7. FLOATING POINT AND DOUBLE LENGTH ARITHMETIC

7.1 Floating Point Arithmetic

The floating point arithmetic facility allows numbers with decimal fractions to be input, processed and output by the computer.

Floating point numbers are represented by two parts, the mantissa and the exponent, e.g. the numbers 107.42 and 0.086 can be represented by the pairs:

<table>
<thead>
<tr>
<th>Mantissa</th>
<th>exponent</th>
<th>Mantissa</th>
<th>exponent</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10742</td>
<td>3</td>
<td>0.86</td>
<td>-1</td>
</tr>
</tbody>
</table>

i.e. $0.10742 \times 10^3 = 107.42$ and $0.86 \times 10^{-1} = 0.086$

7.1.1 Floating Point Numbers in LEO III

Floating point arithmetic in LEO III is a binary arithmetic. Each mantissa may occupy up to 37 bits and each exponent up to 27 bits, giving a range of numbers ($x$) that can be represented:

$$-2^{19} \leq x \leq 2^{19}$$

where the mantissa ($m$) lies within one of the ranges:

$$(0.5 \leq m < 1) \text{ or } -1 \leq m < -0.5) \text{ or } -1 \leq m \leq 1$$

dependent on the mode of arithmetic used. The exponent ($e$) is artificially restricted within the range:

$$-2^{19} \leq e \leq 2^{19}$$

Each floating point number is held in the store in three consecutive short words as shown below:

<table>
<thead>
<tr>
<th>mantissa</th>
<th>(1)</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>exp. (2)</td>
<td>exp. (1)</td>
<td>2</td>
</tr>
<tr>
<td>mantissa</td>
<td>(2)</td>
<td>4</td>
</tr>
<tr>
<td>mantissa</td>
<td>(3)</td>
<td>6</td>
</tr>
<tr>
<td>exp. (4)</td>
<td>exp. (5)</td>
<td>etc.</td>
</tr>
</tbody>
</table>
The mantissa is held in sign and complement form and the exponent in sign and modulus form.

7.1.2 The Floating Point Accumulator

The floating point arithmetic accumulator \( A^0 \) consists of two long compartments of store, namely '28' and '30', which contain the accumulator mantissa and the accumulator exponent in sign and complement form.

Bit 37 of '28' represents the unity bit of the mantissa. This enables two numbers with mantissa unity to be added without causing overflow. The fractional part of the mantissa is located in bits 36 to 1 of compartment '28', with the least significant bit in bit 1 of the word.

The accumulator exponent may lie within the range \(-2^{40} < e < 2^{40}\), which is large enough to justify the omission of any check that the result of an arithmetic operation is within the range of numbers that can be held in \( A^0 \).

7.1.3 Floating Point Binary Normalised Arithmetic

All operations in this mode of arithmetic have built in round-off and the limits on the mantissa size, \(0.5 < m < 1\), and \(-1 < m < 0.5\) are applied to all rounded results except zero.

When a rounded mantissa is outside the prescribed limits, e.g. greater than 1 or equal to \(-\frac{1}{2}\) say, the mantissa is left or right shifted as appropriate and the exponent amended accordingly. An example of this occurs frequently in addition where the sum of two positive mantissae with the same exponent is certain to have a mantissa outside the limits, e.g.

<table>
<thead>
<tr>
<th>Mantissa</th>
<th>Exponent</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.75</td>
<td>3</td>
</tr>
<tr>
<td>+ 0.75</td>
<td>3</td>
</tr>
<tr>
<td>Right shift</td>
<td>1.50</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
</tr>
</tbody>
</table>

In the computer the same calculation would be carried out as follows:

Amendment No. 8
December 1964
The advantage of floating point binary normalised arithmetic lies in the accuracy it retains over long series of calculations, with no attention needed from the programme. The built in round off reduces the error in the mantissa to ± 0.5 in the last bit position.

7.1.4 Floating Point Binary Significant Digit Arithmetic

All operation in this mode of arithmetic have built in round-off and the limits on mantissa size, −16 ≤ m ≤ 1, ensure that the result retains as many digits as the operands' significance justifies. Addition and subtraction retain the un-normalised mantissa, except when sums or differences exceed the limits, −16 m ≤ 1 when a corrective right shift is applied before round off. Multiplication and division retain only the number of bits in the modulus of the rounded product or quotient that there is in the modulus of the lesser mantissa, except where this clashes with the limit requirements, e.g. -1 x -1 gives a rounded product or mantissa + 1 and an exponent adjusted accordingly, thus reducing the number of mantissa bits by one.

The advantage of floating point significant digit arithmetic lies in the fact that the number of bits retained in the result of an arithmetic operation is the number justified by the size of the operands. This enables each calculation to be carried out in a manner that will give just the number of bits in the mantissa that the inaccuracies in calculation permit, e.g. a long string of calculation may yield only an 16-bit mantissa, assuming that the original data was normalised, 10 bits of accuracy have been lost in the course of the calculation.
7.1.5 Standard Floating Point Zero

The artificial limitation on the exponent \(e\), \(-2^{19} \leq e \leq 2^{19}\), is made to give a neater constant for representing standard floating point zero which is in fact represented by, mantissa 0 and exponent \(-2^{19}\). All zero sums, differences, products and quotients are set equal to standard floating point zero. If a transfer or copy is attempted from \(A^e\) with a number whose exponent is less than \(-2^{19}\) then the indicated store locations will be set at standard floating point zero.

7.1.6 Overflow

The overflow interruption facility is used in division to indicate a zero division and to indicate when a transfer or copy from \(A^e\) is attempted with a number whose exponent exceeds \(2^{19}\).
7.1.7 Floating Point Actions

The type of floating point arithmetic for a series of actions is determined by the setting of the control word in location '132' as follows:

(i) normalized arithmetic \( (132) = 37 \) in binary
\( (133) = 2^{13} \) in binary

(ii) significant digit arithmetic \( (132) = -37 \) in binary
\( (133) = -2^{19} \) in binary

In all actions \( N \) is the location of the first short word of the three half words to be occupied by the floating point number.

**Action 20.0.m : ADD**

**Definition:** Add \((N)\) to \((A^*)\)

**Notes:**
- (i) \((C)\) must be zero on LEO III.
- (ii) At the end of the action \((A) = (A^*)\) mantissa.

**Action 20.1.m : SUBTRACT**

**Definition:** Subtract \((N)\) from \((A^*)\)

**Notes:**
- (i) \((C)\) must be zero on LEO III.
- (ii) At the end of the action \((A) = (A^*)\) mantissa.

**Procedure:** The add and subtract actions are carried out as follows:

- (i) The larger exponent of the operands is copied into the \(A^*\) exponent location.

- (ii) \(B\) is cleared and the mantissa of the operand with lesser exponent is selected into \(A\) and is right shifted by the number of bits indicated by the difference between the exponents.

- (iii) The remaining mantissa is then added to \((A)\) or subtracted from \((A)\), in binary.

Amendment No. 6
December 1964
For normalised arithmetic the Procedure continues as follows:

3. (a) If $(A) > 1$ or if $(A) < -1$, $(AB)$ is right shifted 1 bit, $(A)$ is rounded off and copied into the $A^\#$ mantissa location and the $A^\#$ exponent increased by 1.

(b) If, after round off $(A) = 1$, $(AB)$ is shifted right one bit, $(A)$ copied into the $A^\#$ mantissa location and the $A^\#$ exponent increased by 1.

(c) If $-1 < (A) < 1$, $(AB)$ is left shifted until either bit 36 of $A = 1$ for positive $(A)$ or bit 36 of $A = 0$ for negative $(A)$, before round off. $A^\#$ exponent is decreased by the number of left shifts before $(A)$ is normalised.

(d) If rounded normalised $(A) = 1$, $(AB)$ is right shifted 1 bit $(A)$ copied into $A^\#$ mantissa and $A^\#$ exponent increased by 1.

(e) If rounded normalised $(A) = -1$, $(A)$ is left shifted 1 bit, $(A)$ is copied to $A^\#$ mantissa and $A^\#$ exponent decreased by 1.

(f) If rounded normalised $-\frac{1}{2} < (A) < 1$, $(A)$ is copied to $A^\#$ mantissa.

(h) If $(A) = 0$, $A^\#$ is set at standard floating point zero.

For significant digit arithmetic the Procedure continues as follows:

3. (a) If $(A) > 1$ or $(A) < -1$, $(AB)$ is shifted right 1 bit, $(A)$ is rounded off and copied to $A^\#$ mantissa and $A^\#$ exponent increased by 1.

(b) If rounded $(A) = 1$, $(AB)$ is shifted right 1 bit, $(A)$ is copied to $A^\#$ mantissa and $A^\#$ exponent increased by 1.

(c) If rounded $-\frac{1}{2} < (A) < 1$, $(A)$ is copied into $A^\#$ mantissa.
Section 7.7 (Cont'd)

Action 21.0.a: TRANSFER

Definition: Transfer \( A^* \) to \( N \).

Notes: (i) If \( A^* \) exponent \( > 2^{19} \), overflow will occur.
(ii) If \( A^* \) exponent \( < -2^{19} \), then standard floating point zero is transferred to \( N \).
(iii) At the end of the action \( A^* \) is equal to floating point zero and \( (A) = (A^*) \) mantissa.

Action 21.1.a: COPY

Definition: Copy \( A^* \) to \( N \).

Notes: (i) If \( A^* \) exponent \( > 2^{19} \), overflow will occur.
(ii) If \( A^* \) exponent \( < -2^{19} \), then standard floating point zero is transferred to \( N \) but \( A^* \) remains unchanged.
(iii) At the end of the action \( A^* \) is unchanged and \( (A) = (A^*) \) mantissa.

Action 22.0.a: MULTIPLY

Definition: Multiply \( A^* \) by \( (N) \) and place the product in \( A^* \).

Notes: (i) \( (C) \) must be zero on LSB III.

Procedure: 1. The \((N)\) mantissa is compared with \((34^1)\) and if it is not equal to \((34^1)\) then multiples 1 to 15 of \((N)\) mantissa are copied into compartments 34\(^1\) onwards, in complement form.
2. The modulus of the product of the mantissa of \((A^*)\) and \((N)\) is forced in \( A^* \) so that \( A^* \) contains the same number of significant bits as the modulus of the lesser mantissa.

For normalised arithmetic the procedure continues as follows:

Amendment No. 6
December 1964
2. (a) For non-zero operands the positive product is rounded.

(b) If the modulus of the rounded product = 0.5 and is negative the sign of \((A)\) is changed, \((A)\) shifted left one bit and copied into \(A^+\) mantissa. The \(A^+\) exponent is set equal to the \(A^+\) exponent plus \((M)\) exponent minus 2, minus the number of bit shifts taken to normalise the modulus of the greater mantissa.

(c) If the rounded product has the correct number of bits in \(A\) (see 2 above) then the sign of \(A\) is changed if the product is negative and \((A)\) copied to \(A^+\) mantissa.

\(A^+\) exponent is set equal to \(A^+\) exponent plus \((M)\) exponent minus 1, minus the number of bit shifts taken to normalise the modulus of the greater mantissa.

(d) If the modulus of the rounded product = 1 and is positive \((AB)\) is shifted right one bit, \((A)\) is copied to \(A^+\) mantissa and \(A^0\) exponent set equal to \(A^+\) exponent plus \((M)\) exponent plus 1.

(e) If the product is zero then \((A^+)\) is set equal to standard floating point zero.

For significant digit arithmetic the Procedure continues as follows:

2. (a) For non-zero operands the positive product is rounded.

(b) If the rounded product has one bit more than the modulus of the lesser mantissa, \((AB)\) is right shifted one bit. If the product is negative, the sign of \(A\) is changed, \((A)\) is copied to \(A^+\) mantissa and \(A^0\) exponent set equal to \(A^+\) exponent plus \((M)\) exponent minus the number of bits shifts taken to normalise the modulus of the greater mantissa.
(c) If the rounded product has the correct number of bits in \( A \) (see 2 above) then if the product is negative the sign of \( A \) is changed. \((A)\) is copied to \( A^e \) mantissa and \( A^e \) exponent set equal to \( A^e \) exponent plus \((N)\) exponent minus 1, minus the number of bit shifts taken to normalise the modulus of the greater mantissa.

(d) If the rounded product = +1, \((AB)\) is shifted right one bit, \((A)\) is copied into \( A^e \) mantissa and \( A^e \) exponent set equal to \( A^e \) exponent plus \((N)\) exponent plus 1.

(e) If the product = 0, with one operand non-zero, \((A)\) is copied to \( A^e \) mantissa and \( A^e \) exponent set equal to \( A^e \) exponent plus \((N)\) exponent minus 1, minus the number of bit shifts taken to normalise the greater mantissa modulus.

(f) If the product is zero and both operands are zero then \((A)\) is copied to \( A^e \) mantissa and \( A^e \) exponent set equal to \( A^e \) exponent plus \((N)\) exponent less 57.
Section 7.1.7 (Cont'd)

Action 22.1.a : DIVIDE

**Definition:** Divide \( A^a \) by \( N \) and place the quotient in \( A^o \).

**Notes:** Overflow occurs if \( N \) mantissa is zero.

**Procedure:**
1. The modulus of the quotient is formed in \( AB \) so that the number of significant bits in \( A \) is equal to the number of significant bits in the modulus of the lesser mantissa.
2. The next four bits of the quotient are held in Q10 of \( B \).

For normalized arithmetic the Procedure continues as follows:
2. (a) For non-zero operands the positive quotient is rounded.
2. (b) If the modulus of the rounded quotient has one bit more than the modulus of the lesser mantissa, \( AB \) is shifted right one bit. Otherwise \( AB \) is unchanged. If the quotient is negative the sign of \( A \) is changed, \( A \) is copied to \( A^o \) mantissa and \( A^o \) exponent set equal to:
   
   (i) for the modulus of \( A^o \) mantissa \( \geq \) the modulus of \( N \) mantissa:
   
   \[ A^o \text{ exponent less } N \text{ exponent plus the number of bit shifts taken to make } N \text{ mantissa modulus greater than } A^o \text{ mantissa modulus plus the number of bit shifts taken to normalise the modulus of } N \text{ mantissa} \]
   
   (ii) for \( (A^o) \text{ mantissa modulus } < (N) \text{ mantissa modulus}:
   
   \[ + (A^o) \text{ exponent less } N \text{ exponent plus the number of bit shifts needed to normalise the modulus of } (N) \text{ mantissa less 1 if the normalised modulus of } (A^o) \text{ mantissa is less than the normalised modulus of the mantissa of } (N) \]
   
   (c) If the rounded quotient is positive and equal to +1 in \( A \), then \( AB \) is shifted right one bit, \( A \) is copied into \( A^o \) mantissa and \( A^o \) exponent set equal to \( +(A^o) \text{ exponent less } N \text{ exponent} \).
Section 7.1.7 (Cont'd)

(d) If the rounded quotient is negative and equal to \(-\frac{1}{2}\) then \((A)\) is left shifted 1 bit, \((A)\) copied to \(A^*\) mantissa and \(A^*\) exponent set equal to \(A^*\) exponent less (N) exponent less 1.

(e) If the quotient is zero with (N) non-zero, then \((A^*)\) mantissa and \((A)\) are set to zero and \((A^*)\) exponent is set equal to \(-2^{127}\).

(f) If (N) is zero, \(A^*\) is not changed and the overflow condition set.

3. For significant digit arithmetic the procedure continues as follows:

(a) For non-zero operands the positive quotient is rounded.

(b) If the modulus of the rounded quotient has one bit more than the modulus of the lesser mantissa, \((AB)\) is shifted right 1 bit. The sign of \((A)\) is changed if the quotient is negative, \((A)\) is copied to \(A^*\) mantissa and \(A^*\) exponent set equal to:

(i) for \(A^*\) mantissa modulus \(> (N)\) mantissa modulus:
\[ A^*\) exponent less (N) exponent plus the number of bit shifts to make \((B)\) mantissa modulus greater than \((A^*)\) mantissa modulus plus the number of bit shifts needed to normalise the modulus of \((N)\) mantissa.

(ii) for \(A^*\) mantissa modulus \(< (N)\) mantissa modulus:
\[ 1+(A^*)\) exponent less (N) exponent plus the number of bit shifts needed to normalise the modulus of \((N)\) mantissa less 1 if the normalised modulus of \(A^*\) mantissa is less than the normalised modulus of \((N)\) mantissa.
(a) If the rounded quotient is positive and equal to $+1$, then $A_n$ is right shifted 1 bit, $A^*$ copied to $A^*$ mantissa and $A^*$ exponent set equal to $A^*$ exponent less $(N)$ exponent +1.

(b) If the rounded quotient has the correct number of bits in $A$ (see 2 above) then the sign of $A$ is changed if the product is negative. $(A)$ is copied to $A^*$ mantissa and $A^*$ exponent set equal to:

(i) for $A^*$ mantissa modulus $>(N)$ mantissa modulus:

$$(A^*)^*$$ exponent less $(N)$ exponent plus the number of bit shifts to make $(N)$ mantissa modulus greater than $(A^*)$ mantissa modulus plus the number of bit shifts needed to normalise the modulus of $(N)$ mantissa.

(ii) for $(A^*)$ mantissa modulus $<(N)$ mantissa modulus:

$$(A^*)^*$$ exponent less $(N)$ exponent plus the number of bit shifts needed to normalise the modulus of $(N)$ mantissa less 1 if the normalised modulus of the $(A^*)$ mantissa is less than the normalised modulus of the mantissa of the number $(N)$.

(c) If the quotient is zero, with $(N)$ non-zero, then $(A^*)$ mantissa and $(A)$ are set to zero, and $(A^*)$ exponent is set equal to $1 + (A^*)$ exponent less $(N)$ exponent plus the number of bit shifts needed to normalise $(N)$ mantissa.

(d) If $(N)$ is zero and $(A^*)$ is non-zero, then $A^*$ mantissa is not changed, $A^*$ exponent is set equal to $A^*$ exponent less $(N)$ exponent plus 37 and the overflow condition is set.

(e) If both $(N)$ and $(A^*)$ are zero then $A^*$ mantissa and $(A)$ are set to zero and $(A^*)$ exponent is set equal to $(A^*)$ exponent less $(N)$ exponent plus 37 and the overflow condition is set.
7.1.8 Special Select and Copy Actions

Action 17.0.m : SPECIAL SELECT

**Definition:** Select \((N')\) into \(A\) if \(N\) is even, otherwise select \((N+1')\) into \(A\).

**Note:** No change is made between sign and modulus and sign and complement form.

Action 17.1.m : SPECIAL COPY

**Definition:** Copy \((A)\) to \(N'\) if \(N\) is even, otherwise copy \((A)\) to \(N+1'\).

**Note:** No change is made between sign and modulus and sign and complement form.

These actions are designed to be used in conjunction with Floating Point Arithmetic.
7.2 Double Length Arithmetic

7.2.1 Introduction

In this mode of arithmetic, numbers such as $(N+2)'$ and $(N)'$ in the store are manipulated as 20-digit numbers, in sign and modulus form. The sign must be given by both the sign bits of $N+2$ and $N'$, e.g. as set by action 30 below. $(AB)$ is manipulated as a 20-digit number in sign and complement form. The sign must be given by the sign bit of $A$, and the sign bit of $B$ must be zero.

7.2.2 Excess Constants

The radix set is defined by the contents of $C$ as follows:

- Digits 1 - 5 of double length number defined by digits 1 - 5 of $C$
- Digits 6 - 10 of double length number defined by digit 5 of $C$
- Digits 11 - 15 of double length number defined by digits 1 - 5 of $C$
- Digits 16 - 20 of double length number defined by digit 5 of $C$

7.2.3 Double Length Actions

**Action 30.C.a : DOUBLE LENGTH TRANSFER**

**Definition:** Transfer $(AB)$ to $(N+2)'$, $(N)'$.

**Notes:**

1. Register $A$ and $B$ are left clear at the end of the actions.
2. $(AB)$ is a 20-digit number, held in sign and complement form, with the sign bit of $B$ zero.
3. $(N)'$ and $(N+2)'$ are held in sign and modulus form.
4. If $A$ contains $-0$ and $B$ contains $+0$ (i.e., double length minus zero) overflow will occur on $LEO 326$, but on $LEO III$ the result is $-0$ in $N$ and $+0$ in $N+2$ (no overflow).

**Procedure:**

1. Copy $(b)$ with the sign bit of $A$ into $N'$.
2. If $A$ is positive copy $(A)$ into $N+2'$.
3. If $A$ is negative and $B$ is non-zero copy $(A)$ + 1 into $N+2'$.
Section 7.2.3 (Cont'd)

**Action 30.1.m : DOUBLE LENGTH COPY**

**Definition:** Copy (AB) into (N+2)', (N)'.

**Notes:**
1. (AB) is a 20-digit number, held in sign and complement form, with the sign bit of B zero.
2. (N)' and (N+2)' are held in sign and modulus form.

**Procedure:** As for action 30.0.m.

**Action 30.2.m : DOUBLE LENGTH ADD**

**Definition:** Add (N+2)', (N) to AB

**Notes:**
1. (AB) is a 20-digit number held in sign and complement form.
2. (N)' and (N+2)' are held in sign and modulus form.
3. Overflow occurs if the sum exceeds 20 digits.

**Procedure:**
1. Add (N)' to B and (N+2)' to A
2. If the sum of (N)' and B exceeds 10 digits and is positive, increase (A) by one.
3. If the sum of (N)' and B exceeds 10 digits and is negative, reduce (A) by one.

**Action 31.1.m : DOUBLE LENGTH SUBTRACT**

**Definition:** Subtract (N+2)', (N)' from (AB)

**Notes:**
1. (AB) is a 20-digit number held in sign and complement form.
2. (N)' and (N+2)' are held in sign and modulus form.
Section 8

8. SEQUENCE CHANGING AND MODIFICATION

8.1 Sequence Changing

The sequence change actions enable the programmer to arrange for instructions to be obeyed in a different order from the normal sequential order described in section 5.6.

The sequence change actions place the address part of the instruction in the sequence control register so that the next instruction will be selected from the address specified instead of the next compartment in sequence.

Most of the sequence change actions are conditional on the result of a test made on the accumulator.

Action 24.1.C : UNCONDITIONAL SEQUENCE CHANGE

Definition: Change sequence to N.

Action 27.4.a : TEST ACCUMULATOR AND SEQUENCE CHANGE

Definition: Change sequence to N if the test specified by d and a is satisfied, otherwise continue to the next instruction.

- d = 0 specifies : test (a)
- d = 1 specifies : test (AB)
- a = 0 specifies : test zero
- a = 1 specifies : test non-zero
- a = 2 specifies : test > zero
- a = 3 specifies : test negative

Notes:

(i) The sign of (A) is specified by the sign bit of A, hence actions 27.1.2 and 27.1.3 are exactly the same as 27.0.2 and 27.0.3.

(ii) If A or AB has only the sign bit set, the tests 'negative' and 'non-zero' will both be satisfied.
Section 6.2

8.2 Subroutines

Two actions are provided to allow the programmer to write subroutines which may be entered from any part of the programme. The return from a subroutine is to the instruction following the most recent entry to the subroutine.

Subroutines must have a spare compartment immediately before their first instruction - this compartment is used as a link. On entering the subroutine, the return address is automatically placed in the link compartment, and is then referred to when the subroutine is to be left.

The 'enter subroutine' action will be in the main programme, the 'leave subroutine' action at the end of the subroutine.

**Action 26.0.0 : ENTER SUBROUTINE**

**Definition:** Place the location of this instruction +1 in compartment N and change sequence to N+1.

**Note:** N is the location of the link compartment, N+1 the location of the first instruction in the subroutine.

**Action 26.0.1 : LEAVE SUBROUTINE**

**Definition:** Change sequence to the location specified by (N).

**Note:** N is the location of the link set up by action 26.0.0.
8.3 Direct Modification

The modification system is described briefly in section 5.5.

The programmer can arrange for the least significant 15 bits of any modification register to be added to the address of a modifiable instruction after this has been selected into OA in the normal way and after the store division bits from SC have been added to it.

The modification register to be used within the group set is specified by the modification digits of the instruction. If these digits are zero the instruction is not modified.

The modification group (which is normally the same throughout a programme) may be specified by an instruction at the beginning of the programme. This is normally done by the Master Programme before entry to the programme. The contents of a modification register are set by programme instructions.

All modification is carried out in binary and hence numbers in modification registers must be in binary form. Negative numbers cannot be held in modification registers. However, since all modification is carried out on 15-bit numbers and any overflow from the most significant end is ignored, it is possible to modify by \((32768 - x)\) giving the effect of modifying by \((-x)\). (32767 is the largest number which may be held in 15 bits). Hence when a number attempts to reach 32768 it will appear as zero.

To assist the programming of repetitive sequences, modification registers are divided into two parts:

<table>
<thead>
<tr>
<th>40</th>
<th>35</th>
<th>2120</th>
<th>15</th>
<th>1 Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>End Value</td>
<td></td>
<td>Modifier</td>
</tr>
</tbody>
</table>
A facility is provided to step on the modifier by a specified increment (which may be positive or negative) and compare it with the end values. A sequence change is made according to the result of the comparison. 'Negative' end values are specified in the same way as 'negative' modifiers.

Note: This comparison is made by subtracting bits 1 to 20 from bits 21 to 40 and is thus not restricted to 15-bit numbers as in modification. Thus if one and only one of the two parts (modifier and end value) is 'negative', then the positive part must also have 32768 added to it to ensure correct comparison.

**Action 24.1.1 : SET MODIFICATION GROUP**

**Definition:**
Set bits 14 and 15 of I (the indicator register) equal to bits 14 and 15 of (N).

**Notes:**
(i) Bits 14 and 15 of I specify the modification group.
(ii) The group cannot be set by the 'set indicators' action.

**Action 25.0.m : STEP ON AND TEST DIRECT MODIFICATION REGISTER**

**Definition:**
Step on the modifier of modification register m by N (literal), and test if it is now equal to the end value. If it is not, carry on to the next instruction, if it is, skip one instruction.

**Notes:**
(i) m specifies the modification register within the current modification group.
(ii) N is in sign and modulus form. Bit 13 (N3) gives the sign, and bits 1 to 12 the modulus, which may not exceed 4095.
(iii) The action is not directly modifiable.
(iv) If m = 0 computations 0', 6', 16' or 24' according to the setting of the modification group are treated as a modification register and stepped on accordingly.
(v) On LBO 326, overflow cannot occur when the new modifier is formed.
**Action 26.1.a: SET DIRECT MODIFICATION REGISTER**

**Definition:**
Copy \((N')\) to modification register \(m\).

**Notes:**
(i) \(m\) specifies the modification register within the current group.
(ii) The action is not directly modifiable.
8.4 Indirect Modification

The address part of any computer instruction may be modified indirectly by the contents of any short compartment by inserting a special instruction before the action concerned. All addresses including literals may be modified in this way.

These special instructions select the address of the next instruction into Q4, then add the modifier, and finally select the action part of the next instruction into Q1 and transfer control to that instruction in the normal way.

Actions 24,0.0 and 24,0.1 have the same result as direct modification – they add the least significant 15 bits of the modifier to the full 15-bit address of the next instruction.

Actions 24,0.3 and 24,1.2 however, add the least significant 15 bits of the modifier to the 13-bit address of the next instruction – i.e. the store division bits of sequence control are not copied to Q4 when the next instruction address is selected.

The contents of any long compartment may be considered as comprising a modifier and an end value (laid out as those of a modification register) and may be stepped on and tested in a similar way to a modification register. The same rules as for direct modification apply to ‘negative’ modifiers and end values.

Action 23.0.0: STEP ON AND TEST INDIRECT MODIFIER

Definition: Step on the modifier of the long compartment L by \( \bar{W} \) (l literal) and test if it is now equal to the end value. If it is not, skip one instruction, if it is, skip two instructions.

Notes:

(i) The location L of the indirect modifier is specified by bits 1 to 15 of the compartment following the 'step on and test' instruction.

(ii) \( \bar{W} \) is in sign and modulus form. Bit 13 (M3) specifies the sign and bits 1 to 12 the modulus, which may not exceed 4095.

(iii) On LEO 326, overflow cannot occur when the new modifier is formed.
Section 24.0 (Con'd)

**Action 24.0.0 : MODIFY FULL ADDRESS OF NEXT INSTRUCTION, SEARCHING FOR A POSITIVE MODIFIER**

**Definition:** Search locations $N$, $((N))$, $((((N))))$, etc. until a location with positive contents is found. Then modify the address of the next instruction by the contents of bits 1 to 15 of that location after adding the division number from SC.

**Notes:**
1. If the first compartment searched has negative contents bits 1 to 15 of the modulus of those contents specify the location of the next compartment searched and so on.
2. Lockout is inhibited on compartments 64 to 127. (In LSO 326, also on compartments 192 to 255).

**Action 24.0.1 : MODIFY FULL ADDRESS OF NEXT INSTRUCTION**

**Definition:** Modify the address of the next instruction by bits 1 to 15 of $(N)$ after adding the division number from SC.

**Action 24.0.2 : MODIFY ADDRESS OF NEXT INSTRUCTION, SUPPRESSING CURRENT DIVISION**

**Definition:** Modify the address of the next instruction by bits 1 to 15 of $(N)$. Suppress the addition of the division number from SC.

**Action 24.1.2 : MODIFY ADDRESS OF NEXT INSTRUCTION, SUPPRESSING CURRENT DIVISION AND SEARCHING FOR A POSITIVE MODIFIER**

**Definition:** Search locations $N$, $((N))$, $((((N))))$, etc. until a location with positive contents is found. Then modify the address of the next instruction by the contents of bits 1 to 15 of that location suppressing the addition of the division number from SC.

**Notes:**
1. If the first compartment searched has negative contents bits 1 to 15 of the modulus of those contents specify the location of the next compartment searched and so on.
2. Lockout is inhibited on compartments 64 to 127. (In LSO 326, also on compartments 192 to 255).

Amendment No. 8
December 1954
8.5 Multiple Modification

Sometimes it is necessary to modify an address by the sum of two modifiers. This sum must be formed by arithmetic actions in the program.

When arithmetic is being done on modifiers in this way it is sometimes necessary to be able to select a literal, and to have it prefixed by the current division number. Actions 1.1.1, 1.1.2 and 1.1.3 do not give this facility and so a special action is provided.

Action 24.0.2 : SELECT LITERAL AND STORE DIVISION NUMBER

Definition: Select N into A.

Note: N is prefixed by the store division number from SC in bits 14 and 15.
9. GENERAL DESCRIPTION OF INPUT AND OUTPUT

9.1 Introduction

The various input devices used in conjunction with LSO III transmit information signals in a variety of forms. Before reaching the store access control unit, the signals must be reduced to a standard form. On the other hand when information is output, it is taken from the store in the required form, no conversion being necessary.

The equipments used to effect the necessary transformations are called 'assemblers', different assemblers being employed in conjunction with different types of peripheral equipment. Any desired combination of up to 8 assemblers can be linked to the store access control unit, the linkages being termed 'channels'. Each assembler is linked via 'routes' to one or more peripheral devices.

The types of assembler available are detailed in sections 11, 13 and 15; features common to all assemblers are now described.

9.2 Assemblers

9.2.1 Purpose

An assembler functions in response to an instruction from the coordinator specifying input or output. Having received the instruction the assembler proceeds independently of the coordinator (which immediately passes on to the next instruction) to transfer information between the peripheral device specified by the instruction and the store access control unit, checking it and changing its form if necessary (as outlined in section 9.1) as it does so.

9.2.2 Autonomous Working

Since an assembler functions independently of the coordinator, it follows that information may be passing down one route of each of the assemblers simultaneously, and that the arithmetic unit may also be in operation at the same time.

Each piece of peripheral equipment is under assembler control throughout any single operation except 'rewind', 'unload' and 'set to manual', during which the assembler is free to control any operation on another route.
Section 9.2.3

9.2.3 Channels and Routes

A channel is an information path between the store access control and an assembler or the main frame. Each store access control unit has 8 channel outlets for assemblers numbered from 0 to 7. The lowest numbered channels have priority when store access is required. The allocation of channel numbers is described in section 4.4.

A route is an information path between an assembler and the recording medium in a peripheral device. An assembler can normally have up to 8 routes numbered from 0 to 7. The particular route allocated to any device is chosen by setting a switch on the device. Route numbers, which are specified in input and output instructions, are defined as (8 x channel number + route number).

9.2.4 Annexes Starting Locations

Each route in use requires an annex in the store large enough to contain a block of data. The annex allotted to any route R, where R = (8 x channel number + route number), is specified by holding in compartment (64+R) the address of its first compartment. Each assembler contains a register of 14 bits capacity, known as the 'address register', into which the contents of compartment (64+R) are automatically placed when an input or output instruction is obeyed. This address is stopped on each time a word of information is transferred, and thus specifies the long compartment in the store into which each successive word of data is to be written or from which each successive word of output is to be copied. The contents of the address register continue to change in this way until the end of the block is reached.

9.2.5 Conversion to Standard Form

Information transferred between input assemblers and the store must be in the standard alpha form with a control quartet and basic quartet for each character (see 3.5). Character signals are transmitted from input devices in a serial form, consisting of a control sextet and basic quartet (Appendix A). Input assemblers convert each sextet character signal to the standard outlet form in the following manner.

The basic quartet is unchanged while the control quartet of the character is formed from the two-bit control received, a bit 3 (value 4) being added if the basic quartet value is non-zero. If the basic quartet has value 0, nothing is added to the original two-bit control value, e.g.
### Character on Paper Tape vs Character Signal vs Alpha Form

<table>
<thead>
<tr>
<th>Character on Paper Tape</th>
<th>Character Signal</th>
<th>Alpha Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>001 0011</td>
<td>0101 0011</td>
</tr>
<tr>
<td>0 (zero)</td>
<td>11 0000</td>
<td>0011 0000</td>
</tr>
</tbody>
</table>

In these cases where output equipment requires information in character sextets, these are obtained directly from the store, and the assembler does no conversion.

### 9.2.6 Checks

The transfer of data by an assembler between input/output equipment and the store is in two parts: along the route and along the channel. The normal unit of information for transfer along the route is a character. The unit of information for transfer along the channel is a word.

Parity checks are applied to the transfer along the channel as follows:

(a) **On Input**

The assembler sends the word to store accompanied by a 'word parity bit'. This is formed by the assembler with the value 0 or 1 in order to make the total number of '1' bits in the word odd.

Parity is checked by the store access control unit on receipt of the word. If it is incorrect the channel is closed.

(b) **On Output**

A similar process takes place, except that parity is formed by the store access control unit, and checked by the assembler on receipt of the word. Incorrect parity causes the channel to be closed.

Parity checks are applied to the information on the external medium and to the transfer along the route on some equipment, these are fully described in the section dealing with the appropriate assemblers.

In addition there are power supply checks and various interlock checks peculiar to the type of assembler, and lockout checks which are described in section 17.2.
9.2.7 Conditions affecting Channels and Routes

Route Closed
A route is defined as 'closed' if no information can pass through it.

A route is closed in the following instances:

(i) If the channel to which it is connected is closed.
(ii) If the peripheral equipment selected for the route is set to 'manual'.
(iii) If no peripheral equipment is selected for the route.
(iv) If the power supply of the selected peripheral equipment or any part of it is switched off.
(v) If certain mechanical interlocks in the selected peripheral equipment are not satisfied, e.g. if the doors are open on a magnetic tape deck, if no vacuum is present in magnetic tape reservoirs, if the recording medium is exhausted, etc.

A route may be opened by removing the condition which caused it to be closed.

Channel Closed
A channel is defined as 'closed' if no information can flow through it.

A channel is closed in the following instances:

(i) If the 'channel closed' key on the assembler or control desk is operated.
(ii) If the 'test' key on the assembler is operated.
(iii) If a fault of any kind is detected during the processing of a block, except character parity failure on magnetic tape, or 'paper tape'.
(iv) If the power supply of the assembler is switched off.
(v) If the lockout condition for that channel is set (see section 16.2).
A channel may be opened by removing the condition which caused it to be closed and operating the 'assembler reset' key on the assembler (except in cases (i) and (ii) when this key need not be operated).

**Channel Engaged**
A channel is defined as 'engaged' if information is passing through it in response to an earlier computer instruction.

**Route Engaged**
A route is defined as 'engaged' if its associated channel is engaged.

A route is also engaged if it is a magnetic tape route which is rewinding. In this case the associated channel is not engaged, and if the route is tested the interrupt flip-flop will not be set.

**Doubtful Block**
This condition is set when the 'assembler reset' key is pressed (i.e. when a channel which has been closed is re-opened, except after manual closure). The condition is also set when parity failure occurs on magnetic tape or paper tape and when a card is misread (on the ICT card reader only).

**Warning of End**
Some peripheral devices have warnings to signal that the medium being used is nearly exhausted and needs replacing or replenishing. When this occurs a warning of end indicator is set on the assembler.

**Route Available**
A route is defined as 'available' if the route and its associated channel are not closed or engaged.

The conditions 'route closed', 'route engaged', 'doubtful block' and 'warning of end', may be tested by programme instruction.
9.2.8 Interruption Requests (see section 17.1 for details)

An assembler may request interruption on behalf of any route if the assembler interrupt flip-flop is set and the route is available (see section 16.1). The interrupt flip-flop is set if the route is tested and found to be engaged or if the equipment is switched from 'manual' to 'computer' mode.

Interruption by any assembler causes interrupt flip-flops on all assemblers to be reset.
10. INPUT EQUIPMENT

10.1 Elliott 84/25 Photo-Electric Paper Tape Reader

10.1.1 Linkage with the Computer

Paper tape readers are allocated routes on the paper tape and card input assembler (see section 11.1).

10.1.2 Type of Tape Read

The standard tape has 7 tracks and information is punched according to the standard character code described below. Non-standard codes on 5, 6 or 7 track tape can be read if a key is set on the tape reader. Provision is made in the design for a possible later conversion to read 8 track tape.

10.1.3 Standard Character Code

Each row of holes across the tape represents one character. The seven positions are interpreted as a control digit and a basic quartet (together described as a character sextet), and an odd parity. The sextet code, which is given in Appendix A.1, is consistent with the code used with the printer and punched cards.

The alpha/decimal characters are read from the tape in the order in which they were punched, with the exception of the 'erase' character (5/15) and blank tape which are ignored. (Note that the 'space' character 0/0 is not ignored, since it has a hole in the odd-parity position).

When non-standard tape (i.e., using some other code) is being read, characters with the same representation as 'block end' and 'erase' of the standard code will be read, and so will blank tape. Each read instruction for non-standard tape will cause five characters only to be read, and parity checking is suppressed. The programme will be required to convert the data.
10.1.4 Speed

The tape is read at a maximum speed of 1000 rows per second. There is no significant delay before this speed is reached.

10.1.5 Control and Operation

As the next part of each character is read from standard tape an odd parity bit is formed in the assembler. This is compared with the parity bit read in from the tape and if these fail to agree an assembler alarm is set which sets the 'doubtful block' condition, and the reader stops before the next character is reached.

The reading of non-standard tape is permitted by setting a key on the reader, and this also lights a 'non-standard' indicator on the reader. The Channel Warning lamp will be lit on both the assembler and the operators' control desk.
10.2 Elliott 842X Photo-Electric Punched Card Reader

10.2.1 Linkage with the Computer

The readers are allocated to routes on the paper tape and card
input assembler (see section 11.1).

10.2.2 Type of Card Read

The only card which this reader accepts is the standard 80-column,
12-row Hollerith card with rectangular holes, and with column
identification marks printed on the reverse side.

10.2.3 Character Codes

Cards are normally used to hold information in fixed field form
(see section 16.2). Each character is represented in one column on the
card by the standard sextet code shown in Appendix A.2. One card can
contain up to 80 characters, which is equivalent to a block of up to
16 words.

The sextet code differs slightly from the British Standard code as
used for printer and paper tape, in order that the numbers 10 and 11
may be represented by the top two rows, as is conventional with punched
cards. The top three rows 10, 11 and 0 contain the control sextet (with
values 1, 2 and 3 respectively), and the remaining rows (1 to 9) the
basic quartet (value as numbered). For the value zero in either the
basic quartet or control sextet no hole is punched in that region of the
card. A minus sign is represented by 10 (control sextet 1, basic quartet
0) punched in the most significant of the columns allotted to hold the
number. A numeral other than 10, 11 or 0 may be represented in this
column as well as the minus sign.

Cards can also be punched in non-standard (e.g. binary) code.
These are interpreted by the assembler in the way described in section
11.6.
10.2.4 Speed

Cards are read photo-electrically at the rate of 360 cards per minute.

The input hopper has a capacity of 2000 cards. The output stacker has a capacity of 2500 cards - the equivalent of approximately 7 minutes continuous reading time.

The output stacker is detachable in order to simplify unloading and minimise operational delays. When the stacker is full, it is removed and replaced by a spare. This procedure causes a delay of only 5 seconds and allows the cards to be unloaded when convenient.

10.2.5 Control and Operation

The cards are read sideways from left to right, one column (i.e. one character) at a time.

As the Elliott card reader does not accurately control the position of the card relative to the drive mechanism, it is not possible to obtain signals directly from the drive to indicate the instant of time when each column is in the 'read' position. For this reason column identification marks are printed on the back of the card one for each column position. These marks are detected photo-electrically and counted; when a mark is detected, the holes in the column are read photo-electrically. An 'end of card' signal is generated when all twelve photocells are energised simultaneously. On non-standard cards, where fully punched columns may occur, the signal is not generated before 80 columns have been read. If a block and character is punched on a card, it will be read in, but does not stop reading process. No provision is made for reading cards which are not marked on the reverse side with column identification marks. This means that any information on unmarked cards must be copied onto marked cards.

When the penultimate card is being read, the last card moves out of the hopper into a position ready for reading, leaving the hopper empty. The input hopper is fitted with an end of medium alarm which is activated when the hopper is empty, lighting the Cards Exhausted lamp. When the reader is in this condition no further cards can be read; thus the last should be a dummy card.
10.3 ICT 593 Photo-Electric Punched Card Reader

10.3.1 Linkage with the Computer

The readers are allocated to routes on the paper tape and card input assembler (see section 11.1).

10.3.2 Type of Card Read

Standard 80-column, 12-row IC7 or IBM cards with rectangular holes are read.

Column identification marks are not required.

10.3.3 Character Codes

Information is represented on the punched cards in the form and character code described in section 10.2.3.

10.3.4 Speed

Cards are read photo-electrically at a maximum rate of 600 cards per minute. Both the input hopper and output stacker have a capacity of 2000 cards, however the equipment is designed to permit cards to be loaded and unloaded continuously without halting the reader.

It should be noted that because of the cyclic clutch arrangement in the reader the maximum rate of 600 cards per minute can only be maintained if the interval between the assembler becoming disengaged after reading a card and the next input instruction does not exceed 4 milliseconds.

10.3.5 Control and Operation

The cards are fed from the input hopper and gripped by a pair of feed rollers which pass the card through the information sensing station. The sensing station consists of a bank of twelve photovoltaic cells mounted transversely beneath the card track, and illuminated from above by a lamp assembly. Holes punched in each card are sensed column by column from left to right. A second pair of feed
rollers located beyond the sensing station grip the card as column 67
is being read, and pull the rest of the card past the sensing station.
Between the reading of columns 67 and 69 the card is under the
influence of both rollers. The card then passes through a third pair
of rollers and reaches the deflector paws; whose position determines
the destination of the card, either into the bottom of the main output
stacker or into a reject hopper.

Mounted on the same shafts as the first and second feed rollers
are timing wheels which photo-electrically generate 16 clock pulses per
column of the card read. When the leading edge of the card is detected
in the reading station, the clock pulses from the first timing wheel
are counted and the nominal position of the first card column established.
Thereafter three clock pulses are selected from the sixteen generated
such that when a hole appears in the card all three pulses should be
embraced by the data signal. If the data signal is not in its nominal
position due to card skew, card slip in the drive rollers, or misalign-
ment of data holes, the signal will not coincide with all three pulses.
In such cases the following action is taken:

(i) If one pulse fails to coincide with the data signal the counting
    circuits are adjusted so that the three pulses occur earlier or
    later as appropriate in the next column to be read, thus providing
    constant checking and correction of reading accuracy.

(ii) If two or three pulses fail to coincide with the data signal the
    degree of misalignment is such that misreading may have occurred.
    In this case the card is channelled into the reject hopper, and
    the 'doubtful block' flip-flop in the assembler is set.

In order to reduce mechanical wear, a facility is provided to
automatically switch off the main drive if the card reader has not been
called in the last few minutes. The motor is automatically switched on
again when the card reader is selected by a programme instruction.
10.4 24-Hour Clock

10.4.1 Linkage with the Computer

The clock is linked directly to the store access control unit.

10.4.2 Character Code

Input from the clock consists of numeric characters indicating the time of day as shown by the clock. The current time is held in short compartment 156, one quartet being allocated for each character as follows:

<table>
<thead>
<tr>
<th>Hours</th>
<th>mins.</th>
<th>1/10 min.</th>
</tr>
</thead>
</table>

10.4.3 Speed

The time taken to input one word is the normal store cycle time.

10.4.4 Control and Operation

One word is input to compartment 156 automatically every six seconds without any instruction being needed. The clock is reset by hand when necessary. It operates from the mains power supply and its accuracy is as good as this allows.

No key register is associated with the clock.
10.5 Millisecond Timer

10.5.1 Linkage with the Computer

The millisecond timer is linked directly to the store access control unit.

10.5.2 Character Code

Input from the millisecond timer consists of 4 numeric characters indicating time elapsed since the timer was last reset to zero. The current elapsed time is held in the four less significant quartets of short compartment 157 as follows:

1/10  1/100  1/1000
Second Second Second Second

10.5.3 Speed

The time taken to input is the normal store cycle time.

10.5.4 Control and Operation

Four characters are input to compartment 157 automatically every millisecond without any instruction being needed.

The timer is reset to zero by a 19.0.1 instruction with the channel number of the timer specified in bits 4 to 6 of the address. Bits 4 to 5 of address should be zero. No key register is associated with the timer.
10.6 Burroughs B125/2 Card Reader

10.6.1 Linkage with the Computer

The readers are allocated to routes on a modified paper tape and card input assembler (see section 11.1).

10.6.2 Type of Card Read

The B125/2 will read 51, 60, 66 or 80-column, 12-row cards with rectangular holes.

Column identification marks are not required.

10.6.3 Character Code

Information is represented on the punched cards in the form and character code described in section 10.2.3.

10.6.4 Speed

Cards are read photo-electrically at a maximum rate of 800 cards per minute, irrespective of the number of columns on the cards in use.

The input hopper, capacity 2000 cards, and the output stacker, capacity 2500 cards, are capable of being continuously loaded and unloaded during operation.

The clutch is of the non-cyclic type, i.e., there are no programme limitations as to when the reader may be called; hence any reading speed is achievable up to 600 cards per minute.

10.6.5 Control and Operation

The cards are fed from the input hopper and passed through an acceleration roller assembly into the read station, which comprises a bank of twelve cells and an associated lamp assembly.

After the leading edge of a card has passed through the read station, two sets of transport rollers convey it past a horizontal mask assembly, which provides the basic timing pulses for the 80 columns. The card then travels around a drum and into the stacker. Note that with the B125/2 there is no facility for rejecting cards as with the I.C.T. machine.

Each column of the card is effectively read twice, the time between successive readings being approximately 25 microseconds. The data read at the first attempt is transferred to the assembler, but if the second reading does not correspond with the first, a 'doubtful block' condition is also sent; a mispunched hole is the usual cause.
Section 10.6 (Cont'd)

To prepare the reader for the size of card to be processed, it is only necessary for the operator to make simple adjustments to the hopper and stacks; for each card size, these adjustments operate a different microswitch, which controls the number of column pulses sent to the assembler.
Section 10.7

Leo-Parnall AUTOLECTOR On-Line Document Reader

This section describes the principal features of AUTOLECTOR, a device used for on-line document reading with LEO III, LEO 326 and LEO 360.

AUTOLECTOR reads hand-marked documents similar in design to those used with LECTOR, transferring data directly into the computer.

10.7.1 Linkage with the Computer

The system has been designed to provide for up to four AUTOLECTORS working concurrently on a single LEO III or LEO 326/360 channel. No assembler is required for linking the equipment to the channel. No route selection switches are provided; the route occupied by a given AUTOLECTOR is determined by its position along the bus-bar connecting the equipment to the computer. A priority system is built into AUTOLECTOR to establish which machine has priority of access to the store when more than one is connected.

10.7.2 Data Representation

Data is represented on the documents in the form of short horizontal marks made in ink or pencil in designated columns. The selection of columns to be read may be varied by means of a plugboard.

For a description of document design and marking see Part 2, section 3 of this volume.

10.7.3 Speed

Documents for input are packed into cassettes. The machine is designed for continuous loading and unloading; input and output capacities are 5,000 and 2,000 documents respectively. The rate at which documents are fed depends to some extent on the condition of the documents and the way they have been packed into the cassettes, but for operational purposes a maximum rate of 240 documents per minute can be assumed.

10.7.4 Control and Operation

Documents are fed from cassettes designed to suit the size of document in use. The loaded cassettes are placed on the input conveyor belt, which aligns the first cassette in the correct position relative to the feeding unit.

Documents are fed from the cassettes by means of a vacuum system operating through holes in two moving rubber belts. A counter-rotating vacuum drum pushes back any additional documents that may have been withdrawn; double-feed prevention is further assisted by a blast of air that separates the top edge of the documents. The next document is fed when the next set of holes in the transport belts is aligned with the vacuum 'chambers'. Feeding is stopped when the vacuum is released.

Amendment No. 7
August 1964
When a document has been withdrawn from the cassette, it follows the vertical motion of the belts and is turned through 90° so that it is in a horizontal plane. It then passes through a squaring-rail assembly, where it is correctly positioned for transfer to the reading drum. This vacuum drum holds the document flat while it is optically scanned.

Another belt carries the documents from the reading area to the output conveyor, which consists of a large flat surface in a state of continuous vibration such that documents are slowly moved to one end. In addition to the foregoing arrangement, there is also a facility whereby selected or faulty documents are removed from the drum and deposited straight into a 'Nurum' Bin (capacity 50 documents).

Jam-detection mechanisms are fitted at various points throughout the document path.

10.7.5 The Input Instruction and Annexes Requirements

The input instruction (Computer code action 'I9.0.1') causes the data on one document to be read from a specified channel and route to a specified store area.

Once document feeding has been initiated by an input instruction, however, up to four additional documents will be in transit before the second document enters the reading station. It is therefore possible that the scanning of a document may not be preceded by an input instruction. In such cases up to five documents in all will be read into the store before feeding is halted.

The length of the annexes area allocated in store must therefore be arranged such as to accommodate up to 5 data blocks (1 per document). Only one fifth of the area will normally be used, but as a programme has no automatic indication of whether additional documents have been read in, it is necessary to test the condition of the annexes prior to each input instruction, to determine whether the block(s) required are already present.

10.7.6 Document Design

For a description of document design and marking, see Part 2, section 3 of this manual. It is known that some of the document tolerance considerations are less stringent on AUTOLECTOR than on LECTOR. It is therefore anticipated that the tolerances of documents for use on AUTOLECTOR may, in due course, be relaxed.

Special requirements which apply to AUTOLECTOR are listed below:

Amendment No. 7
August 1964
(1) **Overall Dimensions**

Document sizes must lie within the following limits:

- **Length**: 9" - 16"
- **Width**: 7½" - 6½"

with the proviso that the length must always be at least 23% greater than the width.

(2) **Right-Hand Margin**

At least 0.3" of unmarked and unprinted paper must appear on the right-hand side of the document.

(3) **Top Margin**

At least $\frac{3}{8}$" of unmarked and unprinted paper must appear at the top of every document.
11. CARD AND PAPER TAPE INPUT ASSEMBLER

This section describes special features of the paper tape and card input assembler. Features common to all assemblers are dealt with in section 9.

11.1 MEDIA AND ROUTES

This assembler controls input from punched card and punched paper tape readers. They are linked to the assembler via the following routes:

<table>
<thead>
<tr>
<th>Route</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(only three may be connected, but they may be switched to any of the four routes)</td>
</tr>
<tr>
<td>1</td>
<td>Paper tape readers</td>
</tr>
<tr>
<td>2</td>
<td>Standard cards</td>
</tr>
<tr>
<td>3</td>
<td>Non-standard cards</td>
</tr>
<tr>
<td>4</td>
<td>Standard cards</td>
</tr>
<tr>
<td>5</td>
<td>Non-standard cards</td>
</tr>
</tbody>
</table>

11.2 COMPUTER CONTROL

Input is initiated by an instruction from the coordinator. It is continued under assembler control until a whole card or block of tape has been read.

11.3 CHECKS

Each character read from paper tape punched in the standard code, is accompanied by a 'character parity bit'. Parity is checked by the assembler on receipt of the character. If the check fails, the doubtful block flip-flop in the assembler is automatically set and reading halted.

The parity of characters read from punched cards is not checked. The assembler however counts the number of columns read on each card to ensure that a card end signal is not generated before 80 columns have been read. The doubtful block flip-flop will be set if a card is rejected by an I.O.T. card reader due to misalignment of holes.
11.4 Storage of Information

Characters are stored in consecutive long compartments starting at the address which is held in compartment 648 in the store (where R defines the route and channel (see section 9)).

Characters are stored in octet form (see Appendix A), five characters to each long word. Hence 16 long compartments are required to hold the information for one card. On paper tape, enough long compartments to hold the largest expected block are required.

11.5 Non-Standard Paper Tape

Non-standard code 5-, 6-, or 7-track tape may be read by setting a key on the reader. This causes the Channel Warning Lamp to be lit on the assembler and the operators' control desk.

An input instruction will cause five characters only to be read (see section 10.1.3) and parity checking is suppressed.

For non-standard code 7-track tape, the information is placed in the character octet in store as follows:

Tracks 1 to 4   Bits 1 to 4 (basic - as standard)
Tracks 6 and 7   Bits 5 and 6 (control)
Track 5         Bit 8

If track 5 is used to hold parity hole, checking must be done by programme.

When reading 5- or 6-track tape the interpretation is as above, but the positions not covered by the tape (tracks 6 and 7, and track 7 respectively) will always appear to have holes, and the corresponding bits will be set at 1.

11.6 Non-Standard Cards

Cards may be read in a non-standard way on routes 5 and 7.

Information from each pair of adjacent columns is placed in a long compartment of the store.

Information from the first (odd-numbered) column of the pair is placed in bits 18 to 29 of the compartment with row 10 corresponding to bit 18, and from the second column into bits 30 to 40 and 32, with row 10 corresponding to bit 30. 31 is not used.

No card should have more than 40 holes in any row. If this figure is exceeded the card will be weakened and may not be read successfully.
Section 12

12. MAGNETIC TAPE EQUIPMENT

This section is concerned with the magnetic tape system in general and with operational features of the Ampex equipment used.

12.1 Magnetic Tape

12.1.1 Physical Characteristics

Magnetic tape is made of a plastic material coated on one side with a ferrous compound.

Tape is \( \frac{3}{4} \)" wide and is held on reels, usually with 3,600 ft of tape per reel.

Fig. 3 shows the tape configuration. Each tape has two reflective markers, positioned near the ends as shown. These markers, detected photo-electrically, are used to indicate the beginning of a file and to give advance 'warning of end'. The metallic trailer indicates the end of the tape. As shown in the diagram, the metallic trailer is attached permanently to the equipment and coupled to the tape.

For practical purposes the data on the tape may be visualised as a pattern of magnetised spots analogous to the pattern of holes on paper tape. Each character is represented in eight bit-positions across the tape; the eight parallel rows of positions along the tape are known as 'tracks'.

12.1.2 Data Representation

The eight bit-positions used for each character comprise a character sextet (see Appendix A.3 for the full code), a parity bit, and a clock bit, which is automatically written with each character. When the tape is being read, the clock bit arrives at the assembler just after the character and parity bits. It is used to stop on the character counter and to initiate checking of the character.

Characters may be written on the tape in either low or high density mode, giving packing densities of 375 or 600 characters to the inch, the high density mode of operation requiring modified tape equipment.

Data is written into blocks of up to 5,000 characters, a gap known as the Interblock gap being left between successive blocks. The interblock gap is 1.27 inches for low density information and about 0.7 inches for high density information.

The last word of every block must be a special block end word consisting of character sextets as shown below:

```
BB  BB  BB
```

The three sextets in the middle of the word are reserved for the block track sum (see section 13.6.2).
12.1.3 Tape Storage and Handling

If proper care is not taken, magnetic tapes may easily be damaged - particularly by dust or grit particles, which can also cause damage to reading and writing equipment. It is therefore necessary to store reels of tape in special plastic containers. Tapes should always be kept in their containers when not in use.

At the beginning of each tape there is a 10 ft. leader. This is the only part of the tape that should be handled. Care must be taken to avoid pressing on the flanges of the spools, as this can easily buckle the tape and render it unusable.

12.2 Ampex Magnetic Tape Decks

12.2.1 Types of Deck

There are two Ampex tape transport systems in use with LEO III/326. The tape decks and their operating speeds are shown in the table below. In the notes which follow, differences in operation between low density and high density modes are indicated where appropriate. This does not however constitute a complete specification of differences between the two modes.

<table>
<thead>
<tr>
<th></th>
<th>Tape Speed Read/Write ins./sec.</th>
<th>Rewind Speed ins./sec.</th>
<th>Packing Density characters/inch</th>
<th>Max. Transfer Rate, ch/sec.</th>
<th>Timing Overhead chs/block (Low Density)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Low Density</td>
<td>High Density</td>
<td>Low Density</td>
<td>High Density</td>
<td></td>
</tr>
<tr>
<td>TM2</td>
<td>120</td>
<td>240</td>
<td>375</td>
<td>800</td>
<td>45000</td>
</tr>
<tr>
<td>PR300</td>
<td>120</td>
<td>180</td>
<td>375</td>
<td>-</td>
<td>45000</td>
</tr>
<tr>
<td>TM4</td>
<td>75</td>
<td>150</td>
<td>375</td>
<td>800</td>
<td>28125</td>
</tr>
<tr>
<td>TM400M</td>
<td>75</td>
<td>225</td>
<td>375</td>
<td>-</td>
<td>45000</td>
</tr>
</tbody>
</table>

12.2.2 Method of Reading and Writing

Data is recorded by moving the tape over a 'write' head, through which electronic pulses representing the data are passed to induce magnetism on the tape. Data is read by moving the magnetised tape over a 'read' head, thus reproducing the data on the tape in the form of electronic pulses in the read head.

The checking of data is described in section 13.6.
12.2.3 Tape Drive

The tape is driven over the read/write heads by capstan/pinch roller mechanisms; the movement of the actual tape reels is controlled in accordance with the amount of tape in special 'buffers' at any instant (the mechanical buffers are used in this connection on the TM4/TH400M decks). These buffers (see Figs. 1 and 2) serve to avoid undue strain on the tape when stopping and starting.

During normal (forward) running, the tape is wound from the upper to the lower (take-up) spool until the data has been written or read as required; the tape is then rewound onto its original spool. Figs. 1 and 2 show the main features of the decks and the path of the tape through the mechanism.

12.2.4 Loading and Unloading

As shown in Fig. 3 a metallic leader is attached permanently to the take-up spool. At the end of the leader there is a tag which fits into a socket at the beginning of any magnetic tape. When a new tape has been coupled to the leader, the deck is 'loaded' by pressing the LOAD key on the control panel; this causes the tape to be run on until the 'beginning of file' reflective marker is sensed. The deck must then be switched to COMPUTER READ or COMPUTER WRITE to make it ready for operation under computer control. Reading or writing will not commence until the 'beginning of file' marker has passed the read/write heads; hence there is an initial delay before transfer of data begins.

<table>
<thead>
<tr>
<th>Initial Delay Times</th>
<th>Read</th>
<th>Write</th>
</tr>
</thead>
<tbody>
<tr>
<td>TM2/FR300</td>
<td>18-30 µs</td>
<td>40-66 µs</td>
</tr>
<tr>
<td>TM4/TH400M</td>
<td>29-48 µs</td>
<td>64-104 µs</td>
</tr>
</tbody>
</table>

When the deck is being used for writing, the 'advance warning of end' marker indicates to the Master Programme that an End of Reel sentinel is to be written.

The tape is rewound by operation of the manual rewind control or by programme instruction, until the leader is sensed. The tape is then in the 'unload' position, and the deck is set to MANUAL ready for the operator to change reels.

12.2.5 File Manipulation

A computer code Read or Write instruction causes the tape to begin moving.

Amendment No. A
December 1964
Section 12.2 (Cont'd)

Reading or writing does not commence until the tape is moving at the full speed for this mode of operation (i.e., while the inter-block gap is passing the heads). When the first 8E character of the block terminal word has been sensed or recorded the final checks for the block are made, and the remainder of the block terminal word is read or written. The tape then comes to rest awaiting the next instruction.

If the deck is working in the normal mode - as when reading, writing, or stepping forward - the interblock gap will be directly under the read head when the tape stops; but if the tape is being moved backwards preparatory to re-reading or overwriting a block, each block end character will be encountered at the beginning of a block, and hence, when the tape stops, the read head will be positioned just inside a block. Reading or writing of a given block can only take place, however, if the read head is first positioned over the preceding interblock gap. Thus to align on a given block it is necessary first to step back one extra block and then to step on one block (see Fig. 4).

To allow the tape to be quickly positioned for reading from any required block without repeatedly testing block identities, use is made of alignment marks. An alignment mark consists of at least four 2/15 characters in consecutive positions in any word of the block. The computer code actions 'Run On' and 'Run Back' cause the tape to be moved forwards or backwards respectively until an alignment mark is sensed. Note that after 'Run Back' the tape will be misaligned as described above.

12.2.6 Effective Transfer Rates

Because of the presence of interblock gaps on magnetic tape, the overall rate at which characters are read or written is lower than the theoretical maximum calculated from the speed of tape movement and the character packing density.

The effective rate varies according to block length, and is evaluated as:

\[ R = \frac{s}{n+g} \]

where:

- \( s \) = theoretical maximum transfer rate (chs/sec.)
- \( n \) = block size in characters
- \( g \) = timing overhead due to interblock gaps (chs/block)

Fig. 5 shows in graphical form the effective transfer rates for varying block lengths.
Arrows show normal direction of tape.

- Real Brake Release
- Leader Clamp
- Upper Sensing Post
- Photo-Electric Sensing Device
- Vacuum Reservoir
- Tape Follower Arm
- Vacuum Reservoir
- Read/Record Heads
- Lower Sensing Post
- Reel Brake Release
**FIG. 3** View of Back of Tape

- **Forward Direction of Tape**
- **Tape Coupling**
- **Metallic Leader** (Permanently Attached To Take-up Reel)
- **"Advance End of Tape" Reflecting Marker**
- **"Beginning of File" Reflecting Marker**
- **Metallic Trailer**

**FIG. 4** File Manipulation for Re-reading a Block

- **A.** Position of read head after reading Block n.
  - Block n+1
  - BE Block n
  - BE Block n-1
  - BE Block n-2
  - Direction of Tape Movement.

- **B.** Position after stepping back two blocks.
  - Block n+1
  - BE Block n
  - BE Block n-1
  - BE Block n-2
  - Reverse direction of Tape Movement.

- **C.** Position after stepping forward one block.
  - BE Block n
  - BE Block n-1
  - BE Block n-2
  - Direction of Tape Movement.
Effective Information Transfer Rate of Magnetic Tape Equipment.

Mean Transfer Rate,
Thousand characters/second (Low-density mode)

Theoretical Peak Speed 45 K ch/sec.

Theoretical Peak Speed 28.125 K ch/sec.

Number of Characters/Block
13. **MAGNETIC TAPE ASSEMBLERS**

This section describes special features of the magnetic tape assembler, in particular the special checking facilities. Features common to all assemblers are dealt with in section 8.

13.1 Medium and Routes

This assembler controls input and output on magnetic tape. Decks can be switched to any of routes 0 to 7 on this assembler. Up to 10 decks can be connected to the assembler but only 8 may be switched to routes at the same time. A mixture of TM2/FR300 and TM4/TM400H tape decks may not be linked to the same assembler.

13.2 Computer Control

Input or output is initiated by an instruction from the coordinator and continues under assembler control until one block has been read or written.

The following functions can also be performed under assembler control after being initiated by the coordinator:

(a) Run back to run on to next alignment mark (note that when running back to a mark the mark may not be recognized if it is within 80 words of a block end word).

(b) Step back one block.

(c) Rewind to position ready to read first block on tape. (This instruction only engages the route, not the channel. The assembler is free to control other functions on other equipment).

(d) Rewind to 'unload' position and close route, setting it to manual.

(e) Read first word of block and step on to end of block.
13.3 Storage of Information

Characters are read from or written into consecutive words, 5 octets to a word, starting at the most significant octet of the word whose address is contained in compartment 64+R.

13.4 Lockout (see section 16.2)

The tag register of the magnetic tape assembler is set equal to the contents of bits 17 to 20 of compartment 64+R by its input or output instructions.

Lockout will occur under the conditions stated in section 16.2 and if this happens the channel is closed.

13.5 Warning of End

This is given when the 'advance end of tape' marker is sensed photo-electrically. This marker is 10 feet from the end of the tape.

13.6 Checks

13.6.1 Simple Parity Checks

(1) On Output

Checking the accuracy of information recorded on the tape is effected by reading back each character after it has been written, and making a character parity check. If the check fails — via: if the character is found to contain an even number of bits — the 'doubtful block' condition is set in the assembler.

(ii) On Input

The parity of each character is checked as it is read, and if found to be incorrect a 'doubtful block' condition is set, and a '1' inserted in bit 8 of the octet.
13.6.2 Track Sum

In addition to character checking, a further check is performed on each block. When information is being written on the tape the assembler forms a six-bit sum of all the characters up to and including the first block and character. 'End-around carry' is used in the sum, which means that any carry derived from the sixth bit is added with bit 1 significance into the sum. To reduce the chance of compensating errors due to drop-out on adjacent tracks, tracks 1 to 6 have during the summing process the significance 1, 2, 5, 2, 4 and 6 respectively. The reflection of the sum is written in the second and third character positions of the terminal word, digits 3 and 4 in character position 3 and digits 1, 2, 5 and 6 in character position 2 as follows:

<table>
<thead>
<tr>
<th>111</th>
<th>111</th>
<th>...XX</th>
<th>...XX</th>
<th>...XX</th>
<th>...XX</th>
<th>...XX</th>
<th>...XX</th>
</tr>
</thead>
<tbody>
<tr>
<td>65</td>
<td>21</td>
<td>43</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

BE char. Track sum BE char.

When reading, the characters are counted in a similar manner, with the tracks having the same significance as when writing, and when all the characters are counted up to and including the track sum, the reflection of this sum is placed in the fourth character position of the terminal word and is sent to store. This character should be all '0's'; the presence of any '1's' indicates a track sum error and generates a 'doubtful block' condition.

13.6.3 The Two-Level Discrimination System

The purpose of this system is to apply a more rigorous check on output to ensure that no spurious information is recorded, and that the correct information is recorded at a high intensity so that there is a greater assurance of its retrieval during subsequent input.

The system operates as follows:
Section 13.6.3

(i) Output

Each character written is read back from the tape and fed in parallel into two voltage discriminators set at different voltage levels. The high level discriminator only accepts signals above the normal acceptable level, whereas the low level discriminator accepts strong interference or 'noise' which might be interpreted as weak signals. Output from the discriminators is set up on two 7-bit character registers. Three checks are then performed:

(a) A high level parity check which tests whether or not a bit has been lost or recorded badly.
(b) A low level parity check which tests whether a spurious bit has been gained.
(c) A comparison between the contents of the two character registers to ensure they are identical.

All of these tests must succeed for the character to be accepted. If any test fails a 'doubtful block' condition is set.

(ii) Input

A check is made on reading accuracy by submitting each character to a parity check with two-level discrimination. If the parity of the character register associated with the high level discriminator is correct, information from this register is accepted. Otherwise information is accepted from the other register if its parity is correct. If the parity of both registers is incorrect, a '1' is inserted into bit 8 of the octet and a 'doubtful block' condition is set.

The check applied on output is made the more rigorous in order to guarantee accuracy whereas the check made on input favours successful retrieval of the information.
14. OUTPUT EQUIPMENT

14.1 Teletype DS23110 Paper Tape Punch

14.1.1 Linkage with the Computer

The paper tape punch can be allocated to routes on either the general purpose output assembler or the paper tape punch output assembler.

14.1.2 Type of Tape

The punch is normally arranged to punch standard $\frac{3}{4}$" wide, 7-track paper tape. It is also readily adjustable for punching $\frac{5}{8}$" wide 5-track tape or 1" wide 8-track tape.

14.1.3 Character Code

The standard 7-track tape is punched with a character sextet and parity bit according to the British Standard Code. Each row across the tape represents one character.

The six information bits punched on the tape are made up of a basic quartet (tracks 1 to 4 inclusive) and a control duet (tracks 5 and 7). The last quartet code is consistent with that used with the printers and punched cards and is detailed in Appendix A.1. Track 5 contains an odd-parity bit. (Note the odd-parity bit is formed by the assembler, for checking after the tape has been punched).

14.1.4 Speed

The tape is punched at a maximum rate of 110 rows per second.

14.1.5 Control and Operation

The tape reel is equipped with a 'low tape' warning mechanism which measures the amount of tape left on the reel and initiates a 'warning of end' alarm when 2 feet of tape is left. If the tape breaks between the reel and the punching head the route is closed.

A Run Out key is provided on the punch for the continuous output of blank tape for unloading purposes.
14.2 ICT (Hollerith) 582 Card Punch

14.2.1 Linkage with the Computer

The card punch is allocated to routes on the General Purpose Output Assembler.

14.2.2 Type of Card

The standard 80-column, 12-row ICT or IBM cards are used. The ICT and IBM cards are identical in all aspects apart from thickness. The cards are interchangeable provided that when changing from one card to the other, the equipment is reset to suit the appropriate card thickness. The two types of card cannot be mixed.

The input hopper and output stacker will each hold approximately 750 cards.

14.2.3 Character Codes

The standard punching code is as shown in Appendix 1.2, and is as for punched card input (see section 10.1.3).

Non-standard codes (e.g. binary) may also be used, in which case the punch punches holes in whatever pattern it is given by the assembler, as described in section 15.

14.2.4 Speed

The card punch has a maximum punching speed of 100 cards per minute.

14.2.5 Control and Operation

The cards are punched by rows commencing at the top of the card with row value 10. Each card can contain up to 80 characters in standard code, or 40 columns (20 words) in non-standard form.

The output cycle is divided into 14 stages of which 12 control the transfer and punching of information. Each of the 12 information stages corresponds to one row of the card. The remaining two stages control movement of the card.
An 'end of medium' alarm is provided on the input hopper. When the last but one card is being punched, the last card moves out of the hopper ready to be fed into the punch, thus leaving the hopper empty. The hopper empty condition closes the route and lights the 'card out' alarm. When the next output instruction tests that the specified route is available, the test will fail and the operator will be notified. A similar procedure occurs when the output stacker is full.
14.3 **Anelex 56 - 160 Line Printer**

**Note:** This printer is supplied on III/1 only.

14.3.1 **Linkage with the Computer**

The Anelex (56-160) printer may be allocated to routes on the Special Anelex output assembler.

14.3.2 **Stationery**

Stationery sets used by the Anelex must be between .004 inches and .02 inches in thickness, and must be greater than 20 lbs. per ream in weight. When several copies of the printed output are required, the stationery set must be made up by the supplier, as facilities are not provided for feeding separate sheets or for inter-leaving with carbon paper. The maximum thickness quoted above allows 6 copies on 6 or 9 lbs. paper. Where only one copy is required, it may be necessary to introduce a backing sheet to achieve the minimum thickness.

Only continuous stationery may be used; the printer can handle batches up to 10 inches in depth.

The maximum paper width which can be accommodated is 20 inches and the minimum is 7 inches.

The accurate alignment of preprinted stationery is facilitated if a guide mark is printed on the form. The guide mark for each form should be printed 6 inches above the middle horizontal stroke of an E on the head of form line. The head of form line is the first line to be printed, irrespective of where this line is on the form. The guide mark is a horizontal line about ½ inch long placed between ½ and 1 inch to the right of a line passing through the centres of the left-hand sprocket holes. The stationery is inserted in the printer so that the guide mark coincides with the alignment marks on the upper paper tractors (figs. 1(a), 1(b)).
14.3.3 Control of Spacing between Lines

Control of the distance by which the stationery is advanced between each printed line is effected by the format control tape mechanism. The format control tape is a loop of plastic tape, the movement of which is geared to the paper tractors so that when the tape moves on one sprocket position (1/10") the stationery is advanced 1/6th inch (the normal single space). Movement of the format control tape by a given number of sprocket positions causes the stationery to be advanced an equal number of spaces.

When a 'print' instruction is given, directions on the spacing required are included with the information to be printed (see section 15.2.3). The directions specify movement of the format control tape until a hole is sensed in a designated track. Holes will have been punched in the tape in positions which will give the required amount of movement. After the stationery has been advanced an equivalent number of spaces, the line is printed.

The format control tape has 8 tracks in which holes are punched in such a way as to ensure that any combination of spacing required on the print format are available. Track number 8 of the format control tape by convention contains one hole only, used to mark the 'head of form' position for the layouts contained in track numbers 1 to 7.
An example of a format control tape design for a particular print layout is given below:

Example of format control tape design

<table>
<thead>
<tr>
<th>Tracks</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Printing is to take place on a 20-line form using the following arrangements</td>
</tr>
<tr>
<td>2</td>
<td>Print on line 3</td>
</tr>
<tr>
<td>3</td>
<td>Print on line 4 (see note below)</td>
</tr>
<tr>
<td>4</td>
<td>Print on line 6 and/or line 7</td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Print on line 12 and/or line 14</td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Print on line 16</td>
</tr>
<tr>
<td>17</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

Rows

Align to head of next form

Note: To move from line 3 to line 4 no hole is needed. Line space movement may be specified without reference to a particular track

The complete form length must be a multiple of 1/6th inch, up to a maximum of 22 inches. The minimum size of the format control tape loop is 6.6 inches which is equivalent to a form length of 11 inches. To cater for forms less than 11 inches long, the layout must be repeated on the format control tape an integral number of times so that the minimum length of tape is exceeded. For example, a form 7 inches long, which requires only 4.2 inches of tape, must be repeated so as to require 6.4 inches. Two forms will then be printed for each cycle of the tape loop.
Section 14.3.4

For initial synchronisation of stationary and format control
tape, the Format Track Selector on the printer control panel is switched
to track 8 and the Align Format button operated. This causes the tape
to be positioned at a hole representing the ‘head of form’ position,
and the stationary can then be inserted and aligned correctly (see
section 14.3.2).

14.3.4 Character Code

120 characters per line may be printed, spaced at 10 characters per
inch. The character code is shown in Appendix A.4. Characters ‘A’ and
‘B’ cannot be printed under computer control.

14.3.5 Speed

The print roll speed can be set at either 1200 r.p.m. or 800 r.p.m.
by means of the Print Roll Speed Selection Key.

The time taken to print one line is approximately \( 89 + 6.8x \)
milliseconds at 800 r.p.m. or \( 74 + 6.8x \) milliseconds at 1200 r.p.m.,
where \( x \) = number of line spaces required before the line is printed.

Effective maximum printing speeds (with 1 line spacing) are:

- 800 r.p.m. - 570 lines per minute (see also Appendix F)
- 1200 r.p.m. - 750 lines per minute

14.3.6 Control and Operation

Characters are engraved on the face of the print roll in rows,
each row containing a character identical for each of the 120 print
positions. Thus all similar characters in a line must be printed
simultaneously. Signals are received from the assembler indicating
each type of character in succession and set up on the printer’s
matrix store, from which the print coils controlling the hammers are
actuated. The signals are checked for parity between the assembler
and matrix store, and the matrix store and the print coils. In the
event of parity failure the channel is closed and an assembler alarm
lit on the operators’ control desk and on the assembler itself.
When it is desired to print on a narrow field (less than 12 inches wide with printing field central, or 9½ inches with maximum off-set) some of the character positions on the print roll will not be utilised. In this case it is necessary for the printing field, as formed in the store, to be preceded by an appropriate number of spaces (see section 14.3.2).

When there is less than 22 inches of paper remaining a switch is operated which generates an 'advance warning of end of paper' alarm. This alarm indicates that the stationery must be renewed at the end of the current form.

Should the paper tear between the advance warning switch and the printing station a 'paper out' alarm is given which immediately closes the channel and generates a 'selected route' alarm.

The average time required for reloading the machine with a fresh batch of stationery is less than 1 minute. The average time required for changing to new stationery (as distinct from renewing stationery) is 3 minutes.
Section 14.4

14.4 Analex Series 4-1000 - SP Line Printer (also known as Analex 4-1000 printer)

14.4.1 Interfacing with the Computer

This printer may be allocated to routes 0 and 1 on the General Purpose or routes 0 to 3 on the Special Analex output assemblers.

14.4.2 Stationery

The stationery used on this machine must conform to the requirements stated in section 14.3.2 with the exception that the minimum paper width is 4 inches.

The alignment guide marks however are printed 1 1/8 inches below the middle stroke of an 'E' on the head of form line and are lined up with a scribed mark on the lower paper tractors in this case.

A method has been developed to facilitate the lateral setting of the paper tractors on this printer, for any particular stationery. A rectangular metal plate (fig. 1 (b)) is used, as shown in fig. 1 (a). The dimensions of the plate determine the distance of the tractors from the end of the rods on which they are located.

Only one key is required for each type of stationery.
14.4.3 Control of Spacing between Lines
As in section 14.3.3.

14.4.4 Character Code
Up to 160 characters per line may be printed, spaced at 10 characters to the inch. The character code is shown in Appendix A.4. Characters 'w' and 'w' can be printed under computer control only when the printer is linked to the general purpose output assembler.

14.4.5 Speed
The print roll speed can be set at either 1,000 or 1,500 r.p.m. by means of the Print Roll Selection Key. The time taken to print one line is approximately $72 + 8x$ milliseconds at 1,000 r.p.m. or $52 + 8x$ milliseconds at 1,500 r.p.m. Where $x$ = the number of line spaces required before the line is printed.

Effective maximum printing speeds with single line spacing are:
- 1,000 r.p.m. - 750 lines per minute (see also Appendix F)
- 1,500 r.p.m. - 1,000 lines per minute

Note: There is an electronic limitation on the printer which prevents printing at speeds in excess of 1,000 lines per minute.

14.4.6 Control and Operation
Characters are engraved on the face of the print roll in rows, each row containing a character identical for each of the 160 print positions. Otherwise as described in section 14.3.6.
14.5 **IBM Output Writer**

The typewriter provides a low speed printed output from the computer, the principle application being to print out the operating log. It may also be used as an output device for jobs with a small volume of printout.

14.5.1 **Linkage with the Computer**

The typewriter can be allocated to routes 2 to 5 on either the General Purpose output assembler or, the Paper Tape output assembler. (Route 6 to 9 on paper tape assembler on III/1).

14.5.2 **Stationery**

The stationery used must be of the continuous fan-fold type. Maximum paper width is 1 inch less than the carriage length. In the case of LEO III/1 the carriage is 17 inches long.

14.5.3 **Control of Spacing between Lines**

Normally inter-line spacing is controlled by the setting of a lever on the typewriter. This lever may be set to give 1, 2 or 3 line feeds as necessary. If a Formliner is fitted to the typewriter, only one line feed is executed per computer command, but it is fitted with a single track paper tape control loop which causes paper to be fed until a hole is detected in the tape. This movement is initiated by a block end character or by pressing the 'Align Format' key.

14.5.4 **Character Code**

Printing is spaced at 10 characters to the inch and 6 lines to the inch. At the present time on III/2 to III/5, only the lower case symbols can be printed by the computer using the standard LEO III code. However, the following upper case characters may be called using the following code:

- L - 3/11
- A - O/13
- 0 - 0/11
- Z - 3/12

The other upper case characters cannot be obtained. On III/1 and III/6 onwards however the characters available to the computer are as specified in Appendix A.5.
14.5.5 **Speed**

During normal operation when lower case characters only are being printed the speed is 10 to 11 characters per second. In the most extreme case, where upper and lower case characters are being printed alternately the speed falls to 8 characters per second. Tabulation takes 390 milliseconds to shift 4 inches. Carriage return takes 350 milliseconds to shift 4 inches. Throw takes 1 second to throw 4 inches. Ribbon shift takes 30 milliseconds.

14.5.6 **Control and Operation**

The typewriter is fitted with solenoids to permit remote operation, but it may be operated manually if so desired. One typewriter will always be fitted on the operators' console, and any additional typewriters will be mounted on separate desks.

When connected to the General Purpose output assembler, the typewriter will print one character per output instruction, this character being specified by the most significant sextet of the output annex word, i.e. bits 33 to 36. All other bits in the word must be zero.

When connected to the Paper Tape output assembler, it will print 5 characters per output instruction, the characters being specified by the five sextets in word 164' of the store, a fixed output annex location.

When a normalizer is fitted an alarm will be raised if the paper becomes exhausted.

The typewriter keyboard is supplied as shown in Appendix C.
15. **OUTPUT ASSEMBLERS**

This section describes special features of the Paper Tape output assemblers, the Special Anelex printer assembler, the General Purpose output assembler, and the standard Anelex printer assembler. Features common to all assemblers are dealt with in section 8.

15.1 **Paper Tape Output Assembler**

15.1.1 **Media and Routes**

This assembler may be used to control output on paper tape punches, typewriters or digital displays. Output on paper tape punches or typewriter may alternatively be controlled by the General Purpose output assembler if desired (see section 15.3). The equipments are linked to the assembler via the following routes:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>Paper tape punches and typewriters</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>III/7 onwards</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>III/1 only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>only</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Digital displays (on LE0 III/1 only)</td>
<td></td>
</tr>
</tbody>
</table>

15.1.2 **Form and Treatment of Information**

The assembler takes characters from compartment 154' of the store (irrespective of the contents of compartment 64+R). They are in sextet form (see Appendix A).

On the General Purpose Output Assembler however, the assembler takes the least significant sextets from the octets held in compartment 154' of the store (irrespective of the contents of compartment 64+R).
On LEO III/1 only, each output instruction sends all five characters — starting with the most significant — to the assembler, but if one of these is a block end character, the less significant characters — if any — will not be sent. These remaining characters must, however, have an even number of bits, otherwise parity failure will occur. It is recommended that any characters in the word which are less significant than a block end character should be space characters.

On LEO III/2 onwards, there is no restriction on characters subsequent to block end, since all five characters are checked by the assembler. Characters following the block end are not sent to the peripheral equipment.

On LEO III/1 only, if output is to the digital display, the control quarter of each character is ignored, and the value displayed is that of the basic quarter, if this is in the range 0 to 15. A quarter or 12, 13, 14 or 15 will cause a blank to be displayed.

15.3 Lockouts

Lockouts do not occur on this assembler.
15.2 Special Analex Printer Assembler

15.2.1 Medium and Routes

This assembler may be used to control output to Analex printers, with up to 160 characters per line. Output on the printers may alternatively be controlled by the General Purpose Output Assembler (see section 15.3), or the Standard Analex Assembler (see section 15.4). The printers are linked to the assembler via the following routes:

\[
\begin{align*}
0 & \quad \text{Intermediate printer} \\
1 & \quad \text{Analex printers} \\
2 & \quad \text{Analex printers} \\
3 & \quad \text{Analex printers}
\end{align*}
\]

Routes 4 to 7 are not used.

15.2.2 Form and Treatment of Information

The assembler takes long words from successive compartments starting with the compartment whose address is held in compartment 64+2. The first long word is interpreted as a function word. According to the value of this word (see section 15.2.3) the assembler sends a signal to the printer specifying which track of the format control tape is to be used to control paper movement. While this is taking place the assembler takes the characters from the words following the function word, starting at the most significant character of the first word, and sets them up in its buffer store. When a line end or block end character is detected, all characters preceding it are printed.

When a 'line end' character is detected, the assembler takes the next word (after that which held the line end character) as the function word for the next line. The process is then repeated as above, and so on until a block end character is reached. The last line is then printed and this completes the section.
15.2.3 Function Word

The function word in the main store of the computer consists of five octets, of which only the basic quartets of the three least significant octets are used.

The meanings of the various values of the function word are as follows:

<table>
<thead>
<tr>
<th>5th octet</th>
<th>4th octet</th>
<th>3rd octet</th>
<th>2nd octet</th>
<th>1st octet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Bit 1 of function word = 1: Paper movement to be controlled by Track 1
Bit 2 of function word = 1: Paper movement to be controlled by Track 2
Bit 3 of function word = 1: Paper movement to be controlled by Track 3
Bit 4 of function word = 1: Paper movement to be controlled by Track 4
Bit 5 of function word = 1: Paper movement to be controlled by Track 5
Bit 10 of function word = 1: Paper movement to be controlled by Track 6
Bit 11 of function word = 1: Paper movement to be controlled by Track 7
Bit 12 of function word = 1: Paper movement to be controlled by Track 8
Bit 17 of function word = 1: Zero line feed (i.e. no paper movement)

If none of these bits is specified, one line feed (1/6th inch paper movement) will be given.

15.2.4 Lockouts (see section 17.2)

Lockouts do not occur on this assembler.

15.2.5 Warning of End

The 'warning of end' condition is set when there is less than 22 inches of stationary remaining.
15.3 General Purpose Output Assembler

15.3.1 Media and Routes

The general purpose output assembler may be used to control output on all media except magnetic tape and digital display. Output on paper tape punches or typewriters may alternatively be controlled by the paper tape punch assembler (see section 15.1). Output on Anelex printer may be controlled by the special or standard Anelex printer assemblers (see sections 15.2 and 15.4).

The routes used are as follows:

0 Anelex printers - not on III/1, III/2 or III/3
1 Typewriters, card punches and paper tape punches (subject to individual maxima of 2 TMK's, 4 CP's
4 and 2 PTP's. This restriction does not apply to
5 III/9 onwards)
6 Not used

15.3.2 Form and Treatment of Information for Printing

The assembler first takes the long word whose address is contained in compartment 64+H. This is interpreted as a function word, and according to its value, the assembler sends a signal to the printer to control paper movement. The meaning of the values of the function word are as for the special Anelex assembler (section 15.2.3) except that the zero line feed (bit 17) facility is not available, and that bits 37 to 40 inclusive have a special meaning detailed below:

Bit 17 of function word = 1: print space characters in position 1 - 40

Bit 38 of function word = 1: print space characters in position 41 - 80

Bit 39 of function word = 1: print space characters in position 81 - 120

Bit 40 of function word = 1: print space characters in position 121 - 160
Section 15.3.3

When paper movement has ceased, the assembler selects, character by character, the information in the words following the function word. (This information must have been set up by programs). Each character in turn is set up in the assembler buffer register and printed.

Information for printing is set up in the store using action 29/0/1 (see section 16.4). This action treats compartments N' (whose address is contained in compartment 64+8) to N+51D' as a matrix 160 bits by 64 bits thus:

<table>
<thead>
<tr>
<th>Bit No.</th>
<th>Character Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>160-121</td>
<td>0/1</td>
</tr>
<tr>
<td>120-81</td>
<td>0/2</td>
</tr>
<tr>
<td>80-41</td>
<td>0/3</td>
</tr>
<tr>
<td>40-1</td>
<td></td>
</tr>
</tbody>
</table>

Each 'column' of bits controls one print position - bits in column 1 control print position 1, and so on. The character printed in each print position is determined by the position of the bit which is set at 1 in the corresponding column of the matrix. (Only one bit in a column may be set at 1; if no bit is set, a space is printed). The character corresponding to each position in a column is indicated on the above diagram by its control and basic value (see Appendix A).

15.3.3 Form and Treatment of Information for Cards

Information for output to the card punch is taken from a matrix in the store. The matrix for one card is set up by action 29/0/0. Each pair of words in the matrix is used to control the punching of one row of the card. The first pair of words, containing at the location held in compartment 64+8, is ignored. Successive pairs of words are then taken from the store, set up in the assembler buffer register. From each pair of words taken from the store, signals are sent to the punch and the row controlled by the words is punched before the next pair of words is selected.
15.3.4 Form and Treatment of Information for the Paper Tape Punch and Typewriter

One character is output for each output instruction. This is the most significant octet of the word whose address is contained in compartment 64+R. It must be held in the code defined in Appendix A, and the most significant two bits of the control quartet must be zero. The remaining octets of the word must contain an even number of bits. It is recommended that they are space (0/0) characters.

Special characters such as EB, E6, etc. are not recognized as such by the assembler but are output in the normal way. In the case of the typewriter they have special meanings which are recognized by the equipment, (see Appendix A.5).

15.3.5 Lockouts (see section 17.2)

Lockouts do not occur on this assembler.

15.3.6 Warning of End

The 'warning of end' condition is set under the following circumstances:

- Teletype printer - when 22 inches of stationery remain

Note that the General Purpose Output Assemblers on III/7 onwards have optional packages which may be omitted if it is not required to connect the full range of peripheral equipment to the assembler.
15.4 Standard Analex Printer Assembler

This assembler offers the same facilities as the general purpose output assembler offered for Analex printer (see section 15.3.2).

The routes used are as follows:

0  Analex printers
1

Routes 2 to 7 are not used.

This is fitted on III/3 and III/6 only, on later machines a General Purpose Output Assembler is used with redundant packages omitted.