Bendix Computer

Copy No.

Report No.

MAINTENANCE MANUAL
DIGITAL DIFFERENTIAL ANALYZER
MODEL D-12

Bendix Computer

DIVISION OF BENDIX AVIATION CORPORATION

5630 ARBOR VITAE STREET, LOS ANGELES 45, CALIFORNIA

COPY NO.

FEBRUARY, 1956

MAINTENANCE MANUAL

DIGITAL DIFFERENTIAL ANALYZER

BENDIX COMPUTER
Division of Bendix Aviation Corporation
5630 Arbor Vitae Street, Los Angeles 45, California

CONTENTS

OPERATIONAL LOGIC

ARITHMETIC UNIT

ADDRESS UNIT

OUTPUT UNIT

CONTROL UNIT

FILL UNIT

READ OUT UNIT

PLOTTER UNIT

OPERATIONAL LOGIC

The "logic of the computer" is the interconnection of the electronic elements in the computer. The logic will be analyzed in terms of the patterns of states which will be developed in these elements, which are flip-flops, inverters, buffers and diode gates.

1.1 Flip-flops

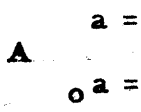
A flip-flop is a bistable electronic device; its two possible states are designated as on and off. Associated with the flip-flop are two outputs called true and false. When the flip-flop is in the "on" state, the true output is high and the false output low and conversely when in the off state, the true output is low and the false output high.

The two inputs have designations similar to the respective outputs, that is, the input which, when high, causes the true output to become high is called the true input and the input which, when high, causes the false output to become high is called the false input.

Two conditions are necessary to cause a flip-flop to change state. First, the proper input must be high and second a clock signal must occur. All flip-flops in the computer are synchronized by means of a clock and the frequency of the clock signal determines the speed of computation. Because of the clock signal, a lag is introduced between the time the input becomes high and the time a change in the output is effected. Let P_i denote the time between a given clock signal, C_i, and the next, C_{i+1}. If the logic for a flip-flop that is off is so arranged that during P_i the true input is high, then the flip-flop is turned on at C_{i+1}.

Once set in a certain state a flip-flop will remain in that state until the opposite input becomes high even if the signal which put the flip-flop in its original state is removed. A flip-flop, therefore, is sometimes referred to as a binary memory or storage device.

Notationally, a flip-flop A is represented by



where "A" indicates that the two outputs will be designated "A" and "A'" and "a" and "$_o$a" represent the true and false inputs respectively.

1.2 Inverters

An inverter has a single input, a single output, and, as the name implies, the latter is the inverse of the former. Unlike a flip-flop, the effect of a change in input is instantaneously transferred to the output, hence an inverter cannot act as a storage device.

1.3 Buffers

A buffer is exactly analogous to an inverter except that its output is the same as its input. Since it introduces no change in the signals already present in the computer, the only reasons for its introduction are to separate circuits and to serve as a power amplifier.

1.4 Diode Gates

The inputs to the elements discussed above are made up of combinations of outputs of similar elements. These combinations, formed in diode gates, are represented by sentences of the propositional calculus. Each symbol stands for an output whose two possible values, high and low, are represented by 1 and 0 respectively. The truth tables for the basic operations may now be given.

1.4.1 Negation

Given a signal A, it may be high or low; the table states under what condition its negation, not-A, symbolized "A'", will be high and low.

A	A'
1	0
0	1

Note that if the true and false outputs of a flip-flop are respectively A and A', then the negation of one output is equivalent to the other.

1.4.2 Disjunction

Given two signals A and B, the table states under what conditions of A and B the proposition "A or B" will be high and under what conditions it will be low. The "or" is here used in the inclusive sense and is symbolized by "+".

A	B	A+B
1	1	1
1	0	1
0	1	1
0	0	0

1.4.3 Conjunction

Given two signals A and B, the table states under what conditions of A and B the proposition "A and B" will be high and under what conditions it will be low. The "and" is symbolized by placing the two terms side by side.

A	B	AB
1	1	1
1	0	0
0	1	0
0	0	0

Some of the fundamental theorems of the propositional calculus may now be stated.

Associativity	(AB)C = A(BC) = ABC	(a)	I
Distributivity	A(B+C) = AB + AC	(b)	
Commutivity	AB = BA	(a)	II
	A+B = B+A	(b)	
DeMorgans Law	(A+B)' = A'B'	(a)	III
	(AB)' = A'+B'	(b)	

$1 = A+A'$	(a)	$O = AA'$	(b)	IV
$A1 = A$	(a)	$AO = O$	(b)	V
$A+1 = 1$	(a)	$A+O = A$	(b)	VI
$A+A'B = A+B$	(a)	$A+AB = A$	(b)	VII
		$AA = A$	(a)	VIII
		$A+A = A$	(b)	

As an example of the construction of diode gates, consider the case in which a gate B_2 is required having the characteristics of the following table.

	A_3	A_2	A_1	B_2	Case
Case 1	0	1	1	0	1
2	1	0	0	1	2
3	1	0	1	1	3
4	0	1	0	0	4

The problem is to construct a combination of the A's which will be high in Cases 2 and 3, and another which will be high in Cases 1 and 4. That is, B_2 and B_2' respectively. The formal approach is to set each of these equal to the disjunction of A signals that are to be true if B_2 (or B_2') is true plus the negation of A signals that are to be false. Hence,

$$B_2 = A_3 A_2' A_1' + A_3 A_2' A_1 + (A_3' A_2 A_1 + A_3' A_2 A_1')'$$

The right-hand side of the equation is now simplified by applying the Theorems stated before.

 $(A+B)' = A'B'$

$$B_2 = A_3 A_2' A_1' + A_3 A_2' A_1 + (A_3' A_2 A_1)' (A_3' A_2 A_1')' \quad \text{(III)}$$

$$= A_3 A_2' A_1' + A_3 A_2' A_1 + (A_3 + A_2' + A_1')(A_3 + A_2' + A_1) \quad \text{(III)}$$

$$B_2 = A_3 A_2' A_1' + A_3 A_2' A_1 + A_3 A_2' + A_3 A_1 + A_2' A_3$$

$$+ A_2' A_1 + A_1' A_3 + A_1' A_2' + A_3 + A_2' \quad \text{(I, IV VIII)}$$

$$= A_2' + A_3 \quad \text{(VII b)}$$

A similar procedure is employed to derive B_2'.

The method just outlined has two drawbacks, however: the final equation may not be the simplest in terms of the amount of equipment it requires, and, if the tables are larger, the method entails a prohibitive amount of time.

A less formal method consists in directly examining the tables for regularities. Thus, in determining B_2' it is evident that the A_2 and A_1 terms are sufficient since the four combinations of A_1 and A_2 all occur. B_2', then, may be set equal to $A_2 A_1 + A_2 A_1'$. But this is equal to A_2. Similarly, an examination will show that

$$B_2' = A_3'$$

and, in the case discussed above,

$$B_2 = A_2'$$

$$\text{or} \quad B_2 = A_3$$

1.5 Number System and Arithmetic

In order to understand the addition logic of the computer it is helpful to have some acquaintance with the characteristics of the binary number system. The symbols of this system are 0 and 1; the weight of any 1 being determined by its position. Starting at the right of a number, these weights are 1, 2, 4, 8, 16, etc., that is, 2^{n-1} where n is the ordinal number of the position with n = 1 at the right end. Thus

$$
\begin{aligned}
01001101 &= (1\times 2^{0}) + (0\times 2^{1}) + (1\times 2^{2}) + \\
&\quad (1\times 2^{3}) + (0\times 2^{4}) + (0\times 2^{5}) + \\
&\quad (1\times 2^{6}) + (0\times 2^{7}) \\
&= 1+0+4+8+0+0+64+0 \\
&= 77
\end{aligned}
$$

An examination of this weighting scheme will show that the greatest decimal number representable by n binary places is $2^{n}-1$.

Addition of binary numbers, like that of decimal numbers, may be carried out serially, digit by digit. At each step in the addition of two numbers, N_1 and N_2, three digits must be considered: the respective digits of N_1 and N_2 and the carry which may have resulted from the addition in the previous place. The table for determining the sum, S_n, and the carry, C_n, in any given place, n, is:

N_1	N_2	C_{n-1}	S_n	C_n
0	0	0	0	0
1	0	0	1	0
0	1	0	1	0
1	1	0	0	1
0	0	1	1	0
1	0	1	0	1
0	1	1	0	1
1	1	1	1	1

Subtraction, in binary, decimal, or any number system, may be performed in two ways. The first is by constructing a table defining the difference corresponding to any two digits in the system and another specifying the carries between digit pairs. The second scheme utilizes

the definitions for addition but changes the subtrahend to its complement. The complement of a number "A" in a number system with radix R is $R^i - A$, where i is the number of digits in the register which holds the result. For example, if R = 10, i = 3, A = 7, then the complement of "A" is 1000 -7 = 993. That 993 will operate in addition like -7 may be seen by adding, say, +9 to it. Since 9 = +7+2 we may consider the + and - sevens to cancel and +2 to remain which is the case since 993 + 9 = 002.

The carry digit out of the left most place is lost because the register has only three places. Construction of complements requires the introduction of a partial table of differences for specifying the difference between R-1 and the remaining digits in the number system. The complement of a number A_1, represented by as many digits as there are in the register, is produced by subtracting each digit of "A" from R-1 and adding 1 to the least significant place of the result. This 1 must be added since the minuend is R^i-1 rather than R^i.

In order to operate correctly on the signs of numbers it is convenient to add an extra place to the left-hand end of the register, although, in general, this place need not take on all R values. If a zero is held in this place for positive numbers, the complementation process will result in an R-1 for negative numbers. If the assumption is made that all computation remains within the limits of i places exclusive of sign, then the correctness of the scheme may be seen as follows. If the two numbers being added are positive, the i+1 place will always be zero. If one is negative and one is positive, lack of a carry into the i+1 place indicates that the negative number is larger; hence, the result in the i+1 place is R-1 since the result is negative. A carry into the i+1 place indicates the positive number is largest and places a zero, (R-1)+1, in that place, the carry out of the i+1 place being ignored. Finally, if both numbers are negative, a carry must eventuate if the number is to stay within the bounds of the register, hence, the i+1 place has

$$1+(R-1)+(R-1)-R = R-1.$$

The initial 1 is the input carry, the final R the output carry.

Examples in which R = 10 and i = 5 will clarify the scheme:

+87665	=	087665
-73413		926586
		1
		014252 = +14252

+18639	=	018639
- 154		999845
		1
		018485 = +18485

The system is particularly apt for R = 2, since R-1 = 1, 1-0 = 1 and 1-1 = 0. This means that complementation proceeds by changing 1 to 0 and 0 to 1 and adding one on the right. The following examples of subtracting by complementation will illustrate the case of R = 2 and i = 6:

+37	+100101	=	0100101
-50	-110010		1001101
			1
			1110011 = -(+64-51) = -13

35	+0100011	=	00100011
-10	- 1010		11110101
			1
			00011001 = +25

1.6 Binary Coded Decimals

The binary number system may be employed to represent decimal numbers by using four binary digits for each decimal digit. Four must be used because it is the smallest n for which $2^n \geqslant 10$. If the normal binary

system is used, the following table shows the binary combination corresponding to each decimal digit; complements being used for negatives.

		0	0000	5	0101
-1	1111	1	0001	6	0110
-2	1110	2	0010	7	0111
-3	1101	3	0011	8	1000
etc.		4	0100	9	1001

Addition must now be modified in two ways: carries from one group of four binary digits must be sensed at 10 rather than at 16, and, in the event of a carry, the sum must be modified. An example will clarify the procedure.

	1	1		
763	0111	0110	0011	
+158	0001	0101	1000	
	1001	1100	1011	- First Addition
921	1001	0010	0001	Correct Sum

Carries result from the two decimal additions on the right since both numbers are greater than 10, and both of these must be corrected by adding 6 (16 - 10). Another method for representing decimal digits with binary numbers is the so-called "excess-three" code. This is the code employed by the computer. Note that the rules for complementation are valid for this code.

1101		-0
1100	+9	-1
1011	+8	-2
1010	+7	-3
1001	+6	-4
1000	+5	-5
0111	+4	-6
0110	+3	-7
0101	+2	-8
0100	+1	-9
0011	+0	

To arrive at the excess-three code from the binary code, add three (0011 in binary) to the binary code if the number is positive, subtract 3 (by adding 1101, the binary complement of 3) if the number is negative.

$$\begin{aligned} \text{Thus } +7 &= 0111 \\ &\quad +0011 \\ &\quad \overline{1010} \end{aligned}$$

$$\begin{aligned} \text{while } -7 &= 1001 \\ &\quad 1101 \\ &\quad \overline{0110} \end{aligned}$$

The advantage of the code is that the normal binary carry out of the most significant binary digit also serves as the carry between decimal digits.

Addition and subtraction differ from normal binary addition and subtraction only in that a correction factor must be made on the first sum in order to give the true sum. The determination of the correction factor requires the consideration of six cases given by the following chart.

Numbers Being Added	No Carry (Add -3)	Carry (Add)
both +	I	IV
one + one -	II	V
both -	III	VI

CASE I

Both numbers X_1, X_2 are of the form

$$X = N + 3$$

where N is the value of the digit represented.

$$\begin{array}{rcl} X_1 & = & N_1 + 3 \\ + \ X_2 & & + N_2 + 3 \\ \hline & & N_1 + N_2 + 6 \end{array} \qquad (0 \leq N_1 + N_2 \leq 9 \text{ since there is no carry})$$

The correction factor is -3, since X_3 is required to have the form $N_1 + N_2 + 3$.

CASE I I

Here X_2 is a negative number and, therefore, it is in a complemented form,

$$16 - N_2 - 3 = 13 - N_2.$$

The addition is in the form

$$\begin{array}{rcl} X_1 & = & N_1 + 3 \\ - \ X_2 & & - N_2 + 13 \\ \hline & & N_1 - N_2 + 16 \end{array} \qquad (9 \geqslant N_2 > N_1 \text{ since there is no carry})$$

The correct sum is a negative number with the form $13 + (N_1 - N_2)$ so that the correction factor is again -3.

CASE III

$$\begin{array}{rcl} - \ X_1 & = & 13 - N_1 \\ - \ X_2 & = & 13 - N_2 \\ \hline & & 26 - N_1 - N_2 \end{array} \qquad (18 \geqslant N_1 + N_2 \geqslant 11 \text{ since there is no carry})$$

Since $N_1 + N_2$ must be larger than 10, the correct form of X_3 is $13 - (N_1 + N_2 - 10)$, so that the correction factor is -3.

CASE IV

Before taking up the cases involving a carry, it should be noted that when a carry occurs it represents 16 in the binary code, consequently 16 must be subtracted from the sum.

$$\begin{array}{rcl} X_1 & = & N_1 + 3 \\ X_2 & & N_2 + 3 \\ \hline & & N_1 + N_2 + 6 - 16 \end{array} \qquad (18 \geqslant N_1 + N_2 \geqslant 10 \text{ since there is a carry})$$

Since $N_1 + N_2 > 10$, X_3 should be $(N_1 + N_2 - 10) + 3$ and the correction factor is +3.

CASE V

$$X_1 = N_1 + 3 \qquad (9 \geq N_1 \geq N_2)$$
$$X_2 = \frac{13 - N_2}{16 + N_1 - N_2 - 16}$$

X_3 is a positive number of the form $(N_1 - N_2) + 3$, so that the correction factor is +3.

CASE VI

$$X_1 = 13 - N_1 \qquad (10 \geq N_1 + N_2 \geq 0 \text{ since there is a carry})$$
$$X_2 = \frac{13 - N_2}{26 - (N_2 + N_1) - 16}$$

X_3, being a negative number, must be $13 - (N_2 + N_1)$, so that the correction is +3.

Reviewing all six cases, a procedure may now be outlined covering all additions in the excess-three code.

1) Add the two numbers in a purely binary fashion treating any carry as a normal "ten" carry.

2) If a carry occurs add +3 (0011) to the result in (1); if no carry occurs subtract 3 (add 1101) to the result in (1), ignoring any carrys between decimal digits.

That the scheme for subtraction by complementation still holds may be seen by considering two points. I. the complementation procedure is applied to the decimal digits individually, each digit is transformed into its complement.

For example

+4	0111	1000	
		1	
		1001	-4
+7	1010	0101	
		1	
		0110	-7

Once the coding scheme is validated, the invidual coded digits may be considered as decimal numbers; the validity of the system for decimals was shown above.

Examples of the complete scheme follow:

EXAMPLE I

+ 1346
- 109
=

+	0100	0110	0111	1001		(1)
-		0100	0011	1100		
0011	0100	0110	0111	1001	Addend	(2) a
1100	1100	1011	1100	0011	Augend	
1 ← 1	← 1	← 1	← 0	← 1	Carry	
0000	0001	0010	0011	1101		
0011	0011	0011	0011	1101		(2) b
0011	0100	0101	0110	1010	= +1237	

(1) shows the numbers in excess-three code as the absolute value and sign. In (2)a the addend remains unchanged except that its sign is coded, while the augend, being negative, is complemented. The carry in the right most addition provides the extra "one" in complementation. The correction factor is determined by the carry "out" of each decimal digit.

EXAMPLE II

(- 673)					
-(- 102)	=				
	- 1001	1010	0110		(1)
	-- 0100	0011	0101		
	1100 0110	0101	1010		(2)
	-1100 1011	1100	1011		
	1100 0110	0101	1010		(3)
	0011 0100	0011	0100		
0	0 0	0	1		
	1111 1010	1000	1111		
	1101 1101	1101	1101		
	1100 0111	0101	1100	=	-(1000 - 429)
				=	- 571

In (1) the numbers are shown as absolute value and sign. Both addend and augend are transformed into complements in (2). The remaining minus sign is accounted for by transforming the augend again and the addition proceeds in (3).

2.0 ARITHMETIC UNIT

Figure III is a logical schematic of the Arithmetic Unit showing the sequence of operation in the arithmetic channels, 4, 3, 2, 1. These are channels on the magnetic drum so arranged that each integrator is read, new information is recorded, and, after a delay of 60 integrators, the new information is read. In this manner all sixty integrators are serially processed.

The grouping of the information read from the magnetic drum is indicated in figure II. This figure shows the information read from the drum during integrator 59 and 00 times in the sequence of 38 pulse times indicated. At 0_2 times (pulse times 3, 7, 11, 15, 19, 23, 27, 31, and 35) the digits of the Y register are read from the drum in channels 4, 3, 2, 1. 0_3, 0_4, and 0_1 times are similarly associated with the initial condition register (Y_i), the R register and the Y_D register respectively.

In figure III, the large squares stand for diode gates with operating characteristics given by the equations within the squares. The small squares represent vacuum tube circuits; those containing "b" are buffers, those containing "I" are inverters. All other letters indicate flip flops, the true signal being on the right in the direction of the arrows. When two inverters are shown together or a buffer with an inverter, the output of the first unit serves as the input to the second unit. R_4, R_3, R_2, R_1, are the read flip flops for channels 4, 3, 2, 1 respectively, and hold the information shown in the top part of figure II at the pulse times indicated, R_4 holding the most significant binary digit of each decimal digit.

The adder proper occupies the upper part of figure III and contains, in addition to the excess-three decimal adder, a gate (G6) for inserting information into the arithmetic channels and a gate (G2) for resetting

initial conditions.

The output and overflow signals are developed in the unit as well as the activate signals R_e and R_f.

2.1 Activate Signals

One of the advantages of the excess-three code is that at least one binary one appears in the code for every decimal digit, that is, 0000 is not used. It is, therefore, simple to detect the presence of numerical information in the arithmetic channels for the purpose of determining that point at which dy inputs are to be added. R_e goes on when the first digit appears, and remains on through P_{36}, the last position of the arithmetic channels that is used.

$$R_e = R_4 + R_3 + R_2$$

R_1 is not required in this equation as it will never be true unless at least one of the other R's is also true. R'_e, for electronic reasons, is developed rather than R_e.

Four pulse times after R_e goes on; R_f goes on at the end of the round-off digit group and, since the output of a flip flop lags its input by one clock pulse, R_f stays on through P_{37}. The "R'_f" term is included in r_f so that r_f itself may be used to identify the least significant digit of R at the end of the round-off digit group.

2.2 First Adder

The augend input for this adder $A_4 - A_1$ is either the information read from the memory channels 4-1 and held in the R flip flops or information from the Fill Unit (figure VIII) coming into G6 on the F lines. A signal F_o, also developed in the Fill Unit, controls which of the two sets of information is to be used as an input. The negation of certain signals (such as A_2 and A_4), rather than the signals themselves, are developed throughout the adder for electronic economy.

The addend input $B_4 - B_1$ is developed in G9. G (from the Fill Unit)

is a signal indicating ongoing computation as opposed to idling or filling. When R_e and G are both true, the number held in the L flip flops serves as the addend. The logic for these flip flops is in G16. The J lines (Address Unit) hold Dy_d, Dy, and D_I at 0_4, 0_1 and 0_2 times, while U_4 - U_1 at each 0_3 time hold the new Y_D. The logic in G19 determines if Y_D is entered directly or complemented in the L flip flops. X_e signifies that there is a non-zero dX while X_s denotes the sign of dX, X_s for +, X_s' for —. If dX = 0, G16 sets the L flip flops to 0011 at the end of 0_3 time. With this arrangement of inputs, again recalling that the output of a flip flop lags its input by one clock pulse, L_4 - L_1 contains Dy_d, Dy, DI and Y_D (dX) in 0_1, 0_2, 0_3 and 0_4 times respectively. Referring to figure II, it can be seen that the contents of the L flip flops are in the proper phase to be added to the contents of the R flip flops.

If either G or R_e is low, the four signals coming out of G9 are low. In this case

B_1' is low $\therefore$ B_1 is high
B_2' is low $\therefore$ B_2 is high
B_3 is low
B_4 is low

The augend is then 0011, or zero in the excess-three code. In this manner the arithmetic information will recirculate unchanged whenever there is no ongoing computation.

G4 contains the logic for the first parallel binary adder, which adds B_4 - B_1 to A_4 - A_1 giving the binary sum s_4 - s_1.

C_d	C_c	C_b	C_a	C_i
	A_4	A_3	A_2	A_1
	B_4	B_3	B_2	B_1
	s_4	s_3	s_2	s_1

The first sum term in the binary adder is:

$$s_1 = A_1 B_1 C_i + A_1 B_1' C_i' + A_1' B_1' C_i + A_1' B_1 C_i' \qquad (I)$$

and the carry term is

$$C_a = A_1 B_1 + A_1 C_i + B_1 C_i \qquad (II)$$

The former is transformed by noting that

$$A_1 B_1' C_i' = B_1' C_i' (A_1 + B_1 + C_i)$$

since

$$B_1' C_i' (A_1 + B_1 + C_i) = A_1 B_1' C_i' + B_1 B_1' C_i' + B_1 C_i C_i'$$

and contradictory terms may be omitted. Applying this to the last three disjunctive terms transforms (I) into

$$s_1 = A_1 B_1 C_i + (A_1 + B_1 + C_i)(B_1' C_i' + A_1' B_1' + A_1' C_i')$$

but the last parenthesis is equal to C_a', hence we have

$$s_1 = A_1 B_1 C_i + (A_1 + B_1 + C_i) C_a' = A_1 B_1 C_i + A_1 C_a' + B_1 C_a' + C_i C_a' + C_i C_a' \qquad (Ia)$$

An examination of G4 will show that, with appropriate subscripts, all terms have the form of (Ia) or (II) or their negation. Since the S flip flops are employed as delays rather than as memory devices, only one input to each is developed as an independent input. The second input to each flip flop is derived from the first by means of an inverter, causing the outputs to follow the inputs exactly except for the one pulse delay inherent in all flip flops.

$C_1 C_2 C_3 C_4$ is a four pulse time delay stepping register which delays C_d, the output carry from the binary addition, until the next appropriate decimal digit.

C_i, developed in G5, is the input carry to the least significant binary digit. $R_e R_f C_4$ is the normal input carry term and $r_f G X_e X_s'$ is the "1" that must be added in the subtraction by complementation scheme outlined above. C_4 is qualified by R_f to assure that no carry is entered from the previous integrator and by R_e to assure that no carries are

added at P_{37}. X_e X'_s indicate that dX = -1 while r_f designates the pulse time of the least significant digit of R, and G designates ongoing computation.

2.3 Second Adder

The augend for the second adder is held in the S_4 - S_1 flip flops, the addend in T_4 - T_1. If the output multiplier is one, M_I and M_{II} will both be low and G7 will be controlled solely by C_1. The T signals are in phase with the S signals (both are one pulse delayed from G4); consequently the two can be added together. The second adder, as indicated before, should have -3 for an addend if no carry resulted in the first addition, and +3 in the event of a carry. These, in binary form are, respectively,

0011 (Carry: C_1)
1101 (No Carry: C'_1)

Ignoring the P_{34} terms, which will be explained in connection with the output logic, the proper codes occur since

T'_1 is always low hence T_1 is high
T_2 is high when C_1 is high
T'_3 is high when T_2 is high, hence T_3 is high when C'_1 is high
T'_4 is high when C_1 is high, hence T_4 is high when C'_1 is high

The second adder is identical with the first except that it has no provisions for either an output or an input carry. The first carry term C_{a1} has, therefore, the form S_1T_1, and when negated becomes $S'_1 + T'_1$.

The output of the second adder is set up in the U_4 - U_1 flip flops two pulse times after the original augend appears in the R_4 - R_1 flip flops.

The initial condition reset is accomplished by G2. U_R is the normally high signal indicating that the output of the U_4 - U_1 flip flops is to be recorded on the drum. During initial condition reset, S_r is made true but only for 0_3 and 0_4 times ($Q_1R_y\overline{R_I}$ term of G14 of the Control Unit).

Also S_r being true causes U_r to be false. The information held in U_4-U_1 during 0_3 and 0_4 times corresponds to the original augend which appeared in R_4-R_1 during 0_1 and 0_2 times. It can be seen that to transfer the Y_i number (see figure II) into the Y and Y_D registers, it is only necessary to diminish the number of pulse times between read and record by one. This is accomplished by by-passing the U_4-U_1 flip flops during reset which in effect steps the four arithmetic channels to the right. This is done on two successive drum revolutions to allow Y_i to reach Y_D as well as the Y position.

S_R is also set high to shift the contents of Y into Y_D when dX = $\pm$ 1 (see gate 14 of the Control Unit, term 0_3 X_e G). This corresponds to the trapezoidal integration method described in the operation manual.

During initial condition reset, R_{II} - R_{I4} reset R to the correct round-off number in accordance with the output multiplier (G12 of Control Unit).

2.4 Determination of Correction Factor in Second Addition

At P_{34} time:

S_4 - S_1 holds the first sum of the addition of the last digits of R and Y_D (dX).

C_1 holds the carry out of the R + Y_D (dX) addition.

T_4 - T_1 holds the correction factor as determined by the output multiplier and the carry digit C_1.

M_I indicates an output multiplier of 2.

M_{II} indicates an output multiplier of 5.

2.4.1 Output multiplier = 1

In this case T_4 - T_1 holds the normal factor depending upon the carry C_1 (0011 or 1101).

2.4.2 Output Multiplier = 2

The value of the most significant digit of R, which is held at P_{33} in R_4-R_1, is limited in value to

0111 (+4)
0110 (+3)
0101 (+2)
0100 (+1)
0011 (0)

To this digit is added in the first adder the most significant digit of (Y_D and an input carry) (dX).

The codes which may be added to the most significant digit of R in the first adder are:

1101	+10	1101	-0
1100	+ 9	1100	-1
1011	+ 8	1011	-2
1010	+ 7	1010	-3
1001	+ 6	1001	-4
1000	+ 5	1000	-5
0111	+ 4	0111	-6
0110	+ 3	0110	-7
0101	+ 2	0101	-8
0100	+ 1	0100	-9
0011	+ 0		

The result is held in S_4 - S_1 and C_1 at P_{34} as is shown in the table on the following page.

The column labeled C_1 indicates the carry associated with each sum in S_4 - S_1. $\beta = S_3 S_4 + S_1 \cdot S_2 S_4$, generated by G8 and the accompaning inverter, is the term which indicates when a single non-zero output has occurred for an output multiplier of 2. T_4 - T_1 is shown to hold either -8, -3, +3 or +8 to obtain the proper second sum in U_4 - U_1 as shown. Z_n, the term developed by G11, indicates that a non-zero dZ output would occur if the output multiplier were 1 (see Section 2.5.1). When the first sum in S_4 - S_1 is between +0 and +4 in value, the correction term of 1101 or 0011 is the normal one determined by C_1. When the first sum is between +5 and +9, a dZ output of +1 occurs and a compensating 5 must be subtracted from the first sum as well as the excess-three subtraction to obtain the proper remainder. When the first sum is between -1 and -5 there is a dZ output of -1 and a compensating 5 as well as a 3 should be added to obtain the proper remainder. When the remainder is greater than +9 or less than -5, a dZ output of +2 or -2 is called for. Since such outputs are not provided for, Z_n serves to detect these cases and computation is stopped (see Section 2.7). The

		P_{34}												P_{35}				
		C_1	S_4	S_3	S_2	S_1	β	T_4	T_3	T_2	T_1		Z_n	dZ	U_4	U_3	U_2	U_1	
Addend Positive	(+14)	1	0	1	0	0	0						1	(+2)					
	(+13)	1	0	0	1	1	0						1	(+2)					
	(+12)	1	0	0	1	0	0						1	(+2)					
	(+11)	1	0	0	0	1	0						1	(+2)					
	(+10)	1	0	0	0	0	0						1	(+2)					
	(+9)	0	1	1	1	1	1	1	0	0	0	(-8)	0	+1	0	1	1	1	(+4)
	(+8)	0	1	1	1	0	1	1	0	0	0	(-8)	0	+1	0	1	1	0	(+3)
	(+7)	0	1	1	0	1	1	1	0	0	0	(-8)	0	+1	0	1	0	1	(+2)
	(+6)	0	1	1	0	0	1	1	0	0	0	(-8)	0	+1	0	1	0	0	(+1)
	(+5)	0	1	0	1	1	1	1	0	0	0	(-8)	0	+1	0	0	1	1	(+0)
	(+4)	0	1	0	1	0	0	1	1	0	1	(-3)	0	0	0	1	1	1	(+4)
	(+3)	0	1	0	0	1	0	1	1	0	1	(-3)	0	0	0	1	1	0	(+3)
	(+2)	0	1	0	0	0	0	1	1	0	1	(-3)	0	0	0	1	0	1	(+2)
	(+1)	0	0	1	1	1	0	1	1	0	1	(-3)	0	0	0	1	0	0	(+1)
	(+0)	0	0	1	1	0	0	1	1	0	1	(-3)	0	0	0	0	1	1	(+0)
Addend Negative	(+4)	1	0	1	0	0	0	0	0	1	1	(+3)	0	0	0	1	1	1	(+4)
	(+3)	1	0	0	1	1	0	0	0	1	1	(+3)	0	0	0	1	1	0	(+3)
	(+2)	1	0	0	1	0	0	0	0	1	1	(+3)	0	0	0	1	0	1	(+2)
	(+1)	1	0	0	0	1	0	0	0	1	1	(+3)	0	0	0	1	0	0	(+1)
	(+0)	1	0	0	0	0	0	0	0	1	1	(+3)	0	0	0	0	1	1	(+0)
	(-1)	0	1	1	1	1	1	1	0	0	0	(+8)	1	-1	0	1	1	1	(+4)
	(-2)	0	1	1	1	0	1	1	0	0	0	(+8)	1	-1	0	1	1	0	(+3)
	(-3)	0	1	1	0	1	1	1	0	0	0	(+8)	1	-1	0	1	0	1	(+2)
	(-4)	0	1	1	0	0	1	1	0	0	0	(+8)	1	-1	0	1	0	0	(+1)
	(-5)	0	1	0	1	1	1	1	0	0	0	(+8)	1	-1	0	0	1	1	(+0)
	(-6)	0	1	0	1	0	0						1	(-2)					
	(-7)	0	1	0	0	1	0						1	(-2)					
	(-8)	0	1	0	0	0	0						1	(-2)					
	(-9)	0	0	1	1	1	0						1	(-2)					

The circled entries in the table under $T_4 - T_1$ are those which differ from the normal operation of C_1 upon the correction term. In G7

$$T'_1 = P_{34}\ M_I\ \beta$$

and

$$T'_3 = T'_1 + T_2$$

so that T_1 and T_3 are set to these circled states when β is high.

2.4.3 Output Multiplier = 5

Here the most significant digit of R is limited to:

0100 (+1)

and

0011 (+0)

The codes which may be added to the most significant digit of R in the first adder are the same as for the multiplier of 2. The following table (shown on p. 24) is analogous to the previous one.

When the first sum in $S_4 - S_1$ is either +0 or +1, the correction term of 1101 or 0011 is the normal one determined by C_1. When the first sum is either +2 or +3, a dZ output of +1 occurs and a compensating 2 should be subtracted from the first sum as well as the excess 3. When the first sum is either -1 or -2, a dZ output of -1 occurs and a compensating 2 as well as a 3 must be added. First sums greater than +3 and less than -2 are prohibited since they call for multiple dZ outputs. If such outputs are indicated, computation is stopped (Section 2.7). The circled terms in the table are accounted for in G7 by the M_{II} terms of:

$$T_2 = C_1 + P_{34}\ M_{II}\ S'_3\ G$$
$$T'_3 = T'_1 + T_2$$
$$T'_4 = C_1 + P_{34}\ M_{II}\ \beta\ G$$

		P_{34}												P_{35}				
		C_1	S_4	S_3	S_2	S_1	β	T_4	T_3	T_2	T_1		Z_n	dZ	U_4	U_3	U_2	U_1
Addend Positive	(+11)	1	0	0	0	1	0						1	(+5)				
	(+10)	1	0	0	0	0	0						1	(+5)				
	(+ 9)	0	1	1	1	1	1						0	(+4)				
	(+ 8)	0	1	1	1	0	1						0	(+4)				
	(+ 7)	0	1	1	0	1	1						0	(+3)				
	(+ 6)	0	1	1	0	0	1						0	(+3)				
	(+ 5)	0	1	0	1	1	1						0	(+2)				
	(+ 4)	0	1	0	1	0	0						0	(+2)				
	(+ 3)	0	1	0	0	1	0	1	0	1	1	(-5)	0	+1	0	1	0	0 (+1)
	(+ 2)	0	1	0	0	0	0	1	0	1	1	(-5)	0	+1	0	0	1	1 (+0)
	(+ 1)	0	0	1	1	1	0	1	1	0	1	(-3)	0	0	0	1	0	0 (+1)
	(+ 0)	0	0	1	1	0	0	1	1	0	1	(-3)	0	0	0	0	1	1 (+0)
Addend Negative	(+ 1)	1	0	0	0	1	0	0	0	1	1	(+3)	0	0	0	1	0	0 (+1)
	(+ 0)	1	0	0	0	0	0	0	0	1	1	(+3)	0	0	0	0	1	1 (+0)
	(- 1)	0	1	1	1	1	1	0	1	0	1	(+5)	1	-1	0	1	0	0 (+1)
	(- 2)	0	1	1	1	0	1	0	1	0	1	(+5)	1	-1	0	0	1	1 (+0)
	(- 3)	0	1	1	0	1	1						1	(-2)				
	(- 4)	0	1	1	0	0	1						1	(-2)				
	(- 5)	0	1	0	1	1	1						1	(-3)				
	(- 6)	0	1	0	1	0	0						1	(-3)				
	(- 7)	0	1	0	0	1	0						1	(-4)				
	(- 8)	0	1	0	0	0	0						1	(-4)				
	(- 9)	0	0	1	1	1	0						1	(-5)				

2.5 Determination of Integrator Output

The existence of a dX is indicated by X_e being on (see Address Unit for dX_e input to X_e). This is a necessary but not a sufficient condition for a dZ output, the other conditions being a function of the output multiplier.

2.5.1 Output Multiplier = 1

The existence of an output, Z_n, is determined at P_{35} time by X_s, C_2 and S_4 (which at this pulse time holds the sign of Y_D).

		X_s	C_2	S_4	Z_n	
Sect. 1.6 Case	V	1	1	1	0	X_s = 1: dX positive
"	IV	1	1	0	1	
"	II	1	0	1	1	S_4 = 1: Y_D negative
"	I	1	0	0	0	
"	IV	0	1	1	1	C_2 = 1: carry from R+ (Y_D) (dX) addition
"	V	0	1	0	0	
"	I	0	0	1	0	
"	II	0	0	0	1	

An examination of G11 indicates that it provides for the four cases in which $Z_n = 1$.

In addition to the normal output, an output results if the integrand is greater than one in absolute value when a dX occurs. This is determined by the most significant digit of Y_D, Y_{DS}, which has its uncorrected new value held in $S_4 - S_1$ at P_{35}. The form taken by this first sum depends upon the sign of dy. (See chart on the following page)

The columns under M_d indicate the state of M_d as determined by G14. Since appropriate scaling will hold Y_D within $+2 > Y_D \geq -2$, M_d is effectively a signal indicating that the integrand is greater than 1 in absolute value.

R_f is required in the M_d term so that M_d cannot come true for an integrator that is not coded. This prevents R_z from turning on and initiating an automatic reset if I_{59} is not coded (see G11 of the Control Unit).

	dy pos		dy neg	
Final Sum	$S_4S_3S_2S_1$	M_d	$S_4S_3S_2S_1$	M_d
+4	1010	1	0100	0
+3	1001	0	0011	1
+2	1000	1	0010	0
+1	0111	1	0001	1
0	0110	0	0000	0
-1	1111	0	1001	0
-2	1110	1	1000	1
-3	1101	0	0111	1
-4	1100	1	0110	0
-5	1011	0	0101	1

2.5.2 Output Multiplier =2

An examination of the table on page 22 indicates an output of ± 1 when $\beta = 1$.

2.5.3 Output Multiplier =5

Here an output occurs when $S_4 = 1$ (see table on page 24).

The output term in G12 is

$$dZ = G' + \underbrace{S'_A M_d + S'_A Z_n}_{X_1} + \underbrace{M_I S_b}_{X_2} + \underbrace{M_{II} S_b}_{X_5} + S_A M'_d S_b$$

S_A is a signal developed in the Fill Unit to hold the indication of servo operation. For normal integrator outputs with a multipler of 1, M_D and Z_n must be qualified with S'_A signifying that the integrator is not a servo. Servos will have outputs of the type $S_A M'_d S_b$; where M'_D assures a zero output if the Y register is equal to or greater than one

in magnitude (decision integrator operation), and S_b is a signal indicating that Y_D is not zero.

S_b is developed in G15 where the fractional digits of Y_D that are to be recorded are examined during 0_3 times. If any digit is not zero, U_4 U_3 U_2 U_1 = 0011, S_b is turned on. For output multipliers of 2 and 5, S_b is turned off at the end of P_{34} if there is no integrator output; this prevents an output from being recorded on the Z line since dZ is qualified by P_{35} (G3 Output Unit) before recording. If not previously turned off, S_b is reset each time R_f' is set high.

The G' term of dZ is used during idle to recirculate Z line information and will be discussed in the section on the Output Unit.

2.6 Sign of Output

The sign of the output is independent of the output multiplier, being a function of the sign of dX, X_s, and the sign of Y_d, S_4:

S_4	X_s	dZ_s	At P_{35} time
1	1	0	S_4 = 1: Y_D negative
1	0	1	
0	1	1	X_s = 1: dX positive
0	0	0	dZ_s = 1: dZ positive

An examination of G12 indicates that it provides for the two cases in which dZ_s = 1. The $G'X_s$ term corresponds to the G' term of dZ and it also will be discussed more completely in the Output Unit Section.

2.7 Error Detection

2.7.1 Multiple Output (gate G13)

When a multiple output occurs, r_D of G13 becomes high as explained below. r_D high turns on R_D which in turn causes R_o (G18) to become high. $R_o P_1$ turns G_1 off stopping computation (see G13 of Fill Unit).

2.7.1.1 Output Multiplier = 1

A double output occurs when both Z_n and M_d are true at P_{35} time.

2.7.1.2 Output Multiplier = 2

A double output occurs when both Z_n and $M_I S_b'$ are true at P_{35} time. Combining the two above cases we have

$$P_{35}\ Z_n\ (M_d + M_I S_b') \qquad \text{(III)}$$

Since Z_n is independent of X_e, it may be high even though X_e' is high, therefore (III) above must be qualified by X_e (see 2.5).

The condition of R_D is irrelevant in servo operation, hence the S_A' term.

G insures that the X_e term refers to dX_e since, when G' is high, X_e is used to process other information.

2.7.1.3 Output Multiplier = 5

A double or triple output is indicated at P_{34} (see 2.4.3) by $S_4\ (S_3 S_2' + S_3' S_2)\ M_{II}$ which is the term appearing in G13. Note that this term is independent of Z_n and therefore need not be qualified by $G\ X_e$. Four or five outputs are not apt to occur unless the problem scaling is greatly in error.

2.7.2 Proscribed Code

There are sixteen possible combinations of four binary digits. Ten of these are employed to represent the ten decimal digits, the remaining six being:

0000
0001
0010
1101
1110
1111

Of these, none can appear in the arithmetic channels except 0000. G17 detects the presence of any of the remaining five and stops computation by turning on R_c which in turn causes R_o of G18 to become high as in the case of a multiple output error. Although all six of these codes may occur, the last three can never be observed in a static condition in the arithmetic channels. When +3 is added to each of these last three codes in the first adder, during the next iteration cycle after the code occurs, a carry results; and, hence, the second adder will also add 3 instead of subtracting 3, changing these codes into 0011, 0100, and 0101 respectively.

ADDRESS UNIT

The incremental inputs to each integrator are developed by the address unit shown in figure IV. The dy counter, G11, receives inputs from the coincidence detector G15. At P_{37} the completed count is transformed as a function of the mode, detected by G12, and transferred to the d_{yd} register by G6. A counting process then transforms the number in the dy counter into the excess-three code during P_{38} and P_1. At the end of P_1 this number is read into the dy register by G5. D_I and dx are picked up by the d_I and dx counters (G13 and G14) and, at P_1 transferred to the d_I and dx registers (G7 and G8). D_y, D_{yd} and D_I are then sequenced by G1 so as to appear in the addend register of the 1st Adder at the correct times. Finally, G16, G17 and G18 serve to clear the address channels.

In figure II, four address channels (Y_2, Y_1, X and I) and four dZ output channels (Z_I, Z_{IS}, Z_{II} and Z_{IIS}) are indicated. The section of these channels shown in integrator 59 constitute the input information for the next integrator, in this case, integrator 00. The pair of read flip flops, Z_I and Z_{IS}, hold the outputs of integrators 1, 3, 5, 7, ..., 51, 53 and 55 at pulse times 6, 7, 8, 9, ..., 31, 32 and 33 of integrator 59. Derived from Z_I and Z_{IS}, Z_b and Z_{bs}, and the output of the present integrator (see Output Unit), a pair of flip flops Z_1 and Z_{1S} hold the outputs of integrators 1, 3, 5, 7....55, 57 and 59 at pulse times 7, 8, 9, 10....34, 35 and 36 of integrator 59. Z_1 and Z_{1S} also hold four function inputs, B_1, B_2, B_3 and B_4, at pulse times 2, 3, 4 and 5 and the machine independent variable (τ) at pulse time 6. Address code pulses in the Y_1 channel in the interval of pulse times 2 through 36 serve to detect the outputs held in Z_1 and Z_{1S} at the times that the code pulses are in the Y_1 flip flop. A similar arrangement (using Z_{II} and Z_{IIS}) gives two flip flops, Z_2 and Z_{2S}, which are examined by code pulses in Y_2.

For the X address channel a code pulse in pulse time 1 indicates that the dX input will be obtained from Z_2 and Z_{2S} and no code pulse in pulse time 1 indicates that the dX input will be obtained from Z_1 and Z_{1S}.

A code pulse in the interval of pulse times 2 through 36 indicates when the appropriate source is to be examined. The I address channel operates in a similar manner to obtain an input to an initial condition register.

3.1 Coincidence Detector (G15)

Coincidence between Y_1 and Z_1 Z_{1S}, indicates a +1 input, while coincidence between Y_1 and Z_1 Z_{1S}' indicates a -1 input. Similarly for Y_2. It may happen, however, that both Y_1 and Y_2 contain "ones" in the same pulse time. In this case a +2 or -2 may be the correct input to the dy counter and the correlative +1's or -1's must be suppressed. Therefore, each Y_i Z_i Z_{is} term is qualified by $(Y_j Z_j)' = Y_j' + Z_j'$ where $j = 2$ if $i = 1$ and $j = 1$ if $i = 2$.

The $P_{2/36}$ term insures that the digit in P_{37} of Y_2 (namely, D_2) is not read as an address because at that time Z_2 and Z_{2S} are not necessarily low as can be seen from figure II (the information held in Z_{II} at P_{36} will appear in Z_2 at P_{37}). The P_6 terms in $\overline{+1}$ and $\overline{-1}$ are for adder operation, the coding for an adder being a one in both Y_2 and Y_1 at P_6. If an integrator is being used as an adder, its own output must be negated and added to itself as a dy. An integrators own output is in Z_{II} and Z_{IIS} at P_6 of the previous integrator, (figure II). The remaining terms in G15 will be considered in connection with the dy Counter in sections 3.2 and 3.3.

3.2 Dy Counter (G11)

For simplicity in operation, the dy counter employs the excess-six binary coded decimal system. The coding scheme is as follows:

N_S	N_4	N_3	N_2	N_1	
0	1	1	1	1	(+9)
0	1	1	1	0	(+8)
0	1	1	0	1	(+7)
0	1	1	0	0	(+6)
0	1	0	1	1	(+5)
0	1	0	1	0	(+4)
0	1	0	0	1	(+3)
0	1	0	0	0	(+2)
0	0	1	1	1	(+1)
0	0	1	1	0	(+0)
1	1	0	0	1	(-1)
1	1	0	0	0	(-2)
1	0	1	1	1	(-3)
1	0	1	1	0	(-4)
1	0	1	0	1	(-5)
1	0	1	0	0	(-6)
1	0	0	1	1	(-7)
1	0	0	1	0	(-8)
1	0	0	0	1	(-9)
1	0	0	0	0	(-10)

L_6 and L_7 provide for discontinuities in crossing between (-1) and (0). Normally a carry from the first place may cause carries in subsequent places so that $\triangle$+1 and $\triangle$-1 must appear as terms in all subsequent stages. The necessity for this may be eliminated, however, by noting that a carry out of the first place occurs when

$\triangle$+1 is high and N_1 is high

$\triangle$-1 is high and N_1 is low.

In these casses N_1 is changed by $\triangle$+1 or $\triangle$-1 but the carry term is handled by $\triangle$+2 or $\triangle$-2. This is accomplished by the N_1 terms in $\triangle$+2 and $\triangle$-2 of G15.

At the end of P_1 the counter is reset to +0 (00110) by making N'_1 and N_2 high as well as L_6 which in turn causes N_3, N'_4 and N'_s to become high.

If a $\triangle$-1 is now received, $\triangle$-2 will immediately become high due to the N'_1 $\triangle$-1 term; $\triangle$-2 high will immediately cause $L_7 = N'_s\ N'_4\ \triangle$-2 $P_{2/36}$ to become high. At the following clock pulse N_1 becomes high due to N'_1 $\triangle$-1, N'_2 becomes high due to N_2 $\triangle$-2, and N'_3, N_4 and N_s become high due to L_7. Thus the $\triangle$-1 has caused the +0 to transform to -1 (11001). Any other input or combination of inputs can be similarly checked to demonstrate that G11 adds or subtracts in accordance with the above table.

3.3 Dy and Dy/2 Registers

At P_{34}, D_s is turned off making D_+ (in G9) true in P_{35}. This sets the D_4, D_3, D_2, D_1 and $D_{1/2}$ flip flops to 00110 and the H_4, H_3, H_2, H_1 flip flops to 0011 at the end of P_{35}. By P_{37} the dy count has been completed and is held, in the excess-six code, in N_4 - N_1 with N_s holding the sign. If the integrator is employing the interpolative mode, the extrapolative mode with dX = 0 (E'_e high), or the multiplicative mode with $dX \neq 0$ (E_e high), $Y_{1/2}$ (in G12) is high at P_{37}. Figure II shows that the different integration modes are coded for in P_{37} of Y_1 and Y_2. Since a right shift of one place in a binary register effectively divides the number held by 2, $dy_d = 1/2 dy$ is obtained by a simple shift to the right in D_4 - $D_{1/2}$. This shift is brought about by the first term of each flip flop input developed in G6; i.e. N'_2 controls $_od_1$, N'_3 $_od_2$, N_4 controls d_3, etc. The number held in the register after P_{37} is $1/2 (N \pm 6) = 1/2\ N \pm 3$, that is, 1/2 the dy count in the excess-three code. If the integrator is employing the extrapolative mode with $dX \neq 0$, $Y_{3/2}$ (in G12) is high at P_{37} and a transformation matrix, G10, is used in conjunction with the $Y_{3/2}$ terms of G6 to transform the number held in the N flip flops in the excess-six code to three halves of that number in the excess-three code. That is, after P_{37} the D flip flips hold in the excess-three code $dy_d = 3/2$ dy. Also at the end of P_{37} the sign of dy is transferred directly from N_s to D_s.

The table below indicates what the N and D flip flops will hold for all possible values of dy and for $Y_{1/2}$ or $Y_{3/2}$ being true.

← $Y_{1/2}$												$Y_{3/2}$ →					
1/2 dy ± 3						dy ± 6						3/2 dy ± 3					
D_4	D_3	D_2	D_1	$D_{1/2}$		N_5	N_4	N_3	N_2	N_1		D_4	D_3	D_2	D_1	$D_{1/2}$	
0	1	1	1	1	(+4 1/2)	0	1	1	1	1	(+9)						
0	1	1	1	0	(+4)	0	1	1	1	0	(+8)						
0	1	1	0	1	(+3 1/2)	0	1	1	0	1	(+7)						
0	1	1	0	0	(+3)	0	1	1	0	0	(+6)	1	1	0	0	0	(+9)
0	1	0	1	1	(+2 1/2)	0	1	0	1	1	(+5)	1	0	1	0	1	(+7 1/2)
0	1	0	1	0	(+2)	0	1	0	1	0	(+4)	1	0	0	1	0	(+6)
0	1	0	0	1	(+1 1/2)	0	1	0	0	1	(+3)	0	1	1	1	1	(+4 1/2)
0	1	0	0	0	(+1)	0	1	0	0	0	(+2)	0	1	1	0	0	(+3)
0	0	1	1	1	(+1/2)	0	0	1	1	1	(+1)	0	1	0	0	1	(+1 1/2)
0	0	1	1	0	(+0)	0	0	1	1	0	(+0)	0	0	1	1	0	(+0)
1	1	0	0	1	(-1/2)	1	1	0	0	1	(-1)	1	0	1	1	1	(-1 1/2)
1	1	0	0	0	(-1)	1	1	0	0	0	(-2)	1	0	1	0	0	(-3)
1	0	1	1	1	(-1 1/2)	1	0	1	1	1	(-3)	1	0	0	0	1	(-4 1/2)
1	0	1	1	0	(-2)	1	0	1	1	0	(-4)	0	1	1	1	0	(-6)
1	0	1	0	1	(-2 1/2)	1	0	1	0	1	(-5)	0	1	0	1	1	(-7 1/2)
1	0	1	0	0	(-3)	1	0	1	0	0	(-6)	0	1	0	0	0	(-9)
1	0	0	1	1	(-3 1/2)	1	0	0	1	1	(-7)						
1	0	0	1	0	(-4)	1	0	0	1	0	(-8)						
1	0	0	0	1	(-4 1/2)	1	0	0	0	1	(-9)						
1	0	0	0	0	(-5)	1	0	0	0	0	(-10)						

At the end of P_{37}, the dy counter starts to count up three if the number is negative and down three if the number is positive. This is accomplished by the P_{37} and P_{38} terms in G15 and effectively transforms the number being held in the counter from the excess-six to the excess-three code. The L_6 and L_7 terms are made ineffective during this period by the $P_{2/36}$ term, insuring that the above counting process will be binary.

Y_N of G3 will be high during P_1 if the $D_4 - D_{1/2}$ flip flops are still in the 00110 configuration. This will result if dy = 0, or if neither $Y_{1/2}$ nor $Y_{3/2}$ was high at P_{37} (this last case corresponds to rectangular mode integration or multiplicative mode integration when dX = 0). If Y_N is high G6 sets the $D_4 - D_1$ flip flops, at the end of P_1, from the corrected number that is now held in the $N_4 - N_1$ flip flops. Also at the end of P_1, G5 sets this corrected number into the H flip flops and G11 resets the N flip flops to zero in the excess-six code.

At P_2, then D holds Dy_d and H holds Dy, and as will be shown below I_s and I_e hold D_I. These are gathered in G1 and timed so as to provide the correct addend for the first adder of the Arithmetic Unit during 0_1, 0_2 and 0_3. First $D_{1/2}$ must be added to the digit in the round off place of Y_D. This is accomplished by the $R'_e D_{1/2}$ terms of G1. Since either 0 = 0011 or 5 = 1000 is to be added to this digit of Y_D, no $D_{1/2}$ term is required for J_3 as it is to remain 0 in either case. Since the addition is qualified by the activate signal, R_e, (see G9 of the Arithmetic Unit) no addition will actually take place until this signal becomes true even though the J's will be holding the addend for this addition at every 0_4 time up until the addition of this digit has been made. After the addition involving $D_{1/2}$ has been made, the rest of Dy_d is added to Y_D starting at the next 0_1 time; this is accomplished by the $(0_4 P_{2/36} R_e)$ terms of G1. At the next pulse time, 0_2, the addition of Dy to Y is started by the $(0_1 R_f)$ terms of G1. Similarly DI is added to Y_I by the $(0_2 R_f)$ terms of G1. The information flowing through the J buffers of G1 must be one pulse time early with respect to figure 11 because of the one pulse delay that will be introduced by the L flip flops in the Arithmetic Unit. After Dy and Dy_d have been added into Y and Y_D respectively, $0_2 R_f$ (see G9) will reset the $H_4 - H_1$ and the $D_4 - D_{1/2}$ registers. If the Dy was positive D'_s will be high, and consequently D_+ will set the two registers to zero in the excess-three code ($H_4 - H_1$ set to 0011 and $D_4 - D_{1/2}$ set to 00110); the registers will now hold this zero until the succeeding integrator is to be considered. If the Dy was negative, the counter formed the complement of Dy and the registers hold

the complements of Dy, 1/2 Dy or 3/2 Dy as the case may be. D_s will be high at the end of P_{37} and D_- will be high for all 0_2's after R_f comes on. Consequently after the first addition of Dy and Dy_d into Y and Y_D has been completed, D_- resets H_4 - H_1 and D_4 - D_1 to 1100 (G5 and G6); this effectively changes all of the zeros to the left of the first digit of Dy and Dy_d to nines completing the complementation process. These nines are then in turn added to the corresponding digits of Y and Y_D in the Arithmetic Unit. The H and D registers will be reset from nines to zeros by D_+ coming high at P_{35} time, and the dy circuits are ready for the next integrator inputs.

G2 and G4 are simply auxiliary gates used to develop the timing signals of G1.

3.4 DI Counter and Register (G13 and G7)

The DI counter consists of two flip flops, V_e and V_s. V_e holds the magnitude of DI, either 1 or 0, while V_s serves the dual purpose of first designating which Z flip flops are to be examined for the DI input and secondly of reading and holding the sign of DI after the Z flip flops have been examined. V_e is always set off at the end of P_1 while V_s is turned on if a one occurs in P_1 of I (position marked S_I in figure II) indicating that Z_2 and Z_{2s} are to be examined or V_s is turned off if a zero occurs in this place indicating that Z_1 and Z_{1s} are to be examined. If in the period $P_{6/36}$ there is a coincidence between a one in the I channel and Z_1 if V_s' is high or Z_2 if V_s is high, V_e is turned on indicating a change of one increment in Y_I. A pulse in Z_{1s} or Z_{2s} indicates that the corresponding digit in Z_1 or Z_2 is positive while the absence of a pulse in Z_{1s} or Z_{2s} indicates that the corresponding digit in Z_1 or Z_2 is negative. At the same time Z_1 or Z_2 is being examined, V_s is reset with the sign of DI in the following manner:

1. If V_s was on Z_{2s} is examined.
   a. If there is a coincidence between I and Z_{2s}, V_s stays on indicating a (+) DI.
   b. If there is a coincidence between I and Z_{2s}', V_s is turned off indicating a (-) DI.

2. If V_s was off (V_s' high) Z_{1s} is examined.

   a. If there is a coincidence between I and Z_{1s}, V_s is turned on indicating a (+) DI.

   b. If there is a coincidence between I and Z_{1s}', V_s remains off indicating a (-) DI.

At the end of P_{37} the DI register, I_s I_e, is set to 00 and at P_1 of the next integrator the number being held in $V_s V_e$ is read into this register by G7. The first time $0_2 R_f$ occurs, G1 will present DI, that is $J_4 - J_1$ will be

0011 if DI = 0
0100 if DI = +1
1100 if DI = -1

At the end of this $0_2 R_f$ time, if DI = +1, I_e is turned off so that zeros will be added to the successive places of I. In the other two cases no change is necessary as zeros will continue to be added in the case of DI = 0 and nines will be continued to be added for the complement of DI in the case of DI = -1.

The following table shows the flow of information from G1 through the J buffers during various 0 and P times:

O	P	$J_4 J_3 J_2 J_1$	R_e	R_f	
1	1	$Dy_{d\,1/2}$ = 0 or 5	0	0	
1	2	0 0 1 1	0	0	
2	3	0 0 1 1	0	0	
3	4	0 0 1 1	0	0	
4	5	$Dy_{d\,1/2}$ =0 or 5	0	0	
1	6	0 0 1 1	1	0	
2	7	0 0 1 1	1	0	
3	8	0 0 1 1	1	0	
4	9	Dy_d	1	0	
1	10	Dy	1	1	
2	11	DI = 0, 1, or 9	1	1	
3	12	0 0 1 1	1	1	
4	13	Sign of Dy_d {0 or 9	1	1	
1	14	and Dy {0 or 9	1	1	
2	15	Sign of DI = 0 or 9	1	1	
3	16	0 0 1 1	1	1	
4	17	Sign of Dy_d {0 or 9	1	1	
1	18	and Dy {0 or 9	1	1	
2	19	Sign of DI = 0 or 9	1	1	etc.

3.5 DX Counter and Register (G14 and G8)

The DX counter consists of two flip flops E_e and E_s. E_e holds the magnitude of DX, either 1 or 0, while E_s serves the multiple purpose of first designating which Z flip flops are to be examined for the DX input, secondly of reading and holding the sign of DX from the Z line, and lastly of picking up any sign reversal pulse that may occur at P_{37} (this sign reversal pulse, which is designated by R_s in figure II, simply reverses the state of E_s). E_e is always set on at the end of P_1 so that time will be the DX input if there is no other DX input coded for. E_e is high for a DX input of $\pm$ 1 and low for a zero DX input. A pulse in the P_1 position of X (designated by S_x in figure II) will turn E_s off indicating that the DX input will come from Z_2 and Z_{2s} while the absence of a pulse indicates that Z_1 and Z_{1s} will supply the DX input and E_s is turned on. The Z lines are now scanned by G14 and if the DX input is zero E_e turns E_e off. At the same time Z_1 or Z_2 is being examined for the magnitude of DX, E_s is reset with the sign of DX in the following manner:

1. If E_s was on, Z_{1s} is examined.
   a. If there is a coincidence between X and Z_{1s}, E_s stays on indicating a (+) DX.
   b. If there is a coincidence between X and Z'_{1s}, E_s is turned off indicating a (-) DX.
2. If E_s was off, (E'_s high) Z_{2s} is examined.
   a. If there is a coincidence between X and Z_{2s}, E_s is turned on indicating a (+) DX.
   b. If there is a coincidence between X and Z'_{2s}, E_s remains off indicating a (-) DX.

At P_{37}, X is examined and if there is a sign reversal pulse, the P_{37} E'_s X or the P_{37} E_s X term of G14 will reverse the state of the E_s flip flop.

If no DX information is coded P_1 X' turns E_s on which in conjunction with E_e being on as previously pointed out, provides a DX of +1 (+ time).

The DX register X_e and X_s (Arithmetic Unit) is always reset to 00 at P_{37} so that it will be ready to read the inputs from G8 at P_1. The dX_e

input at P_1 is E_e and the dX_s input is E_s. The remaining terms of G8 will be explained later as they are concerned with fill and Z line operations.

3.6 Clear Terms (G16, G17, and G18)

C_{AD}' (see G9 of Fill Unit) is a signal which is low during all integrators if the Clear Complete button is pressed or is low only during I_{i-1} if the Clear Address button is pressed. This is used to qualify the four address line read signals so as to provide for erasing addresses.

3.7 Coupler Signals (G19)

C_o and C_k are signals used when two D-12 computers are coupled together and are discussed in the section on the D-12 coupler.

OUTPUT UNIT

4.1 Introduction

The purpose of the Output Unit (figure V) is to store the latest outputs of each of the 60 integrators and present these outputs sequentially to the Address Unit for use as inputs to each integrator in such a manner that the output of any integrator is available as an input to any other integrator. The outputs are stored in two memory channels, Z and Z_s. Each of these channels has two read heads, Z_I and Z_{II} for the Z channel and Z_{1S} and Z_{1IS} for the Z_S channel. Figure II indicates the sequence in which the outputs should be presented to the Address Unit during I_{59}. Z_1 and Z_{1S} (G3) should present the output of I_{59} at P_{36}, I_{57} at P_{35}, etc. Z_2 and Z_{2S} (G1 and G5) should present the output of I_{58} at P_{36}, I_{56} at P_{35}, etc.

4.2 Z_1 Z_{1S} Signal Generation (G3)

Since the operation of the Z channel is exactly duplicated by the Z_s channel (the Z channel processing the integrator outputs and the Z_s channel processing the sign of these outputs) only the Z channel will be considered in detail here.

The P_{35} X_e dZ term of G3 delayed one pulse time by the Z_1 flip flop records the output of the present integrator, I_i, at P_{36} of I_i. The Z_I read head is so placed that this output is read and held by the Z_I read flip flop at P_{33} of I_{i+4}. $P_{6/36}$ Z_I of G3 now picks up this output pulse and recirculates it by again placing it in the Z_1 flip flop, this time at P_{34}. In other words the output of I_i, which was first recorded on the Z line at P_{36} of I_i, is again recorded on the Z line at P_{34} of I_{i+4}, at P_{32} of I_{i+8}, at P_{30} of I_{i+12}, at P_{36-2n} of I_{i+4n} where n = 0, 1, 2, ...14. That is, the same output of I_i is recorded on the Z_1 line 15 times during one iteration cycle, each time appearing two pulse times earlier modulo 38 with respect to the preceding time of recording. Z_I reads this output for the 15th time at P_5 of $I_{i+60} = I_i$, and, hence, the $P_{6/36}$ Z_I term of G3

does not recirculate this output again. However, the computer is now operating upon I_i for the next iteration cycle, and at P_{36} the new output of I_i will be recorded. The Z_1 channel is then in effect a recirculating delay line whose total delay is 4 x 38 - 2 = 150 pulse times.

It can be seen then that if $P_{35}\ X_e\ dZ$ and $P_{6/36}\ Z_I$ were the only terms controlling Z_1, that during I_i the new output of I_i would be recorded at P_{36} while the output of I_{i-4} would be re-recorded at P_{34}, the output of I_{i-8} re-recorded at P_{32}, and so on with the output of I_{i-56} being re-recorded at P_8. All other pulse positions of the Z_1 line would be void of information, and the outputs of only 15 integrators would be available for use as inputs to any other integrator.

In order to make more efficient use of the Z line and to make more integrator outputs available as inputs to any other integrator, Z_a and Z_b flip flops (G2) are added along with a $P_{34}\ Z_b$ term in G3. At the end of P_{36} of I_i, Z_a picks up the output of I_i from Z_1 and holds this output until the end of P_{36} of I_{i+1} at which time the contents of Z_a is transferred to Z_b and Z_a picks up the output of I_{i+1} from Z_1. Z_b is always reset to 0 at P_{35}. Z_c, whose function will be explained later in connection with Z_2, receives the content of Z_b at the end of P_{35} and holds this information until it is reset to 0 by the next P_{34} pulse. The Z_a, Z_b, and Z_c flip flops, therefore, serve as a stepping chain to hold the outputs of the last three integrators. Consequently, Z_b holds the output of I_i at P_{34} of I_{i+2}. $P_{34}\ Z_b$ of G3 picks up this output for recording by Z_1 at P_{35} of I_{i+2}. $P_{6/36}\ Z_I$ will now recirculate this output in the same manner that the original I_i output was recirculated, re-recording it every four integrator times two pulse times earlier modulo 38 with respect to the preceding time of recording. The recirculation of this output of I_i that originally passed through Z_a and Z_b will cease after the 15th cycle. The 14th recording will be made at P_9 of I_{i+54} and this recording will be read by Z_I at P_6 of I_{i+58}; hence, the 15th recording is at P_7 of I_{i+58} and will be read at P_4 of $I_{i+62} = I_{i+2}$ which occurs too early in time to be recirculated by the $P_{6/36}\ Z_I$ term of G3. However, a new output of

I_i, delayed again by Z_a and Z_b, is now ready to be recorded at P_{35} of I_{i+2}.

It is seen that the P_{34} Z_b and $P_{6/36}$ Z_I terms of G3 will re-record during I_i the output of I_{i-2} at P_{35}, the output of I_{i-6} at P_{33}, the output of I_{i-10} at P_{31}, and so on with the output of I_{i-58} being re-recorded at P_7. Furthermore, considering all of the information that can now be put in the Z line, it is seen that during I_i, Z_1 holds the present output of I_i at P_{36}, the output of I_{i-2} at P_{35}, the output of I_{i-4} at P_{34}, the output of I_{i-6} at P_{33}, ...the output of I_{i-56} at P_8, and the output of I_{i-58} at P_7. The outputs of one half of the computers total number of integrators are now available as inputs to any other integrator; and since the addresses for I_i input are coded in I_{i-1}, it can be seen that using only the circuitry discussed already that only the outputs from even numbered integrators are available as inputs to odd numbered integrators and vice versa. A second Z read head, Z_{II}, and associated circuitry is provided in order to allow the outputs from the remaining 30 integrators, and, hence, from all 60 of the computer integrators to be available for inputs to any integrator. Operation of Z_2 and Z_{2s} will be explained in section 4.3.

The $P_5 \tau_e$ term in G3 is the machine independent variable set by a switch on the control panel. If the Time Switch is set to (+) a pulse is recorded by Z_1 and also by Z_{1s} at P_6 in every integrator (signifying + time). If the Time Switch is set to (-) a pulse is recorded by Z_1 in every integrator at P_6 but no pulse is recorded by Z_{1s} (signifying - time). If the Time Switch is set to its middle position no pulses are recorded. Since these pulses recorded at P_6 will be read at P_3, they will not be recirculated, but new pulses will be recorded at every P_6 as long as the Time Switch is set to (+) or (-).

The P_{35} F_e term is used to fill integrator outputs during gross fill and is explained in section 7.3.

The $P_{1/4}$ Z_b term is used for entering incremental inputs (B_4, B_3, B_2 and B_1). These inputs are recorded on the Z line but are not recircu-

Page Missing From Original

latest output of I_{i-1} will be held by Z_a and the latest output of I_{i-3} by Z_c (section 4.2), hence, the P_{35} Z_a and P_{34} Z_c terms of G1. At P_{33} time of I_{i-1}, Z_I reads the latest output of I_{i-5} and at this time G6 sets this output into Z_e where it is held for one integrator time. G1 picks up the contents of Z_e at P_{33} of I_i. Hence, during I_i, Z_2 will hold the following outputs: I_{i-59} at P_7, I_{i-57} at P_8, I_{i-55} at P_9.....I_{i-7} at P_{33}, I_{i-5} at P_{34}, I_{i-3} at P_{35} and I_{i-1} at P_{36}. Therefore, outputs from all of the integrators of the computer appear at either Z_1 or Z_2 during any integrator time.

Incremental inputs B_1, B_2, B_3 and B_4 that, if present, were recorded by Z_1 at P_2, P_3, P_4, and P5 respectively of I_{i-7} will be read by Z_{II} at P_{36}, P_{37} and P_{38} of I_{i-1} and P_1 of I_i respectively. All of these terms except B_1 will be blocked by the $P_{6/36}$ Z_{II} term of G1. B_1 will be read by G1 at P_{36} and transferred to Z_2 where it will be held during P_{37}. The Address Unit, however, only reads Z_2 during $P_{2/36}$; hence, B_1 appearing in Z_2 at P_{37} causes no difficulty.

The $P_{1/4}$ Z_u term of G1 is used for entering incremental inputs (A_4, A_3, A_2 and A_1) into the computer. Four inputs can be made via G1 and an additional four via G3 (section 4.2).

The machine independent variable is also set into Z_2 at P_6 by the P_5 (T_e) term of G1; however, it is of opposite sign than the time term of G3 as is shown by the P_5 (T_s) term of G5.

Figure VIa shows a section of the Z line at P_{33} time of I_{20}, and Figure VIb shows relationships between different Output Unit flip flops during I_{13} to I_{20}.

G8 and G9 are coincidence gates used to enter address pulses during automatic fill and will be discussed in section 6.4.

4.4 Idling and Clearing of Z Lines

A means must be included to prevent the Z lines from losing all information during idling, as during this time no new outputs will be recorded while the precessing in the lines continues, causing all information to be lost within one iteration cycle after idling starts. It is only necessary to make provisions for the same output of I_i to be re-recorded for the sixteenth time with the sixteenth recording taking the place of what would have been the new output of I_i if computation had continued (see section 4.2). Z_I and Z_{IS} read the output of I_i for the fifteenth time at P_5 of $I_{i+60} = I_i$, and, hence, the $P_{6/36}$ terms of G3 will no longer recirculate this output. However, since G' will be true during idle, the P_5 terms of G8 of the Address Unit will read this output and place it in the X_e and X_s flip flops of the Arithmetic Unit to be entered into G3 of the Output Unit at the following P_{35} time as the new output of I_i (dZ will be high during idle, see G12 of Arithmetic Unit). In this way all outputs will recirculate during idle and none will be lost.

The C'_Z terms of G8 of the Address Unit are also used for clearing. To clear C'_Z is made false during idle, preventing these terms from reading Z_I and Z_{IS}; and, hence, blocking X_e and X_s from permitting recirculation as discussed above. All Z information will now precess right off the lines. C'_Z is developed in G9 of the Fill Unit and may be made false for all integrators or for only selected integrators as explained in Section 6.8.

CONTROL UNIT

5.1 Introduction

The control unit shown in figure VII contains two counters (G17 and G5) which specify the integrator and pulse times. The former, called the Integrator Counter, has 60 states while the latter, called the Pulse Time Counter, has 38. Specific integrator and pulse times are developed using combinations of the counter outputs in G16. G2 and G6 serve to identify a selected integrator and its immediate predecessor, I_i and I_{i-1} respectively. The two counters also serve to time information into the computer during automatic fill through G1 and G3; this operation will be discussed in connection with the Fill Unit.

In addition, the Control Unit provides, in G11, 12, and 14, for the re-setting of initial conditions, while G13 develops the output multiplier indicators. G7, G8, G9, and G10 serve to provide the trigger, blanking and four level sweep signals for the scope display. F_i, a signal used in conjunction with manual fill, is also developed in G7 and will be discussed in the section on the Fill Unit. Finally, G18 develops timing signals used in the operation of additional Graph Plotters or a D-12 Coupler and is optional equipment on all computers.

5.2 Integrator Counter G17

The integrator counter has two modes of operation depending upon the position of the Thirty-Sixty switch. When in the Sixty position (X_y) is high, but when in the Thirty position (X'_y) is high changing the logic for the most significant digit. Both counting processes are shown below.

K_7 is not required for counting integrators, but is used to indicate odd and even cycles of computation during 30 integrator operation.

Most Significant Digit

(X_y)				(X'_y)		
K_7	K_6	K_5	Value	K_6	K_5	Value
0	1	1	00	1	1	00
1	0	0	10	0	0	10
1	0	1	20	0	1	20
1	1	0	30			
1	1	1	40			
0	0	0	50			

Least Significant Digit

K_4	K_3	K_2	K_1	Value
0	0	1	1	0
0	1	0	0	1
0	1	0	1	2
0	1	1	0	3
0	1	1	1	4
1	0	0	0	5
1	0	0	1	6
1	0	1	0	7
1	0	1	1	8
1	1	0	0	9

It is seen that the first four flip flop stages comprise a counter that counts from 0 to 9 in the excess three system, while if (X_y) is high the last three stages comprise a counter that counts from 0 to 5 also in the excess three system. The least significant digit stages count P_{38} pulses derived from the Pulse Time Counter by G16, while the most significant digit stages count K_{38} pulses (selected P_{38} pulses -- see G17)

which occur once every cycle of the four L.S.D. stages and at the end of that cycle. K_{38} also serves to reset the four L.S.D. stages from 9 back to 0, causing these stages to become a decade counter. Hence, the seven stages will count from 00 to 59 and then recycle. When (X'_y) is high, K_5 and K_6 count from 0 to 2 while K_1, K_2, K_3, and K_4 count in the same manner as before. K_7 changes its state once for every complete cycle of K_5 and K_6, being high on the first cycle of computation and every other cycle thereafter.

5.3 Pulse Time Counter G5

The 38 pulse times are indicated by the following configurations of the Q flip flops which the logic in G5 produces.

Q_7	Q_6	Q_5	Q_4	Q_3	Q_2	Q_1	Pulse Time	0 Time
1	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	2	1
0	0	0	0	0	1	0	3	2
0	0	0	0	1	1	1	4	3
0	0	0	0	1	0	1	5	4
0	0	0	1	1	0	0	6	1
0	0	0	1	1	1	0	7	2
0	0	0	1	0	1	1	8	3
0	0	0	1	0	0	1	9	4
0	0	1	1	0	0	0	10	1
0	0	1	1	0	1	0	11	2
0	0	1	1	1	1	1	12	3
0	0	1	1	1	0	1	13	4
0	0	1	0	1	0	0	14	1
0	0	1	0	1	1	0	15	2
0	0	1	0	0	1	1	16	3
0	0	1	0	0	0	1	17	4
0	1	1	0	0	0	0	18	1
0	1	1	0	0	1	0	19	2
0	1	1	0	1	1	1	20	3
0	1	1	0	1	0	1	21	4
0	1	1	1	1	0	0	22	1
0	1	1	1	1	1	0	23	2
0	1	1	1	0	1	1	24	3
0	1	1	1	0	0	1	25	4
0	1	0	1	0	0	0	26	1
0	1	0	1	0	1	0	27	2
0	1	0	1	1	1	1	28	3
0	1	0	1	1	0	1	29	4
0	1	0	0	1	0	0	30	1
0	1	0	0	1	1	0	31	2
0	1	0	0	0	1	1	32	3
0	1	0	0	0	0	1	33	4
1	1	0	0	0	0	0	34	1
1	1	0	0	0	1	0	35	2
1	1	0	0	1	1	1	36	3
1	1	0	0	1	0	1	37	4
1	0	0	0	1	0	0	38	1
1	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	2	1
0	0	0	0	0	1	0	3	2
0	0	0	0	1	1	1	4	3

Six flip flops suffice for the counter itself, since $2^6 = 64 > 38$. The table gives 38 successively different configurations of $Q_7 - Q_2$ with the 39th configuration being identical with the first; hence, $Q_7 - Q_2$

constitute an automatic recycling 38 pulse counter. This counting process is such that any two consecutive configurations differ at most in one place. As will be seen, electronic development of individual pulse times is somewhat simplified with this type of pulse counter. Q_1 flip flop follows its driver, Q_2, lagging by one pulse time and is provided so that the 0 times can be easily distinguished.

5.4 Timing Pulses G16

The pulse times developed in this unit are specific configurations of the outputs of the pulse time counter. P_1, P_2, P_5, P_6, P_{31}, P_{33}, P_{35}, P_{36}, P_{37}, P_{38}, $P_{1/4}$, $P_{2/36}$, $P_{3/38}$ and $P_{6/36}$ are either developed directly and supplied with a buffer output or have their inverse developed and supplied with an inverter output in G16. P_{34} is generated by a flip flop being turned on with P_{33} and turned off by its own true output. In addition to depending upon the pulse time counter, P_5, P_6 and $P_{6/36}$ also depend upon the thirty-sixty switch.

When in thirty integrator operation,

P_5 becomes P_{20}

P_6 becomes P_{21}

$P_{6/36}$ becomes $P_{21/36}$.

These changes during 30 integrator operation effect automatic filling of addresses and recirculation of information on the Z lines. See Fig. 47 of the Operation Manual and the section on automatic filling in this Maintenance Manual.

The four 0 signals, 0_1, 0_2, 0_3, and 0_4, which indicate the times at which Y_D, Y, Y_i and R respectively are presented to the arithmetic unit for computation, are also developed in G16. For electronic reasons diode gates develop the inverse of these signals and the 0 signals themselves are then obtained by the use of inverters.

E_c is a signal derived in this unit to indicate the end of each iteration cycle in sixty integrator operation or the end of each even numbered

iteration cycle in thirty integrator operation. In either case E_c is a P_{38} pulse appearing once every 60 integrator times.

Two specific integrator times, I_{59} and I_{02}, are also developed in G16. These times are specific configurations of the outputs of the integrator counter modified by X'_y which causes these signals to be independent of the state of K_7 in thirty integrator operation. In other words, during thirty integrator operation I_{59} can be considered as being I_{29} while I_{02} remains I_{02}, but both signals will now appear twice every 60 integrator times.

P_{L1} and P_{L2} are timing signals used to select the integrator outputs that are to be sent to the graph plotters. The optional unit, G18, develops four more integrator times, I_{20}, I_{22}, I_{24} and I_{26}, and four associated signals, P_{A1}, P_{A2}, P_{A3}, and P_{A4}, that are used if additional graph plotters or a Differential Analyzer Coupler is to be employed.

X_y is a signal developed to qualify the loading of the T channel during readout. It is always high during 60 integrator operation but only high for even iteration cycles during 30 integrator operations. This signal will be discussed in more detail in the Fill and Readout sections.

5.5 I_{i-1} and I_i

G_2 is a gate which compares the fourteen outputs of the seven K flip flops with fourteen signals from the seven λ relays. The λ relays are dual coil relays located in the desk assembly and controlled through one set of coils during manual operations by the integrator selector switch on the console and through the second set of coils during automatic filling by the contacts of the tape reader. When all fourteen comparisons are false, I'_{i-1} is low, hence I_{i-1} is high and will remain high for one complete integrator time. At P_{38} I_{i-1}, I_i (G6) is turned on and remains on for one integrator time.

The integrator counter effectively operates one step ahead of the arithmetic circuit, but in phase with the address circuits. While the integrator counter

reads 03, for example, the addresses for integrator 03 will be operated upon by the computer along with the arithmetic part of the preceding integrator, in this case integrator 02. Therefore, the I_{02} term in G16 is developed from the 03 configuration of the integrator counter.

5.6 Oscilloscope Display

Two flip flops, S_α and S_β, (G10) change state once each revolution at the end of I_i and are cyclical every four iterations. A bridge electrically sums the outputs of these flip flops so as to provide a single signal which varies with the changing states of S_α and S_β. This signal, S_v, provides the vertical deflection voltage for the oscilloscope.

The following list gives the state of the flip flops associated with the four sweeps, the bottom state with the bottom sweep, etc.

$S_\beta \quad S_\alpha$
$S_\beta \quad S'_\alpha$
$S'_\beta \quad S_\alpha$
$S'_\beta \quad S'_\alpha$

G8 provides the oscilloscope sweep trigger, S_T. S_T is high during I_i, if the display switch is at any of the arithmetic positions I, Y, D, or R. whereas it is high during I_{i-1}, if the display switch is at any of the address positions, ΔI, ΔX, Δ1, or Δ2. S'_T is sent to the oscilloscope since an amplifier within the oscilloscope serves as an inverter.

Each of the four states of S_α and S_β is associated in G9 with one of the arithmetic channels, if the display selector is at an arithmetic position, or with one of the Address lines (or the four signal probe), if the display selector is at an address position. In addition the T channel, which holds the marker, is seen on all four lines if addresses are being filled (this is accomplished by the $\oplus T (S_B)$ term of G9). This signal, S_D, is then gated with S_T in G7 so as to provide high signals during the desired integrator only. It is further qualified in such a manner that the final result, S_c, can be high only during one of the 0

times if the arithmetic channels are being observed. The particular 0 time and, hence, the particular information displayed, Y_i, Y, Y_D, or R, is determined by the position of the display selector. If any one of the address channels is selected, all four of the address channels are displayed for the particular integrator selected. F_i, also of G7, is used in conjunction with manual filling and will be discussed with the Fill Unit.

5.7 Multipliers

The multiplier digits are held in P_{38} of Y_1 and Y_2, as may be seen from Figure II. Being in the address channels, they refer to the following integrator and must be held in flip flops during the computation period of that integrator. G13 provides for this storage and also for decoding. M_1 and M_2 hold the multiplier digits while M_I, M_{II} and M_{III} read the information held. $M_I = M_1 M_2'$ signifies a multiplier of 2, $M_{II} = M_1 M_2$ signifies a multiplier of 5, and $M_{III} = M_1 M_2$ signifies that the integrator is coded for typeout control.

If M_I, M_{II} and M_{III} are all false, indicating that there was neither a pulse in position P_{38} of Y_1 nor position P_{38} of Y_2, the multiplier to be used for the following integrator is unity.

5.8 Initial Condition Reset

The reset operation is controlled by two flip flops, R_y and R_z (G11). The signal initiating the operation may originate in one of three places. The first, manual reset, results when the Insert button is depressed, (making (R_I) high), the second is automatic when the integrand of integrator I_{59} exceeds one in absolute value, and the third, R_w, is associated with the processing of reversible input functions. Any of these signals turns R_z on, which in turn sets R_y on at P_{38} of I_{59}. After R_y has been on for one iteration cycle, R_z is turned off if the Reset button has been released or, if the reset is automatic, when R_w is off. The A_f' term of αr_z insures that reset will continue until automatic fill has been completed in case a reset signal occurred before the completion of a read-in operation. After one more cycle R_y is turned off. R_y, then, has been on for at least two complete iteration cycles which is the time required for Y_1 to step through Y to Y_D.

G14 develops the actual timing signals for the reset, S_r for Y and Y_D, and R_r for R, while U_r indicates normal operation. Y_D and Y are reset as they enter the W flip flops in the arithmetic unit, that is, at the end of 0_3 and 0_4 respectively. S_r, then, is high if Q_1 and R_y are high and either R_I is high (indicating a manual reset) or S_A is high (indicating automatic reset initiated by the reset integrator). In addition, Y_D is reset to Y each time a non-zero dX occurs during computation, by the $0_3 X_e$ G term of S_r.

R must be reset during 0_2, but only if R_g is high. R_g is a signal (G1 of the arithmetic unit) indicating that some numerical information has been entered into the Y_i digit immediately preceding the R digit being processed. Since there is a two pulse time delay between the gate forming R_e and the inputs to the W flip flops and considering that during the automatic fill operation only the Y_i digits are filled leaving all other pulse times void of numerical information, it can be seen that the R_e signal could not be used in place of R_g. R_r is further qualified by $(R_I + R_r \cdot S_A)$. The first term indicates manual reset while the second term applies to reset integrator operation during automatic reset. R is set automatically only if R_r is high, which happens if the Read-In switch is in the normal position. When input functions are employed, R is not reset.

If R_r is high, each R digit must be set to 0011 except for the round-off digits which are a function of the output multiplier. The following table indicates the round-off digits; the pulse times referring to the timing of the information into the W flip flops.

M_2	M_1	Mult.	P_{35}	P_{31}	Value of R
0	0	1	1000	0011	.50
0	1	2	0101	1000	.25
1	0	5	0100	0011	.10
1	1	Typeout	0011	0011	.00

The remaining R digits (entering the W flip flops at the end of pulse times 7, 11, 15, 19, 23 and 27) are set to 0011. G12 develops these reset values for the R registers.

5.9 T_α, F_Z and L_Z

T_α of G1, and F_Z and L_Z of G3 are associated with fill operations and are included in the discussion of the Fill Unit.

FILL UNIT

6.1 Introduction

The Fill Unit (figure VIII) provides facilities for entering information into the computer either manually or from a punched tape. Provision is included for reversible function tapes as well as for monotonic function inputs and initial loading.

During manual fill the multi-purpose T channel keeps track of the specific digit that is in the process of being filled. A pulse or marker is set up in this channel and then shifted to the pulse time that corresponds to the position of the information that is to be filled. This marker pulse in the T channel is then recorded on the specific channel or channels being filled, and the process is then repeated until all of the desired information is filled. The marker pulse is entered and shifted in the T channel by G16 under the control of G12, G13, and G14. G10 and G15 provide the input terms for the arithmetic channels, and G20 provides the input terms for the address channels.

For automatic filling, information from a tape reader operating at approximately 6 steps or digits per second must be transferred to the computer which is operating at a speed of approximately 100 iterations or 228,000 digits per second. A buffer system between the reader and the computer is therefore provided in order to satisfactorily accomplish this transfer of information. Besides utilizing various circuits in the computer itself this buffer system consists of a commutator type timer, a stepping switch and auxiliary relays. The timer first steps the tape reader and then notifies the computer that the next code or digit is available to be read. The stepping switch advances from one position to the next as the reader advances from one code to the next. The various positions of the stepping switch are used to indicate to

the computer the meaning of the corresponding code on the tape. Electronic circuits then read the stepping switch positions and the α relays (relays excited through the tape reader contacts) and record the information in the correct location on the magnetic drum. G7 controls the starting and stopping of the automatic fill process. G3 and G6 provide a special control function over G7 during the filling of reversible input functions. G1 controls the direction of the tape reading operation, while G4 detects the reversals in this reading operation. G2 and G5 control the entry of the reader contact information into G15 to be used as the input to the arithmetic channels. G18 and G19 are used to translate the information from the tape that is to be entered in the address channels. G8 provides the necessary control over initial condition reset after inputs are filled in the Y_i registers. The clear terms for the address and Z channels are developed by G9; the arithmetic channels are cleared through G10. G17 has several functions, the normal one being to hold the indication of servo operation. The second terms of G17 enable S_A to serve as an indicator showing which integrators are to be reset during initial condition reset. The third function of G17 occurs during fill when it is used to effectively lengthen the T channel in order that the marker may be advanced to the left.

6.2 Manual Fill

The number of the integrator to be filled is entered in the Integrator Switch. This switch controls the χ relays, Figure VIII C, which hold the integrator number in the excess-three code. The output of the χ relays determine I_{i-1} which in turn determines I_i (See control unit).

The marker pulse is inserted at P_{36} by the $\emptyset$ M_I S_T G P_{36} term of G16 (during gross read-in and gross read-out the marker is inserted at P_1 by the $\emptyset$ M_I S_T G P_1 term). S_T insures that the marker is inserted at the

correct integrator time. The marker is recirculated during fill by the $(M_I')(\phi)\, T\, (G_2' * G_1')(C_A')$ term. Note that the insertion of a marker will clear the T channel of all other information as the recirculation term is qualified by (M_I'). Depressing the clear button causing (C_A') to become low totally clears the T channel.

The marker is moved to the right by being copied from T_{al}, that is, one pulse time early; and it is moved to the left by being copied from S_A one pulse time late. This is accomplished by the $(\phi)\, G_2\, G_1\, [S_A((M_L) + M_g) + T_{al}\, J_a]$ term of G16 only during fill and during the time both G_2 and G_1 are high. J_a (developed in G14) is true if the marker right button, (M_R), is pressed; or if any of the decimal keys including the Null Bar, (L), is pressed when S_A is true signifying that the DISPLAY switch is at an arithmetic position. It is seen that the marker automatically advances to the right while arithmetic information is being filled, but only advances when filling address information if the marker right key is pressed. As will be seen later, the marker advances four timing positions at a time when the arithmetic channels are being filled. The marker is moved to the left manually by the (M_L) switch. The M_g term is used during gross operations and is explained in the discussion of gross read out and read in. The $(\phi')\, T_{OT}\, X_{\text{[illegible]}}$ term of G16 is used to load the T channel during read out (see Read Out Unit).

The G_1 flip flop (G13) is turned on at the end of an interation cycle (E_c true) whenever G_2 is low and M_x (G12) is high. M_x is set high:

1) during a gross operation (M_g true)
2) when the Marker Left Button, (M_L), is pressed
3) if J_a (G14) is high
4) during compute ((C) is high when the Fill-Idle-Run Switch is in the Run position) provided typeouts that are too closely spaced do not occur; $(C)\, T_f'$ is an interlock which causes computation to stop,

permitting the read out operation to catch up.

With G_1 on, G_2 is turned on at the end of the following iteration cycle. When a digit is being manually filled into the arithmetic channels, the marker, which was inserted at an O_3 time (P_{36}), is moved to the right four positions to the next O_3 time. Since the marker pulse can only advance one pulse time per iteration cycle, G_2 and G_1 must be kept high for 4 iteration cycles in order for the advance to the next O_3 time. During this fill operation G_1 is turned off by the $\emptyset G_2\ O_4\ X_y\ T_o\ S_A^i$ term of G13. The O_4 rather than the O_3 signal is used since the content of the T_o flip flop is one pulse delayed from G16. It is seen that during the first iteration cycle after G_2 becomes true, the marker pulse will be in T_a at an O_2 time and in T_o at an O_3 time. Since there is no coincidence between T_o and O_4 the G_1 flip flop will remain on and the marker will again step one pulse to the right during the next iteration cycle. During the cycle where the marker takes its fourth step, there will be a coincidence between T_o and O_4 causing G_1 to go off preventing any further advancement of the marker pulse. Four iterations are counted in a similiar manner if the marker is moved to the left (during the first iteration cycle after G_2 becomes true, the marker pulse is held in S_A at an O_4 time and in T_o at an O_1 time). During thirty integrator operation the T channel does not change in length; and, consequently, 8 iteration cycles corresponding to four complete drum cycles are required to correctly move the marker as discussed above; X_y (see Control Unit) is, therefore, used to qualify this "off" term of G_1.

The remaining terms in the ${}_oG_1$ equation are associated with the following computer operations: I_D is high when the Fill-Idle-Run switch is in the Idle position and acts to turn G_1 (and hence G, see section 6.3 and G11) off,

stopping computation after one machine cycle. T_f (see Read Out Unit) turns G_1 off if a typeout is signaled before a previously signaled typeout is completed. (G) is used to one cycle G_2 and G_1 so as to step the marker one position at a time during gross operations. Similarly, $(S_B)(\emptyset)$ causes the marker to be stepped one position at a time when addresses are being manually filled. Finally, $R_o\ P_1$ turns G_1 off stopping computation when an overflow or proscribed code occurs in the Arithmetic Unit.

G_2 follows G_1 off at the end of the next iteration cycle provided M_x (G12) is also off at this time. The M_x' term of ${}_0g_2$ and the G_2' term of g_1 insure that the marker will not continue to step as long as a control button is held down; the button must be released turning M_x off, causing G_2 to go off, before G_1 can be triggered on again.

During fill, $(\emptyset)$, with G_1 and G_2 true F_o (G10) is set on, permitting new information from G15 to be filled into the four arithmetic channel in the place of the information coming from the drum (see Arithmetic Unit, G6). At this time F_o is triggered on by a concincidence of T_{a1} and F_i. F_i (see Control Unit, G7) depends upon the position of the Display Switch and indicates which arithmetic register is being filled. If this switch is, for example, at the Y position, F_i is true during O_1 so that F_o is true during O_2 (the correct time for entering a digit in the Y register). T_{a1} indicates the marker position, and hence, as used in G10 determines which specific Y_i, Y, Y_D or R position is to be filled. F_o is turned off one pulse time later by the F_o (C_A') (C_{I_i}') term. The arithmetic channels are also cleared by action of G10. If the Clear Complete Button, (C_A), is pressed F_o comes on and remains on until (C_A) is released; F_o true blocks recirculation at G6 of the Arithmetic Unit. If the Clear Variable Button is pressed, F_o is turned on by the I_i (C_{I_i}) term and off by the F_o (C_A') I_i' term; F_o being on during I_i causes the four arithmetic channels to be erased during this same integrator time. The remaining

terms of G10 are used during automatic fill and reversible function inputs and will be discussed in those sections. G15 is a decoding matrix for transforming the decimal digits from the keyboard into their excess-three code equivalents. The terms of G15 qualified by A and V_1 are used to read the tape during automatic fill and will be further explained in a later section.

If address information is being filled, G_1 is turned on only by the marker left and right keys, (M_L) and (M_R), and not by any of the decimal keys. Consequently, the marker pulse does not automatically advance when addresses are being filled or changed, but must always be purposely stepped to the next desired position. G_1 is turned off after only one iteration cycle by $(S_B)(\emptyset)$. With the marker in the correct position, information is entered into the address channels by T (S_{y2}) and analogous terms. The four signals (S_{y2}), (S_{y1}), (S_x) and (S_I) are each "and" combinations formed by interconnecting contacts on three Console switches. If (S_{y2}), (S_{y1}), (S_x) and (S_i) are defined as signals true only when the Scope Display Switch is in the $\triangle Y_2$, $\triangle Y_1$, $\triangle X$ or $\triangle I$ positions respectively, and (1) is defined as true only when the 1 key is pressed down then

$$(S_{Y2}) = (S_{y2})(\emptyset)(1)$$
$$(S_{Y1}) = (S_{y1})(\emptyset)(1)$$
$$(S_X) = (S_x)(\emptyset)(1)$$
$$(S_I) = (S_i)(\emptyset)(1)$$

with $(\emptyset)$ indicating the fill position of the Fill-Idle-Run Switch. Similarly, $(0')$ being a signal false when the 0 key is pressed down, $(S_{Y2}') = (S_{y2})(\emptyset)(0')$, etc.

The Y_2^* (S_{Y2}') and similar terms are the normal recirculation terms for the four address channels. ($Y_2^* = Y_2\, C'_{AD}$ etc., see Address Unit). The Y_2^* T' and similar terms permit recirculation of all pulses with the exception of the pulse in the marker pulse position. With the pressing of the 0 key,

recirculation via the normal recirculation terms stops; consequently, by proper positioning of the marker pulse any single address pulse may be erased. The address channels are cleared by C'_{AD} of G9; (C_A) causing a complete clear while $(C_{I_{i-1}})$ f_{i-1} clears only the address channels of the integrator selected. (The $\boxed{14}$ I_{i-1} term of G9 concerns automatic fill and will be discussed in that section).

The Y_{2F} , Y_{1F}, X_F and I_F terms in G20 are the automatic fill terms while the F_g terms are for gross read in. Finally the R_o GP_1 term in y_{20} acts to identify the integrator in which a proscribed code or an overflow occurred by placing a pulse in P_1 of Y_2 in the integrator that immediately follows the error. Since the scope display shows the addresses in I_{i-1} rather than in I_i, after the error indicator has been located in the Y_2 channel the display switch must be set to an arithmetic position and the integrator selector switch turned back two positions to observe the actual error.

6.3 Compute

After a problem has been filled into the computer, G true (gate 11) will start computation. $G = G_1G_2(\emptyset)R_y'$, $(\emptyset)$ signifies that the Fill-Idle-Run Switch is not in the Fill position. R_y' (Control Unit G11) insures that reset is not taking place. G_1 and G_2 are caused to go true by G13 together with the (C) T_f' term of G12 when the switch is in the Run position; J_a in G12 turns on the two "Go" flip flops when the 0 key is pressed during Idle (provided the DISPLAY switch is set to an arithmetic position), allowing one cycle operation. Note: If the DISPLAY switch is set to an address position, (S_A) will not be true and one cycle operation must be initiated by operating either the Marker Left or Marker Right key.

6.4 Automatic Fill

Automatic Fill is initiated by setting the Read In Switch to the F position and pressing the (A_f) push button which is located in the middle of this switch. Gate 7 shows that (A_f) turns A_f on energizing a relay driver which in turn activates the γ_2 relay, shown in Fig. VIII-b. When this relay closes, the release magnet of the clutch controling the Fill Commutator is energized, starting the commutator. Also in series with the release magnet are a push button switch and a toggle switch (to permit one cycle operation for test purposes) and a set of normally closed contacts of the carriage return relay, C.R. After an integrator has been filled a CR code from the tape will indirectly operate the CR relay which will stop the fill commutator from turning, thereby stopping the fill operation until the stepping switch has completed its cycle and returned to the "home" position indicating that the filling of the next integrater is to begin. During all fill operations except Gross Read-In Segment 1 of the fill commutator (Fig. VIIIb) steps both the tape reader and the stepping switch one step per rotor revolution. The reader is stepped directly but the stepping switch is operated by the θ relay which in turn is activated by the segment 1 signal through a set of contacts on the γ_2 relay. The stepping switch is not to operate during Gross Read-In and since γ_2 will not be activated during a Gross operation the above circuit hookup is used. (The read out commutator is used to control Gross Read-In; consequently, there is a common connection between the two commutators at the tape reader stepping solenoid. The above connection involving γ_2 prevents a "feed-through" of the read out commutator tape reader stepping signal from activating the θ relay and thereby causing the stepping switch to step.)

Segments 5 and 6 of the fill commutator determine the interval during which the tape is read after the new tape position has been set-up by segment 1.

Actually these timer segments activate the σ relay and the contacts of this relay supply the activate and control signals required for the reading operation.

6.4.1 Carriage Returns

The first character read from the punched tape is a Carriage Return (see Operation Manual sect. 1.9.3 Coding). This code is detected by the four fingers of the tape reader while four associated sets of contacts operate the four λ relays (see Fig. VIIIc). One set of contacts from each λ relay are inter-connected in such a way that a B+ signal, marked C.R. on Figs. VIII b and c, is transmitted to one side of the stepping switch home position sensing contacts whenever the tape reader reads a Carriage Return code. The Carriage Return relay is energized through this sensing contact, and it in turn energizes the stepping switch's automatic stepping contact with B+ that is also gated by the home sensing contact. When the armature has returned to home position the sensing contact opens, removing voltage from both the CR relay coil and the auto stepping contact. The CR relay contacts return to the NO positions with one set of contacts energizing again the fill commutator release magnet, permitting the rotor to turn and start the filling of the next integrator.

6.4.2 Selection of the Integrator to Be Filled

The next two digits appearing on the tape following the Carriage Return Code represent the number of the integrator that is to be filled. The first digit, which is the most significant digit, is read and held by relays λ5, λ6 and λ7 (Fig. VIIIc) while the second or least significant digit is read and held by λ1, λ2, λ3, and λ4. Each λ relay has two coils, one allowing the relay to be controlled by the integrator selector switches and the other allowing the relay to be controlled by the automatic fill process as described below. As the reader stepped to the code representing the most significant digit of the integrator number, the stepping switch advanced to the first position away from home, marked B in Figure VIIIb. As timer segments 5 and 6 are excited with a ground signal

from the rotor, the α relay closes supplying B+ to the moving contact of deck 4 of the stepping switch. This B+ signal is transferred through a set of NC contacts on each relay to one side of one set of coils on the λ_7, λ_6, and λ_5 relays. A ground signal is applied from the contacts of the tape reader to the other end of these relay coils if the fingers of the reader are reading a hole in the tape. The relays thus selected then close and an additional ground is applied to the ground end of the coils through another set of contacts on each separate relay, in order that a ground will remain ever after the reader steps to the next position. The contacts of the λ relays supplying the B+ signal to the coils are also wired to latch; i.e. after the relays operate the NO position of the contacts supply B+ from another source to the coils. This source of B+ used for latching is derived from a NO set of contacts of the β relay in series with a set of NC contacts of the CR relay. Hence, during the next Carriage Return the λ relay coil circuits are opened and the relays are prepared to read the next integrator number. The β relay is connected to follow the γ_2 relay if the Read In Switch is in the F (Fill) or M (Monotonic) positions. One set of contacts of β also de-activates the coils of the λ relays that are controlled by the Integrator Selector switches by removing B+ during the automatic fill operation, thereby transferring complete control of the relays to the second set of coils. During the next commutator cycle the λ_4, λ_3, λ_2, and λ_1 relays are similarly set up to hold the least significant digit of the integrator number; during this time the moving contact of deck 4 of the stepping switch is at the A position, thereby supplying only those λ relays connected to this contact with B+.

6.4.3 Integrand Filling

With the integrator selected, the sign of the integrand is next copied by V_1 (G5 figure VIII) with the stepping switch in position 4. Note that a space occurs on the tape between the integrator number and the sign code. In the sign sensing signal of G5, 4 α_2 N, 4 is a signal from deck 2 of the stepping switch, α_2 represents

the code indicating a negative integrand, and Ⓝ is a signal derived from either the F or M position of the Read In switch and a set of NO contacts on the σ relay. [4] signal is always false (see Fig. VIIIb) unless:

1. The moving arm of stepping switch deck 2 is on position 4
2. π relay is not energized indicating that the tape is not moving in the reverse direction.
3. σ relay is energized indicating that the rotor is on commutator segments 5 or 6 and that it is now the correct time to read the tape.
4. C.R. relay is not energized indicating No Carriage Return at the present time.

The integrand is now read during stepping switch positions 5 through 13 and placed in the correct position on the memory drum by F_o which is now controlled by the T_α I_i Ⓝ term of G10. T_α (Control Unit - G1) is high in O_2 times since F_o is to be high during O_3 times, the Y_i positions. Each term comprising the T_α signal is true for only one O_2 time per integrator, see Pulse Time Chart in section 5.3; the following list indicates the sequencing thus cre-ated:

STEPPING SWITCH POSITION	PULSE TIME OF I_i DURING WHICH T_α IS TRUE
5	35
6	31
7	27
8	23
9	19
10	15
11	11
12	7
13	3

The coding used for the various characters on the tape is as follows:

	α_4	α_3	α_2	α_1
0	0	0	1	1
1	0	1	0	0
2	0	1	0	1
3	0	1	1	0
4	0	1	1	1
5	1	0	0	0
6	1	0	0	1
7	1	0	1	0
8	1	0	1	1
9	1	1	0	0
Space	0	0	0	0
*No Action	0	0	0	1
-	0	0	1	0
C.R.	1	1	1	0
.	1	1	1	1

*Note: No action codes do not appear on manually prepared tapes.

$F_4 - F_1$ (G15) hold the digits of the integrand if $A = V_1' (G + A_f)$ is true (G2), signifying a positive number during automatic fill. This accounts for the $A\alpha_1$ and similar terms of G15 and the V_1 term of G2. If the integrand is negative V_1 is true and the $V_1 \alpha_1$ and similar terms of G15 cause $F_4 - F_1$ to hold the "nines" complement of the integrand. The (G) term of A insures that all of the information from the tape will be filled through the A signals of G15 during Gross Read-In. The digits held by $F_4 - F_1$ will be transferred to the Arithmetic memory channels by G6 of the Arithmetic Unit; this transfer is controlled by F_o (see above) which insures that the information will be correctly placed in the memory channels. If the integrand to be filled does not make use of the maximum capacity of the register, it is necessary to reset V_1 as soon as the least significant digit has been filled. Otherwise the spaces on the tape representing the unused capacity of the integrand register will be filled, through G15, into the four arithmetic channels as 1111, a proscribed code in so far as the Arithmetic Unit is concerned. The $T\alpha$ α_2' α_3' α_4' term of G5 recognizes the first space and resets V_1 as soon as $T\alpha$ comes true. Since V_1, as well as all of the α's, is now false nothing more will be filled into the arithmetic register.

6.4.4 Filling of Address Information

With the integrand entered, the address lines of I_{i-1} are cleared when the signal from stepping switch position 14 goes true (the I_{i-1} [14] term of G9 causing C_{AD} to go high). Note, however, that if a carriage return is the 14th character, the CR relay will close before the commutator reaches the 5th segment, which is explained on page , preventing [14] from becoming true. Also in this case the λ relays will become de-energized preventing an I_{i-1} signal; either [14] or I_{i-1} false prevents the address channels from clearing at this time. A carriage return at this time will also reset V_1, the sign flip flop ([CR] signal of G5).

6.4.4.1 Instruction Codes

If a carriage return was not received, the type of integrator operation is read from the tape at position 15 as shown in the following table.

Integrator Operation	α_4	α_3	α_2	α_1	Decimal Equivalent of Code
Rectangular Mode	0	1	0	0	(1)
Interpolative Mode	0	1	0	1	(2)
Extrapolative Mode	0	1	1	0	(3)
Multiplicative Mode	0	1	1	1	(4)
Adder	1	0	0	0	(5)
Servo	1	1	0	0	(9)

Referring to Figure II, it is seen that this information will be properly coded by the following procedure which is carried out by G18 and G20.

1. if $\alpha 1$, enter a pulse in Y_2 at P_{37} of I_{i-1}
2. if $\alpha 2$, enter a pulse in Y_1 at P_{37} of I_{i-1}
3. if $\alpha 3$, enter a pulse in Y_1 and Y_2 at P_6 of I_{i-1}
4. if $\alpha 4$, enter a pulse in I at P_{37} of I_{i-1}

Case 3 above enters both time and minus time as dy addresses which is the code for adder operation and causes the integrator to pick up the negative of its own

output as one of its dy inputs. The I_{i-1} terms of G20 enter the address pulses of G18 (Y_{2F}, Y_{1F}, X_F & I_F) into the memory channels at the correct integrator time.

Similarly for positions [16], [17], and [18] the following information is read and entered into the X and I channels by the [16], [17], and [18] terms of G18.

Position	Command	$\alpha 4$	$\alpha 3$	$\alpha 2$	$\alpha 1$	Typewriter & Tape Code
16	Normal Operation	0	1	0	0	(1)
	Initial Condition Reset Coding	0	1	0	1	(2)
17	Normal Operation	0	1	0	0	(1)
	Coding for Typeout	0	1	0	1	(2)
18	Normal Operation	0	1	0	0	(1)
	Coding for Ouput Sign Reversal	0	0	1	0	(-)

At position 19, the output muliplier is read from the tape as shown below and entered into the Y_2 and Y_1 address channels by the [19] terms of G18.

Multiplier	$\alpha 4$	$\alpha 3$	$\alpha 2$	$\alpha 1$	Typewriter & Tape Code
X1	0	1	0	0	(1)
X2	0	1	0	1	(2)
X5	1	0	0	0	(5)
Coding for Typeout Control Integrator	1	0	0	1	(6)

6.4.4.2 DX Address

At position 20 a space code appears on the tape and [F_L] is true for the first time (see Figure VIIIb). [F_L] will be true at positions 20, 23, 26, 29, 32, 35, 38, 41, 44, and 47. [F_N] will be true at positions 21, 24, 27, 30, 33, 36, 39, 42, 45 and 48. [F_S] will be true at positions 22 and 25. [F_Y] will be true at positions 28, 31, 34, 37, 40, 43, 46 and 49. The [F_L] signal turns off V_1 of G5, if it was on, and clears the Z line by setting C_Z high (G9). The most significant digit of the dx

address occurs at position 21. If this digit is a nine, signifying time as the address, V_1 is turned on by the $F_J \alpha_4 \alpha_3$ term of G5 with F_J being developed as $\boxed{F_J}\ P_{38}$ in G19. At position 22 the $\boxed{22}\ P_6 V_1$ term of G18 enters time as a dx address. Also, the $\boxed{22}\ P_1 R_w$ term records a pulse at P_1 in the X channel if minus time is the address. At position 22, for time as the dx address, the tape reader reads 0 for plus time or 1 for minus time. The $V_1 \alpha_3$ term of G8 turns R_w on if time is minus and the indication of negative time gets recorded as indicated above. Position 23 will be an $\boxed{F_1}$ signal causing V_1 and R_w to reset and clearing the Z line again.

Normally, however, the dx input will not be time, and in this case the most significant digit of the dx address will be held in $\alpha_3, \alpha_2, \alpha_1$ during the 21st commutator cycle. (see sec. 5.2 for the code used). F_Z in G3 of the Control Unit compares the states of $\alpha_3, \alpha_2,$ and $\alpha_1,$ with the three most significant digits of the K counter. The $\alpha_1 \alpha_3$ term of F_Z' assures that F_Z will be false if time is the input. Therefore, F_Z is true for the ten integrator times corresponding to the most significant digit of the dx input. $\boxed{F_J}$ indicates that the most significant digit of an address is being processed. When both $\boxed{F_J}$ and F_Z are true, the $F_J F_Z$ term in the dx_e equation (G8 in the Address Unit) turns the X_e flip flop of the Arithmetic Unit on at the end of the next P_{38} time. That is, since the K counter holds I_{i-1} instead of I_i, X_e will be turned on at the beginning of P_1 of I_i. With X_e staying on through P_{35}, I_i acts as if it has an output and causes a one to be recorded on the Z_1 line (G3 of Output Unit). During fill G' (G11) is high; hence, dz of G12 of the Arithmetic Unit is high, permitting the $P_{35} X_e dz$ term of G3 of the Output Unit to record the one. Hence, during the 21st commutator cycle 10 integrator outputs will be recorded and recirculated in the Z_1 memory line, one for each of the ten integrators corresponding to the most significant digit of the dx input. These outputs will appear on the Z_1 line as shown in Fig. II for any ten integrators possessing the same most significant digit; during any one inte-

grator time five of these outputs will appear in Z_1 and five in Z_2. The least significant digit is similarly processed during position 22. In this case G19 develops a signal $F_F = \boxed{F_s}\, P_{38}$ if a dx or dI input is being processed or $F_F = \boxed{F_y}\, P_{38}$ if a dy input is being processed. The binary digits held in $\alpha_4, \alpha_3, \alpha_2$, and α_1 are compared with the least significant digits of the K counter in L_Z (G3 of the Control Unit). L_Z is, therefore, true for the six integrator times that have the same least significant digit. The $F_F L_Z$ term of G8 of the Address Unit sets X_s high for these six integrators which in turn sets dz_s high (G12 of the Arithmetic Unit) as G' is true during fill. Consequently, 6 "integrator outputs" are recorded on the $\bar{z}_{1s}$ line (G3 of Output Unit); these outputs will appear in the z_{1s} line as shown in Fig. II for outputs of integrators possessing the same least significant digit (i.e. 6 pulses with one pulse appearing in every fifth pulse position). During any one integrator time these 6 pulses will appear at both z_{1s} and z_{2s}.

One of the Z lines now holds "ones" in the positions which correspond to those integrators having the same most significant digit as the address being filled, while the other Z line holds "ones" in all positions corresponding to the least significant digit. The next procedure is to insert a one in the X address line of the integrator being filled in the pulse position in which a coincidence occurs between the two lines as this is the position on the Z lines where the output of the integrator that is to serve as the dx input to the integrator being filled (I_i) will appear during I_{i-1} in normal computation. The above mentioned coincidence may occur either between Z_1 and z_{1s} or between Z_2 and z_{2s} and is detected in G8 or G9 of the Output Unit as Z_{F1} or Z_{F2}. This coincidence will occur at position 22 of the Stepping Switch for the dx address and is entered into the X channel by the $\boxed{22}\,(Z_{F1} + Z_{F2})$ term of G18 in conjunction with the $I_{i-1}\, X_F$ term of G20. If the coincidence Z_{F2} occurred, R_w is turned on by the $\boxed{F_s}\, z_{F2}\, I_{i-1}$ term of G8. The $P_1\, R_w\, \boxed{22}$ term of G18 now records a one in the X channel at P_1 time

indicating that the Z_2 and $\bar{Z}_{2s}$ lines instead of the Z_1 and $\bar{Z}_{1s}$ lines are to be read for determining the dx input. If time is the dx input this pulse at P_1 time in X indicates that minus time is to be used.

6.4.4.3 DI Address

After each address has been filled, the next position on the Stepping Switch sets $\boxed{F_L}$ high, clearing the Z lines and turning V_1 and R_w off. The next address which comes up in positions 24 and 25 is the dI address and its loading is exactly similar to the loading of the dx address. I pulses are entered in the memory by the $\boxed{25}$ term of G18.

6.4.4.4 DY Addresses

The dY address pulses are filled in next in a similar manner. Since two Y address lines are employed no code signal is needed in P_1 as is required in the X and I lines to determine which Z line (Z_1 or Z_2) to examine for the input. Y_2 always examines Z_2 and Y_1 always examines Z_1 for dY inputs. The most significant digit is again filled during an $\boxed{F_J}$ time, but the least significant digit is now filled during an $\boxed{F_Y}$ time (Fig. VIIIb). After the coincidence between the most significant digit positions in Z_1 or Z_2 and the least significant digit positions in $\bar{Z}_{1s}$ or $\bar{Z}_{2s}$ is made and indicated by either $\bar{Z}_{F1}$ or Z_{F2}, the correct address pulse is filled by G18 also during the $\boxed{F_Y}$ time (indicated by the $\boxed{F_Y}\,Z_{F2}$ term of Y_{2F} and the $\boxed{F_Y}\,\bar{Z}_{F1}$ term of Y_{1F}). If a dY input is to be time a nine will be coded as the most significant digit causing V_1 to be turned on as explained above under the dx input discussion. The codes for the least significant digits of time are as follows:

	EXCESS THREE CODE	DECIMAL EQUIVALENT
Plus Time	0 0 1 1	0
Minus Time	0 1 0 0	1

Therefore, if plus time is the input both V_1 and α_1 will be high during $\boxed{F_Y}$ and this term will then fill a pulse into the Y_1 line at $P_6\ I_{i-1}$ (G18 and G20). If minus time is the input V_1 and α_3 will be high during $\boxed{F_Y}$ entering a pulse into the Y_2 line at $P_6\ I_{i-1}$.

6.5 Filling During 30 Integrator Operation

The filling operation is the same during thirty integrator operation. However, P_6 now becomes P_{21} and the recirculation terms of the Z lines are changed from $P_{6/36}$ to $P_{21/36}$ (see G16 of Control Unit and Fig. 47 of the Operation Manual).

6.6 Termination of Filling

After the last address for a particular integrator has been filled, a carriage return code is read which activates the C. R. relay, advancing the Stepping Switch to "home" or zero position and preparing the χ relays to receive the next integrator number. After the last integrator has been filled, a period, $(\alpha_4)(\alpha_3)(\alpha_2)(\alpha_1)$, occurs which turns A_f off (G7), de-energizing the γ_2 relay and stopping the fill commutator ending the fill operation. This off term of A_f is qualified by (N) indicating that the computer is set for problem filling or monotonic function input operation by the setting to the F or M position of the Console Read-In Switch ((N) = (N)(G)). This off term of A_f is further qualified by V_3' which prevents A_f from going off until the Stepping Switch has made at least one complete cycle. Before the start of automatic fill A_f' is high, consequently, V_3 of G6 is true. However, V_3 will be turned off either the first or 2nd time 5 comes high depending upon the initial state of V_2, and it will remain off during the rest of the fill operation permitting the A_f "period" off signal to operate as desired.

The fill process can be stopped at any time by pressing the reset button on the Console. This makes (Co) true which turns A_f off stopping the commutator and resets T_d to the true condition while V_1 and R_w are reset to the false state.

After the tape has been read all integrands appear only in the Y_i register. Before computation can start the Insert Button must be pressed causing (R_I) to become true, which in turn transfers the integrands to the Y and Y_D registers and inserts the roundoff number in the R register as explained in Section 2.3. (R_I) also clears the Z lines of any information that may still remain following the final filling of address codes (G9). The machine is now ready for computation which may be initiated by setting the Fill-Idle-Run Switch to the Run position.

6.7 Function Input

6.7.1 Monotonic

Monotonic function input tapes are coded as described in Section 2.9.1 of the Operation Manual. The values of the integrands contained on the tape are entered into the computer into the Y_i register in the same manner and using the same circuits as described in section 6.4 for automatic fill. The Read In Switch is set on M making signal (R_f) true. The signal that ends the initial conditions reset process, $R_z' E_c Ry$ (Section 5.8 and G11 of Control Unit), turns A_f on to start the filling process (G7). After the first point of the independent variable has been entered, the period on the tape causes A_f to turn off as previously explained. At the end of the next reset the process is repeated by again turning on A_f. If a reset signal occurs before the completion of the previous read-in operation, R_z (G11 of the Control Unit) will not go off as A_f is still true. Consequently, reset will continue until the previous read-in is complete and A_f is turned off; this will then permit R_z to go off. After R_z goes off A_f will go on again at the next E_c pulse and read in the next function value; also at this time R_y will be turned off again and G11 of the Control Unit is ready to accept the next reset signal.

During automatic reset the integrands held in Y_i will be transferred to the Y and Y_6 registers in only those integrators that are coded for reset with a pulse at P_{38} in their I address. This P_{38} address pulse is read at the following P_1 time at the I_o record flip flop (I_o is one pulse time delayed from the I read flip flop) if the reset signal, R_y, is true. If the integrator is so coded, S_A (G17) will be turned on at P_1 time and off again at P_{37} but if the integrator is not coded for reset, and R_y is true, S_A will be turned off

at P_I time by the $\emptyset$ $I_o' P_1 R_y$ term of G17. Although S_A is the flip flop used to hold the indication of servo operation, the above additional functions of this flip flop do not interfere with its normal operation as computation is alwayss toppped during reset by the R_y signal in G11. The reset signal, S_r (G14 Control Unit), is qualified by S_A for automatic operation; hence, only those integrators so coded get reset.

6.7.2 Reversible

Reversible function inputs require considerable additional controls and circuitry than do monotonic function inputs. Also due to the nature of the read-in process only one integrator may receive function values from the tape, and this integrator has been permanently selected as integrator 02. With this type of input, by definition, the function controlling the tape read-in can go both positive and negative causing the tape reader to step forward and backward in accordance.

In both the monotonic and the reversible function read in process it is desired that computation not be held up while filling is going on any longer than is absolutely necessary. This is accomplished in the monotonic case by filling the n +1 st integrator values into the Y_i register and having the reader ready to read the n+2 nd values immediately after the nth values have been inserted into the Y and Y_D registers. When the n+1st values are called for, computation will only have to be stopped for two iteration cycles (G11 of Control Unit) in order for the insertion of the new values into the computation registers to be made. Computation is stopped during this inserting process by G11 (R_y will be true hence G will be off, see Section 6.3). The situation is a little different in the reversible function input case, although once a reversal has taken place the flow of information is to be the same as in the monotonic case. Let n-3, n-2, n-1, n, n+1, n+2 and n+3 be consecutive

values on the tape and consider the case when the nth value of a forward moving tape is in the Y register at the time a reversal signal occurs. Also at this time n+1 is in the Y_i register, and the reader is ready to read the next tape value, n+2. When the reversal occurs the Y and Y_D registers are to hold n-1, the Y_i register should hold n-2 and the reader should be ready to read n-3. The reader must, therefore, immediately step four increments in the reverse direction in order to get the correct values into Y, Y_D and Y_i registers. From this point on, the tape is to proceed one step at a time in the normal manner until the next reversal signal is received, when the reader must again change direction and immediately step through four values on the tape to get the correct values into the arithmetic registers. During this above mentioned stepping, computation must remain off in order to give the tape mechanism a chance to "catch up" to the computer.

For the information contained on a reversible function input tape see sections 2.9.2 and 2.9.3 of the Operation Manual. This tape is set into the reader at a carriage return code and the Insert Button is pushed turning on R_y (G11 Control Unit). The $(R_f) R'_{\neq} E_c R_y$ term of G7 turns on the A_f flip flop starting the reader to begin stepping in the same manner as for Monotonic functions (see Sect. 6.5.1). After the initial value is read into Y_i, the carriage return code on the tape turns A_f off stopping the reader (N') $\boxed{CR}$ $V'_2 V'_3$ term of G7, (N') being a signal developed from the Read-In Switch and only true when that switch is in the R position). The Insert Button is again pressed at this time transferring this initial value into the Y and Y_D registers as well as again turning A_f on and reading the next value from the tape into Y_i. The tape again stops on the next carriage return and the computer is ready to start computation.

It will now be assumed that the tape has been moving in the forward direction and will continue to do so until at least one more value has been read from the

tape into the computer. When the tape control function in the Y_D register of I_{59} becomes one or greater this integrator will have an output, and this output, M_D (G14 Arithmetic Unit), will start an automatic reset operation (G11 Control Unit). A_f also goes on as in manual reset, and the next value is read into the Y_i register. After one value from the tape has been read the (N') CR $V_2'V_3'$ term of G7 again turns A_f off stopping the operation. Here again, if a reset signal occurs before the function value has been read in, the ongoing read in must be completed before the A_f flip flop will go off; then A_f is turned on again as described in Sect. 6.5.1.

If the integrand of integrator 59 now goes negative, W_4 will be high at $P_{38}I_{59}$. Figure II shows that Y_s, the sign of the integrand, is read in the R flip flops at P_{35} time and, consequently, in the W flip flops three pulse times later. Since (N') is always true during Reversible Function Input operation (see above) and T_d is true if the tape has been moving in the forward direction, R_M of G4 will now be true at $E_c = P_{38}I_{59}$ time. E_cR_M will now turn R_w on (G8). With R_w on and A_f still off, the tape direction control flip flop, T_d , is turned off (G1) , indicating that the tape is now to be run in the reverse direction. G21 is simply a gate used to develope the $R_w A_f'$ signal and supply the output with a power amplifier as the signal is to be used in several locations. T_d' energizes a relay driver which closes the π relay of FigureVIIIb. This relay has two functions one of which is to operate the tape reverse solenoid on the tape reader to mechanically permit the reader to step in the reverse direction. The second function of the π relay is also concerned with Reversible Function Inputs but will be covered a little later in this discussion. This R_wA_f' signal also turns A_f on (G7) starting the Stepping Switch and Tape Reader to step, and at this time the R_w (N') term of G11 of the Control Unit turns on R_z thereby starting the reset operation. V_2, V_3 and R_w now serve as a counter to count the number of Stepping Switch cycles and hence the number of values read from the

tape. The tape will keep stepping even after the first value is read since the normal off term during Reversible Function Input operation is qualified by V_2' and V_3' (G7) and V_2 is now true, being turned on by $R_w A_f'$ (G3). With the π relay energized, that is T_d' true, the sequencing of the first few stepping switch signals are reversed and now appear in this order [12], [11], [10], [9], [8], [7], [6], and [5] (see Figure VIIIb Stepping Switch decks numbers I and II together with the Rotary Switch). The reason for this reversing will be explained later in this section.

Since T_d' is true, the $V_2 T_d'$ [5] term of G8 will turn R_w off during the first cycle of the stepping switch. [5] now occurs after [6], hence, the R_w' [6] term of G3 will not turn V_2 off until the second stepping switch cycle. The $V_2' T_d'$ [7] term of G6 is to turn V_3 off, and it cannot occur until the third stepping switch cycle. The carriage return signal occurring after the third cycle, that is after both V_2 and V_3 have been turned off, will now reset A_f to the off condition. As A_f goes off R_z of the Control Unit goes off, and, hence, E_c will again turn A_f on causing the stepping switch to start its fourth cycle and the reader to enter the fourth function value into Y_i. A_f' also turns V_3 back to the on condition (G6); but since V_2 remains off, V_3 will be turned off again during this fourth cycle of the stepping switch. Hence, before the reader gets to the next carriage return both V_2' and V_3' will be simultaneously true, permitting the ensuing [CR] signal to again turn A_f off. The reader will now step over just one function value each time I_{59} has a negative output, just as it was reading only one value on the tape for each positive output of I_{59}.

If after the tape has been stepping in the reverse direction the integrand of I_{59} goes positive, the direction of tape travel must be changed to the forward direction again. In this case the sign digit, Y_s, of I_{59} will change from nine to zero, hence, W_4' goes from true to false at P_{38} I_{59}. A reversal to the normal direction is therefore signaled by (N') $T_d' W_4'$ in G4. This expression is

further qualified by S_b, a signal indicating that Y_D is not zero (Sect. 2.5.3), for the following reason. The normal procedure is to set I_{59} to zero at each reset signal; but the sign of zero is positive and if the tape drive is in the negative position, a reversal would occur on the first cycle of computation. Therefore, it is necessary to discount the zero position, and this is accomplished by the S_b term. Now $E_c R_M$ will turn on R_w and the combination $R_w A_f'$ is true again. This time, however, $R_w A_f'$ turns T_d on (G1). Reset is started, and A_f and V_2 are turned on as before during the previous reversal. However, since T_d is now true causing the tape to step in the forward direction, the stepping switch signals will become true in correct numerical order, i.e. [5], [6], ... [12]. R_w is now turned off by $V_2 T_d$ [7]; V_2 is turned off the following stepping switch cycle by R_w' [6]; and V_3 is turned off during the third stepping switch cycle by $V_2' T_d$ [5]. A_f and R_z are turned off as during the previous reversal, and then A_f is turned on again this time by the $(R_f) R_z' E_c R_y$ term. After the next value has been read the carriage return signal turns A_f off, stopping the read in process until the next reset signal is received. It is seen that during this last reversal the reader stepped over four different function values from the tape as before and as required as explained above. The reader will now proceed in the normal manner entering only one function value each time I_{59} has a positive output until the next reversal signal is received.

When the tape reader is stepping in the reverse direction, the first decimal digit, that is read from the tape is the least rather than the most significant digit. This digit, therefore, must be placed at a different pulse position within the integrator depending upon the number of digits comprising this integrand. The DIGITS switch, shown in Figure VIIIb, must previously be set to the number corresponding to the number of digits of the integrand appearing to the

right of the decimal point, exclusive or the round-off digit. The following chart shows the repositioning that takes place.

STEPPING SWITCH POSITION	"FORWARD" SIGNAL DEVELOPED	"REVERSE" SIGNAL DEVELOPED WITH DIGIT SWITCH SET TO				
		6	5	4	3	2
1		12	11	10	9	8
2		11	10	9	8	7
3		10	9	8	7	6
4	4	9	8	7	6	5
5	5	8	7	6	5	
6	6	7	6	5		
7	7	6	5			
8	8	5				
9	9					
10	10					
11	11					
12	12					
13	13					

Electrically this change over is effected by transferring ground from the 2nd pole of the stepping switch to the 3rd pole, by means of the second set of contacts of the π relay (see Figure VIIIb.) The operation of the π relay is covered earlier in this section. Now, depending upon the position of the DIGITS switch, the lines carrying the position signals [5] through [12] are switched to new points on the stepping switch, as shown in the figure.

If a reversal or a read in signal occurs during the reading in of a function, R_w is turned on; however, as has been shown, it is ineffective until A_f goes

off. The normal reversal or read in procedure takes place again as soon as A_f goes off with $R_w A_f'$ turning A_f on again at the next clock pulse.

The tape reader can be stopped at any time by pressing the Reset button on the Console. This makes (C_o) true resetting A_f to off which in turn resets V_3 to on. (C_o) also resets T_d to on and R_w to off, thereby returning all of the circuits used to their initial conditions.

During tape reading while in Reversible Function Input operation, the information from the tape is entered into the computer in the normal manner through G15 under the time control of F_o of G10 (see Section 6.4). The controlling term of F_o now is T_{α} $I_{\phi 2}$ (N') permitting information to be filled into only integrator 02 during this type of computer operation.

6.8 Gross Read In

For a complete discussion of Gross Read In see Section 7.5 in the chapter on the Read Out Unit. The F_g terms of G20 in the Fill Unit are for the purpose of entering address information during Gross Read In. The arithmetic information during this process is entered also via G15 under the control of A_f, G2, and F_o of G10. The (G) V_1' term $(A=V_1'A_f+(G)V_1')$ is now the effective term of G2 and $T_{a1}G_1G_2(\phi)F_i$ is the effective term controlling F_o. (ϕ) must be true during Gross Read In as will be seen later.

6.9 Servo Indication

The primary purpose of S_A is to hold the indication of servo operation. Figure II shows that servo integrators are coded with a pulse at P_{37} in the I channel. This pulse will be in the I channel record flip flop, I_o, at P_{38} time, and hence, is detected by the (ϕ') P_{38} I_o term of G17. S_A will be turned off by the following P37 pulse. (ϕ') is a signal signifying that the Fill-Idle-Run switch is not in the Fill position. S_A is used for servo

operation as described in Section 2.5.3.

6.10 Clearing of Z Lines

The Z lines are manually cleared through G9 by the Clear Complete switch signal, (C_A), as described in Section 4.4 (operating this switch causes (C_A) to go high and hence C_Z' to go low). (C_A) also clears the address lines through G9 as previously discussed in Section 6.2. The $R_y S_A$ term of G9 is for selective clearing of the Z lines after initial conditions have been automatically reset in those problems where these values are automatically varied in accordance with results obtained during computation. Typical problems of this type would be those involving split boundary conditions. The (R_r) signal is one from the Read In switch and is true only when that switch is in the N (normal) position. S_A is true only during those integrators coded for automatic reset (see Section 6.7.1); hence, C_Z will be true, C_Z' will be false, during an automatic reset for only those integrators so coded. Since G also goes off during automatic reset, the C_Z' terms of G8 of the Address Unit will cause these resetting integrators to act as if they have no output; and, hence, cause any previous output that they might have had to disappear from the Z lines. See also Section 4.4. This in effect erases the Z lines of all outputs from those and only those integrators being reset.

READ OUT UNIT

7.1 Integrand Read Out

If the Read Out switch is set to either N (Normal) or R (Reversible), integrands from the Y registers of those integrators so coded will be read out of the computer each time the integrator coded for typewriter control overflows. If a reversible tape is to be prepared only one integrator should be coded to type out as only values for one integrator, I_{02}, may be fed back into the computer during Reversible Function Input Operation.

To facilitate integrand read out and to permit the arithmetic circuits to continue computing during read out, the values to be read out are first loaded into the T channel in a serial manner and then read back to a decoding system that controls the typewriter and the tape punch. Since the values of interest are now recirculating in the T channel, the rest of the computer circuits may continue with the problem solution. If the integrands are negative they are held in the computer as complements. Therefore, circuitry is also provided in the Read Out Unit to recomplement these complements so that true negative numbers may be typed out.

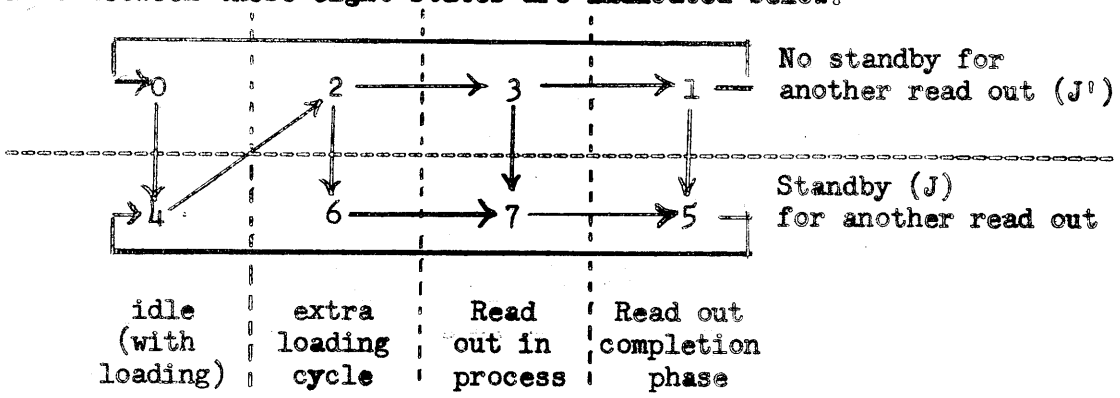
A read out commutator (Figure IX) is provided to control the operating sequence of the typewriter. This is also used during gross read out and gross read in as will be seen later.

7.1.1. J, T_T and T_R Flip Flops (G13)

J, T_T and T_R are three flip flops that indicate the state of the read out unit. These states are shown below.

State	J	T_T	T_R	
0	0	0	0	idle (T channel loading)
4	1	0	0	idle, standby to read out (T channel loading)
2	0	1	0	extra cycle of loading before a read out
6	1	1	0	extra cycle of loading for one read out with standby for another read out
3	0	1	1	read out in process
7	1	1	1	one read out in process with standby for another read out
1	0	0	1	read out process completion phase
5	1	0	1	one read out process completion phase with standby for another read out

The transitions between these eight states are indicated below:



The normal cycle for read out may be 0→4→2→3→1→0. When this is the case, each read out is completed before the next one is signaled by the computer and computation need not be halted.

All read outs are initiated by turning J on in gate 13. This may be done manually during state 0, either by pressing the read out start button which is located in the middle of the Read Out switch or by pressing the selective integrator read out button which is located in the middle of the most significant digit Integrator Selector switch. Both of these switches make (P) true, when operated, starting a read out. The output of an integrator coded for typeout control will also turn J on; this is accomplished by the $GM_{III}\ Z_a\ P_{37}$ term of G13. M_{III} is the typewriter control integrator indicator and is derived, in G13 of the Control Unit, from the address code pulses, while Z_a represents the integrator's output.

State 4, therefore, may be set anywhere during a machine cycle. At the end of the machine cycle during which state 4 is set, state 2 is set by the $J\ T_T'\ T_R'\ E_c$ term which sets J off and T_T on. State 2 lasts until the end of the next machine cycle, when state 3 is set ($t_R = T_T\ E_c$). T_T is later set off when the last information loaded in the T channel is read out (indicated by the $T_W'\ E_c\ T'$ term). Setting T_T off at this time gives state 1, which is changed to state 0 when a completion signal is received from the read out commutator ($o^t R = T_T'\ T_g$).

During states 2, 3, or 1, J may also be set on again giving states 6, 7, or 5 respectively. T_f of G12 indicates the presence of any of these latter three states

and signals computation to stop (Fill Unit G13) until the read out in process is completed. The chain of states, 2→3→1→0, will then be replaced by the chain 6→7→5→4, and at the end of the machine cycle when state 4 occurs in this chain, computation will be restarted (Fill Unit G12) and the "stand by" read out started.

The state of the read out may be set to 0 at any time by pressing the reset button which makes C_o true resetting J, T_T and T_R to the off state.

7.1.2 T_K, T_L, T_M and T_N Flip Flops (G5)

T_K, T_L, T_M and T_N flip flops are used during loading to receive the Y digit codes in parallel from those integrators whose integrands are to be typed out and then step these digits serially into the T channel through G1. These flip flops are then used during the actual read out to step the Y digits in the T channel to the left (shifted so they appear later in time). This shift is brought about by effectively increasing the length of the T channel 4 pulse times (equivalent of one decimal digit) during the integrator that is being typed out at the present. The integrand in this integrator is shifted one decimal digit per commutator revolution. After each shift these four flip flops store the next output digit that is to be typed until it is typed, after which they are cleared and made ready for the next shift.

T_K, under the control of T_c, is also used to complement negative integrands. Complementing is done per integrator at the same time the first shift is made prior to typeout.

7.1.3 T_G and T_W Flip Flops

T_G and T_W are used during the read out process to synchronize the Y digit shifting and selection with the position of the read out commutator. These flip flops permit shifting to occur in only the first non-empty integrator of the T channel, and the shifting in that integrator to be limited to a 'one decimal digit shift' per commutator cycle.

7.1.4 T_p Flip Flop

The T_p flip flop serves to control the loading of typeout integrands. During the read out process it serves to control the typeout of the sign digits of the integrands.

7.1.5 T_I and T_c Flip Flops

The T_I and T_c flip flops control the complementing of negative integrands in the T channel during the read out process.

7.1.6 T Channel Loading

The T channel is loaded, with the integrands that are to be typed out, through G1 (Fig. IX) and the $\phi^0 T_{OT} X_y$ term of G16 of the Fill Unit (Fig. VIII). The loading term of G1 is $T_R' (T_M + T_{p1} + T_{p2} + E_c)$; in other words, loading is continually taking place except when T_R of G13 is true (Section 7.1.1).

All integrators to be typed out automatically are coded with a pulse in the X address channel in the P_{38} position. This address code is read by the $T_R' P_{38} I_T^0 X^*$ term of G3, which in turn sets the T_p flip flop on starting with the beginning of P_1 of each integrator that is to have its integrand typed out. The I_T^0 term is a signal from the Console signifying that the selective integrator read-out button has not been pressed. T_p may also be turned on for any one integrator by setting the Integrator Selector switches to the desired integrator number and pressing the selective integrator read-out button (the $I_T I_{i-1} T_R' P_{38}$ term of G3). With T_p on and T_R off, G1 causes pulses to be recorded in the T channel at P_1 and P_2 times for every integrator that is to be typed out. These pulses at P_1 and P_2 cause spaces to be typed, if a reversible function tape is being made, as will be seen in Section 7.1.7; these spaces are necessary to permit automatic preparation of these reversible tapes (see Operation Manual Sections 2.9.2 and 2.9.3). Since reversible tapes do not contain any integrator number or integrand sign, four spaces or no action codes must be included on the tape between the carriage return codes and the most significant digit of the integrands. This is necessary in order for the correspondence between the digits on

the tape and the various stepping switch signals to be the same regardless of the type of tape being filled. Besides the two spaces punched on the tape for the P_1 and P_2 pulses that were loaded into the T channel, two no-action codes will also be punched as described in Section 7.1.7. One no-action code will be punched immediately following the punching of the carriage return after the integrand value has been read out, and the second no action code will be punched at the start of the read out for the next value of the function. During normal read out the pulses recorded in T at P_2 times initiate the setting up in G5 and the typeout of the signs of the integrands.

With T_p on the $T_R' T_p O_2 R_f$ term of L_f, developed in G8, is used to gate the parallel Y digit codes, exclusive of the round-off digit, from the R_4, R_3, R_2 and R_1 flip flops of the Arithmetic Unit into the T_K, T_L, T_M and T_o flip flops. The gating of these Y digits into T_K, T_L and T_M is brought about by the L_f terms of G5, but the gating into T_o, the T channel record flip flop, is brought about by the $L_f R_1$ term of G1 in conjunction with the T_{OT} term of G16 of the Fill Unit. The contents of R_1 is also gated into T_N by L_f, but this is not a useful part of loading for normal readout, being used only during Gross Read Out as explained in Section 7.2. The round off digit of Y is also gated by G8 if output function tapes are being generated; this is designated in G8 by (T), a switch signal from the Console true only when the Read-Out switch is in the R position. The Y digits are held in the R flip flops at O_2 times (see Figure II); hence, they are in T_K, T_L, T_M and T_o at O_3 times. The stepping gate, G7, now shifts these Y codes from T_K to T_L to T_M to T_N for three successive pulse times, O_3, O_4 and O_1. {L_g of G7 equals $E_c' [T_W' + T_R' O_2']$. It is the O_2' term that causes the stepping just mentioned as $O_2' = O_3 + O_4 \times O_1$. This stepping term is qualified by E_c' in order to block the stepping operation at $P_{38}I_{59}$. If the stepping signal entered G5 at this time (E_c time) it would interfere with the setting up of the carriage return code in T_K, T_L, T_M and T_N at the end of typeout (see Section 7.1.7). This shifting is accomplished by the six L_g terms in G5 and the $T_R' T_M$ term of G1. The stepping of the contents of T_M into T_N is not a useful

part of loading, but is necessary during typeout as explained in Section 7.1.7. With the first shifting pulse ($T_R'0_3$), T_K is set off so that the second shifting pulse sets T_L off and the third sets T_K off, and in this way clearing the three flip flops in preparation to receiving the next Y digit code. The least significant binary digit of each Y code is loaded into the T channel by the L_fR_1 term of G1 while the remaining three binary digits of each Y code are loaded serially by the $T_R'T_M$ term of G1 in conjunction with the above mentioned stepping process.

A pulse is recorded in the T channel at $P_{38}I_{59}$ by the E_cT_R' term of G1. This pulse is used to set up the carriage return code after all integrators have been typed out. This E_c pulse in T will prohibit the correct typeout of integrator 59.

The extra loading cycle, state 2 or 6, is provided to account for the fact that loading is done from the read flip flops of the Arithmetic Unit instead of the record flip flops. During this extra cycle the T channel is loaded with the arithmetic results that were obtained during the iteration cycle in which the typeout command originated. Although continual loading occurs during state 0 and also during state 4, the information loaded during the one extra cycle is the information that is typed out.

7.1.7 Typeout Sequence

When T_T is turned on for the extra loading cycle it also excites a relay driver activating the γ_1 relay. γ_1 when closed energizes the release solenoid of the Read-Out Commutator (Fig. IX) allowing the rotor to turn. The release solenoid is also in series with a push button and toggle switch combination that permits one cycling of the commutator if desired for test purposes. The commutator, when stopped, rests on the δ segment, and due to the inherent delay in the operation of the relay and the release solenoid, the rotor will not commence to move (or if it does start will not move appreciably on the δ segment) until after the extra loading cycle is over and T_R (G13) has been turned on to start the actual read out process.

T_R is turned on by the following E_c pulse after T_T has been turned on. It should be noted that at this point T_g will be off due to T_R' in G14. T_g off causes T_δ to be low (G11). T_W will be on due to the $E_c T_W'$ term of G4. T_K, T_L and T_M will be off since they are reset following the stepping of each Y digit code as explained in the section on T Channel Loading (7.1.6). T_N will be true having been set by the $T_T T_R' E_c$ term of G5. It is seen that at this time the T_K, T_L, T_M and T_N flip flops hold the no-action code, 0 0 0 1. Also, as soon as T_R went true, and loading stopped, the information that had been loaded into the T channel during the last cycle will begin to recirculate due to the $T_R T_W T$ term of G1.

When the commutator's rotor reaches the λ segments, ground will be applied to the coil of the λ relay, activating it (Fig. IX). The contacts of this relay develop two signals (λ) and (λ'). With the commutator on the λ segment, (λ), therefore, becomes true turning T_g on and T_c (G6) off.

When the commutator reaches the τ segment, the τ relay will be activated, signaling the typewriter to type the character corresponding to the code in T_K, T_L, T_M and T_N. The τ relay gates a ground signal, $(G_I' + G_o)$, through its contacts, that is to be applied to the relay decoding matrix as an activate signal for the typewriter (See Fig. IX-b). $(G_I' + G_o)$ is a Console signal formed by inter-connecting the Read-In and the Read-Out switches in such a way that the signal is true (line grounded) if the Read-In switch is in any position except G (Gross), or if the Read-Out switch is in the G (Gross) position; if the Read-In switch in on G and the Read-Out switch is not on G then this signal is false (line floating). As the computer is used, $(G_I' + G_o)$ is true for all operations except Gross Read-In. This τ signal is applied, through the decoding matrix formed with the contacts of the five ρ relays, to the solenoids controlling the typewriter. Furthermore, each typewriter solenoid signal is passed through an isolation rectifier and a rectifier coding matrix, to the four

solenoids controlling the selector armatures of the tape punch (Fig. IX-b). The ρ relays are activated by drivers controlled by the T_K, T_L, T_M, and T_N flip flops during readouts, with T_N activating two relays ρ_{1a} and ρ_{1b} in order to provide sufficient contacts for the decoding matrix. During problem filling the μ relay is activated by the γ_2 relay acting through the Read-In switch; this switches the control of the ρ relays to the fingers of the tape reader (in this situation the typewriter activate signal is supplied by E_σ instead of τ). The μ relay, shown next to the ρ relays in Fig. IX-b, is used only during Gross Read Out; its function will be explained later. As shown above, the first code the typewriter and punch will receive at the start of a read out will be the no-action code 0 0 0 1; this code is now punched if a reversible function tape is being prepared, but the typewriter does not operate with this signal.

After passing over two more unused segments, the timer's rotor next comes to the δ segment and excites the δ relay, causing the (δ) signal to become true. Since T_g went on with (λ), T_δ of G11 now becomes high, clearing T_K, T_L, T_M and T_N and setting T_W off at the end of the first E_c pulse (G4). The (G') signal of Gate 4 signifies that a Gross operation is not going on. T_g is then set off again at the next clock pulse (at a P_1 time) by the T_W' term of G14; consequently, T_δ of G11 again goes false at the start of the iteration cycle following the one during which it became true. With T_W off the recirculation term of G1 is changed from $T_R T_W T$ to $T_W' T_N$, however, there is no information in the T channel to be effected by this change until P_1 time of the first integrator to be typed out. As this first pulse in the T channel is read, the $P_1 T$ term of G4 again turns T_W on and the normal recirculation term of G1 again is effective. Since T_W does not actually come on until the end of P_1, the $T_R T_W T$ term of G1 does not become effective until P_2 time; and since by P_1 time Gate 5, and, hence, T_N still has not received any information from the T channel, the P_1 pulse in the T channel for the first typeout integrator will be lost from the channel and no longer recirculated. This first pulse, P_1, in the T line will not be read by

G5 as the pickup term in this gate, $T_W'T_c'T$, is qualified by $P_{3/38}$. Hence, at this time T_K, T_L, T_M and T_N are all false; i.e. they are holding a space code for the next typeout.

If T_I was on, this P_1 pulse for the first typeout integrator will also set T_c on, due to the $T_W'T_c'T_IT$ term of G6; then T_I will be reset to off two pulse times later by the T_cO_2 term of G2. T_I is now ready to read the sign of the first integrand. The commutator now on its second cycle will again set T_c off and T_g on at the λ segment.

At the τ segment the code 0 0 0 0 in T_K, T_L, T_M and T_N causes a space to be typed. T_δ will become high again at the δ segment; T_W will be set off again at the end of the first machine cycle after which T_g will again be set off. The first pulse to be found in the T channel at this time will be at P_2 of the first typeout integrator. Like the P_1 pulse it will not be recirculated but will set T_W on again. Also at this time the $T_W'P_2T$ term of G3 will set T_P on from P_3 through P_{37} time. At P_{37} the most significant digit of the Y sign digit code for the first typeout integrator will be in the T_{al} flip flop (see Fill Unit). If the sign of Y is negative, indicated by $P_{37}T_{al}$ being true, and if T_P is on indicating the second commutator cycle since the start of the typeout of this integrator, then T_I and T_M will be set on (G2 and G5). T_K remains off as the T pulse at P_2 is blocked in G_5 by the $P_{3/38}$ qualification. If the sign is positive, T_I and T_M will be left in the false state. The typeout code in T_K, T_L, T_M and T_N becomes, therefore, 0 0 1 0 for typing "-" when Y is negative or 0 0 0 0 for typing "space" when Y is positive. With the sign code set in T_K, T_L, T_M and T_N, the commutator will complete another cycle, the third, to type the sign. T_I will remain on for negative Y's to indicate that Y must be complemented unless (T) is true, in which case negative Y's will be tabulated as complements. (T) is true only for auto preparation of reversible tapes and negative integrands are to appear on these tapes as complements (see Operation Manual 2.9.2).

T_c of G6 does not become true at this time since T_W' went false before T_I became true.

When the δ segment is reached the flip flops of Gate 5, T_K, T_L, T_M and T_N, will again be cleared and T_W will be set off by the first E_c pulse that follows. L_g of G7 now becomes true as its controlling term during typeout is $E_c'T_W'$. With T_W off recirculation in the T channel will start with a four flip flop delay ($T \rightarrow T_K \rightarrow T_L \rightarrow T_M \rightarrow T_N \rightarrow T_o$, see the $T_W'T_c'$ terms and the shifting, Lg, terms of G5) with the $T_W'T_N$ term of G1 rather than the T_RT_WT term becoming the controlling factor for recirculation. Since the P_1 and P_2 pulses in the first typeout integrator have previously been lost, all of the digits of this integrator will be subjected to this four pulse time delay as T_W will not be turned on until P_{38} (G4). If the integrator is being typed out at least one of the four signals T, T_K, T_L and T_M must be present at P_{38}. (T_c is also included in this gate as during complementation of negative integrands the digits held in T_K, T_L and T_M will have been complemented while the digit held at this time in T will not have yet been through the complementing gate; and, hence, as will be seen in the following paragraph, it would be possible for T, T_K, T_L and T_M all to be low. Specifically the case, where all four of these flip flops would be low at P_{38} during the complementation cycle of a negative integrand, would be where there is a 0 1 1 1 code following at least one non-zero lower significant digit). This delay shifts the integrand of this integrator four pulses, or one decimal code group, to the left as viewed on the computers oscilloscope. A shift to the left corresponds to a shift to a later time such as a pulse in P_{34} shifting to P_{38}. At the following P_1 the digits that were in pulse positions P_{35}, P_{36}, P_{37}, P_{38} of T are now in T_N, T_M, T_L and T_K respectively, and T_W has been turned off ending the shifting process and again recirculating the remaining T channel information through the T_RT_WT term of G1. The four binary digits that are now in T_K, T_L, T_M and T_N are permanently lost from the T line; and all remaining digits, in the T line, of the integrator being typed out at present have now been

shifted to higher pulse positions. Consequently the digit of the next highest rank to that one just typed is now in the position in the T line to be typed out next, using the same procedure as outlined above.

If T_c is on, indicating a negative integrand, while this shifting is taking place, the complement of Y is entered into the T_K flip flop by the $T_W'T_c$ terms of G5 and stepped into T_L, T_M, T_N and T_O in the same manner as previously described. The sign of the integrand will be typed as the commutator reaches the τ segment during the third cycle. On the δ segment of this cycle T_W and T_g will again be turned off. Now that T_W is off and T_I is on (see above description of the 2nd commutator cycle) T_c of G6 will be turned on by the first pulse in the T channel. Since T_W goes off with an E_c pulse, this hunting for the first pulse in the T channel will always start at the beginning of an iteration cycle. Since the P_1 and P_2 pulses of the first typeout integrator have already been lost, the first pulse in T will be the least significant non-zero binary digit for the least significant decimal digit of the integrand of the integrator that is to be typed out first. This first T pulse will also be read into T_K at this time by the $T_W'T_c'T$ term of G5; however, since T_c now becomes true all following ones (unless the least significant decimal digit is zero, 0011) will be changed to zeros and all zeros to ones by the T_c terms of G5. This seemingly delay in starting the complementing process, until after the first non-zero binary digit in the T channel, effectively adds the extra one to the "nines" complement of the decimal integrand that is required to obtain the "tens" or true complement. If the least significant Y digits are zero (0011) they must be again copied in the T channel unchanged, or in another form that also represents zero, even during complementing; and the inverting of all the remaining binary digits should not commence until after the first non-zero binary digit of the first non-zero decimal code group. For example, if Y is 1100, 1011 1001 1010 0110 (Y= -0.1327), the

complementing gates will give 0011.0100 0110 0101 1010. As another example, if Y is 1100. 1011 1010 0011 0011 (Y= -0.1300), the complementing gates will give 0011. 0100 0110 1101 1101. The code, 1101, corresponds to "10" but the decoding matrix of Fig. IX-b (ρ relays) will decode it as a "0" for typeout provided the μ relay is not activated (μ is only closed in Gross Read Out). T_c will be set off again by the term, $O_1 T_I \overline{T} T_K T_M$, at the end of the first Y digit code if this code represents a zero. $\overline{T} T_K T_M$ represent zero at this time since the least significant binary digit which is a one will be in T_M; the content of T_L (holding the 2nd digit) is unimportant as there will be no 0001 codes in the T line; T_K will be high as the third digit from the T line has been inverted from 0 to 1 by the start of the complementing process; and the most significant digit will be in T as a zero, not having yet reached the complementing gate. T_c will now be off at O_2 times, hence, T_I will remain on. When T_c is left on, following the first non-zero Y digit, T_I is set off by the $O_2 T_c$ term in G2, causing T_c to remain on for the remainder of the integrand. In this way T_c serves to direct the inversion of the code bits as they are entered into the T_K flip flop from the T flip flop.

At the end of the first typeout integrator, T_W will be set on again by the P_{38} $(T+T_K+T_L+T_M+T_c)$ term in G4. At this time, the units digit of the first typeout integrator will be shifted into the T_K, T_L, T_M and T_N flip flops. The code held in these flip flops for the typewriter will be 0011 or 0100 for typing "0" or "1" respectively. If the integrand was complemented, T_c will be set off when the commutator reaches the λ segment (G6). The typeout of the units digit will be made at the τ segment.

The shifting of the integrand to the left and the typing of each digit as it is shifted out of the left end of the integrator will proceed with each cycle of the commutator until the first typeout integrand is completely shifted out of the integrator. The $P_1 T$ and $P_2 T$ pulses in the next typeout integrator cause a space and the next sign to be typed. The second typeout is then complemented if negative and

shifted out in the same manner as the first.

When all the typeout integrands have been processed in this manner and T_W is set off in the δ segment, T_W will stay false for one complete machine cycle and then be set true by $E_c T_W'$. The pulse recorded at E_c time in the T channel during the loading operation will not be recirculated any longer by Gate 1 (the T channel will now be completely cleared) but will cause the $T_W' E_c T$ terms in G5 at this time to set T_L and T_M on and the $T_W' T_c' T P_{3/38}$ term to set T_K on. This output code of 1110 in T_K, T_L, T_M and T_N will be typed as a carriage return (CR) at the next τ segment. The next time T_δ sets T_W off it will again remain false for one machine cycle until it is again turned on by the $E_c T_W'$ term. The $T_W' E_c T'$ term in G5 will set T_N on at this time, making the output code 0001 for punching another no-action code (NA). Also the $T_W' E_c T'$ term in G13 sets T_T off, de-energizing the commutator release solenoid and stopping the commutator approximately one cycle later. If a function tape is being made the no-action code is punched at the τ segment of this last commutator cycle. When the commutator reaches the δ segment for the last time in a typeout operation, the $T_T' T_\delta$ term in G13 sets T_R off and the T_R' term in G14 sets T_g off. The read out unit is now ready for another loading of the T channel, which begins immediately, and the corresponding read out.

The output codes, held for typeout in T_K, T_L, T_M and T_N, are translated by the relay "tree" to operate the typewriter, as shown in Fig. IX-b and indicated in the following table. (See next page).

$T_K T_L T_M T_N$	Typewriter Character
0 0 0 0	space
0 0 0 1	(NA) no action (not typed)
0 0 1 0	—
0 0 1 1	0
0 1 0 0	1
0 1 0 1	2
0 1 1 0	3
0 1 1 1	4
1 0 0 0	5
1 0 0 1	6
1 0 1 0	7
1 0 1 1	8
1 1 0 0	9
1 1 0 1	0
1 1 1 0	(CR) carriage return

The period code, 1 1 1 1, is not obtained in the Read Out Operation.

7.2 Gross Read Out

Gross Read out is a method of reading out the entire contents of the machine onto a paper tape so that all of the information may be re-inserted back into the computer at some later time. This is accomplished in two steps which may be taken in either order: reading out all of the arithmetic information together and reading out all of the address information together. Read out information is filled into the T_K, T_L, T_M and T_N flip flops at the λ segment of the read out commutator. After the correct selective armature solenoids of the tape punch have been energized by these signals through corresponding relay drivers and the ρ relays, an activate signal derived from the τ segment by means of the τ relay energizes the main solenoid in the punch causing it to operate. The δ segment of the timer then clears these output flip flops preparing them to receive the next information that is to be punched on the tape.

To initiate a Gross Read Out, the operator must set the Read Out switch to G, the Read In switch to any other position but G, and the Fill-Idle-Run switch to Fill. This makes (G) (Gross) high, (G') (Not Gross) low, (G_I) (Gross In) low, (G_O) (Not Gross Out) low, and (ϕ) (Fill, manual) high. The DISPLAY switch is set to either the address side, making (S_B) high, or to the arithmetic side, making (S_A) high, depending upon which type of information is to be read out at this time. A marker pulse is then recorded in the T channel at P_1 time of the first integrator to be read out. In a Gross operation, (G) true, this marker pulse automatically goes in at P_1 time of S_T when the MARKER button is pressed (see G16, Fig. VIII); this corresponds to the P_1 position of the scope display at this setting. This pulse in the T channel constitutes the starting point for a gross read out of information; it will step to the next higher pulse time for each commutator revolution, and in this way it serves as an indicator of the magnetic drum position that is to supply the next bit of information that is to be read out.

Next the INTEGRATOR SELECTOR switch is set to the number corresponding to the last integrator to be read out; after the marker pulse has advanced to P_{38} time of this integrator the read out process will automatically stop.

After all of the above preparations have been made, the read out process is finally started by pressing the Read Out Start button (located in the center of the Read Out switch); this makes (P) high which in turn sets J on (G13) as in the typeout scheme previously described in Section 7.1.1. At the first E_c, T_T will be set on; and, at the next E_c, T_R will be set on. When T_R is set on the same E_c pulse sets T_N on so that T_K, T_L, T_M and T_N will be set to 0001, just as in typeout. The commutator, which was released when T_T was set on, moves from the δ segment to the λ segment at which time the gate in G9, $L_h = (G)T_{al}(\lambda)$, becomes true at P_{38} time of the integrator preceding the information to be read out (the marker pulse is in T at P_1 time and, hence, in T_{al} at P_{38} time of the previous integrator).

If (S_B) is high during the read out, the ones, that might be in the Y_2, Y_1, X, and I address channels at the P_{38} time preceding the read out information, are set into the T_K, T_L, T_M and T_N flip flops respectively by the $(S_B)L_h$ terms of G5. If (S_A) is high during the read out, the L_h pulses are gated into L_f (G8) and are then used to gate the ones in the R_4, R_3, R_2 and R_1 flip flops into the T_K, T_L, T_M and T_N flip flops respectively (L_f terms of G5). Since no information is ever held in the arithmetic channels at P_{38} time, no "ones" will be gated into T_K, T_L, T_M and T_N during the first commutator revolution at the start of the read out process. Consequently, if arithmetic information is being read out, the first code that is punched on the tape is the no-action code 0 0 0 1 corresponding to the one set into T_N by the $T_T T_R {}^0E_c$ term of G5 as explained above. If address information is being read out a no-action code will also be the first code punched if there is no information in the address channels at P_{38} of the integrator that immediately precedes the first one being read out; however,

if there is information in Y_2, Y_1 and/or X at this P_{38} time it will be transferred into the T_K, T_L and T_M flip flops and punched out, as well as the one held in T_N, as part of the first code. Whatever this first code turns out to be, this is the place where the tape must be entered into the reader for a Gross Read In. As in the normal typeout scheme (Section 7.1.7), T_g will be set on in the λ segment of the commutator. Then at the τ segment the extraneous code held in T_K, T_L, T_M and T_N will be punched in the tape. However, during the δ segment, T_W will not be set off, as in normal typeout, due to the $\overline{G'}$ term in G4; this disables most of the typeout process previously described. Since T_W remains on, T_g remains on causing T_δ (G11) and, hence, M_g (G10) to be true for the entire δ time. M_g is sent to the Fill Unit where it signals the marker pulse to move to the next highest pulse position (one pulse position to the left as viewed on the computer's oscilloscope). The marker is shifted in a similar manner to that described in Section 6.2 for Manual Fill. M_g from G10 causes M_x (G12 Fig. VIII) to go high, and M_x in turn triggers on G_1 (G13, Fig. VIII). During Gross Read Out, however, G_1 is turned off by the $\overline{G}$ term. G_2 operates in the same manner as always; and the two of them, G_1 and G_2, perform their usual functions in G16 of Fig. VIII, with M_g rather than $\overline{M_L}$ controlling the shift to the left, however. T_δ also clears the T_K, T_L, T_M and T_N flip flops (G5) preparing them to receive the next bit of information to be read out.

On the second passage of the commutator through the λ segment, L_h will be high at the first pulse time of the first integrator to be read out, since the T pulse has shifted one pulse position to the left. Depending on S_B or S_A being high, the address or arithmetic information at this pulse time will be set into T_K, T_L, T_M and T_N (there will be no ones in the arithmetic channels at P_1 time, see Fig. II). This information will then be punched at the τ segment, then the marker shifted to the left and the output flip flops cleared again at the δ segment, then the proper information in the next pulse time (P_2) will be set into the output flip flops at the

λ segment, etc. The continuation of this process is straight forward until the marker appears in the T_{a1} flip flop at P_{37} time with S_A high. The R_4, R_3, R_2 and R_1 flip flops will always be false at this pulse time, consequently, this time may be used to read out Z line information. The most recent dZ output of the integrator being read out can be read in Z_a together with Z_{as} at P_{37} and then punched on the tape at the corresponding position. When the marker is found in the T_{a1} flip flop at the λ segment, L_h (G9) and, hence, L_K (G15) become true. The P_{37} L_K terms of G5 then gate the contents of Z_a and Z_{as} into T_L and T_K respectively, and this latest dZ output is then punched on the tape at the next τ segment. Z_a and Z_{as} are the delay flip flops which hold the most recent dZ output of a given integrator at P_{37} time of the same integrator (see Output Unit discussion in Section 4). When this process reaches the point where the information in P_{38} of the last integrator to be read out is set into T_K, T_L, T_M and T_N, the T_T flip flop is set off by the $P_{38} S_T L_h$ term in G13. The commutator, controlled by T_T, then completes its cycle before it stops and punches this final code. T_R is set off when the δ segment is reached. The T_R' in G14 then sets T_g off, and the gross read out operation is terminated.

7.3 Gross Read In

Gross Read In is a method of re-inserting into the computer a problem that has previously been punched on a paper tape during a Gross Read Out. Consequently, gross fill includes most of the operations of gross read out and uses the same commutator to control the operation.

To initiate a Gross Read In, the operator must set the Read In switch to the G position, the Read Out switch to any position but G, and the Fill-Idle-Run switch to Fill. This makes (G) (Gross) high, (G') (Not Gross) low, (G_I) (Gross In) high, (G_O') (Not Gross Out) high, and ($\emptyset$) (Fill, manual) high. The DISPLAY switch is set to either an

Address position (S_B true) or an Arithmetic position (S_A true) depending on which section is to be filled from the tape at the present time. The tape is set in the reader on the extraneous code that was the first character punched in the Gross Read Out operation. The integrators that are to be used should be cleared, and then a marker pulse is inserted at P_1 of the first integrator that is to be filled (this marker pulse will be in the T flip flop at P_1). As previously explained the marker pulse automatically goes into the T channel at P_1 time of the selected integrator when the MARKER key is pressed with G and ϕ both true. The first integrator that is to be filled need not necessarily be the same integrator that initiated the read out process as a relative shift of all integrators will not necessarily affect the computation process (see Operation Manual Section 2.12). Next, the Integrator Selector switches are set to the last integrator to be filled, and the read in process is started by pressing the Read Out Button, making P high.

Activating the Read Out switch turns J, T_T and T_R of G13 on and releases the rotor of the read out commutator as discussed in the section on Gross Read Out. T_g is turned on at the λ segment. A ground signal, τ_R, is sent from the τ segment through the Read In switch (only if this switch is set on G) to the stepping solenoid of the tape reader. The tape reader will advance, therefore, at the τ segment during gross read in, but the tape punch is turned off so that it will not be operated by τ as it was in the case of gross read out.

On the first passage of the commutator through the τ segment the reader will be signaled to advance from the extraneous code to the P_1 information that is to be filled. The tape information will be read by the α relays, operated by the tape reader fingers, in the same manner as for automatic fill. At the δ segment the P_1 information from the reader will be filled into the address or arithmetic channels, depending upon whether S_B or S_A is true. Also at δ the marker pulse in the T channel will be shifted to the left as M_g of G10 will be true (see Gross Read Out Section 7.2 for discussion of how

the marker shifts during a gross operation).

If (S_B) is high, the gate $F_g = (S_B)(G_I)T$ of $T\,G_1$ in G9 sets the information from the reader into the address channels; this is accomplished by the F_g terms of G20 in Fig. VIII, Fill Unit. This filling into the address channels of the information held by the α relays is accomplished during the δ segment before the marker is shifted, as the marker shift depends upon both G_1 and G_2 being true while F_g depends only on G_1.

If (S_A) is high, the $(G_I)(S_A)$ term makes F_i high (G7 Control Unit); i.e. F_i is always high during Gross Read In if the DISPLAY switch is set to an arithmetic position. The information from the tape will be read into the four arithmetic channels in parallel through the F signals of G15 of the Fill Unit; V_1 will always be off during gross fill making A high and the effective term of this gate. As in all fill operations the F information is filled into the arithmetic channels through G6 of Fig. III. Also as in all fill operations, the information held by the F's will be timed into the arithmetic channels by F_o which is now being controlled by the $T_{a1}G_1G_2(\phi)F_i$ term of G10 of the Fill Unit. Since T_{a1} receives its information one pulse time earlier than does T, F_o and T will contain the same information at the same time during this fill process; hence, information will be placed into the arithmetic channels during each commutator revolution at the pulse time corresponding to the present position of the marker pulse. The marker pulse will be shifted in the T line during the following pulse time after the recording of the present tape code in the arithmetic channels, as the marker is delayed by the S_A flip flop during a shift to the left (see Fig. VIII).

When the marker is in the T flip flop at P_{37} time, and (S_A) is high, the reader will be on the dZ output information of the integrator presently being filled. The F_s and F_e signals developed in G10 are used in G3 of the Output Unit, Figure V, to insert the dZ output from the tape into the output recirculation logic. T_{a2} is used in the F_s and F_e gates as the marker pulse will occur there two pulse times prior to being in

the T flip flop, or at P_{35} time (see Fill Unit). Since P_{35} is the time new outputs are entered into G3 of the Output Unit to be recorded on the Z lines during computation, this is the correct time to fill in the outputs that were obtained during a gross read out; an output from the tape, therefore, will be interpreted by the Z line logic as a new output from the integrator that corresponds to the present position of the marker pulse in the T channel. Once recorded on the Z lines the information will be retained in the lines in the same manner as during idle (see Section 4.4). The dZ information is filled before the shifting of the marker as both F_s and F_e are qualified by G_1 and not also by G_2, while, as noted previously, the marker shifting depends upon both G_1 and G_2 being true. This dZ information will also be recorded as extraneous information in the arithmetic channels, R_3 and R_4, at P_{37} time, in the same manner as all of the previous arithmetic information was filled. This extraneous information will remain unchanged in the R register as computation is restricted to $P_{2/36}$ time during every integrator by G9 of the Arithmetic Unit. Hence, this extra information in the R register will not interfere with computation; however, it must be removed manually, in the event a second gross read out is required, in order to again punch out the correct dZ output information.

The marker will proceed to advance from pulse position to pulse position and from integrator to integrator, determining the correct position to enter the new information from the reader into the computer's memory system in the manner outlined above, until this marker pulse reaches the position corresponding to the P_{38} position of the integrator selected on the Integrator Selector switches. At this point the $P_{38}S_TL_h$ term of G13 will turn T_T off terminating the gross read in process in the same manner as gross read out was terminated (Section 7.2). With the appropriate Console switch settings, the computer will be ready to compute, carrying on the computation from the point in the problem where the gross read out was effected.

PLOTTER UNIT

8.1 Plotter Unit Operation

The Plotter Unit, Figure X, serves as a means of selecting specific integrator outputs, and transmitting these outputs at an appropriate impedance level to external equipment such as graph plotters, graph followers or a D-12 Coupler. Gate 1 develops four signals for each plotter, one each to signal the plotter to move in the +X, -X, +Y and -Y directions. P_{L1} and P_{L2} are signals true during $P_{2/36}$ of I_{59} and I_{02} respectively (see G16 of Control Unit). During I_i, Z_a and Z_{as} hold the latest output of I_{i-1} while Z_b and Z_{bs} hold the latest output of I_{i-2}. Consequently, the ΔX_1 expressions of G1 select the outputs of I_{58} to transmit to a graph plotter while the ΔY_1's select the outputs of I_{57}, the ΔX_2's select the outputs of I_{01}, and the ΔY_2's select the outputs of I_{00}.

The gates of G2 are similar to those of G1 but are developed for use with a D-12 Coupler and their operation is included in the discussion on the Coupler. The gates of G2 may be simply altered to conform with those of G1 if the user requires more than two graph plotters; four additional plotters may be operated in this manner.