

THE ENGLISH ELECTRIC CO., LTD.

NELSON RESEARCH LABORATORIES

STAFFORD
MATHEMATICS DEPARTMENT.

Telephone:—Stafford 700.

Report No. NS 141

Reference

Order No.

59.

Front Sheet.

Data Sheets 1-28

Appendices I-II, Sheets 29-34.

DEUCE News No. 36, April, 1959.
Multiplier and Divider Techniques.

Report by
J. Boothroyd.

SUMMARY.

This report contains a description of various techniques which exploit the logical structure of the DEUCE multiplier and divider for purposes other than straightforward multiplication and division. Such techniques have been employed to advantage in DEUCE Subroutines R21T/1, R23T/2, R24T, R28T, R29T, P22T, the Alphacode Mk. II Compiler (ZC16/1 and ZC16/2) and in DEUCE Programme DGO1T. Those interested should refer to the appropriate reports for further details.

Brief descriptions of the operation of the multiplier and divider are given in Appendices to the report.

J. Boothroyd

MATHEMATICS DEPARTMENT.

HEF

DEUCE News No. 36, April, 1959.

Multiplier and Divider Techniques.

1. INTRODUCTION

The unorthodox uses of the multiplier and divider described in this report are made possible by the availability to the programmer of D16, S21, D21, S22, D22 and D23 throughout the course of operations starting with 0-24 and 1-24. The rules which must be observed are not difficult to learn and the use of these techniques has resulted in fast operation in all routines in which they have been employed. The advantages of parallel programming may be obtained by causing T.C.A. to change the contents of TS16 automatically during multiplication and division and by using DS21 as an automatic shift register.

Changing the contents of TS16 during multiplication makes possible:

- (a) Conversions from binary coded decimal to binary.
- (b) Conversion of card row patterns to binary.
- (c) Re-arrangement of word patterns.
- (d) The rapid counting of digits in a word.

By changing the contents of TS16 during division it is possible to perform the reverse operation of (a).

It is believed that the use of TCA during multiplication was first suggested by B. Munday (then at N.P.L.). So far as is known the detailed coding was first done at N.R.L. The techniques found enthusiastic adherents in M.R. Wetherfield and J. Lucking. Meanwhile J. O'Brien (E.E. Luton) was pursuing his own researches into the misuse of the multiplier. It would not be out of place here to acknowledge the contributions made to these techniques by each of those mentioned.

Those readers who have some knowledge of the operation of the multiplier and divider should not find difficulty in understanding the processes described in the following sections of the report. Others may find it helpful to read Appendix I before Section 3 and Appendix II before Section 4.

2. REFERENCES TO OTHER REPORTS.

Flow diagrams and coding for routines employing these techniques are R21T/1, No. 290, R29T, No. 298, ZC16T/2, No. 471, R23T/2, No. 280, F22T, No. 281, DGO1R, No. 460, R24T No. 267, D19/2, No. 276, R28T, No. 291, ZC16/1, No. 470.

3. OPERATIONS USING THE MULTIPLIER.

3.1 Conversion of Binary Coded Decimal to Binary.

A number $N < 10^8$ expressed in four-bit binary coded decimal can be converted to binary by the following instructions:

NELSON RESEARCH LABORATORIES

Continuation to NS y 141

STAFFORD E. E. CO. LTD.
MATHEMATICS DEPARTMENT.

Sheet No. 2.

$(10^7 \cdot 2^3)$	-	16	
N	-	21_3	
30	-	21_2	
0	-	24	(m.c. m)
$(10^6 \cdot 2^7)$	-	16	(m.c. m+7 or m+8)
$(10^5 \cdot 2^{11})$	-	16	(m.c. m+15 or m+16)
$(10^4 \cdot 2^{15})$	-	16	(m.c. m+23 or m+24)
$(10^3 \cdot 2^{19})$	-	16	(m.c. m+31 or m)
$(10^2 \cdot 2^{23})$	-	16	(m.c. m+7 or m+8)
$(10^1 \cdot 2^{27})$	-	16	(m.c. m+15 or m+16)
$(10^0 \cdot 2^{31})$	-	16	(m.c. m+23 or m+24)
21	-	22	(d, e, o) after MULT.

Result in 21_3 with 21_2 clear.

NOTES: (a) 8 constants are required. The first of these can be stored in any m.c. The others must be stored in the minor cycles stated.

(b) If N is $a_7 \cdot 10^7 + a_6 \cdot 10^6 + \dots + a_0 \cdot 10^0$

the following partial products are available from S21 in the minor cycles shown:

m.c.	m+9	$(a_7 \cdot 10^7) \cdot 2^3$
	m+17	$\sum_{r=6}^7 (a_r \cdot 10^r) \cdot 2^7$
	m+25	$\sum_{r=5}^7 (a_r \cdot 10^r) \cdot 2^{11}$
	m+33	$\sum_{r=4}^7 (a_r \cdot 10^r) \cdot 2^{15}$
	m+41	$\sum_{r=3}^7 (a_r \cdot 10^r) \cdot 2^{19}$
	m+49	$\sum_{r=2}^7 (a_r \cdot 10^r) \cdot 2^{23}$
	m+57	$\sum_{r=1}^7 (a_r \cdot 10^r) \cdot 2^{27}$
	m+65	N $\cdot 2^{31}$

(c) The same conversion can be accomplished using less instructions but at the cost of an increase in the number of stored constants by the use of T.C.A.

The following instructions are used, assuming that D.L.10 and D.L.9 contain the constants:

$$\begin{aligned}
 r=m, m+2, m+4, m+6 & \quad 10_r = 10^7 \cdot 2^3; \quad 9_r = 10^3 \cdot 2^{19} \\
 r=m+8, m+10, m+12, m+14 & \quad 10_r = 10^6 \cdot 2^7; \quad 9_r = 10^2 \cdot 2^{23} \\
 r=m+16, m+18, m+20, m+22 & \quad 10_r = 10^5 \cdot 2^{11}; \quad 9_r = 10^1 \cdot 2^{27} \\
 r=m+24, m+26, m+28, m+30 & \quad 10_r = 10^4 \cdot 2^{15}; \quad 9_r = 10^0 \cdot 2^{31}
 \end{aligned}$$

for a multiplication starting in minor cycle m,

- 30 - 21₂
- 3 - 24
- N - 21₃
- 0 - 24 (m.c. m)
- 9 - 16 (32 m.c.) (m + 32 to m + 63)
- 21 - 22 (d)

A comparison of the two methods is made below and shows that in this application the T.C.A. method is inefficient:

	<u>Programme.</u>	<u>T.C.A.</u>
<u>Storage.</u>	12 Instructions	6 Instructions.
	8 Constants.	32 Constants.
<u>Programme m.c. used during MULT.</u>	7 m.c.	34 m.c.

The multiplicands $10^P \cdot 2^Q$ may be varied, within limits, depending on the range of N and the number of bits in the coded characters. Five contiguous six-bit characters may be contained in one DEUCE word in one of three ways, with the least significant digit of the least significant character in the P₁, P₂ or P₃ positions. For such a number $< 10^5$ the constants required for conversion depend on the P position of the required result and whether the least significant character of N is at P₁, P₂ or P₃ position.

The minor cycles in which the constants are introduced into TS16 also depend on the P₁, P₂ or P₃ positioning of the original number.

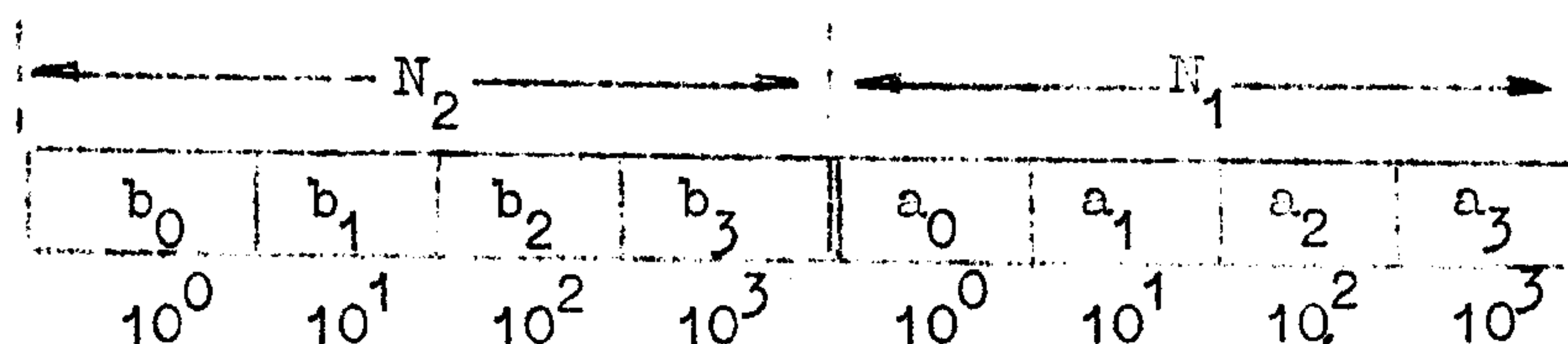
For example, if the final result is required at P₁ position in 21₃ and the original information in character form is also relative to P₁ the constants and minor cycles are

- $10^4 \cdot 2^7$ before mult.
- $10^3 \cdot 2^{13}$ m+15 or m+16.
- $10^2 \cdot 2^{19}$ m+27 or m+28.
- $10^1 \cdot 2^{25}$ m+7 or m+8.
- $10^0 \cdot 2^{31}$ m+19 or m+20.

At the end of MULT the result is relative to (P_{32}) even and 21-22(d) shifts N into 21_3 .

If the original information is at P_2 position the same constants will cause N to appear in 21_3 at the end of MULT and the shift instruction is not required. The minor cycles in which TS16 must be changed in this case are two earlier than those shown above.

When the conditions of a conversion are such that numbers $< 10^4$ are involved the programmer has the choice of several possibilities. Two half words N_1 and N_2 (in character form) may be converted in one operation using only 4 constants



In the final result 21_3 contains $N_1 P_{17} + N_2 P_1$ using the following instructions:

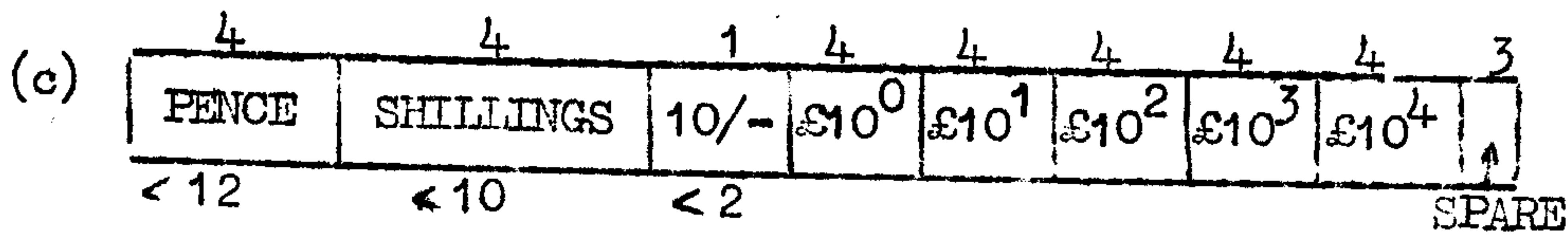
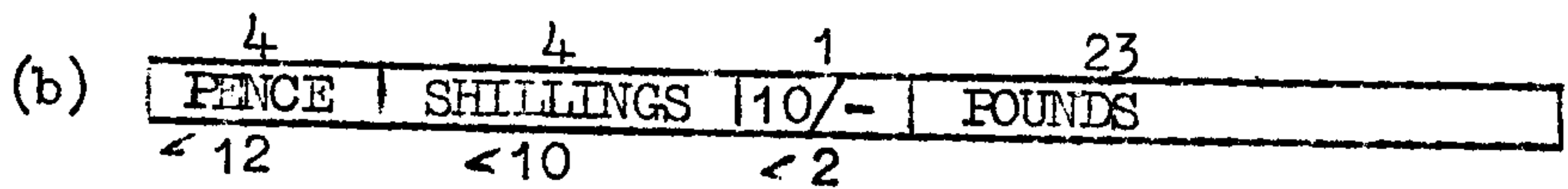
$(10^3 \cdot 2^{19})$	-	16	
N	-	21_3	
30	-	21_2	
0	-	24	(m.c. m)
$(10^2 \cdot 2^{23})$	-	16	m.c. m+7 or 8.
$(10^1 \cdot 2^{27})$	-	16	m.c. m+15 or 16.
$(10^0 \cdot 2^{31})$	-	16	m.c. m+23 or 24.
$(10^3 \cdot 2^{19})$	-	16	m.c. m+31 or m.
$(10^2 \cdot 2^{23})$	-	16	m.c. m+7 or 8.
$(10^1 \cdot 2^{27})$	-	16	m.c. m+15 or 16.
$(10^0 \cdot 2_{31})$	-	16	m.c. m+23 or 24.
21	-	22	(d)

$$\underline{N_1 P_{17} + N_2 P_1 \text{ in } 21_3}$$

The same four constants may be used twice if $10^3 \cdot 2^{19}$ is stored in m+31 or m.

3.2 Conversion of Binary Coded Stirling to E.s.d.

The original coded group information may be stored in several forms as illustrated.



or $\leq \text{£}7.10^5$ (unsigned)

For (a) $30-21_2$

$N-21_3$

£. $(240 \cdot 2^{22})-16$

0-24 (m.c. m)

s. $(20 \cdot 2^{27})-16$ m.c. m+45 or m+46

d. $(1 \cdot 2^{31})-16$ m.c. m+55 or m+56

21-22(d)

Binary pence in 21_3 with 21_2 clear.

For (b) $30-21_2$

$N-21_3$

£. $(240 \cdot 2^{22})-16$

0-24 (m.c. m)

10/- $(120 \cdot 2^{23})-16$ m.c. m+45 or 46.

s. $(20 \cdot 2^{27})-16$ m.c. m+47 or 48.

d. $(1 \cdot 2^{31})-16$ m.c. m+55 or 56.

21-22 (d)

It will be appreciated that the instruction within the dotted lines is unnecessary since $120 \cdot 2^{23}$ is the same as $240 \cdot 2^{22}$ which is already in TS16. This comes about through the binary relationship of 10/- and £. In diagram (b) (£) P_{10} are the same as (10/-) P_9

For (c), assuming $P_{30, 31, 32}$ of the diagram form a 3 bit group ≤ 7 .

$30 - 21_2$

$N - 21_3$

$(240 \cdot 10^5 \cdot 2^2) - 16$

0 - 24 m.c. m

$(240 \cdot 10^4 \cdot 2^6) - 16$ m+5 or m+6

$(240 \cdot 10^3 \cdot 2^{10}) - 16$ m+13 or m+14

$(240 \cdot 10^2 \cdot 2^{14}) - 16$ m+21 or m+22

$(240 \cdot 10^1 \cdot 2^{18}) - 16$ m+29 or m+30

$(240 \cdot 10^0 \cdot 2^{22}) - 16$ m+37 or m+38

$(20 \cdot 2^{27}) - 16$ m+47 or m+48

$(1 \cdot 2^{31}) - 16$ m+55 or m+56

Result

Binary Pence x P_1 in 21_3 with 21_2 clear.

3.3 Pattern Re-arrangement.

The multiplier may be used to reverse the order of digits in a word. The constants required are single P digits chosen from the following table:

<u>Digit.</u>	<u>Examined in m.c.</u>	<u>Multiplied by</u>	<u>Inserted in m.c.</u>	<u>becomes</u>	<u>Digit.</u>
32	m + 0	$2^{31} \cdot 2^1$ i.e. P_2	m+ 1		1
31	2	$2^{30} \cdot 2^3$ P_4	3		2
30	4	$2^{29} \cdot 2^5$ P_6	5		3
29	6	$2^{28} \cdot 2^7$ P_8	7		4
28	8	$2^{27} \cdot 2^9$ P_{10}	9		5
27	10	$2^{26} \cdot 2^{11}$ P_{12}	11		6
26	12	$2^{25} \cdot 2^{13}$ P_{14}	13		7
25	14	$2^{24} \cdot 2^{15}$ P_{16}	15		8
24	16	$2^{23} \cdot 2^{17}$ P_{18}	17		9
23	18	$2^{22} \cdot 2^{19}$ P_{20}	19		10
22	20	$2^{21} \cdot 2^{21}$ P_{22}	21		11
21	22	$2^{20} \cdot 2^{23}$ P_{24}	23		12
20	24	$2^{19} \cdot 2^{25}$ P_{26}	25		13
19	26	$2^{18} \cdot 2^{27}$ P_{28}	27		14
18	28	$2^{17} \cdot 2^{29}$ P_{30}	29		15
17	30	$2^{16} \cdot 2^{31}$ P_{32}	31		16
16	32	$2^{15} \cdot 2^1$ P_2	33		17
15	34	$2^{14} \cdot 2^3$ P_4	35		18
14	36	$2^{13} \cdot 2^5$ P_6	37		19
13	38	$2^{12} \cdot 2^7$ P_8	39		20
12	40	$2^{11} \cdot 2^9$ P_{10}	41		21
11	42	$2^{10} \cdot 2^{11}$ P_{12}	43		22
10	44	$2^9 \cdot 2^{13}$ P_{14}	45		23
9	46	$2^8 \cdot 2^{15}$ P_{16}	47		24
8	48	$2^7 \cdot 2^{17}$ P_{18}	49		25
7	50	$2^6 \cdot 2^{19}$ P_{20}	51		26
6	52	$2^5 \cdot 2^{21}$ P_{22}	53		27
5	54	$2^4 \cdot 2^{23}$ P_{24}	55		28
4	56	$2^3 \cdot 2^{25}$ P_{26}	57		29
3	58	$2^2 \cdot 2^{27}$ P_{28}	59		30
2	60	$2^1 \cdot 2^{29}$ P_{30}	61		31
1	62	$2^0 \cdot 2^{31}$ P_{32}	63		32

This table discloses the operation of the multiplier. The 20th digit of a multiplier in 21_3 , for example, is examined when it reaches P_{32} position i.e. in m.c. m+24. If it is a one, the contents of TS_{16} are added into 21_2 in minor cycle m+25, and suffer a shift of 2^{19} before the end of MULT. In COL. 3 the indices a and b in $2^a \cdot 2^b$ show the shift due to the multiplier and the shift conferred by the P position of the multiplicand respectively.

The instructions necessary, assuming that D.L.10 contains

P_2 in m.c. $m + 0$

P_4 in m.c. $m + 2$

\vdots

P_{30} in m.c. $m + 28$

P_{32} in m.c. $m + 30$ are,

$(a_1 a_2 \dots a_{31}, a_{32}) - 21_3$

30 - 21_2

3 - 24

0 - 24 (m.c. m)

30 - 13

21 - 25 (d) at the end of MULT.

Result $(a_{32}, a_{31} \dots a_2, a_1)$ in TS13.

If A and B are the more and less significant halves of the original word and A', B', are the reversed forms of A and B, the contents of 21 after Mult. are:-

21_2 B'P₁₇

21_3 A'P₁

These are merged into A'P₁ + B'P₁₇ by the last two instructions of the routine.

It should be noted that the constants in D.L.10 are stored in minor cycles one in advance of the insertion minor cycles shown in the table. This compensates for the one minor cycle delay inherent in T.C.A.'s operation.

Another example is the re-arrangement of groups within a word, preserving the order of significance of the binary number within each group.

(a)	E	D	C	B	A	00
(b)	A	B	C	D	E	00

The five six-bit groups A to E in (a) may be re-arranged to the form shown in (b) as follows:

a - 21_3

30 - 21_2

(P₉) - 16

0 - 24 (m.c. m)

(P₂₁) - 16 m+15 or 16

27 - 16 (P₁) m+27 or 28

(P₁₃) - 16 m+39 or 40

(P₂₅) - 16 m+51 or 52

30 - 13

21 - 25 (d)

Result (b) in TS13.

NELSON RESEARCH LABORATORIES
 STAFFORD E. E. CO. LTD.
 MATHEMATICS DEPARTMENT.

Continuation to: NS y 141

Sheet No.: 8.

If the constants are changed to P_{11} , P_{23} , P_3 , P_{15} and P_{27} the two spare digits in the results will occupy P_1 and P_2 positions and P_{10} , P_{22} , P_2 , P_{16} and P_{26} will result in a spare digit at either end of the word in TS13.

This technique of pattern re-arrangement has been used in the Alphacode Mk. II Compiler from which the next example is taken. T.C.A. is not used, all changes to Ts16 are made by instructions and no transfers are made from S21 until the multiplication is completed. The first stage in planning the sequence of instructions is the preparation of a table showing the re-arrangement required, from which the P position of each modifying constant may be determined by the following rules:

If P_m is transformed to P_n

- (1) For $m \leq n$, P_m must be multiplied by P_{n-m+1} to produce P_n in 21_2 at the end of MULT.
- (2) For $m > n$, P_m must be multiplied by P_{n-m+33} to produce P_n in 21_3 at the end of MULT.

The re-arrangement required is shown below.

NELSON RESEARCH LABORATORIES

Continuation to: NS y 141

STAFFORD E. E. CO. LTD.

Sheet No.: 9.

MATHEMATICS DEPARTMENT.

<u>Digit.</u>	<u>becomes</u>	<u>Digit</u>	<u>Examination m.c.</u>		<u>Insertion m.c.</u>		<u>TS16.</u>
32		zero	m+ 0	(21)	m+ 1	(22)	-
31		3	2	(23)	3	(24)	P ₅
30		zero	4	(25)	5	(26)	-
29		14	6	(27)	7	(28)	P ₁₈
28		13	8	(29)	9	(30)	P ₁₈
27		2	10	(31)	11	(0)	P ₈
26		30	12	(1)	13	(2)	P ₅
25		29	14	(3)	15	(4)	P ₅
24		28	16	(5)	17	(6)	P ₅
23		27	18	(7)	19	(8)	P ₅
22		4	20	(9)	21	(10)	P ₁₅
21		22	22	(11)	23	(12)	P ₂
20		21	24	(13)	25	(14)	P ₂
19		20	26	(15)	27	(16)	P ₂
18		19	28	(17)	29	(18)	P ₂
17		18	30	(19)	31	(20)	P ₂
16		17	32	(21)	33	(22)	P ₂
15		10	34	(23)	35	(24)	P ₂₈
14		9	36	(25)	37	(26)	P ₂₈
13		8	38	(27)	39	(28)	P ₂₈
12		7	40	(29)	41	(30)	P ₂₈
11		6	42	(31)	43	(0)	P ₂₈
10		26	44	(1)	45	(2)	P ₁₇
9		25	46	(3)	47	(4)	P ₁₇
8		24	48	(5)	49	(6)	P ₁₇
7		23	50	(7)	51	(8)	P ₁₇
6		5	52	(9)	53	(10)	P ₃₂
5		16	54	(11)	55	(12)	P ₁₂
4		15	56	(13)	57	(14)	P ₁₂
3		1	58	(15)	59	(16)	P ₃₁
2		32	60	(17)	61	(18)	P ₃₁
1		31	62	(19)	63	(20)	P ₃₁

The first two columns specify the required transformation (e.g. a P₂₉ in the original word becomes a P₁₄). The third column shows the minor cycle, in which the digits of the original word are examined as they reach P₃₂ position in 21₃. The figures in brackets are the actual minor cycles appropriate to the coding. The fifth column shows the multiplying constant which must be in TS16 in the minor shown in column four. Changes may be made to TS16 in either of the two minor cycles immediately preceding those shown in the fourth column (e.g. the P₂₈ can replace P₂ in either m.c. m+22 or m+23).
 m+33(22) or m+54(23)

NELSON RESEARCH LABORATORIES

Continuation to: NS y 141

STAFFORD E. E. CO. LTD.

Sheet No.: 10.

MATHEMATICS DEPARTMENT.

At the end of the process some digits are in 21_2 and others in 21_3 . The sum of these gives the result.

The sequence of instruction is:

	2_{29}	10_0	- 16	(P ₅)
	2_1	9_7	- 14	(P _{1-29, 31})
(a)	2_{14}	30	- 21_2	
(b)	2_{17}	25	- 21_3	
(c)	2_{19}	0	- 24	(m.c. 21)
(d)	2_{24}	10_{26}	- 16	(P ₁₈)
	2_{27}	10_{30}	- 16	(P ₈)
	2_{30}	10_0	- 16	(P ₅)
	2_6	10_8	- 16	(P ₁₅)
	2_8	10_{10}	- 16	(P ₂)
	2_{20}	10_{22}	- 16	(P ₂₈)
	2_{22}	10_{25}	- 14	(P ₁₃)
	3_{29}	28	- 16	(P ₁₇ m.c. 1)
	2_7	29	- 16	(P ₃₂ m.c. 9)
	2_9	23	- 16	(P ₁₂ m.c. 11)
	2_{12}	10_{14}	- 16	(P ₃₁)
(e)	2_{21}	3	- 24	
(f)	2_{23}	21_2	- 14	
(g)	2_{26}	30	- 21_2	
(h)	2_{28}	14	- 22_3	

- NOTES.
- (a) In all respects except the constancy of the multiplicand this is a normal multiplication and 21_2 must be clear.
- (b) The word to be transformed is sent to 21_3 and P_{30, 32} are eliminated.
- (c) The first minor cycle of MULT is m.c. 21.
- (d) The next eleven instructions are all concerned with varying the contents of TS16. Use is made of the ability to change TS16 in either of two minor cycles e.g. $\frac{2_{27} 10_{30}-16}{3_{29} 28-16 (m.c. 1)}$ could be $\frac{10_{31}-16}{28-16 (m.c. 0)}$ with P₈ in 10_{31} .
- (e) These instructions anticipate the next part of the programme.
- (f) Multiplication finishes in m.c. 21 and S21 gives the required result any time after this.
- (g) Forms the required answer. Alternative endings are:
- | | |
|------------|--------------|
| 30-13 | 21_2 - 14 |
| 21-25 d | 21_3 - 15 |
| 13-result. | 26 - result. |

3.4 Counting the Digits of a Word.

This is, in effect, a variation of the pattern re-arranging technique which produces an information compression. The counting can be done in one or two ways. The method used here produces, separately

- (a) The sum of all digits.
- (b) The sum of the most significant 16 digits.
- (c) The sum of the least significant digits.

Use is made of the ability to extract information from S21 while the multiplication is in progress.

The digits of the input word are sensed in the odd minor cycles starting at P_{32} . If P_{32} is present, add P_1 into 21_2 in the next even minor cycle. If P_{31} is present, add another "one" to the count in 21_2 . The original digit in 21_2 has moved up one place since it was inserted and the units position of the counter is now at P_2 position. The "one" which scores the presence of a P_{31} has to be a P_2 . In the example given, counting is performed at two places in 21_2 , P_1 and P_{17} . Both counters move up and half way through MULT

we have the sum of the more significant 16 digits, S_M , at P_{16} and P_{32} positions in 21_2 . At the end of MULT we have the sum of the least significant 16 digits, S_L , at P_{16} in 21_2 , the sum of all digits S_A at P_{32} in $21_{2,3}$ and S_m has now moved up to P_{16} and P_{32} positions in 21_3 , (most of the P_{32} part has been lost). S_m at P_{16} and P_{32} positions is extracted, half way through MULT and subtracted from 21_3 at the end of MULT. The instructions are

$N - 21_3$	$30 - 21_2$	$3 - 24$	$0 - 24$ (m.c. m)	$21_2 - 14$ (m.c. m+24)	$14 - 23_3$ After MULT.	$21 - 22$ (d)	$24 - 14$	<u>D.L.</u> ₁₀ contains
								$10_m P_1 + P_{17}$
								$10_{m+2} P_2 + P_{18}$
								$10_{m+4} P_3 + P_{19}$
								⋮
								$10_{m+30} P_{16} + P_{32}$

Result.

- $S_A \cdot P_1$ in 21_3
- $S_L \cdot P_{17}$ in 21_2
- $S_M \cdot P_{17}$ in 14

3.5 Conversion of Card Row Patterns to Binary.

Part of R21T/1, read 6 nine digit numbers, is taken as an example. The technique using T.C.A., to take full advantage of speed and the freedom conferred by parallel programming, converts card row patterns in the limited time between rows of a card.

STAFFORD E. E. CO. LTD.

Sheet No.: 12.

MATHEMATICS DEPARTMENT.

The principles involved are the same as in normal read-routines. Specified sequences of digits within one or two (α and/or β) words read from a row of a card are to be regarded as numbers to base 10 and converted to binary. The binary contributions for each row are summed to produce the binary equivalents of the various decimal numbers on the card.

The word containing the sequence is transferred to 21_3 and multiplication started. The contents of TS16 are varied, via D.L.10 and T.C.A., throughout the multiplication. Successive digits of the sequence are multiplied by successive powers of 10, stored in D.L.10. Because the partial products in 21_2 are built up from the least significant end of the sequence and because they move up by the shift action of the multiplier, the successive powers of 10 must also move up. They are, in fact, powers of 20.

All minor cycles of D.L.10 pass through TS16 twice during multiplication. Two sequences can be processed, one in each major cycle of MULT, provided that the sequences are present in 21_3 in corresponding positions. This can be done either by a pre-arrangement of the words or by using the automatic shift of the multiplier, extracting the next sequence due for conversion at an appropriate time and re-inserting it in 21_3 also at the correct time.

Since this technique is only used when several sequences have to be converted, it is necessary to clear 21_2 after one sequence has been converted and before the next sequence is dealt with.

The first power of 20 is stored in the same odd m.c. of DL10 as that in which the first digit of a sequence is examined in 21_3 . The even minor cycle following is the one in which 21_2 has to be cleared.

Minor cycles of D.L.10 not required for powers of 20 can be used as normal storage locations. The passage of this information through TS16 will not affect the conversions provided that the converted result for each sequence is extracted as soon as it is complete and 21_2 is cleared before dealing with each sequence. The result for any sequence is extracted from 21_2 in the even minor following the even m.c. in which the last power of 20 is added into 21_2 .

The number of binary places to which the ascending powers of 20 are stored in D.L.10 depends on the required layout of the results and on the way in which the converted sequences are extracted.

If the results are all integers $x P_1$ and the powers of 20 are stored $x P_1$ the converted sequence for, say, an eight digit sequence will be stored at P_0 position (i.e. $x 2^7$). The last power of 20 in this case is $10^7 \cdot 2^7$. R24T (Read eight 8 digit numbers) is a case in point. The maximum row contribution is $2^7 \cdot 11,111,111$ and as this is within single length the following sequence for accumulating row contributions can be used.

$21_2 - 14$	Extract converted row sequence.
$23 - 14 \ 1$ (7 m.c.)	
$14 - 25$	Add to previous row sums.

In general however an n digit sequence after conversion will be formed at P_n position (i.e. $x 2^{n-1}$) and the last power of 20 is $(10^{n-1} \cdot 2^{n-1})$.

This raises out of length difficulties in the case of a nine digit sequence. The largest converted sequence, $2^8 \times 111, 111, 111$, is out of single length and requires a double length shifting down register.

To overcome this a device originally used on R23T/2 is employed. The powers of 10 for an n digit number are stored so that 10^r is in position $P_{33 - (n-r)}$ or $P_{32 + m - (n-r)}$ for results required $\times P_m$. Any digits in excess of single length in any power of 20 are placed at the P_1 position of the same word, overflowing into itself!

When all the digits of a sequence have been dealt with the results of the conversion are in two places, in $21_2 \times P_1$ (or P_m) for the excess digits and in $21_3 \times P_1$ (P_m) for the contributions from the powers of 20 which are not out of length. These two parts have to be added together to form the final conversion.

In the example the sequences to be converted are digits 22-30 and 12-20

		<u>T.C.A. ON</u>		<u>Contents of D.L.10.</u>	
(a)	3_{15}	18_1	- 21_3	10_{25}	$20^0 \times P_{25} = 0, 0, 0, 0, 16, 0, 0$
(b)	2_{17}	3_{19}	- 15	10_{27}	$20^1 \times P_{25} = 0, 0, 0, 0, 0, 10, 0$
		2_{19}	- 24	10_{31}	$20^2 \times P_{25} = 1, 0, 0, 0, 0, 8, 2$
(c)	2_{24}	30	- 21_2	10_{31}	$20^3 \times P_{25} = 31, 0, 0, 0, 0, 0, 1$
		4_{27}	- 0	10_1	$20^4 \times P_{25} = 17, 19, 0, 0, 0, 0, 0$
(d)	Q_{30}	4_{45}	- 20	10_3	$20^5 \times P_{25} = 20, 6, 12, 0, 0, 0, 0$
		2_5	- 13	10_5	$20^6 \times P_{25} = 16, 4, 20, 7, 0, 0, 0$
(e)	2_7	21	- 19 (d)	10_7	$20^7 \times P_{25} = 0, 26, 18, 24, 4, 0, 0$
(f)	2_{13}	19_3	- 14	10_9	$20^8 \times P_{25} = 0, 8, 24, 11, 31, 2, 0$
(g)	2_{18}	25	- 25		
(h)	2_{21}	23	- 21_3		
(c)	4_{24}	30	- 21_2		
(i)	4_{26}	19_2	- 25		
		2_{29}	- 20_3		
		2_{31}	- 13		
(i)	2_8	21_2	- 25		
(f)	3_{13}	22_3	- 14		
(g)	2_{20}	25	- 25		
		2_{22}	- 20_2		
		4_{25}	- 0		
(d)	Q_{30}	20	- $4_{4,5}$		

NELSON RESEARCH LABORATORIES

Continuation to: NS y 141

STAFFORD E. E. CO. LTD.

Sheet No.: 14.

MATHEMATICS DEPARTMENT.

- NOTES:
- (a) The multiplier is placed in 21_3 . Digits 22-30 have to be converted as described and added to the number in 4_5 ; digits 12-20 when converted are added to 4_4 .
 - (b) These are collate digits for extracting the eight least significant digits of 21_3 in (g).
 - (c) Multiplication started in m.c. 21. The first digit of the sequence is not examined until m.c. 25; the first addition from TS16 can take place in m.c. 26 and 21_2 is cleared in this m.c.
 - (d) Housekeeping.
 - (e) The powers of 20 are stored ($\times P_{25}$) in the odd minor cycles 10_{25} to 10_9 . The last addition from TS16 into 21_2 takes place in m.c. 10 and the separate parts of the conversion in 21_2 and 21_3 can be extracted in m.c. 12 and 13. 19_3 will contain, in the first 8 digits, part of the converted sequence. The next 4 digits are rubbish and digits 13-32 are digits 1-20 of the original 21_3 .
 - (f) The number transferred to TS14 is what emerged from 21_3 in m.c. 13 - or from $S22_3$ in m.c. 15) as these are the same so far as the bottom eight digits are concerned.
 - (g) Least significant eight digits added to TS13.
 - (h) The new multiplier goes to 21_3 so that the first digit (20) of the next sequence is examined in m.c. 25, second time round.
 - (i) The addend is the contribution to the converted sequence obtained from $S21_2$ in m.c. 12.

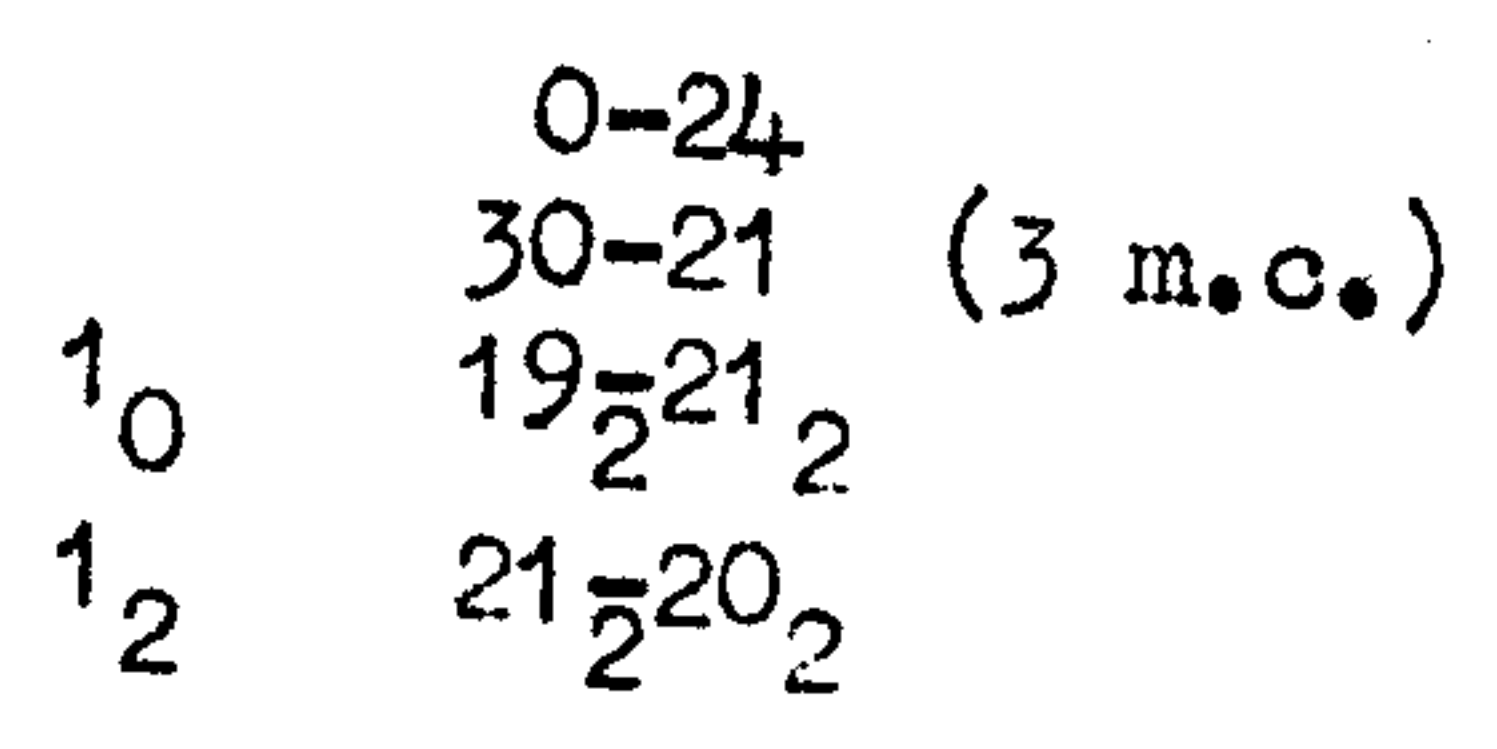
3.6 Using the Multiplier as a Shift Register.

If 21_2 contains x at the start of a multiplication with either 21_3 or 16 clear, then 21_3 contains x at the end of multiplication; x , in fact, has been shifted by 2^{32} . Here however is the place to issue a warning! The detailed operation of D24 is such that if x extends beyond P_{26} a zero will be introduced in P_{27} position and all digits above this are moved up one place with the top digit of 21_2 subsequently moved into P_1 position of 21_3 . When the multiplier is used as a shifting register it can be started in an even minor cycle in which case x can be full word length and is not restricted to $< P_{27}$. The following notes contain recommendations for this mode of operation.

- (a) Clear either 21_3 or 16 if possible.
- (b) Place x in 21_2 and start 0-24 in either an odd ($x < P_{27}$) or even m.c. The maximum shift is 32 if started in an odd m.c. and 33 if started in an even m.c.
- (c) Extract x from $S21_2$ or $S22_2$ in the minor cycle appropriate to the shift required.

- (d) The multiplier can be started before a number is inserted in 21_2 (less total shift will be produced). If 21 is cleared after a 0-24 operation starts a transfer of 30-21 for 3 m.c. (e. o, e) required if TS16 is not clear, otherwise 30-21(d) is sufficient.

Any information sent to D21 after 0-24 starts arrives in D21 shifted up one place, thus after the sequence,



20_2 contains 19_2 shifted up one place. The shifting facility and the use of D23 during multiplication has been used (originally by J.P. O'Brien) to effect multiplication by 10 for the purpose of converting binary to binary coded decimal.

The next example illustrates this technique.

Example.

To convert a binary number N (to 32 b.p.) to 4-bit binary coded decimal.

TS16 contains a decimal fraction N (in binary) to 32 b.p. which it is required to convert to an integer as in normal punch routine practice in 4-bit binary coded decimal with four decimal digits, A, B, C, D. This routine produces the result in 20_3 as AP_{13} , BP_9 , CP_5 , DP_1

	N - 16	
	30 - 21_2	
	$10P_{29}$ - 21_3	
(a)	2_{29}	0 - 24 (m.c. 31)
	2_{31}	30 - 20_3
(b)	2_6	21_2 - 20 (m.c. 8)
(c)	2_8	20 - 23 (4 m.c. eo)
(d)	2_{13}	21_2 - 20 (m.c. 16)
(e)	2_{16}	20 - 23 (4 m.c. eo)
(f)	2_{21}	21_2 - 20 (m.c. 25)
(g)	2_{24}	20 - 23 (4 m.c. eo)
(h)	2_{29}	21_3 - 20_3 (m.c. 1)

- NOTES:
- (a) Multiply ABCD by 10. This is complete in m.c. 6 when A.BCD enters 21_2 and 21_3 with .BCD in 21_2 and A in 21_3 .
 - (b) Extract .BCD in m.c. 8
 - (c) In m.c. 10, 4(.BCD) is entering 21_2 and (0.BCD) is subtracted from it leaving 3(.BCD) This emerges in m.c.12 and enters 21_2 as 6(.BCD) from which another subtraction is made to leave 5 (.BCD). This emerges in m.c.14, is doubled on re-entry and emerges as AB.CD in m.c.16.
 - (d) Extract .CD

- (e) Repeat Stage (c) to give ABC.D
- (f) Extract .D
- (g) Repeat stage (C) to give ABCD
- (h) ABCD to 20_3 .

This technique has been used in P19T/1 P25T and DG01T.

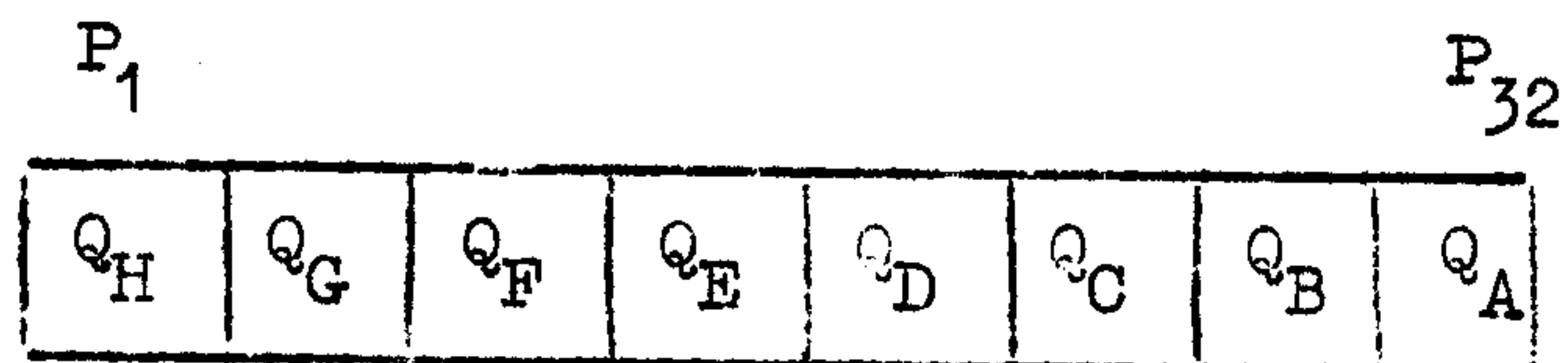
4. DIVISION OPERATIONS.

The DEUCE divider, given A (dividend) and B (divisor) produces a quotient Q and remainder R given by

$$2^{31} A = QB + R \quad 0 \leq R < B \text{ for } B > 0$$

where A, B, Q, and R are all integers.

Let it be supposed that a final quotient Q can be obtained which consists of eight different quotients each in 4-bit character form as shown



These characters can be obtained by using different divisors $B_A B_B B_C B_D$ etc. to B_H so that

$$2^{31} A = (Q_A \cdot 2^{28}) B_A + (Q_B \cdot 2^{24}) B_B + (Q_C \cdot 2^{20}) B_C + (Q_D \cdot 2^{16}) B_D + (Q_E \cdot 2^{12}) B_E + (Q_F \cdot 2^8) B_F + (Q_G \cdot 2^4) B_G + (Q_H \cdot 2^0) B_H$$

$$\text{i.e. } A = Q_A (B_A \cdot 2^{-3}) + Q_B (B_B \cdot 2^{-7}) + Q_C (B_C \cdot 2^{-11}) + Q_D (B_D \cdot 2^{-15}) + Q_E (B_E \cdot 2^{-19}) + Q_F (B_F \cdot 2^{-23}) + Q_G (B_G \cdot 2^{-27}) + Q_H (B_H \cdot 2^{-31})$$

If we choose

B_A	=	10^7	\cdot	2^3
B_B	=	10^6	\cdot	2^7
B_C	=	10^5	\cdot	2^{11}
B_D	=	10^4	\cdot	2^{15}
B_E	=	10^3	\cdot	2^{19}
B_F	=	10^2	\cdot	2^{23}
B_G	=	10^1	\cdot	2^{27}
B_H	=	10^0	\cdot	2^{31}

then $A = Q_A \cdot 10^7 + Q_B \cdot 10^6 + Q_C \cdot 10^5 + Q_D \cdot 10^4 + Q_E \cdot 10^3 + Q_F \cdot 10^2 + Q_G \cdot 10^1 + Q_H \cdot 10^0$ and by this means a conversion from binary to 4-bit binary

coded decimal is accomplished. For positive divisors the result is exact (R = 0) though corrections of the form $(2^n - 10)$ may be necessary and the last character may need to be inverted. These corrections are now discussed.

For any case in which the least significant digit of the least significant character is in the P_1 position the final divisor is 2^{31} . This is treated as a negative divisor and its effect is to interchange the 0's and 1's of the final character. This can be corrected after the division operation is complete.

The second correction arises from the changes in numerical value of the divisors. If any character is even, the character next below is in excess by 6 (i.e. $2^4 - 10$) for 4-bit characters representing binary coded decimal and will be in excess by 54 (i.e. $2^6 - 10$) for a decimal representation using 6-bit characters. The amount by which any character may be in excess depends on the number of bits in the character and the radix of the quantity represented by that character.

Suppose that, at some stage in the process, divisor $B_D = 10^4 \cdot 2^{15}$ has been in use, generating quotient Q_D (the number of fourth powers of 10 in the original number) and further suppose that a remainder R is left after Q_D has been formed. B_D is replaced by $B_E (10^3 \cdot 2^{19})$ and effectively a new division is started with R_D as divided. If R_D is positive all is well; if R_D is negative however B_D has been subtracted too often and the next operation should be an addition of B_D to restore the remainder. B_D is now changed to B_E and instead of adding $10^4 \cdot 2^{15}$ to restore the remainder, $10^3 \cdot 2^{19}$ is added.

i.e. $10^4 \cdot 2^{15}$ should be added to restore remainder

but $10^3 \cdot 2^{19}$ is added instead.

$$\text{now } 10^3 \cdot 2^{19} = 10^4 \cdot 2^{15} + 10^3 \cdot 2^{19} - 10^4 \cdot 2^{15}$$

$$= 10 (10^3 \cdot 2^{15}) + 6 (10^3 \cdot 2^{15})$$

$$10^4 \cdot 2^{15} \text{ represents } 10 \text{ units of } 10^3$$

therefore $6 \cdot 10^3 \cdot 2^{15}$ represents 6 units of 10^3 added in excess.

Corrections of 6 will therefore be required in any character which follows an even character since the presence of a zero in the last digit of a character means that the first operation performed with the next division is an addition.

For the conversion of binary to binary coded decimal the difference between successive divisors is of the form $a \cdot 10^n$ where a has the same value for each character position and therefore the correction required is the same for each character. This would not be so if, for example the conversion is one of binary LB WT to TONS, CWTS, QTR. and LB. The rule that a correction is required in a character next below an even character is still true but the amount of the correction may vary from character to character.

In practice it is not necessary to wait for the full division time (66 m.c.) to initiate correction procedure; time and instructions can be saved by extracting the quotient when the least significant digit of each character is justified (lined up) relative to the final P_2 position of the next character below. An example will illustrate this technique.

4.1 To convert A ($< 10^8$) in binary to 4-bit binary coded decimal.

- (a) $(10^7 \cdot 2^3) - 16$
- (b) $A - 21_3$
- (c) $1 - 24$ (m.c. m)
- $(10^6 \cdot 2^7) - 16$ m+6 or 7
- $(10^5 \cdot 2^{11}) - 16$ m+14 or 15
- $(10^4 \cdot 2^{15}) - 16$ m+22 or 23
- (d) $(P_{2,6,10,14,18,22,26} - 15)$ as convenient.
- $(10^3 \cdot 2^{19}) - 16$ m+30 or 31
- $(10^2 \cdot 2^{23}) - 16$ m+38 or 39
- $(10^1 \cdot 2^{27}) - 16$ m+46 or 47
- $(10^0 \cdot 2^{31}) - 16$ m+54 or 55
- (e) $30 - 13$
- (f) $21_2 - 14$ (m+59)
- (g) $25 - 14$
- (h) $26 - 26$ (3 m.c.)
- (j) $(P_{1-4}) - 15$
- (k) $21_2 - 14$
- (l) $26 - 25$

Result in TS13.

- NOTES:
- (a) First divisor to TS16.
 - (b) Dividend to 21_3 .
 - (c) Divisor started in minor cycle m.
 - (d) Collate digits aligned relative to the P_2 position of the final character positions.
 - (e) Clear TS13 anticipating (h), Any convenient m.c.
 - (f) Extracts the character quotient when P_1 of each character (except Q_H) is aligned relative to the P_2 position of the character next below.
 - (g) Selects digits in those characters with a $P_1 = 1$
 - (h) Forms a word with digits with a $P_1 = 0$ and subtracts this word 3 times from TS13 thus forming the necessary correction.
 - (i) Collate digits for Q_H
 - (k) Uncorrected quotient extracted after division is finished.
 - (l) Inverts Q_H and adds to corrections in TS13.

4.2 Conversions of Numbers $N < 10^4$.

The routine given in 4.1 will obviously work for $N < 10^8$ and if used for $N < 10^4$ the top four characters will be zero. It is both wasteful of time and instructions to use the 10^8 routine for cases where all numbers are less than 10^4 .

If $N = A \cdot 10^3 + B \cdot 10^2 + C \cdot 10^1 + D \cdot 10^0$ the final result can be attained, A, B, C, D, positioned as in (a) or (b) or any intermediate form by correct choice of divisors

(a)

D	C	B	A	zero	zero	zero	zero
---	---	---	---	------	------	------	------

(b)

zero	zero	zero	zero	D	C	B	A
------	------	------	------	---	---	---	---

Since divisors can be chosen so that characters finish as in (b) then during such a process the characters are complete (and at the bottom end of the register) half way through the division process. They can be extracted and corrected while the 1-24 is still in progress and the remaining millisecond of division can be used for any processing and storing required.

Example. To convert $N < 10^4$ to the form shown in (a)

$(10^3 \cdot 2^{19})$	-	16	
N	-	21_3	
1	-	24	(m.c. m)
$(P_{2,6,10})$	-	15	any convenient m.c.
$(10^2 \cdot 2^{23})$	-	16	m.c. m+38 or 39
$(10^1 \cdot 2^{27})$	-	16	m.c. m+46 or 47
$(10^0 \cdot 2^{31})$	-	16	m.c. m+54 or 55
30	-	13	
21_2	-	14	m.c. m+59
25	-	14	
26	-	26	(3 m.c.)
(P_{1-4})	-	15	
21_2	-	14	
26	-	25	

Result in TS13.

The sequence of instructions and minor cycle timing are the same as in 4.1. Inversion correction is still required for the least significant character. Less constants are required and more than one millisecond is available for parallel programming between 1-24 and the insertion of $(10^2 \cdot 2^{23})$.

NELSON RESEARCH LABORATORIES

Continuation to: NS y 141

STAFFORD E. E. CO. LTD.

Sheet No.: 20.

MATHEMATICS DEPARTMENT.

Example.

To convert $N < 10^4$ to the form (a) using constants which finally position the (uncorrected) characters as in (b).

The first task is to choose the constants appropriate to character positions (b).

We require divisors B_A, B_B, B_C, B_D such that

$$2^{31}N = A \cdot 2^{28} (B_A) + B \cdot 2^{24} (B_B) + C \cdot 2^{20} (B_C) + D \cdot 2^{16} (B_D)$$

$$\text{and } N = A \cdot 10^3 + B \cdot 10^2 + C \cdot 10^1 + D \cdot 10^0$$

$$\text{therefore } 10^3 = 2^{-3} B_A \quad B_A = 10^3 \cdot 2^3$$

$$10^2 = 2^{-7} B_B \quad B_B = 10^2 \cdot 2^7$$

$$10^1 = 2^{-11} B_C \quad B_C = 10^1 \cdot 2^{11}$$

$$10^0 = 2^{-15} B_D \quad B_D = 10^0 \cdot 2^{15}$$

Note that B_D , the last divisor is not $\times 2^{31}$ and that final character inversion will not be required, which is an advantage which will always be obtained if the least significant digit of the final character is at any position other than P_1 of a DEUCE word.

The instructions are:-

	$(10^3 \cdot 2^3)$	-	16	
	N	-	21_3	
(a)	30	-	21_2	
	1	-	24	(m.c. m)
	$(10^2 \cdot 2^7)$	-	16	m.c. m+6 or 7
(b)	$P_{2,6,10}$	-	15	
	$(10^1 \cdot 2^{11})$	-	16	m.c. m+14 or 15
(c)	30	-	13	
	$(10^0 \cdot 2^{15})$	-	16	m.c. m+22 or 23.
(d)	21_2	-	14	m+27
(g)	25	-	14	
	26	-	26	(3 m.c.)
(h)	22_2	-	25	m+35
(j)	28	-	26	

$$AP_{13} + BP_9 + CP_5 + DP_1 \text{ in TS13}$$

- NOTES:
- (a) In normal division operations there is no need to clear 21_2 . Here however it is better to do so to avoid separating the required information in $P_1 - P_{15}$ from rubbish which may exist in P_{16-32} at the time the information is extracted from 21_2 .
 - (b) Collate digits for correction sensing.
 - (c) Anticipates use of D26 later.
 - (d) Partial result extracted with LS digit of C at P_2 , B at P_6 , A at P_{10} .
 - (g) Forms word where LS digits of A, B, C are zero and forms correction factor in TS13.
 - (h) Extracts uncorrected characters,

$$AP_{13} + BP_9 + CP_5 + DP_1$$

Note that partial quotients are always $X P_2$ in 21_2 and S 22 is required.

- (g) In all division operations, normal or otherwise, the "first" digit of a positive quotient is a "spurious" one which ultimately drops off the end of 21_2 . Here we are extracting from 21_2 half way through division and the "spurious" one is at P_{17} position (conveniently) from which 28-26 removes it.

An alternative method of clearing 21_2 and at the same time neatly removing the spurious one is

$$\begin{array}{rcl} (10^3 \cdot 2^3) & - & 16 \\ N & - & 21_3 \\ 1 & - & 24 \quad (\text{m.c. } m) \\ 30 & - & 21_2 \quad \text{m.c. } m+3 \\ (10^2 \times 2^7) & - & 16 \\ & \vdots & \\ & \text{etc.} & \end{array}$$

21_2 is cleared after division starts (see D19/2)

4.3

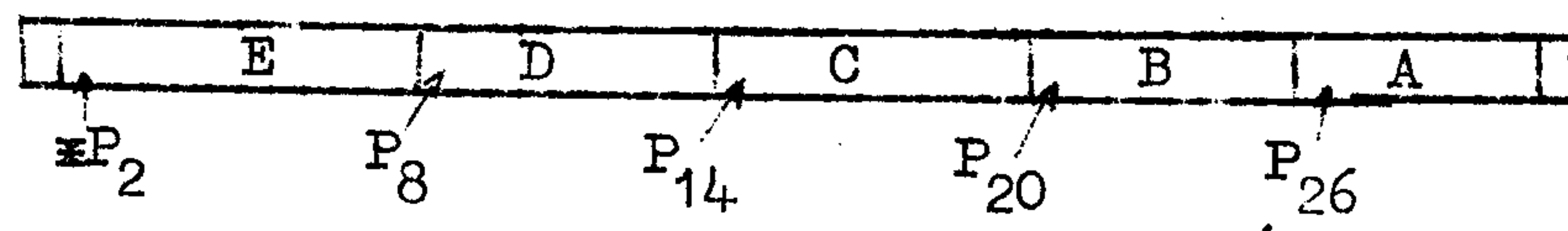
To convert binary to 6-bit binary coded decimal ($N < 105$) Five six-bit characters can be positioned in a word with P_{31}, P_{32} zero, $P_1 P_{32}$ zero or P_1, P_2 zero, i.e. with the two spare bits as zeros both at the top end, one at each end or both at the bottom end.

Three choices of constants can be made to effect conversions, one for each case. If we choose to have P_{31}, P_{32} zero, the least significant character is at P_1 position and the final divisor will be $10^0 \cdot 2^{31}$ and last character inversion is required. The best choice is with P_1, P_{32} zero, avoiding inversion correction and permitting any of the three cases to be obtained by one shift in either direction as appropriate.

The corrections to be made to any character are 54. Any character which is in excess by 54 has digits in the P_5, P_6 positions of the character since the normal decimal digits 0-9 occupy P_1-P_4 only. Hence, any character with P_5 and P_6 needs correcting since the only valid combinations for any character are 0-9.

This logic is used to effect the correction.

The first task is choosing the divisors



We require:

$$2^{31}N = A \cdot 2^{25} (B_A) + B \cdot 2^{19} (B_B) + C \cdot 2^{13} (B_C) + D \cdot 2^7 (B_D) + E \cdot 2^1 (B_E)$$

$$\text{and } N = A \cdot 10^4 + B \cdot 10^3 + C \cdot 10^2 + D \cdot 10^1 + E \cdot 10^0$$

$$\text{therefore } B_A = 10^4 \cdot 2^6$$

$$B_B = 10^3 \cdot 2^{12}$$

$$B_C = 10^2 \cdot 2^{18}$$

$$B_D = 10^1 \cdot 2^{24}$$

$$B_E = 10^0 \cdot 2^{30}$$

The instructions are:

	$(10^4 \cdot 2^6)$	-	16	
	N	-	21_3	
	1	-	24	(m.c. m)
(a)	Strobe	-	15	$(P_{2,3,8,9,14,15,19,20})$
	$(10^3 \cdot 2^{12})$	-	16	m.c. m+12 or 13
	$(10^2 \cdot 2^{18})$	-	16	m.c. m+24 or 25
	$(10^1 \cdot 2^{24})$	-	16	m.c. m+36 or 37
	$(10^0 \cdot 2^{30})$	-	16	m.c. m+48 or 49
(b)	21_2	-	14	(m+57)
(c)	22_2	-	13	after division.
(d)	25	-	26	(9 m.c.)

Result in TS13.

NOTES: (a) The characters are positioned at $P_2, P_8, P_{14}, P_{20}, P_{26}$ by the conversion but (c) using S22 will bring their final positions to $P_1, P_7, P_{13}, P_{19}, P_{25}$. Corrections where required are $54 P_1$ relative to the P_1 of each character i.e. $27 P_2 = 9 (3P_2)$. The strobe provides $3P_2$ in each character position.

(b) At this stage all characters A, B, C, D are formed and the top two digits of E are opposite P_2 and P_3 i.e. the top two ones of any 54's if present are in line with the strobe.

(c) This extracts uncorrected characters at positions

$$P_1 \quad P_7 \quad P_{13} \quad P_{19} \quad P_{25}$$

(d) Nine threes are 27 wherever required.

A word of warning. This logic of character correction is valid if no character exceeds 15, i.e. P_5 and 6 of each character should be zero. This conversion technique will handle numbers greater than the range in which each decimal character is restricted to 9. For example, the routine of 4.1 would convert 139,547,286 ($> 10^8$) correctly, but unusually, as

$$13 \cdot 10^7 + 9 \cdot 10^6 + 5 \cdot 10^5 + 4 \cdot 10^4 + 7 \cdot 10^3 + 2 \cdot 10^2 + 8 \cdot 10^1 + 6 \cdot 10^0$$

4.4 Conversion of binary Pence to Binary £. s. d.

By choosing the appropriate divisors in TS16 it is possible to convert binary pence to three packed integers representing binary £, binary shillings and binary pence. The constants are chosen to effect the conversion with the groupings shown.

5	5	21
PENCE (d)	SHILLINGS	£ POUNDS.

P_1 spare $P_7 - 11$ shillings.

$P_2 - P_6$ PENCE $P_{12} - 32$ POUNDS.

The first task is to choose the divisors. We require

$$2^{31}N = 2^{11} (\text{£}) \cdot (2^a \cdot 240) + 2^6 (\text{s}) (2^b \cdot 12) + 2^1 (\text{d}) (2^c \cdot 1)$$

we choose a, b and c such that

$$11 + a = 31 \quad \text{therefore } a = 20$$

$$6 + b = 31 \quad b = 25$$

$$1 + c = 31 \quad c = 30$$

2^{31} then cancels throughout and

$$N = (\text{£}) \cdot 240 + (\text{s}) \cdot 12 + (\text{d})$$

The divisors are 240×2^{20}

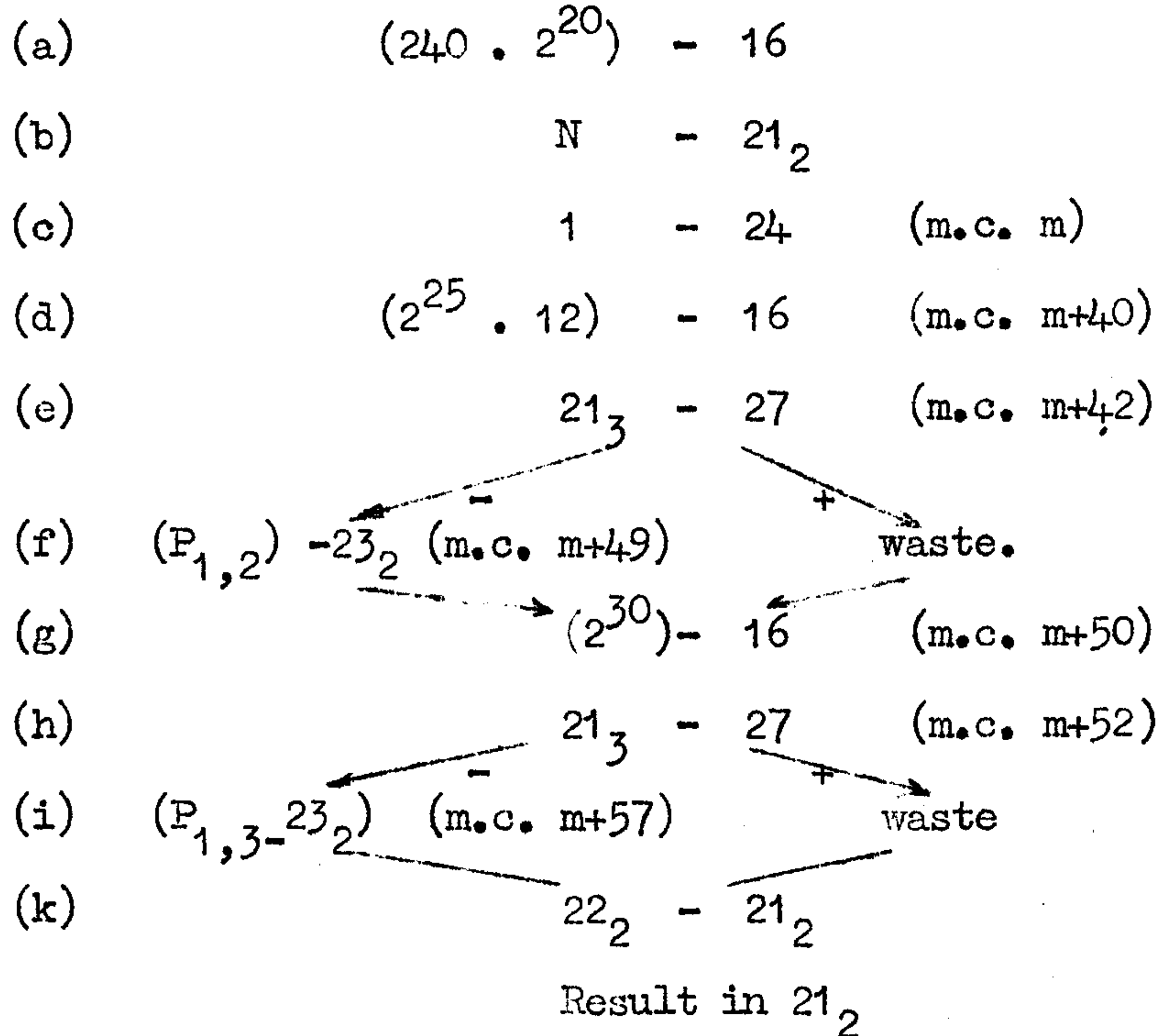
$$12 \cdot 2^{25}$$

$$1 \cdot 2^{30}$$

The next task is to predict the correction terms. If the £ group is even $2^{25} \cdot 12$ will be added where $2^{20} \cdot 240$ should be added. The excess is $144 \cdot 2^{20}$ representing 12 shillings. If the shillings group is even $2^{25} \cdot 12$ should be added but 2^{30} will be added. The excess is $20 \cdot 2^{25}$ representing 20 pence.

The corrections are thus 12 in the shillings group and 20 in the pence group. There are only two corrections, and as these are different (unlike the sixes and fifty four in every character) it is preferable to subtract them while division proceeds. To do this we inspect the remainders at the end of the £ and s conversions. If these are negative, then corrections are required in the s and d groups respectively.

The instructions are:



- NOTES:
- (a) First divisor for £ conversion.
 - (b) Binary pence to 21_3
 - (c) Start Division in m.c. m
 - (d) Second divisor to TS16 for shillings conversion.
 - (e) Test remainder after £ conversion to see if first operation with new divisor is an addition. The minor cycles of (d) and (a) impose certain restraints. If two D.L.'s are used ($2^{25} \cdot 12$) can be stored in m + 40 of one D.L. $21_3 - 27$ obeyed from m+40 of the other D.L. Otherwise $2^{25} \cdot 12$ will require to be transferred to a TS for transfer to TS16 in m.c. m+40.
 - (f) If (e) leads negative, subtracts $3P_1$ from the quotient. Done at this time, this is equivalent to subtracting 12 from the shillings group.
 - (g) Third divisor for pence to TS16.
 - (h) Test if first operation with new divisor is an addition.
 - (j) Subtract $5P_1$ as a correction if necessary. Equivalent to subtracting 20 from final position of pence group.

(k) Shift down one place to remove the spare P_1 position created by the conversion.

4.5 Conversion of Binary LB WT to TONS, CWT. QTRS. LB.

This conversion was required for an N.R.L., programme for which an acceptable character layout is as shown:

3	5	2	4	1	4	4	4	4	1
Spare	LB WT	QTRS.	CWT.	10 CWT.	10^0 TONS	10^1 TONS	10^2 TONS	10^3 TONS	SPARE
	L_1	Q_1	C_2	C_1	T_4	T_3	T_2	T_1	

- P_1 - 3 Spare.
- P_4 - 8 LB WT
- $P_9, 10$ Quarters.
- $P_{11} - 14$ UNIT CWT.
- P_{15} 10^1 CWT.
- $P_{16} - 19$ 10^0 TONS.
- $P_{20} - 23$ 10^1 TONS.
- $P_{24} - 27$ 10^2 TONS.
- $P_{28} - 31$ 10^3 TONS.
- P_{32} Spare.

We require constants as follows

$$2^{31}N = 2^{27} \cdot T_1 (10^3 \cdot 2240 \cdot 2^a) + 2^{23} \cdot T_2 (10^2 \cdot 2240 \cdot 2^b) + 2^{19} (T_3 (10^1 \cdot 2240 \cdot 2^c) + 2^{15} \cdot T_4 (10^0 \cdot 2240 \cdot 2^d) + 2^{14} c_1 (1120 \cdot 2^e) + 2^{10} \cdot c_2 (112 \cdot 2^f) + 2^8 \cdot Q_1 (28 \cdot 2^g) + 2^3 L_1 (1 \cdot 2^h))$$

where, to permit 2^{31} to cancel throughout,

$a + 27 = 31$	therefore	$a = 4$
$b + 23 = 31$		$b = 8$
$c + 19 = 31$		$c = 12$
$d + 15 = 31$		$d = 16$
$e + 14 = 31$		$e = 17$
$f + 10 = 31$		$f = 21$
$g + 8 = 31$		$g = 23$
$h + 3 = 31$		$h = 28$

The divisors required are:

$$2^4 \cdot 10^3 \cdot 2240$$

$$2^8 \cdot 10^2 \cdot 2240$$

$$2^{12} \cdot 10^1 \cdot 2240$$

$$2^{16} \cdot 10^0 \cdot 2240$$

$$2^{17} \cdot 1120$$

$$2^{21} \cdot 112$$

$$2^{23} \cdot 28$$

$$2^{28} \cdot 1$$

Because of the binary relation between 10 Cwt. and 1 Ton and between quarters and Cwt., the bracketed constants are equal and only six constants are needed. The next task is the calculation of correction terms. For the T_2 , T_3 and T_4 groups the correction term is 6. For the C_2 group the correction is

$$\begin{aligned} 2^{21} \cdot 112 - 2^{17} \cdot 1120 &= 2^{17} \cdot 112 (16-10) \\ &= 6 \cdot 112 \cdot 2^{17} \end{aligned}$$

i.e. 6 Cwt.

For the L_1 group the correction is

$$2^{28} - 2^{23} \cdot 28 = 2^{23} \cdot 4$$

i.e. 4 LB.

All correction terms are 6 except for the L_1 term which is 4. This can be treated separately and a correction word formed for the other groups. The method of dealing with the '4' correction is to examine the bottom digit of the quarters group. If this is zero we subtract a P_1 from 21_2 while the division proceeds. The minor cycle in which this is done is chosen so that the subtraction is effectively 4 from the final position of the Lb WT group.

The instruction sequence is:

- (a) $(2^4 \cdot 10^3 \cdot 2240) - 16$
- (b) $N - 21_3$
- (c) $1 - 24$ (m.c. m)
- (d) $(2^8 \cdot 10^3 \cdot 2240) - 16$ m.c. (m+8 or 9)
- (e) $(2^{12} \cdot 10^1 \cdot 2240) - 16$ m.c. (m+16 or 17)
- (f) $(2^{16} \cdot 10^0 \cdot 2240) - 16$ m.c. (m+24 or 25)
- (g) $27 - 15$
- (h) $(2^{21} \cdot 112) - 16$ m.c. (m+34 or 35)
- (j) $2^{28} - 16$ m.c. (m+46 or 47)
- (k) $22_2 - 14$ m.c. m+19
- (l) $25 - 14$ m.c. m+21
- (m) $26 - 23_2$ m.c. m+23
- (n) $21_2 - 14$ m.c. (m+27)
- (p) $(P_{12,17,21,25}) - 15$
- (q) $25 - 14$
- (r) $26 - 14$
- (s) $14 - 23_2$
- (t) $24 - 23_2$

Result in 21_2

- NOTES:
- (a) First divisor to TS16.
 - (b) Binary LB WT to 21_3
 - (c) Division starting in m.c. m.
 - (d) to (f) Successive divisors to TS16.
 - (g) P_1 to 15 anticipating (l) and (m)
 - (h) and (j) More divisors to TS16
 - (k) Extract (S 22) quotient immediately after bottom digit of quarters group is formed.
 - (l) Form P_1 in 14 if quarters group is ODD.
 - (m) Subtract P_1 from quotient if quarters group is EVEN. This subtraction takes place when the fours digit of the LB group has just been formed and effects the correction of 4 from the LB group.

Z.501/23 NELSON RESEARCH LABORATORIES
 STAFFORD E. E. CO. LTD.
 MATHEMATICS DEPARTMENT.

Continuation to: NS y 141
 Sheet No.: 28.

- (n) Extract quotient when bottom digits of T_1, T_2, T_3
 and C_1 groups are in P_{12}, P_{17}, P_{21} and P_{25} positions.
- (p), Form digits in $P_{12}, P_{17}, P_{21}, P_{25}$ if least significant
 (q), digits of corresponding groups are EVEN.
 (r)
- (s), Apply '6' corrections where required.
 (t)

Published subroutines which make use of D21 and S21
 during division operations include D19/2 and P22T.
 Those interested should consult the appropriate reports.
 P22T uses the divider to convert two binary half words
 ($< 10^4$) into 4-bit binary coded decimal, two converted
 numbers to a word.

APPENDIX 1.OPERATION OF THE DEUCE MULTIPLIER.

The initial conditions for normal multiplication are

- (a) Clear 21_2
- (b) Multiplier to 21_3
- (c) Multiplicand to TS16.
- (d) Start multiplication in an ODD minor cycle.

The top (P_{32} position) digits of the word in 21_3 are examined at the end (P_{32} time) of each ODD minor cycle. If the top digit is a zero nothing is added in the following even minor cycle to the contents of 21_2 shifted up one place. If the top digit is a one, this one is rubbed out and becomes a zero and the contents of TS16 are added to the contents of 21_2 shifted up one place, in the following even minor cycle.

Denoting the n th partial product in 21_2 by X_n , the $(n+1)^{th}$ partial product is

$$X_{n+1} = 2X_n + a_{33-(n+1)}B, \left[\begin{array}{l} a_{33-(n+1)} = 0 \\ \text{or } 1 \end{array} \right]$$

Where B is the multiplicand and the digits of the original multiplier are $a_1 a_2 \dots a_{32}$ with a_1 in the least significant position.

As the contents of 21_2 emerge in any even minor cycle they are shifted up one place before passing the addition circuits and re-entering 21_2 .

The following table shows the minor cycles in which the various digits of 21_3 are examined together with the words emerging from and re-entering 21_2 .

The multiplier is $a_1 \cdot a_2 \cdot a_3 \dots a_{31} \cdot a_{32}$ and the multiplicand is b . Multiplication starts in minor cycle m .

TABLE - APPENDIX 1 (MULTIPLIER)

Minor Cycle m +	Digit of 21 ₃ examined	Emerging from 21 ₂ (S21 ₂) ²	Re-entering 21 ₂	Minor Cycle m +	Digit of 21 ₃ examined	Emerging from 21 ₂ (S21 ₂) ²	Re-entering 21 ₂
0	a ₃₂			32	a ₁₆		
1		zero	0+a ₃₂ b=X ₁	33		2X ₁₆	2X ₁₆ +a ₁₆ b=X ₁₇
2	a ₃₁			34	a ₁₅		
3		2X ₁	2X ₁ +a ₃₁ b=X ₂	35		2X ₁₇	2X ₁₇ +a ₁₅ b=X ₁₈
4	a ₃₀			36	a ₁₄		
5		2X ₂	2X ₂ +a ₃₀ b=X ₃	37		2X ₁₈	2X ₁₈ +a ₁₄ b=X ₁₉
6	a ₂₉			38	a ₁₃		
7		2X ₃	2X ₃ +a ₂₉ b=X ₄	39		2X ₁₉	2X ₁₉ +a ₁₃ b=X ₂₀
8	a ₂₈			40	a ₁₂		
9		2X ₄	2X ₄ +a ₂₈ b=X ₅	41		2X ₂₀	2X ₂₀ +a ₁₂ b=X ₂₁
10	a ₂₇			42	a ₁₁		
11		2X ₅	2X ₅ +a ₂₇ b=X ₆	43		2X ₂₁	2X ₂₁ +a ₁₁ b=X ₂₂
12	a ₂₆			44	a ₁₀		
13		2X ₆	2X ₆ +a ₂₆ b=X ₇	45		2X ₂₂	2X ₂₂ +a ₁₀ b=X ₂₃
14	a ₂₅			46	a ₉		
15		2X ₇	2X ₇ +a ₂₅ b=X ₈	47		2X ₂₃	2X ₂₃ +a ₉ b=X ₂₄
16	a ₂₄			48	a ₈		
17		2X ₈	2X ₈ +a ₂₄ b=X ₉	49		2X ₂₄	2X ₂₄ +a ₈ b=X ₂₅
18	a ₂₃			50	a ₇		
19		2X ₉	2X ₉ +a ₂₃ b=X ₁₀	51		2X ₂₅	2X ₂₅ +a ₇ b=X ₂₆
20	a ₂₂			52	a ₆		
21		2X ₁₀	2X ₁₀ +a ₂₂ b=X ₁₁	53		2X ₂₆	2X ₂₆ +a ₆ b=X ₂₇
22	a ₂₁			54	a ₅		
23		2X ₁₁	2X ₁₁ +a ₂₁ b=X ₁₂	55		2X ₂₇	2X ₂₇ +a ₅ b=X ₂₈
24	a ₂₀			56	a ₄		
25		2X ₁₂	2X ₁₂ +a ₂₀ b=X ₁₃	57		2X ₂₈	2X ₂₈ +a ₄ b=X ₂₉
26	a ₁₉			58	a ₃		
27		2X ₁₃	2X ₁₃ +a ₁₉ b=X ₁₄	59		2X ₂₉	2X ₂₉ +a ₃ b=X ₃₀
28	a ₁₈			60	a ₂		
29		2X ₁₄	2X ₁₄ +a ₁₈ b=X ₁₅	61		2X ₃₀	2X ₃₀ +a ₂ b=X ₃₁
30	a ₁₇			62	a ₁		
31		2X ₁₅	2X ₁₅ +a ₁₇ b=X ₁₆	63		2X ₃₁	2X ₃₁ +a ₁ b=X ₃₂
				64	zero		

The use of S21, S22, D21, D22, D23 and D16 during multiplication is governed by the following rules.

- (a) If X_n enters 21_2 in any even minor cycle then S21 gives X_n in the next even minor cycle ~~and S22 gives X_n~~
- (b) If $D21_2$ is used during multiplication, the word sent to $D21_2$ replaces the word emerging from $S21_2$.
e.g. normally in m.c. $m+11$, $2X_5+a_{27}b$ enters 21_2 if Y is sent to $D21_2$
in m.c. $m+11$, $Y+a_{27}b$ enters 21_2 .
- (c) $D22_2$ and $D23_2$ can be used provided that no additions from TS16 occur in that same minor cycle.
e.g. in m.c. $m+11$ if a_{27} is zero and Y is sent to $D22_2$ (or $D23$)
then $2X_5+Y$ enters 21_2 .
- (d) $D22_3$ can be used during multiplication.
- (e) $D23_3$ can not be used for a subtractive transfer starting in an odd minor cycle, though it can be used as the odd minor cycle in a sequence (c, o, e, o)
- (f) The content of TS16 can be changed during a multiplication. If a transfer is made to D16 in an even minor cycle, the previous content of TS16 is added (if at all) to 21_2 in the minor cycle of transfer. The first minor cycle in which a new multiplicand is added to the contents of 21_2 is the even minor cycle following the transfer minor cycle, whether this is even or odd.

APPENDIX II.

OPERATION OF THE DEUCE DIVIDER.

The initial conditions for a division operation are:

- (a) DIVISOR B to TS16.
- (b) DIVIDEND A to 21₃.
- (c) DIVISION started in an ODD minor cycle.

The divider produces a quotient Q₃₁ and remainder R₂₁ defined by

$$2^{31}A = Q_{31}B + R_{31} \text{ where } 0 \leq R_{31} < B \quad B > 0$$

$$B \leq R_{31} < 0 \quad B < 0$$

and R₃₁ is obtained from the machine remainder r by the process:

$$R_{31} = \frac{1}{2} (\frac{1}{2}r + B)$$

The divider compares the signs of the content of 21₃ and TS16 at the end of every odd minor cycle. If the signs are the same 16 is subtracted from 21₃ in the next odd minor cycle; if the signs are different 16 is added to 21₃ in the next odd minor cycle.

In the even minor cycle after an odd minor cycle in which a subtraction occurs a one is inserted into the P₁ position of 21₂. If an odd minor cycle is an addition minor cycle, no change is made to 21₂. The contents of 21₂ and 21₃ are shifted up one place in successive even and odd minor cycles respectively and the quotient scoring P₁ digits which are inserted into 21₂ actually enter 21₂ as P₂ digits because the shift occurs between the point at which they are inserted² and the entry to DS21.

The simplest way of looking at divider operations is to realise that in successive pairs of minor cycles the divider is producing quotients Q_n and R_n which satisfy

$$2^0 A = Q_0 B + R_0$$

$$2^1 A = Q_1 B + R_1$$

$$2^2 A = Q_2 B + R_2$$

$$2^3 A = Q_3 B + R_3$$

$$\vdots$$

$$2^{29} A = Q_{29} B + R_{29}$$

$$2^{30} A = Q_{30} B + R_{30}$$

$$2^{31} A = Q_{31} B + R_{31}$$

The method of scoring only the subtractions means that the quotients lag behind the corresponding remainders and the first digit inserted in 21₂ is spurious. This digit drops off the end and the programmer is not normally² aware of its existence. If however the contents of 21₂ are extracted during a division this spurious digit (and any rubbish left in 21₂ before division starts) must be borne in mind. It is for this reason that 30-21₂ occurs after 1-24 in some of the routines described in the report.

The following table shows the minor cycles in which Q_n and R_n are produced. It will be noticed that the final remainder is $2R_{32}$. A shift down gives R_{32} now $R_{32} = 2R_{31} - B$ therefore adding B and shifting down gives R_{31} .

This is the remainder appropriate to Q_{31} . The divider may give $(Q_{31} - 1)$ as quotient. In this case $R_{32} = 2R_{31} + B$. Adding B and shifting down gives $(R_{31} + B)$ as remainder. This is appropriate to $Q_{31} - 1$ since $2^{31}A = (Q_{31} - 1)B + (R_{31} + B)$.

Minor Cycle $m+$	Entering		Emerging From		Entering		Emerging From	
	21_3	21_2	21_3	21_2	21_3	21_2	21_3	21_2
0	$2R_0$				32	$2R_{16}$	$2R_{15}$	
1		Spurious XP_2		Previous Content.	33		$2Q_{15}$	$2Q_{14}$
2	$2R_1$		$2R_0$		34	$2R_{17}$	$2R_{16}$	
3		$2Q_0$		Spurious XP_2	35		$2Q_{16}$	$2Q_{15}$
4	$2R_2$		$2R_1$		36	$2R_{18}$	$2R_{17}$	
5		$2Q_1$		$2Q_0$	37		$2Q_{17}$	$2Q_{16}$
6	$2R_3$		$2R_2$		38	$2R_{19}$	$2R_{18}$	
7		$2Q_2$		$2Q_1$	39		$2Q_{18}$	$2Q_{17}$
8	$2R_4$		$2R_3$		40	$2R_{20}$	$2R_{19}$	
9		$2Q_3$		$2Q_2$	41		$2Q_{19}$	$2Q_{18}$
10	$2R_5$		$2R_4$		42	$2R_{21}$	$2R_{20}$	
11		$2Q_4$		$2Q_3$	43		$2Q_{20}$	$2Q_{19}$
12	$2R_6$		$2R_5$		44	$2R_{22}$	$2R_{21}$	
13		$2Q_5$		$2Q_4$	45		$2Q_{21}$	$2Q_{20}$
14	$2R_7$		$2R_6$		46	$2R_{23}$	$2R_{22}$	
15		$2Q_6$		$2Q_5$	47		$2Q_{22}$	$2Q_{21}$
16	$2R_8$		$2R_7$		48	$2R_{24}$	$2R_{23}$	
17		$2Q_7$		$2Q_6$	49		$2Q_{23}$	$2Q_{22}$
18	$2R_9$		$2R_8$		50	$2R_{25}$	$2R_{24}$	
19		$2Q_8$		$2Q_7$	51		$2Q_{24}$	$2Q_{23}$
20	$2R_{10}$		$2R_9$		52	$2R_{26}$	$2R_{25}$	
21		$2Q_9$		$2Q_8$	53		$2Q_{25}$	$2Q_{24}$
22	$2R_{11}$		$2R_{10}$		54	$2R_{27}$	$2R_{26}$	
23		$2Q_{10}$		$2Q_9$	55		$2Q_{26}$	$2Q_{25}$
24	$2R_{12}$		$2R_{11}$		56	$2R_{28}$	$2R_{27}$	
25		$2Q_{11}$		$2Q_{10}$	57		$2Q_{27}$	$2Q_{26}$
26	$2R_{13}$		$2R_{12}$		58	$2R_{29}$	$2R_{28}$	
27		$2Q_{12}$		$2Q_{11}$	59		$2Q_{28}$	$2Q_{27}$
28	$2R_{14}$		$2R_{13}$		60	$2R_{30}$	$2R_{29}$	
29		$2Q_{13}$		$2Q_{12}$	61		$2Q_{29}$	$2Q_{28}$
30	$2R_{15}$		$2R_{14}$		62	$2R_{31}$	$2R_{30}$	
31		$2Q_{14}$		$2Q_{13}$	63		$2Q_{30}$	$2Q_{29}$
					64	$2R_{32}$	$2R_{31}$	
					65		Q_{31}	$2Q_{30}$

NOTE: At all stages except the last the contents of 21_2 is $2Q_n$ (headed by the spurious digit).

At the last stage the spurious digit has been pushed off the end of 21_2 and the content of 21_2 is shifted down one place to give Q_{31} (T.C.B. is on)