0.4 from bm

0203 If bm ≠ 0, ba' = n and bm' = bm -1.0
0243 If bm is not zero, place the integer n in Ba and subtract 1.0 from bm

0204 As for 0200
0244

0205 As for 0201
0245

0206 As for 0202
0246

0207 As for 0203
0247

```
0210  If bm is odd, ba' = n
0250        If the least significant bit of bm is 1, place the
            integer n in Ba
0211  If bm is even ba' = n
0251        If the least significant bit of bm is 0, place the
            integer n in Ba
0212  As for 0210
0252

0213  As for 0211
0253

0214  If bm = 0, ba’ = n
0254        If bm is zero, place the integer n in Ba
0215  If bm ≠ 0, ba' = n
0255        If bm is not zero, place the integer n in Ba
0216  If bm ≥ 0, ba' = n
0256        If the most significant bit of bm is 0, place the
            integer n in Ba
0217  If bm < 0, ba' = n
0257        If the most significant bit of bm is 1, place the
            integer n in Ba
0220  If bt ≠ 0, ba' = n and bm' = bm + 0.4
0260        If Bt is non-zero, place the integer n in Ba and add
            0.4 to bm. Otherwise leave Ba, Bm unaltered.
0221  If bt ≠ 0, ba' = n and bm' = bm +1.0
0261        If Bt is non zero, place the integer n in Ba and add
            1.0 to bm
0222  If bt ≠ 0, ba’ = n and bm' = bm -0.4
0262        If Bt is non-zero, place the integer n in Ba and
            subtract 0.4 from bm
0223  If bt ≠ 0, ba' = n and bm' = bm -1.0
0263        If Bt is non-zero, place the integer a in Ba and
            subtract 1.0 from bm
0224  If bt = 0, ba' = n
0264        If Bt is zero, place the integer n in Ba
0225  If bt ≠ 0, ba' = n
0265        If Bt is non-zero, place the integer n in Ba
0226  If bt ≥ 0, ba' = n
0266        If Bt is set ≥ 0, place the integer n in Ba
0227  If bt < 0, ba' = n
0267        If Bt is set < 0, place the integer n in Ba
0230  As for 0234
0270
```

0231 As for 0235
0271

0232 As for 0236
0272

0233 As for 0237
0273

0234 If ax = 0, ba' = n
0274 If ax is zero, place the integer n in Ba. The sign bit of L is not tested.

0235 If ax ≠ 0, ba' = n
0275 If ax is non-zero, place the integer n in Ba. The sign bit of L is not tested

0236 If ax ≥ 0, ba' = n
0276 If ax is positive or zero, place the integer n in Ba

0237 If ax < 0, ba' = n
0277 If ax is negative, place the integer n in Ba

3.5. Basic Accumulator Functions

The accumulator functions all operate on floating point numbers. However, certain of the following instructions can be used for fixed point working provided that, when necessary, the exponents of the numbers being operated on are the same (they will usually be zero).

The instructions involving an arithmetical operation which are suitable for fixed point working are those which set the Accumulator Overflow register when appropriate. Several of the instructions are marked as setting either the Exponent Overflow Register or the Accumulator Overflow Register when appropriate even though no occasions should arise when the registers will be set. They are so marked because the accumulator is tested for these overflow conditions.

For some of the following codes a store operand is not required (i.e. 340, 341, 350, 351, 354, 355, 360, 361, 364-366, 370, 371). Reference is however still made to the store register specified by the S digits in the instructions and consequently a non-equivalence or a “Sacred Violation by Operand” interrupt may occur. No such interrupt occurs if a fixed store address (i.e. an address whose most significant octal digit is a four) or the address of a word known to be in the core store (e.g. the current instruction) is specified.

Further details of the methods of carrying out some of the Accumulator operations are given in Section 3.1.

0300 EO a’ = a + s (Q)

Clear L including the sign digit and add the floating point number in S to am. The result is standardised as a double length number. The exponent overflow register is set and an interrupt occurs if the exponent overflows.

0301 EO a’ = am - s (Q)

If ya≥ys the number in S is negated and the instruction then proceeds as for 0300. If va<ys the number in S is subtracted from the number taken from AM and the result is standardised as a double length number

0302 EO a’ = - am + s (Q)

As for 0301 except that the roles of S and AM are interchanged.

Note In instructions 0300–0302 no arithmetic involving both numbers takes place if the exponents differ by 32 or more, but am is set to the number with the larger exponent, this number being negated if specified by the instruction.

0303 As for 0302

0304 EO am’ = s, l’= 0 (Q)

Clear L including the sign bit, then copy the contents of store location S into AM and standardise as a double length number. No case can arise when the exponent overflow register is set by this instruction.

0305 EO am’ = -s. l’ = 0 (Q)

Clear L including the sign bit then put the negative of the contents of store location S into AM and standardise as a double length number. Exponent Overflow occurs if ys = 127, xs – 1.0

0306 EO am’ = s, l’ = 0 (Q)
This is similar to 0304 except that the “reverse minus” circuits in the accumulator are used. No case can arise when Exponent Overflow occurs.
0307 As for 0306
0310 EO a’ = am + s (non-clear L) (Q)
As for 0300 except that L is not cleared initially
0311 EO a’ = am - s (non-clear L) (Q)
As for 0301 except that L is not cleared initially
Note Instructions 0310 and 0311 are useful for double length working only if ys ≥ ya
0312 As for 0302
0313 As for 0302
0314 AO am’ = s (non-clear L)
Copy the floating-point number in store location S to the exponent of and more significant half of A. The less significant half of A is unaltered.
No case can arise when the accumulator overflow register is set by this instruction.
0315 AO am’ = -s (non-clear L)
Negate the contents of S and copy the result into AM. The accumulator overflow register is set if the contents of Xs = -1.0. The initial contents of L are unaltered
0316 As for 0306
0317 As for 0306
0320 EO am’ = am + s (QR)
Clear L including the sign bit and add the floating point number in S to the floating point number in AM; the result is standardised as a double length number and rounded by forcing a one into the least significant bit of AM, except when L = 0.
If the exponent ya overflows, the exponent overflow register is set and causes an interrupt.
0321 EO am’=am - s (QR)
If ya ≥ ys the number in S is negated and the instruction then proceeds as for 0320.
If ya<ys the number in S is subtracted from the number taken from AM, standardisation and rounding then being as for 0320.
0322 EO am’ = - am + s (QR)
As for 0321 except that the roles of S and AM are interchanged.
Note In 0320, 0321 and 0322 if the exponents differ by 16 or more, no arithmetic takes place involving both numbers. The contents of AM are replaced by the larger[16] of the two numbers, negated if specified in the instruction and standardised. Rounding off would occur if any case needed it, e.g. if ya-ys ≥ 16, and am contained a superstandard number initially with the least significant octal digit in M non-zero.
0323 As for 0322
0324 As for 0304
0325 As for 0305
0326 As for 0306
0327 As for 0306

[16] Presumably this means whichever number has the larger exponent. [ed.]

0330 AO a’ = am + s

Clear L including the sign bit and add the floating point number in S to that in AM. If the result is superstandardised the Accumulator Overflow register is set.

0331 AO a’ = am-s

Clear L including the sign bit and either negate the number taken from S and add it to that taken from AM or subtract the number taken from S from that in AM according to whether ys is less than or equal to or greater than ya.

0332 AO a’= -am + s

As for 0331 except that the roles of S and AM are interchanged.

Note In instructions 0330-0332 no arithmetic involving both numbers takes place if the exponents differ by 32 or more, but am is set to the number with the larger exponent, this number being negated if specified by the instruction.

0333 As for 0332

0334 AO am’ = s; l’ = 0

Clear L including the sign bit and copy the floating point number in store location S to the exponent of and more significant half of A.

Note This instruction is marked AO although it cannot in fact set the accumulator overflow register.

0335 AO am’ = -s, l’= 0

Clear L including the sign bit then put the negative of the number in S into the more significant half of A. The exponent of the number in S being copied to ya. The accumulator overflow register is set if xs = -10.

0336 As for 0334

0337 As for 0334

0340 EO a’ = a (Q)

Standardise the double–length number in A and test for Exponent Overflow. A is set to floating point zero if either (i) the initial contents of M are sub-standard and after standardisation ya is less than –128 or (ii) the initial contents of M and L are zero.

0341 EO a’ = a

Set the exponent overflow register and interrupt if the contents of ya are greater than +127.

0342 EO a’ = am. S (Q)

Form the double-length product of the numbers in AM and S and standardise. If Ya overflows the exponent overflow register is set and an interrupt occurs.

0343 EO a’ = -am.s (Q)

Similar to 0342 but the negative product is formed.

0344 l’= xs, ya’ = ya

Copy the least significant 39 bits of the number in S to the less significant half of A and the sign bit of L. The exponent of A is unchanged.

0345 l’= xs; m’ = sign xs; ya’ = ya

Copy the least significant 39 bits of the number in S into L and copy the sign of the number in S to the 40 digits of M and to the sign bit of L. The exponent of A is unchanged.

0346 s’ = a ; a’ = 0

Copy am to S and clear A to floating point zero i.e. ya = -128, M and L (including the sign bit of L) = 0

0347 s’ = al; l’ = 0

Copy the exponent of A, the sign bit of L and the least significant 39 bits of A to S; clear L including the sign bit of L.

0350 As for 0340

0351 As for 0341

0352 EO, AO a’ = am.s ; sign l’ = sign m’

Form the double length product of the numbers in AM and. S. The sign bit of L is made the same as the final sign of M. Accumulator overflow occurs if the initial contents of M and S are -1.0.

0353 EO, AO a’ = - am.s ; sign l’ = sign m’

Similar to 0352 but the negative product is formed. The result is tested for Accumulator Overflow although it cannot occur unless the initial contents of A were superstandardised.

0354 AO am’ = a (R+)

Add one to the least significant digit of M if the most significant digit of L is a one. The contents of L are left unchanged.

0355 EO, am = l (Q)

Clear M, copy the sign of L into M and standardise A. The sign of L is not cleared. Exponent overflow is tested for although it cannot occur.

0356 s’ = am

Copy the exponent and the more significant half of A to S.

0357 s’ =al

Copy the exponent of A, the sign bit of L and the less significant half of A to S.

0360 EO am’ = a (QR)

Standardise the double length accumulator and round AM by forcing one into its least significant digit except when the final contents of L are zero. The exponent overflow register is set and an interrupt occurs if the contents of Ya are greater than +127.

O361 EO am’ = a (R)

Round AM by forcing a one onto the least significant digit except when l = 0. The exponent overflow register is set and an interrupt occurs if the contents of Ya are greater than +127.

0362 EO am’ = am.s (QR)

Form the double length product of the numbers in AM and S; standardise and round this result. If Ya overflows, the exponent overflow register is set and an interrupt occurs.

0363 EO am’ = am.s (QR)

similar to 0362 but the negative product is formed.

0364 xa’ = 8.xa, ya’ = ya

Shift the contents of the accumulator up one octal place. The exponent is unchanged.

0365 xa’ = 8.xa, ya’ = ya[17]
Shift the contents of the accumulator down one octal place. The shift is signed, copying the most significant guard digit. The exponent is not changed.

0366 EO am’ = |am| (Q)
Clear L, complement the contents of Am if negative, and standardise. Exponent overflow is set and an interrupt occurs if appropriate. L may finally contain one octal digit if M was in superstandard form initially.

0367 EQ am’ = |s| (Q)
Clear L, copy the modulus of the contents of S to AM and standardise the double length accumulator. Exponent overflow is set and an interrupt occurs if appropriate.

0370 As for 0340

0371 As for 0341

0372 EO, AO a’ = am.s; sign L’ = 0
Form the double length product of the numbers in AM and S. The sign bit of L is cleared. If Ya overflows, set the exponent overflow register and interrupt. If the accumulator becomes superstandardised (i.e. by multiplying two numbers whose arguments are both -1.0) the accumulator overflow register is set.

0373 EO, AO a’ = -am.s; sign L’ = 0
Similar to 0372 but the negative product is formed. The accumulator is tested for overflow, although it cannot occur unless the initial contents of A were superstandardised.

0374 EO, DO am’ = am ÷ s (QR)
Clear L and divide am by s, the result being rounded by forcing one into the least significant bit of the quotient before standardisation if there is a remainder. An interrupt occurs if s is sub-standard or zero (the DO digit in line 1 of Central Computer V-store being set) or if exponent overflow occurs.

0375 EO al’ = a ÷ s where a must be positive or al’ = am ÷ s if initial l = 0 ya’ = exponent of the quotient. am’ = positive remainder of a ÷ s where its exponent is taken to be

(i) ya - 12 if |xs| ≤ xa

(ii) ya - 13 if |xs| > xa

An interrupt occurs if exponent overflow occurs. No interrupt occurs if s is sub-standard or zero. The correct result for floating point working is obtained if:

0 ≤ xa < 8.|xs.|

If this instruction is used for “fixed point” arithmetic i.e. when the programmer does not take any account of exponent arithmetic, the correct answer is obtained only if

0 ≤ xa < |xs| < 1 but not if xs = -1.0

---

[17] The ABL manual confirms that this should read “ax’ = ax / 8; ya’ = ya” – ie divide by 8, as per the description given here. [Ed]

0376 EO, DO al' = a ÷ |s| where a must be positive
ya' = exponent of the quotient
m = remainder of a ÷ s[18] where its exponent is taken to be:
(i) ya - 12 if |xs| ≤ xa
or (ii) ya - 13 if |xs| > xa
An interrupt occurs if s is sub-standard or zero or if exponent overflow occurs.

0377 EO, DO al' = |am| ÷ |s|, ya' = exponent of the quotient
M' = remainder of |am| ÷ |s| where its exponent is taken to be
(i) ya - 12 if |xs| ≤ |xa|
(ii) ya - 13 if |xs| > |xa|
An interrupt occurs if s is sub-standard or zero or if exponent overflow occurs0

----

[18] It seems likely that this should strictly be “a ÷ |s|” [Ed.]

Section 4. The Main Store including arrangement of Drum Transfers

4.1 The One-Level Store Concept

The programmer normally writes instructions addressing the available core and drum store directly. Information cannot be transferred between the drum store and the accumulator and B-Registers without first being put into the core store and any drum transfers implied by this are handled by a combination of machine facilities and fixed store program.

As all drum transfers are in blocks of 512 forty-eight bit words the core store can be considered to be divided into p pages each of 512 words. Similarly the drum store can be considered as s sectors each of 512 words. The total storage capacity of an installation is thus b blocks of information, each block consisting of 512 words. The total storage capacity of an installation is thus b blocks of information, each block consisting of 512 words, where b = p + s. Associated with each page of the core store there is a Page Address or Coincidence Register consisting of 12 flip-flops. Eleven of these flip-flops contain the block number of the information at present in the corresponding page. The twelfth flip-flop is a "lock-out" digit which is set when the block is not available to the main program, e.g. during drum and peripheral equipment transfers to that block. (The remaining bits of the addresses do not need to be stored in these registers as they only give the location of a word within the page or block.) These Page Address Registers are part of the Core Store V-store. Also in this V-store is a "Use" digit for each page of the core store.

When an instruction referring to an address in the main store is to be obeyed, a comparison is made between the (B-modified) address and each of the p page address registers to determine if the word referred to already lies in the core store. If it does, the instruction is obeyed in the ordinary way. It should be noted that if the address specified is m.1 (where 1 is the line within the block in the main store), the address of this word in the core store will be c.1 where m and c are normally not the same. This address c.1 is therefore the one whose contents are operated on by the machine.

If the location referred to does not lie in the core store, the current control number is stored and a special drum transfer program in the fixed store is entered. Associated with this program there are (p + s) words in the subsidiary store, i.e. page and sector directories, giving the block numbers of the information stored on each page and sector. Whenever a block of information is transferred to a sector, a digit is set in the corresponding sector directory address to indicate that this sector is now storing useful information. When the block is read to the core store this digit is cleared.

The drum supervisor is divided into two parts, the first consisting of routines for transferring blocks of information between the core store and the drum and the second the routines for determining which core store page should be used for a transfer.

The decision as to which page should be made available is made by a Drum Transfer Learning Program (see section 4.2) with the help of the "Use" digits referred to above. The sector to be written to is decided by reading a sector address from the drum and from the information in the sector directory working out which is the first sector not containing useful information that can be written on.

After the required block has been transferred from the appropriate sector to the page available for it in the core store the necessary alterations to the page and sector directories are made; the Page Address Register associated with the page in the core store now containing the block of information is also altered to contain the block number. These registers are set by the fixed store drum program.

In order that the maximum speed may be gained from Atlas, operands and instructions should be on different pairs of stacks and an attempt is made by the store supervisor to keep them so placed. For this purpose a digit in the V-store (line 34 digit 47), which indicates whether a non-equivalence interrupt is due to an instruction or an operand not being available, is used. Whenever possible instructions are read down to pages 0 - 15 and operands to the remaining pages of a machine. However, if no vacant page is available in one of the appropriate pages for operands the transfer is made to a vacant page even though this page is not in the preferred region. If pages 0 -. 15 are out of action, instructions are read down to any page.

It has been found that for a machine with 32 pages in the core store it is advantageous to make available 31 pages for programmers and always have one page free to receive the next block to be transferred from the drum store. Thus, when non-equivalence is obtained with the Page Address Registers the required block is immediately brought to the core store and whilst this is being done it can be worked out which page shall be made free in anticipation for the next transfer. The writing of this page to the drum is completed whilst the main program is continued.

After the completion of the transfer from the drum to the core store, the control register is reset and the instruction which caused interruption of the program recommenced.

It is envisaged that most programs will use the core and drum store as a one-level store. However, in certain circumstances it may be useful to be able to specify that a given block of information should be either in the core store or that it can be written to the drum. For this purpose the Store Extracode orders listed in section 2.4.2 are provided. It should be noted that the same routines are obeyed in a transfer called in by an extracode order as by a non-equivalence interrupt.

4.2 Drum Transfer Learning Program

This routine is one used as a sub-routine by the store supervisor. Its purpose is to indicate which block of information in the core store is least likely to be referred to and hence which block should be written to the drum when further space in the core store is required for a new block of information, either from the drum or from a magnetic tape. The store supervisor specifies that the selected page should be one of pages 0 - n, inclusive, and this limitation can be used to influence the choice of either an "instruction" page (0 - 15) or an "operand" page being made available.

To enable it to make this decision, the Use digits have been provided. These are a set of digits, which, with the Page Address Registers, are part of the Core Store V-store. There is one Use digit associated with each page in the core store. The use digit for a page is set (or operation) whenever reference is made to the block of information currently stored in that page.

A further machine facility which is used by the Learning Program is an "Instruction Count Interrupt". In the Peripheral Equipment V-store there is an Instruction Counter and the count in this register is increased by one whenever an instruction is obeyed on either main or extracode control. The appropriate "look at me" is set every 2048 instructions causing an automatic interrupt.

Every time such an interrupt occurs the Use digits are read out of the Core Store V-store and put in a list kept in the Subsidiary store. The registers containing the Use digits are automatically cleared i.e. reset to zero when the information is read from them. The number of instructions obeyed during one of these interrupts is (6 + i + 2n/16) where

i is the number of instructions obeyed on entering and leaving an interrupt. This is six if on leaving the routine a further interrupt has occurred and is waiting to be obeyed and it is nine if no further interrupt digit is set.

n is the number of pages of core store in the machine.

If the Use digit for a particular page is a one when it is read from the V-store this indicates that that page has been referred to at least once since the last Instruction Count Interrupt. Conversely, if the Use digit is zero, that page has not been referred to since the last such Interrupt.

With each block of information the Learning Program associates three parameters t, ζ, and T. The object of these parameters is to give some guidance to the frequency with which each block of information is referred to. The three parameters are:

t the length of the present period of inactivity of the block at present in the appropriate page of the core store i.e. the number of successive Instruction Count Interrupts that have occurred since the block was last referred to.

T the length of the last period of inactivity of a block, part of which was time spent on the drum.

ζ the start of the present period of inactivity of a block which is written to the drum. Each program has associated with it a "clock" (i.e. a programmed count of the number of Instruction Count Interrupts that occur whilst it is being obeyed) in the Subsidiary Store and it is possible to calculate T from ζ and the number of interrupts that have occurred between the writing up and the reading down of the block.

The parameter t is brought up to date either (i) when a non-equivalent interrupt occurs or (ii) when the Use digit table is full or (iii) on a program change. An exception to the first condition i.e. non-equivalence, is if there is an empty page available for the next transfer from the drum and in this case the list of Use digits is left unaltered. Only the values of t for the program being obeyed or left are altered and the values of t for any pages remaining in the core store from a previous program are not altered.

The parameters T and ζ are set as appropriate whenever a drum transfer takes place.

With these parameters and the limitation specified by the store supervisor that the selected page must be in the range 0 - n, the following rules are used to decide which block shall be written to the drum:-

1) check if any page in the core store is still vacant because no transfer to the drum is required in this case.

2) see if any block at present in the core store is waiting to be written to the drum. This is indicated by the setting of digit 0 in the half word containing the parameter T. (For this and similar criteria if two or more blocks satisfy the relation then the one on the highest numbered page is selected.)

3) look for a page which has a negative value of (T+1 - t); if more than one pick the page with the largest negative value.
The relation t>(T+1) means that a block has been inactive in the core store for a period longer than its last inactive period. The value (T+1) rather than T is used to compensate for transfers taking place at arbitrary times with respect to the Instruction Count Interrupts, thus leading to different values of the parameters on different occasions for the same inactive period.

4 if for any page (T-t) is positive and t ≠ 0 select the page according to the following list of priorities
(a) containing a block of instructions in page 16 or above
(b) containing a block of operands in page 16 or above
(c) containing a block of operands in pages 0 - 15.

(d) containing a block of instructions in pages 0 - 15.

If two or more blocks have the same priority (a, b, c or d) select the one with maximum positive (T-t)

5) select the page with maximum value of T

4.3 Extracodes for Drum Transfers

It is envisaged that most programmers will use the core and drum stores as a one-level store. However, in certain circumstances it may be useful to be able to specify that a given block of information should be either in the core store or that it can be written to the drum store. For this purpose the following drum transfer extracode orders are provided.

1050 Read block P

If there is no suitable empty page a block of information at present in the core store is written to the drum. If the block is already in the core store nothing is done. The original (drum) copy is lost i.e. the sector originally containing it is indicated as "empty" in the sector directory.

1051 Write block P

If P is already on the drum nothing is done. The page in the core store originally containing the information is now regarded as free.

1052 Duplicate

Duplicate block P1 calling the copy P2; P2 in core store/drum depending on whether P1 was on the drum/core store originally.

1053 Rename

Rename block P1 and call it block P2.
The original block called P2 (if any) is lost.

Before using extracodes 1054, 1055, 1056, which refer to sector numbers, drums are reserved by a statement in the title of the programme. The programmer always numbers the drums allocated to him as drum 0,1 etc.

1054 Read n blocks

Read n blocks beginning at sector s and label the new copies in the core store blocks P onwards.
The transfer continues round the specified band of the drum so that n ≤ 6. The contents of the sectors are retained but any original blocks P to (P + n - 1) are lost.

1055 Write n blocks

Write n blocks from P onwards to n sectors beginning at sector s.
The transfer continues round the specified band of the drum so that n < 6.

The blocks are retained in the core store and the original information on the sectors is overwritten. If waiting time between successive transfers in a multiple length transfer to (or from) the drum is to be avoided it is necessary that the sectors concerned are all on the same band. This leads to two limitations in this and the previous instruction.

(i) $n \leq 6$ and

(ii) the sectors written to or read from are all on the same band.

For example, if n = 4, P = 10, s = 3

block 10 in the core store is written to sector 3
block 11 in the core store is written to sector 4
block 12 in the core store is written to sector 5
block 13 in the core store is written to sector 0 (and not on band 1).

If, therefore, a multiple length transfer involving two (or more) bands is required, it must be specified in two (or more) instructions.

1056 <u>Lose Sector n</u>

The sector n is made empty and becomes available to the one-level store.

1057 <u>Lose block P</u>

The sector or page occupied by block P is made empty.

4.4 Basic Drum Transfers

When a programmer requires a drum transfer (specified by an extracode order or implied by non-equivalence), the transfer is initiated by a fixed store programme setting various digits in the V-store. In the following paragraphs (describing essentially the fast drum system) in this section all references to particular lines and digits are to the appropriate lines and digits in the Drum V-store (see Section 13.4).

The following information is necessary to specify the location on the drum to or from which the transfer is to take place

(a) which cabinet the drum is in (line 52, digits 28-26)
(b) which drum within the cabinet is required (line 48, digits 27-26)
(c) which band on the drum is required (line 44, digits 28-26)
(d) the starting θ (line 40, digits 29-27)
    The θ registers (lines 0-31, digits 29-27) give the angular position of each of the fast drums. There are six sectors on each band for the fast drums and hence these registers are set in the range 0-5 only. They are set automatically by the machine.
(e) how many blocks of information are to be transferred (line 36). For a multiple block transfer all the sectors must be on the same band.

Whenever it is necessary to change the band selected (i.e. different band, drum or cabinet) a period of approximately 1.5 milliseconds must be allowed for "settling-down" time before the next transfer can be carried out. In many cases (e.g. multiple length transfers or writing to the next available vacant sector) it is advantageous to be able to find out if it is possible to select and transfer to or from the next sector without waiting for a complete drum revolution. The "1.5- 2" digits (lines 0-31, digit 26) are provided for this purpose. The appropriate flip-flop is reset to zero whenever the contents of the corresponding θ register are changed and is set to one when it is too late to select the next sector for an immediate transfer. To give the fixed store programme time to complete the necessary organisation for the transfer a setting time of 0.1 m.s. is allowed for this digit i.e. if the digit is read as a zero there is at least 0.1 m.s. for the programme to select the next sector for an immediate transfer. This 0.1 m.s. setting time has, of course, to be regarded as the maximum time allowed otherwise the digit must be re-examined.

To specify a page or pages in the core store for a drum transfer it is necessary that a Page Address Register or Registers be set as follows:

digit	22	21	20	19	18	17	16	15	14	13	12
	1	1	1	1	0	1	1	1	δ	δ	δ

For a single block transfer digits 11+-12 must be all zero. For a multiple block transfer (also a File drum transfer) digits 1J+-12 must be set appropriately for the pages that are to be affected by the transfer. Thus, for a transfer involving 2,048 words it is necessary to set digits 14-12 of any four Page Address Registers to 000, 001, 010, and 011 respectively. Digit 23 is the lock-out digit and is normally set as a one.

If non-equivalence is obtained between the transfer address and all the page address registers due to either a page address register not being set correctly or a fault occurring the "Non-equivalence, drums or tape" digit in the Central Computer V-store (line 1, digit 28) is set and an automatic interrupt occurs.

The required starting sector and number of blocks having been specified in lines 36-52, the control digits in line 56 must be set to initiate the transfer. For this it is necessary,

(a) to set the start digit,
(b) to write a one to the Selection Change digit whenever a different band is required from that last selected
(c) to specify either a read or write transfer.

After the completion of each transfer an automatic interrupt occurs. When a transfer has been completed the relevant page address registers, lockout digits and page and sector directories are brought up-to-date. The automatic interrupt on a multiple block transfer occurs when all the blocks have been transferred and not after each individual block transfer.

After being initiated the transfer is carried out automatically with one word being transferred about every 4µ.s. Alternate words are transferred to two stacks of the core store and the appropriate stack is occupied for one cycle of about 2µ.s. during the transfer of each word to it. If other reference is made to the core store normal operations are continued at a reduced rate until the transfer is completed, the drum transfer having priority.

If a transfer involving a cabinet is specified and it is impossible to select this cabinet (due to a fault of some kind as the Supervisor only allows transfers to existent drums) the Drum Cabinet Absent Look At Me digit is set and an automatic interrupt occurs.

Associated with each twenty-four bits of information on the drum there is a parity digit. (The parity digit is such that the sum of the digits is odd). The Drum Transfer Parity digit (line 0, digit 30 of the Central Computer V-store) is set and an interrupt occurs if incorrect parity is detected

(a) in the contents of the core store page involved in a write transfer or
(b) in the information on a drum sector for a read transfer.

In either case the transfer is completed, the parity digit for each word being checked but no new parity digit being generated in the case of incorrect parity being detected. No indication is given in the V-store as to which word has incorrect parity.

The parity digit for the even half-word of a 48 bit word is stored on the drum with the odd half word of the pair and the parity digit for the odd half-word is stored with the even half-word of that pair. If a clock pulse is missed during a reading transfer, i.e. a half-word is not transferred, the subsequent half words read will not be checked against the parity digits associated with them. In most cases of this kind therefore a Parity Failure interrupt can be expected to occur.

As a further check a marker is placed in the gap between successive blocks. When this marker is detected on either read or write transfers a check is made to see that the count of the number of words having been transferred is a multiple of 512. If the count is not correct a Drum Transfer Failure occurs. This check prevents encroachment on the next block, particularly important on write transfers, should clock pulses be missed since the transfer is only deemed complete when the count reaches 512.n where n is the number of blocks specified in the transfer.

If it discovered that a transfer is being carried out incorrectly (e.g. if a Page Address Register does not contain the expected number and a non-equivalence, drum or tape interrupt occurs) it is possible to terminate the transfer immediately by writing a one to the Stop Command register (digit 29 of line 60 of the Drum V-store). The One Second Interrupt programme checks that any transfer initiated before any such interrupt is completed before the next similar interrupt. Thus if an incorrect sector is specified ($\Theta = 6$ or 7), or a drum, or a band on a drum is not selected correctly, at most two seconds will elapse before the fault is discovered.

There are 32 Band Isolation switches for each cabinet, one for each band of the drums. When a band is isolated it is impossible to write to a sector on it but no indication is provided in the machine to say that either a band is isolated or a write transfer has not been carried out due to the band being isolated. In this latter case the machine attempts to write on the drum but the information there remains unchanged. These isolation switches are used to protect certain programmes stored more-or-less permanently on the drums.

Section 5 - The Operating System

5.1 The Internal Organisation

Introduction

The operating system (controlled by the Supervisor programme) described below is applicable to any Atlas computer and is to be used on Muse. It is designed to operate with any configuration of one level store, magnetic tape system and peripheral input and output equipments. If no magnetic tapes are available the system can still be used but its efficiency may be reduced.

During the operation of most slow peripherals the central computer is required for a small percentage of the time, e.g. about 1% for a paper tape reader. Thus even with many slow peripherals operating, the central computer is available for direct execution of a problem for a large proportion of the time. The design of Atlas is such that these problems cannot interfere with the fixed store programmes controlling the peripheral equipments, and the "Supervisor Programme" prevents the fixed store programmes from interfering with the main problems in the machine. The time taken to switch control between the main programme and the fixed store programme is about 10μsecs. This time-sharing of the central computer is referred to as the "overlapping" of input, output, and computing.

If the slow peripheral equipments could always transfer information at the rate required by the central computer for any problem, then the maximum overlap of input, computing and output could be easily achieved. However, the central computer requires and produces information at widely varying rates, with an upper limit in excess of three hundred 512 word blocks per second. On the Manchester University Atlas, with four paper tape readers and one card reader, the maximum input rate with all equipments operating simultaneously is half a block per second. The maximum output rate with four paper tape punches, one card punch, and one line printer is also half a block per second. Magnetic tape input and output can increase these rates to sixteen blocks per second per channel. The use of magnetic tape for input and output requires a large amount of off-line editing equipment, and unless a large number of channels is available, it is still possible for the central computer to be input or output limited for short periods of time.

One method by which the amount of overlap could be increased is to hold several problems within the central computer at any one time, and, by means of the supervisor programme arrange that control is switched between these problems in such a way that the rates at which information is required or produced by the central computer are reduced to those which can be dealt with by the available slow

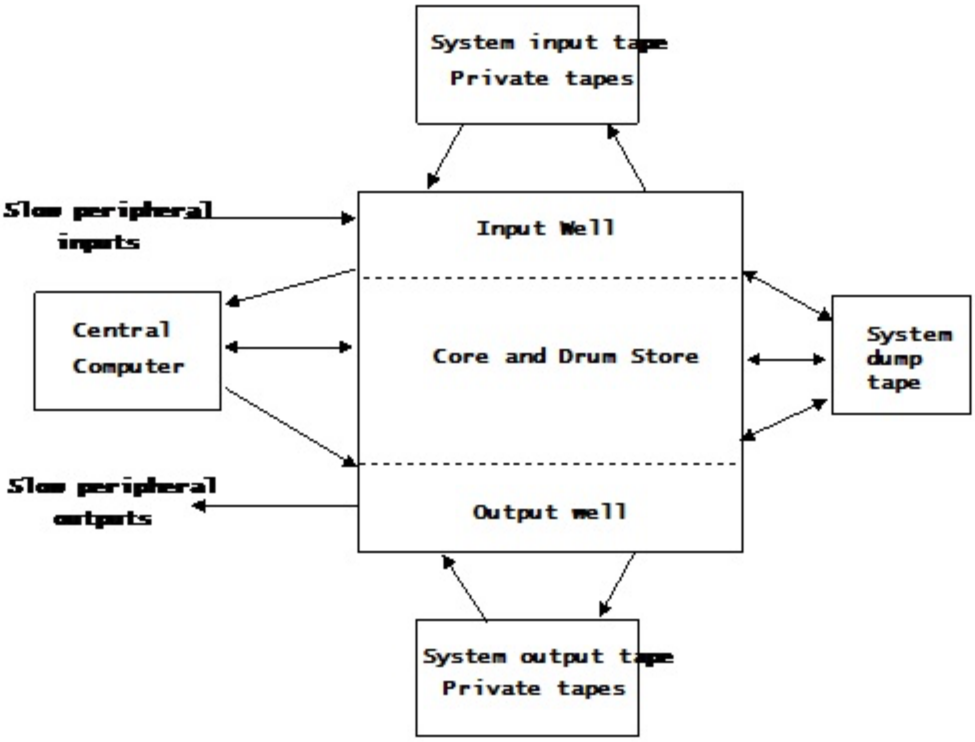
---

This section is based on a paper submitted by Prof Kilburn and Drs Howarth, Payne and Sumner for publication in “The Computer Journal”.

peripherals. In order to achieve this, it is necessary that the available problems are such that, whilst some of them are producing results, others are computing and others require the input of further information. This needs either a large number of programmes available within the central computer or off-line organisation of the problems presented to the computer. Furthermore, this switching between problems can considerably reduce the efficiency of the computer, as it is necessary to store and reset the contents of all the common working registers, requiring up to 1.5msecs on Muse.

An alternative solution is to designate two areas of the one level store as input and output wells, and to arrange that the central computer draws information from one well and supplies it to the other. These wells are filled and emptied by the slow peripherals operating in parallel. The use of these wells "smooths out" the variations in the rates at which information is required or produced by the central computer. The larger the wells, the greater the "smoothing out" and the greater the overlap obtainable for any sequence of problems. This method overlaps the computing of one problem with the input and output of others; only one problem need be in the central computer at any one time, and no switching of control between problems is necessary. The division of the one level store between the central computer and the two wells is arranged by the supervisor programme, and is capable of continuous variation. The size of the wells is increased by the use of magnetic tapes, and another magnetic tape is provided to enable the suspension and dumping of problems. whose continued operation would disrupt the efficient operation of the system. Suspended problems are re-entered when the supervisor programme considers such a re-entry to be worthwhile.

A block diagram of the system is shown below.



<u>The Input System</u>

In order to accelerate the filling of the input wells by the slow peripherals, all slow peripherals are permitted to operate simultaneously. Furthermore, in order that the Input System should be as flexible as possible, it is not necessary for the input of one problem to be on one "tape" (in this context, "tape" implies a paper tape or a stack of cards) or to be through one equipment, or even for the component parts to be supplied in any particular order. All sections of input are headed by a title for the purpose of identification. A complete problem is referred to as a "job" and one of the component sections contains a job title together with a list of the titles of the other sections. The input information is organised within the system by the supervisor programme, which maintains lists of the titles and lists of incomplete and complete jobs. Only jobs whose slow peripheral input is completed are made available to the central computer. The system is best described by considering what happens when "tapes" having the various permissible headings are read into the core and drum store. A tape bearing a "job" title causes this title to be added to the list of incomplete jobs, and a system number is allocated to this job. The component titles of the job description are added to the list of titles. Subsequently, any "tape" bearing one of these titles causes the relevant entry in the title list to be labelled as "present" and the information is stored together with the title and the job number. The first of these "tapes" to arrive is stored after the job description and whenever any piece of information overflows from one 512 word block into another, then the last line of the first block contains the number of the next block, and the first line of the second block contains the job number. When all the component titles have been labelled as "present", then the job title is transferred to the list of complete jobs. Whenever the central computer requires a new job it extracts a title from this list, and all the appropriate blocks from the input well, which are then declared to be empty. If a component title is read before its job description, then the title is added to the title list, and the first action on reading a job description is to scan the title list for any component titles which may already be present.

So far no mention has been made of the system input magnetic tape; the use of this tape is not essential to the system provided that the input well in the core and drum store can be made sufficiently large to assemble complete problems input from slow peripherals and to retain them until they are required by the central computer for execution. However, without the system input tape it is possible for the organisation to become inefficient under certain circumstances. For instance, if computation on one problem continues for a long time, the amount of input supplied by the slow peripherals will exceed the size of any reasonable well; again, if a problem requires a large amount of store for its execution, then the size of the input well has to be reduced.

When the system input tape is in use, the input well within the core and drum store is divided into two parts. One part is filled by slow peripherals, and complete blocks of information are transferred from it to the system input tape. The other part is filled with complete problems by scanning the system input tape, and these problems are then transferred to the central computer as required. If information is being written to the system tape then the length of the system tape which may be scanned without affecting the flow of information from the slow peripherals is directly related to the size of the first part of the input well. For example suppose this part of the input well consisted of five 512 word blocks (out of a total of over two hundred blocks in the core and drum store on Muse), these can be filled by four paper tape readers and one card reader in 10 seconds. The length of the scan is that amount of magnetic tape which can be scanned in 5 seconds, that is 75 blocks. Thus the effective size of the input well is 80 blocks, only 5 of which need be in the core and drum store. These 80 blocks can be filled by the slow peripherals in 2.7 minutes. If this represented 80 programmes in source language, then this well itself is capable of smoothing out an irregularity of 80:1 in computing to input time for an isolated problem. If at any time problems not yet dealt with by the central computer extend further along the system input tape than can be efficiently scanned, then either the input well can be increased, which increases the length of the permitted scan, or, if this is not possible, part of the system input tape is transferred to the system dump tape in order to reduce the length to be scanned. Problems so transferred are treated later at a suitable time chosen by the supervisor programme. The size of the second part of the input well is allowed to change between the widest possible limits, since the rate at which blocks of information can be processed by the central computer varies extensively.

When the system input tape is being used, all information from slow peripherals is stored on this tape, and a permanent record of all input is preserved. Thus it is possible to correct programmes within the computer by reading in a short correction; the amount of off-line editing is therefore correspondingly reduced. The second half of the input well may also be loaded from previous system tapes and from private magnetic tapes, this operation being under the control of the supervisor programme.

The Output System

The output well is divided between the slow peripheral output equipments in the approximate ratio of the rates at which they can transmit information. This division can be varied by the supervisor whenever the output for any slow peripheral exceeds its well size, provided such a change does not affect the operation of the other equipments.

The maximum rate at which the output well can be emptied when all the slow peripheral output equipments are operating simultaneously is 1760 characters per second on Muse, where the output equipment comprises one line printer, one card punch and four paper tape punches. For a problem which produces a large amount of output for one or more peripheral equipments, the output well could be filled very quickly. If the wells cannot be extended, and if the system output tape is not being used, then the programme producing the results must be suspended and possibly replaced by another.

The organisation of the system output tape is similar to that of the system input tape. The output well is divided into two parts, the first part of which is filled by the central computer, and the second of which is filled by the system output tape. The system output tape is partitioned into sections; each section contains one block for each paper tape punch and card punch and ten blocks for each line printer i.e. fifteen blocks on Muse when all slow peripherals are required. A complete section keeps all the slow peripherals operating for approximately 40 seconds. The second part of the output well consists of one or more of these sections. If the slow peripherals are to operate at maximum efficiency, it is necessary that only complete sections should exist in the second output well. These complete sections could be assembled by the supervisor in the first part of the output well, but this would require a very large output well if any problem produced several blocks of output for any particular slow peripheral. This could be overcome by permitting a programme to produce only one block of output, and then suspending it and obeying other programmes until a complete output section is obtained. The organisation of such a system would decrease the efficiency of the central computer due to the necessity of programme switching. The system adopted is to organise the compilation of complete sections along the system output tape.

In this way a programme can produce several blocks of information, partially filling several sections, before it need be suspended. The supervisor organises the allocation of output channels in such a way that only complete sections are transferred to the second part of the output well.

Suppose the system output tape is transferring complete sections to the second output well from the point A, and a programme has produced so much output for a particular slow peripheral that this output is being written into a section of the system output tape at the point B. Then if the system output tape can be scanned from A to B and back whilst the slow peripherals are dealing with the contents of the second well, the programme producing this data need not be suspended. On Muse if the second output well is one section of fifteen blocks, this will occupy the slow peripherals for 40 seconds. During this time, 630 blocks of magnetic tape may be scanned. Thus the maximum separation of the two points A and B is 315 blocks, or 21 sections. Therefore in such a system, a programme can produce 21 blocks of information even for the slowest output before it need be suspended. Ten times this amount of information could be produced for the line printer.

The first part of the output well is not divided into sections, and if necessary the whole of this well can be filled with information for one particular slow peripheral output equipment. The size of the first part of the output well is varied by the supervisor and if at any time more output is produced than can be contained in this well, then this output is written to the dump tape and transferred into the well as soon as possible. Alternatively, the programme supplying this large amount of output can be suspended.

The slow peripheral output equipments cannot operate continuously for more than about 10 to 15 minutes, because they have to be disengaged for loading a new reel of tape or more paper. The reloading operation may take two minutes and it is possible for the system to take account of this situation. At any time the supervisor knows how many blocks of information have been supplied by the central computer to any slow peripheral, and can know when the tape or paper requires changing. It informs the operator of this, and suspends sending information to this output for a reasonable time during which the tape or paper can be renewed. This is easily done by writing incomplete sections on the system output tape. Furthermore, if a section arrives in the second output well and requires a slow peripheral which is disengaged, then the appropriate part of the section is transferred to the first output well, and thence to the system output tape again. If necessary the supply of information for this slow peripheral output from the central computer is suspended. In a similar way, if a slow peripheral output is permanently disengaged and information for it still exists on the system output tape, then this information is recirculated from the second to the first output well and reassigned to one of the slow peripherals which is operating.

The System Dump Tape

The use of this magnetic tape to store excess input and output information has already been described. Also in the above description reference has been made to the suspension of programmes. When this occurs, the programme is transferred to the system dump tape to make room in the core and drum store for a programme which succeeds it. Another occasion when a programme may be dumped is when it computes for so long that either the input well overflows or the output well is emptied. In either case, it is more efficient to proceed to a different problem in an attempt to maintain continuous overlap of input, computing and output.

5.2 A Users Description of the Operating System

Introduction

Muse will be used to deal with a large number of problems of many types, some of which will complete computing in a few seconds. Consequently it is important that the computer operators are relieved of as much work as possible to ensure a smooth flow of work through the machine. To achieve this, "operating instructions" are, whenever possible, given direct to the computer as part of the input rather than to the operator.

The layout of all "Systems Tapes" is the same, sufficient information being recorded to identify each part of a programme or job once the initial address of the programme is known. The location on the system tapes of all slow peripheral input and output, dumps, etc, is printed with the programmer's results. Facilities are provided for the programmer to use information from these magnetic tapes instead of repeating input on slow peripherals, e.g. if a long paper tape is input it may be used again by referring to its location on a system tape.

When large amounts of input or output are involved the programmer may use private magnetic tapes to record the information. This is done by suitable specification in the title.

Whilst most of the operating instructions and parameters are optional (the parameters being set to fixed values if not otherwise specified) each complete programme must contain the following information;

a) a title, preceded by the word JOB as an identifier,
b) the input scheme to be used, preceded by the word COMPILER,
c) the programme itself in source language,
d) an end of tape or last card marker (***Z).

Titles and Headings

Jobs are initiated on the computer by input of information on a slow peripheral equipment. A job may consist of several sections of information, each preceded by an identifying title. This title by which input information is known consists of one line of printing following a heading such as

    COMPILER   (x)
    DATA JOB

where "X" may be

    INTERMEDIATE INPUT
    MERCURY AUTOCODE
    FORTRAN
    ATLAS AUTOCODE

If the heading of the input information is

    COMPILER MERCURY AUTOCODE
    (the title)

This section is based on a paper submitted by Drs. Howarth, Payne and Sumner for publication in “The Computer Journal”.

then the information is a source programme in Mercury Autocode language. More generally, the information obeys the rules of Mercury Autocode, and may therefore include data as well as autocode instructions.

If the heading of the input information is

DATA
(The title)

then the information following is data which is read by a programme during execution, which obeys no rules known to the system.

The heading "compiler" does not itself initiate the appropriate compiling, which is only commenced when a "job" heading is read. If the heading is

JOB
(The title)

then the information following is the job description. In general this information is optional. It is terminated by an end of tape marker, in the case of a separate steering tape, or by "compiler" or "data", in which case the title is not repeated. The "job" heading normally precedes the source programme tape e.g.

JOB
(The title)
COMPILER MERCURY AUTOCODE

followed by the source programme itself.

Further optional information may be included in the job description such as

1. data and programme tapes (input)
2. output equipments used.
3. magnetic tapes
4. store required, computing time.

These sections of the job description are explained below.

<u>Job description - Input</u>

A programme reads in data by means of instructions which are effectively "read next character/string of characters from data tape n" where "n" is a decimal integer. (In this context, "data tapes" are intended to include stacks of cards). The programmer's number of the data tape is specified in the "input" section of the job description. This section begins with the word

INPUT

and is followed by a list of the titles of data tapes used in this job, each preceded by the programmer's number, n e.g.

INPUT
1 (the title of data 1)
2 (the title of data 2)

where there are two data tapes known by the programmer's numbers 1 and 2. These may have been read into the machine on the same input equipment as the "job" tape, either before or after it, or another input equipments. The programmer's number, n = 0, is reserved for the programme itself (and may be used in the programme to read in data which allows the programme as part of the same tape). A separate steering tape might be

```
    JOB
    (the title)
    INPUT
    1 (the title of data 1)
    0 (the title of the programme tape).
```

In this case, the name of the compiler to be used is written at the head of the programme tape. When the "job" heading is on the beginning of a data tape, the "input" section of the job description must include.

    SELF = (n)

where "n" is the programmer's number by which this data is known within the programme.

If the input section of the job description is omitted, it is taken as if

```
    INPUT
    SELF = 0
```

were included, and the programme following is compiled and executed.

Since all input is automatically copied to the system input magnetic tape, a programmer may read his tape in again direct from this input tape (e.g. to make a correction). To do this, in the "input" section of his job description he writes

```
    TAPE (a)/(b)/(c)
    (n) (the title of his input)
```

where "a" is the system tape "number"
"b" is the number of the 512 word block of tape, and
"c" is the line within the tape block where his input starts.

His title is, of course, written on the tape at this point, but the title is specified again as a check.

Job description - Output

A programme outputs data by means of instructions which are effectively "print" next character/or string of characters on output n" where "n" is a decimal integer. The output equipments are specified in the "output" section of the job description. This section begins with the word

    OUTPUT

and is followed by a list of the output mechanisms used in this job, each preceded by the programmer's number, n e.g.

```
    OUTPUT
    1 (the type of equipment for output 1) (m) BLOCKS
    2 (the type of equipment for output 2) (m) BLOCKS
```

The type of equipment may be

```
    LINE PRINTER
    TELETYPE
    CARDS
    FIVE-HOLE TELETYPE
    ANY
```

Where
"teletype" means a 7-hole (teletype) paper tape punch,
"cards" means a card punch

"any" means output on. A line printer, teletype punch or cards.

The operator can control which equipments are used most by disengaging the other output equipments. "m" defines the limit of the output, and if the output exceeds m blocks of 4096 characters the programme is stopped. If the number of blocks of output is not specified, it is taken as "1 block". Further, if there is only one output used, the output section may be omitted, and this is taken as if

OUTPUT
0 ANY 1 BLOCK

were included in the job description.

When printed, the output information itself is preceded by

OUTPUT (n)
(the title of the job)

and output of system information is always on output 0.

Job Description - Tapes

If a programmer uses magnetic tapes directly in his programme (by use of tape instructions as distinct from using tapes in connection with input or output) then he specified each tape used by two lines in the job description

TAPE
(n) (the title which is stored on block 0 of the tape)

where "n" is the programmer's number of the tape. When a new tape is required, the appropriate two lines of the job heading are

TAPE FREE
(n) (the title to be written on block 0)

If a file extends over several tapes, this is specified by a modified "tape" heading

TAPE/ (m)
(n) (the title of block 0)

where "m" is the number of the continuation counting from 1 upwards. The programmers number n is the same for all m. The final tape of this file has the heading

TAPE / (m) END

If a programme involves extensive input, then the job is preceded by copying this input to private magnetic tape. To initiate this copying process; the input is headed,

COPY TAPE FREE
(the title to be written on block 0)

If a previously used tape is employed, the heading is

COPY TAPE (b)
(the title on block 0)

where "b" is the number of the tape block (the programmer must always begin at the beginning of a tape block).

Information may be read from this tape subsequently by specification of the tape and title of the information in the "input" section of the job description.

If a programme involves extensive output then the output can be written on a private magnetic tape. This is specified in the "output" section of the job description as follows:-

OUTPUT

```
    (n) TAPE FREE/ (the type of equipment) (m) BLOCKS
    (the title on block 0)
```
where "n", "the type of equipment" and "n" are as for direct output and where the title specified is written on block O. If a previously used tape is employed, the specification is
```
    (n) TAPE (b) / (the type of equipment) (m) BLOCKS
    (the title of block 0)
```
where "b" is the number of the to block.
This private tape is printed by a steering tape consisting of
```
    PRINT TAPE
    (the title of block 0)
```
if the whole tape is to be printed, or
```
    PRINT TAPE (a)  (b)  (c)
    (the title of his output)
```
if one section of tape only is to be printed, from tape a, block b, word c.

<u>Job description - Miscellaneous</u>

Further information may be given in the job description to indicate

1. the amount of core and drum store used
2. the time the programme is expected to compute for
3. the number of drums the programme requires for programmed drum transfers

All three apply to the execution stage of the programme, i.e., excluding input from slow peripherals, compiling and output to slow peripherals, They are specified by
```
    STORE      s
    COMPUTING  p.q HOURS
or  COMPUTING  p.q. MINUTES
or  COMPUTING  p.q SECONDS
    DRUM       d
```
where "s" is the maximum number of core and drum 512 word blocks of store in use within the programme during the execution stage, "p.q' is a fixed decimal number such as
```
    COMPUTING 7.5 SECONDS
```
where the programme is expected to run for not more than 7½ seconds, (if the estimate for store used and computing time is exceeded the programme is stopped), and where "d" is the number of drums the programme requires to reserve for programmed drum transfers.

If the total execution time is significantly different from the actual computing time, because there is considerable tape waiting time, the actual computing time should also be specified, e.g.,
```
    EXECUTION 5 MINUTES
    COMPUTING 30 SECONDS
```
If information is not supplied in the job description, then
```
    20 store blocks (10,240 words)
     4 seconds computing time
```
and, of course, 0 drums are reserved. Estimates of the computing and execution times are taken as being equal unless both are specified explicitly.

End of Tape Markers

The end of a section of tape is indicated by

***(x)

where "x" is Z, A, B, C or T.

The marker

***Z indicates the genuine end of the tape/stack or cards
***A indicates abandon previous incomplete section if any (this may be required by a machine operator)
***B indicates that a binary tape follows
***C indicates the end of a section, and that there is another section following on the same tape
***T indicates a temporary stop within a section,

On reading the " ***Z" the peripheral equipment is disengaged by the computer. When the operator next engages this equipment, a new section (with the appropriate heading and title) is read. The marker "*** C" indicates the end of a section of tape but the equipment is not disengaged and the next section is automatically read.

On reading the marker " ***T" for a temporary stop, the equipment is disengaged as for "***Z". However, when the operator next engages this equipment, a continuation of the current section (without a new heading) is read,

On reading the marker "***B", the computer reads the information following in binary without testing for further end of tape marker. The number of characters, n on a binary tape may be indicated by

(n)***B

where "n" is a decimal number.

A better method of specifying the continuation of a section of data, without use of the marker" ***T", is by means of a modified "data" heading

DATA/ (n)

where "n" is the number of the continuation of the section of data, e.g. for a programme with data on two distinct paper tapes, the data may be headed

DATA / 1

(the title of the data)

and

DATA / 2 END

(the same title)

and each tape ends with the marker "***Z". The continuation data tapes may be read into the computer in any order.

Section 6 Input Languages

6.1 The Standard Compiler Routine

Introduction

This assembly programme for Atlas may be thought of as an autocode to write autocodes. More precisely it is a system which enables the user to define, by means of _statement formats_ and _phrase definitions_, the form of the statements (and their constituent expressions) which are to be used in the source language. The 'meaning' of these statements, or the method of translating them into object coding, is described by means of _statement definitions_. The system is self expanding and allows the user to define the 'meaning' of new statements in terms of previously defined statements as well as in terms of _basic statements_ (whose 'meaning' is built-in). It also permits recursive definitions of both the form and meaning of statements. This is particularly useful, for example, in dealing with algebraic expressions involving parentheses to any depth.

It is unlikely that every user will want to write his own autocode, and standard definitions will be available for the more common autocodes. For example, it is hoped that definitions of the following languages will be available:

1) Fortran
2) Algol
3) Mercury Autocode
4) Pegasus Autocode
5) Nebula (the Orion business language)

The user may then use any of these autocodes as they stand, or he may add further definitions to extend the autocode language into his particular problem field. If, however, the user elects to use a source language of his own design then the form and 'meaning' of every statement in this language has to be defined. Although the basic statements are available for defining the 'meaning' of other statements they cannot be written in the source language. The mechanism for defining an autocode is described below with reference to Mercury Autocode.

THE PHRASE DEFINITIONS

These are used to build up classes of logically similar phrases. To each class is assigned a name, the class identifier, which may then be used in further phrase definitions and statement formats to indicate that any phrase of the class in question may be substituted in its place. Class identifiers are represented by a string of characters enclosed in square brackets (e.g. INDEX might be assigned to the class of index letters). The following, therefore, would be the start of a formal definition of Mercury Autocode:

```
phrase defn. [v] = a, b, c, a, e, f, g, h, u, v, w, x, y, z, π
phrase defn. [v] = a', b', c', d', e', f', g', h', u', v', w', x', y', z'
phrase defn. [INDEX] = i, j, h, l, m, n, o, p, q, r, s, t
phrase defn. [VARIABLE] = [V],[V'],[V][N],[v][INDEX],[V]([INDEX] + [N])
```

The class [N] denotes an integer and it is a built-in class which does not require further definition as is the class [K] denoting a floating point constant.

Two features which are very often present in a phrase or statement are 'repetitive appearance' and 'optional appearance' of some item. In order to describe these two situations the qualifiers * and ? may be used and the relevant formal definitions will be constructed behind the scenes. Thus:

```
    [A][B*][C] means [A][B][C] or [A][B][B][C] or [A][B][B][B][C] etc
and [A][B?][C] means [A][B][C] or [A][C]
and [A] [B*?][C] means [A][C] or [A][B][C] or [A][B][B][C] etc
```

where only [A], [B] and [C] require formal definitions.

The general arithmetic expression in Mercury Autocode could now be defined as follows

```
phrase defn: [Q] = [VARIABLE], [K], [INDEX]
phrase defn: [TERM] = [Q*][Q?]
phrase defn: [|Q] = /[Q]
phrase defn: [±] = +, -
phrase defn: [±TERM] =  [±][TERM]
phrase defn: [GENERAL EXPRESSION] = [±?][TERM][ ±TERM?*]
```

STATEMENT FORMATS

These are similar to the phrase definitions except that they describe one phrase only, namely the form of a statement. It is convenient sometimes to introduce statements which are not a feature of the source language, but which are to be used only in defining the meaning of other statements. This type of statement is distinguished by writing the word auxiliary before it. Thus the form of the class of arithmetic instructions in Mercury Autocode is defined as follows;

```
statement format: [VARIABLE] = [GENERAL EXPRESSION]
```

In defining the 'meaning' of this class of instruction (see later) several auxiliary statements are required including the following

```
statement format (auxiliary) : acc = [GENERAL EXPRESSION]
statement format (auxiliary) : [VARIABLE] = acc
statement format (auxiliary) : acc = [±?][TERM]
statement format (auxiliary) : acc = acc [±][TERM]
```

STATEMENT DEFINITIONS

To each statement format corresponds a statement definition. The former describes the form of a statement, and the latter describes the action which is to be taken when the form is encountered in a source programme. In most cases the action to be taken is to assemble the equivalent set of instructions in the target programme, but in the case of declarative statements such as the "[V][N]" of Mercury Autocode the equivalent action is to enter certain information in lists to be used by other statement definitions. Statements can of course be partly imperative and partly declarative, and both operations in fact are effected by means of the list compiling instructions (a type of basic statement) the only difference being that in the first case the 'list' in question is the object programme.

Very often the 'meaning' of a statement, in the above sense, can be expressed in terms of a sequence of other less complex statements, where the sub-expressions of the main statement are the <u>parameters</u> of the <u>sub-statements</u>. It is necessary, therefore, to have some means of resolving a statement (and in general an expression or phrase) into sub-expressions consistent with its known structure, and if necessary to build up new expressions from these sub-expressions. It is also desirable to be able to compare different expressions in order to select distinct courses of action. For these reasons a second type of <u>basic statement</u> is available, namely the <u>expression handling operations</u>. Also, for the purpose of 'control branching', a floating address system is employed, and any sub-statement or basic statement to which 'control' is to be passed out of sequence can be labelled;

for example,

10] →3 unless [VARIABLE] = [V][I]

is a basic statement of the expression handling type with label 10.

A statement definition therefore may consist of the three types of 'instruction'

1)	substatements		defined by the user
2)	list compiling instructions	)	basic statements built
3)	expression handling operations	)	into the system

Before illustrating the use of these by examples the list compiling instructions require some further explanation.

<u>THE LIST COMPILING INSTRUCTIONS</u>

Associated with these instructions is a <u>central</u> group of 24-bit registers denoted by α_1 , α_2, α_3 ... which do not form a field (i.e. cannot be referred to as α_r). In addition to these there is a further set of <u>local</u> 24-bit registers β_1, β_2, β_3,.. associated with each statement definition. The list compiling instructions are concerned with selecting, processing, and comparing the information in these registers and in the registers whose addresses are contained in them.

Thus e.g.

$$\alpha_{10} = \beta_2 + (\alpha_1 + 3)$$

means set register α_{10} equal to the contents of register β_2 plus the contents of the register whose address is given by α_1 + 3; and

$$\to 1 \text{ if } (\beta_1) > \alpha_3 + 2$$

is typical of the testing variety and means 'jump' to the instruction labelled 1 if the number in the register whose address is in β_1 is greater than α_3 + 2.

Now in general if α_1 (say) is the address of the first item of a conventional list (i.e. one in which consecutive items lie in consecutively addressed registers) the (α_1 + n) is the (n + 1)th item in that list. In addition to the conventional list, however, the <u>chain</u> list is also of frequent use. In this list each 24-bit word containing an item is accompanied (in the next register) by a further word which contains the address of the next item (i.e. word pair).

Generally the address of the first word of such a list is recorded in the link word of the last item thus making the list circular. The advantage of the chain list is that it is easy to manipulate, e.g. to insert and delete items simply means inserting or detaching a link. However given α_1 (say) as the first item of such a list the n+1 th item is not now (α_1 + n). Instead the address of the second item in fact is given by B_1 (say) = (α_1 + 1), and the third by β_1 = (β_1 + 1), and so on. As a shorthand way of referring to various items in a chain the symbol $\oplus$ is used. The n+1 th item is thus denoted by (α_1 $\oplus$ n) and this item can be transferred to β_5 (say) by the instruction,

$$\beta_5 = (\alpha_1 \oplus n)$$

Thus $\alpha_1 \oplus 1$ is equivalent to (α_1 + 1) if α_1 is an address in a chain. The use of the list compiling instructions and of the expression handling operations and sub-statements can be illustrated by the following examples of statement definitions:

```
statement defn: [VARIABLE] = [GENERAL EXPRESSION]
             acc =[GENERAL EXPRESION]
             [VARIABLE] = acc
             END
```

The first item in every statement definition is the <u>statement heading</u>. This serves the dual purpose of relating the definition it appears in to a particular form of statement, and also of 'locating' sub-expressions (in the above case [VARIABLE] and [GENERAL EXPRESSION]) within the statement which can be written as the parameter of the 'instruction' which follows. In the above example the 'instructions' which follow are simply <u>sub-statements</u>, but in general the parameters which appear in a statement heading cannot be taken over directly by the sub-statements, and they must both be resolved into more elementary expressions by means of 'instructions' of the parameter handling class.

```
statement defn: acc = [GENERAL EXPRESSION]
             let [GENERAL EPRESSION] = [±?] [TERM][±TERM *?]
             acc =[±?][TERM]
             ->1 unless [±TERM*?] = [±TERM*]
             3] ->2 unless [±TERM*] = [±TERM][ ±TERM*]
             let [±TERM] = [±][TERM]
             acc = acc [±][TERM]
                3
             2) let [±TERM] = [±][TERM]
             acc = acc [±][TERM]
             1] END
```

In this case only one parameter appears in the statement heading so obviously if the statement is to be defined in terms of more simple statements this parametric expression must be resolved into its sub-expressions. The first 'instruction' of the definition does this, and the expressions appearing on the right hand side of the equality can then be referred to. The next 'instruction' is a sub-statement which is not defined here but which is intended to set the accumulator equal to the first (possibly signed) term of the [GENERAL EXPRESSION]. Next the nature of the [±TERM*], has to be determined as this may be either [±TERM*] or "nil". In this latter case control is passed to the end by the conditional parameter resolving instruction. If control passes sequentially to the next 'instruction' the [GENERAL EXPRESSION] must involve a sub-expression [±TERM*]. Now in order to resolve this into more elementary form it is necessary to know how the '*' classes are defined inside the machine. In general [IDENTIIFIER*] is defined recursively as [IDENTIFIER][IDENTIFIER*], [IDENTIFIER]. The recursive structure of [±TERM*] is thus expanded by a cycle of instructions which deal with each [±TERM] in the sequence until the last which is dealt with at the point labelled 2. As in normal programming it can be seen that the same name, in this case [±TERM*] can be dynamically assigned to a sequence of different expressions.

Although the above examples do not contain basic listing instructions it is obvious that the definitions of some of the sub-statements used will eventually lead to sequences of basic listing instructions which will compile an object programme. Some of these lower level statement definitions will require a knowledge of the variable directives (i.e. statements of the type [V]->[N] which have gone before. One way this information might be made available is for the statement definition associated with [V]->[N] to record the [N] associated with each of the [V] letters in a particular position in a conventional list of 15 registers separate from the object programme. The obvious way of associating the letters of [V] with positions in the list is to use the same ordering as in the phrase definition of [V] i.e. a in position 1, b in position 2, etc. This however, requires that a mechanism is provided to determine which alternative within a phrase definition a particular expression represents. The built-in instruction

[αβ] = category of ["any phrase identifier”]

is provided for this purpose. Thus in the example below, whenever the [V] in question is a ‘d' (say) then β_i will be set to 4. It is assumed below that a_3 has been reserved for the address of the directive list and that it will not be altered by other statement definitions, but they may, of course, refer to it.

```
statement defn: [V]->[N]
             β1 = category of [V]
             β1 = β1 - 1
             (α3 + β1) = [N]
             END
```

Conclusion

Obviously the brief description given above does not describe the system fully and the reader is referred to the papers listed below for a more complete description. It does however illustrate the relative ease with which compilers for particular programming languages can be assembled.

References

1) An assembly Programme for a Phrase Structure Language, Computer Journal, October 1960

2) An Assembly Programme for a Phrase Structure Language (Concluded), Computer Journal, January 1961

3) A Description of Mercury Autocode in terms of a Phrase Structure Language, Annual Review in Automatic Programming, 1961.

6.2 Mercury Autocode for the Manchester University Atlas

This version of Mercury Autocode consists mainly of the facilities described in chapters 1,2,3 and 4 of the Mercury Autocode Manual (Ferranti list CS24A) plus all those (except 'short integers') described in Appendix 4. Although slightly more general forms of some instructions will be accepted this is because it has proved more convenient to write the compiler this way and not because it was considered worthwhile to generalize Mercury Autocode. In fact the Mercury Autocode compiler is provided mainly so that programs currently being developed on Mercury can later be run on Atlas. However, facilities will be provided for Atlas machine orders to be used to replace Mercury machine orders, and for programs to exploit the larger store of Atlas. The form and meaning of all the permitted orders can be discovered from the formal description of the compiler. Below the principal additions to the language are described.

Storage Allocation

The following directives MAIN

```
        MAIN → [N]
        AUXILIARY ([± ?][N][,][±?][N])
        DEPTH [N]
        DUMPS [N]
```

can be written at the head of a program (i.e before the first chapter) to indicate the amount of storage required. Thus if a program started

```
        MAIN -> 1000
        AUXILIARY   (-1400, 20000)
        DEPTH 5
        DUMPS 4
```

then 1000 main variables would be available (for use in the usual way e.g. A->1500, B->99 etc);21401 auxiliary variables numbered from -1400 to 20000 would also be available; there would be 5 levels of sub-chapters; and 4 dumps would be allowed for use by PRESERVE and RESTORE instructions. If any of the above declarations are omitted, the corresponding one of the following will be assumed

```
        MAIN -> 400
        AUXILIARY   (0, 10751)
        DEPTH 2
        DUMPS 2
```

Query printing

Query printing can be controlled in two ways. If a directive

```
        IGNORE QUERIES
```

appears in a program the queries will be ignored (as on Mercury with key 4 down) throughout the rest of the program, unless a further directive

```
        COMPILE QUERIES
```

appears. This second directive is equivalent to putting key 4 up on Mercury and the query printing will be compiled throughout the rest of the program or until

```
        IGNORE QUERIES
```

appears again. However, even if some queries are compiled, the printing can still be suppressed by information written in the

'Job Description'. This sets a switch and causes control to skip around the print instructions.

Input/Output

In order to permit input and output instructions to operate on several channels the instructions

```
        SELECT INPUT[I or N]
        SELECT OUTPUT [I or N]
```

are provided (where [I or N] may be replaced by an index or an integer).

These instructions have dynamic significance. That is they influence all the input (or output) instructions which are subsequently executed, until they are over-ridden by further select instructions. On entry to a program channel 1 will be selected for both input and output. The query printing and dynamic fault monitoring will also appear on the currently selected channel but fault monitoring done at compiler time will be confined to channel 0.
Since some existing Mercury Autocode programs involve output of alpha-numeric information, the following Mercury machine orders will be simulated assuming the usual 5-hole tape code:

```
        620,[N]
        630, ([ INDEX])
```

(The order 592,0 used in the integration of differential equations will, of course, also be simulated).

Machine Orders and Constants

The following are allowed

```
[FD][,][I or N][,][[I or N][,?][ADDRESS]
H[ADDRESS][,][SOURCE LABEL?][ADDRESS]
[+][K]
```

The first is the general form of a machine order. Its function digits may be written in the usual way, e.g. 121, 0121, 1400. The B-digits can be written explicitly e.g.

```
0121, 127, 127, 1   (skip 1 instruction)
```

or an index can be used to denote the B-line associated with that index e.g.

```
0121, I, I, 1       (I =I + 1)
```

An address part may be symbolic or absolute thus:

```
[ADDRESS]=([SYMBOLIC ADDRESS]),A[N],[±?][N].[0-7],[+?][N],J[0-7*]
[SYMBOLIC ADDRESS]=[V-LETTER][N],[V-LETTER'],[V-LETTER]
```

If a symbolic address (i.e. the name of a floating point variable) is used its address will be substituted. For example

```
        0324, 0, 0, (C10)
```

would transfer the variable C_{10} to the accumulator. The remaining alternatives are consistent with the ABL notation Thus

```
        0234, 127, 0, A10,    jumps to label 10 if ACC = 0
        0121,  20, 0, 8.7     sets B20 = 0....01000111
        0121,  20, 0, 10      sets B20 = 0....01010000
        0121,  20, 0, J71     sets B20 = 1110010.....0
```

The second of the above formats, namely:

```
        H [ADDRESS][,][SOURCE LABEL?][ADDRESS]
```

is a means for writing two integers into a long line. Here [ADDRESS]'s are interpreted as in an instruction.

The third format allows floating point constants to be written into the program. All three of the above can be labelled in the usual way. Also the second of a pair of integers can be labelled, as shown.

Additional Instructions

If any user requires to run a program which contains instructions not generally accepted by this version of Mercury Autocode he must first define them in the language of the Compiler Compiler. For example, in the case of the Mercury machine order 400 (A_{10}) the following would suffice

```
        FORMAT      [SS]= 400(A10)[NEWLINE]
        ROUTINE     [SS]≡ 400(A10)[NEWLINE]
        *0324  [,]0[,]0[,](a10)[NEWLINE]
        END
```

This could of course be generalized to deal with 400 ([V-LETTER][N] (say).

Monitoring of Information

As on Mercury many programming faults can be detected during translation. When such a fault is encountered, information describing it will be monitored on Channel 0 and the translation continued so that all faults of this nature may be found in one run. To assist in the interpretation of this information, the number of every new chapter and sub-program is also monitored. (When any of these faults occur a digit in the switch B9 is set to prevent the program being obeyed after compilation). The PSA directive will print the number of locations filled by instructions up to that point.

An example of the style of this printing is

```
START OF CHAPTER 1
LABEL 3 SET TWICE
START OF P-1
START OF CHAPTER 1
LABEL 10 NOT SET
CHAPTER CONTAINS 500 INSTRUCTIONS UP TO PSA
START OF CHAPTER 2
TOO FEW REPEATS
START OF CHAPTER 0
CHAPTER ENTRY LABEL 1/1 - 1 NOT SET
```

Dynamic faults (such as extracting the square root of a negative argument) will also be detected, the monitoring taking place on the current channel. Control is then transferred to label 100, or, if this is not set, the program is terminated. To further aid fault-finding, the directive TRACE is introduced. This causes label numbers to be monitored every time a labelled instruction is obeyed. Similar information is monitored when ACROSS, DOWN and UP instructions are obeyed, thus dividing the monitored information into larger units. The TRACE directive only influences the program which follows it and can be inhibited by means of the directive.

```
        STOP TRACE
```

The style of printing by the TRACE facility will be as follows:-

```
L1          )
            )
            )  for labels
            )
L13         )

C2/4        )
            )
            )  for both ACROSS and DOWN
            )
C1/1-501    )

UP             for UP
```

<u>The use of B-Lines</u>

At run time, B-Lines 1 to 12 are used for the indices I to T. B-Lines 13 to 24 (see section on closed subroutines) and 33 to 42 are used by the format routines behind the scenes, while B25 to B32 have special roles as described below. B43 to B90 are available to the user for his own purposes.

Summary of the Roles of B-Lines at Run Time

B_1	)	
	)	
	)	Indices I - T
	)	
B_{12}	)	
B_{13}	)	
	)	
	)	Indices I – T (used by format routines)
	)	
B_{24}	)	
B_{25}		miscellaneous working space for indices
B_{26}		depth counter (initially 0)
B_{27}		address of complex accumulator
B_{28}		counter used by preserve and restore
B_{29}		link
B_{30}	)	= if queries are to be skipped
	)	
	)	= 0 otherwise
B_{31}		current label directory
B_{32}		address of matrix dump
B_{33}	)	
	)	
	)	working space of closed subroutines
	)	
B_{42}	)	
B_{43}	)	
	)	
	)	Available to the user
	)	
B_{90}	)	

6.3. Intermediate Input

This input scheme exists in several slightly different forms of which the four main ones are,
(I) a language within the Supervisor. Input may be by either five or seven channel paper tape or punched cards.
(II) A means of reading the Supervisor and associated routines into the store during the development of these routines. Input is by seven-channel paper tape.
(III) a means of reading seven channel paper tape into Atlas. This routine is obeyed in the Subsidiary Store and can be used to read programmes into either the first 300 words of the Subsidiary Store or into the main core store. It is intended mainly for use in the development of program during the commissioning of a machine when only the Subsidiary Store is available and consequently has less facilities than the other versions.
(IV) a means of reading seven channel paper tape into Mercury for conversion into five channel tape which is subsequently read into Atlas using the Octal Input routine in the fixed store.

The routine described below is the first of those listed above and the differences between this and the other three are given at the end of this section.

I) Intermediate Input under Supervisor Control

A forty-eight bit Atlas word can be interpreted in three different ways.

(1) as an instruction
(2) as two twenty-four bit half words
(3) as a floating point number

Intermediate Input provides "formats" to deal with these three interpretations of the Atlas word and it also has formats which enable the user to

(a) set preset parameters,
(b) write his program as a number of routines.
(c) enter the program

All these formats are terminated by either "End of record", "|" (vertical line) or " [" (open square brackets).

1) Instruction format

An instruction is punched

F, Ba, Bm, S

where F consists of one binary and three octal digits. The binary digit may be omitted if it is zero.
Ba and Bm are decimal numbers in the range 0-127 and consist of one, two or three decimal digits.
S is the address and occupies the second 24 bits of the word.
The "commas" are used as separators.
The address may be either decimal, octal, floating or general i.e. a combination of the other three

A decimal address may be written
(a) as a sign (optional if positive) followed by a series of decimal digits, e.g. 31913 or +976 or -7928.
In this case the binary point of the number in the computer is assumed to be three binary digits (or one octal digit) from the least significant end.

(b) as (a) above but also followed by a point and one octal digit e.g. 976.7 or -325,1. In this case the octal digit goes into the three least significant bits of the word.
(c) as (a) above but also with the symbol "n" or "N". This symbol may be punched anywhere in the address but it is recommended that it is punched either immediately before the first or immediately after the last decimal digit. e.g. n736, -n219, -514n, N99. In this case the number in the machine is stored with the binary point at the least significant end. The effect of the "n" is thus to shift a number down one octal place (only one such shift is allowed).

An <u>octal address</u> is punched
(d) as an asterisk (*) followed by up to eight octal digits e.g. *7001, *40050014 The first digit punched is shifted to the most significant octal digit of the 24 bit word, the second digit to the next octal position, etc. Zeros at the right hand end of the word may be omitted if desired.

A <u>floating address</u> (or <u>label</u>) is punched as
(e) either a series of decimal digits enclosed in brackets e.g (37) or as two sets of decimal digits separated by a "/" and enclosed in brackets e.g. (4/5).
The floating address (0) has a special meaning as it is taken to be the address of the current instruction when it is in an instruction format.
The largest numbered floating address allowed in a routine is 100.

A <u>general address</u> is punched either
(f) as an octal address (d) preceded by a decimal address (a), (b) or (c) e.g.

```
     +80.6 * 216
    -n27 * 53
```

The final result is the sum of the two parts and the octal representation of the above two examples is 21601206 and 52777745 respectively or

(g)as a string of labels (e) which may have (a), (b) , (c), (d), or (f) preceding or terminating the string or between any members of the string e.g. -2.4 (4) 6* 412 (3) -3n.
The final address is the sum of the constituent parts.
A particular example of this type when used in an instruction format is the relative address e.g.
1 (0) is the address of the instruction immediately following the current address
2 (a) is the one after that, etc.
-1(0) is the address of the instruction immediately preceding the current address.

The following are examples of Instruction Formats.

```
  121, 1, 0, 1          [set B1 = 1
 0121, 127, 0, (2)      [set B127 (control)
                        [to the value of
                        [floating address 2
 224, 127, 0, (4/2)     [if Bt = 0 transfer control to
                        [label 4 of routine 2
```

2) Half Word Format

Anything which can appear as an address can be put in either of the two half words. The two addresses are separated by a "/" e.g.

```
                   0/n7
                   l/-n1
                   -2/4 (0)
                   2 * 7 (2) -4/(3/1)
```

When the relative address (0) is used in the half-word format it is taken to be the address of the half-word in which it appears.

3) Floating Point Number Format

A Floating Point Number is punched as an argument followed by an optional exponent. The argument must be signed and may be fractional e.g.

```
                   + 123
                   - 123.456
```

If an exponent is present it is separated from the argument by a comma and it must be an integer which may be signed. e.g.

```
                   + 123, -2
                   - 123.456,2
                   + 123, +4
```

The setting of floating address

A floating address may be set either by labelling oneof the preceding formats or by means of a directive (see next section). When an address is set by labelling a format then the value of the label is set to the address of the register in which the format is to be planted, e.g. suppose that the following formats are being planted in register 4 onwards

```
        1)     324, 0, 1, 10
        2)     4/3) *7
        4)     +123
               2(0) / -1(0)
```

Then label 1 will be set equal to 4
label 2 will be set equal to 5
label 3 will be set equal to 5.4
and label 4 will be set equal to 6

The value of the two half words 7 and 7.4 will be 9 and 6.4 respectively.

Directives

The formats concerned with setting preset parameters, setting routine numbers and entering the programme are called Directives. The directives are:-

1) A floating address may be set by means of the directive
   (floating address) = general address

```
    e.g  (1)    =  4    | floating address 1 = 4
         (4/2)  =  (4)  | floating address 4 of routine 2 =
                        | floating address 4 of current routine
         (0)    =  10   | plant the next instruction in
                        | register 10
```

2) "R" followed by a decimal number, is the directive which marks the beginning of a routine, e.g.

```
                   R2
```

3) "E" followed by a general address, is the directive to enter the programme, e.g.

```
E 4          | enters at register 4
E(2/3)       | enters at label 2 of routine3
```

The programme is entered on Main control at the address specified.

<u>Faults</u>

The following faults are detected by the input programme and an indication given when they are detected.

1) <u>Illegal Character</u>
   A character in the format which is not an allowable character of the set, e.g. the letter "X" appearing in an instruction, or one of the decimal digits 8 or 9 appearing in an Octal address.
   The number of characters on a line (or before any "[" or "|") must not exceed 40 "useful" characters (Space, Tab., Erase, U.C. and L.C. are not "useful" characters).
2) <u>Wrong Format</u>
   An allowable character of the set occurring in the wrong place, e.g. a comma following a comma.
3) <u>Label set twice</u>
4) <u>Label not set</u>
5) <u>Too many forward references</u>
   The number of forward references to labels which have not been set is restricted to 256.
   Examples of fault print-out are,

```
F1
10
121 X, 4, 3, 0
```

   The format which would have gone into location 10 has an illegal character in it, i.e. "X"

```
F3
12
3) 101, 10, 4, 0
```

   The instruction which should have been read into register 12 refers to a label i.e. 3, which has already been set.

```
F4
4 / 5
```

   Label 4 of routine 5 has not been set.

If a fault is detected the incorrect format is ignored after a fault printout has been given i.e. no attempt is made to interpret and copy the format to the store. The remainder of the programme is however read but any Enter directive is not obeyed. It is thus possible to detect all punching faults on a tape by one pass through the machine.

<u>Ignored Characters</u>

1) All characters after a "|" or a "[" are ignored until the next "End of Record". This enables comments either after a format or on a line by themselves to be ignored,
2) In a format the characters "space", "backspace", "paper throw" "tabulate" and "erase" are ignored.
3) "End of Record"s are ignored unless they occur on the same line as a format. If they do they are used to terminate the format.

II Intermediate Input for use in Developing the Supervisor

This version is primarily for use in reading the Supervisor into the machine. It operates in the Subsidiary Store on Interrupt control and reads seven-channel paper tape. The programme is read into the main store and, following an Enter directive, is entered on Interrupt control. As the Supervisor is not in the machine it is necessary to provide sub-routines for reading in characters and punching out fault information.

Sub-routines

The following five sub-routines are available. They are obeyed on Interrupt control and all, except the last, must have the return address set in B90 before entry.

a) read one character

This uses TR5 number 0 and is entered at address 991*7 (label (71)). On exit B89 is set as follows

Digit 9	= 1 if 5-channel tape is being read
	= 0 if 7-channel tape is being read
digits 8–2	contain the information digits 32 - 26 of the V-store line for this reader i.e. the character is read in as it appears on the tape and is not converted to internal code.

The remaining digits of B89 are zero. B88 is used as temporary working space.

b) punch one character

This uses the Teletype punch number 0 and is entered at address 1002*7 (label (72)). On entry, B89 should contain the character to be punched, in Flexowriter code, in digits 9 – 3. The remaining digits of B89 are irrelevant. The character in B89 is not retained. B88 is used as temporary working space.

c) octal output

This punches out on Teletype 0 the contents of the address specified. Each 24 bit half-word is punched as 8 octal digits using the Flexowriter code. An Upper Case character is punched first followed by either New Line or Space depending on whether the first half-word address specified is the more or less significant half of a word. Subsequent half-words are similarly punched preceded by either a New Line or Space character.

The routine is entered at address 947*7 (label (73)). Initially B81 contains the address in digits 23 - 2 of the first half-word to be punched and B82 contains the address in digits 23 - 2 of the last half-word to be punched. Digits 1 and 0 must be zero in both B-lines.

B81, B85 - B90 and Bt are used as working space.

d) punch B lines

This punches on Teletype 0 the contents of the B-lines specified. An Upper Case character is punched initially. The contents of each B line are punched in octal in Flexowriter code preceded by a New Line character.

The routine is entered at address 973*7 (label (74)). Initially

digits 8-2 of B121 contain the address of the first B line,
digits 8-2 of B83 contain: the address of the final B line,
the remaining digits of B83 must be zero.

B81, B85 - B90, B 121 and Bt are used as working space. The B lines punched (except those used as working space) are left unchanged.

Address 986*7 (label (80)) onwards contains the decimal digits of 0 - 9 in Flexowriter code in bits 9 - 3 of consecutive half-words i.e. 32/49 etc.

Address 895*7 (label(81)) is the address of two consecutive half-words used as working space.

e) end programme

This sets B120, display, to *70700000 and enters a loop stop in *70017660.

It is entered at address 1013*7 (label (75)).

Interrupt Loop Stops

If an interrupt occurs the machine enters a loop stop as follows, where the first one is the lowest priority interrupt.

Address	*70017670	for a digit set in line 1 of the central computer V-store
	*70017700	non equivalence
	*70017710	P3
	*70017720	P2
	*70017730	P1
	*70017740	T
	*70017750	PO
	*70017760	Parity
	*70017770	"Winking" L.A.M.

The final loop stop is entered when an interrupt occurs but none of the digits in line 2 of the central computer V-store is set on examination by the programme.

After an Entry directive is read but before it is obeyed, lines 0, 1 and 2 of the Central Computer V-store are reset to zero.

Title Directive

After a title directive T, which must be punched on a new line, is read all characters (except UC and LC) until the next New Line character are ignored. (A title directive can thus be punched T, TITLE, Title etc.). The next line is copied to the output punch, preceded and followed by "New Line".

Wait Directive

After a wait directive W, which must be punched on a new line, all characters, except UC and LC, are ignored until the next New Line character. A wait directive disengages the tape reader; when the Engage button is pressed input is continued and the first character read is ignored.

Tape Character with Even Parity

If a character with an even number of holes punched is read from the tape fault 1 is indicated.

Fault Indications

These are as in Version I except that the address of the wrong format is punched as 8 octal digits instead of in decimal form. Also the label and routine numbers for Fault 4 are punched as octal numbers.

Illegal and Ignored Characters

These are as in version I except that W and T have significance as described above.

The first character read on any programme tape is ignored as is the first character following a "New Line" after a W directive

III Intermediate Input to read Programmes into the Subsidiary Store

This version is mainly for use in the commissioning of a machine. It is similar to Version II except that

a) programmes can be read into only the first 300 words of the Subsidiary Store, the remainder of the store being occupied by Intermediate Input.
b) routines and cross references to routines are not allowed
c) no floating point numbers are allowed.

IV Mercury Version of Intermediate Input

This is another version mainly for use during the commissioning of a machine. The programme is punched on seven-channel paper tape mainly as in Version II, and converted on Mercury to a five-channel octal tape. This latter tape is read into Atlas using the Octal Input routine in the Fixed Store and hence all 1024 words in the Subsidiary Store are available for programme.

It differs from Version II in that

a) no sub-routines (e.g. read one character) are included.
b) no floating point numbers are allowed.
c) a Title directive, i.e. the line containing the T and the following line, is ignored.

Section 7 Magnetic Tape

7.1 The Ampex Mechanism

The Tape Layout

The tape mechanism is the Ampex TM2 (FR 300) using one inch wide magnetic tape. There are 16 tracks across the tape, used as follows:

- 12 information tracks
- 2 clock tracks
- 1 block marker track, each block marker consisting of 13 digits
- 1 reference marker track, each reference marker consisting of 5 digits. (This is used for addressing purposes).

The reference markers are about 0.39 inches ahead of the corresponding block marker. These markers are on the two outside tracks of the tape.

Each clock digit is associated with a set of 6 information digits and the clock tracks are in the middle of the six relevant information tracks.

Information is stored on tapes in blocks or sections of 512 words. Each such section is preceded by an address and followed by a checksum; the address and checksum being written across the tape in the information channels. Associated with each information block are two block markers, "leading' and "trailing'. The address of the block is written "parallel" to the leading block marker for each section of tape; a zero address is on the tape "parallel" to each trailing block marker. Each address on tape consists of four stripes each of twelve bits; the first two stripes are zero, the third consists of the most significant bit of the address plus eleven zeros and the fourth contains the least significant twelve bits of the address. Only the thirteen meaningful bits are read into the Present Block Address register. Tapes are pre-addressed by a special run on the machine before they are used, and the fixed position of addresses permits selective over-writing of sections. Checksums are of 24 bits with end-around carry and are written as two stripes: they are used to check the accuracy of all reading and writing operations.

Each 512 word section of information is about 5.46 inches long, with a gap of about 2.3 inches between sections. The distance between the first digits of the leading and trailing markers is about 6.93 inches and the block markers are about 0.032 inches long. There are independent write and read heads separated by about 0.39 inches; when not operating at full speed the tape is stopped with the heads in the gap between sections, ready to write or read the next section. Tapes are 3600 feet long and hold some 5500 sections or two and a half million 43-bit words.

Performance

The normal tape speed is about 120 inches per second and there are 375 binary digits per inch on each track. This gives an instantaneous transfer rate of 90,000 six-bit characters per second, or one 48-bit Atlas word in 89μs. (Allowing for the block gap the effective transfer rate is about 64,000 characters per second per channel). There are also fast wind and rewind operations at about 180 inches per second.

It is possible to read when the tape is moving either in the forward or reverse direction, but writing is only possible when the tape is moving forwards.

Control.

Atlas may control a maximum of 32 magnetic tape mechanisms. Each mechanism is connected via one of eight channels which can operate simultaneously, each channel controlling one read, write or search operation. Wind and rewind operations are autonomous and only need the channel to initiate and, if required, terminate them.

The exact layout of the control system depends on the individual installation. For the Manchester University computer with eight mechanisms and eight channels one mechanism can be connected to each channel at any one time. For an installation with more than eight mechanisms it is envisaged that a switching system will be available whereby the mechanisms are in groups, with up to nine mechanisms in a group, and each group of mechanisms is controlled through two channels. Thus each channel may control any one of nine mechanisms and each mechanism may be controlled by either of two channels. The limitation of a total of 32 mechanisms still applies with this switching system.

Transfers between the core store and the tape are effected via a fast core store buffer containing two words for each channel. To provide synchronisation between the core store and this buffer a fast flip-flop register is provided. One word can be transferred every 89μs. and hence access to the core store may be required every 11μs. if all eight channels are operating (in the worst case this becomes every 8μs.). Of this the selected stack of the core store is occupied for one cycle time of about 2μs. and during the remainder of the time the computer can continue with normal operations. A priority system for drum, tape, and machine accesses to the core store is provided to ensure that no information is lost on drum or tape transfers due to the core store not being "available".

Protection

A Write Permit ring must be fitted to a reel of tape before that reel can be written on. Tapes containing permanent information do not have such a ring.

A Write Inhibit switch is also provided on each mechanism which the operator can use to countermand the presence of the Write Permit ring. It is only possible to write on a tape when both the Write Permit ring is on and the Write Inhibit switch is off.

The write current is automatically inhibited when the tape is moving in reverse or at fast speed, either wind or rewind. It is also automatically switched off after the completion of each write transfer.

Control Buttons and Lights

There are three buttons of the Honeywell-Brown push-button type and one light on each deck. They are:

1) Mains ON/OFF. The button has an amber light behind it which is lit for ON.
The button has to be pressed to change its state from ON to OFF or from OFF to ON.
2) Write Inhibit. This is an alternate action switch which enables the operator to inhibit writing on a tape even though that tape reel carries a Write Permit ring.
On a TM2 deck the Write Inhibit lamp is lit, (green) if either
a) The Write Permit ring is omitted
or b) The Write Inhibit switch is at Inhibit.
On an FR 300 deck only (b) applies.
3) Engage. This button is white and can only be used to Engage the deck. The deck cannot be engaged from the computer nor can a deck be disengaged by a push-button.
4) Disengaged Light. This is a red light at the top of the deck which is lit when the deck is disengaged and the tape is not rewinding.

<u>Loading and Unloading Tapes</u>

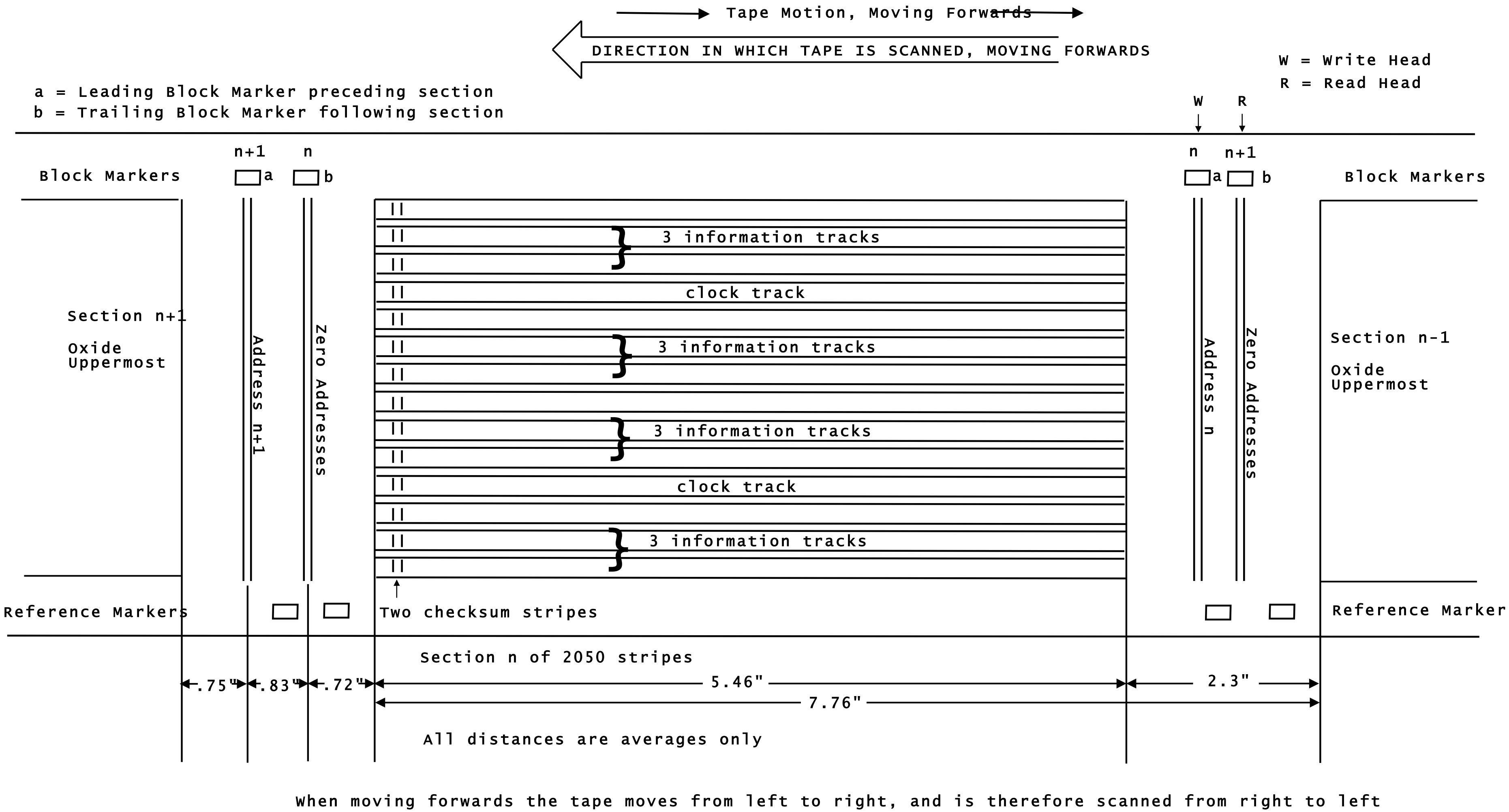
It is only possible to load or unload a tape from the top spool. The procedure for changing a reel is:

1) check that the red lamp is lit
2) open the cover door
3) clamp the leader against the sensing post.
4) unfasten the buckle connecting the tape and the leader
5) remove the tape reel
6) insert the new reel, checking that the Write Permit ring is on if required, and tighten up on the spindle
7) fasten the tape to the leader by the buckle
8) unclamp the leader from the sensing post
9) close the cover door
10) press the engage button
11) check that the disengaged light goes out.

For removing a reel proceed as 1 to 5 above and 6) close the cover door (the disengaged light should stay on)
For placing a reel on an empty deck proceed as for changing a reel except that steps 3, 4 and 5 are omitted.

<u>Failure of Deck Caused by Operator</u>

If the operator should open the door whilst the deck is engaged i.e. when the red disengaged lamp is extinguished, a Deck Failure Interrupt occurs. This occurs immediately if the deck is "on channel" or, if not, when the deck is next brought "on channel". In either case any tape movement is stopped.
After closing the door the "Engaged" button must be pressed to re-engage the deck.



GENERAL LAYOUT OF ATLAS MAGNETIC TAPE (Not to scale)

7.2. Magnetic Tape Extracode Instructions.

1. Identifying Tapes

When a magnetic tape has useful information on it a descriptive title of the information is stored in section 0 of the tape. Before using such a tape a programme or its steering tape must specify the correct name and allocate a number B ($0 < B \leq 100$) to that tape. Subsequent instructions refer to tape B of that programme: normally B is written in the Ba digits of an instruction. Tape numbers greater than 100 are reserved for special uses by subroutines.

The word "Free", used as a title, indicates that the tape described has no useful information on it and is available for general use.

2. Fixed Length Transfers

Basic tape operations transfer information between tape sections and 512 word blocks of the main store. The tape instructions available to the programmer are extracodes based on these operations. Greatest efficiency can be obtained by regarding the store as divided into 512 word blocks and using block transfer instructions for magnetic tape.

In all the following instructions tape section numbers, when specified, are held in the full word address position.

2.1 Block Transfers

In the following instructions the parameter K ($0 \leq K \leq 7$) is used as a counter, but if K = 0 the count is set as 1.

1001	Search for section S on tape B and stop just before it
1002	Read the next K sections from tape B into store blocks P, P+1, ..., P+K-1
1003	Read the previous K sections from tape B into store blocks P+K-1, ..., P+1, P
1004	Write store blocks P, P+1, ..., P+K-1 on to the next K sections on tape B
1005	Move tape B forwards K sections.
1006	Move tape B backwards K sections.

2.2 Organizational Instructions

1010 Mount
Allocate the number B to the tape whose title is stored in locations S, S+1, etc. If this tape is not already available instruct the operator to mount it on channel K of this programme.

1011 Mount Free
Select a free tape on channel K of this programme; if such a tape is not already available instruct the operator to mount one. Write on section 0 of this tape the title stored in locations S, S+1, etc. and allocate it as tape B of this programme. (If K = 0 in instructions 1010 and 1011 it will be assumed that any channel may be used. Instructions to mount tapes may also be given on a steering tape or card.)

101~~2~~4 Write Title
Write on section 0 of tape B the title stored in S, S+1, etc. Also search for section 1.

101~~3~~5 Read Title
Read the title of tape B from section 0 to locations S, S+1, etc. Also search for section 1.

101~~4~~6 Unload (Preserve for later use)
Rewind tape B and disengage the tape mechanism. Instruct the operator to remove the tape, ensure that the correct title is written on the reel and store it for later use.

101~~5~~7 Free Tape (Not required again)
Erase the title on tape B and return the tape to the Supervisor programme for general use. (Tapes may also be freed by means of a steering tape.)

10~~16~~20 Release Tape (Pass it to another programme)
Delete tape B from the allocation of this programme and make it available for another programme, without freeing or disengaging it.
If $n \neq 0$ in instructions 1014 to 1016 above, the number of tape mechanisms reserved for the programme is reduced by one.

10~~17~~21 Release mechanisms
Reduce by S the number of tape mechanisms reserved for use by the programme.

102~~0~~2 Re-allocate
Allocate the name S to the tape which was previously referred to in this programme as b_w.

102~~1~~3 How long?
h'_w = number of 512 word sections available on tape B (excluding section 0). In full word address position of specified half word.

102~~2~~4 Where am I? (See also section 4.2)
After block transfer orders s' = A/Zero
where A =Address of next section on tape B, going forwards. In full word address position of first half word.

3. Variable Length Transfers

3.1 Form of Information on Tape

Some programmes are easier to write if the information on magnetic tape can be treated in variable lengths. For this purpose there is a set of extracodes which use core store blocks as buffers to provide a form of variable length transfer.

The information is stored on tape in groups or strings of words, with a 24-bit marker on each side t o denote the ends of the string. Each writing transfer writes one string on to the tape. A reading transfer may either read a specified number of words of information or read up to the end of a group: in both cases markers are omitted from the transfer.

A number of writing transfers often form a larger unit of information, such as a record or a complete file, and it may be desirable to mark this in some way. For this reason 7 orders of marker, numbered 1 to 7, are provided. Ordinary strings may be terminated by a marker of order 1, records consisting of several strings by a marker of order 2, groups of records by a marker of order 3, and so on, Reading transfers may then read up to a marker of a specified order or greater.

The two markers which indicate the end of one string and the beginning of the next are packed into one word on tape. This contains two 21-bit counters giving the lengths of the two strings and three bits to indicate the order of the two markers, which must be the same.

3.2 Form of Instructions

Variable length working must always be initiated by a start instruction. This selects the magnetic tape to be operated upon and the mode of operation, whether write, read forwards or read backwards. It also sets up the main store buffer and ensures, when reading, that this is replenished whenever there are less than Q words in it, where Q may be specified by the user. If possible Q should be chosen so that the next section can be read from tape before the programme has finished dealing with the remaining Q words. If this can be done the programme need not be held up waiting for magnetic tape: the reading transfer only has to copy from the main store buffer to the specified addresses. Q must be less than 512, but if desired another parameter K may be used to specify a buffer of 512K + Q words.

If the programme reads information in large groups which take a long time to process it may be wasteful to keep a large main store buffer all the time. In this case the start instruction may specify a small buffer. A suitable time before the information is required a select order may be given asking for the buffer to be filled with at least 512K + Q words ready for the forthcoming reading transfer.

A single transfer instruction is provided which operates on the tape which was last selected by a start or select instruction. It transfers information between the main store buffer and the specified programme address, in the pre-selected mode: read forwards, read backwards or write. To change the mode of transfer it is necessary to obey another start instruction. A separate start order must be given for each tape on which variable length transfers are required, but thereafter the select instruction may be used to choose the tape number.

The start operations always initiate variable length transfers to or from the next word on tape, or the previous word in the case of reading backwards. To begin working at a particular word on the tape the start operation must be preceded by a search instruction: in the case of reading transfers the selected address must be the address of a marker at one end of a string. When using the start instruction to change from one mode to another it is not essential to stop at a marker.

The variable length writing operations do not provide a means of overwriting selected words in a magnetic tape section: complete new sections are formed in the core store buffer and the previous contents of the tape are not preserved. If it is required to preserve the beginning of the first section this may be done by preceding the start writing instruction by a start reading forwards instruction with Q = K = 0.

A special start instruction is provided for reading information written by block transfers, with no markers at the end of strings.

4. Variable Length Instructions

4.1 Start Instructions

1030 Start Reading Forwards
Select tape B to be read forwards using variable length transfers, starting at the next word on the tape. Henceforth ensure that at least Q + 512K words are held in the main store buffer awaiting transfer. The buffer is in blocks P, P+1, ..., P+K+1 (or P+K if Q = 0).

1031 Start Reading Backwards
Select tape B to be read backwards using variable length transfers, starting at the previous word on the tape. Henceforth ensure that at least Q + 512K words are held in the main core buffer awaiting transfer. The buffer is in blocks P, P+1, ..., P+K+1 (or P+K if Q = 0).

1032 Start Writing Forwards
Select tape B to be written forwards using variable length transfers, starting at the next word on the tape. Up to Q + 1 buffer blocks are used as required. A marker K is written before the first word of information. ($1 \leq K \leq 7$). The buffer is in blocks P, P+1, ..., P+Q.

1033 Select
Select tape B for succeeding variable length operations, in the mode previously specified for that tape.
If S is not zero in a select instruction in a reading mode, then magnetic tape instructions are initiated

as required to ensure that at least Q + 512K words are available in the main store buffer. This action will only occur if Q = K = 0 in the appropriate preceding start instruction. The buffer will be as specified in that start instruction.

1034 Start Reading Forwards from fixed blocks

1035 Start Reading Backwards from fixed blocks

Instructions 1034 and 1035 are the same as 1030 and 1031 except that they initiate variable length reading for tapes which have been written by block transfers, and are therefore not divided into strings.

4.2 Transfer Instructions

1040 Transfer

Transfer bw words between store addresses starting at S and the selected tape, in the mode (reading forwards, reading backwards or writing) appropriate to that tape. The transfer is terminated on a marker bk.

On Writing, bw words from locations S, S+1, ..., S + bw - 1 are written to the next bw locations on the selected tape. A marker bk is written on tape after them.

On Reading, the transfer continues until bw words of information have been read or until a marker ≥ bk is encountered, whichever is the sooner,

- bw' = The number of words of information actually read.
- bk' = 0 if no marker ≥ bk was encountered.
  = m if a marker m (≥ bk) terminated the transfer or came immediately after word bw.

When reading forwards the next bw' words are read from tape to store locations S, S + 1, ..., S + bw' - 1.

When reading backwards the previous bw' words are read from tape to store locations S + bw - 1, S + bw - 2, ..., S + bw - bw'

If b = 0 when reading, the transfer continues until the first marker is encountered, as though bw were equal to bw'. When reading backwards this means that bw' words are read to store locations S + bw' - 1, S + bw' - 2, ..., S.

1041 Skip

Skip bw words, terminating on a marker b_k.

Skip operates in the same way as transfer except that no words are transferred.

When in a writing mode bw addresses on tape are skipped and a marker b_k is written after them. Note, however, that the previous contents of these addresses, whether information or marker, are not preserved on tape, except when complete 512 word tape sections are skipped.

When in a reading mode the skip continues until b_w words of information have been passed or until a marker $\geq b_k$ is encountered, whichever is the sooner.

b_w' = The number of words of information actually skipped.

b_k' = 0 if no marker was encountered.
= m if a marker m $\geq b_k$) terminated the transfer or came immediately after word b_w.

Note that skip is much less efficient than search for positioning the tape, and should not be used for skipping more than a few sections along the tape.

1042 Mark

Available only when in writing mode.

Writes a marker K after the last word on the selected tape. This marker replaces any marker which was previously on the tape at this point.

After writing a string on tape it may be discovered that the end of a group has been reached. The mark instruction may then be used to change the marker at the end of the string. It may be used again if it is later found that the end of an even higher order group has been reached.

1043 Stop

Stop variable length operations on tape B.

After variable length operations for a given tape have been completed a stop instruction may be given. It will release the buffer blocks associated with those operations. After writing operations it will cause the last part section to be written immediately from the buffer to magnetic tape, but this will also be done by any of the operations:
start, search, unload, release tape, block transfer.

102~~2~~4 Where am I? (see also section 2.2)

After variable length transfers
s = A/W

where W = Address within the section of the current marker on tape B or, if not on a marker, of the next word going forwards.
A = Address of the section containing word W.

1044 Word Search

Search for word W, section A of tape B, where s = A/W

7.3 The Processing of Magnetic Tape Orders

Magnetic tape extracode orders, before they are processed, are broken down by the tape supervisor into one or more orders of the following form:

1. Search for a section S on tape B and stop just before it.
2. Skip forwards or backwards K sections on tape B.
3. Read forwards or backwards K sections from tape B into store pages starting at page p.
4. Write K pages onto tape B starting with page p.
5. Wind or rewind tape B.

These tape orders are placed in the Magnetic Tape Queue. This queue consists of up to 64 entries contained in the Subsidiary store. If the queue should be full when a tape extracode order is given the programme is halted until such time as space has been made available. Orders in the tape queue are processed for each channel in the same sequence as they are entered.

Tape orders are moved in the queue by the extracode updating routine. This routine selects the first queue entry for a given channel and either moves it into position to be processed, or if it is a read or write order moves it into an intermediate location to wait until the necessary store blocks are ready. The other queue entries for the same channel are then moved up accordingly. Usually, the updating is done whilst the tape is stopped, but if the next tape order for the channel refers to the same deck and does not involve a change of direction in the tape movement from the previous order, and it is not a search or a rewind, the updating is done whilst the tape is moving. By this means a series of tape orders can be processed without stopping the tape.

Once the tape order has been placed in position to be processed, the appropriate bits are set in the V-store and the tape is started. From then on until the completion of the order, the tape is controlled by two interrupt routines:

1. The block address interrupt routine.
2. The deck failure interrupt routine.

The block address interrupt routine is entered when a block address interrupt occurs. This routine begins by checking that the block address read into the P.B.A.R. is the same as the one expected. Block addresses at the beginning of each block on tape are numbered in sequence alternating with zero block addresses at the end of each block. Any discrepancy between the value read and the expected value causes the tape to be stopped and the monitor routine entered. If the block address is as expected the routine then goes on to process the current tape order. At the initiation of the tape order it sets the appropriate bits in the V-store according to the order which is to be obeyed. At the termination of the tape order it checks for certain fault and error conditions, such as checksum error, not 512 words in the block transferred, and parity faults. If any error conditions are indicated, the "End Transfer" digit is set, the tape is stopped, and the monitor routine is entered. If no error conditions are detected the routine is free to process the next order.

The deck failure interrupt routine is entered when an interrupt occurs either due to tape mechanism failure or due to an end of a tape being reached. In either case the tape movement is stopped automatically. The tape is stopped if an End of Tape signal is given for a mechanism which is off-channel but in this case no deck failure interrupt occurs. If the interrupt is due to mechanism failure and the routine is unable to reset the interrupt digit then the deck is disengaged. If the routine succeeds in resetting the interrupt digit control of the tape is passed to the monitor. If the end of tape is indicated and the tape order being processed is a search or a rewind the routine deals with it accordingly, otherwise a block address error is suspected and control passed to the monitor.

As soon as a tape order has been successfully completed a routine is entered which unlocks the store blocks. The updating routine is then free to move the next queue entry into position to be processed.

7.4 Basic Magnetic Tape Operations

All magnetic tape orders specified by the programmer are Extracodes which are broken down into semi-basic instructions and placed in a queue. The execution of a basic tape operation is initiated by writing to the appropriate digits in the Magnetic Tape V-store. References in the following paragraphs to particular lines and digits are to the appropriate lines and digits in the Magnetic Tape V-store.

There are eight independent channels, numbered 0 to 7, each of which can initiate and control one search, read, or write order on one of the mechanisms connected to the channel at a time. It is possible by means of these channels to have up to eight magnetic tape search, read, or write orders, plus wind or rewind orders on the remaining mechanisms, all being performed simultaneously.

Each mechanism has a red lamp on it which it lit when the mechanism is disengaged. There is also a corresponding digit in the V-store (lines 16 and 17, digits 24-39) for each mechanism which is set to one when the mechanism is disengaged. The red lamp goes out and the flip-flop is reset to zero to indicate "engaged" when a tape is correctly loaded, the mechanism door closed and the Engage button pressed. The flip-flop is set to Disengage (and the red lamp lit automatically) by the fixed store programme when the mechanism is no longer required. It is not possible for a programmer to give an "Engage" command. A deck may only be Disengaged by the computer when it is on channel.

Associated with each channel in the V-store are the following registers and flip-flops:

a Present Block Address Register (P.B.A.R.)
a Tape Command Register (T.C.R.)
a Select Deck Register
a Block Address Interrupt flip-flop
a Deck Interrupt flip-flop.

Magnetic tape control is achieved by writing to and reading from these registers.

A Magnetic tape mechanism is connected to one of the eight channels by specifying the deck and channel in the Select Deck and Channel Register (line 20, digits 36-24).

The action of connecting a mechanism to a channel automatically disconnects from that channel any other mechanism which may have been connected previously.

36	35	34	33	32	31	30	29	28	27	26	25	24
δ	δ	δ	δ	δ	7	6	5	4	3	2	1	0
		Deck						Channel				

The Channel is selected directly by writing a one in the appropriate digit of digits 31-24.

The Deck is specified by writing its number in binary form to digits 36-32.

It is not possible to read from this register and hence the fixed store programme must keep a list of which mechanism is connected to each channel. The tape mechanism is then controlled by the computer through the Tape Command Register for the channel.

Up to three attempts are made to carry out each magnetic tape operation i.e. if the first attempt is unsuccessful (check-sum or deck failure) the Supervisor initiates one or two further attempts. Information is printed out for the Engineers on every failure even if the transfer is successfully completed at a second or third attempt. (No second attempt is made in the Block Address Search Routine when a Block Address is misread providing it is not the required address. An indication of a failure is however printed).

For the purpose of searching for a particular block address on tape the Present Block Address Registers are provided (lines 0-7, digits 36-24). Each line contains a thirteen bit Block Address for its respective channel. Whenever a block marker is encountered on the tape the block address (which is zero for the trailing block marker) is read into the P.B.A.R., the Block Address Interrupt flip-flop is set for the appropriate channel (line 18, digits 31-24) and an automatic interrupt occurs. When it is required to stop a tape, a Stop command must be given within 1.5 milliseconds of a Block Address interrupt occurring. A "Stop interrupt" occurs when the tape movement has been stopped.

The P.B.A.R.'s can all be read by the computer and, in addition, for purposes of addressing and re-addressing tapes only, the computer can write to the P.B.A.R. for channel 7. The Block Address Interrupt flip-flops are set to 1 by the tape co-ordinator; they are reset to 0 by writing 1 from the computer.

The Tape Command Registers (lines 8-15) receive instructions from the computer and contain information regarding tape control on the respective channels as follows:

(a) The "Check Sum Fail" digit (digit 24) is set to a 1 automatically when the Block Address interrupt digit is reset. If the Check Sum Comparison is successful the digit is reset by the Tape Co-ordinator. It can be read by the Computer.

During a transfer half-words of 24 bits are added together, with an "end around carry", until a checksum for the complete block has been formed.

For a write operation, the checksum is written immediately after the block. During a write operation the read head, which "follows" the write head, reads back the information and another checksum is computed. Then the checksum written on the tape is read back. When the write operation has been completed the three checksums, namely the one calculated during the write transfer, the one calculated during the reading back, and the one written on the tape and read back, are compared. If there is no discrepancy, the checksum failure digit in the associated tape command register is reset to zero.

For a read operation a checksum is calculated as the data is read and this is compared with the checksum read from the tape. If there is no discrepancy the appropriate checksum failure digit is reset to zero.

(b) The "Not 512 Word Transfer" digit (digit 25) is set to a 1 automatically when the Block Address interrupt digit is reset. It is reset by the Tape Co-ordinator when 512 words have been read. If more than 512 words are read the digit is set, this time by the Tape Co-ordinator. It is read by the computer after the second Block Address Interrupt. If this digit is read as a one when the transfer is completed this indicates a fault unless an Orion tape is being read.

(c) "End of Tape" (digit 26) is signalled when the metal backed leader at either end of the tape is sensed by the mechanism. This digit can only be read by the computer. End of Tape gives a Deck Failure Interrupt at either Fast or Slow speeds if the mechanism is on-channel. This digit is reset automatically either when the metal backed leader comes off the sensing-post or when the deck is disengaged.

(d) "Write Permit" (digit 27) indicates that the Write Permit ring is on the tape reel, and the Write Inhibit switch is off for the mechanism connected on-channel. This digit can only be read by the computer, and is one only when both conditions are satisfied.

(e) "End Transfer" (digit 28). This digit is set to terminate a transfer when some fault condition is detected. This may be (i) if insufficient pages have been allocated for an Orion tape transfer, (ii) if a non-equivalence is obtained between the transfer address and the Page Address Registers or (iii) for instance in a read forwards transfer, the leading address is checked and found to be in error. If the tape motion is allowed to continue the "End Transfer" signal can be given to stop the Tape Co-ordinator from wasting time in the Core Co-ordinator.

It may also be used to reset the "Write" flip-flop and switch off the write current if the automatic switch-off of the write current at the end of a write transfer has failed.

(f) "Write" (digit 29). The Write digit must be set before a write transfer can take place. It is reset automatically when the trailing block marker is read. To guard against overwriting tapes due to a fault in this register the fixed store programme tests the Write digit to make sure it has been reset correctly when the transfer is completed,

(g) "End Read at next Block Address" and "Read at next Block Address" (digits 30 and 31).
The "Read at next Block Address" digit is set for a read transfer when the read head is in the inter-address gap before the required block. When the block address interrupt occurs, the fixed store routine sets the "End Read at next Block Address" digit. The reading transfer is then concluded at the end of the block. Digit 31 is read as a one if examined when a read transfer is taking place.

(h) "Normal Read" and "Recover Read" (digits 32 and 33).
The "Normal Read" digit sets the bias level so that tape signals below a certain strength are not detected, thus not reading spurious signals. After a check-sum failure or a failure to read all 512 words whilst carrying out a Read transfer the Supervisor, by specifying "Recover Read" may instruct a further Road Transfer with the Bias level reduced to enable partially written signals to be recovered. This procedure will however only be used in extreme circumstances and it will necessitate the Engineer in charge of the machine giving instructions to the Supervisor for it to be carried out.

(i) "Buffer Parity Fault" (digit 34).
For a Write transfer the information is checked for correct parity as it is read from the buffer. If a 24 bit word is found to have incorrect parity both the Parity 6 digit in the Central Computer V-store and also the Buffer Parity digit for the appropriate channel are set. An interrupt occurs due to the setting of the Parity 6 digit but the transfer is completed. This digit is reset to zero automatically when the Block Address interrupt digit is reset.

(j) "Disengage Deck" (digit 35).
Writing a one to this digit automatically disengages the mechanism at present on-channel. The red Disengage lamp on the mechanism is not lit however until any current Wind or Rewind operation is completed.

(k) "Stop" (digit 36).
About 1.5 milliseconds after reading a Block Address the equipment examines the Stop digit and, if it is set, stops the tape. Normally when it is desired to stop the tape this digit is set by the fixed store block address interrupt routine within 1.5 milliseconds of reading the block address. After this time interval, no further changes can be made to the Tape Command Register until the tape has stopped and the Stop Interrupt occurs. This interrupt uses the same "look at me" as the Block Address interrupt but is distinguished from it by the fact that the contents of the P.B.A.R. remain unchanged from the previous

tape interrupt. The tape must be stopped before a change of tape direction order can be initiated.

(1) "Start" (digit 37).

(m) "Forward" and "Reverse" (digits 38 and 39).

(n) "Normal Speed" and "Fast Speed" (digits 40 and 41).

These registers are set to indicate the required direction of motion and speed (120 in/sec. or 180 in/sec) of the tape.

The write current is automatically inhibited if either the Reverse or Fast Speed digit is set.

The Deck Interrupt flip-flop (line 27 digits 31-24) is set and an interrupt occurs if a fault is detected, or the end of a tape is reached, when the deck is on-channel. The faults may be an overload relay, a loop warning, vacuum failure, incorrect commands, etc. The tape supervisor attempts to clear the failure digit by writing a one to it but if the digit remains set an indication is printed out that a fault condition exists on the particular mechanism and the mechanism is disengaged by programme. If the failure digit can be reset i.e. the fault has cleared itself, the mechanism can obey further commands. An indication is printed out that a failure did occur.

When a programmer has finished with a reel of tape, the appropriate mechanism is set to Fast Wind, Reverse, and Start. It is also disengaged and may be put off-channel if the channel is required for another mechanism. The rewind continues after the mechanism has been taken off-channel until the metal backed leader is reached, the red disengage lamp on the mechanism then being lit.

To specify the page used in a tape transfer one of the Page Address Registers must be set as follows:

digit	22	21	20	19	18	17	16	15	14	13	12
	1	1	1	1	1	δ	δ	δ	ϵ	ϵ	ϵ

Digits 17-15 specify the tape channel concerned.

Digits 14-12 are set to 000 for a forward transfer and 111 for a read backwards transfer. (When reading an Orion tape with blocks of more than 512 words they are set to 000, 001 etc., or 111, 110 etc., for the different pages affected by the transfer).

Digit 23 is used as a lockout digit and is set as a 1. When the transfer is complete the lockout digit must be cleared and the P.A.R. reset to give the correct Main Store Address for the block of information.

For engineering purposes a Magnetic Tape Transfer Address is manufactured and held in the machine as follows:

digit	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3
	1	1	1	1	1	δ	δ	δ	ϵ	ϵ	ϵ	α	α	α	α	α	α	α	α	α

Digits 17-15 and 14-12 have the same significance as above.

Digits 11-3 are used as the address within a page of the word being transferred and also as a count up to 511. For a transfer in the Forwards direction they are initially

all set as 0's and for a transfer in the Reverse direction they are initially all set as l's.
Digits 2-0 are always zero and do not need to exist.
The "equivalence" operation between this transfer address and the Page Address Registers is carried out on digits 22-12 only. If non-equivalence is obtained between the transfer address and all the Page Address Registers due to either a register not being set correctly or a fault occurring the “Non-equivalence, drum or tape” digit in the Central Computer V-store (line 1, digit 28) is set and an automatic interrupt occurs.

On writing to tape the parity of each 24 bit word is checked in the Distributor and the Parity 3 digit (digit 29, line 0 of the Central Computer V-store) set if incorrect parity is detected. The parity is also checked in the Tape Co-ordinator at the output to the Check Sum adder and the Parity 6 digit (digit 26, line 0) set if incorrect parity is detected. In this case the Buffer Parity Fault digit is also set for the appropriate channel.

On reading from tape the parity digit is formed in the Tape Co-ordinator and is checked in the Distributor, the parity 3 digit being set if appropriate.

If incorrect parity is detected in any of these cases an automatic interrupt occurs. The transfer is completed and no indication is given in the V-store as to which word had incorrect parity.

7.5 The Magnetic Tape Monitor

As a magnetic tape order proceeds, from the time it enters the tape queue until the time it has been completed, several checks are made both on the order itself and on the equipment employed. Before the order is processed checks are made to determine if the deck is engaged and, if so, a check is made to determine if it is at the last block on the tape. If the order should be a write order a check is made to determine if the write permit ring is present. If the order is a search order a check is made to make sure that the block is on the tape. Any of these fault conditions cause the programme to enter the programme monitor. However, if an error condition is encountered after the tape order has been initiated the tape is stopped and control is passed to the magnetic tape monitor.

The magnetic tape monitor routine is a main store programme which is obeyed in extracode control. It begins by first printing out the type of error involved and goes on, in most cases, to try to repeat the order three or so more times to see if the error has cleared itself. If the error still persists more involved techniques are employed, to identify more precisely the type of error which has occurred.

The magnetic tape monitor routine deals with the following error conditions: check sum failure on read or write, block transferred different from 512 words but not an Orion tape, block address missed either due to tape creep or to an interrupt failing to register, block address incorrect in sequence, failure to reset the write digit after a write order, parity checks on transfer orders, and deck failure interrupts. Once an error has been detected, the tape unit involved remains under monitor control until the monitor releases control of it.

7.6 Addressing and Re-addressing Tapes

Atlas normally uses pretested and preaddressed tapes of a fixed block length of 512 words. The pretesting and preaddressing helps to ensure that the tapes are reliable and that any bad patches have been eliminated from use. If a faulty block is detected during normal operation the relevant block number is stored and later recorded on some output medium (e.g. a punched card) for future use with the readdressing routine.

Atlas tests and addresses its own tapes and for this purpose the Tape Address Command Register (T.A.C.R.) is provided in the Magnetic Tape V-store (line 21). This register, which exists only on channel 7, provides the instructions required for addressing tape, such as the writing of reference marks, the writing of block addresses, the testing of block addresses, and the testing of blocks on tape by writing ones across it.

In conjunction with the T.A.C.R. are two interrupt control routines in the main store; namely, the addressing routine, and the readdressing routine. The addressing routine tests the tape for bad patches, writes in the good blocks with their reference marks, and numbers the block addresses consecutively. The readdressing routine, which is called in when bad blocks have been found in normal use, eliminates the bad blocks from the tape and re-numbers the remaining block addresses consecutively.

In order to address or readdress a tape, an instruction is given to the computer by means of an input device. The supervisor then prints out an instruction to the operator to modify deck K for addressing or readdressing purposes and to load onto it the reel of tape. After the operator has done this, the supervisor interrogates the Deck Modified flip-flop (digit 35) in the T.A.C.R. to determine if the deck has been modified for addressing; if so the computer proceeds with the addressing or readdressing routine.

The addressing routine allows about six feet of tape before the first reference marker is written. A reference marker is written on the tape when a one is written into digit 31 of the T.A.C.R. At the same time that the writing of the first leading reference marker is initiated the Permit Count, Address Tape and Write Ones digits are set. Two counts are then carried out automatically by the machine. The first indicates when the trailing reference marker is to be written and the second indicates when the leading marker for the next block is to be written. These counts are automatically reset whenever a leading reference marker is written. The programme forms the next address (zero for a trailing address) to be written and puts this to the P.B.A.R. for channel 7. When a reference marker is read a block marker and address are automatically written on the tape. After the address is written the machine writes ones in the information tracks until immediately before the trailing reference marker is written. After being written each address is read back, a block address interrupt occurring, and checked by the programme. The ones are also read back and checked automatically. If a failure is detected the Address Fault flip-flop (digit 24) is set and after writing the leading markers for the next block the tape is reversed and the reference and block markers for the faulty block erased. The addressing is continued from after the faulty block.

When readdressing a tape the addresses of the faulty blocks are fed into the computer and a search carried out for the block address one greater than the highest numbered faulty one. The tape is reversed and the reference and block markers eliminated for the faulty blocks. After the last of these has been done the tape is stopped and the remaining blocks on the tape readdressed. A similar procedure is followed to that in the addressing routine except that the reference markers remaining are used to indicate the presence of good blocks.

The Leading Address Indication bit (digit 34) is provided for use in the readdressing routine as a check against a reference marker being missed and the trailing markers being interpreted as leading markers. This digit is set by the programme whenever a leading address is written and it is reset automatically between 10 and 50 m.s. later. If it is still set when a "trailing” marker is reached the tape and programme are out of step.

On completion of the addressing or readdressing routine, an instruction is printed out for the operator to restore deck K for normal use. After this has been done, the routine finally writes in the first block of the tape the number of blocks available on the tape before returning control to the supervisor.

7.7 Reading Orion Magnetic Tapes on Atlas

A magnetic tape block written on an Atlas computer contains 512 forty-eight bit words. A block written on an Orion computer may contain up to 4,096 words. Provision is therefore being made for reading tapes on an Atlas that have been written on an Orion.

If a block of less than 512 words is being read in the forward direction the transfer is terminated in the normal way when the trailing block marker is read. This also gives the signal to compare the checksum read from the tape with the checksum calculated from the information read from the tape; the appropriate channel checksum digit (digit 24, line 8-15 of the Magnetic Tape V-store) is set to 0 or 1 according as there is agreement or discrepancy. The Not 512 Word Transfer digit (digit 20, lines 8-15) is left set but no indication is given as to how many words have been read.

For a block of more than 512 words the fixed store programme must set two or more Page Address Registers appropriately. For a "forwards" transfer the count takes place in digits 11-3 of the tape transfer address as usual and after 512 words have been transferred a "carry" into digits 14-12 occurs. The first 512 words are read to the first page (i.e. the page with digits 14-12 of the Page Address Registers set to 000), the second 512 words are read to the second page (i.e. digits 14-12 of the P.A.R. set to 001), and so on until the transfer is complete. If the transfer is of more than 512 words the Not 512 Word Transfer digit is set.

If the order is "Read Backwards" a similar procedure is followed except that as the words are read from the tape they are transferred to registers 511, 510, 509, etc. of the appropriate page. For a transfer of more than 512 words the first page involved must have 111 in digits 14-12 of a Page Address Register, the second page must have 110 in these digits, etc. The fixed store programme then has to re-number these blocks appropriately. The transfer is signalled as complete when the leading block marker is read.

Section 8. The Peripheral Equipments

8.1.1. General Principles

For consistency it is desirable that all peripherals should obey the same rules as far as possible. These rules are given below and further details as they apply to the different peripherals are given in the subsequent sections.

All peripherals have the following digits associated with them in the appropriate part of the V-store.

a) a Disengaged/Engaged digit
b) a Stop/Start digit
c) one or more Look At Me digits and corresponding Put Out Look At Me digits.
d) Information digits

Most equipments also have

a) an Overdue digit
b) an Operator and/or Engineer Attention digit or digits
c) a Disabled digit

It is assumed that if the cable carrying the control signals from the peripheral to the computer is disconnected or the peripheral is switched off the peripheral appears to the computer to be Disengaged, no Look at Me’s for the peripheral are set and the Stop/Start digit reads Stopped. The state of the remaining control digits is irrelevant. It is also assumed that when the cable is re-connected or the peripheral is switched on the equipment appears Disengaged with no Look At Me's set and the Stop/Start digit reading Stopped. When an equipment is in the disengaged state any control signals which cause mechanical action which could injure an operator or engineer must be ignored.

Control Signals

a) A peripheral can only be Engaged by push-button i.e. it cannot be engaged by an order from the computer. If the button is pressed when one or more of the warning devices indicate a fault exits (e.g. fuse blown) or the peripheral is not in an operational condition (e.g. card hopper empty for the card reader) the peripheral remains in the Disengaged state.
b) A peripheral can be Disengaged by push-button or by the computer writing a one to the Disengage digit in the V-store. The T.R.7 is also disengaged if the Rewind Stop button is pressed during a rewind initiated by the computer.
c) When a peripheral is Disengaged, either by the push-button or by the computer, a signal is sent to the Disengage/Engage digit in the computer to indicate this. The fixed store program reads this digit as a one for Disengaged and a zero for Engaged. This signal is arranged so that the peripheral appears Disengaged if the cable from the peripheral to the computer is not connected.

d) Stop/Start. After a peripheral is engaged a one is written by the computer fixed store programme to the Start digit before any transfer of information takes place. All Look At Me's from the peripherals (except the Overflow L.A.M. from the Anelex Printer and the End of Operation L.A.M. from the I.B.M. Magnetic Tape) are inhibited when this flip-flop is set to Stop, and a signal is sent to the computer to indicate that the peripheral is Stopped.

e) Look At Me/Put Out Look At Me. Every peripheral has associated with it one or more Look At Me signals which indicate to the computer that it either (i) requires further information or (ii) that further information is awaiting transfer to the computer or (iii) that some mechanical action initiated by the computer is taking, or has taken, place. These signals can only be given by the peripheral when it is connected to the computer, it is Engaged and its Stop/Start digit is not reading Stopped.
For some equipments the failure to reset the Look At Me causes an Overdue signal to be given, setting an Overdue digit in the V-store. Subsequent action by the computer differs according to the type of peripheral and is described later for each equipment individually.

f) If the Disengage button on the Hammer Printer, or the Card Reader, or the Card Punch is pressed during a print or card cycle the peripheral completes that cycle before stopping. For the Hammer Printer the Disengaged digit in the V-store is set immediately but a delay of 100 m.s. occurs before the Stop/ Start flip-flop is set to Stopped. For the Card Reader and Punch the appropriate digit in the V-store is set immediately but the peripheral waits until the next End of Card Signal before it stops. In all cases the fixed store program examines the Disengaged/Engaged digit before starting a new print or card transfer.

g) Fault signals. The following faults are detected in the peripherals specified. They set a digit in the V-store to indicate the type of fault and also set the Start-Stop flip-flop to Stopped. This fault signal is read by the fixed store program approximately one second after it is given (by the One Second Interrupt routine) and the peripheral is immediately Disengaged. The fault digit may be reset either (a) when the peripheral is disengaged or (b) when the peripheral is re-engaged or (c) when the fault condition is removed or (d) by a reset signal from the computer. It is an essential condition that the fault digit cannot be reset whilst a fault still exists on the peripheral.
   (i) Overdue as a fault - T.R.5; T.R.7; Card Punch; Card Reader
   (ii) Paper Out - Anelex Printer; I.C.T. Hammer Printer; Teleprinter; Tape Warning (tape out or tape torn) - Teletype punch; Creed 3000 punch; Card Levels - Card Punch; Card Reader
   (iii) Disabled (Mechanical Failure) - Card Reader; Card Punch; Anelex Printer; Hammer Printer;

I.B.M. magnetic tape; Teletype punch; Creed 3000 punch; Teleprinter; T.R.7.

h) Warning Signals. The following signals indicate that some action by the fixed store program is necessary. The Overdue signal for the Hammer Printer is reset by writing a one to it by fixed store program. All the other digits are reset automatically either when the appropriate action has been taken, or the Put Out End of Operation Look At Me signal is sent to the peripheral (I.B.M. tape only).

(i) Overdue as warning - Hammer Printer; I.B.M. tape;

(ii) Paper or Tape Warning (Low) - Anelex Printer; Teletype punch; Creed 3000 punch; Teleprinter.

(iii) Lateral and Longitudinal parities; Tape Indicator; Load Point Indicator; Rewind Complete - I.B.M. tape.

Layout of the V-store

The tables in the following sections include the V-store addresses of the control and information signals of the various peripherals. The notation used is

R where it is possible to Read a digit only; writing has no effect and digits not so marked are always read as zero.

W1 where it is possible to Write Ones only; reading gives zeros and writing zeros has no effect.

RW1 where it is possible to Read and also to Write Ones; writing zeros has no effect. A digit is reset to zero either automatically (e.g. information digits) or by writing a one to its Reset digit (e.g. Put Out Look At Me).

Each peripheral belongs to one of 16 "types" and each type may contain up to 16 separate equipments. The peripherals within a type do not have to be identical. For example the seven-channel and five-channel teletype punches are both Type 8 and the fixed store program indentifies a teletype punch as seven or five channel according to its number within the type. The addresses of the lines for the lowest numbered peripheral of each kind are given in the tables below.

8.1.2. Cable Connections For 80 Way Sockets for Peripheral Equipments

1. Cable and Plug Types

All signals will enter or leave the peripheral by way of Plessey multi-way connectors and co-axial leads.

The peripheral will itself carry an 80 way Socket type 2CZ 108605.

The cable from the computer must be terminated at the peripheral end by an 80 way Plug type 2CZ 108602.

2. Pin Allocations

Pins are arranged in 4 columns "a" "b", "c", "d" and in 20 rows numbered 1 to 20.

To accommodate 40 co-axial leads, the following scheme will be adopted for every plug.

The inner (signal) conductors of the first set of 20 leads will be connected to pins 1 to 20 of column "a". The corresponding outer conductors will be connected to pins 1 to 20 of column "b".

For the second set of 20 leads, the inner conductors will be connected to pins 1 to 20 of column "d" and their outer conductors to pins 1 to 20 of column "c".

3. Information and Control Signal Cables.

The sockets in the peripheral will be numbered beginning at 1.

The allocation of cables and pins for all machines is shown below, only the pin connections for the inner conductors being shown. It is to be understood that signals taken to a pin in column "a" have their outers on a pin of the same number in column "b" while those going to a pin in column "d" have their outers on a pin of the same number in column "c".

Pin Allocations for Information Cables.

	I.C.T. Hammer Printer	I.C.T. 582 Punch
Cable 1 Col "a" Pins 1 to 20 Col "d" Pins 1 to 20	Print Info. Cols. 1 to 20 Cols 21 to 40	Punch Info. Cols. 1 to 20 Cols 21 to 40
Cable 2 Col "a" Pins 1 to 20 Col "d" Pins 1 to 20	Print Info. Cols 41 to 60 Cols 61 to 80	Punch Info. Cols 41 to 60 Cols 61 to 80
Cable 3 Col "a" Pins 1 to 20 Col "d" Pins 1 to 20	Print Info. Cols 81 to 100 Cols 101 to 120	None
Cable 4 Col "a" Pins 1 to 20 Col "d" Pins 1 to 20	None	Check ReadInfo. Cols 1 to 20 Cols 21 to 40
Cable 5 Col "a" Pins 1 to 20 Col "d" Pins 1 to 20	None	Check ReadInfo. Cols 41 to 60 Cols 61 to 80
Cable 6 Col "a" Pins 1 to 20 Col "d" Pins 1 to 20	None	None
Cable 7 Col "a" Pins 1 to 20 Col "d" Pins 1 to 20	None	None

<u>5. Pin Allocations for Control Signal Sockets</u>

	I.C.T. Hammer Printer.	I.C.T. 582 Punch	I.C.T. 593 Card Reader	
Col “a”	Socket No 4.	Socket No 6.	Socket No 1.	Socket No 2.
Pin 1			Read Information Row No 9 (bottom row V-digit 36)	
2			8	
3			7	
4			6	
5			5	
6	Paper Warning	Operator Attention	4	Operator Att.
7			3	
8			2	
9	Paper Feed Busy		1	Reset Overdue
10	Character Overdue	Overdue	0	
11	Disengaged	Disengaged	11	Disengaged
12	Disabled		10 (top row V-digit 47)	Disabled
13	Put Out LF L.A.M.	Card Wreck		
14				
15				
16	LF L.A.M. (+ve)	Brush L.A.M. (+ve)		End of Card L.A.M. (+ve)
17	LF Overdue	Put Out Brush L.A.M.		Put Out End of Card L.A.M.
18				
19				
20				
Col “d”				
Pin 1			Check Information Row No 9 (bottom row V-Digit 36)	Put Out Column Ready L.A.M.
2			8	
3	Character L.A.M. (+ve)	Punch L.A.M. (+ve)	7	Column Ready L.A.M. (+ve)
4	Disengage	Disengage	6	Disengage
5	Start	Start	5	Start

6	Put Out Character L.A.M.	Put Out Punch L.A.M.	4	
7	Datum		3	Stopped
8	Stop	Stop	2	Stop
9	Start Throw	Do not off set	1	Don't Reject
10	Feed 1 line		0	
11	Feed 2 lines	E.O.C. L.A.M. (+ve)	11	
12	Feed 3 lines	P.O. E.O.C. L.A.M.	10 (Top Row V-digit 47)	
13	Feed 4 lines			
14	Feed 5 lines			
15	Stopped	Stopped		
16	Stop Throw			
17				
18				
19	Reset Overdue	Reset Overdue		
20				

8.1.3. Cable Connections for 60-way Sockets for Peripheral Equipments.

Inner	Outer	Paper tape and Teleprinter Equipments	Anelex
1C	1B	Information 1	Information 1
2C	2B	Information 2	Information 2
3C	3B	Information 3	Information 3
4C	4B	Information 4	Information 4
5C	5B	Information 5	Information 5
6C	6B	Information 6	Information 6
7C	7B	Information 7	Information Stroke
8C	8B	Information 8	
(9C and	9B)	Reference Level	Reference Level
10C	10B		End of Line
11C	11B	Put Out Look At Me	Put Out Look At Me's
12C	12B	Disengage	Disengage
13C	13B	Start	Start
14C	14B	Stop	Stop
15C	15B	Reset Overdue or Check Fail	Clear Core Store
16C	16B	Rewind	
17C	17B	Stop Rewind	
18C	18B	Plugs out link 18B and 18C	Plugs out
19C	19B		
20C	20B	Tape Warning (Output only)	Paper warning
1A	2A	Look At Me (+ve)	Print Look At Me (+ve)
3A	4A	Disengaged	Disengaged
5A	6A	Stopped	Stopped
7A	8A	5 Track	
9A	10A	Overdue	Overflow Look At Me (+ve)
11A	12A	Tape broken (Input only)	
13A	14A	Disabled	Disabled
15A	16A	Check fail	
17A	18A		
19A	20A		

All signals are negative unless otherwise stated.

All pins in the associated plug are wired with coax except 9C and 9B which are wired with 14/.0076 and 18C and 18B which are linked in the plug.

8.2 The I.C.T. Card Reader.

Performance

The card reader is the I.C.T. Type 593 (591) which reads 80 column cards at a rate of 600 cards per minute, reading successive 12 bit columns. Cards are transported through the reading mechanism from the right, where the hopper is situated, to the left where the main and subsidiary stackers lie.

There are two separate sensing stations, read and check-read stations, separated by one column width i.e. when column 2 is over the read station, column 1 is over the checking station. The two stations are identical and comprise a block of twelve photocells suitably masked.

The two 12-bit columns over the reading stations are separately registered on two sets of staticisors, this information being available for practically the whole of the period between two successive columns nearly 700µsecs.

At 600 cards per minute the card cycle occupies 100ms, the cycle being divided into 360°. All occasions within the cycle are identified by quoting its position in degrees and time (milliseconds). Cards are fed into the throat of the card track at 0° and reach the first sensing station at 90° (25ms). The distance between the two sensing stations is one column-pitch i.e. 2.5° (2/3ms) and information is passing over the sensing stations for a period of 200° (54⅔ms).

The sequence of actions which feed the next card is initiated at 320°, that is 14° (4.0ms) after the end of the preceding card.

No independent means of recognising the end of a card is built into the reading head but an End of Card Look At Me signal is given about 10µsecs after the 81st Column Ready interrupt for a card (i.e. the Check Read interrupt for the 80th column). The number of cards in the reading track is normally three when the machine is running at full speed, although in certain circumstances four cards may be involved in a card wreck.

The Card Hopper

Cards are placed face down in the hopper with column one leading, up to a maximum capacity of 2000. Cards are fed in two separate stages; a reservoir of about 1600 cards is held up by a special clamp, while a separate lower "working block" contains about 400 cards which are sequentially available to the feed knives. The level of the working block is crudely sevoed by a micro-switch which arranges to release approximately 75 cards from the reservoir into the working block when the level becomes low. Another micro-switch is arranged to sense when there are only 300-325 or so cards in the working block and none in the reservoir. When the reservoir has released the last of its cards the operator may either put the card-weight in position in which case the cards will run through uninterruptedly, or, coming across the machine when it has been automatically stopped (and Disengaged by the fixed store program) Disengaged because the level of cards has run lower than the level of 300, she will be required to press the Engage button, having first put the card weight in position. A micro-switch at the extreme right-hand side of the hopper senses when the final card has been despatched. This indication is almost simultaneous with the movement of the final card because of its positioning but the Stop digit is not set until 315° (87.5ms) i.e. after the card has been read and checked.

The One Second Interrupt routine detects a card reader which has Stopped with a low level of cards, and Disengages it. If there are more cards to be read, the operator must place them in the reservoir and press the Engage button again.

Alternatively, cards may be placed on top of the reservoir while the machine is running, but this must be done with care.

A vacuum operated device ensures that the cards are bent into a shape acceptable by the feed throat.

The Card Stackers.

After the cards have passed the sensing stations they are passed to one of two stackers.

Normally, after reading the card, the fixed store routine gives the Do Not Divert signal and the card is fed to the main stacker. This holds a total of 2000 cards, at which level a micro-switch senses the condition and Stops the machine, also setting the Card Levels digit in the V-store as an indication to the fixed store programme of the reason for the stoppage. Cards fed into this stacker are first turned through an arc of 180 and then stacked i.e. they are added to the bottom of the stack. This results in the cards being face up and in the order in which they were originally put into the hopper.

Should a card be detected as having been faultily read, the computer omits the Do Not Divert signal. This causes a flap to be operated within the machine which diverts the card down another track leading to a subsidiary stacker. Cards are stacked face down with the last card on top. Should the level of this stacker exceed 500 cards a micro-switch operates to Stop the machine, and also sets the Card Levels digit in the V-store.

The signal inhibiting the operation of the diverter flap must be given before 45° (12.5ms) of the next card cycle. On the detection of a reading error the monitor programme is called in and the alternatives of continuing to read cards or stopping the machine and re-feeding the mis-read card (and any subsequently read) are available.

Card Misfeeds, Jams and Wrecks.

Whether a card has actually been fed will be detected at 306° (85ms) during that cycle. The mechanism is stopped completely and the Stop digit is set immediately should a misfeed or jam be detected.

Should a card jam at the reading station the next card causes a mis-feed thus stopping the machine as in the previous paragraph.

A card wreck may occur at any point along the card track from the feed knives up to either of the stackers. At 306° of the first cycle the back edge of the card is registered. If during the fourth cycle no such signal is received at 27° ± 6° when the card should be in the stacker, the failure to receive the signal is detected. Thus three cycles elapse after the feeding of the card was initiated before an indication of a wreck after the diverter is given, and up to four cards may be involved. A similar detection of a pile-up at the diverter-stacker is also given.

Controls and Indicators.

a) Engage

This is a single action push button which when pressed Engages the reader provided (i) there are no fault conditions on the reader (ii) the stackers are loaded and the hoppers are not loaded beyond their defined limits and (iii) there are no card wrecks or feed failures which have not been cleared. Immediately the reader is Engaged a signal is sent to the computer resetting the Disengaged digit in the V-store to zero. This change is detected by the fixed store programme which then initiates the reading of cards. The reader motor is switched on when the Engage button is pressed and is automatically switched off if no cards are read for a period of 30 seconds at any time when the reader is Engaged. If this happens a delay of 1.5 seconds to allow the motor to be run up again occurs after the next Start order before a further card is fed.

This indicator has a Green light behind it which is lit when it is Engaged and a White light which is lit when it is Disengaged.

b) Disengage

This is a single action button which when pressed Disengages the reader. Immediately it is pressed a signal is sent to the computer to set the Disengaged digit to one. If it is pressed when the Stop/Start flip-flop reads Started, i.e. during the feeding of cards, the Stop/Start Flip-flop is not set to Stopped until the next End of Card signal.

This button is lit Red when the reader is Disengaged and White when it is Engaged.

c) Started/Stopped Indicator

This is an indicator which is lit Blue when cards are being read by the reader and White when the reader is stopped.

d) Power On Indicator

This indicator is lit Green when the power supply to the logic is on and is in darkness at other times.

e) Card Wreck Indicator

This indicator is lit Yellow when a card wreck, jam or misfeed has been detected and is in darkness at other times. The light is extinguished when the "Clear Card Wreck" button on the engineer's control panel of the reader is pressed.

f) Card Levels Indicator

This indicator is lit Yellow when one of the card hoppers or stackers requires attention. It is extinguished automatically when the appropriate card level is attended to.

g) Manual Warning
This is lit Yellow when the Automatic/Manual switch on the engineers control panel is in the Manual state; it is in darkness when this switch is at Automatic i.e. in the normal operating condition.
The following controls and indicators are intended for engineer's use only and are situated on the Engineers Control Panel for the reader.
a) a switch to enable marginal testing of the photocells to be carried out.
b) an Automatic/Manual switch so that by switching to Manual the reader can be tested without involving the computer.
c) a switch to feed cards continuously
d) a switch to extinguish the Card Wreck Indicator and to reset the Card Misfeed digit in the V-store.

The V-store digits

Type 0	(*60040000)		
line	digits		
0	47-36	R	Information
	30	RW1	Read: ~~Disabled~~ Overdue Write: Reset Overdue
	29	W1	Do not Divert
	28	RW1	Read: Card Levels Write: Put Out End of Card Look At Me
	27	W1	Start
	26	RW1	Read: ~~Overdue~~ Disabled Write: Put Out Column Ready Look At Me
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage
8	47-36	R	Check Information

Type 14	(*60043670, *60043700)	
line	digit	
23	24	End of Card Look At Me
24	24	Column Ready Look At Me

(strictly speaking these digits are Type 15 lines 7 and 8 respectively)

Signals to and from the Computer

a) Disengage/Disengaged (digit 24)
To engage the reader it is necessary to press the Engage button; it is not possible for the reader to be engaged directly by the computer. Provided the reader is in an operational condition, pressing the Engage button sets the Disengaged digit in the V-store to read as zero, and runs up the motor. The Disengaged digit is examined by the peripheral supervisor during every "One Second Interrupt" and the reader may be started by the fixed store programme, if it is engaged.
The reader can be disengaged either by pressing the Disengage button or by the computer writing a one to the Disengage digit in the V-store. Disengaging manually has no apparent effect if the reader is in the process of reading a card until the end of the current card cycle is reached. The Disengaged digit in the V-store is set immediately but the Stop digit is not set and the reader not stopped until the end of the cycle. No card is left in the reading track when the reader is disengaged, whether this is done manually or by fixed store programme.

b) Start (digit 27)
Writing a one to the Start digit initiates the feeding of cards. If the motor is not running, a 1.5 seconds delay prevents any feeding taking place until it has reached a stable speed. Cards are then fed continuously until a Stop signal is given either from

the computer or from the reader itself as a fault indication.

c) Step (digit 25)
Writing a one to the Stop digit inhibits all further Look At Me signals and prevents card feeding. If the fixed store routine requires to stop the reader without losing any information from it, a one must be written to the Stop digit within 2.5 m.s. of the End of Card signal being given.
If the Stop digit remains set for more than 30 seconds the motor is automatically switched off.

d) Put Out Column Ready Look At Me (digit 26)
As the card passes over the reading stations an automatic interrupt occurs each time information is available in the computer for the appropriate column. There are 81 such interrupts for each card. For the first, information is available from the main reading station only, zeros being read from the checking station. For the next 79 interrupts information is available from columns 2-80, respectively, of the main reading station and columns 1-79, respectively, of the check reading station.
For the last interrupt information is available from the checking station for the last column of the card, zeros being read from the main station.
Failure to reset this Look At Me digit by fixed store program within 650μsecs of it being given causes both Overdue and Stop signals to be given. The failure is then detected by the One Second Interrupt routine.
The Look At Me signal is not given if the Stop/Start flip-flop is set to Stopped.

e) Put Out End of Card Look At Me (digit 28)
An automatic count is made of the columns read and an End of Card L.A.M. signal is sent about 10μsecs after the 81st Column Ready L.A.M. The fixed store programme writes a one to the "Do Not Divert" digit if all the columns have been read and checked correctly. The Overdue and Stop signals are automatically given if this L.A.M. is not reset by the fixed store programme within 2.5ms of it being given.
This Look At Me signal is not given if the Stop/Start flip/flop is set to Stopped.
After dealing with the End of Card interrupt control is transferred to a routine on extracode control to convert the information read from binary to the specified code.

f) Overdue (digit 26) and Reset Overdue (digit 30)
An overdue signal is given if either (i) the Column Ready Look At Me is not put out within 650μsecs of it being set or (ii) the End of Card Look At Me is not put out within 2.5 m.s. of it being set. For the Column Ready L.A.M. the overdue signal is sent approximately 50μsecs before the information staticisors are reset if no put out L.A.M. signal has been sent. The overdue signal also sets the Stop/Start digit to Stopped, inhibiting any further Look At Me's for that card, and the reader is normally disengaged by the fixed store programme. The Overdue digit is

reset automatically when the Engage button is pressed, or by the fixed store programme writing a one to the Reset Overdue digit.

g) Do Not Divert (digit 29)
This digit is automatically reset to zero i.e. to "Divert" when a card reaches the sensing stations and the first column is read i.e. at 90deg (25ms). It is set by programme to "Do not Divert" when the End of Card signal is received if
(i) the count of the column interrupts has been correct and
(ii) the information read by the main and checking stations has been the same.
This digit is examined by the card reader at 45° (12.5ms) of the next card cycle and the card diverted to the subsidiary stacker if the digit is set to "Divert".

h) Card Levels (digit 28)
This signal is given if
(i) the card hopper is empty
(ii) the card hopper is low (less than 300 cards) and the card weight is not in position.
(iii) the main stacker is full (2000 cards)
(iv) the diverter stacker is full (500 cards)
The Card Levels Indicator is also lit but no discrimination between which of the levels is in question is made. A survey of the various holders can be made quite rapidly by the operator.
This signal also sets the Stop digit and the reader is disengaged by the fixed store programme. If this signal is received part way through a card cycle the reader is stopped at the end of the current cycle, leaving the card track clear. The digit is reset automatically and the Warning Indicator extinguished when the appropriate card level has been dealt with.

i) Disabled (digit 30)
The Disabled signal is given if a card wreck, jam or misfeed occurs or if the illumination source to the reading head fails. The Stop/Start flip-flop is also set to Stopped immediately, inhibiting further Look At Me's for any card under the reading heads, and the reader is disengaged by fixed store programme. For an illumination source failure the digit is reset automatically when the Engage button is pressed provided that the cause of the stoppage has been cleared.
After a card wreck, jam or misfeed the Disabled digit is reset by the engineer pressing a button on the card reader's control panel. The digit is reset to zero when this button is pressed whether or not the fault has been cleared. The reader cannot be re-engaged after a wreck until this button has been pressed and the digit reset.

j) Information (digits 47- 36, lines 0 and 8)
The twelve bits from each column read by the main reading station are copied into digit position 47 - 36 of line 0, the bit from the top row of the card being

read to digit 47 and the bit from the bottom row being read to digit 36.
Similarly the 12 bits read by the check reading station are read to digit positions 47 - 36 respectively of line 8.
These digits are automatically cleared about 40μsecs before each column is read.

8.3 The Anelex Line Printer.

General

The standard installation comprises an Anelex High Speed Printer series 4-1000 with

1. a print barrel rotating at 1000 revolutions per minute,
2. a row of solenoid driver hammers,
3. a code wheel which is locked to the print barrel to determine the time for triggering a hammer for the appropriate character to be printed.
4. a mechanism for automatically feeding inked ribbon and advancing paper through the printer between the hammers and the print barrel.
5. a Format Control which works in conjunction with the paper feed mechanism to automatically vary the vertical spacing of printed lines. It consists of a brush-read paper tape reader which reads sprocket fed paper tape with up to eight channels.
6. mechanisms for manually adjusting the position of the printed line, the paper tension, and the print quality, as well as the phasing of the hammer triggers with respect to the instantaneous barrel position.
7. transistorised pulse shaping, gating and amplifying circuits on printed boards.
8. solid state electronics for driving the hammers,
9. power supplies,
10. core store and associated electronics to generate the control signals from peripheral to computer. The core store holds a complete line of 120 characters.

The printer has 120 print positions spaced 0.1 inches apart along the barrel. Although the barrel speed is normally 1000 revolutions per minute a control can reduce this to 667 r.p.m. when improved printing quality is required.

The traction mechanisms for feeding the paper can be varied in position laterally so that widths of paper from 4 to 19 inches can be used. No indication is given to the computer when the width of paper being used is too narrow to accommodate 120 characters on a line and hence it is the responsibility of the operator to ensure that the required width of paper is loaded and also that the paper is loaded so that the Paper Low detection light is intercepted by the paper.

Each hammer module has two adjustments, one for flight time to control the printing level along the line and the other for penetration to control the thickness of print for the character position.

A single line feed takes 16 m.s. and each additional line feed takes a further 6.67 milliseconds.

The printing of the top copy is by inked silk ribbon with a life of approx. ¾-1 million lines. Up to five copies, plus the top copy, can be obtained using one-time carbon. Changing of the ribbon takes about three minutes.

There are 64 different characters on the barrel in each print position. It is possible to print at 1000 lines per minute if only 48 consecutive characters on the circumference of the barrel are being used. A single line feed can be completed in the time for a quarter of a revolution of the barrel and hence a line of up to 48 consecutive characters may be printed and a single line feed given every barrel revolution. If more than 48 consecutive characters are being used the printing speed is reduced and if all 64 characters are being used 4 lines may be printed every 5 revolutions of the print barrel i.e. 800 lines per minute.

The required logic to detect the presence of characters in the different positions round the barrel for a particular line is contained in the printer.

All paper handling equipment is integral with the printer. The paper feed is controlled by a pair of tractors, one above and one below the print head and tensioning of the paper can be carried out during printing. The form can be raised or lowered by about 3 line spaces during printing.

Lateral tensioning and positioning can be carried out by moving the appropriate tractors while the printer is not operating.

<u>Controls and Indicators</u>

a) <u>Mechanical Controls</u>

These are situated on the front of the print mechanism and are as follows:

1. <u>Penetration</u>

   This controls the depth of print to allow for the thickness of the paper and the number of copies being printed.

2. <u>Positioning</u>

   While printing is in progress the form can be lined up by shifting it up or down by a maximum of three line spaces.

3. <u>Tensioning</u>

   This controls the lateral tensioning of the paper and can only be used while the printer is stopped.

4. <u>Phasing</u>

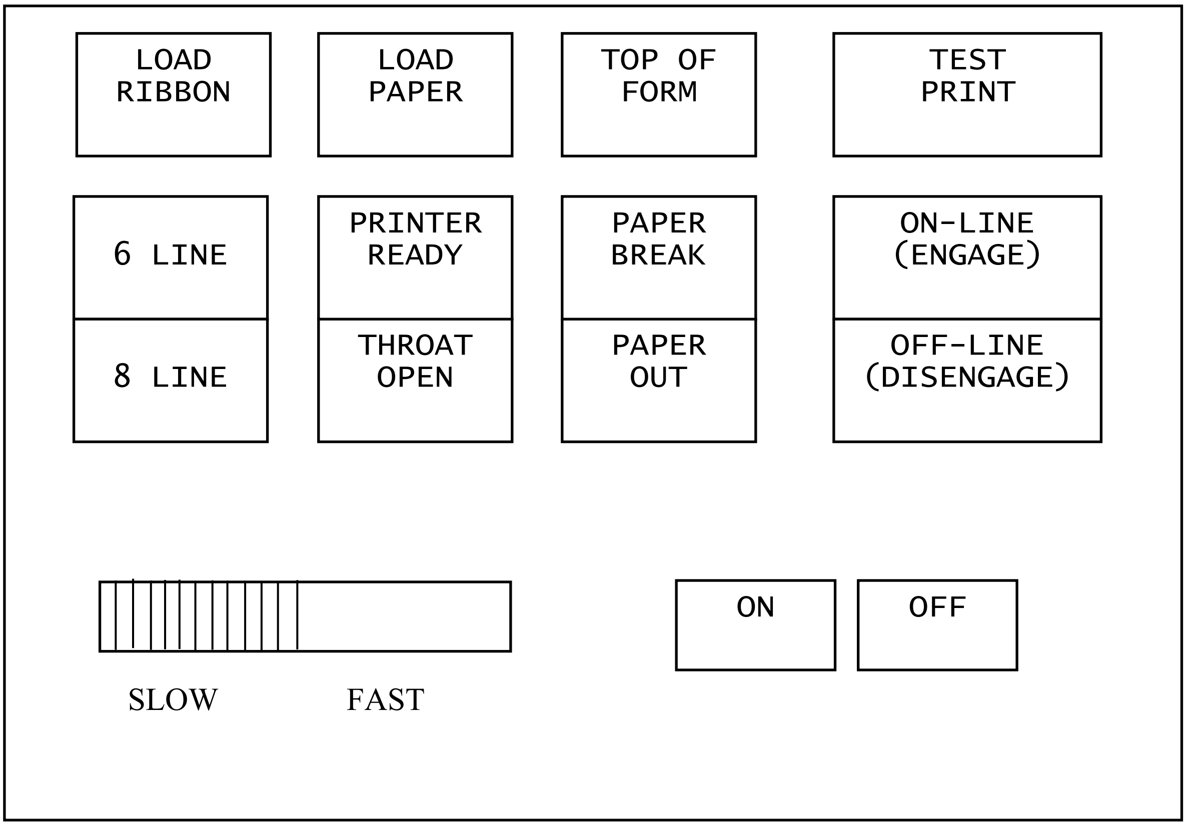
   This controls the timing of the trigger to the hammer drivers relative to the instantaneous barrel position to allow for different thicknesses of paper, or multiple copies, and the two barrel speeds.

5. <u>Barrel Speed</u>

   This is a switch to enable either full or 2/3 full speed (Fast or Slow) for the print barrel to be selected. It is situated on the control panel illustrated on the diagram below.

b) <u>Electronic Controls</u>

The control panel is situated on the front of the printer and is as follows:



1) ON/OFF
These are two push button switches and indicators at the bottom right of the control panel which indicate whether the electronics of the printer are switched on or off. The OFF button is lit Red when power is available to the printer but the electronics are switched off; in this case the remainder of the indicators are unlit. The ON button is lit Green when the power to the electronics is on and in this case the OFF button is unlit.

2) ON-LINE/OFF-LINE (ENGAGE/DISENGAGE)
This is a double-action push button and indicator which corresponds to the Engage/Disengage button on the other equipments and for the sake of consistency it will in future paragraphs be referred to as the Engage/Disengage button. Only one half of this button is lit at one time, the upper half being lit Green when the printer is engaged (On-Line) and the lower half being lit Red when the printer is disengaged (Off-Line). The printer is always in the disengaged state when initially switched on. It is only possible to engage the printer when it is in full working order i.e. no fault condition exists and some paper is in the feed mechanism.

3) PAPER BREAK/PAPER OUT
This is a double indicator which is normally unlit but the top or bottom half is lit white when appropriate. The Paper Out is detected by a reflected light device below the feeding mechanism and gives a warning to the computer but does not set the Stop digit in the V-

store. This therefore allows the output to be discontinued at some convenient point. The "Paper Break" is detected near the print head by microswitch and in addition to giving the Paper Warning signal sends a Stop signal.

4) <u>PRINTER READY/THROAT OPEN</u>
This is a double indicator, the top half of which is lit Green when the printer is in an operational condition; the lower half is lit white when the cover over the print mechanism is open.

5) <u>6 LINE/8 LINE</u>
This has no relevance on the Atlas Anelex.

6) <u>TEST PRINT</u>
This button is normally lit Yellow and is used for testing the printer off-line. It is used in conjunction with six switches which are situated at the back of the printer (and inside the outer covering) and which enables any one of the 64 available print characters to be selected. If this button is pressed when the printer is Disengaged the selected character is printed continuously in all 120 positions across the paper. The printing continues until the first Top of Form indication is detected after the button is released. A further feed of one form takes place, without printing, before the printer stops. Pressing the button when the printer is Engaged has no effect.

7) <u>TOP OF FORM</u>
This button is normally lit Yellow and when pressed a one is "OR"-ed into the Disengaged digit in the V-store making the printer appear to be temporarily disengaged. Paper is fed to the next "top of form" digit specified in Channel 1 of the paper tape used for format control. If there is no top of form digit on the paper tape the paper is fed continuously.
No control signals are normally sent to the printer when it is disengaged. If however the End of Line and Vertical Format signals are sent after the run-out button has been pressed and before the Top of Form feed is finished the contents of the buffer are printed.
If the button is pressed when the printer is engaged the buffer is not cleared but a further Print Look At Me signal is sent at the completion of the paper feeding.

8) <u>LOAD PAPER and LOAD RIBBON</u>
The Load Paper button is pressed before loading paper to allow the top of a form to be lined up with a predetermined mark on the lower module cover or on the left hand tractor. It is used in conjunction with the Load Paper digit in the paper feed loop.
The Load Ribbon button is used to facilitate loading the ink ribbon by filling the inner ink ribbon mandril with ribbon until all the ribbon is on that mandril.
These two buttons and indicators are normally lit Yellow and are only effective when the printer is disengaged.

The V-store Digits

Peripheral V-store Type 4. (*60041000)

Line	Digits		
0	36	W1	Clear Core Store
	35	W1	End of Line
	34	W1	Information Strobe
	33-28	W1	Information
	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Paper Warning Write: Put out Overflow Look At Me’s
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage

Type 14. (*60043430)

Line	Digits		
3	24	R	Overflow Look At Me
	28	R	Print Look At Me

The pin allocations for the above signals are given in Section 8.1.3

Signals to and from the Computer

To engage the printer it is necessary to press the Engage button; it is not possible for the printer to be engaged directly by the computer. Pressing the Engage button resets the Disengaged digit in the V-store (digit 24) to read as zero provided that neither a Disabled nor a Paper Out signal is being given by the printer. If either of these signals is present pressing the Engage button has no effect. The Disengaged digit is examined by the peripheral supervisor during every "One Second Interrupt" and the printer may be started if it is engaged.

The printer's buffer store is not automatically cleared when the Engage button is pressed and hence it is necessary for the computer to send a "Clear Core Store" signal (digit 36) before any information is transferred. The buffer is subsequently cleared automatically whenever a line of information is printed.

The printer may be disengaged either (a) by the computer writing a one to the Disengaged digit or (b) by an operator pressing the Disengaged button or (c) (temporarily) by an operator pressing the Top of Form button.

After the Start signal has been sent from the computer (digit 27), a Print Look At Me signal is sent when the printer is fully operational. This indicates that the transfer of information from the computer to the printer's core store (via the V-store, digits 33-28) can take place. The characters are sent one at a time and may not be sent more frequently than one every 16μsecs. There is no maximum time during which the characters for a line are sent to the V-store and hence the Put Out Look At Me (digit 26) must be sent as soon as the computer has recognised the Print interrupt. This thus enables the computer to deal with other interrupts if necessary during the transfer of information. Simultaneously with sending each six-bit character to digits 33-28 a further bit must be written to the Information Strobe digit (digit 34). This enables the printer to recognise information and control signals.

When a complete line of characters has been transferred (this might be any number of characters up to a maximum of 120) an "End of Line" signal (digit 35) is given together with, or followed by, a Vertical Format character. The latter is written to the Information digits and a one is also written to the <u>Information Strobe</u> digit. After the line has been printed and whilst the specified paper feeding is taking place a further Print Look At Me signal is sent to the computer, thus enabling further information to be put in the core store during the paper feeding operation. There is a delay of approximately 8 m.s. after sending the Vertical Format character before the next Print Look At Me is given. This is to enable the information in the buffer core store to be transferred to the electronics. No further Look At Me signal is sent if the Start/Stop flip flop has been set to Stopped. This may have been done either (1) by the computer writing a one to the Stop digit (digit 25) because no further information is available for printing or (2) by the printer being disengaged or (3) because a Disabled or Paper Out Signal has been given by the printer.

During the transfer of information the buffer automatically scans the characters and printing starts with the first available appropriate character. When all the characters have been printed the line feeding commences and a further Print Look At Me signal is sent to the computer. This Print Look At Me is given about 8msecs after the printer receives the Vertical Format character. The transfer of information for the next line thus takes place during the line feeding and further printing may start with the first available character after the line feed has finished.

A Print Look At Me is sent to the computer only if all the following conditions are satisfied:

(a) the system is "Able" i.e. the printer is switched on, all fuses are intact and paper is loaded.
(b) the printer is engaged.
(c) the system has received a "Start" signal from the computer. Only when conditions (a) and (b) above are fulfilled can the computer switch the system into the "Started" condition.
(d) the system is not "Print busy". The printer starts the printing cycle immediately it receives an "End of Line" or a "Clear Core Store" signal from the computer. It remains "print busy" in the former case until the core store has transferred its contents into the mechanics and has emptied itself which can take a

maximum of 60msecs. If a "Vertical Format" character (essentially an Information Strobe) is not received within this time the printer remains in the print busy state until one is received. After a "Clear Core Store" signal the printer is "print busy" until the core store has been reset to zero which takes about 720μsecs.

(e) the system is not "Vertical Format busy". The paper feed cycle commences immediately a Vertical Format character is received and the system remains "Vertical Format busy" until the printer 'receives the appropriate stop signal from the paper feed mechanism. The paper feeding does not commence whilst the system is "print busy" and it takes about 7msecs to come to a halt after the stop signal. This latter period is masked on a non-zero vertical format command by an 8msecs "Print Interlock" which prevents the beginning of the next print cycle until this time has elasped. (This is also the period in which an information transfer for a line of print from the computer must be completed to obtain full speed working).

The total time of the electrical Feed Paper command takes about 8msecs (between 6.2 and 8.2) for one line plus 6.67msecs for each line succeeding the first i.e. a paper feed rate of 150 lines per second. There is a "Vertical Format interlock" of 20msecs at the end of the "Vertical Format busy" state which prevents further paper feeding being initiated within this period.

The Print Look At Me is thus generated

(i) within 2μsec if conditions a, b, d and e are all satisfied and the printer receives a "Start" signal.

(ii) 720μsecs after receiving a "Clear Core Store" signal.

(iii) between 720μsecs and 60msecs (Print cycle time) plus 8+(n-1) 6.67msecs (Line feed time) after receiving an "End of Line" signal where n is the (non-zero) number of lines contained in the Vertical Format character sent with or following the End of Line signal. The minimum time of 720μsecs is obtained for the "printing" of all spaces and zero line feed. The maximum is 60msecs (whenever all the 64 different characters on the barrel are to be printed) plus the maximum number of lines to be fed which is the number of lines on a complete form (6 lines to the inch). In all other cases it is difficult to predict the print cycle time as it not only depends on the number of different characters, and their distribution round the print barrel, in the line to be printed but also on the exact location of the print barrel at the time the print cycle starts.

If the printer is disengaged by push-button during a transfer of information to its core store, the transfer is continued until completed. Before an "End of Line" signal is given the fixed store program examines the Disengaged digit and only sends the signal if the printer is Engaged.

Otherwise the information is retained in the core store buffer until the printer is re-engaged. Disengaging the printer does not inhibit any mechanical actions. Hence if the printer is disengaged in the time between the fixed store program reading the Disengaged digit as zero and sending the End of Line signal the line feeding still takes place.

If a fault condition is detected by the fixed store program during the transfer of a line of information the characters already copied to the core store can be cleared by writing a one to the "Clear Core Store" digit. A further Look At Me signal is given by the printer when the core store has been cleared and the transfer can be restarted.

A <u>Disabled</u> signal is given by the printer (digit 27) and the Stop/ Start flip-flop set to Stopped if a failure (fuses blown or. D.C. not ready) is detected. The printer is disengaged by the fixed store program at the next "one second interrupt" and the digit is reset automatically when the fault has been corrected. The printer cannot be re-engaged whilst the fault condition exists.

The <u>Paper Warning</u> signal is given (digit 26) if the Paper Low light detects that there is no paper in the bottom container or if the Throat Cover over the print head is open but the printer is not stopped in this case. This digit is reset automatically when more paper is loaded.

This digit is also set, and the printer is stopped, if the Paper Out microswitch operates because the paper is torn. The fixed store program disengages the printer at the next one second interrupt. The digit is automatically reset when more paper is loaded. The printer cannot be re-engaged before the Paper Out condition is corrected.

An "Overflow Look At Me" signal is given if the computer sends more than 120 characters for a line. This interrupt occurs even if the printer is stopped or paper feeding and is given within about 8μsecs of the 121st character having been sent to the V-store. On entering the Overflow Interrupt routine the fixed store program writes an End of Line Digit and a Vertical Format character to the V-store thus initiating the printing of the first 120 characters which were sent to the core store for that line. The fixed store program can then send this 121st character to form the first character of a new line on the next Print Look At Me interrupt. The Overflow Look At Me is reset by writing a one to digit 26 as for the Print Look At Me.

If the printer is disengaged when the Overflow Look At Me is set the printer becomes busy until either a Vertical Format character (i.e. an Information Strobe) or a Clear Core Store signal is received or the Test Print button is pressed. The Look At Me is automatically reset.

<u>Paper Feeding.</u>

The paper feeding may be specified by using the Vertical Format character in either of two ways

(i) in conjunction with a paper tape loop or

(ii) by specifying in binary the number of lines required to be thrown.

In the first case the three most significant digits of the vertical format character must all be ones. The three least significant digits specify in binary the channel of the vertical format control which is required. This format control is situated in the printer and consists of a brush-read tape reader which, for Atlas, reads standard 7-channel paper tape. The loop of paper tape has the same number of characters as the form under the print head has lines and for every line feed which takes place on the printer the paper tape loop is advanced by one character. When one of the channels of the paper tape loop is specified by the three least significant digits of the vertical format character the form is thrown until a hole is sensed in that channel of the paper tape, or until the next top of form digit. One channel is reserved for the "top of format" signal (required by the Paper Run Out button) and this is always channel 1 i.e. character 111001. In other applications of the printer where eight channel paper tape is used the character 111111 is used to specify channel 7. For Atlas, using seven Channel tape this character is not applicable.

The layout characters are thus

111000	Channel 0	load paper
111001	Channel 1	top of form
111010	Channel 2	
111011	Channel 3	
111100	Channel 4	
111101	Channel 5	
111110	Channel 6	
111111	not applicable	

The second method of specifying paper feeding is where one or more of the three most significant digits of the vertical format character is a zero. In this case the number of line feeds is given by the binary interpretation of this character. The number of lines which can be fed by this means varies from 0 to 55, i.e.

000000	no line feed
000001	single line feed
000010	two line feeds
.	.
.	.
.	.
110111	55 line feeds

The character 000000 suppresses line spacing to allow overprinting. The line fed is terminated if a top of form digit is detected in the paper feed loop before the specified number of line feeds has been completed.

The first information character written to the core store after a vertical format character is printed in column one irrespective of where the printing for the last line finished.

Details of Signals

The signals to the printer from the computer conform to the following specification:

a) two wire inputs for use with sub–miniature co–axial cable of 50 ohm impedance and approximately 30 pF per foot.

b) Levels:

Levels:	logic 1	–1 volts minimum
		–1.5 volts maximum
	logic 2	+1.5 volts minimum
		+2 volts maximum

c) pulse width a minimum of 2µsecs at the peripheral end with edges at 4.5 volts / µ sec. measured at 10 per cent and 90 per cent of peak to peak amplitude in both directions.

d) maximum D.C. loading 1 m. A at the peripheral end. Six lines carry the 6 bit character transfers at a rate not exceeding 62.5 Kc/s.

The signals from the printer to the computer conform to the following specification:

a) two wire outputs for use with sub–miniature co–axial cable of 50 ohm impedance and approximately 30 pF per foot.

b) Levels

Levels	Logic 1	-2	minimum
		-4.5	maximum
	Logic 0	+2	minimum
		+4.5	maximum

c) edge speeds are 4.5 volts per µsec. when measured at 10 per cent and 90 per cent of the peak to peak amplitude in both directions when loaded with 3000 pf.

d) the maximum current to be supplied to the cable is 10 m.A.

e) the reference level 0 volts must be floating, i.e. insulated from mains ground.

Installation Details

The approximate dimensions of the control pedestal and printer unit are:

Width 58 inches
Depth 30.5 inches
Height 55.25 inches

Total weight is approximately 1900 lbs.

The power supply is single phase A.C. at 50 c.p.s. and 240 volts ± 15 per cent. The power consumption of the printer pedestal is 2.0 KVA approximately.

Character Lay-out

The following characters are on the wheel in the sequence given:

Seq-uence	Internal code (Octal)	Octal code sent to printer	Character	Seq-uence	Internal code (Octal)	Octal code sent to printer	Character
-	01-I	40	Space	32	52-I	00	J
1	20-I	01	0 Zero	33	53-I	41	K
2	34-O	02	½	34	54-I	42	L
3	21-I	43	1 One	35	55-I	03	M
4	36-I	04	- Minus	36	56-I	44	N
5	22-I	45	2	37	57-I	05	O
6	35-I	46	+	38	60-I	06	P
7	23-I	07	3	39	61-I	47	Q
8	37-I	10	. Point	40	62-I	50	R
9	24-I	51	4	41	63-I	11	S
10	03-O	52	£ Pound Sterling	42	64-I	12	T
11	25-I	13	5	43	65-I	53	U
12	15-I	54	&	44	66-I	14	V
13	26-I	15	6	45	67-I	55	W
14	10-I	16	(	46	70-I	56	X
15	27-I	57	7	47	71-I	17	Y
16	11-I	20	)	48	72-I	60	Z
17	30-I	61	8	49	17-O	21	:
18	16-I	62	* Asterisk	50	40-I	22	‘ Apostrophe
19	31-I	23	9	51	21-O	63	[
20	17-I	64	/	52	22-O	24	]
21	35-O	25	10 Ten	53	32-I	65	<
22	36-O	26	11 Eleven	54	33-I	66	>
23	41-I	67	A	55	34-I	27	= Equals
24	42-I	70	B	56	26-O	30	_ Underline
25	43-I	31	C	57	27-O	71	| Vertical line
26	44-I	32	D	58	14-I	72	?
27	45-I	73	E	59	12-I	33	,
28	46-I	34	F	60	30-O	74	^{2} Super script 2
29	47-I	75	G	61	13-I	35	Π Pi
30	50-I	76	H	62	32-O	36	α Alpha
31	51-I	37	I	63	33-O	77	β Beta

8.4. The I.C.T. Card Punch

General

This equipment comprises an I.C.T. type 582 Card Punch and associated electronics. The punch handles 80–column cards at the rate of 100 cards per minute. Cards are fed broadside on, face down, with the Y edge leading, thus punching one row at a time. The card track comprises magazine, punching station, check reading station and stacker. The card capacities of the magazine and stacker are approximately 700 and 800 cards respectively.

Reading brushes are placed one card cycle after the punch knives. It takes four card cycles for a card to pass from the magazine to the stacker, the sequence being.

1st cycle, the leading edge of the card approaches the punch knives.
2nd cycle, the card passes through the punch station.
3rd cycle, the card passes under the read brushes.
4th cycle, the card passes into the stacker.

On a "Start" (or "Feed Card") order a card is always fed to the punch knives. If there is no card already approaching the punch knives two cycles are automatically carried out and two cards taken from the magazine. The first card is punched and the second card left with its leading edge approaching the punch knives. Cards are fed continuously, one per card cycle, until a Stop signal is given.

A plugboard is provided which allows any of 80 computer channels to be connected to any of the 80 punch magnets. Not more than two magnets may be driven from one channel. A similar distribution of the 80 check–reading brushes is possible. It is not anticipated that this facility will be used on Atlas except for maintenance purposes.

Provision is made for off–setting a card in the stacker by about 3/8 inch. Thus if the information read at the checking station does not agree with the information that should have been punched the fixed store program gives an "Offset" order and punches the information on a second card. The operator removes all off–set cards when unloading the stacker.

No limitation is placed on the number of holes that can be punched in a card.

The punch may be used off-line as a gang punch, by changing the plug–board and switching to local control. This is intended as a maintenance rather than an operational facility. For this reason the plugboard also carries the twelve outlets for an emitter, each corresponding to one row of a card.

The V-store Digits

Type 9

line	Digits		
0 (*60042200)	47-40	RW10	Information. One digit of each of columns 8-1 respectively where column 1 is that first in the card punch. Read: Brush Information Write: Punch Information
	39	RW1	Read: Card Levels Write: Put Out Brush Look At Me
	38	RW1	Read: Card Wreck Write: Reset Overdue
	37	R	Overdue
	29	W1	Do not Offset
	28	W1	Put Out End of Card Look At Me
	27	W1	Start
	26	W1	Put Out Punch Look At Me
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage
4 (*60042240)	47-24	RW10	Information (see line 0). One digit of each of columns 32-9 respectively
8 (*60042300)	47-24	RW10	Information (see line 0). One digit of each of columns 56-33 respectively
12 (*60042340)	47-24	RW10	Information (see line 0). One digit of each OF columns 80-57 respectively

Type 14 (*60043460)

line	Digits		
6	24	R	End of Card Look At Me
	26	R	Punch Look At Me
	28	R	Brush Look At Me

Controls and Indicators

a) Engage Push-button and Indicator

It is only possible for the punch to accept information from the computer if it has been Engaged and it can only be Engaged by an operator pressing the Engage push-button. It is impossible to Engage unless the punch is able to deal with information from the computer, e.g. unless there are cards in the magazine, there is no card wreck, the receiver is not full, the fuses are intact etc.

Combined with the push-button there is an indicator which lights Green when the equipment is Engaged and White when the punch is Disengaged.

b) Disengage Push-button and Indicator

The punch may be Disengaged by instruction from the computer, or by an operator pressing the Disengage push-button.

When the button is pressed a Disengaged signal is sent immediately to the peripheral co-ordinator setting the appropriate V-store digit. The equipment is Disengaged immediately if it is not currently punching (or check-reading) otherwise it is disengaged automatically at the end of the current cycle.

Combined with the push-button is an indicator which lights Red when the equipment is Disengaged and White when the punch is Engaged.

c) Start Push-button and Indicator

When, and only when, the equipment is Disengaged is it under local control. Pressing the start button then causes the punch to cycle, feeding cards if there are any.

In this condition the equipment may be used as a gang-punch, provided a suitable plugboard is used, or the facility may simply be used to clear the card tracks.

Combined with the push-button is an indicator, coloured White, which lights while the machine is performing a cycle whether under computer or local control.

d) Stop Push-button

Pressing this stops the punch cycling only if it is under local control, i.e. Disengaged. If the punch is Disengaged and cycling it is only possible to Engage it after the Stop button has been pressed.

e) Power On/Off Switch and Indicator

There is a power on/off switch on the punch. When the switch is on the Power On indicator, coloured Green, lights up.

f) Magazine Empty Indicator

g) Stacker Full indicator

h) Card Wreck Indicator

These are indications that the punch is disabled until it receives attention. All are coloured Yellow.

Signals between the Computer and the Punch

During normal operation three separate causes of interruption may occur. These are

i) when a card is ready to have a row punched

ii) when a row of the card has been read at the check-read station

iii) when an End of Card signal is received

The digits referred to below are the digits of line 0 of the Card Punch V-store.

a) Disengage (digit 24)
This digit is read as a one when the card reader is Disengaged, and read as a zero when it is Engaged. It is reset to zero only by the operator pressing the Engage button on the card reader.
It is set to one either when the computer writes a one to it or when the Disengage button is pressed. In this latter case the punch is not stopped and disengaged until after the current cycle has been completed although the Disengaged digit in the V-store is set immediately. The punch must be Engaged before any cards can be fed.

b) Start (digit 27)
Writing a one to the Start digit initiates the feeding of cards. This signal starts the motor and when it has run up to a stable speed, taking up to 800 m.sec., feeds a card to the punch knives. If there is no card in the track waiting to pass the punching station the machine automatically performs a second cycle. Cards are then fed continuously until a Stop signal is given either from the computer or from the punch itself. This latter may be either due to a fault indication or because the punch has been disengaged.

c) Stop (digit 25)
Writing a one to the Stop digit inhibits all further Look At Me's and prevents card feeding. A one must be written to the Stop digit within 12msecs of the End of Card Look At Me otherwise a further card is fed. At the end of punching a pack of cards two blank cards are in the card track, one about to pass under the check read station and the other about to pass under the punch knives.

The Stop signal may also be given by the punch if either

(i) the Disengage button is pressed or

(ii) either Operator or Engineer attention is required.

In these cases the Stop digit is set and the punch stopped 2msecs after the next End of Card signal. The motor is automatically run down if the Stop digit remains set for more than 400msecs.

For mechanical reasons it is not possible to re-engage the clutch whilst it is in the process of disengaging. This effectively means that the punch cannot be started again between 10msecs and 1 sec. after a Stop signal has been given to it. A delay is therefore incorporated to hold-off any Start signal sent during this period.

d) Put Out Punch Row Look At Me (digit 26)
An automatic interrupt occurs whenever the card is ready to have information punched on to a row. These signals occur every 43msecs and are present for about 35msecs of this time. If the signal is not put out in this time an Overdue digit is set, the Stop/Start flip-flop set to stopped and no further interrupts are given.

e) Put Out End of Card Look At Me (digit 28)
The End of Card interrupt occurs simultaneously with the last row interrupt from the Check Read (Brush),

station, but has a lower priority than the Brush interrupt. If no card is in this station it occurs at a similar time in the cycle for the card at the punch station.

The fixed store program sets the Do Not Off-set digit if the card just checked has been punched and checked correctly. The Stop signal is also given at this time if it is required to terminate the feeding of the cards. In order to check the last card in a pack it is thus necessary to have an extra card passing under the punch knives whilst this last card is being checked. No holes are punched in this extra card.

f) Put Out Brush Look At Me (digit 39)

The Brush Look At Me interrupt for the card at the reading station occurs 40msecs after the Punch Row interrupt for the corresponding row of the card at the punch station. If the information has not been read and the Put Out Brush Look At Me signal been given within 10msecs of the interrupt occurring the Overdue digit is set, the punch is stopped and no further interrupts occur.

g) Do not Off-set Card (digit 29)

This digit is automatically reset to zero to indicate "Off-Set Card" at the time the first row signal should be given by the punch station whether or not a card is actually passing through the punch station. It is set by program to "Do Not Offset Card" when the End of Card signal is received if the count of the rows for both stations has been correct, no overdue signals have been received and if the information read by the checking station was the same as that sent to the punch station. This digit is examined 9msecs, after the End of Card signal and the card off-set if the digit has not been reset by the program.

h) Card Levels (digit 39)

This is set, and the Stop/Start flip-flop set to Stop, two msecs after the next End of Card signal is given if Operator attention is required because either

(i) the card magazine is empty or

(ii) the card receiver is full.

The digit is reset automatically when the card level in question has been attended to.

i) Card Wreck (digit 38)

This is set and the reader stopped 2msecs after the next End of Card signal if Engineer attention is required because a card wreck, jam or misfeed has occurred. The appropriate digit is reset automatically, provided the fault condition has been corrected when the Engage button is pressed.

j) Overdue (digit 37)

This is set and the punch stopped immediately if the appropriate Put Out Look At Me signal is not sent either:

(i) within 35msecs of a Punch Look At Me

(ii) within 10msecs of a Brush Look At Me

(iii) within 10msecs of an End of Card Look At Me

The Overdue digit is reset either when the Engage button is pressed or by writing a one to digit 38.

In each of the above three cases (i.e. either digit 39, 38, or 37 set and the reader stopped) the reader is disengaged by fixed store program.

8.5 Paper Tape and Teleprinter Output

8.5.1. The Teletype Punch

The Teletype punch operates at up to 110 characters per second punching either five or seven channel paper tape. There is no facility, e.g. switch or push button, for the operator to convert a punch from five to seven channel operation, and vice versa, but the change can be made by an engineer by the repositioning of a package and by changing the position of the tape guide. A maximum of twelve teletype punches is permissible and of these, punches 0 - 7 always operate in the seven channel mode and punches 8 - 11 always operate as five channel punches.

The V-store Digits

Type 8

line	digits		
0	35-29	W1	Information, seven channel punches
(8	33-29	W1	Information, five channel punches)
	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Tape Warning Write: Put Out Look At Me
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage

Type 14

8	24	R	Punch Look At Me

To Engage a punch it is necessary to press the Engage button on the punch; it is not possible to engage a punch directly from the computer. Pressing the Engage button resets the Disengaged digit in the V-store to zero (digit 24 of the appropriate line of Type 8). When the Supervisor wishes to output on a punch it examines the Disengaged digits of the punches not already in use and if one of these digits is read as a zero that punch is used for the output. If all the punches not in use are disengaged the Supervisor instructs the operator to engage one. The disengaged digits are examined during the "one second interrupts" and if a punch is subsequently engaged the output is commenced. If one of the other punches, i.e. one of those previously being used, becomes available before a further one is engaged by the operator this is used for the output.

After a punch is engaged it is necessary to write a one to the Start digit (digit 27) before sending characters to be punched. This digit remains set until the Stop digit (digit 25) is set which may be done either

a) by writing a one by programme to the Stop digit
b) by writing a one by programme to the Disengage digit
c) by pressing the Disengage push-button

The character to be punched is held in the V-store until the appropriate time in the punch cycle for it to be punched. The V-store digits are then cleared and a further Look At Me signal given.

If the Stop digit is set, either from the computer or by pressing the Disengage button, no further Look At Me signals are given. The Look At Me signal may be given between 5μsecs and 9msecs after the previous character has been punched.

Therefore to prevent a further Look At Me being given after the last character in a block of output has been punched the Stop command must be given within 5μsec of copying the character to the V-store and giving the Put Out Look At Me signal. In effect this means that the Stop signal must be given simultaneously with the last character.

If the Disengage button is pressed after a Look At Me signal has been sent to the machine this interrupt is dealt with in the normal way and the character sent to the V-store is punched. The setting of the stop digit is detected by the Supervisor in the next One Second Interrupt.

If the stop digit remains set for about 30secs the motor is automatically run down.

Digit 27 (Disabled) is read as a one and the Stop/Start flip-flop is set to Stop if a fault (power off only) is detected by the punch. The punch is disengaged by fixed store program. The digit is reset automatically when the fault has been corrected. It is not possible to re-engage the punch whilst the fault still exists.

A <u>Tape Warning</u> digit (digit 26) is set in the V-store when the diameter of the tape reel is such that only 40 feet of tape remains. The setting of this digit depends on a mechanical arrangement and obviously the amount of tape remaining when it is first set cannot be guaranteed to within a few feet. This digit is reset automatically when the contact is broken.

The same digit is set and the Stop/Start digit set to Stop when a Tape Out or Tape Torn condition is detected. In this case the punch is disengaged by fixed store program. The digit is reset automatically when the fault condition is corrected. The punch cannot be re-engaged whilst the fault condition exists. The punch may stop, depending on when the warning is given, either with the character resulting from the last Look At Me stored but not punched or having punched the last character. In either case it is essential that the fixed store program is able to restart the output at some earlier point.

A <u>Run Out</u> button on the punch outputs blank tape (figure shift) for five channel tape and the three least significant bits of a character (upper case) for seven channel tape. If the run out button is pressed at a time when the punch is operating the character resulting from the last Look At Me is either

a) stored but not punched or
b) punched.

In case a) the character is punched after the run out button is released, the Look At Me also being set.

Normal output is resumed when the run out button is released, the fixed store program not being aware of the run out characters having been inserted in the output.

The location hole is punched between digits 31 and 32. The bit sent to digit 29 is always punched in the outside channel on the 3-channel side of the tape. For five channel tape the bit sent to digit 33 is punched in the outside channel on the two channel side of the tape. For seven channel tape the bit sent to digit 35 is punched on the outside channel of the four channel of the tape. For correct punching a seven channel teletype can only be connected to one of the positions 0 - 7 of type 8 and a five channel teletype can only be connected to one of the positions 8 - 11. The fixed store program determines from its address whether a teletype is seven or five channel.

8.5.2 The Teleprinter

This is a Creed Model 75 teleprinter which operates at 10 characters per second. It has six information channels with all its characters on a single shift. The characters available are:

Space	0	, (comma)	P
* (asterisk)	1	a	Q
N.L.	2	b	R
(red)	3	c	S
(black)	4	d	T
Π (pi)	5	e	U
] (close square brackets)	6	f	V
[(open square brackets)	7	g	W
(	8	h	X
)	9	i	Y
<	α (alpha)	j	Z
>	β (beta)	k	?
:	½	l	=
_ (underline)	+	m	‘ (prime)
| (vertical bar)	-	n	& (and)
/ (oblique stroke)	.	o	ER (erase)

The V-store digits

Type 11

line	digits		
0	35,24,32-29	W1	Information
	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Paper Warning Write: Put Out Look At Me
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage

Type 14

line	digit		
1	24	R	Print Look At Me

The operation of the teleprinter is the same as that for the Teletype punch described in the previous section. Digit 33 in line 0 corresponds to the parity digit in the seven-channel Flexowriter code and is not used in the teleprinter code. This enables the teleprinter and a seven-channel Teletype punch to be interchanged provided that the character sets being used are common to both equipments.

A delay is built into the teleprinter to prevent printing during the Carriage Return/Line Feed operation.

8.5.3. Creed 3000 Paper Tape Punch

This punch operates at up to 300 characters per second punching seven-channel tape. The punch can also be set up to punch 5, 6 or 8 channel tape although for punching 5-channel tape a different block is required.

The V-store DIGITS

Type 6

line	digits		
0	35-29	W1	Information, seven channel punches
	28	RW1	Read: Check Fail Write: Reset Check fail
	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Tape Warning Write: Put Out Look At Me
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage

Type 14

line	digit	
14	24	Punch Look At Me

The Creed 3000 punch is operated in a similar manner to the Teletype punch except that it also contains a Check Read station. This reads a character by means of photocells three characters after it has been punched. This character is compared, within the punch, with the character expected and the Check Fail (digit 28) and Stop signals are sent in the event of a discrepancy. This digit is reset by writing a one to the Reset Check Fail digit (also digit 28).

Digit 27 (Disabled) is set to a one and the Stop/Start flip-flop set to Stop if a failure of the punch occurs e. g. a power failure or one or both of the covers are open. The digit is reset to zero automatically when the fault condition is cleared.

The location hole is punched between digits 31 and 32. The bit from digit 29 is punched in the outside channel on the 3-channel side of the tape and the bit in digit 35 is punched in the outside channel on the 4-channel side of the tape.

8.6 Paper Tape Input

8.6.1. The TR5 Paper Tape Reader

The TR5 operates at up to 300 characters per second reading either five or seven channel tape. There is a Honeywell double-action switch and indicator on the reader to select whether five or seven channel tape is to be read. This button must be pressed to change the mode of reading. The upper half is lit (Yellow) when reading seven-channel tape and the lower half is lit (blue) for five channel tape. If this is set to read five channel tape the digits in the other two channels are always read as zero. There is also a mechanical guide which must be placed appropriately for either five or seven channel reading. This guide is independent of the push-button.

The V-store Digits

Type 7	(*60041600)		
line	digits		
0	34	R	Overdue
	33	R	5/7 switch (1 if 5-channel)
	32-26	R	Information
	28	W1	Reset Overdue
	27	W1	Start
	25	W1	Put Out Look At Me
	26	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage

Type 14	(*60043530)		
11	24	0	Character Look At Me

To Engage the reader it is necessary to press the Engage button; it is not possible for the reader to be Engaged directly by the computer. Pressing the Engage button resets the Disengage digit in the V-store to zero. This digit is examined by the peripheral supervisor during the "one second interrupt" programme and the reader may be started if it has been Engaged. The reader can be disengaged either by writing a one to the Disengage digit from the computer or by pressing the Disengage button. In either case the tape is stopped with a character stored in the information digits in the V-store. When the reader is engaged and started again a Look At Me interrupt occurs immediately and this character is transferred to the computer. The Supervisor can distinguish between when a reader is newly engaged and when it has been disengaged and re-engaged (e.g. because the tape is twisted and requires operator attention) because in the latter case the input has been terminated without any appropriate "end of Tape" characters having been read.

After the Start signal has been sent to the reader, characters are read at the rate of 300 per second to the information digits (26-32) of the appropriate V-store line. On both 5 and 7 channel tape the location hole is between digits 28 and 29, digits 31 and 32 being zero for 5-channel tape. In both cases the outside digit on the 3-channel side of the tape is read to digit 26. For 7-channel tape the outside bit on the 4-channel side of the tape is read to digit 32 and for 5-channel tape the outside bit on the 2-channel side of the tape is read to digit 30.

The Look At Me digit is set, causing an automatic interrupt, as the sprocket hole is passing under the reader. The character is available in the information digits from the time the Look At Me digit is set coinciding with the “late location hole” signal to the time the “early location hole” signal for the next character is received. The time between the “early” signal, when the information digits are reset, and the “late” signal, when the next character is available, is about 2µsecs. The Look At Me normally remains on until put out by the fixed store programme.

8.6.2 The T.R.7 Paper Tape Reader

General

The T.R.7 operates at up to 1,000 characters per second reading either five or seven channel tape. Automatic spooling is provided to facilitate the handling of long tapes although it is possible to use the reader without using the automatic spooling. A reel of tape is loaded onto the right hand capstan and is taken up on the left hand spool. A fast rewind facility enables an 1,100 ft. tape to be rewound in approximately 30 seconds. If the automatic spooling is being used the first and last ten feet of the paper tape must be blank (or not containing relevant punching) i.e. tapes which are less than twenty feet in length must be read without using the spools. Tapes which are longer than twenty feet may be used either with (in which case there must be the ten feet of blank tape at either end) or without the spooling facility. If the spooling facility is not used then the reel of tape is dropped into the right-hand trough of the reader. The tape which has been read is passed out through the left-hand side of the reader and can be collected in a tape bin. In this case there must be twenty inches of blank tape at the beginning of the reel.

The V-store digits

Peripheral Line	V-store, Type 2 (*60040400) Digits		
0	36	R	Disabled
	35	R	Tape Warning
	34	R	Overdue
	33	R	5/7 Channel switch (1 for 5-channel)
	32-26	R	Information
	30	W1	Stop Rewind
	29	W1	Rewind
	28	W1	Reset Overdue
	27	W1	Start
	26	W1	Put out Look At Me
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read:Disengaged Write: Disengage
Type 14	(*60043640)		
20	24	R	Look At Me

Push-Buttons and Indicators

There are nine buttons and/or indicators on the front on the T.R.7.Eight of them are grouped together in the centre of the reader and the other (Mains On/Off) is placed separately. The purposes and effects are given below together with the colours of the lights behind the buttons which are lit to indicate the relevant states.

1) Mains ON/OFF

This is a double action button and pressing it changes its state from On to Off or from Off to On as appropriate. When switched to On, it switches on the power supplies, the cooling fans, the reading head light and the pilot lights (used by the servo system) in the tape troughs. Its colour is Green when the reader mains are switched on.

2) Engage

This is a single action button and when pressed the following actions take place:-

a) the tape is clamped in the region of the reading head,
b) the drive-unit motor is switched on,
c) the Green light comes on immediately
d) the complete servo-system becomes operational if the spooling facility is being used i.e. the reader is in the "with spooling" mode,
e) the new state of the Engage flip-flop is sent to the computer setting digit 24 of the appropriate V-store line to zero to read as Engaged, after about five seconds. This delay is so that if the "With Spooling" mode is being used this signal is sent only after the tapes in the troughs have been servoed into the correct positions. This button is lit White when the reader is disengaged and Green when it is engaged. It remains Green if the computer initiates a Rewind without also disengaging the reader.

3) Disengage

This is a single action button and when pressed the following actions take place:-

(a) the drive-unit motor is switched off,
(b) if in the "With Spooling" mode the servo system is switched off, the spools being held stationary. The suction fan remains running, this being switched off during "Reload" or "Without Spooling" conditions.
(c) a disengaged signal is sent to digit 24 of the appropriate line of the V-store immediately.

If the computer sends a Disengage signal or the Disengage button is pressed at the same instant as the Engage button is pressed the reader takes up the Disengaged condition. If the reader is disengaged during the delay period between pressing the Engage button and the Engaged signal being sent the normal action taken on disengaging the reader occurs and no Engaged signal is sent to the computer.

This button is lit White when the reader is engaged and Red when it is disengaged.

4) 5 Track/7 Track

This is an alternate action switch which either switches out or in the two additional channels of information. The masking plate and the front panels of the suction troughs have to be adjusted manually. The top half of this button is Yellow when 7-track reading is selected and in darkness when 5-track reading is selected. The lower half is in darkness for 7-track reading and Blue for 5-track reading.

5) Started/Stopped Indicator

This is an indicator which is lit Blue when the reader is Started (i.e. the Start/Stop flip-flop reads Started) and is White when the reader is Stopped.

6) Rewind

This is a single-action switch which, provided that the reader is disengaged and in the "With Spooling" mode, when pressed initiates a fast rewind of the tape. The reader cannot be Engaged during a rewind operation. The tape is stopped a short distance from the beginning by means of a metallic strip attached to the tape and a pick-up arm. The tape can then be re-read without reloading. If it is not required to re-read the tape the metallic strip can be left off the tape and in this case the reader stops rewinding when all the tape has been rewound. The spool drive motors are switched off at the end of a rewind on a signal from the pick-up arm. This is given either on contact with the metallic strip or when it comes into contact with the hub of the spool due to the tape having been rewound completely. Pressing the button or attempting to initiate a rewind from the computer when a rewind is in progress has no effect. Attempting a second rewind after a previous one has been completed has no effect if the tape has already been wound off the left hand spool. If the first rewind were stopped by contact between the pick-up arm and the metallic strip a second rewind is performed which is stopped after the tape has been wound off the left hand spool.

A rewind will not start if the pick-up arm is in contact with the metallic strip.

This button is lit White when the reader is disengaged and in the "With Spooling" mode but is not rewinding (i.e. when a rewind may be initiated) and Red at all other times.

7. Stop Rewind.

This is a single action switch and, provided that the reader is rewinding, when pressed it stops the rewinding of the tape. For a rewind initiated from the computer without the reader being disengaged, pressing this button stops the rewind and also disengages the reader.

The button is lit White during a rewind and Red at all other times.

8. Reload/Ready

This is a single action switch which must be pressed before a tape can be loaded or unloaded. Pressing this button when the reader is engaged or is rewinding has no effect. Pressing the button when the reader is disengaged (and not rewinding) unclamps the tape at the reading head and switches off the suction fans to facilitate the appropriate loading or unloading operation to take place during the subsequent "Ready" (i.e. Ready for reload) period.

After a "Reload" operation the reader is made operational again by pressing either the "Engage" or the

"Rewind" button. (The latter case is only effective in the "With Spooling" mode).

A change from "With Spooling" to "Without Spooling" or vice-versa can only be done during the "Ready" period.

The top half of this button (Reload) is lit White when the reader is disengaged and not rewinding (i.e. when a Reload may take place) and in darkness at all other times. The lower half of the button is lit Blue (Ready) when the fans are switched off and the actual reloading operation can take place (the top half is still lit White at this time) and is in darkness at all other times.

9. With Spooling/without Spooling

This is an alternate action switch effective only when the reader is in the "Reload Ready" condition. When pressed it either switches on or off the servo system, in the latter case allowing tape to be read without using the spools.

The top half, With Spooling, is lit Blue when the servo system is on; it is lit White when the servos are off and the reader is in the Reload Ready state; otherwise it is in darkness.

The bottom half, Without Spooling, is lit Yellow when the servos are off; it is lit White when the servos are on and the reader is in the Reload Ready state; otherwise it is in darkness.

<u>Reloading</u>

The reader can only be reloaded if it is disengaged and not rewinding. This is indicated by the top half of the Reload/Ready button i.e. the top half is lit White when a tape can be loaded or unloaded. When the Reload/Ready button is pressed the bottom half lights up Blue to indicate that the fans are off, the tape is unclamped and the reader is Ready to be reloaded.

A tape is then reloaded as follows:

1) The pick-up arm is moved manually to its reload position,
2) the right hand spool is rotated in an anti-clockwise direction until all the tape has been wound onto it,
3) the right hand spool is removed and replaced by a new one. This new spool is fitted so that rotation in a clock-wise direction unwinds the tape,
4) the end of the tape is passed between the right hand roller and microswitch, under the clamping mechanism between the left hand roller and microswitch, and wound onto the left hand spool in such a manner that rotation of the spool in a clockwise direction winds the tape on. The tape is loaded so that the outside track of the 3 track side of the tape is the nearer to the reader for both 5 and 7 track tape,
5) the left hand spool is rotated until the metallic strip (if being used) on the tape is wound past the pick-up head,
6) the pick-up arm is released from its reload position manually,
7) the "With Spooling/Without Spooling" button is pressed if appropriate,
8) the Engage button is pressed. The light behind the lower half of the button (Ready) goes out. The green

light behind the Engage button and the White light behind the Disengage button come on immediately. The suction fans come on, the fan run-up relay is actuated and, when completed, the servo system is operated; after a delay of about 5 seconds an Engaged signal is sent to the computer.

Signals to and from the Computer.

To engage the reader it is necessary to press the Engage button; it is not possible for the reader to be engaged directly by the computer. Pressing the Engage button resets the Disengaged digit in the V-store (digit 24) to read as zero provided neither the Disabled digit nor the Tape Warning digit is set. If either of these digits is set pressing the button has no effect. The Disengaged digit is examined by the peripheral supervisor during every "One Second Interrupt" and the reader may be started, by the fixed store programme, if it has been engaged.

The reader can be disengaged either (a) by the computer writing a one to the Disengage digit or (b) by an operator pressing the Disengage button or (c) by an operator pressing the Stop Rewind button during the course of a rewind initiated by the computer. If in the first two cases, the Start/Stop flip-flop for the reader was set to Started a further character is read and stored in the information digits of the V-store before the reader is stopped and disengaged. When the reader is re-engaged and re-started a Look At Me signal is given immediately and this character is transferred to the computer.

After the Start signal has been sent to the reader characters are read at the rate of 1,000 per second to the information digits (26-32) of the appropriate V-store line until the reader is stopped. The sprocket hole on the tape is between the digits read to positions 28 and 29 i.e. the digit in the outside track on the 3-track side of the tape is always read to digit 26. For five track tape the digit on the outside track of the 2-track side of the tape is read to digit 30, and for seven track tape the digit on the outside track of the 4-track side of the tape is read to digit 32. If the 5-track/7-track switch is set to 5 Track (digit 33 reading as a one) digits 31 and 32 are always zero.

The Look At Me signal is sent, causing an automatic interrupt, as the sprocket hole is passing under the reader. The character is available in the information digits from the time the Look At Me is given, coinciding with the "late location hole" signal to the time the "early location hole" signal for the next character is received. The time between the "early" signal, when the information digits are reset, and the "late" signal, when the next character is available is about 5μsecs.

The Look At Me normally remains on until put out by the fixed store programme. If however, it has not been extinguished by the time the next character is due to be copied into the information digits an Overdue digit (digit 34) is set and the Start/Stop flip-flop set to Stopped. The tape comes to rest after reading this next character, the original character in the information digits having been cleared (and hence lost to the computer) and the new character written in. The setting of this Overdue digit is detected by the "One Second Interrupt" and the digit is reset either automatically when the reader is disengaged (either from the machine or by push button) or by the machine writing a one to the "Reset Overdue" digit (digit 28).

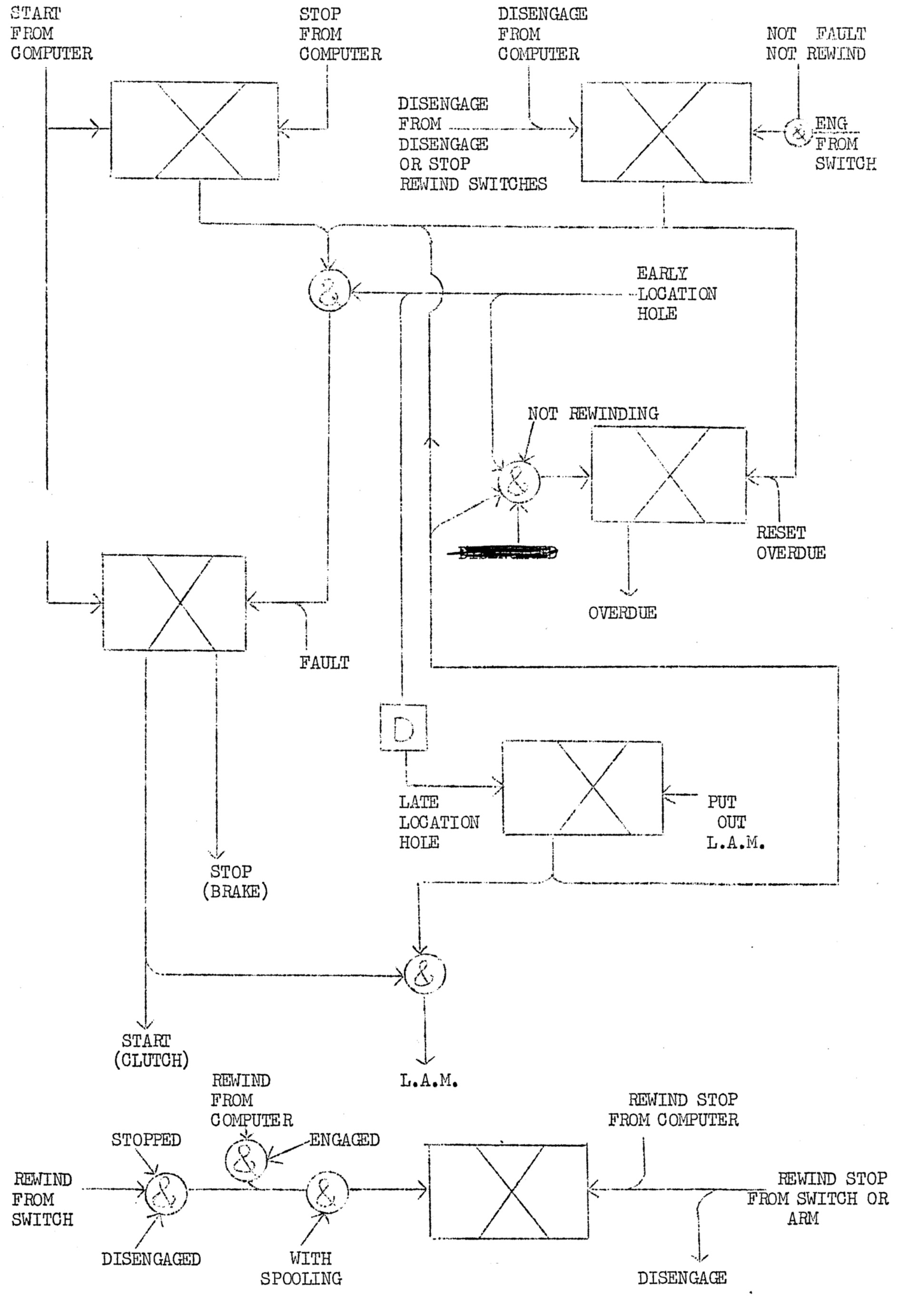
Digit 36 (Disabled) is set and the Start/Stop flip-flop set to Stopped if a fault condition is detected automatically by the reader. (The fault conditions are (a) fuses blown (b) door open). The reader is disengaged by fixed store programme at the next one second interrupt and the digit is reset automatically when the fault has been corrected. The reader cannot be re-engaged whilst the fault digit is still set.

Digit 35 (Tape Warning) is automatically set and the Start/Stop flip-flop set to Stopped if the tape runs out or is broken. This condition is only detected during the "With Spooling" mode of the reader. The reader is disengaged by fixed store programme during the next one second interrupt after digit 35 is set. A Tape Torn or Out condition can be detected either by a "tape tensioning" microswitch or by the failure to receive a character at the reading head within 200msecs of the last one when the reader is started. This digit is reset automatically either (a) when the fault is corrected if it was detected by a microswitch or (b) when the Reload button is pressed if the fault was detected at the reading head. The reader cannot be re-engaged whilst the fault digit is set.

There are two microswitches for detecting the tape out or tape broken conditions, one between the right hand spool and the reading head and the other between the reading head and the left hand spool. The pick-up arm associated with detecting the end of a rewind does not set the Tape Warning digit.

Digits 29 and 30 are used to control the Rewind facility from the computer. This is only effective when the "With Spooling" mode is selected and the reader is Engaged. (Rewind from the push button is only possible when the "With Spooling" mode is selected and the reader is Disengaged). When a rewind order is given from the computer the reader appears to the computer to be temporarily disengaged i.e. digit 24 reads Disengaged, but the Green light behind the Engage button and the White light behind the Disengage button both remain on. If the tape is moving the computer must send a Stop signal before initiating a rewind. During a rewind the characters on the tape are read into the V-store information digits as they pass under the reading head but no Look At Me signals are given (inhibited because the Stop/Start flip-flop is set to Stopped) and the state of the Overdue digit is not changed.

At the completion of the Rewind (beginning of tape or a Stop Rewind order given by the computer) the reader automatically resumes the Engaged state again. If however a Disengage command is sent to the reader from either the computer or the Disengage push button after the Rewind order the reader does not resume the engaged state again. If it is required to send both Rewind and Disengage commands to the reader the Rewind order must be sent first. If they are sent simultaneously the Rewind order may not be effective. If the "Stop Rewind" button is pressed during the course of a rewind initiated by the computer the rewind is stopped and the reader disengaged i.e. digit 24 remains set at one and the lights behind the Engage and Disengage buttons changed to White and Red respectively.



SIMPLIFIED LOGICAL DIAGRAM OF THE T.R.7

17.9.62.

SIMPLIFIED LOGICAL DIAGRAM OF THE T.R.7

8.7 Character Codes

8.7.1 A Description of the Internal Character Code

Atlas uses several different types of peripheral equipment each of which has its own character code. An internal character code has been chosen so that programs or data may be fed in on any relevant medium (seven channel paper tape i.e. Flexowriter code, five-channel paper tape, i.e. Mercury/Pegasus code, cards punched in I.B.M. code, N.E.P. tape, or I.B.M. magnetic tape) and appear in the machine in exactly the same form, the translation being done character by character while the tape or card is read. Similarly for output the conversion to the code for the specified equipment is done character by character. This means that a program which confines its output to the limited range of characters common to all equipments may send this output to any punch or printer. Only those programs which produce a wider range of characters have to specify the type of equipment to be used.

The characters available on the different equipments are divided into an "inner set" and an "outer set". As a six bit code is used a total of 128 different characters including the shift characters can be represented. With the exception of "<", "&", and "backspace", the inner set of characters is common to all output peripherals.

Input

During input each character is checked for correct parity (if on seven channel tape) or correct punching (if on a punched card) and converted to its internal code equivalent. Every permitted character on the input has one and only one equivalent in the internal code. The paper tape characters Figure Shift, Letter Shift, Upper Case and Lower Case are not normally stored in the internal code but their significance is noted by the fixed store program. The "Shift in" and "Shift out" characters of the internal code are inserted automatically by the fixed store program where appropriate. Redundant shifts, however, are stored (e.g. letter shift when on letter shift and upper case when on upper case) so that the main program can make its own checks against spurious characters.

The carriage control characters NL, CR, LF, Paper Throw, and also End of Card, are treated differently from the normal printing characters. Information in the internal code is stored in "records"; each record corresponding to a line of paper tape input or one card. When one of the carriage control characters is detected on input the count of the number of characters in the record is recorded, and the carriage control character is inserted at the end of the record. In effect, the six-bit carriage control character belongs to a third character set, distinct from the inner or outer sets; it is recognised only by being the last character in each record.

Each internal code record is assumed to start in the inner set, no matter how the previous record ended. If the first printing character in any record is in fact an outer set character, the input routine will precede it with the "shift out" character even though the previous record may have ended in outer set.

If an inadmissible punching on a card, or a parity error on paper tape is detected the character is replaced by the "Fault" character if the computer is operating in the mode where the programmer does not wish it to stop reading on detecting an error.

Output

On output each character is checked as being permissible for the particular equipment being used. If it is unsuitable it is replaced by the "." (full stop) character.

A programmer may include F.S., L.S., U.C., and L.C., characters for output on paper tape, if desired, and these will appear where specified. This is not essential however for the fixed store program automatically inserts the appropriate characters where necessary.

Output which has originated as input from another peripheral will contain only the limited set of carriage control characters: NL, CR, LF, Paper Throw, or End of Card (=NL) as appropriate. But when output is originated by program, the program is able to specify paper throwing with or without carriage return, and line feeding with or without carriage return. Owing to the way the different printers are made it is not always possible for these instructions to be obeyed precisely. In these cases, a compromise is made according to the following rules:

a) Paper throwing: the channel number n is interpreted module m, where m is the number of homing channels available on a printer. If no paper throwing facility exists on the printer, the output routine initiates one line feed instead.
b) Line feeding: the number of line feeds performed is always correct, $0 \leq n \leq 15$.
c) Carriage return is performed (if requested) only if it can be done while still retaining the correct number of line feeds. Carriage return is performed (even though not requested) if this is necessary in order to achieve the correct number of line feeds.
d) Repeated spaces or backspace are not inserted by the output routines in any attempt to position the carriage correctly.

In order that on output equipments Tab., UC, LC, and Lower case letters are treated as usefully as possible, on the

Anelex

Tab. is treated as one space.
UC is ignored.
LC is ignored
All outer set letters a-z are printed as inner set letters A-Z.

Card Punch

Tab. is treated as one space

UC is ignored.

LC is ignored

All outer set characters a-z are punched as inner set characters A-Z.

5 Channel Teletype

Tab. is treated as one space.

All outer set characters a-z are punched as inner set characters A-Z.

Binary

Instead of having the input and output information automatically translated to or from the internal code by a fixed store routine it is possible to specify "binary" input or output. When this mode is specified the fixed store program simply copies the pattern of 0's and 1's from the input device to the computer or from the computer to the output device.

A complete binary paper tape is stored as a single record on input. Each card is also stored as a single record. The carriage control character is zero in both cases.

Binary output which has originated from program may contain any carriage control character, but whatever character is found by the output routine it is ignored when punching paper tape, and causes a single card feed when punching cards.

8.7.2 The Internal Character Code

Inner Set

Character	Internal code (octal)	Flexowriter code (binary bits & shift)	Five hole code (binary bits & shift)	IBM Fortran card code (holes punched)	Anelex (bits 35-27 octal)
(Unassigned)	00	** 0000.001	..	..	..
Space	01	** 0010.000	FS 01.110	None	300
Tabulate	02	** 0000.100	..	..	..
Backspace	03	** 0010.101	..	..	..
Shift to outer set	04	..	..	..	..
Shift to inner set	05	..	..	..	..
Shift to LC/LS	06	** 0010.110	** 11.011	..	..
Shift to UC/FS	07	** 0000.111	** 00.000	..	..
(Open brackets	10	LC 0111.000	FS 10.100	0,4,8	234
) Close brackets	11	LC 0101.001	FS 01.100	+,4,8	240
, Comma	12	LC 0101.111	FS 11.110	0,3,8	266
π Pi	13	LC 0111.011	LS 01.111	-,3,8	272
? query	14	LC 0101.100	LS 10.111	-,5,8	364
& Ampersand	15	LC 0111.101	..	5,8	330
* Asterisk	16	LC 0111.110	FS 11.000	-,4,8	344
/ Oblique	17	UC 0011.111	FS 11.101	0,1	350
0 Zero	20	UC 0100.000	FS 00.001	0	202
1	21	UC 0110.001	FS 10.000	1	306
2	22	UC 0110.010	FS 01.000	2	312
3	23	UC 0100.011	FS 11.001	3	216
4	24	UC 0110.100	FS 00.100	4	322
5	25	UC 0100.101	FS 10.101	5	226
6	26	UC 0100.110	FS 01.101	6	232
7	27	UC 0110.111	FS 11.100	7	336
8	30	UC 0111.000	FS 00.010	8	342
9	31	UC 0101.001	FS 10.011	9	246
< Less than	32	LC 0100.011	..	0,5,8	352
> Greater than	33	LC 0110.100	FS 10.001	+,5,8	354
= Equals	34	LC 0100.101	FS 01.010	3,8	256
+ Plus	35	UC 0111.101	FS 01.011	+	314
- Minus	36	UC 0111.110	FS 11.010	-	210
. Point	37	UC 0101.111	** 00.111	+,3,8	220

Notes

** : This paper tape character appears in both paper tape shifts.

(Unassigned) : There is no printing symbol corresponding to this internal code character.

U.C. and L.C.: These are ignored by the peripheral supervisor for equipments where they have no relevance

Anelex codes : The three octal digits given include the information strobe

Tabulate : Tabulate is treated as a single space by the peripheral supervisor on output equipments where it does not otherwise exist.

Inner Set (continued)

Character	Internal code (octal)	Flexowriter code (binary bits & shift)	Five hole code (binary bits & shift)	IBM Fortran card code (holes punched)	Anelex (bits 35-27 octal)
‘ Apostrophe	40	LC 0100.000	FS 10.111	4,8	244
A	41	UC 1010.001	LS 10.000	+,1	356
B	42	UC 1010.010	LS 01.000	+,2	360
C	43	UC 1000.011	LS 11.000	+,3	262
D	44	UC 1010.100	LS 00.100	+,4	264
E	45	UC 1000.101	LS 10.100	+,5	366
F	46	UC 1000.110	LS 01.100	+,6	270
G	47	UC 1010.111	LS 11.100	+,7	372
H	50	UC 1011.000	LS 00.010	+,8	374
I	51	UC 1001.001	LS 10.010	+,9	276
J	52	UC 1001.010	LS 01.010	-,1	200
K	53	UC 1011.011	LS 11.010	-,2	302
L	54	UC 1001.100	LS 00.110	-,3	304
M	55	UC 1011.101	LS 10.110	-,4	206
N	56	UC 1011.110	LS 01.110	-,5	310
O	57	UC 1001.111	LS 11.110	-,6	212
P	60	UC 1110.000	LS 00.001	-,7	214
Q	61	UC 1100.001	LS 10.001	-,8	316
R	62	UC 1100.010	LS 01.001	-,9	320
S	63	UC 1110.011	LS 11.001	0,2	222
T	64	UC 1100.100	LS 00.101	0,3	224
U	65	UC 1110.101	LS 10.101	0,4	326
V	66	UC 1110.110	LS 01.101	0,5	230
W	67	UC 1100.111	LS 11.101	0,6	332
X	70	UC 1101.000	LS 00.0111	0,7	334
Y	71	UC 1111.001	LS 10.0111	0,8	236
Z	72	UC 1111.010	LS 01.0111	0,9	340
(Unassigned)	73	UC 1101.011	..	..	..
(Unassigned)	74	UC 1111.100	..	..	..
(Unassigned)	75	UC 1101.101	..	..	..
(Unassigned)	76	UC 1101.110	..	..	..
Fault	77	..	..	..	..

Notes

** : This paper tape character appears in both paper tape shifts.

(Unassigned) : There is no printing symbol corresponding to this internal code character.

Anelex codes : The three octal digits given include the information strobe.

Outer Set

Character	Internal code (octal)	Flexowriter code (binary bits & shift)	Five hole code (binary bits & shift)	IBM Fortran card code (holes punched)	Anelex (bits 35-27 octal)
(Unassigned)	00	** 0000.001	..	..	..
Space	01	** 0010.000	FS 01.110	None	300
% Percent	02	..	..	..	..
£ Pound sterling	03	..	..	..	324
Shift to outer set	04	..	..	..	..
Shift to inner set	05	..	..	..	..
Shift to LC/LS	06	** 0010.110	** 11.011	..	..
Shift to UC/FS	07	** 0000.111	** 00.000	..	..
(Unassigned)	10	** 0001.000	..	..	..
(Unassigned)	11	** 0011.001	..	..	..
(Unassigned)	12	** 0011.010	..	..	..
(Unassigned)	13	** 0001.011	..	..	..
(Unassigned)	14	** 0011.100	..	..	..
(Unassigned)	15	** 0001.101	..	..	..
(Unassigned)	16	** 0001.110	..	..	..
: Colon	17	LC 0011.111	..	6,8	242
ϕ Phi	20	..	FS 00.011	..	..
[Open square brackets	21	LC 0110.001	..	-,7,8	346
] Close square brackets	22	LC 0110.010	..	-,6,8	250
→ Arrow	23	..	FS 00.101	..	..
≥ Greater or equal	24	..	FS 01.001	..	..
≠ not equal	25	..	FS 10.010	..	..
_ Underline	26	LC 0100.110	..	+,6,8	260
| Vertical bar	27	LC 0110.111	..	+,7,8	362
² Superscript 2	30	LC 0101.010	..	..	370
≈ curly equal	31	..	FS 00.110	..	..
α Alpha	32	UC 0101.010	..	..	274
β Beta	33	UC 0111.011	..	..	376
½ Half	34	UC 0101.100	..	..	204
10 Ten	35	..	..	..	252
11 Eleven	36	..	..	..	254
(Unassigned)	37	UC 1000.000	..	..	..

Notes

** : This paper tape character appears in both paper tape shifts.

(Unassigned) : There is no printing symbol corresponding to this internal code character.

Anelex codes : The three octal digits given include the information strobe.

U.C. and L.C.: These are ignored by the peripheral supervisor for equipments where they have no relevance.

Outer Set (continued)

Character	Internal code (octal)	Flexowriter code (binary bits & shift)	Five hole code (binary bits & shift)	IBM Fortran card code (holes punched)	Anelex (bits 35-27 octal)
(Unassigned)	40	LC 1000.000	..	..	..
a	41	LC 1010.001	..	..	..
b	42	LC 1010.010	..	..	..
c	43	LC 1000.011	..	..	..
d	44	LC 1010.100	..	..	..
e	45	LC 1000.101	..	..	..
f	46	LC 1000.110	..	..	..
g	47	LC 1010.111	..	..	..
h	50	LC 1011.000	..	..	..
i	51	LC 1001.001	..	..	..
j	52	LC 1001.010	..	..	..
k	53	LC 1011.011	..	..	..
l	54	LC 1001.100	..	..	..
m	55	LC 1011.101	..	..	..
n	56	LC 1011.110	..	..	..
o	57	LC 1001.111	..	..	..
p	60	LC 1110.000	..	..	..
q	61	LC 1100.001	..	..	..
r	62	LC 1100.010	..	..	..
s	63	LC 1110.011	..	..	..
t	64	LC 1100.100	..	..	..
u	65	LC 1110.101	..	..	..
v	66	LC 1110.110	..	..	..
w	67	LC 1100.111	..	..	..
x	70	LC 1101.000	..	..	..
y	71	LC 1111.001	..	..	..
z	72	LC 1111.010	..	..	..
(Unassigned)	73	LC 1101.011	..	..	..
(Unassigned)	74	LC 1111.100	..	..	..
(Unassigned)	75	LC 1101.101	..	..	..
(Unassigned)	76	LC 1101.110	..	..	..
Erase	77	** 1111.111	** 11.111	..	..

Notes

** : This paper tape character appears in both paper tape shifts.

(Unassigned) : There is no printing symbol corresponding to this internal code character.

Lower case letters: The outer set letters a-z are treated as inner set letters A-Z for the Anelex printer, the card punch and the 5-channel Teletype.

Carriage Control Characters

Internal Code (octal)	Effect on printed output (see also 8.7.1)
00	None
01	1 line feed without carriage return
02	2 line feeds without carriage return
03	3 .. ..
04	4 .. ..
05	5 .. ..
06	6 .. ..
07	7 .. ..
10	8 .. ..
11	9 .. ..
12	10 .. ..
13	11 .. ..
14	12 .. ..
15	13 .. ..
16	14 .. ..
17	15 .. ..
20	Carriage return without line feeding
21	1 line feed with carriage return
22	2 line feeds with carriage return
23	3 .. ..
24	4 .. ..
25	5 .. ..
26	6 .. ..
27	7 .. ..
30	8 .. ..
31	9 .. ..
32	10 .. ..
33	11 .. ..
34	12 .. ..
35	13 .. ..
36	14 .. ..
37	15 .. ..

Carriage Control Characters (continued)

Internal code (octal)	Effect on printed output (see also 8.7.1)
40	Paper feed without carriage return: home on channel 0
41	 1
42	 2
43	 3
44	 4
45	 5
46	 6
47	 7
50	Paper Feed with carriage return: home on channel 0
51	 1
52	 2
53	 3
54	 4
55	 5
56	 6
57	 7
60	None on existing equipments
61	.. ..
62	.. ..
63	.. ..
64	.. ..
65	.. ..
66	.. ..
67	.. ..
70	.. ..
71	.. ..
72	.. ..
73	.. ..
74	.. ..
75	.. ..
76	.. ..
77	

8.7.3 The FORTRAN Card Code

The twelve card rows are divided into an upper curtate of three rows, the +, - and 0 rows, and a lower curtate of 9 rows, numbered I - 9. Each character is represented by at most one hole in the upper curtate and at most two holes in the lower curtate of a column. If two holes are punched in the lower curtate one of these must be in the '8' row. The characters are divided into four groups or 'zones', depending on the upper curtate punching; no zone, where no hole is punched in the upper curtate, and the +, - and 0 zones where a hole is punched in the +, - and 0 rows respectively.

The table below gives the relation between a character punched on a card, its printed representation and its internal representation. On input if a character which is in the table below is read it is given the internal representation shown. Any characters read which are not listed below e.g. 8,9; +,I,8, are represented by the fault character 77. On output any internal character which has no equivalent punched character in the table below is replaced by internal character 37 and punched as +,3,8, printed as '.' (full stop).

No Zone			+ Zone		
Holes punched	Printed as	Internal Character Octal	Holes punched	Printed as	Internal Character Octal
None	Space	01-I	+	+	35-I
1	1	21-I	+,1	A	41-I
2	2	22-I	+,2	B	42-I
3	3	23-I	+,3	C	43-I
4	4	24-I	+,4	D	44-I
5	5	25-I	+,5	E	45-I
6	6	26-I	+,6	F	46-I
7	7	27-I	+,7	G	47-I
8	8	30-I	+,8	H	50-I
9	9	31-I	+,9	I	51-I
3,8	=	34-I	+,3,8	.	37-I
4,8	‘	40-I	+,4,8	)	11-I
5,8	&	15-I	+,5,8	>	33-I
6,8	:	17-O	+,6,8	_ (underline)	26-O
			+,7,8	| (vertical line)	27-O

- Zone			0 Zone		
Holes punched	Printed as	Internal Character Octal	Holes punched	Printed as	Internal Character Octal
-	-	36-I	0	0	20-I
-,1	J	52-I	0,1	/	17-I
-,2	K	53-I	0,2	S	63-I
-,3	L	54-I	0,3	T	64-I
-,4	M	55-I	0,4	U	65-I
-,5	N	56-I	0,5	V	66-I
-,6	O	57-I	0,6	W	67-I
-,7	P	60-I	0,7	X	70-I
-,8	Q	61-I	0,8	Y	71-I
-,9	R	72-I	0,9	Z	72-I
-,3,8	π	13-I	0,3,8	,	12-I
-,4,8	*	16-I	0,4,8	(	10-I
-,5,8	?	14-I	0,5,8	<	32-I
-,6,8	[	22-O			
-,7,8	]	21-O			

8.7.4 <u>The Flexowriter Code</u>

In general the inner and outer set representations of an internal computer character correspond to the upper and lower case printing respectively of a character on the tape. In the following table each computer character is marked with either 'I' to indicate an inner set or 'O' to indicate an outer set character or 'CC' to indicate a carriage control character. All tape characters should have an odd number of holes in the tape. Those which have not will be replaced by the '77' (Fault) character in the computer.

Two of the keys and the corresponding paper tape characters have alternative printed symbols on some Flexowriters. Thus the key which punches the character 0101.010 gives <u>either</u>

α on upper case and ^{2} (superscript 2) on lower case <u>or</u>
10 on upper case and % (percent) on lower case.

Similarly the character 0111.011 gives <u>either</u>

β on upper case and π (pi) on lower case <u>or</u>
11 on upper case and £ on lower case.

It is not possible on input for the computer to differentiate between the alternative meanings for these characters on the paper tape and hence the former punched character will always be given the Internal Code of 32-Outer Set (α) or 30-Outer Set (2 superscript 2) and the latter character will always be given the Internal Code of 33-Outer Set (β) or 13-I(π) on upper case and lower case respectively. Conversely for output on a seven-channel paper tape punch the Internal Code representations for 10, %, 11 and £ are treated as non-admissible.

Character on tape	Upper Case printing	Lower Case printing	Internal Character octal
0010.000	Space		01
0000.001			00
0000.010	Newline		21-CC
0010.011	Paper throw		40-CC
0000.100	Tabulate		02-I
0010.101	Backspace		03-I
0010.110	Lower Case		06
0000.111	Upper Case		07
0001.000			10-O
0011.001			11-O
0011.010			12-O
0001.011			13-O
0011.100	Stop		14-O
0001.101	Punch On		15-O
0001.110	Punch Off		16-O

Character on tape	Upper Case printing	Lower Case printing	Internal Character octal	
0011.111	/	:	17-I	17-O
0100.000	0	.	20-I	40-I
0110.001	1	[	21-I	21-O
0110.010	2	]	22-I	22-O
0100.011	3	<	23-I	32-I
0110.100	4	>	24-I	33-I
0100.101	5	=	25-I	34-I
0100.110	6	_	26-I	26-O
0110.111	7	|	27-I	27-O
0111.000	8	(	30-I	10-I
0101.001	9	)	31-I	11-I
0101.010	α (10)	^{2} (superscript 2) (%)	32-O	30-O
0111.011	β (11)	π(£)	33-O	13-I
0101.100	½	?	34-O	14-I
0111.101	+	&	35-I	15-I
0111.110	-	*	36-I	16-I
0101.111	.	,	37-I	12-I
1000.000			37-O(U.C.) 40-O(L.C.)	
1010.001	A	a	41	
1010.010	B	b	42	
1000.011	C	c	43	
1010.100	D	d	44	
1000.101	E	e	45	
1000.110	F	f	46	
1010.111	G	g	47	
1011.000	H	h	50	
1001.001	I	i	51	
1001.010	J	j	52	
1011.011	K	k	53	
1001.100	L	l	54	
1011.101	M	m	55	
1011.110	N	n	56	
1001.111	O	o	57	
1110.000	P	p	60	
1100.001	Q	q	61	
1100.010	R	r	62	
1110.011	S	s	63	

1100.100	T	t	64
1110.101	U	u	65
1110.110	V	v	66
1100.111	W	w	67
1101.000	X	x	70
1111.001	Y	y	71
1111.010	Z	z	72
1101.011			73
1111.100			74
1101.101			75
1101.110			76
1111.111	Erase		77-O

8.7.5 <u>The Five Channel Paper Tape Code</u>

Three of the keys when on figure shift and one of the keys when on letter shift, and the corresponding paper tape characters, have alternative printed symbols on some Creed teleprinters. Thus

(i) 00.110 on figure shift has the alternative printed symbols of ≈ (curly equals) and v

(ii) 00.011 on figure shift has the alternative printed symbols of ϕ (phi) and x

(iii) 10.111 on figure shift has the alternative printed symbols of ' (apostrophe) and n

and

(iv) 01.111 on letter shift has the alternative printed symbols of π and £. The former symbol in each case is a Mercury Autocode character and the latter symbol is the standard Mercury/Pegasus/Sirius character.

On input the computer assigns the Internal Code value corresponding to the Mercury Autocode symbol i.e. ≈, ϕ, ', and π. On output for five-channel paper tape the Internal Code representations for v,x,n and £ are treated as non-admissible.

Character	Character Printed		Internal Character	
	Letter Shift	Figure Shift		
00.000	Figure Shift		07	
10.000	A	1	41-I	21-I
01.000	B	2	42-I	22-I
11.000	C	*	43-I	16-I
00.100	D	4	44-I	24-I
10.100	E	(	45-I	10-I
01.100	F	)	46-I	11-I
11.100	G	7	47-I	27-I
00.010	H	8	50-I	30-I
10.010	I	≠	51-I	25-O
01.010	J	=	52-I	34-I
11.010	K	-	53-I	36-I
00.110	L	≈ v	54-I	31-O
10.110	M	LF	55-I	01-CC
01.110	N	Space	56-I	01
11.110	O	,	57-I	12-I
00.001	P	0	60-I	20-I
10.001	Q	>	61-I	33-I
01.001	R	≥	62-I	24-O
11.001	S	3	63-I	23-I
00.101	T	→	64-I	23-O
10.101	U	5	65-I	25-I
01.101	V	6	66-I	26-I
11.101	W	/	67-I	17-I
00.011	X	ϕ(phi)(x)	70-I	20-O
10.011	Y	9	71-I	31-I
01.011	Z	+	72-I	35-I
11.011	Letter Shift		06	
00.111		.	37-I	
10.111	?	'(apostrophe)(n)	14-I	40-I
01.111	π (£)	C.R.	13-I	20-CC
11.111	Erase		77-O	

8.7.6 The Teleprinter Code

The teleprinter can only be used by the Supervisor and Test programs. When the output is in Internal Code the code conversion used is that for the teletype punch. Hence the correct printing is obtained only if a restricted set of characters is used.

Octal Value (digits 35-27)	Internal Code (octal)	Printed Character	Octal Value (digits 35-27)	Internal Code (octal)	Printed Character
100	01	SP (space)	400		,(comma)
004		* (asterisk	504	41-I	A
010	21-CC	NL (newline)	510	42-I	b
114		(red)	414	43-I	c
020		(black)	520	44-I	d
124		(pi)	424	45-I	e
130		]	430	46-I	f
034		[	534	47-I	g
040		(	540	50-I	h
144		)	444	51-I	i
150		<	450	52-I	j
054		>	554	53-I	k
160		:	460	54-I	l
064		_ (underline)	564	55-I	m
070		| (vertical bar)	570	56-I	n
174	17-O	/ (oblique stroke)	474	57-I	o
200	20-I	0	700	60-I	p
304	21-I	1	604	61-I	q
310	22-I	2	610	62-I	r
214	23-I	3	714	63-I	s
320	24-I	4	620	64-I	t
224	25-I	5	724	65-I	u
230	26-I	6	730	66-I	v
334	27-I	7	634	67-I	w
340	30-I	8	640	70-I	x
244	31-I	9	744	71-I	y
250	32-O	α (alpha)	750	72-I	z
354	33-O	β (beta)	654		?
260	34-O	½	760		=
364	35-I	+	664		'(prime)
370	36-I	-	670		&(and)
274	37-I	. (point)	774	77-O	ER(erase)

8.8. I.B.M. Magnetic Tape.

8.8.1. Tape Layout and Specification

1.1. The tape is ½" wide and has 7 tracks. Of these 7 tracks 6 are used for data and 1 for parity. There is no clock track provided and characters are recognised by the presence of bits in at least one of the tracks. Two densities of recording are possible, 200 and 555 characters/inch.

1.2. Information is recorded using the NZR1 method i.e. the tape is saturated in either of two possible directions. A one is recorded on the tape by a reversal of flux whilst a zero is the absence of such a reversal.

1.3 Data is written in two different character representations. They are:

a) Binary mode.
When it is desirable to use magnetic tape as a backing store to the computer the data will be transferred to tape directly from the main store without any code conversion and hence will be in straight binary form. In the Binary mode the parity bit is such that the number of bits in a stripe is odd.

b) BCD Mode.
If however the magnetic tape is considered as an output device then alpha-numeric information will be required. The data must therefore be coded and is (known as being) written in the BCD or Binary Coded Decimal Mode. In the BCD mode the parity bit is such that the number of bits in a stripe is even.

1.4. Blocks of data on tape, known as RECORDS, can in theory be any length and are unaddressed. Because records are of variable length selective overwriting is virtually impossible.

1.5. After each record is written a check character, known as the Longitudinal Redundancy Check Character or LRCC. The character is such that the number of ones in the record, including these in the character itself, in each of the 7 tracks, is even.
The check is a biproduct of the method of recording. As stated in para. 1.2 ones are recorded by reversals of flux due to reversals of current in the writing heads. At the end of a record the current in each head is returned to the initial direction, a one being written where-ever the direction of current in a head is switched over.

1.6 In order that tape wastage should be as little as possible the inter-record gap is ¼”. For compatibility between all IBM tape units this gap length should be kept as accurate as possible.

8.8.2. The IBM 729 IV Tape Unit.

2.1. Performance
Tape speed is 112.5 ins/sec. The character transfer rates are therefore 22.5 and 62.5 Kc/s when the

recording densities are 200 and 555 characters/in respectively.

2.2. 729 IV Tape Units have separate read and write heads. This allows the data written on tape to be checked for accuracy immediately after writing. The inter-head gap is .300 inches (or 2.67ms at nominal tape speed).

2.3 <u>Writing on Tape</u>

The current in the write head is always left on when writing on a tape. This is done to avoid depositing noise in the inter-record gap when switching the current on and off and also to erase any previous information written in the present inter-record gap (As a consequence of writing variable length blocks the inter-record gap is not likely to occur in the same place). This feature is not usually troublesome; however special measures do have to be taken when Backspacing the tape (see below).

2.4 <u>Backspacing a Tape</u>

It is not possible to write or read a tape backwards but it is possible to backspace the tape a record at a time. To avoid erasing a tape while Backspacing the tape can only be moved backwards when the tape unit is in the Read Status i.e. with the write current switched off. Special action is taken if a Backspace order is attempted while the tape unit is in the Write Status (the write current switched on). In this case the tape is moved forward a distance slightly greater than the inter-record gap while still in write status. This action leaves a clean inter-record gap after the record. Write status is dropped and the noise bits, written on tape, are well clear of the preceeding record and in a position where they will be removed should further records be written on this tape. The backspace operation then proceeds in the normal way, i.e. the tape is moved backwards until the beginning of the record is reached and tape motion is stopped at a time which will ensure that a Reread or Rewrite operation will be obeyed satisfactorily. (Tape drive is switched off 3ms after the beginning of the record is detected). Logic is built into the tape units to prevent a backspace operation being carried out if the tape is at the Load Point.

2.5 <u>Rewinding Tape</u>

To avoid erasing the tape it can only be rewound when the tape unit is in the Read Status. If an attempt is made to rewind in the Write Status a similar procedure is adopted to that for an attempted Backspace order when in the write status (See para 2.4). The tape is moved forward in the write status a distance greater than the inter-record gap. Read status is picked up and the tape rewound.

For full details of control and operation of IBM 729 IV tape units see IBM 729-11 and 729-IV Magnetic Tape Units, Technical Information for Original Equipment Manufacturers; and also, 729-II, 729-III, 729-IV Magnetic Tape Units, IBM Customer Engineering Manual of Instruction.

8.8.3 General Description of the System

3.1 General

The Control Unit organises the transfer of data between IBM 729 IV Tape Units and the Atlas Peripheral Controller and is considered as a fast Atlas peripheral device. Associated with the unit there is a small buffer store. Transfers between Atlas and this buffer are organised by a fixed store programme. The programme is called in on an interrupt basis whenever the control unit signals by means of a Look-at-Me that the buffer requires attention (Buffer Attention Look-at-Me, BLMI, para. 4.2.)

The buffer is made up of two registers V and A each 4x 24 bits long.

3.2 Basic Description of Tape Control Unit Performance.

3.2.1 Orders that can be obeyed by the Control Unit

(1) Read- transfer the next record from tape to Atlas.

(ii) Write - transfer the next record from Atlas to tape.

(iii) Backspace - move the tape backwards one record.

(iv) Rewind - rewind the tape to the load point.

(v) Rewind and Disengage - as (iv) except that the tape unit is also disengaged.

(vi) Disengage - disengage the tape unit from Atlas.

3.2.2 Transfers may be in any combination of High/Low Density and Binary/ BCD mode.

3.2.3 The Tape Control logic consists of one road/write channel such that only one of the orders Read, Write, Backspace, or Disengage may be in progress at any one time. However Rewind if previously initiated may be in progress simultaneously on other tape units.

3.2.4. Up to three IBM 729 Mk. IV tape machines may be connected to the tape control logic.

3.2.5. Character conversion will be done in the BCD mode for both read and write transfers.

3.3. General Description of the Logic.

3.3.1 When reading data from tape.

Characters are received in seven read flip flops known as Read Stats, staticisation being essential to accommodate any tape skew within each character. The characters are then gated to successive six bits of register A (the parity digit is not transferred to the buffer). When A is full the data is transferred to V and BLMI is set. In due course the interrupt is dealt with and the data in V is read to Atlas.

3.3.2 When writing data to tape.

The data is written to V from Atlas during the period while the data in A is being dealt with. When A has been emptied the new data held in V is transferred to A and BLMI is set to indicate to Atlas that V needs refilling. The data in A is extracted 6 bits at a time and written on tape via 7 write stats, one extra digit being added to give the character the correct parity.

8.8.4. Specification of Lines between the Tape Control Unit and Peripheral Co-ordinator.

4.1 Layout of Tape Control V-store

Type 5

Line	Digit	Direction	Name
0	24-27	RW1	Character Count
	28	RW1	Read: TU1 Disengaged Write: Stop Write
	29	RW1	Read: TU2 Disengaged Write: Select BCD Mode
	30	RW1	Read: TU3 Disengaged Write: Select Low Density
	31-33	RW1	Read: Rewind in Progress TU1-3 Write: Select TU1-3
	34	R	Low density selected
	35	RW1	Read: BCD selected Write: Recover read
1	24	RW1	Read: Buffer Attention Overdue Write: Read Order
	25	RW1	Read: Lateral Parity Write: Write Order
	26	RW1	Read: Longitudinal Parity Write: Backspace
	27	RW1	Read: Tape Indicator Set Write: Rewind
	28	RW1	Read: Load Point Indicator Write: Rewind and Disengage
	29	RW1	Read: Mechanical Failure Write: Disengage
	30	W1	Put out End of Operation Look at me
	31	W1	Put out Buffer Attention Look at me
	32	W1	Inhibit Buffer Attention Look at me
	33	W1	Allow Buffer Attention Look at me
5	47-24	RW1	Information
4	47-24	RW1	Information
3	47-24	RW1	Information
2	47-24	RW1	Information

Type 14

Line	Digit	Direction	Name
24	30	R	I.B.M. Magnetic Tape, Buffer Attention
	31	R	I.B.M. Magnetic Tape, End of Operation

4.2 Description of V Store

Titles of each digit are followed by Atlas Reference Name and Tape Control waveform name.

I.B.M. Magnetic Tape Buffer Attention (Type 14 line 24 digit 30)(BLMI)

This occurs
(a) when the buffer is full on a Read Transfer.
(b) the buffer requires filling on a Write Transfer.

I.B.M. Magnetic Tape, End of Operation (Type 14 line 24 digit 31)(ELMI)

This is set by the Control Unit at the end of each operation as follows:
(a) Read After LRCC has been read and last buffer transfer completed
(b) Write After LRCC has been read back and checked.
(c) Backspace 3ms after the last character of the block has been detected.
(d) Rewind, Disengage, Rewind and Disengage When a relay closure return signal operates in the tape unit (~10ms).
(e) When reading blank tape and the End of Tape mark is photosensed. The Tape Indicator is also set.
(f) When the selected tape unit does not respond to control signals of an order. Mechanical failure is also set.
(g) When a tape unit is Disengaged manually whilst obeying an order.

End of Operation Look at Me indicates that the previous operation is complete so far as the Tape Control Unit is concerned, and that the next order may proceed.

Character Count (Type 5 line 0 digits 24-37) (V10/24-27)

(a) Read Set by Tape Control to indicate the number of significant characters in the last transfer of a Read operation. The count is transferred to the V store in parallel with the information bits. Zero in the count indicates that the Buffer is empty.
(b) Write Set by Write extracode simultaneously with the last transfer of information to be written to tape (and also Stop Write), to indicate the number of significant characters in the buffer. Zero in the count indicates that the buffer is full.

Stop Write (Type 5 line 0 digit 28) (V10/28)

Set by Atlas when, and simultaneously with, the transfer of the last word (or part word) to the information buffer for a Write Operation. The Character Count is also transferred at this time.

Select BCD Mode (Type 5 line 0 digit 29), (VIC)

When set this digit causes future transfers associated with the present order to be carried out in the BCD mode. If this digit is not set transfers will be carried out in the binary mode.

Select Low Density (Type 5 line 0 digit 30), (VISL)
When set this digit causes information associated with the present order to be recorded on tape at low density. If this digit is not set information will be recorded at high density.
This digit must also be set when reading information, which is recorded at low density, from tape.

Select Tape Unit 1-3 (Type 5 line 0 digit 31-33), (UI1-UI3)
The tape unit is selected by setting the appropriate line. The unit remains selected until a further unit is selected. When the control unit is switched on (as in the morning) TU1 is automatically selected.

Low density selected and BCD selected (Type 5 line 0 digits 34,35)
These allow the supervisor to check that the two flip-flops have been set correctly.

Recover Read (Type 5 line 0 digit 35,) (VIT)
If during a normal read operation a parity failure occurs a further attempt may be made to read a record by using a reduced bias level in the peak sensors. The reduced bias level is selected by Recover Read Digit.

Read Order (Type 5 line 1 digit 24), (RI)
An order digit which causes a Read operation to be performed on the selected tape unit, when it is set.

Write Order (Type 5 line 1 digit 25), (WI)
As above except for Write.

Backspace Order (Type 5 line 1 digit 26), (BI)
An order digit which, when set causes the selected tape unit to move backward one record.

Rewind Order (Type 5 line 1 digit 27), (VIM)
An order digit which when set causes the selected tape unit to rewind to the Load Point.

Rewind and Disengage Order (Type 5 line 1 digit 28), (VIND)
As above except that the tape unit is disengaged and the tape rewound until the Load Point is reached.

Disengage Order. (Type 5 line 1 digit 29,) (VID)
An order digit causing the selected tape unit to be disengaged from Atlas (returned to manual control).

Put Out End of Operation Look at Me (Type 5 line 1 digit 30), (IRELM)
Reset by the End of Operation interrupt routine.

Put Out Buffer Attention Look at Me. (Type 5 line 1 digit 31), (IRBLM)
Reset by the Buffer Attention interrupt routine.

Inhibit Buffer Attention Look at me (Type 5 line 1 digit 32)
This sets a flip-flop to inhibit the BALAM interrupt being sent to the peripheral co-ordinator, and will be used to prevent wasting storage space in the event of failure while

reading tape. It has no effect on the End of Operation look at me, which will come up at the end of the block being read.

Allow Buffer Attention Look at me (Type 5 line 1 digit 33)

This resets the flip flop to allow BALAM interrupts to be sent to the peripheral co-ordinator.

Mechanical Failure (Type 5 line 1 digit 29) (FIM)
Set up by the Tape Control unit as follows:-

(i) When the selected tape unit does not respond to the initial control signals of an order. In this case execution of the order is inhibited and ELMI is set.
(ii) When the selected tape unit is disengaged manually whilst it is obeying an order. ELMI is also set.
(iii) During a Write order when a Write Echo failure occurs.

Tape Unit 1-3 Disengaged (Type 5 line 0 digits 28-30 (RPIEU1-3)

Set to one while a tape unit is disengaged. A tape unit may be disengaged by a Disengage order from Atlas or by pressing the 'Reset' key on the tape unit. A tape unit may be engaged only if the deck is loaded with all interlocks closed and the ‘Start' key has been pressed.

Rewind in Progress Tape Unit 1-3 (Type 5 line 0 digits 31-33),(RPINU1-3)

Set to one while a Rewind order is carried out on the Tape Unit.

Buffer Attention Overdue (Type 5 line 1 digit 24), (OIBLM)

Set by the Tape Control Unit if information is being destroyed due to the inability of the Buffer Attention interrupt routine to transfer information to or from the buffer in the required time.

Lateral Parity Failure (Type 5 line 1 digit 25), (PILA)

Set up during a Read, Write or Backspace operation by Tape Control if a lateral parity check failure is detected.

Longitudinal Parity Failure (Type 5 line 1 digit 26), (PILO)

As above, except for longitudinal parity failure.

Tape Indicator (Type 5 line 1 digit 27), (RPIIND)

Set by tape unit when the reflective spot at the end of tape is photosensed. Reset by Put Out End of Operation signal.

Load Point Indicator (Type 5 line 1 digit 28), (RPILP)

A signal which is in the "one" condition if the tape on the selected tape unit is at the Load Point. When the tape is moved from the Load Point position the Load Point Indicator falls to the "zero" condition.

Information Digits (Type 5 lines 5-2 Digits 47-24), (V15-V12)

The information is set in or read from the buffer in 4 half Atlas words. The most significant digit is digit 47 in line 5 and the least significant digit is digit 24 in line 2.

Write Strobes for Input Lines. (Logical Design Only),(ITVA-ITVF)

6 input strobes are provided, one for each line of the V store. These are used to strobe the information set in each digit of the line into the V store.

These strobes are labelled ITVA-ITVF corresponding to lines 0-5 respectively.

Console Reset (Logical Design Only), (IRC)

A strobe used to reset all stats in the Tape Control Unit to their standard states when the machine is switched on (in the morning etc.)

4.3 Lines between Tape Control and the Tape Units.

Use is made of the IBM tape unit bus system. In this system all control lines are common to all tape units with the exception of those actually specifying which Tape Unit is selected.

The signals required by these control lines swing about either of two reference levels. Those reference levels are:-

(a) N type; whose level is ground

and (b) P type; whose level is -6 volts.

For reliable operation a minimum swing of ±0.4 volts is required but IBM recommend that the swing should be ±1 volt. Level changers have been built which convert between standard Atlas signal levels and both N and P type levels. The standard IBM lines to and from the Tape Units are as follows:-

4.3.1 Lines to IBM Tape Units

+P Select Waveform Names; UI1, UI2, U13

Each of these 3 lines is used to select a particular tape unit from the group connected to the common Control Unit. The signal gates the tape unit selected allowing it to receive all subsequent signals from the Control Unit. The line must be held at +P level for the duration of the operation. Dropping the Select Line immediately stops any tape operation except rewind.

+P GO Waveform Name; WPIGO

This line is used to control tape motion. Bringing GO to +P level starts tape motion; whilst Dropping GO stops it except for rewind.

+P Set Read Status Waveform Name; WPISR

When pulsed at +P level for 1μ sec this line switches the Read/Write flip-flop in the selected tape unit to read status (if not already there).

-N Set Write Status Waveform Name; StWI

Same as above except for write.

-N Backwards Waverform name; ILBK
A line which when held at +N level causes any tape motion to be in the forward direction and when held at -N causes it to be in the backward direction. It must not be changed whilst the GO line is at +P level.
-N Write Pulse Waveform Name; WPIWP
A line which, when pulsed for 1μsec, writes a character, held on the Write Bus, on tape. Pulses are repeated every 16μsec for High and 44.5μsec for Low Density Recordings.
NOTE: For convenience in logical design and also to conform with IBM practice pulses are 2.78μsec wide when recording at Low Density.
-N Write Check Character Waveform Name; WPILRCC
This line is brought to -N level when the first character of a block is written and returned to the +N level after the last character of the block has been written. When the line is returned to the +N level all 7 stats connected with the Write Bus are returned to 'zero' state, and this writes the Longitudinal Redundancy Check Character. This Character is written 64 or 178μsec after the last Write Pulse according to whether recording is at High or Low Density.
-N Turn on Tape Indicator (Not Used)
Turns on Tape Indicator when pulsed for 1μsec if the tape unit is under automatic control.
+P Turn off Tape Indicator Waveform Name; WPIRIND
Turns off Tape Indicator when pulsed for 1μsec if the tape unit is under automatic control.
-N Start Rewind Waveform Name; WPIN
This line is brought to -N level to start a rewind operation. The Tape Unit indicates to Control that the rewind operation has been initiated by bringing Select and Rewind line to the -N level (about 10ms later). Control is now free to select another tape unit leaving the first tape unit to rewind autonomously. The tape will be left in the loaded state when the rewind operation is complete.
+P Rewind and Unload Waveform Name; WPINU
As above except that the tape is left in the unloaded state when the operation is completed.
+P Set High Density (Not used)
A line which causes the High/Low Density flip-flop to be switched to the High Density state (if not already there).
The operation is completed when the Density status line from the tape unit takes up the required status. This flip flip has no direct control over the circuits in the tape unit. In the IBM system the flip flop is used by the Tape Control Unit as a memory indicating the density of recording of the tape on the tape unit.
-N Set Low Density (not used)
As above except for Low Density.

4.3.2 Lines from IBM Tape Units

+P Select and Ready   Waveform Name; RPIS
This line is at +P level if the deck is selected and all power supply interlocks are made, the tape unit is not in the rewind status and the Start Key has been depressed.

-N Select and Tape Indicator on Waveform Name; RPIIND
This line is at -N level if the deck is selected and the Tape Indicator Stat is turned on. The Tape Indicator is turned on when the reflective spot at the end of a tape is photosensed.

+P Select and at Load Point Waveform Name; RPILP
This line is at +P level if the deck is selected and the tape is at the Load Point. When the tape moves away from the Load Point this line falls to -P level.

+P select and not a Load Point (not used)
This is the inverse of Select and at Load Point.

+P Select Ready and Read   Waveform Name; RPISR
This line is at +P level provided the tape unit is selected and ready and that the Read/Write flip-flop is in the read status.

+P Select Ready and Write   Waveform Name; RPISW
As above except for write.

+P High Density, -P Low Density   Waveform Name; RPID
A line which is at +P level if High/Low Density stat is in the High status and at -P level if it is in the Low status

-N Select and Rewind   Waveform Name; RPIN
This line is at -N level while the tape unit is rewinding provided the unit is selected.

-N Write Echo   Waveform Name; RPIWE
This line gives out a 1μsec pulse every time a Write Pulse causes a ‘one’ to be written in any of the seven tape tracks. This pulse is used to test that writing is being performed correctly. There is no echo pulse when the Longitudinal Redundancy Check Character is written.

Read Bus Waveform Names; RPI1, RPI2, RPI4, RPI8, RPIA, RPIB, RPIC.
Seven Lines one for each track, fed directly from the read head preamplifiers. These lines are connected to modified Atlas/Orion peak sensors which give out standard Atlas signal levels.
IBM head preamplifiers deliver signals of 10 volts peak-to-peak amplitude.

4.4. Due to differences between the IBM and Atlas operating conventions the following extra control lines have been added to the tape units. For convenience they use IBM signal levels between tape units and Tape Control.

4.4.1 Extra Lines to IBM Tape Units

-N Disengage   Waveform Names; IDU1 - IDU3
Three lines, one to each tape unit, which when brought to -N level operate the 'Reset' key and so place the chosen tape unit in the Manual Status.

4.4.2 Extra Lines from IBM Tape Units.

+P Tape Unit Engaged Waveform Names; RPIEU1 - RPIEU3

One line from each tape unit which when at +P level indicates that the unit is ready to obey an order i.e. the 'Start' key has been pressed, all power interlocks have been made, and the tape unit is not in the rewind status.

Note. These lines are similar to the Select and Ready line except that the tape units do not have to be selected to be engaged.

-N Rewind in Progress   Waveform Names; RPINU1-RPINU3

One line from each tape unit which when at -N level indicates that the unit is rewinding.

Note. These lines are similar to the Select and Rewind line except that the tape units do not have to be selected to be at the -N level.

8.8.5 <u>Details of the Control Unit, its operation and Design.</u>

5.1 <u>Timing</u>

The major part of the control unit is synchronised with the character transfers to and from tape. The character transfer rate is of course governed by the recording density and tape speed and is 22.5 Kc/s at Low density and 62.5 Kc/s at High density.

5.1.1 <u>Writing</u>

The character transfer rate is controlled by either of two crystal oscillators dependent on whether the recording density is high or low. For reasons which will appear later it is convenient to use basic oscillator frequencies which are 16 times greater than the character rate, i.e. 360 Kc/s and 1 Mc/s.

It is important to point out that the same hardware is used for both densities, a change of density being merely a change of oscillator.

5.1.2. <u>Reading</u>

The circuitry is controlled by the rate of arrival of characters from the tape itself.

In theory therefore it is unnecessary to know what the density of recording is. However the end of a record is determined by the absence of any characters within a given time. As the time is different for the two recording densities the density of any particular tape, in fact, must be known.

5.2 <u>Buffer Registers</u>

There are 96 bits in each of the two buffer registers, V and A, This number is a compromise between the amount of hardware used in the buffer and the amount of central computer time used dealing with Buffer Attention Interrupts. With a 96 bit buffer 20% of the central computer's time is used in dealing with tape control interrupts when working with long blocks on high density tapes. The general formula for the percentage computer time used for dealing with interrupts is:-

$$\frac{T + q\ n + [(n + p - 1)/p]\ t \times 100}{7300 + 128n}$$

Where T = Terminal computing time (-100/μsec)
      Q = time of loop to transfer one 48 bit Atlas word
           (between 8 and 14μsec)
      n = size of block in words
      p = size of buffer in words
      t = time to deal with an end of interrupt (~30μsec)
      [] = means the integral part of

5.2.1. The information flip-flops are numbered lines 2, 3, 4 and 5 digits 24 - 47, digit 47 in line 5 being the most significant. Information is strobed into buffer A from the Read Stats character by character during a Read order, but it is strobed 24 bits at a time from the Peripheral Co-ordinator into V during a Write order. The strobe in the latter case is supplied by the Peripheral Co-ordinator for with the interrupt method of operation the Tape Control Unit has no indication when the Buffer Attention Look At Me will be dealt with.

To reduce the loading on each output of the flip-flops in the registers to one standard load, transfers between the two registers are done by the Reset and Single Sided set technique.

Should a read or write transfer conclude with a partially filled buffer a count indicating the number of significant characters will be sent to or from Atlas respectively.

5.3 <u>Significant Character Count and Stop Write Digits</u>

As the number of possible characters in each buffer register is 16, 4 flip-flops are needed for the count. It is convenient to consider the flip-flops as part of the buffer. These digits are referred to as the Significant Character count, VIO digits 24 - 27.

With Write transfers only, a digit must be sent to the tape control unit from Atlas to indicate when the last transfer is taking place. The digit is also considered as part of the buffer and is referred to as the Stop Write Digit, AIO/28.
(No digit is required to indicate the last read transfer as this transfer is accompanied by End of Operation Look at Me).

5.3.1. <u>When Writing</u>

These five extra digits are written to V at the same time as the last characters of a record. They are transferred to A under Transfer Control where the Stop Write Digit is recognised immediately and is used to inhibit setting BLMI.

It also allows comparison between the Significant Character Count and the number held in the Character Counter (para 5.14). As the characters are written on tape the Character Counter is increased and at some count coincidence occurs between the number in the counter and the Significant Character Count. Waveform ICIN is generated which is used by Write Control to terminate Write orders.

5.3.2. <u>When Reading</u>

When the last buffer transfer takes place to V the current character count is transferred to the significant Character Count, VIO digits 24-27, where it is read with the last few characters in the buffer.

<u>Note.</u> All Read and Write Transfers transfer the Significant Character Count between V and A but they are only meaningful when dealing with the last characters of information.

5.4 <u>Write Control</u>

5.4.1 <u>Performance.</u>

The logical design is governed mainly by IBM specification for Writing information. The relevant requirements are that:

(a) The character repetition rate is every 16μsec (44.5)

(b) That the 7 bit stripe should be set up on the Write Bus with as little skew as possible.

(c) That a Write Pulse Line should be pulsed with a 1μsec (2.78) wide pulse. This pulse should not occur before 4.5μsec after the character has been set up on the Write Bus.

(d) That the Write Bus should be cleared 15μsec (41.7) after it is set up.

(e) That the Longitudinal Redundancy Check Character should be written 64μsec (178) after the completion of the last Write Pulse.

The figures without brackets refer to High Density and the figures in brackets refer to Low Density.

As the repetition cycle time even at High Density is 16μsec the Write Bus is cleared for 1μsec. which gives adequate time for a new character to be set up on the Write Stats. The LRCC is written by dropping the Write Check Character line (para 4.3.1.) at the correct time. This line is set at the same time as the Write Pulse line is pulsed for the first character.

5.4.2 <u>Logic Design.</u>

From the above required performance it is apparent that a basic frequency of 1 Mc/s (360 Kc/s) will be useful.

The control unit of the Write Control is a 4 bit counter which is fed from either of two crystal oscillators. These oscillators produce 100 mμsec pulses every 1μsec (2.78). The counter is decoded to give the necessary timing required during the write cycle.

The decoded waveforms are gated with the crystal oscillator to give 100mμsec pulses.

The Write Stats are connected directly to the Write Bus so that Information is set up on the Write Bus by setting the Stats and is cleared from the Bus by resetting them.

The general sequence of events is:

(i) Set character on Write Bus.

(ii) add one to counter.

(iii) test if zero.

A consequence of this layout is that zero in the Significant Character Count represents a full buffer.

5.4.3 Operation

After completion of the WRITE DELAY, which allows the tape to get up to a speed and also defines the length of the inter-record gap, waveform ILWRA occurs (para 5.7.5). This waveform gates one of the crystal oscillators to the Write Counter. Flip flop WRI is used to clean up the oscillator pulse if it should occur at the same time as ILWRA. Only 4 bits of this counter are relevant initially.

The 100mµsec pulse waveforms StITWS(mo), ITWKA (M7), ITWTA (m8), StIRWS (m15) are generated cyclically; the timing of these waveforms is given in brackets.

StITWS strobes the character selected by the current character count from Buffer A to the Write Stats (para. 5.18) and hence to the Write Bus (para. 4.3.1)

ITWKA Sets stat. WPIWP switching the Write Pulse line on (para. 4.3.1.)
adds one to the Character Counter (para. 5.14)
sets Write Echo Stat. WIWE.
sets stat. WPILRCC switching the Write Check Character line (para. 4.3.1.) to the 'on' state at the same time as the first Write Pulse.

ITWTA resets WPIWP switching the Write Pulse line off.
tests whether a buffer transfer is required (para 5.6.1.).

StIRWS resets the Write Stats (para 5.18)
/1 & 2 tests that WIWE ahs been reset by a Write Echo pulse, RPIWE, (Para 4.3.2). If WIWE has not been reset then the Mechanical Failure digit is set (paras 4.2 and 5.7.2.1.)

When ICIN (para 5.3.1.) occurs the counter is extended by 3 stages. The waveform occurs after m7 because the Character Counter is advanced then but well before m15. The counter now becomes modulo 128 instead of modulo 16, so that after the last character has been written on tape mO, m7, m8 and m15, and their corresponding waveforms are logically inhibited.

When the counter reaches m72 (m8 + 64) StITCW is generated which resets the Write Check Character stat and inhibits further counting.

5.5 Read Control

The general sequence of events is
(i) Detect Character
(ii) Add one to counter
(iii) test if zero

A character in the Read Stats (para 5.19) is detected by mixing the outputs of the stats together. Read control accepts the character provided that the last character of the record has not so far been detected. The Longitudinal Redundancy Check Character is rejected by gating with ILRCCD. The pulse generated is delayed by 7.0µsec at High or 20.0µsec at Low Density to allow the whole character to be set up even in cases of serious tape skew.

The sequence of events is then as follows:-

ITCB is sent to the Longitudinal Parity Logic where the bits of the character including parity, held in the Read Stats, are 'non equivalently' added into the Longitudinal Parity Register (para. 5.16)

StITL/3 and via this logic to the Record Gate Logic (para 5.13)

ITCB sets the 'Lateral Parity Failure stat, PILA, if the number of ones in the character held in the Read Stats. is inconsistent with the mode of recording (para 5.15.)

ITCA gates the character away to buffer A if the a Read order is being obeyed (see ED/A/19.1D)

ITCB resets the End of Blank Tape Stat, INDIA, (para 5.10) if this stat has been set by the Tape Indicator 0.5μs later

ITCC advances the Character Counter by one (para 5.14) 0.5μs later

ITCD tests if a buffer transfer is required (para 5.6.1.)

StIRRS resets the Read Stats in preparation for the arrival of the next tape character (para 5.19)

5.6 Transfer Control

5.6.1. Normal Operation

Any transfer which is not the first or the last in a record is considered as normal in this context. After every character has been read by Read Control or written by Write Control a signal (para 5.4.3. and 5.5) is sent to Transfer Control to test whether a buffer transfer is required. This is so if the character count is 16 i.e. waveform StISGOO is operative (para. 5.14).

Firstly a check is made to ensure that Buffer Attention Look at Me, BLMI, set by the previous transfer, has been reset by Atlas. If it has not then the Overdue Digit, OIBLM, is set.

The transfer sequences for Read and Write orders are as follows:-

READ:-	RESET V	TRANSFER A to V	RESET A
WRITE:-	RESET A	TRANSFER V to A	RESET V

Lastly BLMI is set to inform Atlas that Register V needs attention.

5.6.2 Special Operation

Special cases may occur at the beginning and end of a record.

5.6.2.1. First Read Transfers:- The initial characters are dealt with in the normal way as described above.

5.6.2.2 First Write Transfer:- Waveform ITWS from MAIN Control (para.5.7.5.1) sets flip-flop WIS. and also BLMI (It also causes buffer transfers but these are irrelevant). BLMI is detected by an interrupt routine; data is fed to V and BLMI is put out by IRBLM. IRBLM also inspects whether WIS is set. If it is, as in this case, a transfer, V to A, is executed and BLMI is set again. In time the interrupt will be dealt with and V will be refilled.

After the Write Delay ITWRA (para 5,7.5) tests BLNI to make sure that both these transfers have been completed. If these have not been done OIBLM is set.

Transfers then follow the normal pattern.

5.6.2.3 Last Read Transfer:- When dealing with the last read transfer there are two possibilities:

(a) The characters exact fill a buffer register (i.e. the current character count is 16). In this case ITCD from Read Control (para 5.5) will initiate a Normal Transfer before the Record Gate Logic has determined that characters are in fact the last in the record.
Eventually the Record Gate Logic will determine that the end of the record has been reached and READ DISCONNECT, StITRD, PULSE (para. 5.13.1.1) will be sent to Transfer Control.

(b) The characters only partially fill a buffer register. The final ITCD waveform (para 5.5) will find that a buffer transfer is not required and these odd characters will remain in A until StITRD (para 5.13.1.1) arrives.

When the StITRD does arrive the action taken depends on whether or not BLMI is set.

BLMI is set:- The pulse is held up on flip flip RIE until BLMI is reset, the BLMI Not set condition is then obeyed.

BLNI is not set:- The final read transfer is performed and END OF READ SIGNAL, IERD, is generated.

NOTE. When dealing with data which exactly fills the buffer the last read transfer still takes place but is redundant. The interrupt routine dealing with End of Operation has been designed to ignore this transfer.

5.6.2.4. Last Write Transfer:- The final characters are transferred in the normal way as described in para 5.6.1 except that the Stop Write digit, AI0/28, is detected immediately the transfer has taken place and inhibits ITDD from setting HMI. AIO/28 is also used to inhibit ITWTA whilst writing the last characters. This is necessary for if the number of characters in the buffer happens to be 16 a further transfer would be initiated. This transfer would fill buffer A with 'zeros' and in particular set AI0/28 to zero. ITTD would then be able to set BLMI and so cause a misleading interrupt.

5.7 Central Control

5.7.1 Detect Order

At any time an order may be sent to the Tape Control Unit V store. The presence of this order is detected by mixing all possible order lines together. Waveform ILDO is produced when an order is detected.

5.7.2 To Check that the Order can be obeyed

5μs are allowed for all the necessary details about the order to be set up on the V store control flip-flops and lines from the selected tape unit to respond to the Select Signal. A test is then made to ascertain whether the selected tape unit is in a position to obey this order. All orders require that the selected unit is READY (para. 4.3.2) and in particular Read and Write orders require that the tape unit is in the Read and Write Statuses (para. 4.3.2) respectively.

If the test is successful then the order digits are transferred by StITSO to a second bank of stats from which the order specified is obeyed. If however the test is unsuccessful the digits are not transferred and ELMI and the Mechanical Failure Digit FIM are set by ITS.

5.7.2.1. <u>The Mechanical Failure Stat (FIM)</u>

The stat is set by:

(a) by ITDO gated with ILS when a tape unit fails to respond to the initial control signals to it (para 5.7.2).ELMI is also set.

(b) by RPIS gated with ILOO if the tape unit is disengaged manually after the order has been initiated. A tape unit may be disengaged at the tape unit by pressing the ‘Reset’ key or by any major failure occurring such as a tape break., ELMI is also set.

(c) by IRGWD gated with ILGW 3ms after IRGW is generated (para 5.7.5.2). ILGW is inhibited by ELMI.

This piece of logic is used during a Write Order, to set ELM and FIN if ELMI is not set by the Record Gate Logic as it is when the system is working correctly.

The Record Gate Logic will not generate StITRD (used to set ELMI) if the logic is never given any characters from Read Control. This will occur if the Reading Circuits in either the Tape Unit or Tape Control fail or if Write Circuits in the Tape Unit fail.

(d) by StIRWS/2 gated with WIWE if a Write Echo failure occurs (para. 5.4.3). In this case ELMI is <u>not</u> set.

5.7.3 The possible orders available are READ, WRITE, BACKSPACE, REWIND, REWIND AND DISENGAGE, and DISENGAGE (see para 4.2). These orders are setup in V store line VI1, digits 24-30. If the order can be obeyed it is transferred to line AIl by StITSO (para 5.7.2)

5.7.4 <u>Read</u>

5.7.4.1 <u>Start</u>

Operation is held by ISFI = until any delay to starting tape motion in the forward direction is complete (para 5.9). ITG sets the GO flip flop. After a delay of 3 or 16ms according to whether the tape is not or is at the Load Point information may be read from the tape to the Read Stats (para 5.19) StILRS allows information to enter the Stats. This delay referred to as the READ DELAY is introduced to avoid reading any unwanted noise that may be written in the inter-record gaps on tape. StILRS remains open until ILRS stat is reset by Read Disconnect Pulse, StITRD.

5.7.4.2 Finish

The Record Gate Pulse, ITRG, (para 5.13.1) is generated by the Record Gate Logic after the last character of a block has been read but before the LRCC is read. ITRG is used to reset the GO flip-flop (para 5.8) and also to set the ILRCC stat. ILRCCD is used to inhibit reading the Longitudinal Redundancy Check Character by Read Control (para 5.5). The ILRCC stat is not reset until IRELM is received from the Peripheral Co-ordinator. This is done to avoid the LRCC being detected as a spurious character by Read Control at some time later before the Read Stats have been reset.

The read order is terminated by IERD waveform (para 5,6.2.3) from Transfer Control setting ELMI.

5.7.5 Write

5.7.5.1 Start

Operation is held up by the ISFI= until any delay to starting tape motion in the forward direction is completed (para 5,9). ITG sets the GO flip flop and ITWI puts the first buffer transfer into operation (para 5.6.2.2)

To allow the tape to get up to speed and to define the inter-record gap no information is written to tape until 5 or 76msec afterwards depending on whether the tape is not or is at its Load Point. Writing is initiated by flip flip, ILWRA, the output of which goes to Write Control, and sending ITWRA to Transfer Control to test whether both Buffer Attention interrupts have been dealt with (para 5.6.2.2). At the same time that part of Read Control used to check the data writing on tape is brought into action.

5.7.5.2 Finish

2ms after StITCW is received from Write Control (para 5.4.3) the GO flip-flop is reset by IRGW. StITCW is also used to reset flip-flop ILWRA and so switch off the write counter. StITRD (para 5.13.1.1) generated after the LRCC has been read back and checked, is gated with WRITE to set ELMI.

5.7.6 Backspace

The action on receiving a backspace instruction depends on the Read/Write status of the selected tape unit,

(a) Tape Unit in Road Status:- operation is held up by ISBI until any delay to starting tape motion in the reverse direction is complete (para 5.9). The tape direction stat ILBK, is set to BACKWARDS (para 4.3.1) and the GO flip flop set by ITG. After the READ DELAY (para 5.7.4.1) data may be read from tape to the Read Stats.

Data is read until the Record Gate logic detects the beginning of the record and generates the READ DISCONNECT PULSE, StITRD, (para 5.13.1.2.). The Pulse is gated with BACKSPACE and 3ms later the GO flip flop is reset and ELMI is set by IRGB.

(b) Tape Unit in Write Status:- GO is set by ITBN and reset after 5ms by ITBNA: during this time the tape will move forward a distance greater than the inter-record gap (see para 2.4). The status of the tape unit is switched to READ STATUS by WPISR and a Backspace Operation with the Tape Unit in the Read Status is commenced.

5.7.7 <u>Rewind</u>

The action on receiving a Rewind instruction depends on the Read/ Write status of the selected tape unit.

(a) Tape Unit in Read Status:- START REWIND, WPIN, is sent to the tape unit and after about 10ms SELECT AND REWIND, RPIN, is returned. This latter signal is gated with WPIN to set ELMI.

(b) Tape Unit in Write Status:- GO is set and reset after 5ms during which time the tape will have moved forward a distance greater than the inter-record gap (sec para 2.5) The status of the tape unit is switched to READ STATUS by WPISR and a Rewind operation with the Tape Unit in the Read Status is commenced.

The REWIND IN PROGRESS, RPINU, line associated with the particular tape unit is ‘on’ while the Tape Unit is rewinding. The signal is independent of the tape unit being selected (para. 4.4.2)

5.7.8 <u>Rewind and Disengage</u>

As above except that when the tape unit has completed the rewind the tape is unloaded out of the columns and the tape unit disengages itself from the Tape Control and the Tape Unit Engaged line, RPIEU, falls to the Disengaged level.

5.7.9 <u>Disengage</u>

The Disengage line AID is gated with the selected tape unit to set one of IDU1, IDU2 or IDU3. ELME is set when the Tape Unit Engaged, RPIEU, line from the selected tape unit falls to the Disengage Level.

5.8 <u>The GO Flip-Flop, WPIGO</u>

It is set by ITG to start tape motion when obeying Read, Write and Backspace (READ STATUS) and by ITBN when obeying Backspace (WRITE STATUS). However it is only set after any GO Down Time has been completed (para 5.9).

It is reset by various waveforms according to the order being obeyed:-

Write: Reset by IRGW 2ms after WRITE CHECK PULSE, StITCW, (para 5.4.3) has been received by Central Control.

Read: Reset by RECORD GATE PULSE, ITRG, (para 5.13.1.1) gated with READ.

Backspace (READ STATUS): Reset by IRGB 3ms after Read Disconnect Pulse, StITRD, (para 5.13.1.2) has been received by Central Control.

Backspace and Rewind (WRITE STATUS) : Reset by ITBNA 5ms after setting GO.

It is also reset when reading blank tape and the tape indicator is set (para 5.10)

5.9 <u>Minimum Down Time of the GO Flip Flop</u>

5.9.1 <u>Operation</u>

When the tape direction is reversed the Drive Mechanism of the Tape Unit requires 11ms to change over. Hence the GO line to the particular tape unit should be down for at least 11ms.

In theory information must be held in control about when and what was the last order, requiring tape motion, obeyed by each tape unit before deciding whether a further order involving tape motion can be obeyed immediately. In practise this comprehensive system would require an excessive amount of hardware and so a compromise has been adopted. The normal operation of the tape system is to read or write records and Backspace only to re-read or re-write a record if its parity is incorrect.

The number of sequences of orders which will contain many changes between tape units is likely to be negligible. The compromise adopted is therefore as follows:-

Each time the GO flip-flop is reset, either of two pairs of delays are triggered according to whether the tape direction for the order just complete was Backward or Forward, the outputs of these flip flops are used to inhibit Forward or Backward motion respectively taking place within a further 11ms. The limitation of the system is that the condition is applied to the Control Unit as a whole and not, as it should be, to each tape unit separately.

<u>Note.</u> The commencement of Backspace and Rewind Orders, when the selected tape unit is in the Write Status, cannot logically be held up because the previous order must have been a Write Order. However with the present system this case is taken into account automatically.

5.9.2 <u>Logic</u>

IRGO, which resets WPIGO, triggers the two pairs of delays according to whether the tape motion is Backward or Forward. The flip flops feed delays totalling 12ms. 1ms excess of nominal 11ms is given to allow for tolerances in the delays. Two delays in series are required because the orders may occur more frequently than 12ms and a second trigger arriving within this time would be lost in the delay circuit which is an AC coupled monostable flip-flop.

$\overline{\text{ISFI}}$ = controls forward tape motion

$\overline{\text{ISBI}}$ controls backward tape motion.

5.10 <u>End of Blank Tape Logic</u>

If an attempt is made to read a non-existent block the Tape Control Unit will go on searching for the start of the block until the tape is pulled off the spool at its physical end. This logic is designed to prevent this happening by using the Tape Indicator signal RPIIND (para 4.3.2) to stop the tape before its physical end is reached. The logic also informs the Supervisor by setting the End of Operation Look at Me.

The INDIA flip-flop is set by RPIIND provided a Read Order (St RI/1) is being obeyed; that the GO flip-flop is set (WPIGO) and that the Read Stats have been open for 3ms (StILRS, para 5.7.4.1.)

Normally INDIA is reset by ITCB = (para 5.5) when a character is read from the tape or by StITRD if the end of the record has been detected (para 5.13.1.1). If INDIA is still set 3ms after being set it is assumed that the tape being read is blank and tape motion is stopped (para 5.8) and ELMI is set (para 5.11.1.) by INDI.

5.11 Operation of Look At Me, ELMI

5.11.1 Setting ELMI

Write: Set when the Read Disconnect Pulse, StITRD (para 5.13.1.1) is received from Record Gate Logic.

Road: Set when IERD (para. 6.5.2.3.) is received from Transfer Control.

Backspace: Set 3ms after StITRD (para 5.13.1.2.) is received by Central Control.

Rewind, Rewind and Disengage: Set when RPIN (para 4.3.2) is received from the Tape Unit.

Disengage TU1, TU2, TU3: Set when the RPIEU line of the selected tape unit has fallen to the disengage level (para 5,7.9 and para 4.4.2).

Tape Unit Failure: Set by ¬ITS when a tape unit fails to respond to Control signals (para 5.7.2.1)

End of Blank Tape: Set by INDI (para 5.10) when reading blank tape and the Tape Indicator is set.

5.11.2 Resetting ELMI

Reset by IRLM delayed by 1µs. The 1µs delay is included to ensure that all ELMI setting waveforms, also reset by IRELM, have dropped and that the Tape Indicator if set is reset (WPIRIND)

5.12 Reset

Most control flip flops must be reset before a new order can be obeyed. These are reset by IRELM which activates Resetting Strobes StIR/1 - StIR/13. The strobes are also activated by Console Reset, IRC. However IRC is only used when the machine is switched on in the morning.

5.13 Record Gate Logic

5.13.1. Operation

The logic detects the beginning and the end of a record. This is achieved by setting a counter going as soon as the first character is read. The counter is fed from a crystal oscillator with a pulse rate of 1Mc/s at High Density or 360 kc/s at Low Density. Every time a further character is read the counter is reset. When the end of the record is reached no more characters will be read and the counter will build up unchecked. The end of a record is defined by the counter reaching a given value. The actual value depends on the order being obeyed.

5.13.1.1 For Read and Write The RECORD GATE PULSE, ITRG, is generated when the count is 36. As the nominal number of counts between information characters is 16 (16µs High Density, 44.5µs Low Density) the pulse will be generated after the last information character has been read but before the Longitudinal Redundancy Check Character, LRCC, has been read (LRCC is nominally 64 counts after the last information character). During a Read Order the RECORD GATE PULSE resets the GO flip flop and so stops tape motion. The READ DISCONNECT PULSE, StITRD, is generated when the count is 144. This value is chosen to ensure that the LRCC has been read.

5.13.1.2 For Backspace When reading tape backwards the first character of a record that is picked up is the LRCC. Obviously it is pointless to stop then. So unlike Read and Write end of Backspace is first detected when the count has reached 114 and the READ DISCONNECT PULSE, StITRD, is then generated.

5.13.2 Logical Design.

Read Control detects a character on tape and sends a READ PULSE, StITL/3 (para 5.5) to the circuit. The pulse is held up on flip flop RG11 until a pulse arrives from the crystal oscillator, IC. RG12 is used to clean up the oscillator pulse if it should unfortunately arrive while RG11 is setting. The cleaned up pulse sets RG13 and resets RG11, RG12, and the Record gate Counter. Once RG13 is set further clock pulses are added into the counter. At High Density 800mμsec are available for the counter to be reset before the next clock pulse adds into the counter. The counter is switched off by the next READ PULSE resetting RG13; it also sets RG11 and the cycle is repeated until no further READ PULSES arrive. The counter builds up and at counts of 36 and 144 the ITRG and StITRD respectively, are generated for Read and Write orders. For Backspace only the StITRD is generated for reasons given in para. 5.13.1.2. StITRD also resets RG13 and inhibits further counting.

5.14 Character Counter

The counter is used to gate characters into and out of buffer register A. As the buffer holds 16 characters a four stage counter is used. The counter is fed by either ITCC (para 5.5) from Read Control or by ITWKA (para 5.4.3) from Write Control. The output of the counter is decoded on 16 inverters to supply gating for both Read and Write Orders. The 16 waveforms are labelled StISGOO to StISG15, StISGOO is also used to Control entry to Transfer Control (para 5.6.1)

5.15 Lateral Parity Logic

5.15.1 Operation The seventh track of the tape is used to hold a digit which is such that the overall parity is odd if recording is in the Binary Mode, or even if recording is in the BCD Mode.

The required digit is therefore

No. of ones in 6 bit char.	Parity Digit BCD	Binary
EVEN	0	1
ODD	1	0

Writing The parity digit WIBAP is generated from the 6 information digits according to the above table.

Reading The actual parity digit read from tape is compared with the theoretical parity digit generated from the 6 information digits. If these digits are not identical the Lateral Parity Failure digit, PILA, (para 4.2) is set.

5.15.2 Logical Design.

The parity of the six digits is derived by first finding the parities of three sets of two digits using LTPs. The parity is formed of two of these "2 digit" parities and this is combined with the parity of the original 3rd pair to give the parity of the whole.

5.16 Longitudinal Parity Logic

The logic determines whether the number of 'ones' read from each of the seven tape tracks is even or odd. The determination is taken over all the characters of a record except the Longitudinal Redundancy Check Character, LRCC, with which it is compared. There should be complete agreement between the two. However, if this is not so the Longitudinal Parity Failure flip flop, PILO, (para 4.2) is set. The logical design is made up of seven independent one digit counters. The counters are known as a whole as the Longitudinal Parity Register. Every time ITCB (para 5.5) is received from Read Control the character currently held in the Read Stets is "added" into the Longitudinal Parity register. The "adding" process amounts to a non-equivalence operation between the character in the Read Stats and the word in the LP register. ITCB does not occur when the LRCC is read to the Read Stats (para 5.5) so that the LRCC is not "added" into the Longitudinal Parity register. However, it is compared with the word held in the register by the READ DISCONNECT PULSE StITRD, (para 5.13.1). If the comparison fails the Longitudinal Parity Failure flip flip PILO (para 4.2) is set.

NOTE It is worth pointing out that although reading backwards is not allowed the parity checks are performed correctly when a Backspace order is carried out.

5.17 BCD Conversion Logic

Information transferred from the Buffer Register A to the Write Stats and from the Read Stats to the Register is, or is not, code converted according to whether the information is recorded in the BCD or Binary Mode.

The conversion logic for the BCD mode allows complete 6 bit to 6 bit character conversion. This general solution was necessitated because of the considerable difference between the Atlas and IBM internal codes.

The incident characters are decoded to the 64 possibilities and these are recoded to form the required code converted characters.

5.17.1 The Conversion Code is as follows:-

Atlas Internal	BCD Tape	Atlas Internal	BCD Tape	Atlas Internal	BCD Tape	Atlas Internal	BCD Tape
00	(00)	20	12	40	14	60	47
01	20	21	02	41	61	61	50
02	16	22	02	42	62	62	51
03	17	23	03	43	63	63	22
04	32	24	04	44	64	64	23
05	52	25	05	45	65	65	24
06	36	26	06	46	66	66	25
07	37	27	07	47	67	67	26
10	34	30	10	50	70	70	27
11	74	31	11	51	71	71	30
12	33	32	35	52	41	72	31
13	53	33	75	53	42	73	56
14	55	34	13	54	43	74	57
15	15	35	60	55	44	75	72
16	54	36	40	56	45	76	76
17	21	37	73	57	46	77	77

Note: The character 00 both internally in Atlas and on Magnetic Tape when recording in the BCD mode is an unallowed character but is included in the above table for completeness.

5.18 Write Stats

The Write Stats (WP11, WP12, WP14, WP18, WPIA, WPIB and WPIC) staticise the character on the Write Bus (para 4.3.1) so that it may be written to the tape by the Write Pulse (para 4.3.1) The stats are set by StITWS at Write Control timing m0 and reset by StIRWS at timing m15 (para 5.4.3).

5.19 Read Stats

Provided StILRS (para 5.7.4.1) is open the Read Stats are connected without gating to the peak sensors and so to the Read Bus (para 4.3.2). Therefore a character read on the selected tape unit is set up, as it appears from tape, in the Read Stats. The Read Stats are numbered RIBO-5 and RIBP. When the character held there has been dealt with the stats are reset by StIRRS (para 5.5)

8.8.6 Programming Notes

These notes are to be considered as supplementary to the previous paragraphs and are not complete in themselves. They are provided to help in writing the extracode programmes necessary for controlling the Tape Control Unit.

6.1 Setting orders in the V store

Orders are initiated by writing a ‘one’ to one and only one of the six order digits (line 1 digits 24-29) in the V store.

6.2 End of Operation Look At Me, ELMI. (Type 14 line 24 digit 31)

This Look at Me is set when the order has been carried out and all the information required by the extracode routine has been set in the V store (paras 4.2 and 5.11.1). No attempt is made to deal with the failures as they occur (with the one exception given in para 6.7.3e) once the order has been accepted. The digits giving information on failures are inspected as part of the End of Operation interrupt routine.

6.3 Put out End of Operation Look At Me IRELMI, (line 1 digit 30)

As IRELMI Is used to clear the Tape Control V store (para 5.12) all the required information contained in the V store must have been extracted before IRELMI is sent to Tape Control. All other digits in line 1 must be zero when IRELMI is sent to Control.

A new order should not be sent to the Control Unit within 1.5μsec of sending IRELMI.

6.4 Rewind in Progress and the Disengaged Digits (line 0 digits 28-33)

When a rewind is initiated on a tape unit the unit disengages itself, setting the TU Disengaged digit. The tape unit remains disengaged until the rewind is completed. If the 'Reset' key has not been pressed during the rewind or if the order is not a Rewind and Disengage Order the tape unit then re—engages itself and the TU Disengaged digit falls to the zero level.

6.5 End of a Read Order.

When ELMI is set at the completion of a read order a test must be made on the Character Count (see para 5.6.2.3.)

6.5.1 Character Count not zero: The count gives the number of characters to be read from the buffer. These odd characters are held in the more significant end of the buffer.

6.5.2 Character Count zero: a zero count means that the buffer is empty. This is unlike a Write Order where zero in the Character Count means that the buffer is full.

6.6. Backspacing the tape

Although reading tape Backwards is not allowed the parity checks are performed correctly when a Backspace order is carried out.

6.7. Failures

The following failures may be detected by extracode:

6.7.1. <u>Lateral and Longitudinal Parity Failures PILA PILO</u> Digits 25 & 26. Line 1 may be set, while reading from or writing to tape, due to detection of errors caused by imperfections in the magnetic tape.

6.7.2 <u>Buffer Attention Overdue</u> Digit 24 line 1 will be set if information is being destroyed due to the inability of the Buffer Attention interrupt routine to transfer information to or from the buffer in the required time, (para 5.6).

6.7.3 <u>Mechanical Failure, FIM</u> Digit 29 line 1 may be set as Follows:

(a) 5μs after an order is sent to the V store a test is made to ensure that the selected tape unit is responding to initial instructions from Tape Control (para 5.7.2). If this is not so the order is abandoned and ELMI and FIM are set. However Lateral and Longitudinal parity failures will not have occurred and PILA and PILO should not be set.
This failure may simply be due to the selected tape unit being disengaged, which can be checked by inspecting the TU Disengaged digits

(b) Sporadic Writing failure during a Write Order:
If the writing mechanism in the tape unit fails sporadically at some period while writing a block a character may not be written on tape. When this occurs no Write Echo pulse will be received in the Tape Control and FIM is set. However writing will not stop and the block will be completed before this failure will be detected.
It is very probable that either or both PILA and PILO will be set.

(c) Continuous writing failure during a Write Order:
If 3 or more consecutive characters fail to be written on tape then the Record Gate Logic will think that the end of the record has been found and will set ELMI even though the Write Order has not been completed by the extracode programme. FIM is set and so very probably are PILA and PILO.

(d) Total writing failure during a Write Order:
If the writing mechanism has failed before the first character is written to tape no characters are ever written to tape. In this case the Record Gate Logic never becomes operative and it is only entered by reading a character from tape. As the Record Gate Logic is responsible for setting ELMI when the tape system is working correctly a different method of setting ELMI is provided in this case. ELMI is set 3ms after Write Control has dropped the Write Check Character line. This provision is overridden when the tape system is working correctly.

(e) If the 'Reset' key on a tape unit is pressed while it is carrying out an order from the Control Unit, ELMI and FIM are set within about 150mμsec of the TU Disengaged line rising to the 'on' level. This condition may be diagnosed by the fact the ELM, FIM and TU Disengaged are all 'on'.
As parts of the Tape Control unit may continue functioning after ELMI is set this debris must be

cleared out of the Control Unit before any further order, even on a different tape unit, may be obeyed. This may be done by sending two IRELMI signals to the Tape Control Unit with at least a 1μsec spacing between them, (i.e. two Put Out End of Operation L.A.M. signals).

8.9 The Instruction Counter and the Clock

The Instruction Counter

This consists of eleven digits with a one being added to the least significant of these digits every time an instruction is obeyed (except when the counter is stopped) on either Main or Extracode control. An interrupt occurs every 2048 instructions, (i.e. every time the most significant digit is changed from a zero to a one) to enable the drum learning program to bring up-to-date the record of which core store pages have been used since the last such interrupt.

It is possible to set these digits from the computer.

The information digits are cleared by pressing the Reset button on the engineers console and it is necessary for the Supervisor to write a one to the Start digit of the counter every time the machine is switched on. Any of the Supervisor routines which are obeyed on Extracode control preserve the existing value of the instruction counter and reset it after being obeyed, thus allowing a separate count to be assembled of instructions obeyed by the Supervisor.

The Clock

This is a Venner Transistorised Digital Clock situated in the console. It provides five outputs:-

1) a one-tenth of a second interrupt signal. No automatic count is made of these signals. This interrupt is to enable the Magnetic Tape supervisor to check the arrival of Block Address interrupts. It is also used by the log keeping program to add up the number of tenths of a second a program has been operating.
2) a one second interrupt. The routine associated with this interrupt
    a) checks whether any peripheral equipments or magnetic tape mechanisms have been engaged or disengaged since the last such interrupt.
    b) checks for fault conditions on the peripheral equipments e.g. a "tape out" digit being set for a Teletype punch.
    c) terminates magnetic tape wind or rewind operations when these are used as fast search operations.
    d) checks that the current programs have not exceeded their specified time limits.
    e) initiates routines run at fixed intervals i.e. engineers tests.
3) a one-hundredth of a second signal. This is used by the X-ray Diffractometer but it does not cause an automatic interrupt.

4) twenty information digits read from a V-store address. Of these digits six represent the hours (two for the ten digits and four for the units) seven represent the minutes and seven the seconds (three for the tens and four the units in each case.
5) the time is displayed on the wall. This representation is in decimal form giving hours, minutes and seconds on a 24-hour clock basis. The output from the clock is

not staticised and it is arranged that the clock is not switched off when the peripheral coordinator is off. The output goes through relays which prevent any signals going to the peripheral coordinator when this is switched off.

The One-second interrupt signal to the computer does not coincide with every tenth of the One-tenth of a second interrupt signals. In order that the information digits are not read as they are being changed a delay of 200µsecs is built into the former signal. The time thus remains unchanged in the information digits for one second less 200µsecs after every One second interrupt.

Separate interrupt and Put Out Look At Me signals are provided for the two interrupts with the One-second interrupts having the higher priority. Due to the 200µ sec delay the routine for this interrupt will however normally be obeyed after the corresponding One-tenth of a second interrupt routine; also the Clock Interrupt program is written so that if both interrupt digits are set on entering the program the One-tenth of a second routine is obeyed first.

It is possible to inhibit these interrupts (but not to stop the clock) either (a) by writing a one to the Stop digit or (b) by pressing the Reset button. Writing a one to the Start digit allows the interrupt signals through to the machine. It is thus necessary for the Supervisor to "Start" the clock before it may be used.

V-Store Digits

Type 12 (*60043000)

(a) Instruction counter

line	digits		
0	46-36	RW10	Information. One is added in digit position 36 every time an instruction is obeyed except when (a) using interrupt control or (b) the counter is stopped.
	27	W1	Start
	26	W1	Put out "Look at Me"
	25	RW1	Stop

(b) Clock

line	digits		
1	45-44	R	hours, tens
	43-40	R	hours, units
	38-36	R	minutes, tens
	35-32	R	minutes, units
	30-28	R	seconds, tens
	27-24	R	seconds, units
	27	W1	Start
	26	W1	Put out 1 sec Look At Me
	25	W1	Stop
	24	W1	Put out 1/10 sec Look At Me.

Type 14 (*60043400)

line	digits		
0	24	R	Instruction Counter Look At Me
	25	R	Clock, One-tenth Second Look At Me
	26	R	Clock, One Second Look At Me

8.10.1 The On-Line X-ray Diffractometer. (OLDMAN)

1. Nature of the equipment

The equipment consists of X-ray detector(s), a crystal orienter (a goniometer), an X-ray source and a control unit.

1.1 The X-ray Detector

The detector can be either a scintillation crystal or a proportional counter. In the Manchester University installation a scintillation crystal is used. This is directly coupled to a photomultiplier tube, the output of which is fed through a pulse-height analyser to the control unit which arranges for the detected pulses to be counted.

1.2 The 4-circle goniometer

The X-ray detector is mounted on an arm (the counter arm) which can rotate about a vertical axis. This axis is called the 2Θ (two-theta) axis, and the horizontal plane described by rotation of the counter arm about this axis the 2Θ -plane. The X-ray beam (see below) intersects the 2Θ axis at right angles and is adjusted, mechanically, to lie within the 2Θ plane. The crystal is placed at the intersection of the X-ray beam and the 2Θ axis. The crystal is mounted, usually on a fine fibre, on a three-circle (i.e. a 3-axis,) device which permits the adjustment of the crystal to any orientation with respect to the incident X-ray beam. The three axes of this device intersect at a point, which is the crystal location, and permit rotation of the crystal about (i) a vertical axis, the ω (omega) axis, (ii) a horizontal axis lying in the 2Θ-plane called the χ (chi) axis and (iii) a third axis, the ϕ (phi) axis of variable inclination.

The ω and 2Θ axes are thus coincident; the chi circle is mounted on the ω axis and turns with the ω axis. The ϕ circle is mounted on the chi-circle so that the ϕ axis is vertical and coincident with the ω and 2Θ axes when the chi-angle is at 0 and 180° and is contained in the 2Θ - plane at the chi-angles of 90° and 270°. At other chi-angles the ϕ axis makes an angle whose value is chi with the ω axis.

There are thus four independent angular shaft settings required to position the goniometer and counter arm. The angular positions of each of these four circles will be sensed by Moiré fringes which permit setting accuracies (in the case of the Manchester University instrument) of 1/100 degree for the ω, χ and ϕ axes and 1/50 degree for the 2Θ axis. The instrument at Manchester University is based on a design of the U.K.A.E.A. neutron diffractometer.

1.3 X-Ray Source

This consists of a collimated beam of X-rays usually produced from a molybdenum or copper target and may be stabilised or unstabilised. In the latter case provision is made for monitoring the output of the X-ray tube.

1.4 The Control Unit

This consists of the circuits associated with (i) setting the circle positions, (ii) counting the signals obtained from the pulse-height analyser(s) of the diffracted X-ray beam (and the monitor if provided), (iii) providing two-way information paths between the Atlas peripheral co-ordinator and the diffractometer.

It consists essentially of four main registers, two of which are 20 bits and two 18 bits long. In additional there are four special registers referred to as "Clamp registers" each 6 bits long. These latter four registers form part of the four closed loop servos associated with each circle of the 4-circle goniometer.

Each of these eight registers can obtain access to a Half-Adder/Subtractor for the purpose of adding or subtracting one from the specified register.

The peripheral co-ordinator is able to write to, or read from, any of the eight registers.

1.5 The V-store Digits

Type 9

Line 5 (*60042250)

Write		Read	
Digit		Digit	
24-43	Information	24-43	Information
44	Load Information (LIv)	44	Reflected Count Overflow
45	Start Count (SCv)	45	Check read
46	Set the Position (SPv)		

Line 9 (*60042310)

Digit			
24	Disengage	24	Disengaged
25	2Θ (Two-theta) (2Θv)	25	2Θ
26	ω (Omega) (Wv)	26	ω
27	Φ (PHI) (Xv)	27	φ
28	X (CHI) (Xv)	28	χ
29	Z-line (Zv)	29	Clamp out of range
30	V-Reset	30	Limit switch failure
31	Clamp registers* (CRv)	31	References reached
32	Read selected register (RIv)	32	Test A
33	Put out position set L.A.M.	33	Test B
34	Put out count complete L.A.M.	34	Test C
35	Put out contingencies L.A.M.	35	Test D
36	Open shutter	36	Test E

37 Close shutter

38 Drive ω - circle
(CWDRv)

39 Drive 2Θ - circle
(2ΘDRv)

40 Drive to references
(DRv)

Line 13 (*60042350)

Digit

24 Test and isolate motors and nucleonics (TIMN)

25 Reset test and isolate motors and nucleonics (RESET-TIMN)

26 Test-count-start * (TCS)

27 Test-count-stop * (TCst)

28 Test set-add * (TSA)

29 Test set-subtract * (TSS)

30 Test add ± 1 *

31 Test fringes (TFv)

Type 14

Line 6 (*60043460)

25	Position Set	Look At Me
27	Count Complete	Look At Me
29	Contingencies	Look At Me

1.6 The Z-line

The Z-line consists of ten staticisors which are accessible by writing to and reading from digits 24-43 of line 5 of type 9 of the Peripheral V-store. This is a special line associated with controlling the X-ray beam, i.e. it enables the beam to be centred using four controls and it provides for the insertion or removal of filters, attenuators, etc.

Writing a one to digit 29/9 causes the subsequent information sent to line 5 to be transferred to the Z-line, the transfer taking place when a one is written to digit 44/5.

The digits in the Z-line are:

Digit	Write	Digit	Read
24	Set window 1	24	Window 1 in
25	Reset window 1	25	Window 2 in
26	Set window 2	26	Window 3 in
27	Reset window 2	27	Window 4 in
28	Set window 3	28	Balanced filter 1 in
29	Reset window 3	29	Balanced filter 2 in
30	Set window 4	30	Attenuator 1 in
31	Reset window 4	31	Attenuator 2 in
32	Set balanced filter 1	32	Counting clock
33	Reset balanced filter 1	33	Motor speed 1
34	Set balanced filter 2		
35	Reset balanced filter 2		
36	Set attenuator 1		
37	Reset attenuator 1		
38	Set attenuator 2		
39	Reset attenuator 2		
40	Count clock		
41	Count X-Rays		
42	Set motor speed 1		
43	Set motor speed 2		

8.10.2 Basic Operations

There are seven basic operations performed by the device. These are:-

(1) Clamping
(2) Loading information
(3) Reading information
(4) Setting a position
(5) Counting
(6) Driving to and between circle reference markers
(7) Testing

All the V-addresses associated with this device are Type 9. Consequently only digit and line numbers are referred to (e.g. 33/9 means digit 33 of line 9).

2.1. Clamping

There are four clamp registers, one associated with each of the ω, 2Θ, ϕ and χ registers. Each register can be up to 6 bits long. Of these the most and the next-most significant digits are a guard and sign bit respectively. The remaining bits are used to clamp to circles to their set positions within 15 quarter fringes (i.e. 15/100°). Thus, any circle can be +15/100° to -16/100° away from its set position without the device losing count of this position. The clamp registers are used in conjunction with the circle servos to attempt to keep the clamp registers reading 0, this indicating zero clamping error.

Clamping is always operative except when

(1) the circles are being set to new positions,
(2) the circles are being driven to or between references and
(3) an X-ray intensity measurement is being made which involves the movement of the ω and/or 2Θ circles, during the counting.

After setting a position, driving to a reference marker or taking a count, clamping again becomes operative.

If the sign and guard bits of the clamp register take different values, then the clamp becomes 'out of range', i.e. the system will lose the positional information of the circle or circles concerned. This is indicated to the Atlas by giving a 'Contingencies' Interrupt and setting 29/9, together with one of 25/9->28/9 to 'one'. The Contingencies Interrupt routine identifies the cause of the interrupt by inspecting 29/9, 30/9 and 31/9, and the circle concerned by inspecting 25/9->28/9. Writing to 35/9 puts out the 'Contingencies' L.A.M. and resets 29/9 and 25/9->28/9 to zero. It also resets 30/9 and 31/9 and hence these must be checked for a further interrupt before writing to the Put Out L.A.M. digit.

2.2. Loading Information.

Information can be written from Atlas into any one of the four registers ω, 2Θ, ϕ and χ and also into the 'Z-line'. This is achieved by setting 44/5 which causes the information present on 24/5->43/5 to be written into the register specified by 25/9->29/9. 30/9 must also be set along with one of 25/94 29/9; this digit resetting 25/9->29/9 before they are set. The information is loaded through the Half Adder/Subtractor buffer, at the input gate of which it exists for 4μs. The 'Load information' digit 44/5 is regarded as an internal OLDMAN interrupt, and is thus dealt with as soon as the Rhythm Unit is free. This free condition is guaranteed to occur not more than 450mμs after the load information digit is set; the actual loading of information will take about 200mμs, thus if the information is available for at least 1μs it will be loaded correctly. The normal peripheral co-ordinator operation will ensure that it is available for 4μs.

2.3. <u>Reading Information Into Atlas.</u>

Information can be read into Atlas from any of the eight registers.

Two V-orders are needed to read information into Atlas.

(1) Write to 32/9 specifying Read, together with one of 25/9->29/9 selecting the register to be read (30/9 must be set for reasons given in 2.2.)

This order must be immediately followed by the order

(2) Read Line 5. The information will be present on Digits 24-43 together with a check read Digit (45) which is always set to one on reading.

The rhythm of the read information sequence is that the setting of 32/9 (read selected register) is regarded as an internal OLDMAN interrupt which will be dealt with not later than 450mμs after setting 32/9. The Rhythm Unit then makes available at the Peripheral Co-ordinator, for a period of about 10μs, the information contained in the register selected by 25/9->29/9.

2.4. <u>Setting A Position.</u>

The settings of the four circles ω, 2θ, φ and χ are known by the Atlas. (Each circle has at least one well-defined reference point from which the Atlas can compute positional information).

Setting a position involves loading the four main registers with four differences, these being the differences between the present settings and the desired settings. Positional differences are 18-bit binary numbers where the most significant digit specifies the sign of the difference and the remainder the magnitude in steps of 1°/100 for the ω, φ and χ circles, and of 1°/50 for the 2θ circle.

The differences are loaded in the manner specified in Section 2.2.

Setting 46/5 during the loading of the last difference information causes the Rhythm Unit to move all the circles by the indicated differences to set the new positions.

Setting proceeds at full speed when the difference exceeds $|2^5|$ and at a reduced speed when $< |2^5|$. This system is used to avoid unnecessary overshoots when the position is being set.

When a circle has been set, it is returned to the clamping state.

When all the circles have been set, a 'Position Set' L.A.M. is given to the Atlas. This is put out by writing to 33/9.

2.5. Counting.

Several methods of counting the diffracted quanta from a crystal can be employed.

In all cases the counting can be for a defined time (Count Clock) or for a defined Monitor Count (Count X-Rays). The method used is selected by digits 40 or 41 of the Z-line.

The count may be made with

(1) All circles stationary
(2) ω circles moving at one of 2 speeds
(3) 2Θ " " " " " " "
(4) ω and 2Θ circles moving at one of 2 speeds, these being in the ratio ω:2Θ of 1:2.

The actual counting method to be used is selected by setting appropriate V-digits before the commencement of the count. In all cases a positive number must be loaded into the χ register to specify either counting time or monitor count. This number is a 20-bit positive number.

If Counting Clock, then this number will be reduced by one every 1/100 sec. If Counting X-Rays it will be reduced by one every time a pulse is received from the Monitor circuit. In both cases counting will continue for as long as it takes for the contents of the χ register to be reduced to zero. (This count number must obviously be loaded after the circles have been set, and is done as indicated in Section 2.2.).

If the count involves movement of the ω and/or 2Θ circles, then a record of the angles through which these circles are moved during the counting is kept in the ω and 2Θ registers. This information can be read by the Atlas in order for it to up-date its circles-position log.

Counting is initiated by setting 45/5. (Normally in the same V-order 24/5->43/5 and 44/5 will be set to load the count number).

Prior to initiating the count digit 40 or 41 of the Z-line must have been set to count clock or X-Rays, also if the count involves movement of the ω and/or 2Θ circles digit 42 or 43 of the Z-line must have been set to determine circle speeds.

The Rhythm Unit first inspects 38/9 and 39/9 to determine whether or not to move the ω and/or 2Θ circles. Digit 36/9 must have been set to open the shutter. The shutter can be opened at any time but counting will not commence until the shutter actually has opened; closure of the shutter is done automatically when the count ends or by writing to 37/9.

Counting stops when the contents of the χ register reach zero. If the ω and/or 2Θ circles have been moved during this time then they are stopped and clamped in the new position. 38/9 Drive ω -circle and 39/9 Drive 2Θ circle are reset to zero and the shutter closed.

A 'Count Complete/ L.A.M. is given. This is put out by writing to 34/9.

If during the counting an overflow occurs in the ϕ register (i.e. 21st bit) then 44/5 is set. This may be read when reading the count information to determine the validity of the count. This digit is reset to zero at the commencement of a count.

2.6. Referencing.

Since all the position setting is done by driving the circles through incremental differences, at least one well defined point per circle must be provided to permit the conversion of relative positions to absolute positions. This is achieved by providing at least one reference point per circle. The accuracy of the reference points will be at least as good as that of the setting mechanism (i.e. to better than 1/100 degree).

The reference points will be placed at one end of the limit of travel of each circle, a small distance away from limit switches; also, a small distance away from the reference point on the operational side will be a reference marker. This marker is used to indicate whether or not a circle is between the reference marker and the reference point. Setting 40/9 to a one causes all circles to be driven to their reference points.

If when referencing commences, the circle is outside the reference marker then it will drive at high speed until it crosses the reference marker when it will be switched to low speed. If when referencing commences it is inside the reference marker then it drives only at low speed. This is done to prevent the overshoot, which will be put into the clamp-register, from putting the clamp out of range which in turn would cause the circle to clamp on a point other than the reference point itself.

When all the circles have reached the reference point then a 'Contingencies L.A.M.' is given and 31/9 is set to 'one'. Atlas identifies the 'Contingencies L.A.M.' as being 'References Reached' by reading 31/9 as a 'one'.

The incremental differences through which the circles have been moved to reach their reference points are stored in the four registers ω, 2Θ, ϕ and χ and are consequently available to be Read (Section 2.3) by the Atlas in order for it to check its position log.

2.7. Testing.

Several V-digits are provided to permit testing (by programme). The testing is mainly confined to the Central Unit of the device, but does include a facility to permit the testing of the Moiré Fringe Systems.

V-digits marked thus (*) are only operative when 24/13 is set to one. This digit when set isolates the motors (i.e. the circles are free), inhibits internal OLDMAN interrupts from the circle fringes and isolates the Nucleonics circuitry. In this state special signals from a Test Programme can be fed into the Rhythm Unit through the Test V-digits to facilitate the testing of the Central Unit.

Information can be written to and read from any of the four main registers in the manner already presented in Sections 2.2 and 2.3 In addition, in the Test state, setting 31/9 together with one of the 25/9->28/9 permits writing to and reading from the clamp registers.

In the Test state all L.A.M.'s are inhibited, and a number which puts a clamp register out of range can be written to check that the appropriate digits (25/9->29/9) are set to one thus indicating to the Atlas the out of range state.

All information which is being loaded sets Tests A,B,C and D thus if suitable numbers are loaded then the Test A,B,C and D digits (32/9-35/9) will be set, enabling the Tests to be checked.

Testing of the Half Adder/Subtractor is achieved by use of 30/13. Each time this digit is set, +1 is added to or subtracted from the contents of the 2θ register according to whether 28/13 or 29/13 is set to one. The operation takes about 450mμs and thus there is no limitation on the frequency of V-orders to add +1.

Setting 26/13 starts the system counting in any four registers specified by 25/9-> 28/9 and 31/9 i.e. all 4 main registers or all four clamp registers. The counting pulses are generated from a 2μs oscillator, thus +1 gets added to or subtracted from (according to 28/13 and 29/13) the nominated registers every 2μs. This operation is used to test the time-sharing control of the system. (A manual adjustment of the frequency of the oscillator will be provided to check the time-sharing limits of the system). This counting continues until 27/13 is set. The total counts are left in the registers for reading into Atlas.

At any time, setting 25/13 returns the system to the operational state, the circles clamping in whatever position they happen to be.

31/13, the Test Fringes digit, is used in conjunction with 40/9, the drive to references digit, to test the operation of the fringe system. The system should be driven to the reference points and when it is at the reference points, if 31/13 is set, all circles will drive to a second reference point counting fringes as they go. The counts obtained between references should be constant. When the second references are reached the circles are stopped, but the clamps may go out of range as no provision is included for approaching the second references at slow speed. Thus absolute positional information may be lost. A Contingencies L.A.M. is given which is identified as 'Second References Reached' by 31/9 being read as one. A new absolute position is found by driving to references by setting 40/9.

2.8. <u>Extras</u>

Limit switches are provided on all the 4-circles of the goniometer. Limit switches should not be activated in the normal course of operation of the instrument. When a circle hits a limit switch then the supplies to the motors of all circles are cut off and a 'Contingencies L.A.M.' is given. Investigation of 29/9->31/9 will reveal that there is a limit switch failure. The equipment should be disengaged by writing a one to 24/9 (which inhibits all L.A.M.'s) and the engineer informed. A manual engage/disengage button will be provided following standard peripheral equipment procedure, the state being indicated to Atlas by 24/9 which is set to one when the equipment is engaged. This digit reads one for engaged rather than disengaged as on the other peripherals because cable drivers and receivers are used instead of the standard Atlas connections.

8.11 GRAPHICAL OUTPUT

This equipment provides a means of displaying, in graphical form, the output from a computer. Information in the form of illuminated spots is sent from the computer to a Cathode Ray Tube, which has a display area of 8cm.sq. The resultant display is photographed on 35m.m. film by a Shackman camera. The display can have a maximum array of 1024 x 1024 illuminated spots with 1024 spots on the x-axis, which is horizontal, and 1024 spots on the y-axis, which is vertical. Each illuminated point has a diameter of 0.4m.m. with a "bright up" time of 1µsec. The camera has to be wound-up manually but signals from the computer control the opening and closing of the shutter and winding on of the film.

Each reel of film contains 200 frames. There is no mechanical counter showing the number of frames not yet exposed.

Controls and Indicators.

a) Power ON/OFF switch and Mains on indicator
   This is an ordinary 'tumbler" switch, similar to the power ON/OFF switch on the teletypes. The indicator is lit RED when the mains power is on,
b) ENGAGE/DISENGAGE switches and indicators
   These are two single action switches and indicators. The equipment can only be engaged by push-button but may be disengaged either by push-button or by a signal from the computer. When the equipment is engaged the Engage button is lit Green and the Disengage button is lit White; when the equipment is disengaged the Engage button is lit White and the Disengage button is lit Red.
c) CAMERA FAULT indicator
   This indicator is normally unlit but, when a Camera Fault exists it is lit Yellow. The fault condition exists when
   - (i) the camera is not fitted
   - (ii) the camera has no film
   - (iii) the film is jammed in the camera
d) SHUTTER OPEN indicator
   This indicator is in darkness when the camera shutter is closed and it is lit Yellow when the shutter is open.

The V-store digits:

Peripheral V-store	Type 3.	(*60040600)	
line	digit		
0	27	RW1	Read: Camera Fault Write: Start
	26	RW1	Read: Brilliance Write: Put Out Look At Me
(Command Register)	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage
8	47-38	W1	x - co-ordinate (in binary, digit 38 least significant)
(Transfer Register)	35-26	W1	y - co-ordinate (in binary, digit 26 least significant)
	25	W1	Display
	24	W1	Dim

Look At Me	Type 15	(*60043620)	
line	digit		
2	24	R	Look At Me

Signals to and from the computer

To engage the Graphical Output it is necessary to press the Engage button; it is not possible for the equipment to be engaged directly by the computer. Pressing the Engage button sends a pulse which resets the Disengaged digit in the V-store (digit 24) to zero.

On writing a one to the Start digit (digit 27) the following events occur:

(i) the Stop digit is reset to zero.
(ii) the film is wound on one frame. The camera shutter is opened (if closed)
(iv) the Brilliance digit is reset to zero.
(v) after a delay of 1/8th of a second a Look At Me signal is sent to the computer.

On writing a one to the Stop digit (digit 25)

(i) the Stop digit is set to read one.
(ii) the camera shutter is closed.

A point is "plotted" on the C.R.T. by sending its (x,y) co-ordinates and simultaneously writing a one to the Display digit. The effect is to "Bright-up" the specified spot for 1μsec. Two intensities of illumination are possible, namely "Bright" & "Dim". "Bright" illumination is automatically selected unless a one is written to digit 24 in the same instruction that the x, y coordinates and the Display digit are transferred. It is necessary for the program to plot any axes that are required.

Points may be illuminated on the C.R.T. at any time but they can only be photographed when the camera shutter is open.

The Graphical Output can deal with information as fast as it can be transferred from the computer to the V-store (maximum rate, one reference every 5μsecs) and there is no maximum time interval between sending commands or information. Consequently the computer plots points on Extracode control so that other interrupts may be dealt with as they occur.

No provision is made for indicating to the computer whether the camera shutter is opened or closed or the film has moved on after a start command (except when the film is jammed and the Camera Fault digit is set). A Look At Me signal is however given 1/8 sec, after a Start signal is given and it is anticipated that the specified mechanical action will have been completed in this time.

After all the points for one frame have been plotted and before the film is wound on, the computer reads the "Brilliance" digit (digit 26). This is used in conjunction with circuits inside the C.R.T. to determine whether a certain number of Bright-up pulses has been detected on the face of the tube. If an insufficient number of pulses has been detected the Brilliance digit is not set. This latter digit is automatically reset on writing a one to the Start digit.

Digit 27 is read as one if a Camera Fault condition exists. This digit is automatically reset when the fault is cleared.

9.2 Overlapping of Instructions

The times for carrying out the various accumulator operations are given in the previous section. These however are for the accumulator operations only and do not include store access time for the instruction and operand, decoding the instruction, or B-modification. The time taken to complete an instruction is dependent on:-

(i) the type of instruction; defined by the function digits

(ii) the exact location of the instruction and operand in the core or fixed stores as this affects the access time

iii) whether the operand address has to be B-modified

(iv) the actual numbers themselves for floating point accumulator orders

(v) whether drum and/or tape transfers are taking place.

The approximate times in microseconds for various instructions are as follows:-

Type of Instruction	Number of modification of address	Instructions and operands in core store	Instructions in Fixed Store operands in Core Store	Instructions and operands in Fixed Store
Floating Point Addition	0 1 2	1.4 1.6 2.03	1.65 1.65 1.9	1.2 1.2 1.9
Floating Point Multiplication	0, 1 or 2	4.7	4.7	4.7
Floating Point Division	0, 1 or 2	13.6	13.6	13.6
Add contents of Store Line to a B register	0 1	1.53 1.85	1.65 1.85	1.15 1.85
Add contents of B-register and store line and rewrite to store line	0 1	1.63 1.8	1.65 1.7	- -

The above figures relate to the times between completing instructions when a long sequence of the same type of instruction is being obeyed. This way of quoting times is necessary because in practice one instruction is overlapped in time with some part of three other instructions. If no overlapping were performed a sequence of single B-modified accumulator addition instructions for example) could take 6.05μsecs for each instruction to be obeyed. To obey such an instruction the central computer makes two requests to the core store, one for the instruction and one for the operand. After the instruction is received the function part has to be decoded and the operand address modified by the contents of one of the B registers before the operand request can be made. Finally after the operand has been obtained the actual accumulator addition takes place to complete the instruction. An approximate timing schedule for this operation is given below:

Sequence	Time Interval between steps in μsecs	Total Elapsed time, μsecs
1. Add one to main control (Addition time)	0.3	0
2. Make Instruction Request (Transfer times, equivalence time, and stack access time)	1.75	0.3
3. Receive Instruction in Central Computer (Load register and decode)	0.2	2.05
4. Function decoding complete (Single address modification)	0.85	2.25
5. Request operand (Transfer times, equivalence time, and stack access time)	1.75	3.10
6. Receive Operand in Central Computer (Average floating point addition including shift, round off, and standardise)	1.1	4.85
7. Addition complete, copy answer to accumulator (Load register)	0.1	5.95
8. Instruction complete		6.05

(Step 4. above gives the time for single modification of the operand; zero or double Modification take 0.25 and 1.55μsecs respectively).

Thus if one instruction is not started until the previous one has been completed the average time to obey such an (a+s) instruction is 6.05μsecs; if, however, more than one instruction is being carried out simultaneously the total time to perform a sequence of instructions, and hence the average time per instruction, can be reduced. In the limit the average time for an instruction might therefore be reduced to the time taken to carry out the part involving the longest component in the total time. Various facilities have been built into the machine to reduce the total time needed to carry out a sequence of instructions. There are, of course safeguards when for example an instruction is dependent in any way on the completion of a preceding instruction.

In the time sequence given above by far the longest component is that between a request in the central computer for an instruction or operand from the core store and the receipt in the central computer of the information from that store. This effective "access time" is 1.75μsecs made up as follows:-

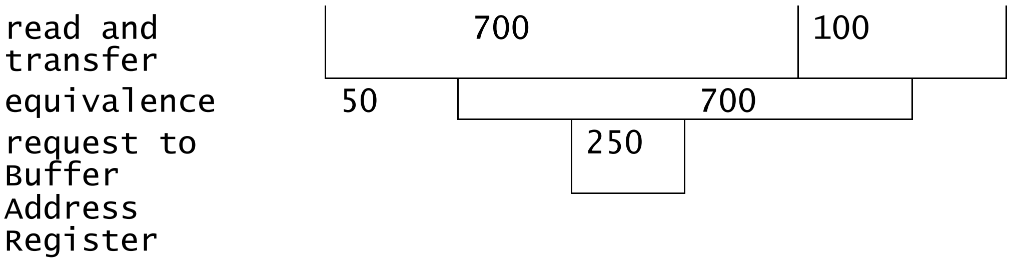
Sequence	Time taken	Elapsed time
1. the request is in the machine		0
2. the request is sent to the core store co-ordinator	0.25	0.25
3. the equivalence operation is complete and a request made to the selected stack	0.7	0.95
4. the information is in the core store co-ordinator	0.7	1.65
5. the information is in the central computer	0.1	1.75

The normal engineering definition of access time relates to the time required to extract information from the core store, and with this definition the access time is 0.7μsecs as given in step 4 above.

The difference between the 0.25μsecs sending the request to the core store co-ordinator and the 0.1μsecs receiving the information from the co-ordinator is due to the initial store t pe [?] address decoding necessary.

The effective access time of 1.75μsecs is reduced in practice because of the addressing system (i.e. two instructions are taken from the core store simultaneously to the Present Instruction Even and Present Instruction Odd registers) and also by the provision of two buffer registers, one in the central computer and the other in the core store co-ordinator. These latter registers allow the equivalence and transfer times to be overlapped with the organisation of requests in the central computer, and, provided the machine can make requests fast enough the effective access time is reduced to 0.?μsecs.

The first of these registers, the Buffer Address Register, stores the address requested by the central computer and therefore permits the computer to be doing further operations during the remainder of the access time i.e. whilst waiting for the information it has just requested. The second, the Core Store Address Register, holds the result of the equivalence operation during the time that it is needed by the core store. This enables a second equivalence operation to be carried out during the read phase for the previous request. Also, after the equivalence operation has commenced a further request may be sent to the Buffer Address Register and this address is held there until the necessary equivalence operation can be performed on it. This overlapping is illustrated in the diagram below:-



For the first instruction 800mµsecs are taken to read the information from the core store and transfer it to the central computer.

For the second instruction the equivalence operation can be commenced 50µsecs after starting to read the information from the core store for the previous instruction and it is completed before this information has reached the central computer.

For the third instruction the request for information can be stored before the previous equivalence operation is complete.

Further, as three accesses are needed to complete two instructions (one for the instruction pair and one each for the operands) the theoretical minimum time for an instruction is 1.2µsecs (3 x 0.8/2) and the machine then becomes store limited. Thus for arithmetic operations taking 1.2µsecs the capabilities of the store and the accumulator arc well matched.

A further method of reducing effective access time is to assume that successive instructions are from the same page and to store the address of the instruction being obeyed. For the next instruction the equivalence operation is started as usual but the reading operation is also commenced by copying the result of the last equivalence operation into the Core Store Address register. If it is found that an instruction from a different page is required the read operation must be restarted on the new address. This enables the read operation to be started 400mµsecs earlier than would otherwise be possible. It is, however, only effective when the limiting factor is not the core store itself i.e. if the time taken is dependent on the core store only (effectively taking 1.2µsecs per instruction) no further gain in speed can be made. If, however, other factors limit the speed, such as an accumulator operation taking more than 1.2µsecs, some or all of this 400mµsecs might be saved.

A schematic diagram of the practical timing of a sequence of floating point addition orders is given below. The overlapping is not perfect because it is not always possible to know beforehand when an operation is about to finish and to have another operand available to use the same circuits immediately. From this table it can be seen that in the time between successive instruction pairs the computer is obeying four instructions for 25 per cent of the time, three for 56 per cent, and two for 19 per cent. It is therefore to be expected that the practical time for the complete order is greater than the theoretical minimum time; it is in fact approximately 1.6µsecs.

For certain types of function the reading of the next pair of instructions before completing both instructions of the first pair would be incorrect, e.g. functions causing a control transfer. Such situations are recognised during the function decoding and the request for the instruction pair is held up until a suitable time.

In a sequence of floating point addition orders with the operand addresses unmodified the limit is again 1.2µsecs whereas the time taken is 1.4µsecs. For accumulator orders in which the actual accumulator operation imposes a limit in excess of 2µsecs (e.g. multiplication) the actual time taken is equal to this limit.

A more realistic way of defining the speed of the computer is to give the time for a typical inner loop of instructions. A frequently occurring operation in matrix work is the formation of the scalar product or [of] two vectors, this requiring a loop of five instruction:-

- (i) copy an element of the first vector into the accumulator, the operand B-modified,
- (ii) multiply the contents of the accumulator by an element of the second vector, the operand B-modified,
- (iii) add the partial product into the accumulator,
- (iv) copy the new partial product from the accumulator to a store line,
- (v) test and alter the counter for the next element and repeat.

The time for this loop with instructions and operands in the core store is 12.2µsecs. If this loop is obeyed as an extracode with instructions in the fixed store the time is 10.5µsecs.

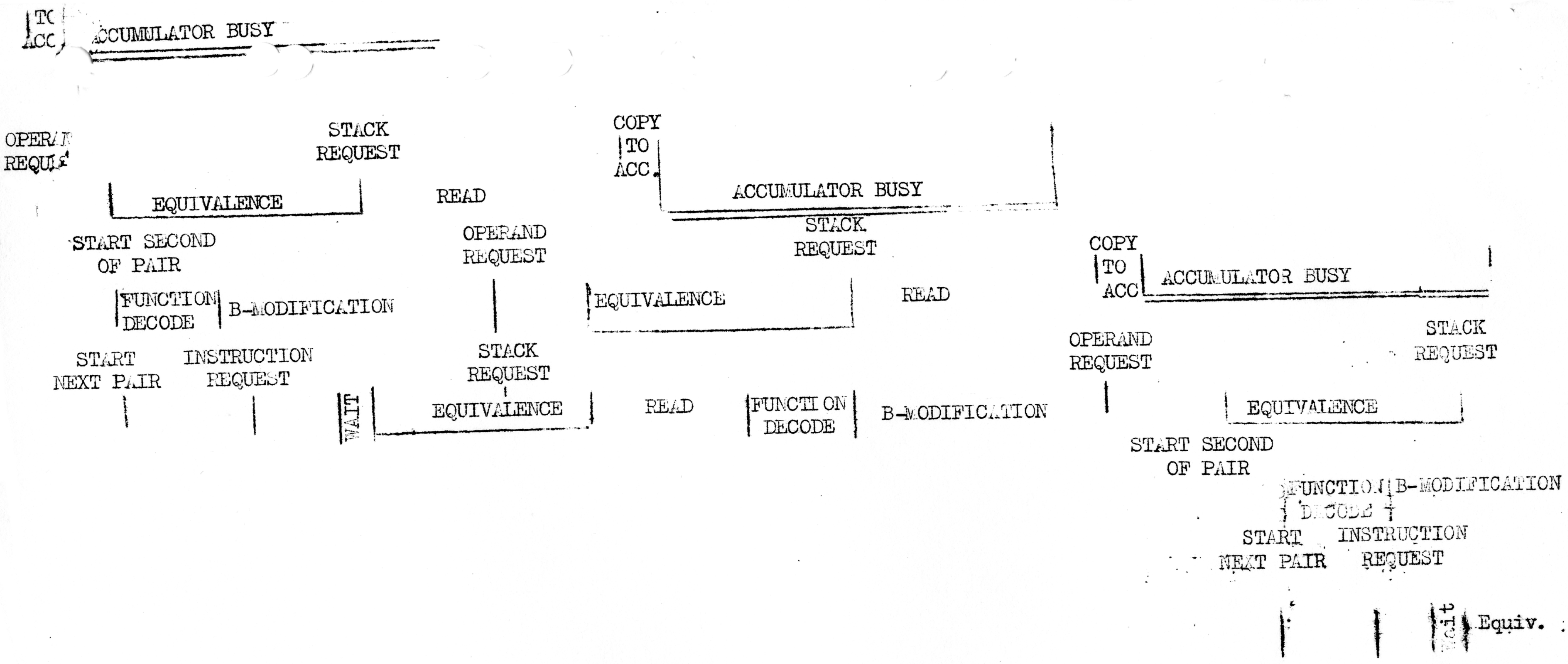
A second common operation is the evaluation of a polynomial. The inner loop here contains three orders

- (i) add a number into the accumulator,
- (ii) multiply the contents of the accumulator by a number from a store line,
- (iii) test and alter the counter for the next element and repeat.

The times for this loop are 6.3µsecs and 5.9µsecs for the instructions in the core store and the fixed store respectively.

A third example is for 16 consecutive B-modified B orders which occur in the compiler routine. Here the times are 29.1µsecs and 28.9µsecs for the instructions in the core store and the fixed store respectively.

When the magnetic drum or tape systems are transferring information to or from the core store the rate of obeying instructions which also use the core store is affected. The degree of slowing down is dependent upon the time at which a drum or tape request occurs relative to machine requests. It also depends on the stacks used by the drum or tape transfer and those being used by the central computer. The approximate slowing down is by a factor of 25 per cent during a drum transfer and by 2 per cent for each tape channel.



Timing Diagram for a Sequence of Floating Point Addition Orders

(Single Address Modification)

Section 10 Engineering Facilities

10.1 The Engineer's Console

A control panel for engineering use only is provided which contains the following controls and indicators:-

1) Display

A set of 24 lights which are addressed as B120 for output purposes. If an attempt is made to read from B120 a zero operand is obtained.

2) Parity Circuit On/off Buttons

There are six parity circuits in the computer but these do not have a one to one correspondence with the six parity interrupts. For each circuit there is a double action push button for switching the circuit on and off. There is a light behind each button which is lit when the circuit is switched off. The six buttons are for

a) Parity A
   This circuit checks all instructions with even addresses read from the core store, all operands read from the core store, and the information read from the core store being written to either the magnetic tape or the drums,
b) Parity B
   This circuit checks all instructions with odd addresses read from the core store and all information written to the core store from the drum and tape co-ordinators. It also generates a parity for information written by the machine to the core store.
c) Parity C
   This checks instructions and operands read from the subsidiary store.
a) Parity D
   This circuit generates parity for information written to the subsidiary store.
e) Parity E
   This circuit checks instructions and operands read from the fixed store.
f) Parity F
   This circuit checks the output of the magnetic tape buffer on writing to tape. It also generates parity for information read from magnetic tape,

3) Parity Indicators

There are thirteen lights which indicate whether incorrect parity has been detected on reading

a) an instruction with even address from the core store
b) an instruction with odd address from the core store
c) an operand with even address from the core store
d) an operand with odd address from the core store
e) information from the core store for transfer to a drum
f) information from a drum for transfer to the core store
g) information from the core store for transfer to the tape co-ordinator
h) information from the tape buffer for transfer to magnetic tape
i) information from the tape buffer for transfer to the core store
j) an instruction from the subsidiary store.

k) an operand from the subsidiary store

l) an instruction from the fixed store

m) an operand from the fixed store.

Where an instruction or 48-bit operand is involved the parity digit is checked for each 24-bit half word. If a 24-bit operand is called for, the parities of that half word and also the other half word in the 48-bit word are checked.

4) <u>Parity Lamp Reset</u>

The parity indicator lights listed above remain lit after the appropriate interrupt flip-flops have been reset by programme. A push-button with a light behind it which is always lit when the machine is switched on is provided which puts out any indicators which are lit when it is pressed.

5) <u>Manual Order Handkeys</u>

Forty eight keys are provided to enable a manual order to be carried out. They are arranged in two sets of twenty four with the upper row comprising ten keys for the function, seven keys for Ba and seven for Bm. The lower twenty four keys are for the operand or the address of the operand.

6) <u>Operand Buttons</u>

Eight push buttons are provided which can be read as digits 31-24 of line 7 of the Central Computer V-store. These are double action buttons with lights behind them and the light is on when the digit is read as a one.

7) <u>Engineers Interrupt Button</u>

This is a single action push button with a light behind it which is always lit when the machine is switched on. On pressing the Engineers Interrupt button three actions take place

a) a digit in the Central Computer V-store (digit 27, line 5) is set to one and the Engineers Reader and Punch switched on.

b) the address of the first instruction in this Engineer's Tests (2560 in the Fixed Store, total 4005,0000) is sent to Interrupt Central*.

c) control is switched to Interrupt Control. No record is kept of the state of the I/ME digit and hence it is not normally possible to resume any programmes that were being carried out when the button was pressed.

Exit from the Interrupt is by programme depending on a pre-determined number being set on the Engineer Handkeys. The Interrupt is cancelled by re-setting the V-store digit by programme, which automatically switches the Engineers Reader and Punch off, and transferring control as required.

8) <u>Stop Half Adder Button</u>

Under normal circumstances, whenever an instruction is obeyed, one is added to the current control number and the next instruction is extracted either from the appropriate store or from the Present Instruction Odd register. The Stop Half Adder button is a means of preventing the one being added and the next instruction being extracted. A light behind this button is lit when in the Stop state.

9) <u>Auto Rate/Manual Rate</u>

This is a double action push button with lights behind the upper and lower halves of it. The light behind the upper half is lit when the computer is obeying instructions at the normal speed (Auto Rate). The light behind the lower half is lit when it is required to obey instructions not at full speed (Manual Rate).

* Probably ‘Interrupt Control’.

10) <u>100 K.C/S/Stop</u>

When on Manual Rate instructions may either be obeyed singly (i.e., by giving a single prepulse when on Stop) or so that one instruction is obeyed before the next one is started (100 K/Cs). This again is a double action button with either the top or bottom half lit.

11) <u>Single</u>

When the previous two buttons indicate Manual Rate and Stop respectively, a prepulse is given and an order obeyed each time the Single button is pressed. The light behind this button is on all the time when the machine is switched on.

12) <u>Auto Order/Manual Order</u>

This is a double action push button with lights behind the upper and lower halves. A light behind the upper half (Auto Order) indicates that orders are being obeyed from the store. A light behind the lower half (Manual Order) indicates that the order currently on the handkeys is being obeyed. In either case the speed at which orders are being obeyed is governed by the Auto Rate/Manual Rate button,

13) <u>Reset</u>

Pressing this button resets various rhythm controlling flip-flops throughout the machine. It also resets the contents of B120 (Display) to zero. It is normally used only when the machine has just been switched on. This button has a light behind it which is lit when the machine is switched on,

14) <u>Start</u>

When the machine is switched on the various flip-flops are set in an arbitrary manner and also one or more spurious prepulses might be in the machine. It is therefore necessary to switch to Stop (Auto/Manual to Manual, and Stop) to get rid of the spurious prepulses, to press the Reset button, to press the Engineer's Interrupt button, to switch back to Auto and to press the <u>Start</u> button. Conditional upon the number set up on the Engineer's Handkeys various initial test programmes are obeyed and the necessary flip-flops set to a standard state (e.g. the Look at Me's are reset).

The button must be pressed to give a prepulse on switching back from the Stop state to Automatic Rate. There is a light behind this button which is lit whenever the machine is switched on.

15) <u>Prepulse Volume Control</u>

This controls the volume of sound from the hooter of a steady note generated by an oscillator. The oscillator is rendered operative when prepulses are being obeyed continuously (at more than 10 Kc/s).

16) <u>V/Address Volume Control</u>

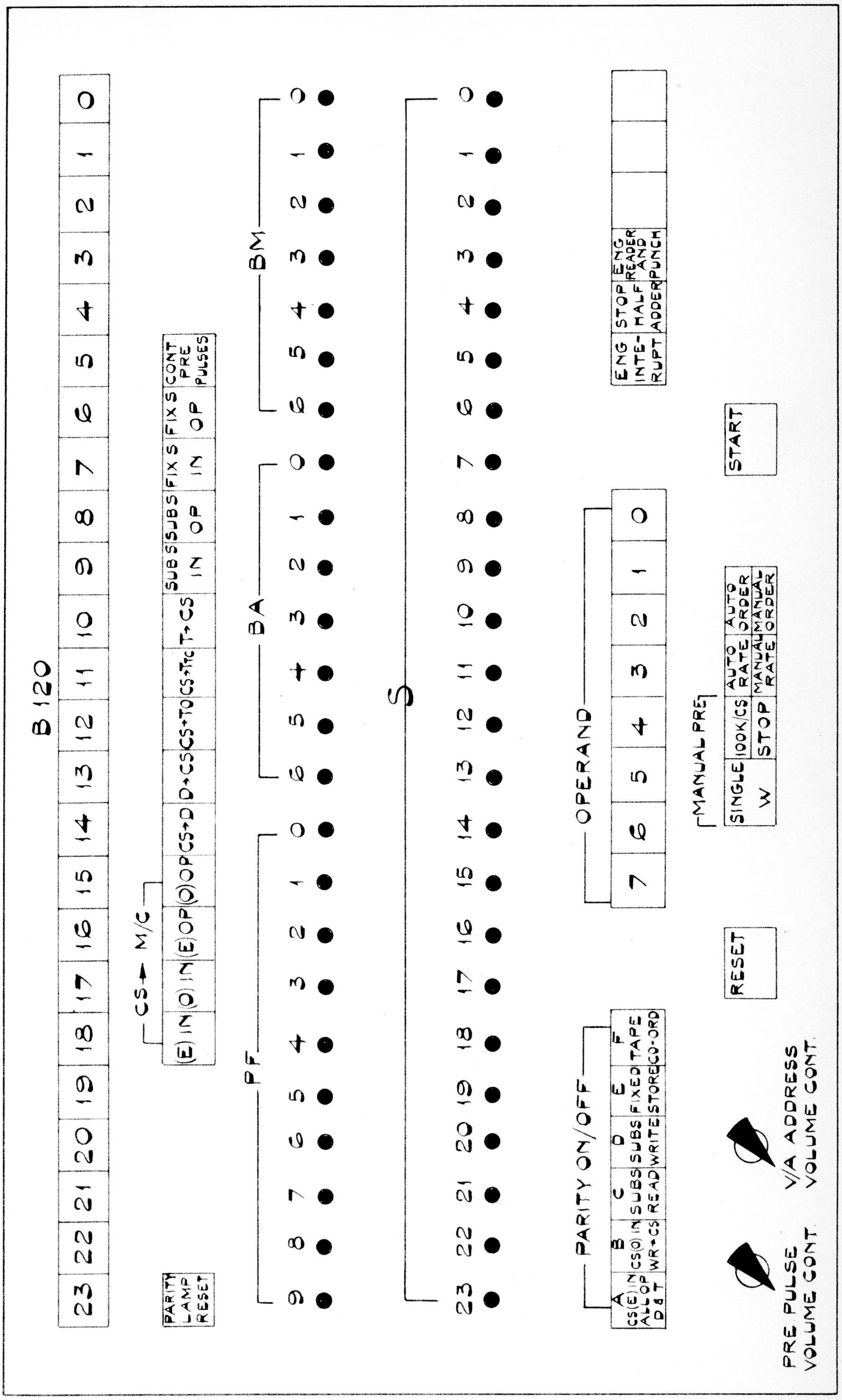
This controls the volume of the output from the hooter addressed as digit 27 of line 7 of the Central Computer V-store.

17) <u>Continuous Prepulses Indicator</u>

This indicator is lit under normal conditions when prepulses are being continuously supplied.

18) <u>Reader and Punch Indication</u>

This light is on when the Engineer's Reader and Punch digit in the Central Computer V-store is set to a one i.e. when the reader and punch are switched on.



ATLAS ENGINEERS CONSOLE

10.2 The Engineer's Paper Tape Reader

This is provided to give a simple means of input during the Engineers test programs. It is automatically switched to this mode when the Engineer's Interrupt button is pressed and it remains so until the "Engineer's Reader" digit in the Central Computer V-store is reset (line 3, digit 27). It may also be switched to this mode by writing a one to this digit. The equipment used is T.R.5 number 0. The Engage button on the reader need not be pressed before the equipment can be used.

The reason for always using this reader for engineering purposes is that there is an automatic hold up of instructions which read from or write to the peripheral V-store during Engineers Interrupt if the engineers reader is started and busy. This leads to a simpler method of operation for the computer because the normal interrupt programs for the reader are not entered.

During normal operation of the computer (i.e. when not on Engineers Interrupt) this reader is operated in the standard manner.

The Stop command is given after reading each character but providing another Start signal is given before the next location hole is detected the reader operates at maximum speed.

When on Engineers Mode the interrupt action is different from that on normal operation of the computer. If an interrupt occurs when on Main or Extracode control the fixed store address *40060000 is forced into B125 (instead of *40040000). The fixed store contains the following orders in this and the subsequent address.

```
*4006 124,  127,    0,      0  set pair flip-flop to not pair
      121,  125,    118,    0  jump to address specified in B118
```

This facility is used in various test programs.

Section 11 DETAILS OF THE ATLAS COMPUTER INSTALLATIONS

11.1 The Manchester University Atlas (Muse)

Muse is an Atlas I computer with

16,384	words of core store in two pairs of stacks.
8,192	words of fixed store.
1,024	words of subsidiary store.
4	M.D.5 drums (24,576 words each).
8	Ampex Tape Mechanisms (8 channels)
1	I.C.T. type 593 Card Reader (600 cards per minute).
1	I.C.T. type 582 Card Punch (100 cards per minute).
1	Anelex Line Printer series 4-1000 (maximum 1000 lines per minute
4	TR5 paper tape readers (300 characters per second).
4	Teletype paper tape punches (110 characters per second).
1	TR7 paper tape reader (1000 characters per second).
1	Creed 3000 paper tape punch (300 characters per second).
2	Creed 75 teleprinters (10 characters per second).
1	Venner Clock.
1	Graphical Output.
2	N.E.P. tape mechanisms (1000 characters per second) 1 X—ray Diffractometer
1	Engineer's console.

11.2 The London University Atlas

This is an Atlas I computer with

32,768	words of core store in four pairs of stacks.
8,192	words of fixed store.
1,024	words of subsidiary store.
4	M.D.5 drums (24,576 words each)
8	Ampex T.M.2 Magnetic Tape Mechanisms (8 channels with one 2 x 8 switching unit).
2	I.C.T. Type 593 Card Readers (600 cards per minute).
1	I.C.T. Type 582 Card Punch (100 cards per minute).
2	Anelex Line Printers series 4-1000 (maximum 1000 lines per minute).
4	T.R.5 paper tape readers (300 characters per second).
4	Teletype paper tape punches (110 characters per second).
3	Creed 75 teleprinters (10 characters per second).
1	Venner Clock.
1	Engineer's Console.

11.3 The N.I.R.N.S. Atlas

This is an Atlas I computer with

49,152	words of core store in 6 pairs of stacks.
8,192	words of fixed store.
1,024	words of subsidiary store.
16	Ampex Type T.M.2 Magnetic Tape mechanisms (8 channels with four 2 x 8 switching units).
4	M.D.5 drums (24,576 words each).
2	I.C.T. type 593 Card Readers (600 cards per minute).
1	I.C.T. type 582 Card Punch (100 cards per minute).
2	Anelex Line Printers series 4-1000 (maximum 1000 lines per minute).
2	I.B.M. type 729 Mark 4 Magnetic Tape Units (one channel).
3	T.R.5 paper tape readers (300 characters per second).
3	Teletype paper tape punches (110 characters per second).
1	Venner Clock.
1	Engineer's Console.

Section 12 Interrupts

12.1 The Interrupt Flip-Flops

The fixed store contains various routines which, together with certain built-in machine facilities, control the organisation of, and transfers of information between the core store and, the peripheral equipments, the magnetic drums and the magnetic tapes. These routines also determine the action to be taken if a parity failure is detected, or if a programme sets the exponent overflow register, divides by a non-standard or zero number in certain of the division orders, attempts to obey an unassigned function or refers to one of the private stores illegally.

A special control, Interrupt Control (B125), is provided and whenever action is required for one of the above reasons control is automatically switched from either Main or Extracode Control to Interrupt Control. Associated with each cause of interrupt there is a flip-flop in the V-store which is set whenever an interrupt is required. In the case of the peripheral equipments these are called the "look at me" flip-flops. These flip flops are grouped together, with not more than eight per group, and are connected via various OR gates to one of eight digits in line 2 of the Central Computer V-store. Using the B-log facility (B123, see Section 12.2) it is thus possible to identify the cause of Interrupt in at most five instructions.

An Interrupt may occur at an arbitrary time with respect to any interrupt from another source and consequently a system of priorities is needed in case two or more interrupts occur simultaneously. Highest priority is given to parity failures and second highest priority to the peripheral equipments. Each peripheral equipment has a "critical time" during which certain action must be taken immediately after its "look at me" flip-flop is set. For example for the Hammer Printer information must be sent to the appropriate digits to indicate if the relevant character has to be printed within 1.3m.s. of the interrupt signal or that character will be missed. The priorities for the peripherals are therefore based on these critical times i.e. an interrupt for an equipment with a short critical time is dealt with before a simultaneous interrupt for an equipment with a longer critical time.

The following table gives a list of all the possible interrupts arranged in order of priority. Where more than one equipment of a type is attached the one with the higher number has the higher priority. The table gives the addresses of the digits which are set for each interrupt in the Central Computer V-store (Vc), the Magnetic Tape V-store (Vm), the Drum V-store (Vd) or the Peripheral Equipment V-store, type 14 (Vp). For the peripheral equipments the critical time is also given where appropriate and for possible machine or programme faults the information to be printed out and the action to be taken are given.

Cause of Interrupt	V-store digits set	Critical Time or Information to be printed out.
Parity 1 (Core Store)	digit 31, line 0,Vc digit 31, line 2,Vc	p1 Also Page, Line, and

		Pattern Failing
Parity 2 (Drum Transfer)	digit 30, line 0,Vc digit 31, line 2,Vc	p2 r or w if r, Cabinet, Drum and Band if w, Page, Line and Pattern
Parity 3 (Magnetic Tape Transfer)	digit 29, line 0,Vc digit 31, line 2,Vc	p3 r or w if r, Channel if w, Page, Line and Pattern
Parity 4 (Subsidiary Store)	digit 28, line 0,Vc digit 31, line 2,Vc	p4 Line and Pattern
Parity 5 (Fixed Store)	digit 27, line 0.Vc digit 31, line 2,Vc	P5 Line and Pattern
Parity 6 (Magnetic Tape Buffer, Channels 7-0)	digit 26, line 0,Vc, (digit 34, lines 15-8,Vm) digit 31, line 2,Vc	p6 Channel
Non-equivalence when on Interrupt Control	digit 25, line 0,Vc digit 31, line 2,Vc	nei
I.B.M. Magnetic Tape End of Operation	digit 31, line 24,Vp digit 30, line 2,Vc	
Magnetic Tape Buffer Attention	digit 30, line 24,Vp digit 30, line 2,Vc	256μsecs
Card Readers 3-0 Column Ready,	digits 27-24, line 24,Vp digit 30, line 2,Vc	660μs
Magnetic Tape Block Address, Channels 7-0	digits 31-24, line 18,Vm digit 29, line 2,Vc	1 m.s.

Cause of Interrupt	V-store digits set	Critical Time or Information to be printed out.
Card Readers 3-0, End of Card	digits 27-24, line 23,Vp digit 31, line 31,Vp digit 28, line 2,Vc	2.5 m.s.
Xeronic Printers 1-0	digits 25-24, line 22,Vp digit 30, line 31,Vp digit 28, line 2,Vc	
T.R.7's 3-0	digits 27-24, line 20,Vp digit 28, line 31,Vp digit 28, line 2,Vc	1 m.s
T.R.7 Checks 3-0	digits 27-24, line 19,Vp digit 27, line 31,Vp digit 28, line 2,Vc	1 m.s
Graphical Outputs 1-0	digits 25-24, line 18,Vp digit 26, line 31,Vp digit 28, line 2,Vc	
Hammer Printers Character Interrupt	digits 25-24, line 17,Vp	1.3 m.s

	digit 25, line 31,Vp digit 28, line 2,Vc	
Hammer Printers 1-0 Line Count Interrupt	digits 25-24, line 16,Vp digit 24, line 31,Vp digit 28, line 2,Vc	5.7 m.s.
I.B.M. Magnetic Tape Mechanical Failure	digit 25, line 15,Vp digit 31, line 30,Vp digit 27 line 2,Vc	–
Fast Paper Tape Punches 3-0	digits 27-24, line 14,Vp digit 30, line 30,Vp digit 27, line 2,Vc	4 m.s.
T.R.5 Paper Tape Readers 11-8	digits 27-24, line 12,Vp digit 28, line 30,Vp digit 27, line 2,Vc	3.3 m.s.
T.R.5 Paper Tape Readers 7-0	digits 31-24, line 11,Vp digit 27, line 30,Vp digit 27, line 2,Vc	3,3 m.s.
Teletype Punches 11-8	digits 27-24, line 9,Vp digit 25, line 30,Vp digit 27, line 2,Vc	9.1 m.s.

Cause of Interrupt	V-store digits set	Critical Time or Information to be printed out.
Teletype Punches 7-0	digits 31-24, line 8,Vp digit 24, line 30,Vp digit 27, line 2,Vc	9.1 m.s.
Card Punches 1-0 Check Read	digits 25-24, line 7,Vp digit 31, line 29,Vp digit 26, line 2,Vc	10 m.s.
Card Punches 1-0, Punch Row	digits 25-24, line 6,Vp digit 30, line 29,Vp digit 26, line 2,Vc	38 m.s.
Card Punches 1-0, End of Card	digits 25-24, line 5,Vp digit 29, line 29,Vp digit 26, line 2,Vc	
Teleprinters 15-8	digits 31-24, line 2,Vp digit 26, line 29,Vp digit 26, line 2, Vc	100 m.s. for full speed
Teleprinters 7-0	digits 31-24, line 1,Vp digit 25, line 29,Vp digit 26, line 2,Vc	100 m.s. for full speed
Clock, one second	digit 26, line 0,Vp digit 24, line 29,Vp digit 26, line 2,Vc	100 m.s.
Clock, tenth of a second	digit 25, line 0,Vp digit 24, line 29,Vp digit 26, line 2,Vc	100 m.s.
Instruction Counter	digit 24, line 0,Vp digit 24, line 29,Vp digit 26, line 2,Vc	-
Non-equivalence or Lock-out	digit 25, line 2,Vc	-
Drum Transfer Fail	digit 28, line 60,Vd digit 31, line 1,Vc digit 24, line 2,Vc	-
Drum Cabinet Absent	digit 27, line 60,Vd digit 31, line 1,Vc digit 24, line 2,Vc	-

Cause of Interrupt	V-store digits set	Critical Time or Information to be printed out.
Drum Transfer Complete	digit 26, line 60,Vd digit 31, line 1,Vc digit 24, line 2,Vc	-
Magnetic Tape Deck Failure, Channels 7-0	digits 31-24, line 27,Vm digit 30, line 1,Vc digit 24, line 2,Vc	-
Exponent Overflow	digit 29, line 1,Vc digit 24, line 2,Vc	eo - Programme Trapped or Monitored
Non-equivalence on magnetic drum or tape transfers	digit 28, line 1,Vc digit 24, line 2,Vc	-
Sacred Violation by Operand	digit 27, line 1,Vc digit 24, line 2,Vc	svo - Programme Monitored
Sacred Violation by Instruction	digit 26, line 1,Vc digit 24, line 2,Vc	svi - Programme Monitored
Division by non-standard or zero number (374, 376, 377 basic orders only)	digit 25, line 1,Vc digit 24, line 2,Vc	do Programme Trapped or Monitored
Unassigned Functions	digit 24, line 1,Vc digit 24, line 2,Vc	ilinst Programme Monitored.

12.2 Action on an Interrupt

If one or more interrupt flip-flops are set and interrupts are not inhibited then before a further instruction is started

1) the octal number 40040000 is put into B125 (40060000 if digit 27, line 3 of the Central Computer V-store is a one).
2) the I/ME flip-flop is set to I.
   The effect is to switch to interrupt control and to obey the instruction in address 2048 of the fixed store.

Interrupts are inhibited when

(a) I/ME = I or
(b) the Inhibit Interrupts flip-flop is set (e.g. when a sequence of instructions in a program has to be obeyed without interruption).

An exception is if non-equivalence occurs when Interrupt control is being used. In this case control is immediately transferred to line 2048 of the fixed store and the routine entered to find the cause of the interrupt. This interrupt can only be caused by a machine fault and no record is kept of the control number at the time of the non-equivalence.

To assist in the Interrupt Routine one of the B-registers (B 123) performs in a special manner. This register is loaded in the normal way but its output is 8 times (i.e. a shift of 3 binary places) the characteristic of the logarithm to base 2 of the 8 least significant digits of the input i.e. the output is the position of the most significant one.

	INPUT									OUTPUT					
digits	23-8	7	6	5	4	3	2	1	0	23-7	6	5	4	3	2-0
	δ	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	δ	0	0	0	0	0	0	1	δ	0	0	0	0	1	0
	δ	0	0	0	0	0	1	δ	δ	0	0	0	1	0	0
	δ	0	0	0	0	1	δ	δ	δ	0	0	0	1	1	0
	δ	0	0	0	1	δ	δ	δ	δ	0	0	1	0	0	0
	δ	0	0	1	δ	δ	δ	δ	δ	0	0	1	0	1	0
	δ	0	1	δ	δ	δ	δ	δ	δ	0	0	1	1	0	0
	δ	1	δ	δ	δ	δ	δ	δ	δ	0	0	1	1	1	0
	δ	0	0	0	0	0	0	0	0	0	1	0	0	0	0

To identify the cause of an interrupt the method in the interrupt routine is as follows

1) transfer line 2 of the central computer V-store to B123
2) transfer control to one of eight programs (via a jump table) modified by 8123.

This identifies the type of interrupt and also ensures that for simultaneous interrupts the one with the highest priority is dealt with first.

If the interrupt is a parity interrupt, line 0 of the central computer V-store is transferred to B123 and control transferred (again via a jump table and modified by B123) to an appropriate program.

If the interrupt is due to one of the peripheral "look at me" indicators being set, the relevant mechanism can be identified by one or two further "references" to B123. For a card reader or an I.B.M. magnetic tape interrupt the cause of interrupt can be identified by one further "reference" to B123. For a magnetic tape block address interrupt the appropriate channel can be identified by a further use of B123 and the mechanism currently on that channel is known to the magnetic tape supervisor program. For all other peripheral equipments it is necessary to copy the appropriate line (31, 30 or 29) of "Type 14" of the peripheral equipment V-store to B123 and transfer control (modified by B123) to the corresponding program. This identifies the type of equipment and one further use of B123 identifies the equipment itself that caused the interrupt.

For a "non-equivalence or lock-out" interrupt the fixed store program must determine whether the interrupt is due to a non-equivalence of either the operand or instruction address or whether the interrupt is caused by the required block being locked out. To do this it first examines the "Operand non-equivalence or lock-out" digit in the Core Store V-store (line 34, digit 35). If this is set it is possible to find out from the page directory in the subsidiary store whether the required block is in the core store and if it is whether it is locked-out. If the Operand non-equivalence digit is not set the address of the required instruction can be determined from examination of the M/E digit and the contents of B126 or B127, as appropriate. The reason for the interrupt is then determined as before from the page directory.

If the fixed store routine finds that the interrupt is due to a block being locked-out it either enters the Engineers Tests or switches programmes. If the interrupt is due to the required block not being in the core store a drum transfer program is entered.

For the other interrupts line 1 of the Central Computer V-store is transferred to B123 and the appropriate programme is entered (e.g. for exponent overflow). The "Non-equivalence, tapes or drums" digit is set during a magnetic drum or magnetic tape transfer if non-equivalence is obtained either due to a page address register not being set correctly or a fault occurring. The appropriate transfer can be terminated by writing to the "Stop Command" digit for a drum transfer or the "End Transfer" digit for the appropriate channel for a tape transfer. The non-equivalence for this digit is only over the block digits (i.e. 22-12) of the address and does not include the lock-out digit. It is set independently of the Non-equivalence or Lock-out digit in line 2 of the Central Computer V-store.

The interrupts are dealt with in order of priority and after an interruption has been dealt with the flip-flop which led to the interrupt is reset (e.g. a card reader Look At Me). After the appropriate action for an interrupt has been taken, control is transferred to line 2048 of the fixed store and a further examination is made of the interrupt digits of line 2 of the Central Computer V-store. If one of these digits is set the appropriate routine is entered. If none of these digits is set the control number in either B126 or B127 is adjusted if necessary and control switched back to Main or Extracode i.e. the I/ME digit is reset to ME.

For most interrupts no adjustment of the number in the control being obeyed before the interrupt occurred is necessary on leaving the interrupt program. However for an interrupt due to non-equivalence or lock-out of the operand (digit 35, line 36 of the Core Store V-store set) control is advanced by either one or two before the interrupt program is entered and must be adjusted accordingly before a switch back to this control takes place. The adjustment required is given by digit 26 of line 3 of the Central Computer V-store (the +1/+2 digit).

This digit is normally reset to zero (+2) but it is set to one either (a) if an unscrambled code (Z-code) causes a non-equivalence or lock-out interrupt for its operand or (b) if another interrupt occurs whilst an instruction is being obeyed but before the non-equivalence or lock-out is detected.

The "Non-equivalence or Lock-out on Extracode Exit" digit (line 3, digit 28 of the Central Computer V-store) is set to one whenever Main Control is advanced by one and it is reset to zero on switching to Extracode control. If a Non-equivalence or Lock-out interrupt occurs when an Extracode exit instruction is obeyed both the Main and Extracode control numbers are advanced by one. Consequently the Non-equivalence interrupt routine examines digits 26 of line 3 and 24 of line 4 of the Central Computer V-store and if these are both zero (i.e. +1/+2 set as +2 and M/E set as E) it also examines digit 28 of line 3. If this latter digit is set then both Main and Extracode control numbers must be reduced by one before returning to the extracode exit instruction.

One use of the Inhibit Interrupt flip-flop is when control is being switched from E to I. The program to do this is

```
121,    100,     0,      0.1     )
                                 )  Inhibit Interrupts
113,    100,     0,      3 * 6   )

121,    125,     0,        (1)      Set contents of
                                    B125

121,    100,     0,      0.2     )
                                 )  Switch control to I
113,    100,     0,      3 * 6   )
```

If the Inhibit Interrupts flip-flop were not set an interrupt might occur immediately after B125 had been set. After carrying out this latter routine B125 would be left set from this routine and not as required by the program specifying the change of control.

Section 13 Layout of the V-store

13.1 Addresses of the V-store

A V-store address is identified by having digits 23-21 as 110 respectively. Digits 20-15 and digit 11 are spare and digits 14-12 identify the part of the V-store being referred to. Digits 10-3 give the address of the appropriate word within the relevant part of the V-store. As the full allocation of words is not needed in all the parts of the V-store the most significant of these digits are not used in some cases. Only the more significant half-words are used in the V-store and hence address digits 2, 1 and 0 are always zero.

The allocation of digits 14 - 3 in a V-store address is given in the following table.

digit	14	13	12	11	10	9	8	7	6	5	4	3	
	0	0	0	0	0	0	0	0	0	δ	δ	δ	Central Computer
	0	0	1	0	0	0	δ	δ	δ	δ	δ	δ	Core Store
	0	1	0	0	0	0	δ	δ	δ	δ	δ	δ	Drum
	0	1	1	0	0	0	0	δ	δ	δ	δ	δ	Magnetic Tape
	1	0	0	0	δ	δ	δ	δ	δ	δ	δ	δ	Peripheral Equipments
	part of V-store			Spare	Word within the appropriate section								

The notation used in the following sections for reading from or writing to the digits in the V-store is

R where it is possible to <u>Read</u> only; writing has no effect and digits not so marked are always read as zero

W1 where it is possible to <u>Write Ones</u> only; reading gives zeros and writing zeros has no effect

W10 where it is possible to <u>Write Ones and Zeros</u>; reading always gives zeros

RW0 where it is possible to <u>Read</u> and where a digit is reset by <u>Writing Zero</u> to it

RW1 where it is possible to <u>Read</u> and also to <u>Write Ones</u>; writing zeros has no effect. A digit is reset to zero by writing a one to it.

W10 where it is possible to <u>Read</u> and also to <u>Write Ones and Zeros</u>

13.2 Central Computer V-store

The first three lines are concerned with interrupts and are used to identify the source of an interruption.

line	digit			
0	31	RWO	Parity 1	Core store addressed by the machine
	30	RWO	Parity 2	Drum transfer. This digit is set if incorrect parity is detected (a) in the contents of the core store page involved in a write transfer (b) in the information on a drum sector for a read transfer.
	29	RWO	Parity 3	Magnetic Tape Transfer. This digit is set if incorrect parity is detected (a) in the contents of the core store page involved in a write transfer (b) in the information read from the tape co-ordinator Buffer Store to the Main Core Store page.
	28	RWO	Parity 4	Subsidiary store
	27	RWO	Parity 5	Fixed store
	26	RWO	Parity 6	Magnetic Tape Buffer
	25	RWO	Non-equivalence in Interrupt control. This digit is set if non-equivalence occurs between digits 22-12 of the required address and the corresponding digits of all the Page Address Registers due to either a machine or fixed store programme fault when on Interrupt Control. The setting of this digit also sets address 2048 of the fixed store in interrupt control and the routine for finding the cause of an interrupt is entered immediately. No record is kept of the previous interrupt control number.	
1	31	R	Drums. The “OR” of digits 29, 28, 27 and 26 in line 60 and digit 27 in line 56 of the drum V-store	
	30	R	Magnetic tape deck failure. The “OR” of line 19 of the Magnetic Tape V-store.	
	29	RWO	EO Accumulator Exponent overflow	
	28	RWO	Non-equivalence tapes or drums.	
	27	RWO	SVO Sacred violation by the operand. An illegal reference by the operand to one of the private stores.	
	26	RWO	SVI Sacred violation by an instruction. An illegal attempt to obey an instruction in the private stores.	
	25	RWO	DO Division by non-standard or zero number (set by basic functions 374, 376, 377 only)	
	24	RWO	Unassigned functions	
2	31	R	Parities The “OR” of line 0	
	30	R	P0	Peripherals 0 (the card readers, I.B.M. tape)
	29	R	T	Magnetic tape block markers. The “OR” of line 18 of the magnetic tape V-store

	28	R	P1 Peripherals 1 (Graphical outputs, TR7 and N.E.P tape)
	27	R	P2 Peripherals 2 (the Teletype punches, TR5's and Creed 3000 paper tape punches)
	26	R	P3 Peripherals 3 (the instruction counter, clock, tele-printers, Anelex printer, card punches and X–ray Diffractometer)
	25	RWO	Non–equivalence or Lock–out. The non–equivalence of all page address registers or a programme reference to a locked–out page of the main core store
	24	R	the "OR" of line 1
3	28	R	Non-equivalence on Extracode exit (see Section 12.2)
	27	W10	Engineers tape reader. Selected by writing a one
	26	R	+1/+2 (1/0). On a non-equivalence or lock-out interrupt due to an operand not being available this digit indicates whether one or two has been added to control
	25	RW10	I/ME Control flip-flop (1 if I)
	24	RW10	II Inhibit Interrupts (set to 1 to inhibit interrupts)
4	24	RW10	M/E control flip-flop (1 if M)
6	29	RW10	12/13 shift. "1" if 12 shift. Reset by every division order.
	28	RW10	Qs. Sign of quotient in basic division order
	27	RW10	AO (Fixed point) accumulator overflow
	26	RW10	Bt < 0)
	25	RW10	Bt ≠ 0) B test register
	24	RW10	Bc B-carry
7	31-24	R	8 handswitches
	27	W10	Hooter

13.3 Core store V-Store

13.3.1 32-Page Coordinator

The layout of the Core Store Coordinator for the Manchester University Atlas is given below. The layout of the V-store for all other machines is given in the next section.

Lines		Digit	
0-31	W10	47	Page lock out) Page address registers
	W10	46-36	Block address)
32	R	47-32	"Use" digits of pages 15 to 0 respectively.
33	R	47-32	"Use" digits of pages 31 to 16 respectively.
34	R	47	Instruction non-equivalence or lock out
	R	46-36	address on operand non-equivalence or lock out
35	W10	47	Ignore all lock-out digits
	R	46-37	Line address on operand non-equivalence or lock out (digits 11-2 respectively of the specified address).

The digits in lines 32 and 33, also digits 46-36 in line 34 and digits 46-37 in line 35 are reset to zero automatically when the information is read (by program) from them.

Digit 47 of line 34 is automatically reset to one when line 34 is read.

Digit 47 of line 35 is unaltered when line 35 is read. It is always read as zero but the lock out digits are ignored (i.e. no lock out interrupts occur) if the last digit written to it was a one. It is reset by writing zero to it.

The "non-equivalence or lock out" block and line addresses are set when an interrupt occurs due to the required operand not being immediately available in the core store i.e. the required information block may be either not in the core store or it may be locked out. These digits are not set on an interrupt due to an instruction not being available.

The Instruction non-equivalence or lock out digit is set to zero (i.e. Operand) when a non-equivalence interrupt occurs due to the required operand not being available.

13.3.2. General Core Store V-Store

The layout of the Core Store coordinator for machines of up to 128 pages (other than the Manchester University Atlas) is given below.

	Lines		Digits	
(a)	0-31	)		
(b)	32-63	) W10	47 Page lock out	) Page Address Registers
(c)	64-95	)	46-36 Block address	)
(d)	96-128	)		
(a)	(128	)		) Pages 15-0
	(129	)		) 31-16
		)		
(b)	(130	)		) 47-32
	(131	)		) 63-48
		) R	47-32 Use Digits	
(c)	(132	)		) 79-64
	(133	)		) 95-80
		)		
(d)	(134	)		) 111-96
	(135	)		) 127-112
	(136	)	47 Instruction not	)
(a)	(	) R	equivalence	) Set on not equivalence
	(	)	46-36 Block Address	)
	(137	)	46-37 Line Address	) or lock out of operand
		W10	47 Ignore all lock out digits	

For 96 or less pages, lines (d) vacant
For 64 or less pages, lines (c) vacant also.
For only 32 pages, lines (b) vacant also.

The digits in lines 128 - 135, digits 46 - 36 in line 136 and digits 46 - 37 in line 137 are automatically reset to zero when the information is read by program from them.

Digit 47 of line 136 is reset to one when line 136 is read.

13.4 Drum V-store

(maximum of 8 Cabinets, each cabinet containing either one file drum or up to four fast drums, type MD5)

Lines		Digits	
0-31	R	29-27	Θ-registers
	R (for drums 0-31 respectively	26	"1.5 - 2" digit
32	R	29-27	Blocks complete (cleared to zero when write to Start digit, digit 29 line 56)
36	RW10	29-27	Number of blocks requested.
40	RW10	29-27	Starting Θ
44	RW10	28-26	Band on the appropriate drum
48	RW10	27-26	Drum (within a cabinet)
52	RW10	28-26	Cabinet
56	RW1	29	Read: Started (indicates a drum transfer is in progress) Write: Start(initiates a drum transfer
	W1	28	Selection change
	RW1	27	Read: Drum Transfer Complete Look At Me Write: Read
	RW1	26	Write
60	RW1	29	Read: Drum Count Failure Look At Me (In the gap between sectors the count of words is tested; it should be a multiple of 512) Write: Stop
	R	28	Drum Cabinet Absent Look At Me (Drum cabinet non existent or unplugged)
	RW1	27	Read: Drum Band Isolated Look At Me. (Trying to write to a band whose isolation switch is on.) Write: Reset DTC Look At Me
	RW1	26	Read: Drum Request Ignored Look At Me (the core store has not accepted address from the drum in time) Write: Reset Drum Failure LAM's (i.e. DCF, DCA, DBI and DRI)

The "OR" of line 60 and line 56 digit 27 provides the Drum Interrupt, D (digit 31 of line 1 of the Central Computer V-store).
DCA and DBI are set when appropriate after a one is written to the Start digit (digit 29 line 56) but a drum transfer is not initiated. When DCF and DRI occur, the drum transfer is automatically stopped. DTC cannot occur at the same time as a drum failure.

13.5 The Magnetic Tape V-Store

Line	Digits		
0-6	36-24	R	Present Block Address Register for Channels 0-6
7	36-24	RW10	Present Block Address Register for Channels 7
8-15	41-24		Tape Command Registers
	41	RW1	Fast Speed
	40	W1	Normal Speed
	39	RW1	Reverse
	38	W1	Forward
	37	RW1	Start
	36	W1	Stop
	35	W1	Disengage Deck
	34	R	Buffer Parity Fault (on writing to tape).
	33	RW1	Recover Read
	32	W1	Normal Read
	31	RW1	Write: Read at next Block Address Read: Read Transfer
	30	W1	End Read at next Block Address
	29	RW1	Write
	28	W1	End Transfer
	27	R	Write Permit
	26	R	End of Tape
	25	R	Not 512 Word Transfer
	24	R	Check sum fail
16	39-24	R	Disengaged Flip Flops for Decks 15-0
17	39-24	R	Disengaged Flip-Flops for Decks 31-16
18	31-24	RW1	Block Address Interrupt for Channels 7-0 Writing '1' resets the digit to '0"
19	31-24	RW1	Deck Failure Interrupt for Channels 7-0 Writing '1' resets the digit to '0'
20	36-24	W10	Select Deck and Channel
21	35-24		Tape Address Command Register for Channel 7 only
	35	R	Deck Modified
	34	RW1	Leading address indication
	33	RW1	Permit count
	32	W1	Do not permit count
	31	RW1	Write Reference Marker
	30	W1	Do not write Reference Marker
	29	RW1	Write '1's

28	W1	Do not write '1's
27	RW1	Address tape
26	W1	Do not address tape
25	RW1	Address Fault
24	W1	Reset Address Fault

The "OR" of line 18 is Digit 2~~89~~ of line 2 of the Central Computer V-store. The "OR" of line 19 is Digit 30 of line 1 of the Central Computer V-store. The tape V-store includes registers in the mechanisms. Some of these are flip-flops of the peripheral type, requiring long trigger pulses. The generating of these pulses causes the Write Access time for the tape V-store to be 5μsec.

13.6 Peripheral. Equipment V-Store

13.6.1 V-store Addresses

Each equipment has associated with it one or more lines of the V-store. Equipments of the same type are in consecutive lines (or groups of consecutive lines) and only the most significant half-words are used.

The peripheral equipment V-store addresses are:-

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	0	0	0	0	0	0	0	1	0	0	0	δ	δ	δ	δ	ϵ	ϵ	ϵ	ϵ	0	0	0
V-store		Spare							Peripheral			Spare	Type				Number						

The "types" and the corresponding interrupt digits in line 2 of the Central Computer V-store are:

Type 0	Card Readers 0-3 (P1)
Type 1	Rank Xeronic Printers 0-1 (P2)
Type 2	TR7 Paper Tape Readers 0-3 (P2)
Type 3	Graphical Outputs 0-1 (P2)
Type 4	I.C.T. Hammer Printers. 0-1 (P2)
Type 5	I.B.M. Magnetic Tape (P1)
Type 6	Fast Paper Tape Punches 0-3 (P3)
Type 7	TR5 Paper Tape Readers 0-11 (P3)
Type 8	Teletype Punches 0-11 (P3)
Type 9	Card Punches 0-1 (P4)
Type 10	
Type 11	Teleprinters 0-15 (P4)
Type 12	Clocks and Instruction Counter (P4)
Type 13	
Type 14	"Look at me's" (also includes some lines allocated as type 15)

13.6.2 V-store Digits

Type 0. Card readers (maximum 4) (*6004)

lines	Digits		
0-3	47-36	R	Information (Rows +,-, 0-9 respectively)
(card readers number 0-3	30	RW1	Read: Overdue Write: Reset Overdue
respectively)	29	W1	Do not Divert
	28	RW1	Read: Card Levels Write: Put out End of Card Look At Me
	27	W1	Start
	26	RW1	Read: Disabled Write: Put out Column Ready "Look At Me"
	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage
8-11	47-36	R	Check information (rows +,-,0-9 respectively)

Type 1.

Type 2a. TR7 Paper Tape readers (maximum 4) (*600404)

lines	Digits		
0-3	36	R	Disabled
(tape readers	35	R	Tape Broken
number 0-3	34	R	Overdue
respectively)	33	R	5/7 switch ("1" if five channel;"0" if seven channel)
	32-26	R	Information. The location hole is between digits 28 and 29. Digits 31 and 32 are zero for five-channel tape.
	30	W1	Stop Rewind
	29	W1	Rewind
	28	W1	Reset Overdue
	27	W1	Start
	26	W1	Put out "Look At Me"
	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage

Type 2b. N.E.P. Tape (*6004041 on Manchester University Atlas)

line	Digits		
1	36	R	Mode of Operation. 1 if Writing; 0 if Reading
	35-29	W1	Write: Information
	35	R	Read: Disabled
	34	R	Read: Overdue
	33	R	5/7 Channel, 1 if 5 Channel. 0 if 7 Channel.
	32-26	R	Read: Information
	28	W1	Write: Reset Overdue
	27	W1	Write: Start
	26	W1	Write: Put out Look At Me
	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage

Type 3. Graphical Outputs (maximum 2) (*600406)

lines	Digits		
0-1 Graphical	27	RW1	Read: Camera Fault Write: Start
Outputs number 0-1	26	RW1	Read: Brilliance Write: Put out Look At Me
respectively)	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage
8-9	47-38	W1	x - co-ordinate
	35-26	W1	y - co-ordinate
	25	W1	Display
	24	W1	Dim

Type 4. Anelex Printer (maximum 4) (*60041)

lines	Digits		
0-3	36	W1	Clear Core Store
(Anelex	35	W1	End of Line
Printers 0-3	34	W1	Information Strobe
respectively)	33-28	W1	Information
	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Paper Warning Write: Put out Look At Me's
	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage

Type 5. I.B.M. Magnetic tape (*600412)

line	Digits		
0	24-27	RW1	Character Count
	28	RW1	Read: TU1 Disengaged Write: Stop Write
	29	RW1	Read: TU2 Disengaged Write: Select BCD Mode
	30	RW1	Read: TU3 Disengaged Write: Select Low Density
	31-33	RW1	Read: Rewind in Progress TU1-3 Write: Select TU1-3
	34	R	Low density selected
	35	RW1	Read: BCD selected Write: Recover read
1	24	RW1	Read: Buffer Attention Overdue Write: Read Order
	25	RW1	Read: Lateral Parity Write: Write Order
	26	RW1	Read: Longitudinal Parity Write: Backspace
	27	RW1	Read: Tape Indicator Set Write: Rewind
	28	RW1	Read: Load Point Indicator Write: Rewind and Disengage
	29	RW1	Read: Mechanical Failure Write: Disengage
	30	W1	Put out End of Operation Look at me
	31	W1	Put out Buffer Attention Look at me
	32	W1	Inhibit Buffer Attention Look at me
	33	W1	Allow Buffer Attention Look at me
5	47-24	RW1	Information
4	47-24	RW1	Information
3	47-24	RW1	Information
2	47-24	RW1	Information

Type 6. Creed 3000 Paper Tape Punch (maximum 4) (*600414)

lines	Digits		
0-3	35-29	W1	Information
(Tape Punches 0-3 respectively)	28	RW1	Read: Check Fail Write: Reset Check Fail
	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Tape Warning Write: Put Out Look At Me

	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage

Type 7. TR5 Paper Tape Readers (maximum 8) (*600416)

lines	digits		
0-7 (Tape Readers	34	R	Overdue
Number 0-7 respectively)	33	R	5/7 switch ("1" if five-channel; "0" if seven-channel)
	32-26	R	Information. The location hole is between digits 28 and 29. Digits 31 and 32 are zero for five-channel tape
	28	W1	Reset Overdue
	27	W1	Start
	26	W1	Put out Look At Me
	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage

Type 8. Teletype Punches (maximum 8) (*60042)

lines	Digits		
0-7	35-29	W1	Information. The location hole is between digits 31 and 32
Teletype punches	(33-29 for five channel punches)		
number 0-7 respectively)	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Tape Warning Write: Put out Look At Me
	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage

Type 9a. I.C.T Type 582 Card Punches (maximum 2) (*600422)

lines	Digits		
0-1	47-40	RW10	Information. One digit of each of columns 8-1 respectively where column 1 is that first in the card punch.
(Card Punches	39	RW1	Read: Card Levels Write: Put Out Brush Look At Me
0-1 respectively)	38	RW1	Read: Card Wreck Write: Reset Overdue
	37	R	Overdue

	29	W1	Do Not Offset
	28	W1	Put Out End of Card Look At Me
	27	W1	Start
	26	W1	Put Out Punch Look At Me
	25	RW1	Read: Stopped Write: Stop
	24	RW1	Read: Disengaged Write: Disengage
4-5	47-24	RW10	Information (see line 0). One digit of each of columns32-9 respectively
8-9	47-24	RW10	Information (see line 0). One digit of each of columns 56-33 respectively
12-13	47-24	RW10	Information (see line 0). One digit of each of columns 80-57 respectively

Type 9b. The X-ray Diffractometer

Line 5 (*60042250)

Digit	Write	Digit	Read
24-43	Information	24-43	Information
44	Load Information (LIv)	44	Reflected Count Overflow
45	Start Count (SCv)	45	Check read
46	Set the Position (SPv)		

Line 9 (*60042310)

Digit	Write		Digit	Read
24	Disengage		24	Disengaged
25	2Θ (Two-theta)	(2Θv)	25	2Θ
26	W (omega)	(Wv)	26	W
27	ϕ (PHI)	(ϕv)	27	Φ
28	X (CHI)	(Xv)	28	X
29	Z-line	(Zv)	29	Clamp out of range
30	V-reset		30	Limit switch failure
31	Clamp registers(CRv)		31	References reached
32	Read selected register(RIv)		32	Test A.
33	Put out position set L.A.M.		33	Test B
34	Put out count complete L.A.M.		34	Test C

35	Put out contingencies L.A.M.	35	Test D
36	Open shutter	36	Test E
37	Close shutter		
38	Drive W - circle (CWDRv)		
39	Drive 2Θ- circle (2ΘDRv)		
40	Drive to references (DRv)		

Line 139(*60042350)
Digit

24	Test and isolate motors and nucleonics (TIMN)
25	Reset test and isolate motors and nucleonics (RESET -TIMN)
26	Test-count start * (TCS)
27	Test-count-stop * (TCst)
28	Test set-add (TSA)
29	Test set-subtract * (TSS)
30	Test add ± 1
31	Test fringes (TFv)

<u>The Z-line</u>

Digit	Write	Digit	Read
24	Set window 1	24	Window 1 in
25	Reset window 1	25	Window 2 in
26	Set window 2	26	Window 3 in
27	Reset window 2	27	Window 4 in
28	Set window 3	28	Balanced filter 1 in
29	Reset window 3	29	Balanced filter 2 in
30	Set window 4	30	Attenuator 1 in
31	Reset window 4	31	Attenuator 2 in
32	Set balanced filter 1	32	Counting clock
33	Reset balanced filter 1	33	Motor speed 1
34	Set balanced filter 2		
35	Reset balanced filter 2		
36	Set attenuator 1		
37	Reset attenuator 1		
38	Set attenuator 2		
39	Reset attenuator 2		
40	Count clock		
41	Count X-Rays		
42	Set motor speed 1		
43	Set motor speed 2		

Type 10.

Type 11. Creed 75 Teleprinters (maximum 4) (Printer Only) (*600426)

lines	digits		
0 – 3 (Teleprinters number 0-3 respectively)	35,34, 32-29	W1	Information
	27	RW1	Read: Disabled Write: Start
	26	RW1	Read: Paper Warning Write: Put Out Look At Me
	25	RW1	Stopped/Stop
	24	RW1	Disengaged/Disengage

Type 12. Instruction Counter and clock

(a) Instruction counter (*60043)

line	digits		
0	46-36	RW10	Information. One is added to digit position 36 every time an instruction is obeyed except when (a) using interrupt control or (b) the counter is stopped.
	27	W1	Start
	26	W1	Put Out Look At Me
	25	RW1	Stopped/Stop
	24	R1W1	Write Strobe (always read as a one)

(b) Clock (*6004301)

1	45-44	R	hours, tens
	43-40	R	hours, units
	38-36	R	minutes, tens
	35-32	R	minutes, units
	30-28	R	seconds, tens
	27-24	R	Seconds, units
	27	W1	Start
	26	W1	Put out 1 sec Look At Me
	25	W1	Stop
	24	W1	Put out 1/10 sec Look At Me

Type 13

Type 14. "Look at me's" (*600434)

Line	digits		
0	24	R	Instruction counter
	25	R	Clock, every tenth, of a second
	26	R	Clock, every second
1	24-27	R	Teleprinters 0 – 3

3	24-27	R	Anelex Printers 0 – 3, Overflow
	28-31	R	Anelex Printers 0–3 Print
6	24	R	Card Punch O. End of Card
	25	R	Card Punch 1, End of Card or X–ray Diffractometer, Position set
	26	R	Card Punch O. Punch
	27	R	Card Punch 1, Punch or X–ray Diffractometer, Count Complete
	28	R	Card Punch O. Brush
	29	R	Card Punch 1, Brush or X–ray Diffractometer, Contingencies
8	24-31	R	Teletype punches 0 - 7
11	24-31	R	TR5 paper tape readers 0 – 7
14	24-31	R	Creed 3000 paper tape punches 0 - 3

Type 15. (*600436)

2	24-25	R	Graphical outputs 0 - 1
4	24-27	R	TR7 paper tape readers
	25	R	N.E.P tape (Manchester University Atlas)
7	24-27	R	Card Readers 0 – 3, End of Card
8	24-27	R	Card Readers 0 – 3, Column Read
	30	R	I.B.M. Magnetic tape, Buffer Attention
	31	R	I.B.M. Magnetic tape, End of Operation
	The “or” of line 8 is the digit P0 of the central computer V-store		
13	24	R	The “or” of line 0, type 14
	25	R	The “or” of line 1, type 14
	27	R	The “or” of line 3, type 14
	30	R	The “or” of line 6, type 14
	The “or” of line 13 is the digit P3 of the central computer V-store		

14	24	R	The “or” of line 8, type 14
	27	R	The “or” of line 11, type 14
	30	R	The “or” of line 14, type 14
	The “or” of line 14 is the digit P2 of the central computer V-store		
15	26	R	The “or” of line 2, type 15
	28	R	The “or” of line 4, type 15
	31	R	The “or” of line 7, type 15
	The “or” of line 15 is the digit P1 of the central computer V-store		