# DA-1 DIFFERENTIAL ANALYZER

# FOR THE CONTROL DATA G-15 COMPUTER

## SERVICE MANUAL

CONTROL DATA
CORPORATION

# DA-1 DIFFERENTIAL ANALYZER

# FOR THE CONTROL DATA G-15 COMPUTER

## SERVICE MANUAL

Pub. No. 60064800
May, 1965

# TABLE OF CONTENTS

## A P P E N D I X

# LIST OF ILLUSTRATIONS

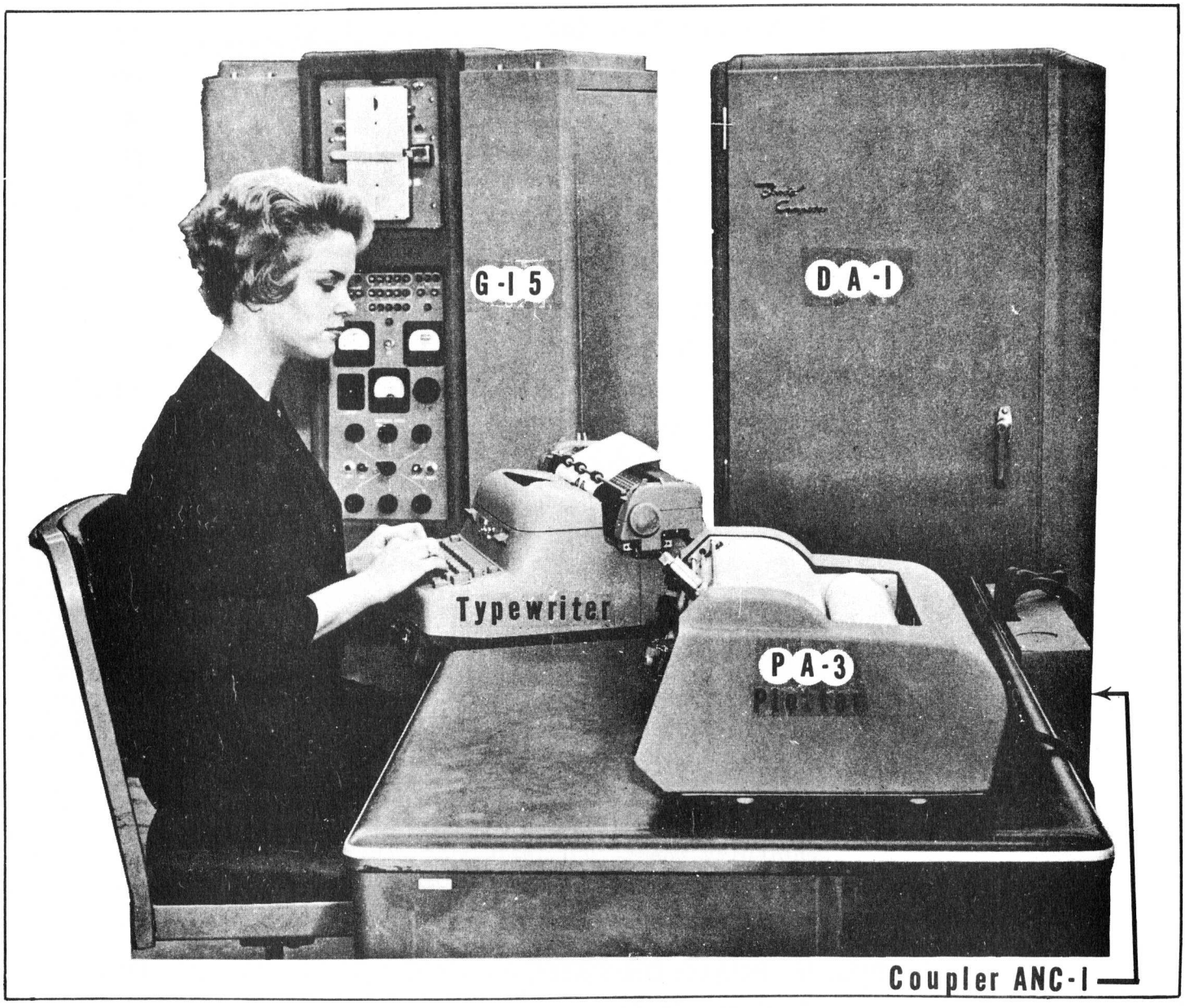



Figure 1. DA-1 with G-15 and PA-3

# SECTION I

# INTRODUCTION

## 1.0 INTRODUCTION TO THE DA-1

The Digital Differential Analyzer, Model DA-1, is one of the many electronic devices manufactured by the Bendix Computer Division of The Bendix Corporation (See Figure 1).

This machine provides a simple means for programming the numerical solutions of problems which can be expressed in the form of differential equations, including linear and non-linear simultaneous equations, solutions for roots of transcendental equations, and the simulation of many real systems.

## 1.1 OBJECT AND SCOPE

In order to solve differential equations on a general purpose computer without a differential analyzer attachment, it is necessary to transform the equation into suitable arithmetic form by use of techniques of numerical analysis and then to formulate coded instructions for the computer. This process is both time consuming and costly.

By making use of the DA-1 Accessory, the differential equations are fed into the computer, and their solutions, in digital form, are realized. These digital solutions can be plotted on a graph by using the Bendix PA-3 Plotter. The DA-1 may also be used in combination with a graph follower. The follower is used for feeding data into the DA-1 from the already plotted graph with the use of a photo sensitive attachment. A greater accuracy may be obtained from the DA-1 than is possible from earlier differential analyzers which are of an analog type.

## 1.2 HISTORICAL DEVELOPMENT

The idea of a differential analyzer is not a new one. Lord Kelvin conceived it in 1876. His brother actually constructed a mechanical disc, ball and cylinder integrator.

About thirty years ago, Dr. V. Bush designed and constructed a working mechanical differential analyzer at MIT. Soon after, the Soviets had one built and operating in Leningrad. These mechanical analog computers were followed by electronic analog designs in which integration is carried on by voltage changes using time as the independent variable.

A more recent innovation in the computing field than even the electronic analog computer is the Digital Differential Analyzer. The forerunner of the present-day digital differential analyzers was built at Northrop Aircraft Corporation under an Air Force contract. Although the prototype computer was designed for possible airborne applications, it did not take long for the designers to realize the commercial possibilities of the computer. A production model computer was built and placed on the market in 1950. A series of unfortunate circumstances, and the awkwardness experienced in its use, delayed its widespread acceptance.

Later, more mature versions of the digital differential analyzer were in commercial production, and one was the D-12 produced by The Bendix Corporation. However, the D-12 is no longer in production.

The DA-1 was developed by Bendix in an effort to eliminate undesirable features inherent in the use of general purpose or analog computers. The DA-1 is used in conjunction with the G-15 general purpose computer for the programming of some highly involved problems. It is easier to program, and has greater accuracy and versatility than analog computers.

## 1.3 THE BASIC PRINCIPLES

Basically, a machine which is capable of solving ordinary differential equations need only consist of a number of integrating mechanisms, connected together so that the assembly can be constrained to solve any type of differential equation. The interconnections are developed from the various terms and their relationships.

The DA-1 is functionally a hybrid. It handles information and performs arithmetic operations digitally as would a general purpose digital computer, but its programming is based on the same concept of integrators which forms the foundation of the mechanical and electronic analog differential analyzers. The means of execution embodied in the DA-1, the use of digital techniques and the association with a general purpose computer, permits an accuracy, versatility and ease of use which greatly extend the range of usefulness of this machine.

The basic computing unit in the differential analyzer is the "integrator". A detailed representation of an integrator is given in Section II. For the present, an integrator may be thought of as a physical unit (a black box) which has two input lines and one output line as shown in Figure 2.

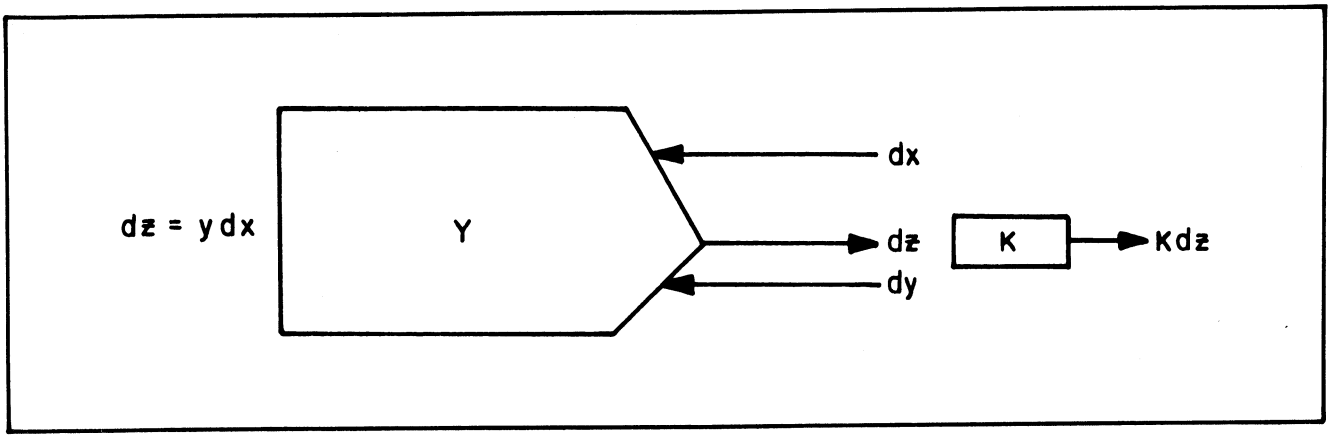



Figure 2. DA-1 Integrator

These units (black boxes) may be interconnected so that the output lines of some integrators become the input lines of the others. The inputs are differentials of dependent and independent variables arising from the differential equations being solved. They are called "dy" and "dx", respectively.

An idea of how integrators can be used to program the solution of a differential equation, is shown by a simple example in Figure 3.

The details will be discussed in the latter part of the manual.

For example, equation $\frac{d^2y}{dx^2} + \frac{dy}{dx} - y^2 = 0$ is to be solved for y, which is the dependent variable.

The equation could be easily transformed into a more desirable form: $\frac{d^2y}{dx^2} = \frac{-dy}{dx} + y^2$.

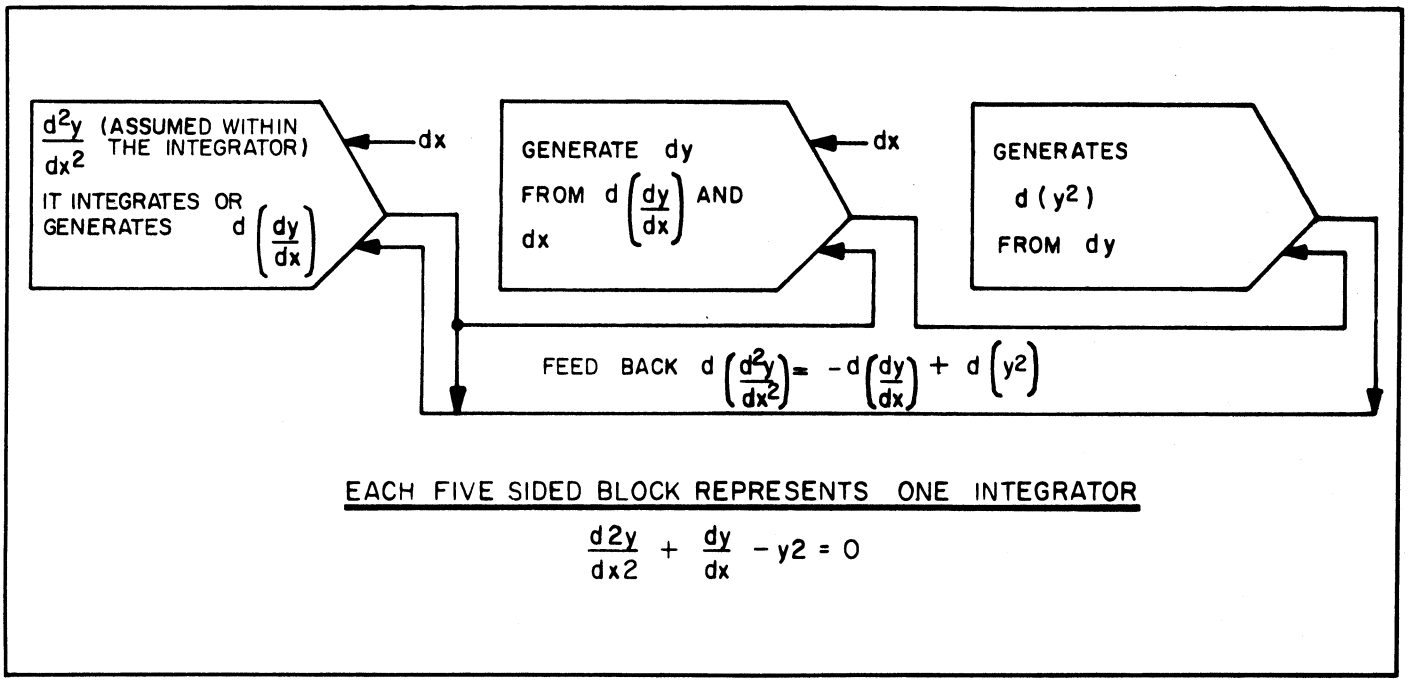



Figure 3. Simplified Diagram Integrator Connections

This problem is, of course, a trivial one. Problems having many more terms, with constant multipliers and with empirical functions can be handled if desired. Mathematical functions such as algebraic, trigonometric, exponential, logarithmic, hyperbolic, Bessel, and probability may all be generated internally by integrator interconnections. However, it should be pointed out at this time that the integrators are not physically separable because the interconnections between integrators are not made physically; i.e. by means of wires or plug boards. These interconnections are made by electronic coding means which involves time delay (or memory) circuits.

After the information concerning each integrator is processed, it is stored on the memory drum of the G-15 while the same circuits process the other integrators. The information is taken from the memory drum only when the circuits are ready to process that particular integrator again. The unmultiplied output of one integrator could be gated as the input to the other integrator without being stored on the memory drum.

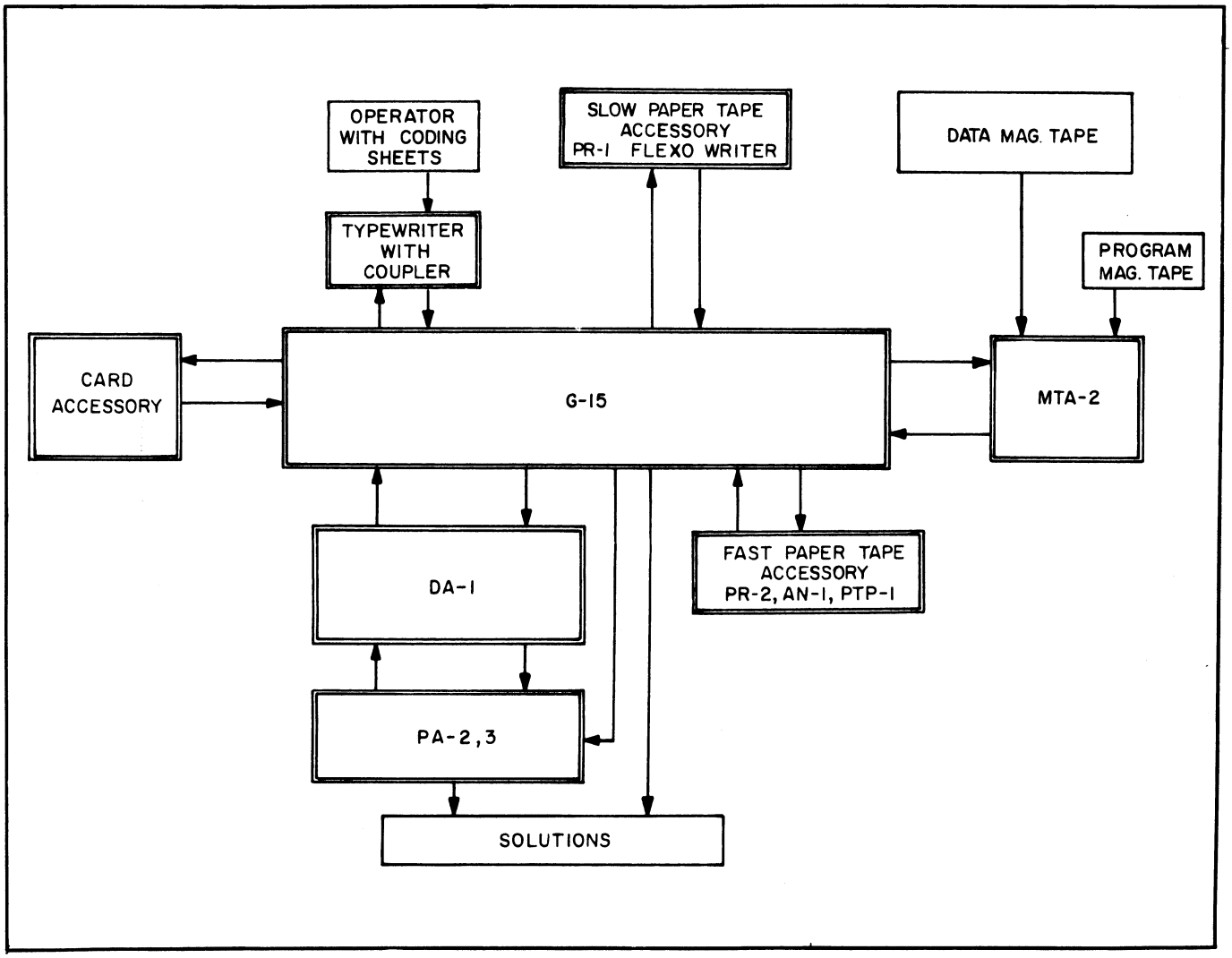



Figure 4. G-15, DA-1 and Accessories

1.4 G-15 and DA-1 (See Figure 4.)

The computing facility of the G-15 may be used in conjunction with

## THE CORNU SPIRAL

Plot of y vs. x where

$$y = \int_0^t \sin \frac{\pi u^2}{2}\, du$$

$$x = \int_0^t \cos \frac{\pi u^2}{2}\, du$$

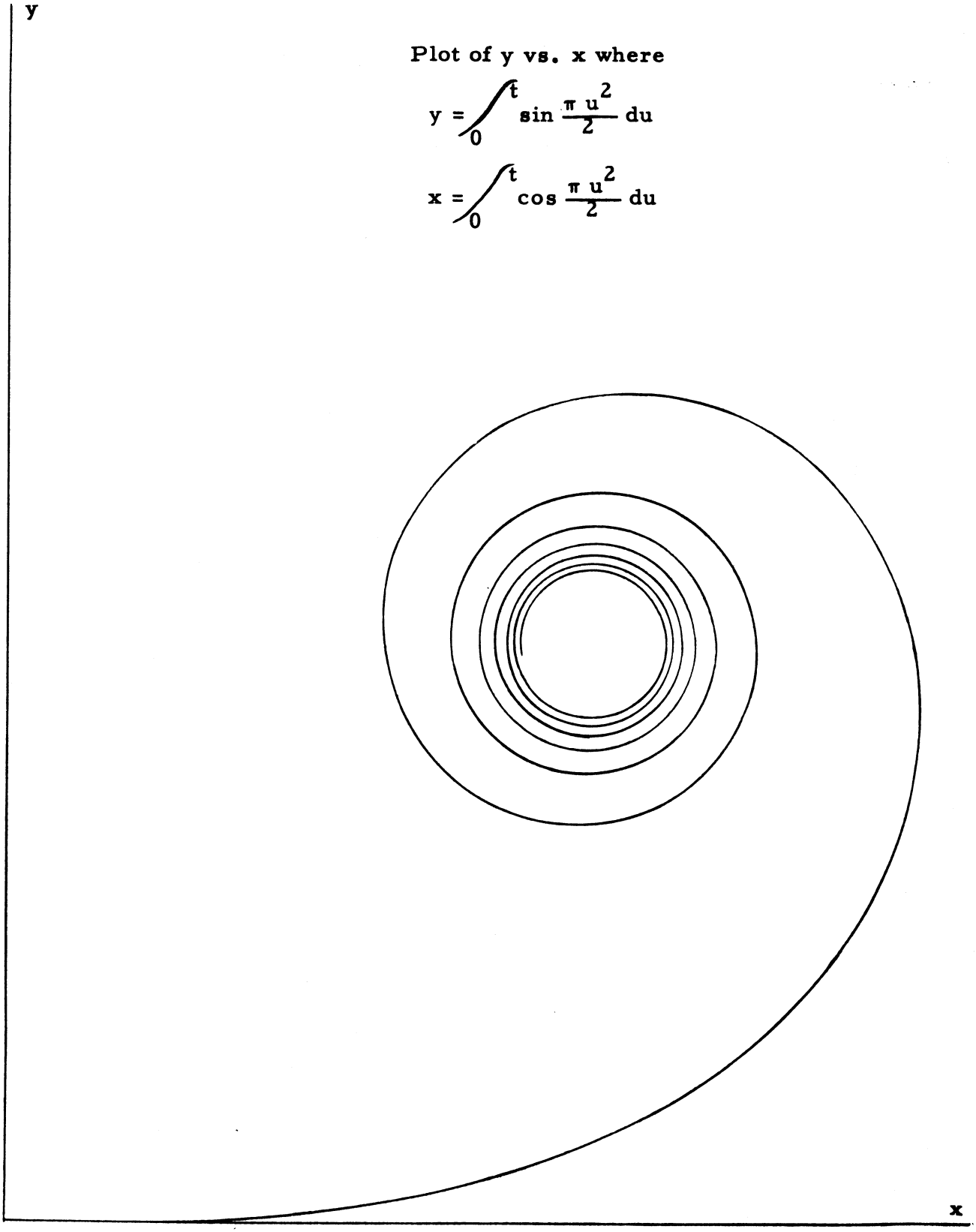



Figure 5. The Cornu Spiral

the DA-1 for the programming of highly involved problems (see Figure 4). The DA-1 is controlled by the G-15 General Purpose Computer. The DA-1 has no memory of its own, and therefore, may not be used independently. The input can be programmed from a standard G-15 and be used with such other G-15 accessories as:

(1) PR-2, AN-1 (punched paper tape)
(2) CA-1 (punched cards)
(3) MTA-2 (magnetic tape)
(4) PFA-1 (graph follower).

Empirical functions input may also be programmed from any of the G-15 input devices or from a graph follower. The output may be recorded by G-15 output equipment.

### 1.4.1 PHYSICAL RELATION

Figure 4 shows the input and output accessories to the G-15 and the DA-1.

The graph-plotter plots the relationship between two variables generated by the computer in 1/100 inch increments on paper, having a maximum size of 12 by 18 inches. It is an optional accessory to the DA-1. Figure 5 shows a graph plotted by the graph-plotter (PA-3).

A photo of the interconnecting cable connections are shown in Figure 6.

### 1.4.2 COMMUNICATION BETWEEN G-15 and DA-1

The DA-1 has a capacity of 108 integrators and 108 multiplier constants. This, however, does not imply that there exists duplications of hardware necessary to make up an integrator and a multiplier constant. Since the process of integration and multiplication is merely the manipulation of the contents of a particular word time on the G-15 memory drum, and the DA-1 processes all long lines during one cycle of operation, then there are 108 integrators

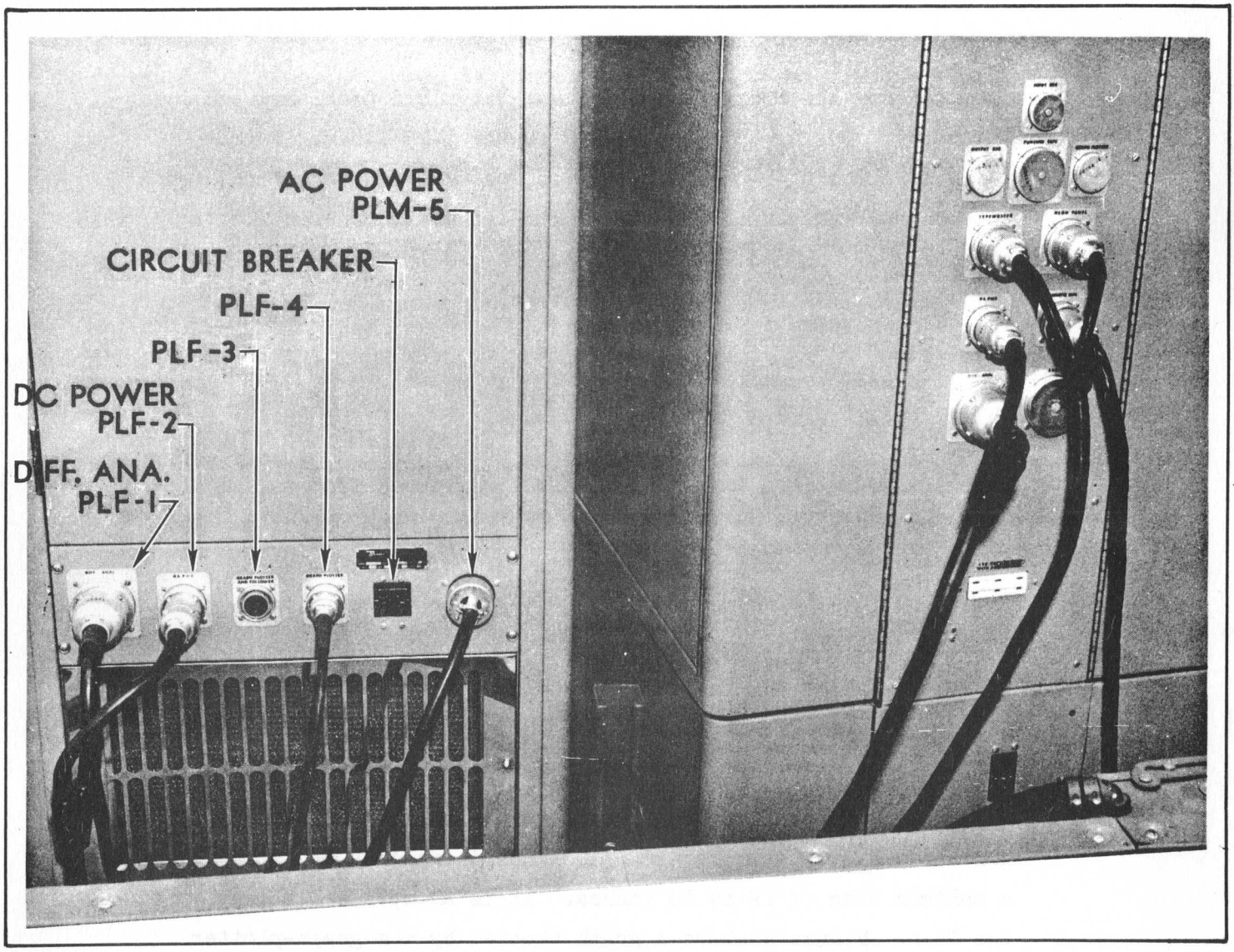



Figure 6. DA-1, G-15 Interconnecting Cables

and 108 multiplier constants available during one drum cycle or iteration. Since a word time represents the processing time of one integrator, then the maximum speed of the DA-1 is 34 iterations per second. An iteration is defined as one G-15 drum cycle, during which time all of the 108 integrators may be processed. The processing of information within the DA-1 is done in the same manner as in the G-15, i.e., all the information is in binary form and is processed serially. The memory lines used by the combined operation of G-15 and DA-1 are listed in paragraph 1.4.3. The purpose of each of these lines and the form in which information appears in them is also given.

## 1.4.3 MEMORY DRUM UTILIZATION

There are twenty - 108 word lines or "long lines" on the magnetic drum in the G-15 computer. When the DA-1 is connected and running, fourteen of these lines plus two "short lines" are used as follows:

| | Line |
|---|---|
| Control Information | 18 |
| Y Registers | 17 |
| R Registers | 16 |
| K Registers | 15 |
| r Registers | 14 |
| Output (Z Lines) | 21, 22 |
| dy ADDRESSES (If DAPPER-1A or DAPPER -2 is used). | |
| 27 Integrator Operation | 13 |
| 54 Integrator Operation | 12, 13 |
| 81 Integrator Operation | 11, 12, 13 |
| 108 Integrator Operation | 10, 11, 12, 13 |
| dx ADDRESSES (If DAPPER-1A or DAPPER -2 is used). | |
| 27 Integrator Operation | 9 |
| 54 Integrator Operation | 8, 9 |
| 81 Integrator Operation | 7, 8, 9 |
| 108 Integrator Operation | 6, 7, 8, 9 |
| Yo REGISTERS (If DAPPER-1A or DAPPER -2 is used). | |
| 27, 54, 81 Integrator Operation | 10 |
| 108 Integrator Operation | 5 |

Routine Dapper-1A or Dapper-2 requires for its program the remaining long lines on the drum. Long memory lines available for general use by the programmer when the DA-1 is in operation are summarized below:

| | |
|---|---|
| 27 Integrator Operation | 5, 6, 7, 8, 11, 12 |
| 54 Integrator Operation | 5, 6, 7, 11 |
| 81 Integrator Operation | 5, 6 |

## 1.5 WORD STRUCTURE G-15 and DA-1

The DA-1 Accessory utilizes the memory drum of the G-15 for information storage during computation as shown in Figure 7.

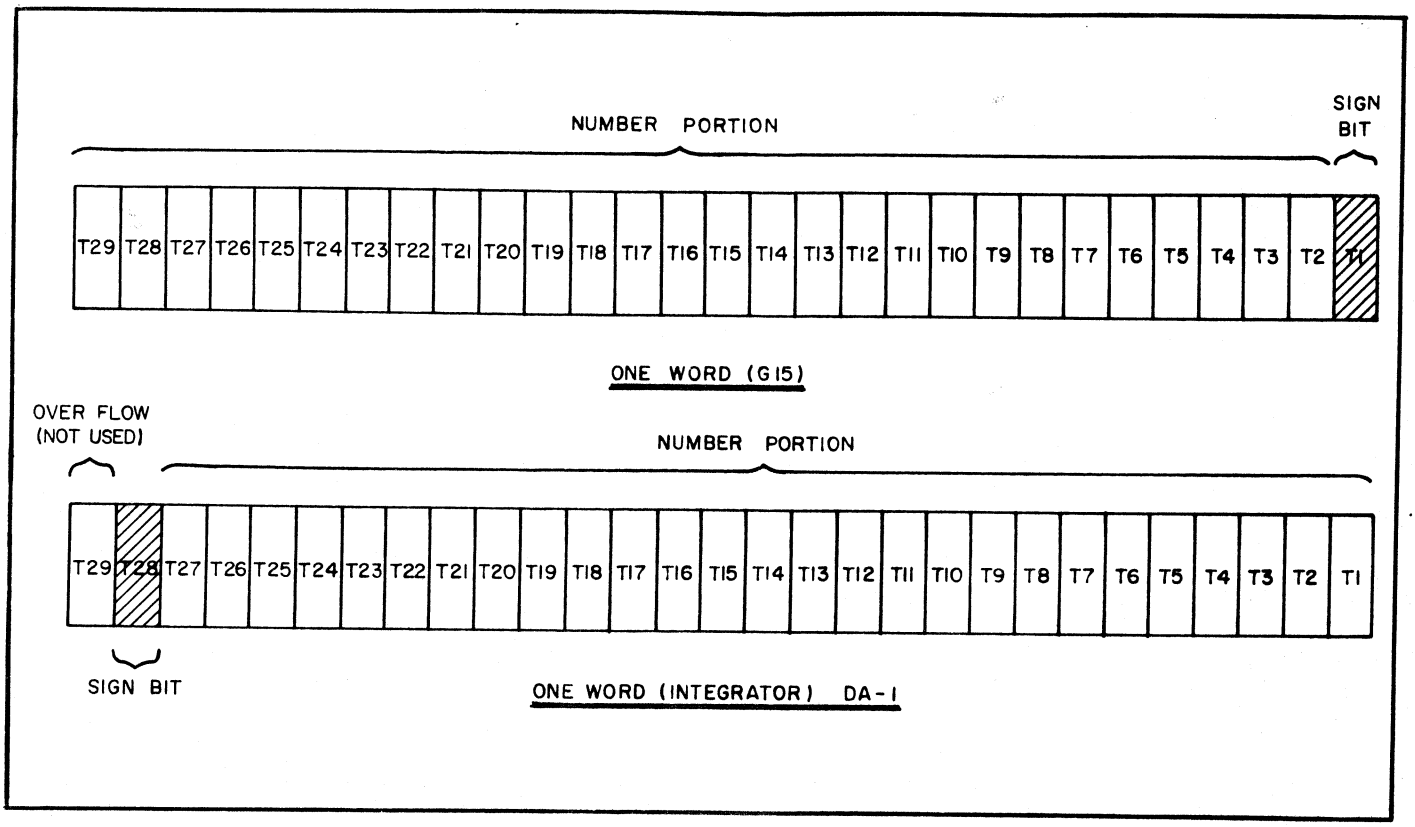



Figure 7. Word Structure G-15 and DA-1

Each of the 108 word times on the drum periphery corresponds to one integrator time. On the whole, the structure of an integrator in the DA-1 is the same as that of one word in G-15, but there are some preassigned differences for different bit positions. Figure 7 clearly shows the difference between the two. $T_{28}$ is considered as the sign bit instead of $T_1$. $T_{29}$ bit position is not used. In the DA-1 the binary point is to the left of $T_{27}$ while in the G-15 it is to the left of $T_{29}$. It is necessary to know the distribution and nature of the information stored on the drum in order to understand coding and the logic which will be described in the following sections.

When the DA-1 is OFF, the G-15 has unlimited freedom in reading and writing in all portions of the memory.

When the DA-1 is ON, the G-15 can record any portion of the DA-1 information without restriction, except that the contents of M21 and M22 will be precessing.

The following type of information appears in the G-15 memory when DA-1 is ON.

Line 18 (Control Information): Can be written in a normal manner. New data must obey the usual instructions regarding the start pulse and T26 (the clear pulse).

Line 17 (Y or Integrand Register): The bit by bit logical sum of the information to be used by DA-1 (Y) and the new integrand value YN = (Y + dY) will be written into M17.

Line 16 (R Register): The bit by bit logical sum of the information to be used by DA-1 (R) and the new remainder RN is recorded into M16.

Line 15 (K Register): Can be written into in a normal manner and is used as a multiplier constant.

Line 14 (r Register): The bit by bit logical sum of the information to be used by the DA-1 (r) and the new remainder (rn) is recorded in Line 14.

Line 06 - 13: There is no restriction on writing in these lines when the DA-1 is ON.

Lines M21 and M22 (4 word SHORT LINE): Writing is avoided in these registers because they are precessing during DA-1 computation. If the contents of these two lines are altered during an interruption in DA-1 computation, it will make corresponding changes in the first cycle of DA-1 computation when the DA-1 is turned ON again.

Line 05: The initial conditions of Integrand Y are stored in this line when all the 108 integrators are being used by the plotter. But, in Memo No. 723, a change can be made to use 81 integrators when the plotter is used with the DA-1. This will free line 05 and 06 for general purpose programming.

# SECTION II

# INTEGRATORS

## 2.0 THE INTEGRATOR

Each of the operational units in a digital differential analyzer is called an integrator because of the type of operation which it may perform. This operation is the performance of quadratures, in accordance with a fixed program which results in an approximation to integration. Before we discuss the actual process of integration, it would be advisable to discuss the original ideas and developments of the integrator.

## 2.1 ORIGINAL MECHANICAL TYPE OF INTEGRATOR

The original differential analyzers accomplished the operation of integration mechanically. This was accomplished by Dr. V. Bush with the use of a wheel and a disc, as shown in Figure 8.

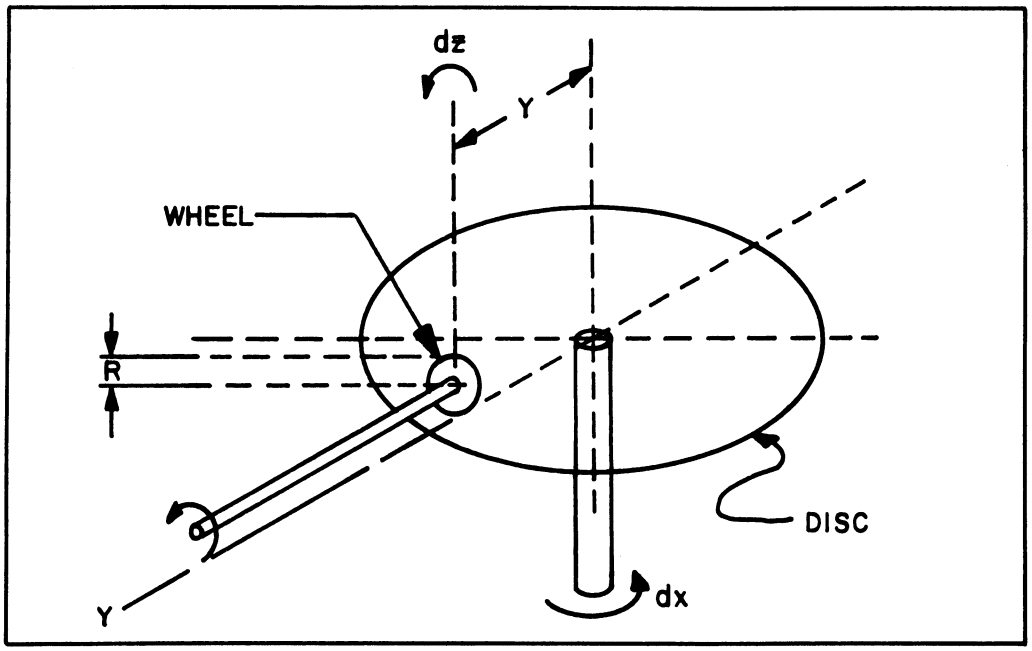



Figure 8. Mechanical Integrator

The primary input, or independent variable dx, can be thought of as the rotation of a constant speed motor. The secondary input, or dependent variable, is introduced to the shaft of the wheel as indicated by Y motion of the shaft.

Any incremental rotation of the disc dx, causes a rotation of the wheel dz, proportional to Y and dx and inversely proportional to the size of the wheel R. The angular incremental rotation of the wheel may, therefore, be defined as:

$$Y\,dx = Rdz$$

or

$$dz = \frac{Y}{R}\,dx$$

If, at the time X is rotated, a differential input dy is applied to the lead screw, the shaft position Z is proportional to the integral of y with respect to x, or

$$Z = \frac{1}{R}\int y\,dx$$

The following example will show how this mechanical integrator is used to find the area under a hypothetical curve, see Figure 9.

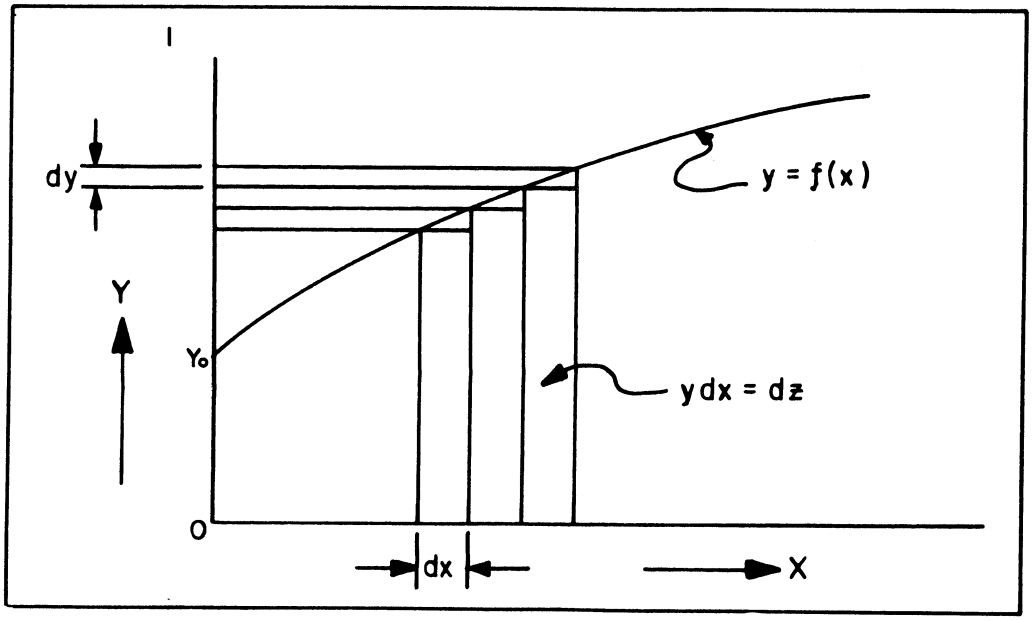



Figure 9. Integration of Hypothetical Function

Consider the Y motion of the shaft as representing some function of X; $Y = f(x)$. Also, consider that an incremental output; dz, represents a unit of area; dx in width and one unit in height.

If the physical dimensions of the integrator are such that one revolution of the wheel occurs each time an increment of area has been traversed, then to establish the total instantaneous value of the area, Z, it is necessary only to accumulate a count of the number of revolutions the wheel has gone through. A mechanical adder could be simulated by a differential gear where the output of the adder, which would be in the form of a rotating shaft, would be the sum of its inputs.

A mechanical constant multiplier could very well be represented by a gear train. It is also conceivable that the integrator could be used for this purpose simply by setting Y to some constant value; hence, the input rotation, x, would be directly proportional to the output, or the Z rotation.

## 2.2 DIGITAL TYPE INTEGRATOR

The digital integrator operates in a manner very similar to the mechanical integrator. Where as, the mechanical integrator indicates an incremental output, dZ, by a specific amount of rotation of the wheel, the digital integrator indicates the same incremental output by a pulse. The accumulation of the dy's takes place in the Y register as shown in Figure 10.

The instantaneous value accumulated in the Y register is added to the R register whenever there is a dx pulse. This dx pulse controls the flow of the contents of the Y register to the R register. This value represents the incremental area Yn dx. The R register will eventually overflow and generate the dz pulse, which, as before, indicates

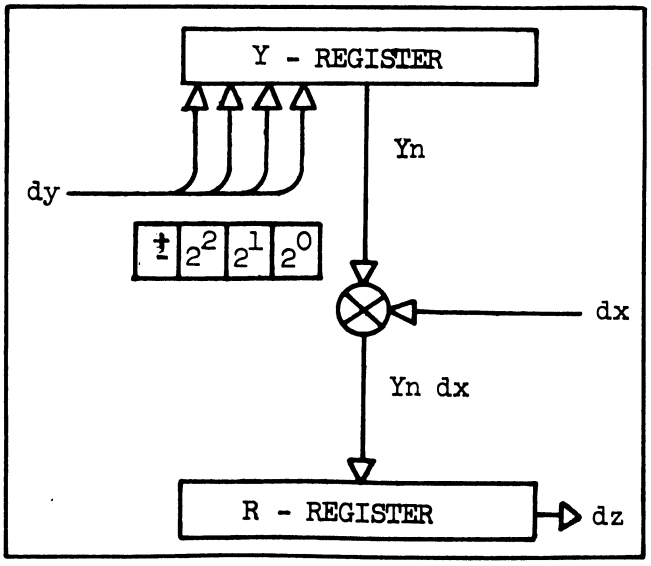


Figure 10. Digital Integrator

that an incremental area of dx width and one unit in height has been traversed. Since only some fraction of the last Yn dx will cause the overflow to occur, the remainder must be retained and be added to the following Yn dx values. For this reason, we refer to this register as the R (Remainder) register.

Multiplication by a constant is also analogous to the mechanical system. Suppose that we wish to multiply the incremental outputs dz, by the constant 1/2. Mechanically, this could have been done by applying dZ in place of dx, and placing the wheel in such a position that it would require two rotations (or dz's) of the disc to turn the wheel one revolution. This would result in an output equal to one-half of the input. As the hardware of the mechanical integrator can be used in multiplication, the DA-1, in the same manner, can provide a method of multiplication by a constant using the components of an integrator.

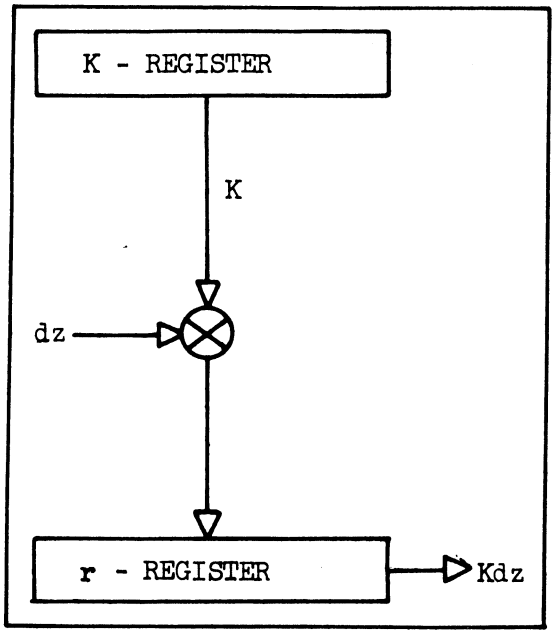


Figure 11. Multiplier Constant - "K"

Figure 11 illustrates a method of doing this. The contents of the K register (constant) are gated with dz and added into the "r" (remainder) register.

Using the previous example of one half, the K register will contain the one half configuration. The dz will gate one half into the "r" register. This will not cause overflow. However, the second dz will gate in another one half. This second gating will result in an overflow of the "r" register, thus resulting in a multiplied output, Kdz, or as in the example, 1/2 dz. This is the output which may be used as inputs (dx, dy, or both) to other integrators.

## 2.3 FUNCTION OF THE INTEGRATOR

A detailed description of the functions of the integrator as being used in the DA-1, is given in the following discussion.

### 2.3.1 DA-1 INTEGRATOR FUNCTION

Before discussing the process of integration in detail, a complete understanding of Figure 13 would be helpful. The memory lines and the selected integrator No. 32 is shown. It is obvious from the previous discussion that numerical integration can be performed with two integrators. An incremental output, dz, (which is equal to either +1, -1, or zero) is obtained from one integrator so that dz = ydx.

The incremental outputs are accumulated in the Y register of a second integrator in order to obtain;

$$z = \int dz = \int_{xo}^{x} ydx$$

see Figure 12 for graphical explanation.

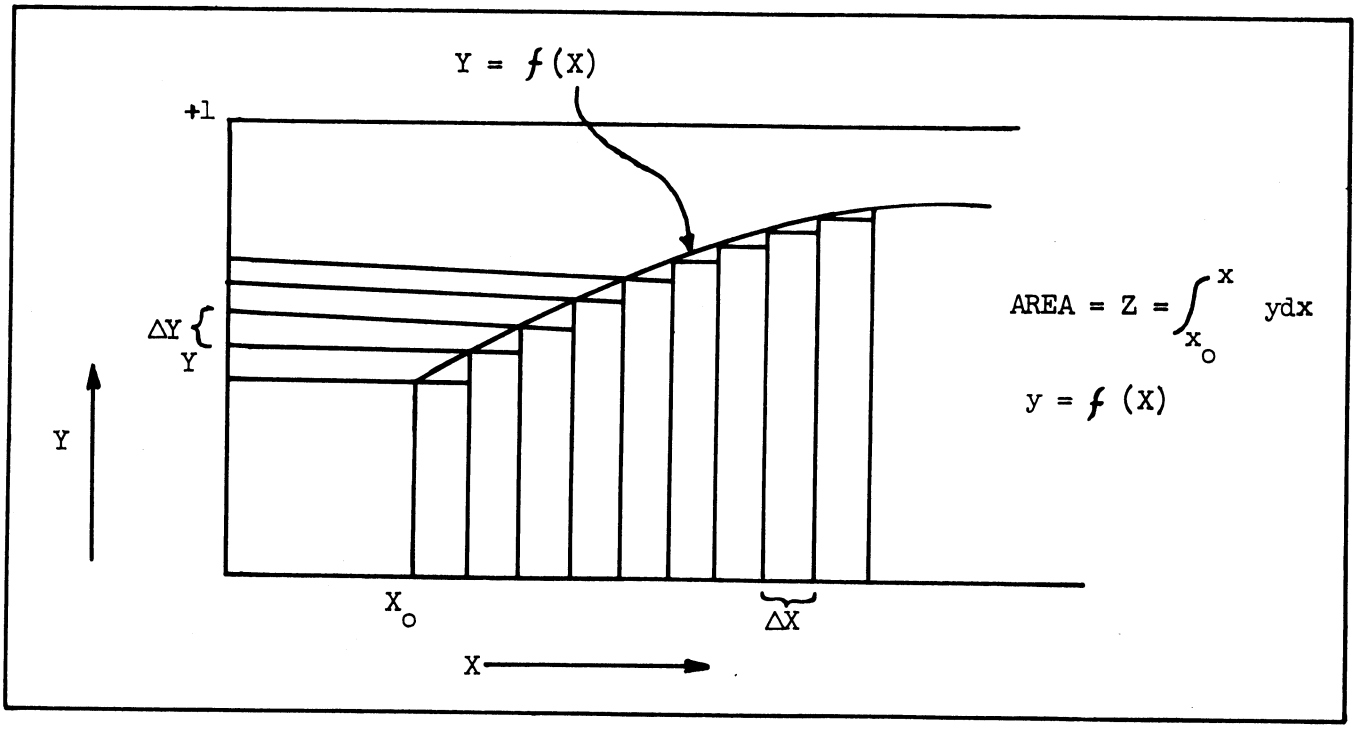



Figure 12. Simple Integration

Consider that we want to evaluate the definite integral of y with respect to X by continuously accumulating the area under the curve $Y = f(X)$ as X increases. The curve starts at the point (Xo, Yo).

If Y is the instantaneous value of $Y = f(X)$, each time X increases by one increment dx, a rectangle of width dx and of height Y will be added to the already accumulated area.

Each time an increment of area has been covered, a notification is sent out and the total area is allowed to accumulate elsewhere. The rectangle, width dx and unit height, is our increment of area represented by a dz pulse as shown in Figure 12.

It is in this manner that the DA-1 integrator operates:

(a) Each integrator may receive two or more input lines of information,

(b) The dx input notifies the integrator each time that X has increased or decreased one increment.

(c) The dy line informs the integrator of the corresponding changes in Y, and

(d) the dz line is the output through which the integrator signals each time an increment of area has been traversed, and is fed into another integrator as its dy input.

The second integrator will then contain its Y value, as the total instantaneous value of the area Z. It is necessary for the integrator to contain two registers, Y and R. Y algebraically accumulates the value of $Y = f(X)$ and R contains the fractional part of increments of area. This value in the R register will range between zero and one increment.

The actual digital operation is carried out in a simple manner. dy increments are added to $Y = f(X)$ algebraically. Each time a dx pulse occurs, Y is added to the number which exists in the R register.

Since neither register is cleared in the process, the R register will overflow periodically. It is the overflow of the R register which causes the output signal.

The multiplication of an integrator output by a constant K, is similarly done by two registers as shown in Figure 11. The K register has no modifying input. The other register, r, has the value in K added to it whenever a dz output occurs. The overflow of the r register provides Kdz output signals.

Electronically, the Y, R, K and r registers are parallel channels on the memory drum of the G-15 computer. Each word-time in the channels represents one integrator. We can store information for 108 integrators on the drum. Consequently, integrators are processed serially, beginning with integrator 00.

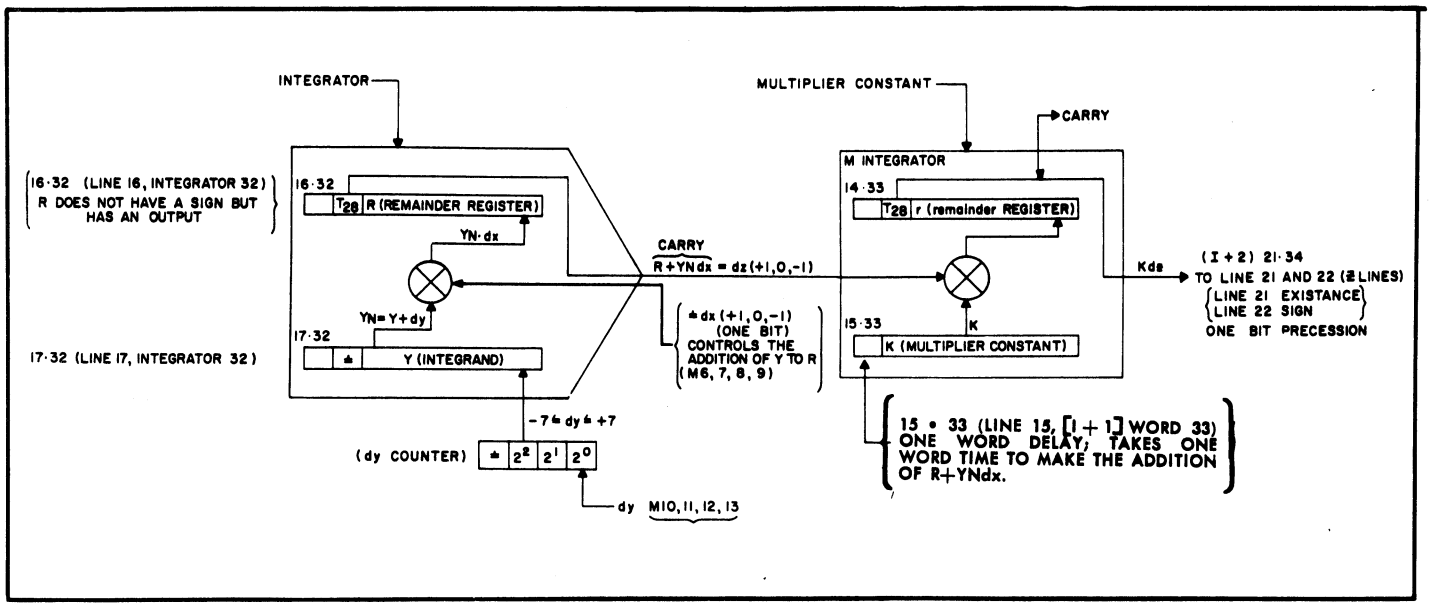



Figure 13. Function of an Integrator, No. 32

## 2.3.2 CONNECTION BETWEEN INTEGRATORS

Most of the engineering problems to be solved with the DA-1 take the form of equations. The first step in obtaining the solution is to arrange the integrators to generate all of the terms appearing in the equations. The generation of terms is accomplished by taking the output of one integrator and feeding it into another integrator as its inputs. The basic integrator equations can be represented schematically. Each pentagonal and rectangular block represents an integrator and its multiplier constant (coefficient), respectively. Refer to Figure 3 for the generation of the following function.

$$\frac{d^2y}{dx^2} + \frac{dy}{dx} - y^2 = 0$$

The following simple example will show how the multiplier constant is used in the process of generating a function.

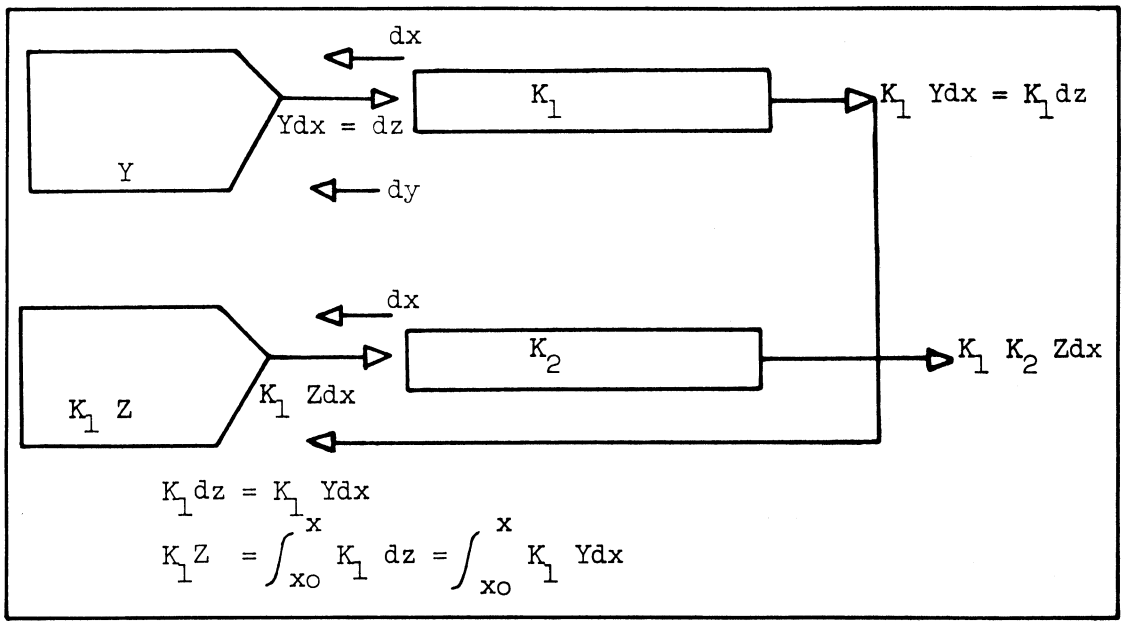



Figure 14. Multiplier Constant Usage

It is assumed that prior to the receipt of any dy or dx inputs, the Y registers of both integrators were empty. In the top integrator in Figure 14, dy is integrated or accumulated to form Y. A dz

output is generated with the independent variable dx. It is multiplied by the multiplier constant associated with the integrator to form $K_1 dz$. The $K_1 dz$ output is sent as the secondary input to the lower integrator where it is again accumulated in its Y register. This output is again multiplied by $K_2$ and the output $K_1 K_2 Z dx$ is used as needed.

Similarly, the two inputs to an integrator can be tied together as shown in Figure 15.

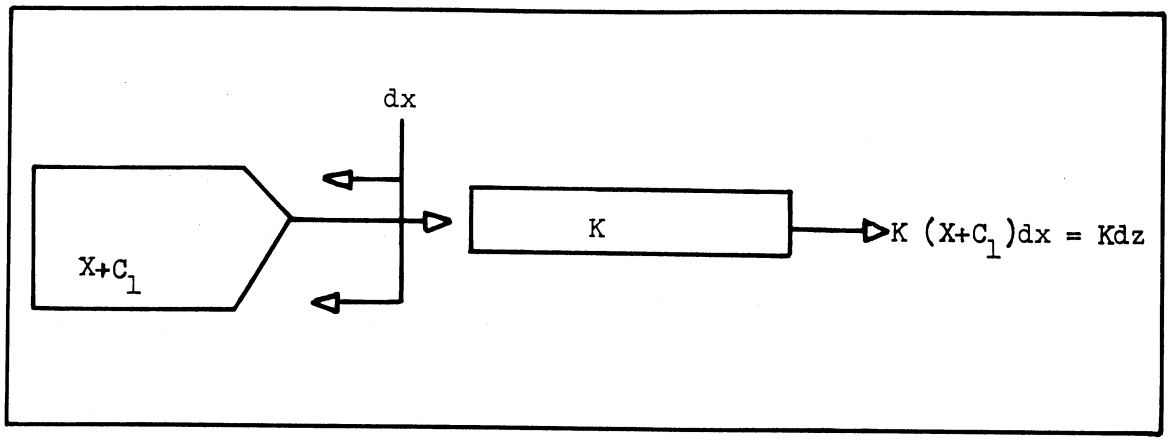



Figure 15. Quadratic Function

$Kdz = K (X+C_1) dx$ which, when accumulated, will produce the quadratic function:

$$C_2 + KZ = K \left( \frac{X^2}{2} + C_1 X \right) + C_2$$

This is the general procedure for generating different functions.

Other examples can be formulated. Reference is made to the DA-1 Programming Manual for additional information.

## 2.3.3 OTHER OPERATIONS OF INTEGRATORS

Integration is not the only type of operation that can be performed by integrators. The other operations have nothing to do with integration as such, but the name "integrator" is retained in the interests of uniformity.

In all of these operations the integrator receives the same two

inputs, dY and dX, and has an output, dZ. The method of operation of the integrators when performing these other operations is different from that explained in paragraph 2.3.1 and is determined by the type of the result desired. None of these other operations makes any use of the R register; the dZ output is not the overflow of this register.

There are basically three types of special operations which an integrator may perform: servo, decision, and adder operation.

2.3.3.1 SERVO AND DECISION OPERATIONS

The operation of an integrator when performing either servo or decision operation is essentially the same, differing only in the mathematical sense as determined mainly by the initial conditions set into the Y register of the integrator at the start of the problem. The value of the dZ incremental output on any given cycle (i) is determined by the dX increment and the value held in the Y register in that cycle. The following table shows the output for different values of Y.

(a) if $2 > |Y| \geq 1$, $dZ = 0$
(b) if $0 > Y > -1$, $dZ = -dx$
(c) if $0 < Y < 1$, $dZ = dX$
(d) if $Y = 0$, $dZ = 0$

The main difference between the operation of the integrator as a servo and as a decision integrator is the relative value about which the Y register value is operating.

As the name implies, a servo should always have an output which tends to reduce an error value to zero. The error value is actually the value which the integrator holds in its Y register. This indicates that the Y register value will operate around zero, and in reality the Y register value is usually very small. The conditions (b), (c) and (d) are the only ones used by a servo under these circumstances. A decision integrator, on the other hand, does not

necessarily have its Y value near zero. It can use the fact that the output increments essentially reverse the sign as the integrand passes through zero, [conditions (b) and (c)], or it can use the condition that the output increments become zero when the intergrand exceeds one in absolute value.

2.3.3.2 ADDERS

An adder generates an incremental output which is the accumulated sum of several incremental inputs. The operation of an adder is very similar to a servo in that the output increment is determined in the same manner as in a servo. The difference between the two is that an adder has the negative of its own output fed into itself as a dY input. The resulting operation of an adder is as follows:

> If the value in the Y register is slightly positive on a given iteration, the output increment will be positive. On the next iteration this positive increment will be multiplied by -1 and used as a dY incremental input. This resulting negative increment will be added algebraically to the Y register value, making this value less positive. This will continue until the value in the Y register is reduced to zero, at which time the output increments will become zero.
>
> Likewise, if the value in the Y register is negative, the output increment will be negative and the Y register value will be increased until it is zero.

The term "adder" is readily apparent when it is considered that other dY incremental inputs are also present on each iteration (the series of increments of the two variables which are being added together). These increments will tend to make the Y register value other than zero, i.e. positive when the increments are positive and negative when they are negative. Hence, when one of the incoming increments is positive, the adder will have one, and only one, positive output increment; when there is a negative input increment, a negative output increment will result.

It can be seen then, that whenever either of the two (or more) dy inputs have non-zero increments, the output of the adder is a non-zero increment and the number of incremental outputs of the adder is the sum of the incremental inputs from dy sources.

# SECTION III

# DA-1 LOGIC

## 3.0 DA-1 LOGIC

The previous sections are the building blocks for the understanding of the basic principles of the DA-1. It is presumed at this stage the reader understands the function of the integrator and the basic concepts of Boolean Algebra in addition to basic G-15 logic.

## 3.1 DA-1 BLOCK DIAGRAM (See Figures 16 and 17).

Physically, the registers mentioned in paragraph 1.4.3 are specifically located on the G-15 drum in lines 14, 15, 16 and 17. In addition to these lines, there are other long lines (address lines) used to specify integrator interconnections and two four word lines (Z lines) used to hold integrator outputs. M-18 is used to provide scaling and mode control of each integrator.

### 3.1.1

The understanding of the flow of information is made quite easy by Figure 16, Block Diagram and Figure 17, Flow Functional Diagram. This will give an overall picture of the flow of information in the DA-1.

As T29 of word I-1 is read, the dy register is loaded from one of four dy counters (which, in turn, get information from Z lines in addition to M10, 11, 12 and 13 "dy address lines "). At T29 of word I-1, the DA-1 starts looking in line 18 for a start pulse. At the next bit time after the start pulse has been read, two operations start:

(1) dy is added to Y(M-17) which forms YN (new);
(2) YN is multiplied by dx in the Z lines and this YN·dx is added to R to make RN = R + YNdx.

The dx pulse is looked for at the address specified by the dx address lines (M6, 7, 8 and 9). dx can be a positive one, negative one, or zero ($-1 \leq 0 \leq +1$). Both of the above operations are

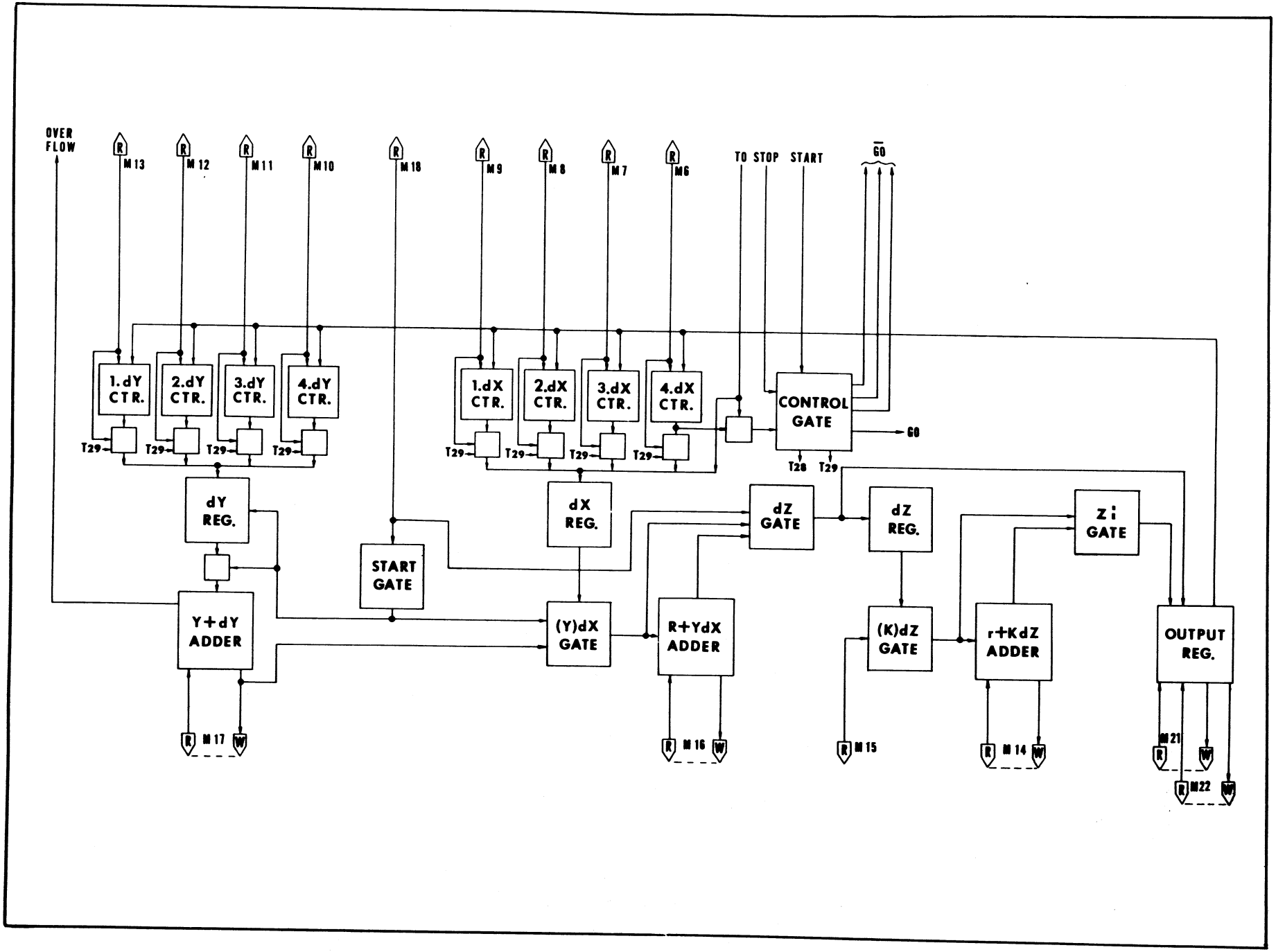



Figure 16. Block Diagram

complete at T28. The R register (M-16) may or may not have an overflow (i.e. a carry into the T28 position). The presence of a T28 carry of R + YN•dx is gated into the dz gate. Lack of an overflow in the R register denotes that dz = 0. The R register does not have a sign bit and is absolute. YN•dx can have either sign. The R register can be made to overflow either positively or negatively. Therefore, the sign of the overflow is the sign of the dz output: it is also the sign of (Y + dy) dx = YN•dx. If this dz output (unscaled output) of integrator "I" is to be used as an input to another integrator, it must be read during T28 of word I.

3.1.2 The dz output, existence and sign, is held throughout word time I +1 to control the addition of K•dz into r. K (the multiplier constant) is held in M-15 and gated into the (K dz) gate where it scales off dz. This output K dz is added into r. The addition begins at T29 of word I and continues through T27 of word I +1.

The sign of dz controls the complementation of K if necessary. The r register is held in line 14. At T28 of word I +1, the existence and direction of overflow of the r register provides the existence and sign of the Kdz output of integrator I. The Kdz output of integrator I is held until T1 of word I +2, at which time it is recorded on the two Z lines, (M21, 22). The dz (Unmultiplied) output is never recorded on the Z lines. The scaled output of integrator "I" is recorded on two Z lines, i.e. M-21 for existence and M-22 for sign at T1 of word I +2. While the DA-1 is in operation, these two lines undergo a precession of one bit in each four word times, so the output of integrator I is in T2 at word I +6, in T3 at I +10, and in T27 at word I +106.

Figure 18 shows the bit positions of all the integrator outputs by integrator number during word times 3, 4, 5, and 6.

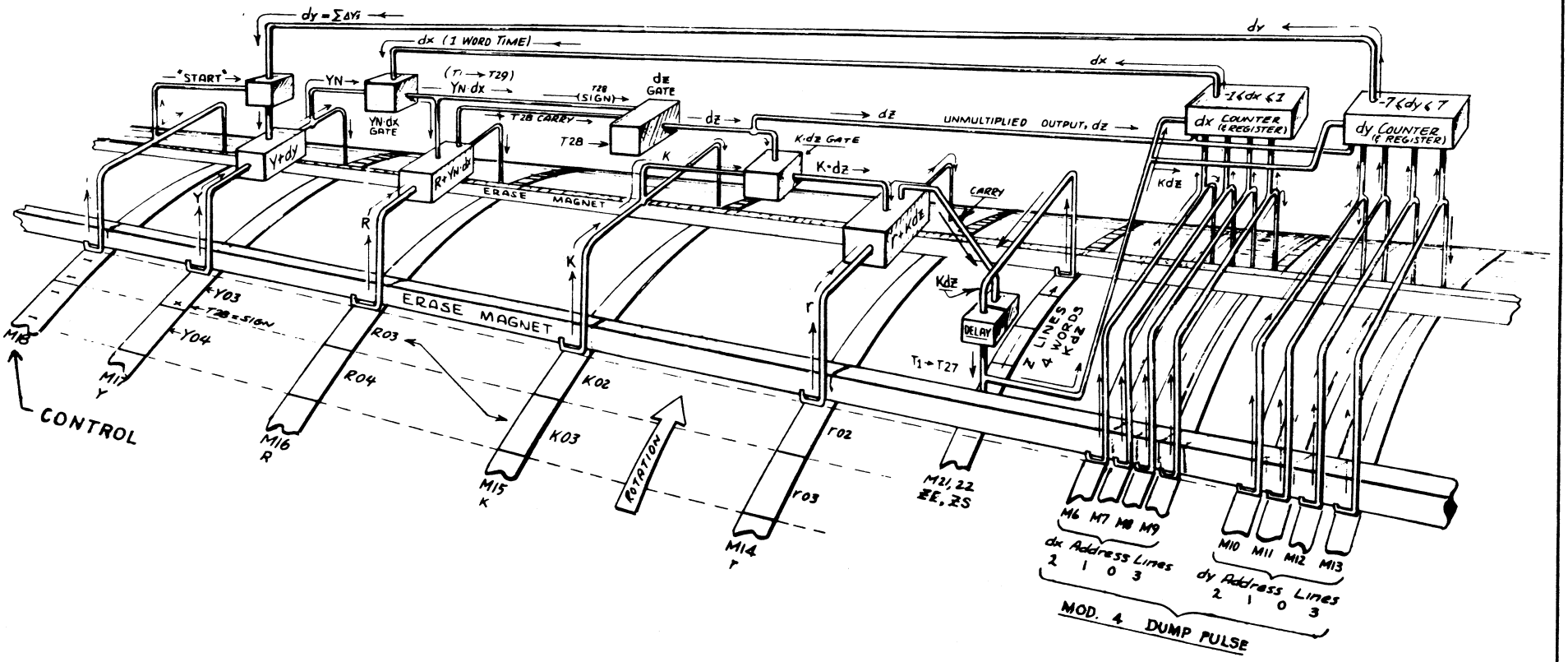



Figure 17. DA-1 Flow Functional Diagram

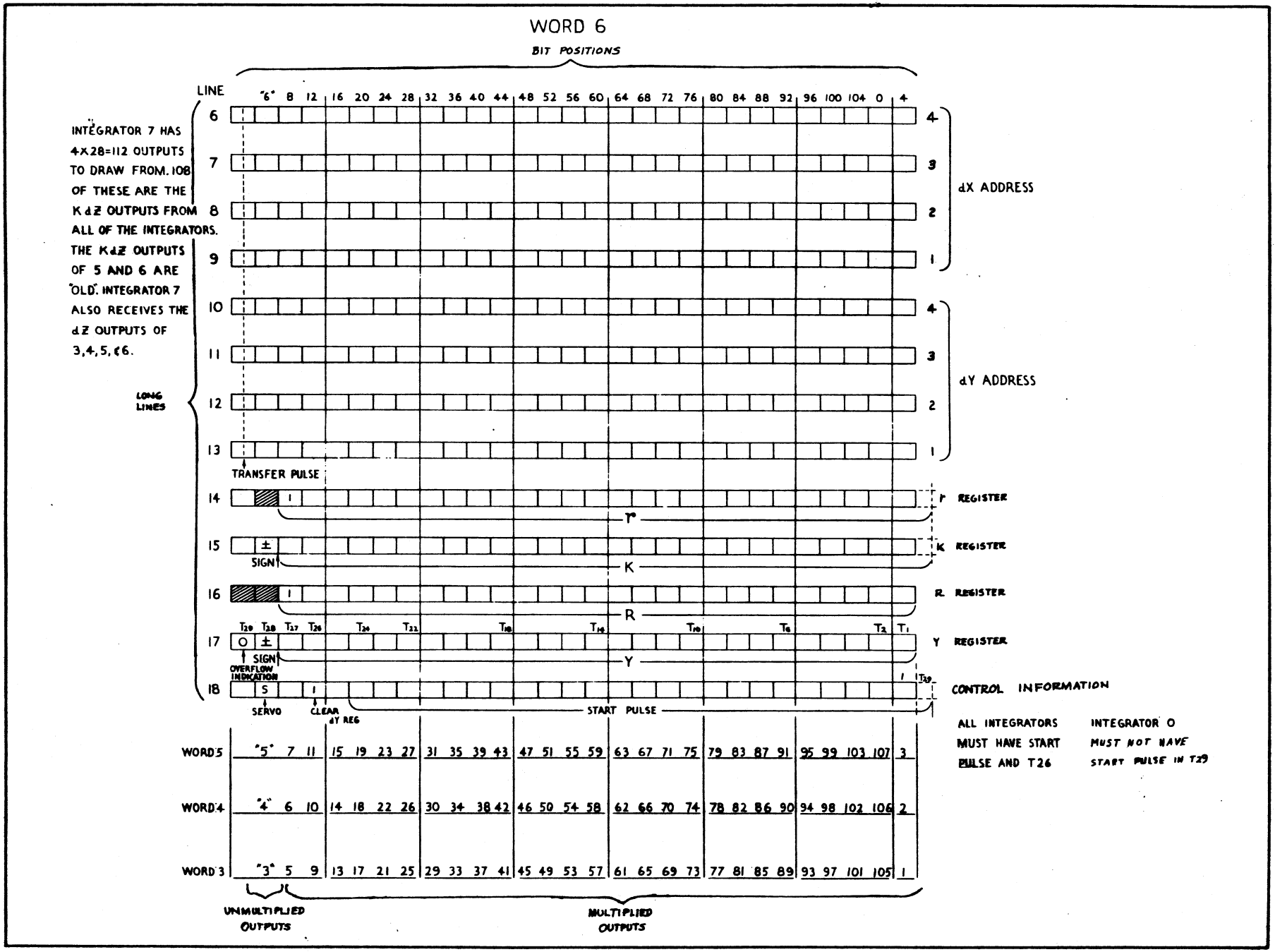



Figure 18. Chart, Program Information

## 3.2 TIMING

This section includes the discussion of the various important signals which control the logical operation of the DA-1. This will help in understanding the internal logic of the machine to be discussed in the following sections. The flow of information in DA-1 has already been discussed.

The DAPPER-1 and DAPPER-2 programming routines convert decimal integrator numbers into bit positions during the specification of integrator interconnections; floating point decimal numbers and other scaling data into words and bit-positions during numerical input, DA-1 words into decimal output form for typeout; and provide the user with other input and control operations.

### 3.2.1 INTEGRAND (See Figure 18).

At the bottom of Figure 18 are shown words 17.06 and 18.06. These contain the integrand and timing control information, respectively, for integrator 06. Integrand number I corresponds to word time I all around the drum (00-u7).

The complement form of Y and dY permits the use of the trailing sign bit. Like AR register in the G-15, no "second pass" is needed to correct the sign bit.

With the 4 bit dy register, it is possible to add the least significant bit of dy into T25 position of the integrand. Other circuitry establishes the low-order (right hand) limit of this addition at T1. One word of line 18 (M18) is shown in Figure 19.

The start pulse for the Y + dy addition in integrator I is initiated by the right most bit in M18-I. This bit must be located one bit to the right of the position where the least significant bit of dy is to be added. It can be located in any position between T1 and T24 of word I, or it can be located in T29 of word (I-1) in which

case dy is added into T1 of word I. Integrator 00 is an exception because the start pulse must be located between T1 and T24 inclusive of word 00 and can not be in word u7. There are two other important pulses in M18.

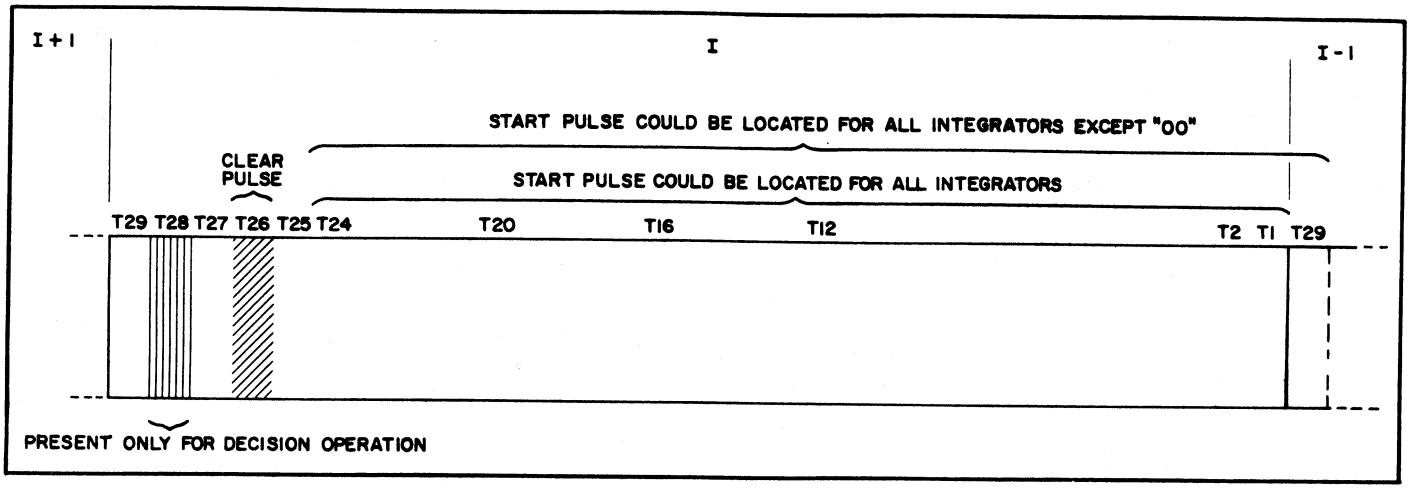



Figure 19. One Word Structure of Line M18

Position T26 must have a "one" bit because it is a timing control pulse required by the DA-1 hardware and has no programming importance. The dy register is cleared beginning at T27 in preparation for the processing of the following integrator (I + 1).

T28 bit position specifies whether the output of the integrator is of the dz = Ydx type or of the dz = (Sign Y) dx type. A "one" bit in T28 (M-18) position calls for decision operation. The result of R + (Y +dy) dx addition is ignored when T28 is present. One flip-flop detects the (non-zero) existence of (Y + dy)dx and at T28 this flip-flop is checked to determine the existence of dz output on decision operation. This output has the sign of (Y + dy)dx.

### 3.2.2 K, R, and "r" REGISTERS (See Figure 20).

The word structure of constant K (M-15) is quite similar to that of Y. The sign bit is in T28; the numerical portion extends from

T27 down through T29 of the previous word (I-1) and is complemented if negative. Since the value of K is not changed by the DA-1, there is no provision for overflow and T29 is available for numerical use. K register for integrator I is located in word time I + 1 because the Kdz output is restricted by the unmultiplied dz output which is not available until T28.

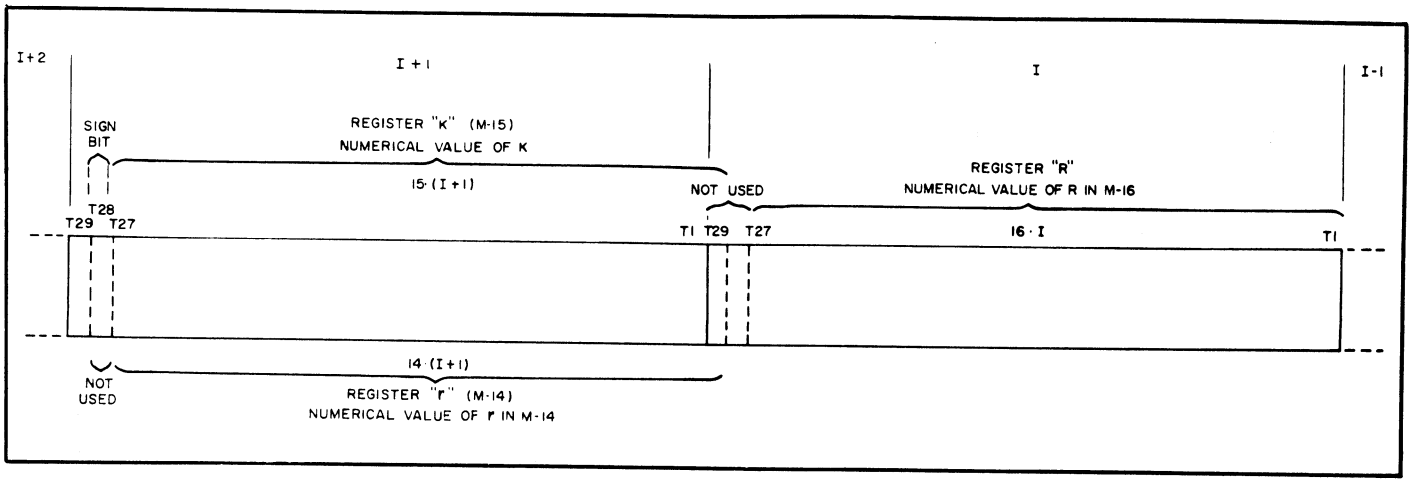



Figure 20. Location and Word Structure of K, R and "r" Registers

The output of an integrator is the result of accumulated addition of $\pm$ Y into another register called the R register. R is held in line 16 (in word I for integrator I). The value of R is restricted to be positive, therefore, bit positions of T28 and T29 are not used by DA-1. The numerical value of R extends from T1 to T27.

Register "r" corresponds exactly to register R in function. The value of "r" register for integrator I is found in line 14 (I + 1). The "r" register for integrator I extends from T27 of word I + 1 down to T29 of word I as does the numerical portion of K. T28 is not used.

### 3.2.3 INFORMATION ADDRESSING

Because of the addressing scheme based on 4 word short lines, each integrator output is available for counting only every four word times.

Therefore, any dy counter must search for four word times in order to be able to accept information from each of the 108 integrators. Four dy counters are necessary because of the length of the Z lines. The most appropriate arrangement of the counters is that they use the same counter for integrators 00, 04, 08 - u0, and u4. A second counter is used for 01, 05, 09, ..., and u5. Similarly, with the other two counters; the control of the counters is at the discretion of the programmer. Other possible arrangements restrict the number of integrator interconnections. However, in specific problems this may be feasible and could free some long lines of the memory for general purpose use.

The addressing of dx inputs is identical to that of dy inputs. Lines 6, 7, 8 and 9 are associated with dx counters 4, 3, 2, and 1 respectively. Each of the dx counters hold only one count (existence and sign), and the circuit operation is such that if more than one dx input is addressed, each term will be set as though the multiple counts were presented simultaneously at "OR" gates on the input terms.

The association of specific address lines with specific integrators is at the programmer's discretion. The table given below indicates the choices made for DAPPER-1 and DAPPER-2 (Programming Routines for DA-1).

3.2.4

| LINE | INTEGRATOR | COUNTER | COUNTER DUMP PULSE LOCATIONS IN T29 OF WORDS: |
|---|---|---|---|
| 09 | 0 module 4 | dx #1 | 3 module 4 |
| 08 | 1 " 4 | dx #2 | 0 " 4 |
| 07 | 2 " 4 | dx #3 | 1 " 4 |
| 06 | 3 " 4 | dx #4 | 2 " 4 |
| 13 | 0 " 4 | dy #1 YA | 3 " 4 |
| 12 | 1 " 4 | dy #2 YB | 0 " 4 |
| 11 | 2 " 4 | dy #3 YC | 1 " 4 |
| 10 | 3 " 4 | dy #4 YD | 2 " 4 |

3.2.5 Experiments with integrator numbering in a specific problem may indicate that not all counters need to be used for 108 integrator operations. The counters will no longer search in uniform four word blocks. In this event, one of the address line selector switches inside the back panel of the DA-1 (shown in Figure 21) can be opened, disconnecting the associated counter. Thus, the associated long line can be used for a general purpose program.

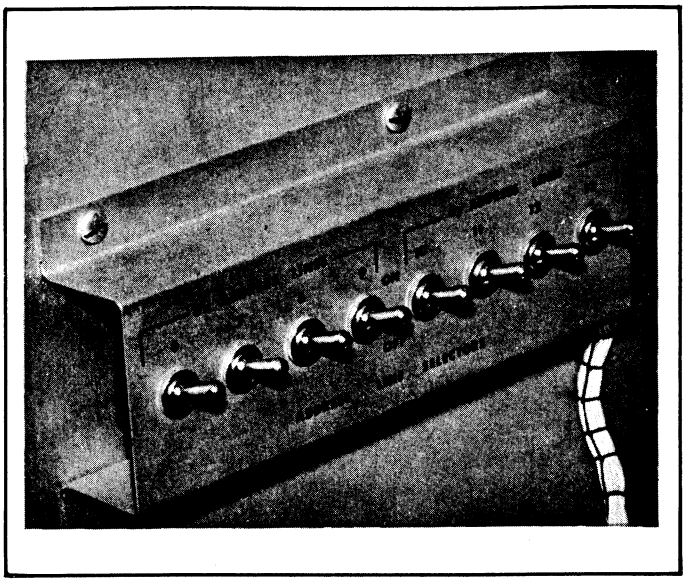


Figure 21. Address Line Selector Switches

## 3.3 MANUAL CONTROL

The operation of the DA-1 is controlled from the G-15 typewriter.

### 3.3.1 G-15 COMMANDS

Command 01931 (27z) unconditionally starts the DA-1. This command must be operative during word time u7. The dx inputs are supplied at T0 time (= T29 of u7) to integrator "00" regardless of any counter dump pulses or lack thereof in the dx address lines. Only inputs to integrator "00" are provided and the integrand must be non-zero to obtain any output.

Command 11931 (67z) unconditionally halts the DA-1. This command must also be operative during word time u7.

Command 32821 (z9z) tests the GO flip-flop on the DA-1. The CQ test flip-flop in the G-15 is set if the DA-1 is off.

### 3.3.2 OVERFLOWS

An overflow in any integrand immediately halts DA-1 computation and sets the G-15 overflow flip-flop.

If dx counter #1 (associated with line 09) contains a non-zero dx at time T0, DA-1 computation will halt (the automatic dx inputs to integrator "00" will no longer be supplied). The G-15 overflow flip-flop is not set.

In the DAPPER ROUTINE, dx counter #1 is programmed to work with

integrator "00" by means of T29 in Line 9, Word 107.

When DAPPER detects a DA-1 halt during normal computation, it next tests the overflow flip-flop to decide whether to call for a normal typeout or to call for an overflow seeking procedure which allows the typeout of the computation results at regular intervals.

## 3.4 LOGICAL EQUATIONS

In this section the logical equations are discussed and the respective timing diagrams are given. The equations are discussed according to the flow of information with some necessary examples. A frequent reference should be made to Figures 23, 24, and 25 for the following discussion.

### 3.4.1 Y + dy ADDER Reference Figure 22.

The three terms involved in this adder are Y, dy and YC. The addition is bit by bit and the output of the adder is YN and $\overline{YN}$.

Y is the augend and is stored in M17. dy is the addend and is the output of the dy register. The additional carry term YC completes the addition of dy to Y. Refer to Figure 22 to understand the arrangement of the eight "and" gates which make various addition terms for the adder to form YN and $\overline{YN}$.

dy = Y1·ST Y1 is the output of Y1 flip-flop of the dy register and ST is the output of start flip-flow ($ST_s = M18 \cdot \overline{ST}$). The bit by bit sum of three terms is:

(1) $$YN = (Y \cdot dy \cdot YC) + (\overline{Y} \cdot dy \cdot \overline{YC}) + (Y \cdot \overline{dy} \cdot \overline{YC}) + (\overline{Y} \cdot \overline{dy} \cdot YC)$$
$$= (Y \cdot Y1 \cdot ST \cdot YC) + (\overline{Y} \cdot Y1 \cdot ST \cdot \overline{YC}) + Y \cdot \overline{Y1 \cdot ST} \cdot \overline{YC}) + (\overline{Y} \cdot \overline{Y1 \cdot ST} \cdot YC)$$

and,

(2) $$\overline{YN} = (Y \cdot \overline{dy} \cdot YC) + (\overline{Y} \cdot dy \cdot YC) + (Y \cdot dy \cdot \overline{YC}) + (\overline{Y} \cdot \overline{dy} \cdot \overline{YC})$$
$$= (Y \cdot \overline{Y1 \cdot ST} \cdot YC) + (\overline{Y} \cdot Y1 \cdot ST \cdot YC) + (Y \cdot Y1 \cdot ST \cdot \overline{YC}) + \overline{Y} \cdot \overline{Y1 \cdot ST} \cdot \overline{YC})$$

The addition of pulses in case of YN is shown in the Timing Diagram No. 1 for integrator No. "01". The timing diagram for $\overline{YN}$ can be similarly drawn.

3.4.1.1 **YC ff (CARRY)**

YC is the output of YC flip-flop which handles the carry in the process of addition. The handling of carry is very important as it is the end carry that causes an overflow (T29) which halts the DA-1. Therefore, the setting and resetting of YC flip-flop is controlled by various "and" gates to handle the different possibilities. The examples given in the following pages will show clearly where the overflow occurs and how it is prevented by controlling the YC flip-flop. The terms required to set and reset YC flip-flop are as follows:

$$YC_s = \overline{YC}\cdot Y\cdot dy = \overline{YC}\cdot Y\cdot Y1\cdot ST$$

3.4.1.2 If YC is not set and the addend and augend are both high, the combination of these terms sets YC flip-flop. Refer to Timing Diagram No. 2 for $YC_s$. It is clear that $YC_s$ sets and not the output of YC flip-flop. $\overline{YC}$ is necessary in order to avoid the conflict with case C.

$$YC_r = \overline{T28}\cdot YC\cdot\overline{dy}\cdot\overline{Y} + T28\cdot YC\cdot(Y + dy)$$

3.4.1.3 For normal operation of the adder, the reset equation need only be $YC_r = YC\cdot\overline{Y}\cdot\overline{dy}$, but to detect against overflow in the adder ($Y \geq +1$, $Y \leq -1$) an additional feature has been incorporated. Refer to Timing Diagram, No. 3.

Normally, YC is reset at times other than T28 (sign bit). This accounts for the first term of the $YC_r$ equation. Refer to example D with regard to T28.

The second term $\left[(T28\cdot YC\cdot(Y + dy) = (T28\cdot YC\cdot Y) + (T28\cdot YC\cdot dy)\right]$ is

obtained by the use of two "and" gates. The justification of this term and T28 of the first term is shown in the following examples:

**Note**

The negative numbers are in a complemented form in the DA-1, and both the complemented and uncomplemented forms of YN are shown in the Chart, page 38.

3.4.1.4 Examples A, B, and C all show cases where the limits to the Y register have not been exceeded. Note that in each of these cases Y, dy, or both are negative (as indicated by T28 bit). Also, each case results in a set condition of the YC flip-flop at T28. Examples D and E show additions which cause Y to exceed the limits of the register. Example D shows the addition of a (+) positive dy to a (+) positive Y, resulting in YC flip-flop being high at T28. Example E shows addition of a negative dy to a negative Y, resulting in YC flip-flop being in the reset condition at T28, but high at T29. Both cases show the state of Y register overflow. D has the YC flip-flop high at T28 and E has the YC flip-flop high at T29. Since normal operation implies zeros in both Y and dy at T29, it may be assumed that a set condition applied on YC at T28 will result in YC being set at T29. To determine, by observation of YC, whether or not an overflow has occurred, we must be able to differentiate between Examples A, B and C; and between Examples D and E. The qualifying signal $\overline{T28}$ in the first term ($\overline{T28}\cdot YC\cdot\overline{dy}\cdot\overline{Y}$) is also necessary for the differentiation because YC must not be reset at T28 if Y and dy are positive - (Example D).

The second term of the reset equation T28.YC.(Y + dy) accomplishes this differentiation by resetting the YC flip-flop at T28 if Y, dy, or both are negative and YC is set. This means that YC will be set at T29 if and only if the Y register has overflowed. If YC is set at T29 (Y and dy being zero), one bit will be written on YN in the T29 position (1 + 0 + 0 = 1). This one bit in the T29 position of YN is sensed as an overflow and halts the DA-1. See Timing Diagram, No. 3 for Example E which illustrates the overflow.

| Y NOT OVERFLOWED | | NORMAL Y + dy ADDER | | | | | | ACTUAL Y + dy ADDER | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | T29 Sign | T28 | T27 | T26 | T25 | T24--T1 | T29 Sign | T28 | T27 | T26 | T25 | T24 --T1 |
| (A) | YC = | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| | Y = + 1/2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| | dy = - 1/2 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| YN = Y + | dy = 0 = | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| (B) | YC = = | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| | Y = - 1/2 = | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| | dy = + 1/2 = | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| YN = Y + | dy = 0 = | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| (C) | YC = = | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| | Y = - 1/8 = | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| | dy = - 1/8 = | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| (COMPLEMENTED) YN = Y + dy | = - 1/4 = | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| YN = - 1/4 (UNCOMPLEMENTED) → | | | | | | | | 0 | 1 | 0 | 1 | 0 | 0 |
| Y OVERFLOWED | | | | | | | | | | | | | |
| (D) | YC = = | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| | Y = + 3/4 = | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| | dy = + 1/2 = | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| YN = Y + | dy = +1 1/4 = | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| (E) | YC = = | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| | Y = - 3/4 = | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| | dy = - 1/2 = | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| (COMPLEMENTED) YN = Y + dy | = -1 1/4 = | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| YN = - 1 1/4 (UNCOMPLEMENTED) → | | | | | | | | 1 | 1 | 0 | 1 | 0 | 0 |

NOTE: ALL THE NEGATIVE NUMBERS ARE IN COMPLEMENTED FORM.



COMPLEMENTED AND UNCOMPLEMENTED FORMS OF YN, CHART

3.4.2 YNdx GATE (See Figure 22.)

YN and $\overline{YN}$ "the output of (Y + dy) adder" is input to YN•dx gates which consist of two "and" gates, two cathode followers (C.F.) and a buffer inverter (B.I.). The other inputs to these gates are the outputs of two flip-flops XS and XE (dx Sign and dx Exists). Refer to Figure 22 which shows the four input terms to the two "and" gates which are shown below:

$$\boxed{YN\ dx = YN \cdot XE \cdot \overline{XS} \cdot ST + \overline{YN} \cdot XS \cdot XE \cdot ST = ST \cdot XE\ (YN \cdot \overline{XS} + \overline{YN} \cdot XS)}$$

(Refer to Timing Diagram, No. 4.)

3.4.2.1 The following explanation is given for the justification of each term involved in the output of these gates.

3.4.2.2 $\boxed{ST \cdot XE}$ This term shows that a start pulse has been read in M18 and that the integrator has received a dx pulse.

3.4.2.3 $\boxed{\overline{XS} \cdot YN}$ shows that the sign of dx was positive and the YN information is passed through unmodified.

3.4.2.4 $\boxed{XS \cdot \overline{YN}}$ indicates that the sign of dx was negative, and the $\overline{YN}$ is passed through.

3.4.2.5 The following example is given to show how to get ONE'S complement and TWO'S complement as used in DA-1 arithmetic.

Example: If YN = 0.10101, then

$\overline{YN}$ = 1.01010

Now ONE'S complement of YN is

1.11111 (ONE'S)

(Subtract YN) = -0.10101 (-YN)

1.01010 which is the same as $\overline{YN}$.

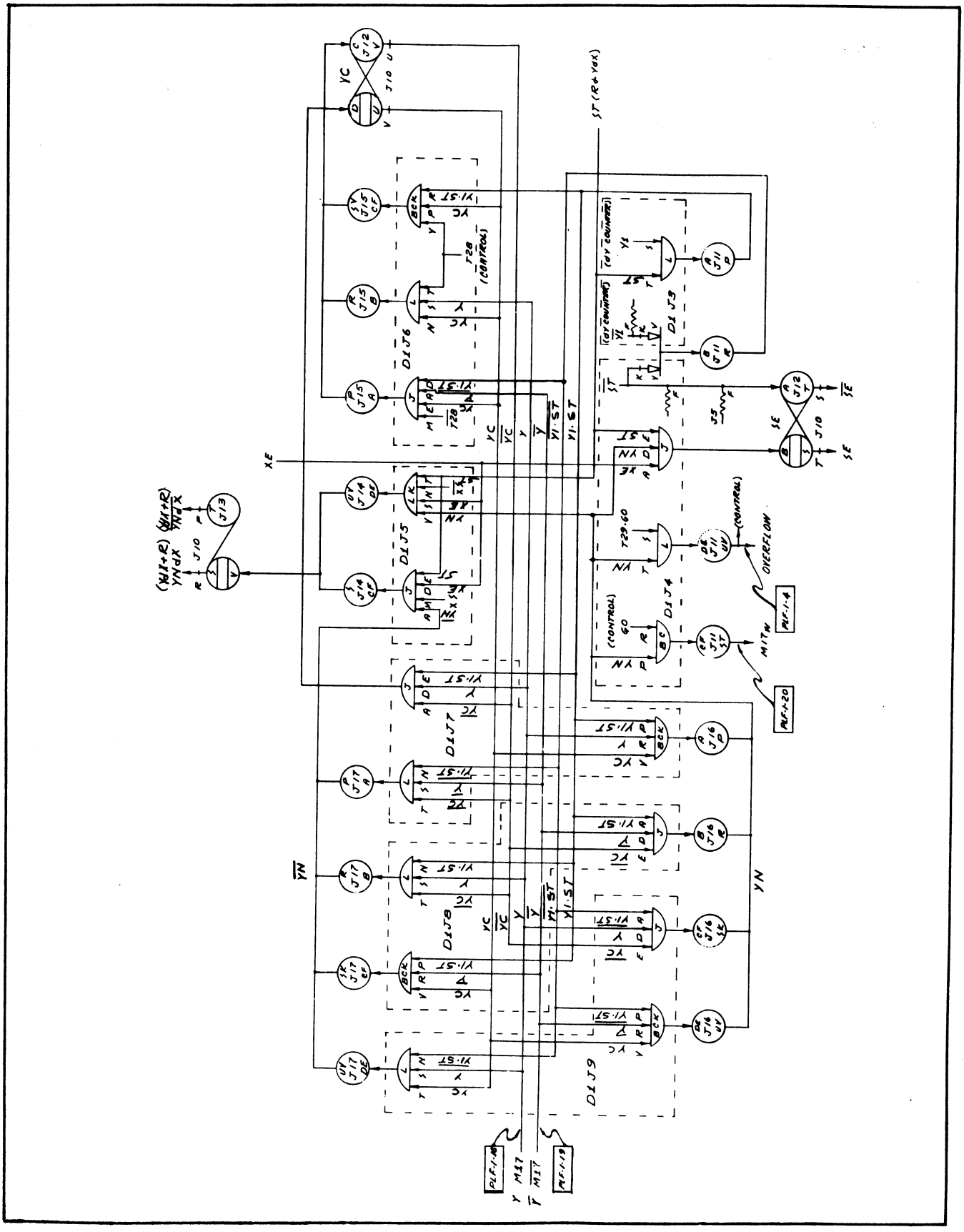


Figure 22. Schematic "Y" + "dy" Adder & YN·dx

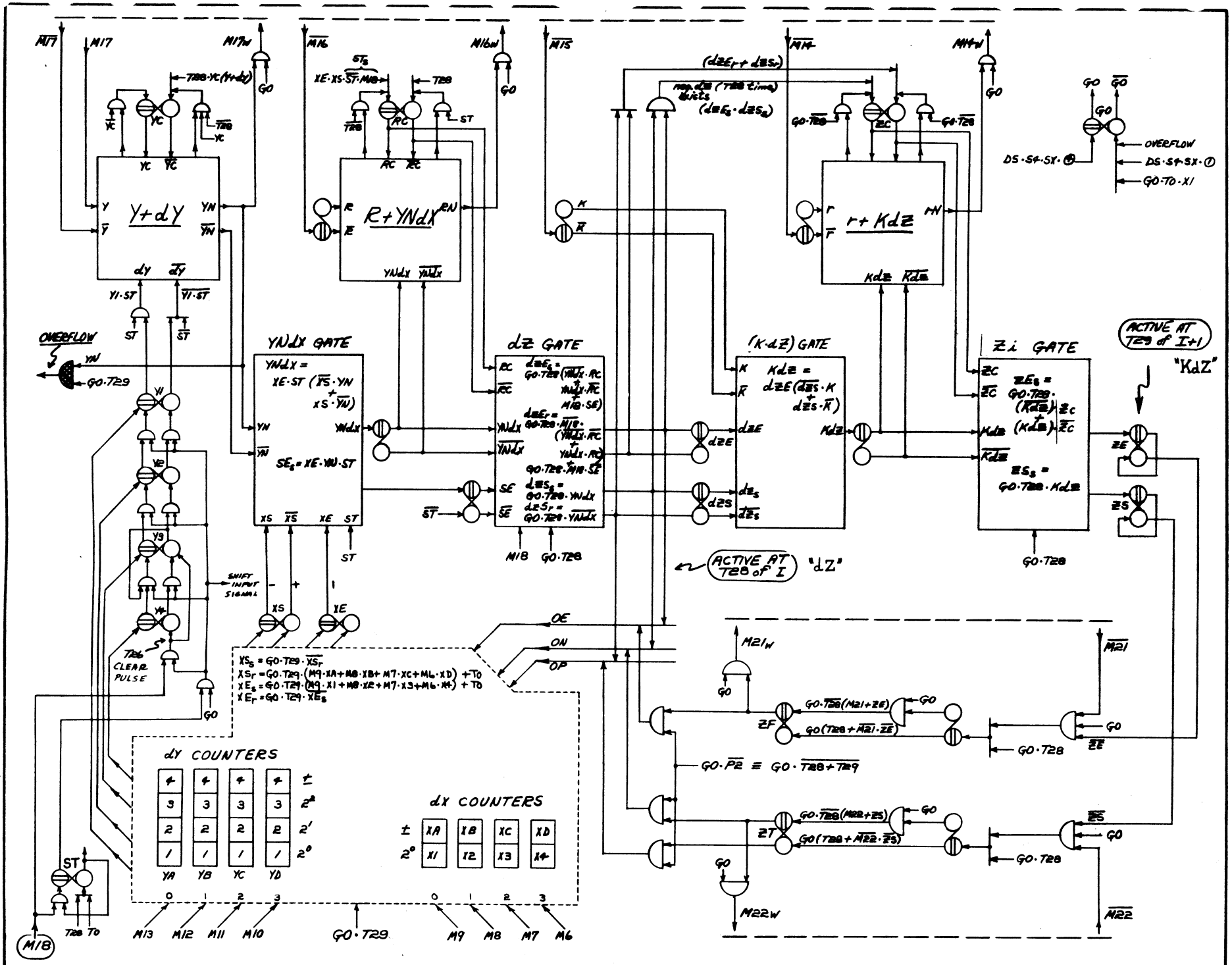



Figure 23. DA-1 Simplified Logic

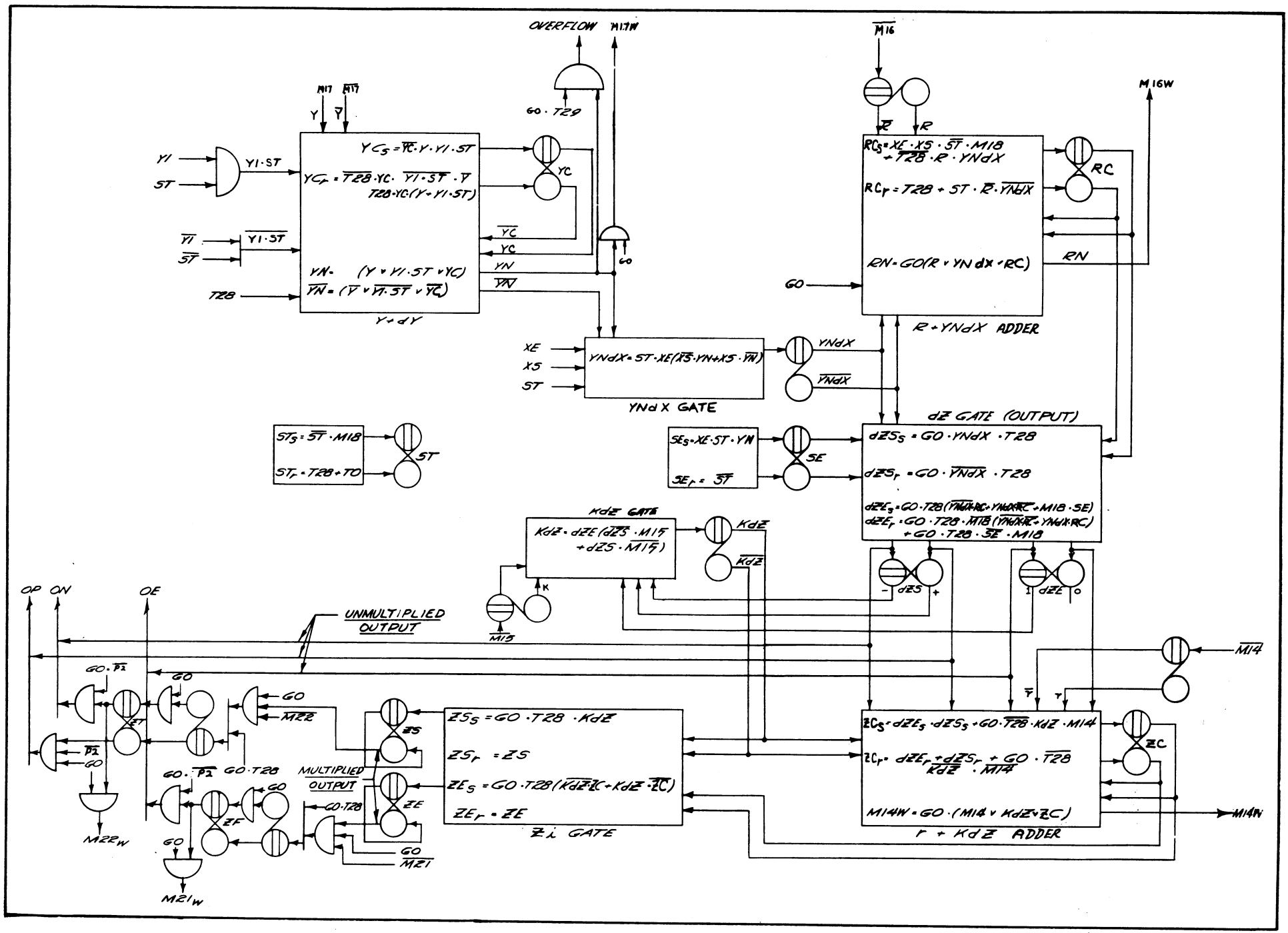



Figure 24. DA-1 Arithmetic Logical Equations

For TWO'S complement of YN:

| | |
|---|---|
| | 1.11111 |
| Add one bit to the least significant bit position: | +0.00001 |
| | 10.00000 |
| (Subtract YN) = | -0.10101 |
| | 1.01011 which is the same as $\overline{YN}$, plus one bit in the least significant bit position. |

Note that $\overline{YN}$ is equal to ONE'S complement of YN. To get the TWO'S complement of YN, it is only necessary to add a single bit to the $\overline{YN}$ in the least significant bit position.

3.4.3 R + YN dx ADDER (Refer to Figure 26.)

3.4.3.1 R + YN dx = RN

R is stored in M-16 of the G-15 memory. YN dx is the output of the YN dx gate through a B.I. RC is the output of carry flip-flop for the adder and R is the information already recorded in M-16. RN is recorded back in M-16 and the carry (RC) is gated in to the dZ gate.

$$RN = GO\ \left[(R \cdot \overline{YNdx} \cdot \overline{RC}) + (\overline{R} \cdot YNdx \cdot \overline{RC}) + (\overline{R} \cdot \overline{YNdx} \cdot RC) + (R \cdot YNdx \cdot RC)\right]$$

After the above accumulation is made, the new value of R (RN) is recorded back on M-16. The four "and" gates (See Figure 23.) make the above addition. The bit by bit addition of the above terms is shown in Timing Diagram, No. 5.

3.4.3.2 RC (CARRY FLIP-FLOP)

Refer to Timing Diagram, No. 6.

$$RC_s = XE \cdot XS \cdot \overline{ST} \cdot M18 + \overline{T28} \cdot R \cdot YNdx$$

The two "AND" gates formed by the above signals are required to set the RC flip-flop. The justification of the above equation is given as follows:

$$XE \cdot XS \cdot \overline{ST} \cdot M18$$

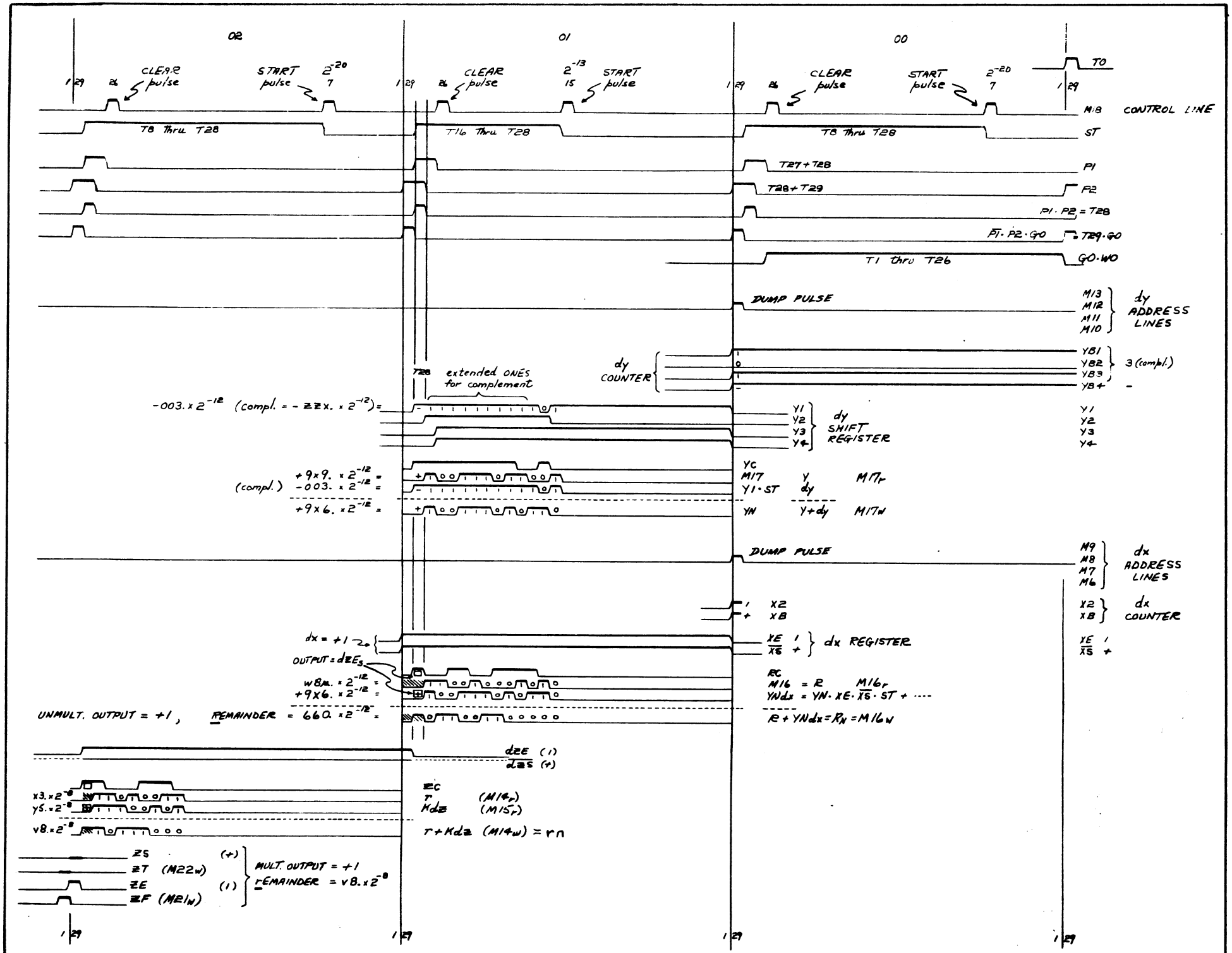



Figure 25. Timing Diagram of Integrator "01"

This is the term necessary to adjust for the ONE'S complementation mentioned before while discussing YNdx gate. It indicates that:

(1) A dx has been addressed to the integrator.

(2) The dx has a negative sign.

(3) ST flip-flop is in the reset condition and a start pulse is present in M-18.

Together, these terms set RC flip-flop for the first bit of an incoming YNdx resulting from a negative (-) dx to correct the ONE'S complement to TWO'S complement. Note that no information gets through the YNdx gate until the ST flip-flop is set. The same signals that set the ST flip-flop also sets RC if dx is negative, so that the first bit through the YNdx gate has one added to it.

$$\boxed{\overline{T28} \cdot R \cdot YNdx}$$

The output of this "AND" gate sets RC when the addend and augend are both one.

$RC_r = T28 + ST \cdot \overline{R} \cdot \overline{YNdx}$ (See Timing Diagram, No. 7)

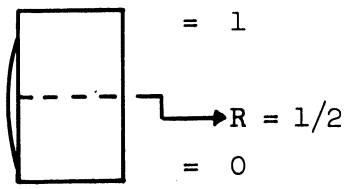
RC flip-flop is reset unconditionally by T28 at the end of each word so that the next integrator is assurred of starting properly.

$$\boxed{ST \cdot \overline{R} \cdot \overline{YNdx}}$$

This term is to reset RC when the addend and augend are both zero. The RC reset terms are shown in Timing Diagram, No. 7. The output of this "AND" gate is used to make sure that RC is reset even when there is nothing in the addend and augend terms.

3.4.4 dz GATE (See Figure 26 and Timing Diagram, No. 8.)

The contents of the R register are always considered positive, i.e: $(0 \leq R < +1)$ or +YNdx = 1




-YNdx = 0

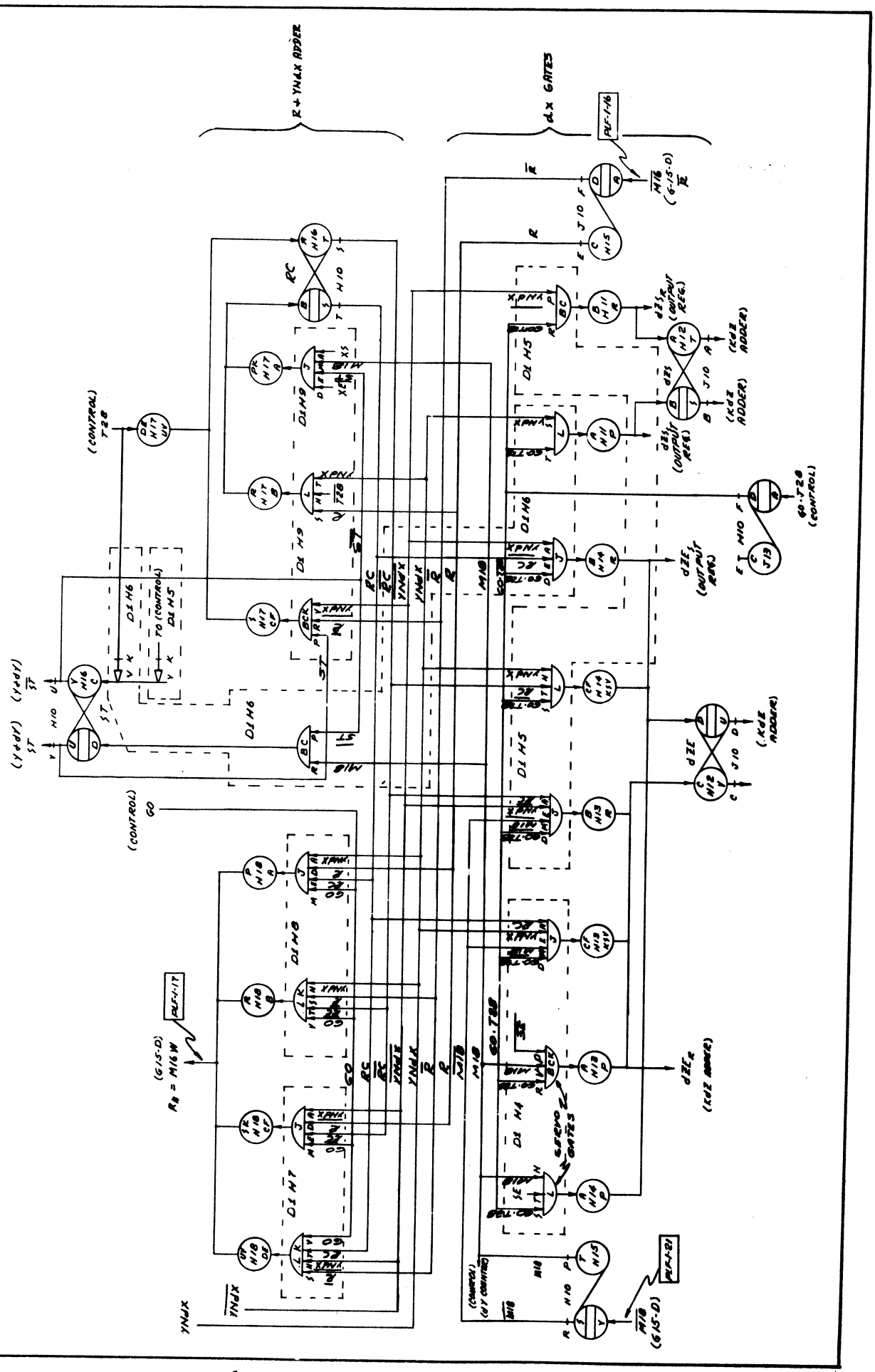





Figure 26. Schematic, R + YNdx Adder & dz Gate

YNdx is added to R. If YNdx is positive (+), the magnitude of the number in R can only increase. If it increases above the upper limit, +1, (i.e. R + YNdx $\geq$ + 1), this is called a positive (+) dz. If YNdx is negative (-), the magnitude of the number in R can only decrease. If it decreases below the lower limit, 0, (i.e. R-YNdx $< 0$), this is called a negative (-) dz.

Examples A through D, following, show whether or not the limits of the R register have been exceeded.

WHEN YNdx IS POSITIVE:

| A | T29 | T28 | T27 | T26 | T25 | T24--T1 | |
|---|---|---|---|---|---|---|---|
| R = 1/2 | | 0 | 1 | 0 | 0 | | |
| YNdx = 5/8 | | 0 | 1 | 0 | 1 | | |
| RN = 9/8 | | 1 | 0 | 0 | 1 | | |
| RC flip-flop | | 1 | 0 | 0 | 0 | | |

| B | T29 | T28 | T27 | T26 | T25 | T24--T1 | |
|---|---|---|---|---|---|---|---|
| R = 1/2 | | 0 | 1 | 0 | 0 | | |
| Yndx = 1/8 | | 0 | 0 | 0 | 1 | | |
| RN = 5/8 | | 0 | 1 | 0 | 1 | | |
| RC flip-flop | | 0 | 0 | 0 | 0 | | |

WHEN YNdx IS NEGATIVE:

| | T29 | T28 | T27 | T26 | T25 | T24--T1 | |
|---|---|---|---|---|---|---|---|
| C | | | | | | | |
| R = 1/2 | | 0 | 1 | 0 | 0 | | |
| YNdx = -5/8 | | 1 | 0 | 1 | 1 | | (COMPLEMENTED) |
| RN = -1/8 | | 1 | 1 | 1 | 1 | | (COMPLEMENTED) |
| RC flip-flop | | 0 | 0 | 0 | 0 | | |

| D | T29 | T28 | T27 | T26 | T25 | T24--T1 | |
|---|---|---|---|---|---|---|---|
| R = 1/2 | | 0 | 1 | 0 | 0 | | |
| YNdx = -1/4 | | 1 | 1 | 1 | 0 | | (COMPLEMENTED) |
| RN = 1/4 | 0 | 0 | 0 | 1 | 0 | | |
| RC flip-flop | 0 | 1 | 0 | 0 | 0 | | |

Examples A and B both show YNdx as positive, while C and D both show YNdx as negative. Examples A and C both show overflow (CARRY) of the R register, while B and D both show no overflow (CARRY) of the R register. When YNdx is positive and RC is set at T28, (Example A), an overflow (CARRY) or dz exists. Since R was increased by YNdx, this is a positive (+) dz.

When YNdx is negative and RC is reset at T28, (Example C), an overflow or dz exists. Since R was decreased by YNdx, this is a (-) negative dz. We can determine the existence of an overflow by observing the sign of YNdx and the condition of the RC flip-flop. The sign of YNdx is the sign of dz as is shown in the following table.

| SIGN OF YNdx | CONDITION OF RC FLIP-FLOP AT T28 | dz OUTPUT (EXISTENCE AND SIGN) | |
|---|---|---|---|
| + | Set | + dz | +1 Ternary |
| + | Reset | 0 | 0 Output |
| - | Set | 0 | |
| - | Reset | - dz | -1 |

Since the dz output is ternary, two flip-flops are necessary to store a dz. These are called dZE (existence) and dZS (sign) flip-flops. Refer to Figure 26. The following table shows all the possible conditions of dZ, dZE and dZS flip-flops.

| dZ | dZE | dZS |
|---|---|---|
| + 1 | Set | Reset |
| ± 0 | Reset | Reset or Set |
| - 1 | Set | Set |

The above discussion is to show how the dZ output (sign and existence) is handled in general. The following discussion shows how these dZ flip-flops are controlled by various signals.

3.4.4.1 dZS FLIP-FLOP

$$\boxed{dZS_s = GO \cdot YNdX \cdot T28}$$

$dZS_s$ is the set condition of the dZS flip-flop which shows that dZ has a sign and it is negative. The controlling signals are:

$$\boxed{GO \cdot YNdX \cdot T28}$$

This indicates that the sign of YNdX is negative. Since the output pulse should have the same sign as YNdX, this becomes a setting term to the dZS flip-flop. If there is a dZ pulse, it is negative as R is being diminished. Refer to Timing Diagram, No. 8, which shows the bit by bit position of these terms.

$$\boxed{dZS_r = GO \cdot \overline{YNdX} \cdot T28}$$

The reset condition of dZS flip-flop shows that sign of YNdX is positive. The dZ output is positive, since R is being increased. Timing Diagram, No. 8 shows the bit positions of the controlling terms for the resetting of dZS flip-flop.

3.4.4.2 dZE FLIP-FLOP (See Timing Diagram, No.9)

$$\boxed{dZE_s = GO \cdot T28\ (YNdX \cdot RC + YNdX \cdot \overline{RC} + M18 \cdot SE)}$$

dZE flip-flop is to detect the existence of dZ signal. The above equation shows the terms which control the setting of dZE flip-flop. The justification of each individual term is discussed below and the timing relationship is shown in Timing Diagram, No. 9. The dZ output can be taken directly to the output register without multiplying it, and also to the KdZ gate for multiplying by constant K.

The overall effect of the pulse in T28 of Line 18 (M18) is to control the dZE flip-flop by the contents of YN rather than by the arithmetic in the R + YNdX adder.

## Note

There is no change in the operation of dZS when shifting to decision operation. The result is that YN is non-zero and there will be an output of like sign as YNdX.

This is interpreted by the programmer as dX multiplied by the sign of YN.

$$\boxed{GO \cdot T28}$$

This term restricts the setting of dZE to T28 (sign bit) when the DA-1 is computing. It is the qualifying term for all the "and" gates which set the dZE flip-flop. This is why the output of dZE is one word time delayed.

$$\boxed{\overline{YNdX} \cdot RC}$$

This term indicates that YNdX has to be positive and if RC is set at this time, i.e. RN is > + 1, there is an overflow that means dZ exists. Therefore, dZE flip-flop gets set to note this dZ.

$$\boxed{YNdX \cdot \overline{RC}}$$

This term causes dZE to get set when YNdX is negative and the arithmetic does not cause RC to get set. This indicates that a (-) negative dZ has been generated.

$$\boxed{M18 \cdot SE}$$

This term is used to include the decision operation. It indicates that a pulse at T28 in Line 18 (M18) will set dZE, if SE is set. In other words, the integrator has a dZ output if the decision pulse (T28) in Line 18 and the necessary signals to set SE are present. See Timing Diagram, No. 10.

If T28 in M18 is high, then the output of R + YNdx addition is ignored. The non-zero existence of YNdx is detected and at T28 the detector (a flip-flop) is inspected to determine the existence of dZ (dZ = sign YN dx) output for a decision or a servo operation. The output has the sign of YNdx.

$$\boxed{dZEr = GO \cdot T28 \cdot \overline{M18}\ (\overline{YNdX} \cdot \overline{RC} + YNdX \cdot RC) + GO \cdot T28 \cdot \overline{SE} \cdot M18}$$

The dZE flip-flop has been set to record the existence of dZ pulse.

It will remain set unless the above signals reset it. The justification of the above signals is given in the following discussion. Refer to Timing Diagram, No. 11 for timing signals. It is to be noted specially that of the two terms, one is qualified by $\overline{M18}$ and the other by M18.

$$\boxed{GO \cdot T28 \cdot \overline{M18}}$$

This is the controlling signal for the first two terms to differentiate between the normal operation and the decision operation.

$$\boxed{\overline{YNdX} \cdot \overline{RC}}$$

It indicates that although YNdX was positive, it did not cause R to overflow.

$$\boxed{YNdX \cdot RC}$$

This indicates that YNdX was negative and did not cause R to overflow.

$$\boxed{GO \cdot T28 \cdot \overline{SE} \cdot M18}$$

This term involves the decision operation. If there is a decision pulse in Line 18 (M18) and SE has not been set, the dZE flip-flop will be reset which shows that:

(a) No dX has been addressed, or
(b) the ST flip-flop has not been set, or
(c) YN is zero.

3.4.5 KdZ GATE

This is just an intermediate stage between r + KdZ adder and dZ gate. This is for the scaling of the output of the dZ gate with a constant "K". K is stored in Line 15 (M15). The multiplication or scaling is accomplished with two "and" gates.

$$\boxed{KdZ = dZE \ (\overline{dZS} \cdot M15 + dZS \cdot \overline{M15})}$$

The above equation takes care of both the possible forms of the (positive and negative) output. If the unscaled output of the integrator is present and it is to be scaled, the contents of Line 15 (K)

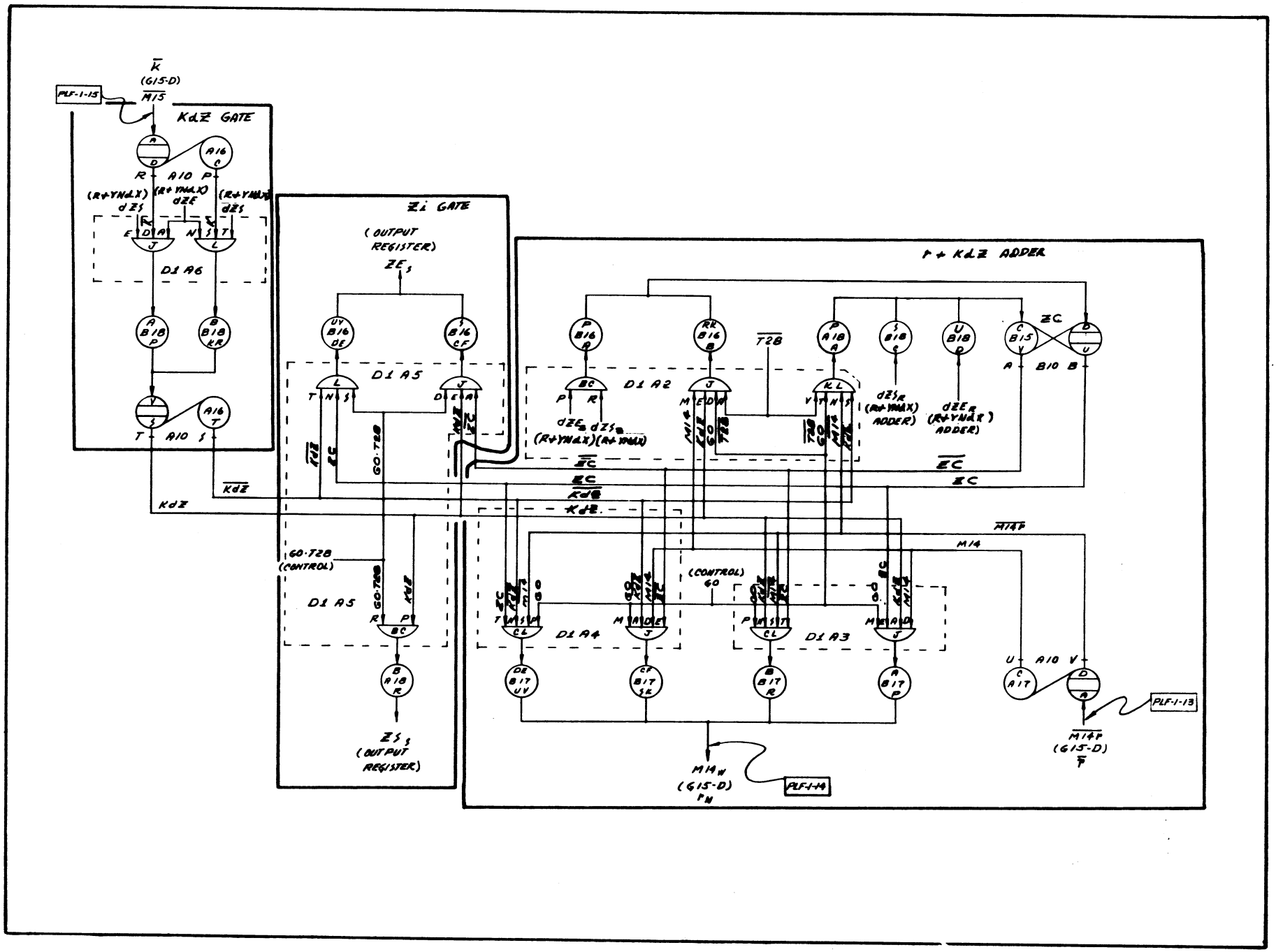



Figure 27. Schematic, r + Kdz Adder

will be the qualifying term for the "and" gates and the output is allowed to be passed to the r + KdZ adder. When the dZ is positive ($dZE \cdot \overline{dZS}$), M15 (K) is gated, but when dZ is negative ($dZE \cdot dZS$), $\overline{M15}$ (ONE'S complement of K) is gated. This is modified to TWO'S complement in r + KdZ adder.

3.4.6 r + KdZ ADDER (Reference Figure 27)

The remainder "r" is stored in Line 14 and KdZ is the multiplied output of the KdZ gate. After the carry of the adder has been recorded, the new remainder (rn) is written back on Line 14 (M14W). In the following discussion r is replaced by M14r.

3.4.6.1

$$\boxed{rn = M14W = GO\ (M14r \cdot KdZ \cdot ZC + \overline{M14r} \cdot KdZ \cdot \overline{ZC} + M14r \cdot \overline{Kdz} \cdot \overline{ZC} + \overline{M14r} \cdot \overline{KdZ} \cdot ZC)}$$

(See Timing Diagram, No. 12). This equation is just like equation (RN) of R + YNdX adder in paragraph 3.4.3.1. The bit by bit addition of the signals of only one "and" gate is shown in Timing Diagram, No. 12. The value of "rn" is again recorded in Line 14 (M14W) and the carry goes to the Zi gate through ZC flip-flop.

3.4.6.2 ZC (CARRY) FLIP-FLOP (See Timing Diagram, No. 13)

$$\boxed{ZC_s = dZE_s \cdot dZS_s + GO \cdot \overline{T28} \cdot KdZ \cdot M14r}$$

The carry (ZC) flip-flop gets set when any one of the above two "and" gates is qualified. Timing Diagram, No. 13 shows how the ZC (carry) flip-flop gets set. The function of both "and" gates is given below.

$$\boxed{dZE_s \cdot dZS_s}$$

This term provides the necessary adjustment to convert ONE'S complement out of KdZ gate to TWO'S complement. It indicates that the unmultiplied output was negative.

$$\boxed{GO \cdot \overline{T28} \cdot KdZ \cdot M14r}$$

This sets the carry flip-flop (Zc) during the writing of KdZ into Line 14 (M14W) when both the addend and the augend are one's.

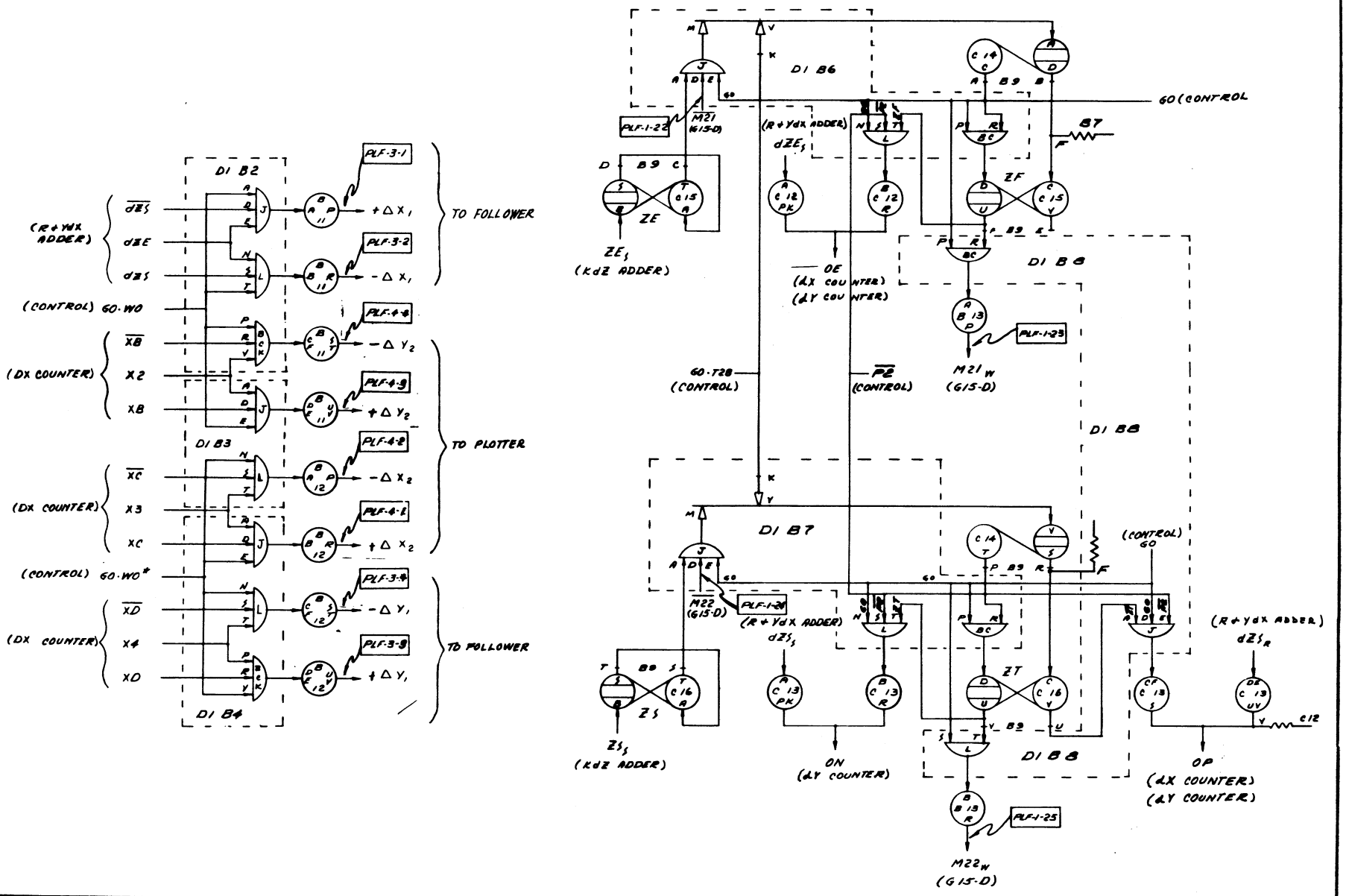



Figure 28. Schematic, Output Register

$$\boxed{ZCr = dZEr + dZSr + GO \cdot \overline{T28} \cdot \overline{KdZ} \cdot \overline{M14r}}$$

Note that dZEr causes ZC flip-flop to get reset when there is no unmultiplied output.

$$\boxed{dZSr}$$

This also causes ZC flip-flop to be reset when the unmultiplied output is positive.

$$\boxed{GO \cdot \overline{T28} \cdot \overline{Kdz} \cdot \overline{M14r}}$$

This resets ZC when both addend and augend are zero. The timing diagram can easily be drawn by referring to Timing Diagram, No. 13.

3.4.7 Zi GATE (Refer to Figure 27 and 28).

The discussion of the overflows associated with Zi gate is almost the repetition of the discussion of overflows associated with dZ gate already stated in paragraph 3.4.4, so it will not be repeated here.

3.4.8 OUTPUT REGISTER (Refer to Figure 28).

3.4.8.1 ZS FLIP-FLOP

$$\boxed{ZS_s = GO \cdot T28 \cdot KdZ}$$

ZS is the sign flip-flop for the output Z. If it is set, Z is negative. T28 in the setting signals given in the equation is to show that KdZ has to be negative to set ZS. Refer to Timing Diagram, No. 14 for easy check.

$$\boxed{ZSr = ZS}$$

This shows that ZS will never stay in the set condition for more than one bit time because its own true output (ZS) serves as its reset term (ZSr). It is also shown in Timing Diagram, No. 14.

3.4.8.2 ZE FLIP-FLOP (Refer to Figure 27).

$$\boxed{ZEs = GO \cdot T28 \cdot (\overline{KdZ} \cdot ZC + KdZ \cdot \overline{ZC})}$$

The existence of Z (ZE) is possible under both conditions (positive and negative). The signals given in the above equations specify both conditions.

$$\boxed{GO \cdot T28}$$

This is the qualifying term for both "and" gates.

$$\boxed{\overline{KdZ} \cdot ZC}$$

If the KdZ is positive and causes a carry in the r + KdZ adder at T28, Z exists and it is positive.

$$\boxed{Kdz \cdot \overline{ZC}}$$

If a negative KdZ does not cause a carry at T28, Z exists but it is negative.

$$\boxed{ZEr = ZE}$$

ZE will never be in the set condition for more than one bit time because its own true output serves as a reset term unconditionally. The ZEs and ZEr signals are shown in Timing Diagram, No. 15.

3.4.8.3 ZT FLIP-FLOP (Refer to Figure 23 and 28).

ZT is the true output of ZT flip-flop which is associated with ZS flip-flop to record and precess the sign of the output. ZT is recorded in Line 22 (M-22, 4-word line) for precession and recirculation. Also, it generates ON signal (output negative) which goes to the dy counters. $\overline{ZT}$ shows that output is positive and it generates (OP) signal which goes to the dX and dy counters. The control of ZT flip-flop is as follows:

$$\boxed{ZT_s = GO \cdot \overline{T28}\ (M22 + ZS) = GO \cdot \overline{T28} \cdot M22 + GO \cdot \overline{T28} \cdot ZS}$$

$$\boxed{GO \cdot \overline{T28} \cdot ZS}$$

GO•T28 controls the setting of ZS flip-flop, but GO $\overline{T28}$ controls the setting of ZT flip-flop. It is clear that ZS output will be

high for T29. Therefore, ZT output will be high for T1 of the following word, i.e. (I + 2).

$$\boxed{GO \cdot \overline{T28} \cdot M22}$$

M22 is used for recirculation and one bit precession of the sign bit. The Timing Diagram, No. 18 shows the ZTs signals.

ZT·GO qualify the "and" gate for the sign bit (ZT·GO = M22W) to be written in M22 for recirculation and one bit precession.

$$\boxed{ZTr = GO\ (T28 + \overline{M22 \cdot \overline{ZS}})}$$

GO·T28: - This term resets ZT to avoid T28 (sign bit) being recorded in line 22, and helps to generate OP (output positive) signal. GO·T28 erases the previous ON signals after they have been precessed from T1 through T28.

$$\boxed{GO \cdot \overline{M22 \cdot \overline{ZS}}}$$

This is to show that if the output sign does not exist, it is positive. See Timing Diagram, No. 16.

3.4.8.3.1 ON SIGNAL (OUTPUT NEGATIVE)

$$\boxed{\begin{aligned} ON &= dZSs + ZT \cdot GO \cdot \overline{P2} \\ &= dZSs + ZT \cdot GO\ (\overline{T28 + T29}) \end{aligned}}$$

dZSs: ON signal is present when the dZSs is taken directly to the output register and from there to the dy counter. It shows that sign of dZ exists, i.e., dZ is negative.

$$\boxed{ZT \cdot GO \cdot \overline{P2}}$$

ZT is the true output of ZT flip-flop which is high during T1 of next word time. To make sure that ON is not high because of ZT at T28 or T29, $\overline{P2}$ term is included. See Timing Diagram, No. 16.

3.4.8.3.2 OP SIGNAL (OUTPUT POSITIVE)

$$\boxed{OP = dZSr + \overline{ZT} \cdot GO \cdot \overline{P2}}$$

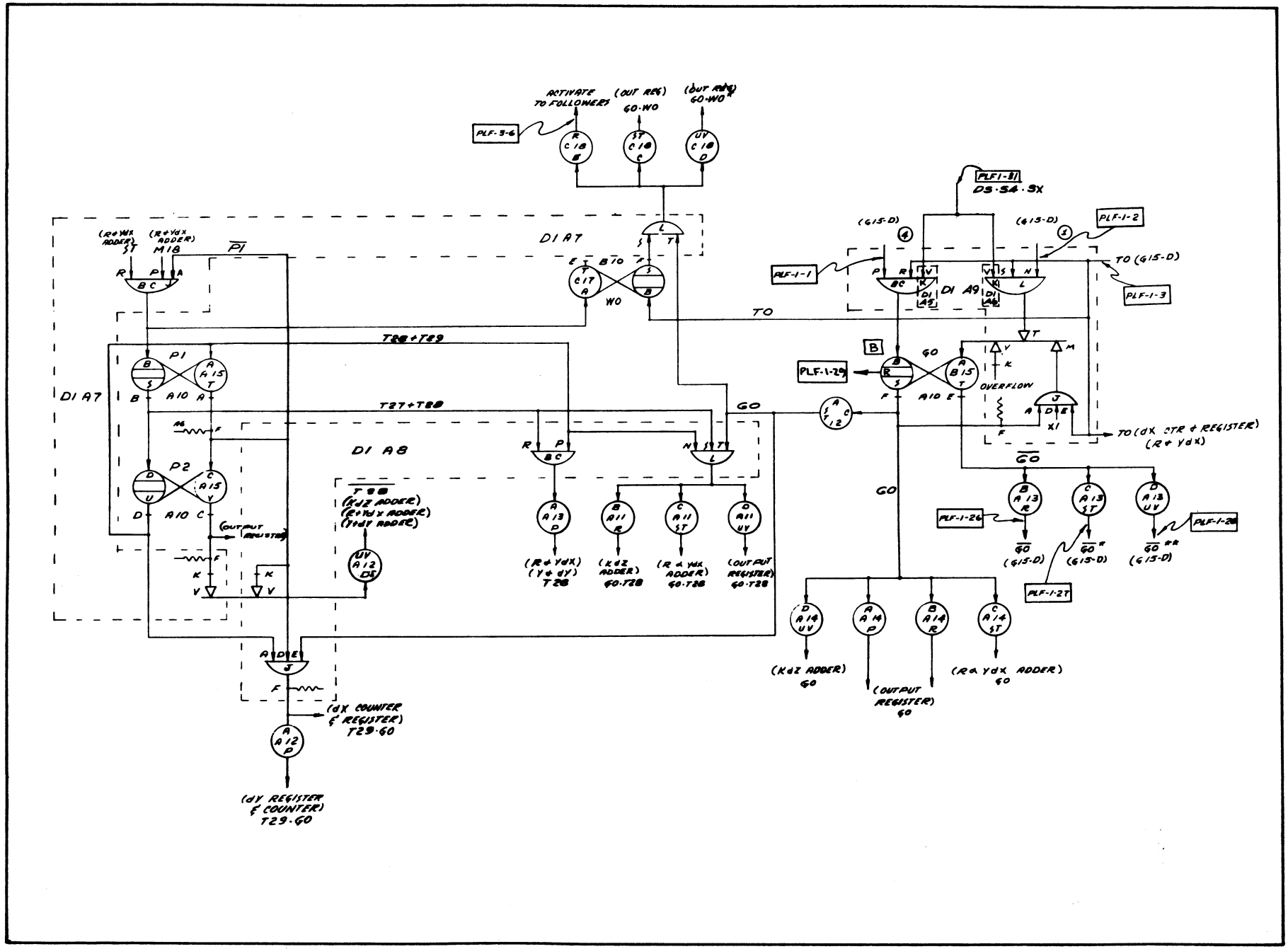



Figure 29. Schematic, Control

OP signal is the same as the absence of the ON signal. Therefore, the control signals for the "and" gates are the same except that the qualifying term is $\overline{ZT}$. See Timing Diagram, No. 16.

3.4.8.4 ZF FLIP-FLOP

ZF flip-flop is associated with ZE flip-flop which shows that the output exists. The output ZF is recorded in Line 21 (M21W). ZF is the controlling flip-flop for the output. The setting and resetting terms are almost the same as that of ZT flip-flop except for ZE and M21 instead of ZS and M22. The true output forms an OE (output exists) signal which goes to dX and dy counters and completes the cycle of flow of information in DA-1.

$$ZFs = GO \cdot \overline{T28}\ (M21 + ZE)$$
$$ZFr = GO\ (T28 + \overline{M21} \cdot \overline{ZE})$$
$$M21W = GO \cdot ZF\ ; \quad \overline{P2} = \overline{T28 + T29}$$
$$OE = dZE_s + ZF \cdot GO \cdot \overline{P2}$$

The calculation of this equation is shown in Paragraph 3.5.12.2. See Timing Diagram, No. 17 for the details of the OE signal.

3.4.9 CONTROL (See Figure 29.)

3.4.9.1 M-18

Line 18 (M18) contains most of the information which controls the operation of DA-1. T26 is always the clear pulse. Start pulse is also present. TO is used to synchronize the operation of DA-1 with G-15. Timing Diagram, No. 18 shows all the control signals and their functions. One word structure is shown in Timing Diagram, No. 19.

3.4.9.2 ST FLIP-FLOP (See Figure 24 and 29.)

ST flip-flop and M18 controls the start of DA-1.

$$\boxed{ST_s = \overline{ST} \cdot M18}$$

If ST flip-flop is reset, the first pulse encountered in Line 18 (M18) will set it.

$$\boxed{ST_r = T28 + TO}$$

ST will be reset unconditionally at T28 of every word and also at TO. TO is T29 of word 107 and $\overline{CN}$. The TO assures that random conditions which occur at the turn on and warm up cycle are corrected and that the DA-1 is synchronized with the G-15.

3.4.9.3 T28 = P1•P2

T28 is generated with two flip-flops, P1 and P2. P1 is set high for two bits time (T27 and T28); P2 is set for T28 and T29. The combined output generates T28 as is shown in Timing Diagram, No. 16.

3.4.9.4 DECISION OPERATION

When there is a discontinuity or an abrupt change in a function, normal integration does not help. This discontinuity in the function, is taken care of by the decision pulse (SE). The integrator is converted to a decision element by placing a one bit in the T28 position of Line 18 (M18).

3.4.9.4.1 SE FLIP-FLOP (See Figure 22.)

$$\boxed{SE_s = XE \cdot ST \cdot YN}$$

If the integrator has received a dX pulse and the ST flip-flop has been set, the presence of any bit in YN will set SE. The Timing Diagram, No. 19 shows this.

$$\boxed{SE_r = \overline{ST}}$$

SE is reset when ST is reset. (i.e. SE is reset at T1 through the following steps):

ST receives a reset pulse at T28;
ST resets at T29;
$\overline{ST}$ (high at T29) serves as the reset term for SE;
SE resets at T1 of the next word.

## 3.5 INTERNAL LOGIC OR THEORY OF OPERATION

The flow of information in general has been discussed in the previous sections without paying too much attention to the hardware and circuitry of the machine. The discussion in this section is mainly concentrated about the circuitry of the various adders, registers and counters, etc. The circuitry involved in the generation of various control pulses is also discussed. The main purpose of this section is to show the operation of various parts of the circuitry of DA-1.

### 3.5.1 SCHEMATIC CONTROL (Refer to Figure 29. and Timing Diagram, No. 18)

All the signals associated with the control of DA-1 are discussed as follows.

#### 3.5.1.1 START DA-1

With the command 01931 (27z) in G-15, DS·S4·SX signal is generated in G-15 along with ④ and TO (T29 of word u7). ④ is the characteristic part of the command. (Reference Figure 30.)

$$④ = \overline{CX}\cdot\overline{CW}$$

$$TO = T29 \text{ (word u7)}\cdot\overline{CN}$$

For DS·S4·SX signals, refer to schematic of control switch on Figure 31.

$$GO_S = TO\cdot(④\cdot DS\cdot S4\cdot SX)$$

The output of this three term "AND" gate sets the GO flip-flop and

turns the DA-1 ON, and lights the GO neon on the front panel of G-15. GO signal is used in the Kdz adder, output register, R and Ydx adder, and also to reset GO flip-flop. It is pointed out that although DA-1 is ON, the computation will not start unless there is a control signal (start pulse) in M18 to set the ST flip-flop (See Figure 26). ST is set by M18·$\overline{ST}$ and it is reset by TO or T28 (control signals).

3.5.1.2 HALT DA-1

DA-1 can be stopped by any one of the three signals discussed below.

The command 11931 (67x) unconditionally halts the DA-1. To get an output of the "AND" gate which resets the GO flip-flop, this command has to be operative during TO (T29 of word u7).

$$GO_r = TO\cdot(\textcircled{1}\cdot DS\cdot S4\cdot SX) + GO\cdot X1\cdot TO + OVERFLOW$$

$$\textcircled{1} = \overline{CX}\cdot CW \qquad \text{(G-15 signal)}$$

An overflow in any integrand immediately halts DA-1 computation and sets the G-15 overflow flip-flop. Also, if dx counter #1 contains a non-zero dx at TO time, DA-1 computation will halt.

The false output of the GO flip-flop ($\overline{GO}$) is used back in G-15 for blocking the recirculation of M21, M22, M16, M17 and M14.

3.5.1.3 T28 AND T29 (See Figure 29 and Timing Diagram, No. 18)

T28 and T29 are generated in DA-1 and are not used from G-15. The first pulse (start pulse) in Line 18 sets the ST flip-flop and T26 in Line 18 qualifies the three term "AND" gate, the output of which sets the P1 flip-flop. T26 of Line 18 will set P1 and it will remain set until two bit times later. The output of P2 resets it. P1 will be set for T27 and T28 bit times and P2 will be set for T28 and T29 bit times. The output of the "AND" gate formed by the outputs of P1 and P2 is T28. Signal GO·T28 formed by another "AND" gate is used in the Kdz adder, R and Ydx adder, and in the output register.

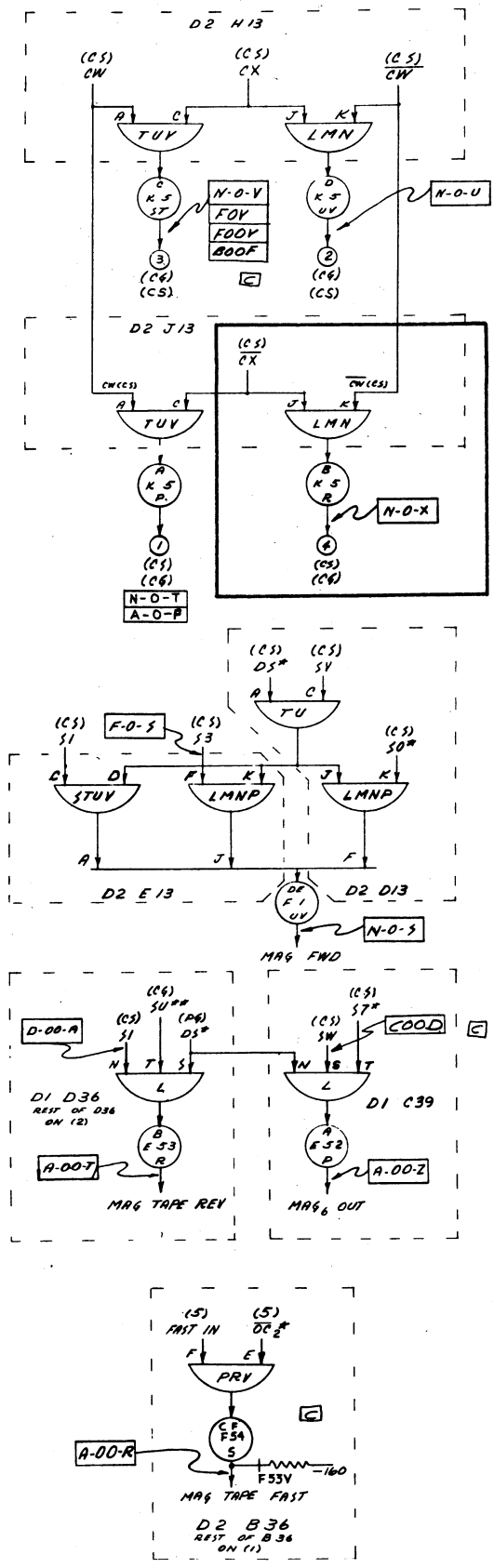



Figure 30. Schematic, Magnetic Tape Control, G-15, 3D597 (Blocked Area)

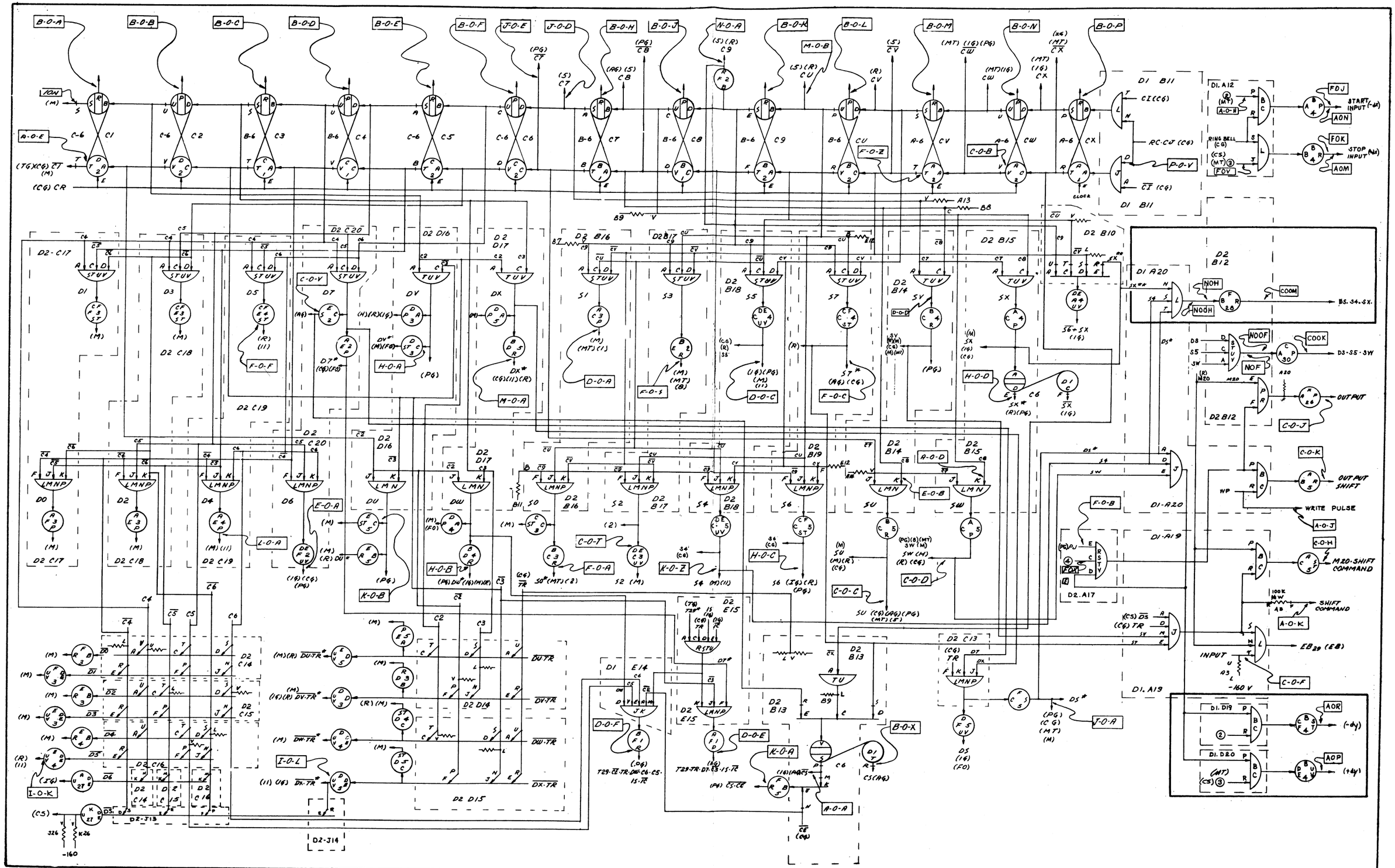



Figure 31. Schematic, Control Switch, G-15, 3D594
(Blocked Area)

$T28 = P1 \cdot P2$

$T28 \cdot GO = P1 \cdot P2 \cdot GO$

$\overline{T28} = \overline{P1} + \overline{P2}$

$GO \cdot T29 = P2 \cdot \overline{P1} \cdot GO$

$\overline{T28}$ is used in the adders.

$GO \cdot T29$ are used in dx, dy counters, and registers.

WO (Word "OO") flip-flop is set by TO and the signal $WO \cdot GO$ is used to activate the follower and plotter and goes to the output register. It is reset by the same signal which sets P1, i.e. ($P1_s = ST \cdot M18 \cdot \overline{P1}$).

Thus, the three important control signals of Line 18 (TO, T26 and start pulse) are used to generate other control signals which control the operation of DA-1.

3.5.2 SECONDARY INPUT (dy) (Reference Figure 32 and Figure 33.)

dy is assumed to be the dependent variable in almost all the equations which are the index of most present day problems. Being a dependent variable, dy is called Secondary input to the integrator.

3.5.2.1 dy COUNTERS

There are four dy counters and each of them is connected to a long line in the G-15 through four toggle switches, located inside the back panel of DA-1. (See Figure 21.)

Long line 10 is connected to dy counter #4, M11 to #3, M12 to #2 and M13 to #1. Each counter consists of four flip-flops. The information bits are called the coincidence pulses because they have to be coincidence with the output pulses to load the counter. Coincidence detection is employed such that if M10 has a "one" bit in any bit position except T28 and T29, the Zi output from M21, M22 in the same bit position is algebrically added to the previous contents of the counter.

If T28 bit is present in word I (M10), the dz output of integrator I will be counted into counter #4 instead of the T28 contents of Z lines.

The presence of T29 bits in line 10 indicates to counter #4 when to

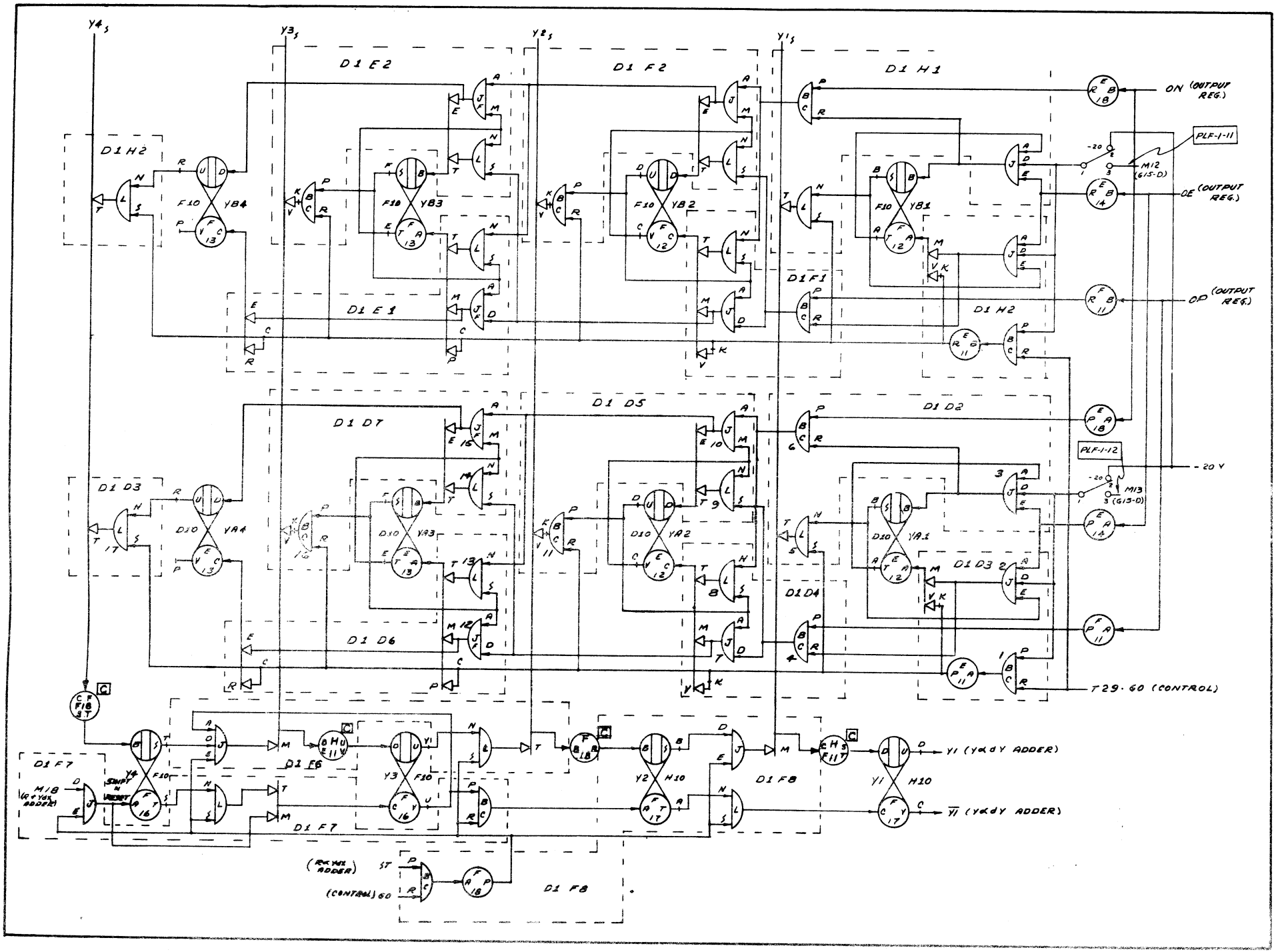



Figure 32. Schematic, dy Counters, No. 1, 2 & dy Register

load its accumulated ΔY count into the dy register. If counter #4 is to be used for integrator I, a T29 bit must be programmed in word I-1 of M10. The occurrence of any T29 in one of the address lines dumps the contents of the associated dy counter into the dy register and clears the counter. That is why T29 of these address lines is called the dump pulse.

The location of (T29) dump pulse in one of the four word (Mod 4) for different dy counters is shown in the form of a Table in paragraph 3.2.4.

Only dy counter #4 associated with M10 is described in the following discussion. The function of other three counters is exactly the same except that their information is stored in different long lines and their dump pulses are located in different word times.

3.5.2.1.1 dy COUNTER #4 (Reference Figure 33.)

There are four possibilities for the input to the dy counter from the output register. The output register is discussed in paragraph 3.4.8.

(a) OE·ON (output exists and it is negative).
(b) OE·OP (output exists and it is positive).
(c) $\overline{OE}$·ON (output does not exist and the sign is negative).
(d) $\overline{OE}$·OP (output does not exist and sign is positive).

The first two possibilities are normal, but the last two are not the normal cases. Each possibility is discussed below.

3.5.2.1.1.1 OUTPUT NEGATIVE ⟷ (ON)

The OE·ON signals are the simultaneous outputs of the output register. dy's are stored in address Line M10. The address lines are connected through toggle switches located inside the back panel of DA-1. Dump pulses for M10 (T29) are located in word time 2, module 4.

It is assumed that all the four flip-flops of this counter are in the reset position because of the last dump pulse. They are reset by M10

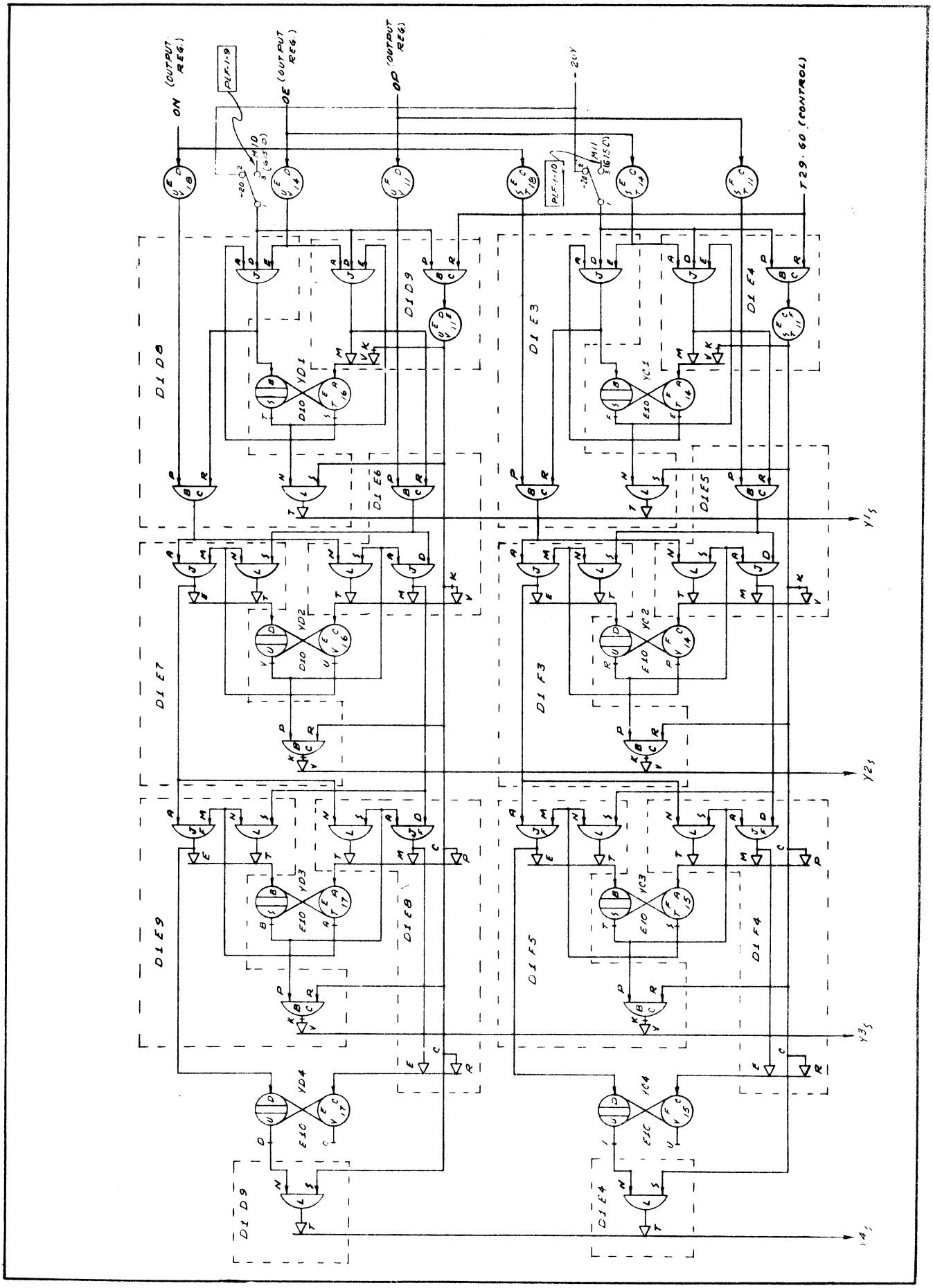



Figure 33. Schematic, dy Counters, No. 3, 4 & dy Register

and (T29•GO) control signal.

YD1 flip-flop is reset and OE and dy (M1O) are present. The output of the "AND" gate sets the YD1 flip-flop.

$$\boxed{YD1_s = \overline{YD1} \cdot OE \cdot M1O}$$

When YD1 is set, the other "AND" gate is qualified and the output will reset YD1.

$$\boxed{YD1_r = \overline{YD1} \cdot M1O \cdot OE + M1O \cdot GO \cdot T29}$$

If M1O has information in all of its bit times and OE exists at each bit time, YD1 will never remain set or reset for more than one bit time. But, from the practical point of view, the simultaneous presence of OE and M1O will set YD1 if it is already reset and will reset it if it is already set. YD1 will also be reset by M1O•GO•T29. The true output of YD1 and T29•GO and M1O will qualify an "AND" gate, the output of which will set Y1 flip-flop of the dy register. See Figure 32.

$YD1_s$ and ON (output negative) will set YD2 if it is reset (assumed for this discussion).

$$\boxed{YD2_s = YD1_s \cdot \overline{YD2} \cdot ON \quad + \quad \text{other terms}}$$

Similarly, output of YD2 and T29•GO will set Y2 of dy register. Similarly, YD3 will be set by $YD2_s$ and $\overline{YD3}$. $YD3_s$ will also set YD4. There is no delay in setting YD4. YD4 is the sign flip-flop for the dy counter #4.

(T29•GO•M1O) resets all four flip-flops of the dy counter #4. The output of YD4 will set Y4 of dy register.

All four flip-flops are set or reset simultaneously (depending upon their preceding conditions) by the first input signal. The dump pulse (T29 of M1O) and (T29•GO) control signal will reset all of them and dump their information into the dy register.

The preceding discussion holds true for the other three counters too. The only difference is that dy counters #3, #2 and #1 are connected to M11, M12 and M13 respectively, and their dump pulses are located in different word times (as shown in table of paragraph 3.2.4).

3.5.2.1.1.2 COMPLEMENTATION

Negative numbers appear in the complemented form in DA-1. The complementation is performed in the counters. Suppose YD4 and YD1 are set and YD2 and YD3 are in reset condition. The configuration shows 1 0 0 1 = -1 which is the complemented form of -7. If, at this stage another OE·ON comes in, it will reset YD1 and will set YD4 which will show the configuration of 1 0 0 0 = -0.

Now, if the T29 dump pulse comes in and dumps this information into the dy register, it will set Y4, but Y3, Y2 and Y1 will remain reset. Now, if the ST pulse is in T23, it will start shifting the information at T24 from Y3, Y2 and Y1, and the contents of Y3 will be out of Y1 at T27 bit time. Y1 will be high at T27 and T28. The T26 (M18) clear pulse resets Y4 and Y3. Then, the contents of Y4 and Y3 are shifted along by the shift pulse alone. Thus, the contents of Y register coming out of Y1 will be 1 1 0 0 0 = -8, which is the same as -0.

The above is a special example for one of the sixteen possible configurations of the counter flip-flops. ON and OP signals are not gated into YA1, YB1, YC1 or YD1 because these flip-flops change state with every OE signal regardless of sign.

3.5.2.1.1.3 OUTPUT POSITIVE↔(OP)

OE·OP (output exists and is positive). When the output is not negative (ON is absent) it is positive (OP is present). The operation of OE signal is the same as in case of (ON) output negative.OP and $YD1_r$ qualify the "AND" gate which helps in set-

ting or resetting YD2 depending upon the condition that YD2 is in.

$$\boxed{\begin{aligned} YD2_r &= YD2 \cdot OP \cdot YD1_r \\ YD2_s &= \overline{YD2} \cdot OP \cdot YD1_r \end{aligned}}$$

Similarly, YD3 is reset or set by $YD2_r$ signal depending upon the condition of YD3.

$$\boxed{\begin{aligned} YD3_r &= YD3 \cdot YD2_r \\ YD3_s &= \overline{YD3} \cdot YD2_r \end{aligned}}$$

$YD3_r$ signal also resets YD4 (sign flip-flop). The absence of the true output of YD4 flip-flop indicates that the contents of the dy register are positive, while the presence of the signal indicates that it is negative.

All the four flip-flops of the counter are cleared and four flip-flops of the dy register are loaded simultaneously at T29.

3.5.2.1.1.4 OUTPUT DOES NOT EXIST ($\overline{OE}$)

$\left.\begin{matrix} \overline{OE} \cdot ON \\ \overline{OE} \cdot OP \end{matrix}\right\}$ = When the output does not exist ($\overline{OE}$), there will not be any action in the counter because the first two "AND" gates are not qualified which start the setting and resetting of the counter flip-flop.

3.5.2.2 dy REGISTER (Reference Figure 32.)

The four flip-flops (y1, Y2, Y3 and Y4) of the dy register are loaded from the sixteen flip-flops of the dy counter at their respective dump pulse time (T29). This is because each dy address line has a T29 pulse present after every 4 word time interval. Each counter dumps its information into the register at T29 bit time after every fourth word time. Consequently, dy register is cleared four times before the cycle is repeated again.

3.5.2.2.1 Information coming from Y (A1 + B1 + C1 + D1) sets Y1; from Y (A2 + B2 + C2 + D2) sets Y2; from Y (A3 + B3 + C3 + D3) sets Y3; and from Y (A4 = B4 + C4 + D4) sets Y4. There are sixteen possible configurations that these four flip-flops can attain including that of $\pm$ 0.

The following table shows the association of each flip-flop with its binary count.

| Flip-flop = | Y4 | Y3 | Y2 | Y1 |
|---|---|---|---|---|
| | $\pm$ (sign) | $2^2$ | $2^1$ | $2^0$ |

For the sake of description of the dy register, it is assumed that all four flip-flops are set which shows that information stored in the register is -7.

| - | 4 | 2 | 1 | decimal count |
|---|---|---|---|---|
| - | $2^2$ | $2^1$ | $2^0$ | |
| 1 | 1 | 1 | 1 | binary count |

Y4 is sign flip-flop and it also perpetuates 1's needed for complementation. $Y4_r$ = M18. (ST·GO) clear pulse.

The output of the "AND" gate (ST·GO) is called the shift and reset signal, and the dy register is called the dy shift input register.

$Y4_r$ resets Y4 and Y3. T26 (clear pulse; M18) resets both Y4 and Y3. The false output of Y3 is used to reset Y2 and also to qualify an "AND" gate which in turn sets Y3.

$$Y3_s = (YA3 + YB3 + YC3 + YD3)\ T29 \cdot GO + \overline{Y3} \cdot Y4 \cdot GO \cdot ST$$

Where GO·ST is shift input signal.

All these terms indicate that the setting terms are coming from the counters, but the second term shows that Y3 cannot be set every time when there is Y4 output.

3.5.2.2.2 Y3 is also reset by $\overline{Y4}$ and shift input (ST·GO) signal:

$$Y3_r = ST \cdot GO \cdot (M18 + \overline{Y4})$$

3.5.2.2.3 Y2 and Y1 are set and reset by the shift input signal (GO·ST) and the false output of the preceding flip-flop.

$$Y2_s = GO \cdot T29\ (YA2 + YB2 + YC2 + YD2) + Y3 \cdot GO \cdot ST$$
$$Y2_r = \overline{Y3} \cdot GO \cdot ST$$
$$Y1_s = GO \cdot T29\ (YA1 + YB1 + YC1 + YD1) + Y2 \cdot GO \cdot ST$$
$$Y1_r = \overline{Y2} \cdot GO \cdot ST$$

The output of Y1 is ΔY (accumulated dy) and is taken to the Y and dy adder. Y1·ST is called dy in the adder.

$$\frac{dy = y1 \cdot ST}{\overline{dy} = \overline{y1 \cdot ST} = \overline{y1} + \overline{ST}}$$

3.5.3 PRIMARY INPUT (dx) (Reference Figure 34.)

The dx inputs are identical to that of dy inputs. Lines 6, 7, 8 and 9 are associated with dx counters #4, #3, #2 and #1, respectively. Each dx counter holds only one count (existence and sign), and the circuit operation is such that if more than one dx inputs are addressed, each term (existence and sign) will be set as though the multiple counts were presented simultaneously at the "OR" gates on the input terms. The number of integrators, counters, and their respective dump pulses are shown in the table in paragraph 3.2.4. The dx pulse is like a control signal which initiates the addition of YN to R. Because of its independence, it is called the primary input.

3.5.3.1 dx COUNTERS

The dx counter consists of eight flip-flops. The four address lines (M6, M7, M8, M9) are connected to these flip-flops through toggle switches. Each line is associated with two flip-flops. The following table will show the combinations of these flip-flops.

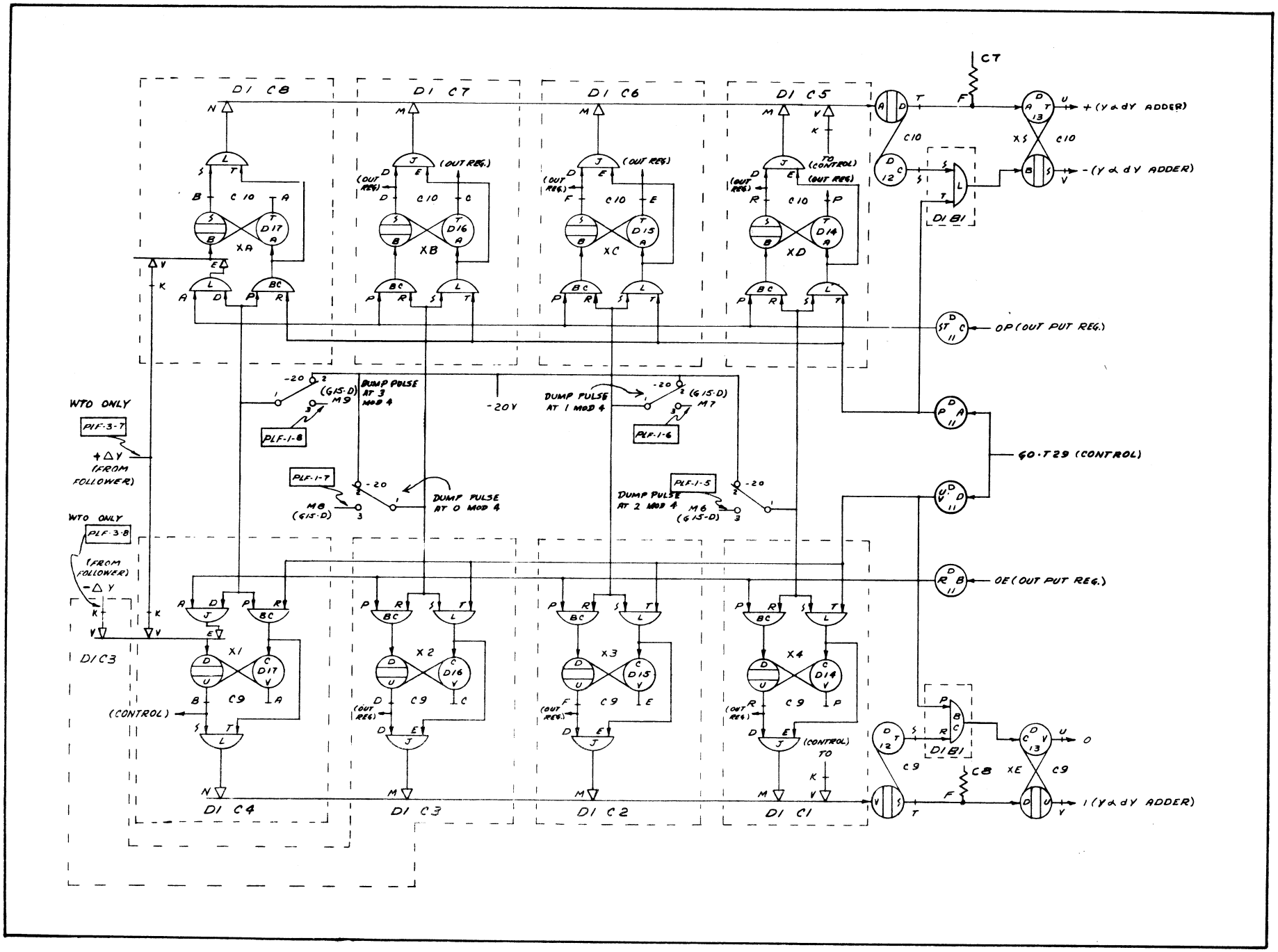



Figure 34. Schematic, dx Counter & Register

| ADDRESS LINE | EXISTENCE FLIP-FLOP | SIGN FLIP-FLOP |
|---|---|---|
| M9 | X1 | XA |
| M8 | X2 | XB |
| M7 | X3 | XC |
| M6 | X4 | XD |

3.5.3.1.1 The four flip-flops associated with the existence of dx pulse (X1, X2, X3 and X4) are set by the OE pulses from the output register and the coincidence pulses in the address lines. They are reset by the control pulse (GO•T29) and the dump pulse. The same resetting pulse is also used to qualify another "AND" gate along with the true output of these flip-flops. The outputs of all these four "AND" gates are used to set the Buffer Inverter (B.I.), the true output of which sets XE flip-flop of the dx register.

$$X2s = OE \cdot M8$$
$$X2r = GO \cdot T29 \cdot M8$$

3.5.3.1.2 Similarly, X3 and X4 are set and reset with the exception of X1.

$$X1s = OE \cdot M9 + (+\Delta Y) + (-\Delta Y)$$

$+\Delta Y$ and $-\Delta Y$ are the outputs of the graph follower. When the follower is used, dx signals to the follower are provided by the unmultiplied output of integrator u7. The $\Delta Y$ signal from the follower sets dx counter #1 during word 00. If some other input has already arrived at dx counter #1 (or arrives later, but before the next counter dump signal to the counter), each of the two terms (existence and sign) will be set in "OR" gate fashion. The output of X1 is used in the schematic control circuitry to reset the GO flip-flop. The true output of X2, X3 and X4 is used in the output register as an input to the graph plotter and follower. The dump pulse location of each line is given in the table in paragraph 3.2.4.

3.5.3.1.3 The other four (XA, XB, XC and XD) are the sign flip-flops. Each of them is set by OP (output positive signal from the output register) and the coincidence pulse from the respective address line. The

reset signals are the same as for the existence flip-flops:

$$XBs = OP \cdot M8$$
$$XBr = T29 \cdot GO \cdot M8$$

but, XA is also set by + $\Delta Y$ from the follower.

$$XAs = OP \cdot M9 + (+\Delta Y)$$

The true and false outputs of XB, XC and XD are used as inputs to the follower and plotter. (See Figure 28.)

The "AND" gates formed by the true outputs of these flip-flops and their resetting signals are "OR" gated into the dx register through a (B.I.) buffer inverter.

3.5.3.2 dx REGISTER

The dx register consists of two flip-flops and two buffer inverters. XS is for the sign of dx and XE for the existence of dx pulse. During each word time the dump pulse in one of the four address lines resets one of the four existence flip-flops (X1, X2, X3, X4) and one of the corresponding sign flip-flops (XA, XB, XC, XD).

The output from each of the four existence counters sets the XE flip-flop through the true side of the B.I. "TO" is also used to set XE which provides a fixed machine time input to Integrator "00". The control signal (T29•GO) and the false output of the buffer inverter reset XE. Thus, the true or false output of XE (XE or $\overline{XE}$; 1 or 0 respectively) goes to the Y and dy adder, but only XE is used.

In case of the sign flip-flop (XS), the four counter flip-flops (XA, XB, XC and XD) are similarly cleared during T29. Either the outputs of the four "AND" gates or TO will reset the XS flip-flop through the true side of the buffer inverter. TO makes the machine time input a positive dx. XS is set by the false output of the buffer inverter and GO•T29 (control signal). The ($\overline{XS}$) false output is positive and (XS) true output is negative. Each output of XE has a sign associated with it. Therefore, the simultaneous output of XS and XE will give + or

-dx ($\pm$1). The output of both (XE and XS) flip-flops ($\pm$ dx) is used in the Y and dy adder which is discussed in paragraph 3.5.4.

3.5.4 Y and dy ADDER (Reference Figure 22, and Paragraph 3.4.1).

Refer to Timing Diagrams, No. 1, 2, 3 and 4.
Y is stored in Line 17 and dy is the output of the "AND" gate formed by yl (true output of Y1 flip-flop of the dy register) and ST (true output of ST flip-flop).

$$\boxed{dy = Y1 \cdot ST}$$

The third term involved in this logical accumulation is YC which is handled by YC flip-flop. The output of the adder (YN and $\overline{YN}$) is added to R (M16) with the occurrence of dx pulse. There are four "AND" gates which take care of all the possible combinations of these three terms to form YN, while the other four form $\overline{YN}$. The equations showing various combinations of these terms are given in paragraph 3.4.1. The same equations are given below for ready reference. The signals are shown in Timing Diagram, No. 1.

$$\boxed{\begin{aligned} YN &= (Y \cdot dy \cdot YC) + (\overline{Y} \cdot dy \cdot \overline{YC}) + (Y \cdot \overline{dy} \cdot \overline{YC}) + \overline{Y} \cdot \overline{dy} \cdot YC) \\ \overline{YN} &= (Y \cdot \overline{dy} \cdot YC) + (\overline{Y} \cdot dy \cdot YC) + (Y \cdot dy \cdot \overline{YC}) + (\overline{Y} \cdot \overline{dy} \cdot \overline{YC}) \\ \overline{dy} &= \overline{Y1} \quad \text{(and, not Y1} \cdot \text{ST, as before)} \\ dy &= Y1 \cdot ST \end{aligned}}$$

Y1 and $\overline{Y1}$ are the outputs of Y1 flip-flop of the dy register. Y and $\overline{Y}$ is the information already recorded in M17.

3.5.4.1 YC FLIP-FLOP

YC is the true output of the carry flip-flop which takes care of the carry when logical accumulation of binary bits is going on. If YC is reset and there is a bit present in M17 (Y is present) and Y1 and ST are high, the output of "AND" gate ($\overline{YC} \cdot Y \cdot Y1 \cdot ST$) will set YC. The signals are shown in Timing Diagram, No. 2. YCs and YCr equations are given in paragraph 3.2.2. The output of three "AND" gates resets

YC. Whenever there is YN signal formed by YC·$\overline{Y}$·$\overline{Y1}$, it will reset YC only during $\overline{T28}$ time. The other two "AND" gates are qualified by T28 to keep the end carry suppressed, otherwise if there is a YN present during T29, the overflow "AND" gate will be qualified by GO·T29 and will halt the DA-1 computation. The overflow of Y register is shown in example D and E in Section 3.4.2, and it is shown how the two "AND" gates qualified by T28 prevent YC from being high at T29 unless there is to be a T29 bit present in the Y register.

The output of these eight "AND" gates (YN and $\overline{YN}$) is taken as input to YNdx gates. Also, YN is written back into M17 through the "AND" gate qualified by GO signal. YN with T29·GO will set the overflow flip-flop in the G-15.

3.5.5 YNdx GATE (Reference Figure 22 and Timing Diagram, No. 4)

YN and $\overline{YN}$ are used to form two "AND" gates which are qualified by ST, XE and the sign of dx. When dx is positive, the new value of Y (YN) is passed through and it goes to the R + YNdx adder through a buffer inverter. But, when dx is (-) negative, it is needed to complement the number YN. As has been discussed in paragraph 3.4.3 that $\overline{YN}$ is ONE'S complement of YN, so $\overline{YN}$ is used to qualify the second "AND" gate with XS and taken to R + YNdx adder through the buffer inverter. XE, XS and $\overline{XS}$ are the output signals of the dx register.

3.5.6 SE FLIP-FLOP (Reference Timing Diagram, No. 8 and No. 9)

SE is the servo flip-flop used for the decision operation. When dx is present with YN and ST, the SE flip-flop is set and stays set until reset by $\overline{ST}$. The output of SE is used to qualify two "AND" gates in R and YNdx adder (Figure 23) for the servo operation of the integrator. The timing of the signal is shown in Timing Diagram, No. 8 and 9.

3.5.7 R and YNdx ADDER (Reference Figure 26; Timing Diagrams, No. 5, No. 6, and No. 7; and Paragraph 3.4.4.)

The value of the remainder (R) is recorded in Line 16 (M16) and it is usually programmed +1/2 at the start of the problem and the contents of the R register are always considered positive. YNdx and $\overline{YNdx}$ are the outputs of the YNdx gates. The four "AND" gates are qualified by GO, RC and R. RC is the output of the RC flip-flop which handles the carry in the process of accumulation. The value of RN is written back into Line 16 (M16W); but the carry at T28 (the output of the adder) is further used in the flow of the information in DA-1. The equations and examples are given in paragraph 3.4.4. The function of the four "AND" gates is similar to those of the Y and dy adder.

3.5.7.1 RC FLIP-FLOP

Like YC flip-flop in the Y and dy adder, RC flip-flop performs the same function in R +YNdx adder. It handles the carry during binary summation and also provides one bit needed for the TWO's complementation.

$$RCs = XE \cdot XS \cdot \overline{ST} \cdot M18 + \overline{T28} \cdot R \cdot YNdx$$

3.5.7.1.1 The first "AND" gate is qualified by the first bit in M18 when dx is negative and ST flip-flop is reset. It sets RC and when YNdx comes in, it adds one bit in the least bit position which makes TWO's complements. YNdx gates are qualified by ST and ST flip-flop is set by the same signals ($\overline{ST} \cdot M18$), which sets RC when dx is negative. This is what makes one bit ready for addition in the least bit position to make TWO'S complements. The second "AND" gate is qualified by ($\overline{T28} \cdot R \cdot YNdx$) when addend and the augend are both high except during T28. This is the normal function of the carry flip-flop.

$$RCr = T28 + ST \cdot \overline{R} \cdot \overline{\overline{YNdx}}$$

3.5.7.1.2 RC is unconditionally reset by T28 (control signal) at the end of each

word to insure the proper start of the next integrator. The second term resets RC when both addend and augend are zero. This term is to make sure that RC does not stay high after the carry is placed in the sum. Timing signals are shown in Timing Diagram, No. 6 and No. 7.

ST flip-flop has been discussed in paragraph 3.5.1.1 under Schematic Control.

3.5.8 dZ GATE (Reference Figure 26; Timing Diagrams, No. 8, 9, 10, 11; and Paragraph 3.4.5.)

The output from the R and YNdx adder is the carry during T28 which is actually an overflow of the R register. This is not, however, the same nature of the overflow that halts the DA-1 computation. It has been shown in paragraph 3.2.5 that the initial value of R is +1/2 and that the sign of dZ depends upon the sign of YNdx. Four examples are also given to show the various possible forms which YNdx could be in. The output of dz gates is channeled through two flip-flops (dZE and dZS). The setting and re-setting signals have been discussed and their timing diagrams shown. Servo gates are also included in the setting and re-setting of dZE flip-flop. The servo flip-flop (SE) is discussed in paragraph 3.4.10.3. The output of the dZ gates is also taken directly to the output register as unmultiplied output of the integrator. The output of dZE and dZS flip-flop is taken to the KdZ adder. The output of dZ flip-flop is delayed by one word time from the start of the computation.

3.5.9 KdZ GATE (Reference Figure 27 and Paragraph 3.4.6.)

The output of dZE and dZS flip-flops is gated through two "and" gates which are qualified by M15. The value of constant K is stored in M15. The reason for using ($\overline{M15}$) the false output of the Read Amplifier (R.A) like some other lines, is that the true side of the Read Amplifier is already loaded enough in the G-15. A buffer inverter is used to get the true output (M15). Both gates are qualified according

to the following equation.

$$KdZ = dZS \cdot \overline{K} \cdot dZE + \overline{dZS} \cdot K \cdot dZE$$

Thus, the output of dZ gates is multiplied by the constant K. The output of both "and" gates is gated into the buffer inverter which has the output as KdZ and $\overline{kdZ}$ (for getting the ONE's complement of kdZ). The product kdZ is added to r (remainder) by the r and kdZ adder.

3.5.10 r and KdZ ADDER (Reference Figure 27; Timing Diagram, No. 12 and No. 13; and Paragraph 3.4.7).

The remainder (r) is stored in M14 and is added to KdZ in the same way as R was added to YNdx. The carry flip-flop ZC handles the carry during the process of addition. After the overflow (end carry), (one bit during T28) has been used to set ZE flip-flop, the new value of r (rn) is recorded back in Line 14 (M14W). The content of rn is the output of four "and" gates qualified by Go and various logical combinations of the output of ZC flip-flop with the addend and augend signals. The Timing Diagrams and equations are given in paragraph 3.4.7.

3.5.10.1 ZC FLIP-FLOP

ZC is the carry flip-flop which performs the same function as YC and RC in the previous two adders. Moreover, ZC also takes care of making TWO'S complements like RC. ZC is immediately set by $dZE_s$ and $dZS_s$ (i.e. when dZ is negative) from the R and YNdX adder. Therefore, one bit will be added to the oncoming first bit of KdZ (ONE'S complement of Kdz) to form TWO'S complements. When dZ is positive or when it does not exist, ZC is reset by dZSr and also by dZEr. ZC is also reset by the "AND" gate which is qualified by the absence of addend, augend and T28.

$$\overline{T28} \cdot GO \cdot \overline{M14} \cdot \overline{KdZ}$$

The new value of r (rn) is written back into Line 14 by four "and" gates qualified by Go.

3.5.11 Zi GATE (Reference Figure 27.)

The output flip-flops ZE and ZS are set by the output of Zi gate. If the multiplied output of KdZ gate is negative (T28 bit is present), $ZS_s$ signal will set the ZS flip-flop in the output register. By the way, this T28 bit will be of the I + 1 word time. The ZE flip-flop (in the output register) will be set by ($\overline{KdZ}$·ZC and GO·T28 + KdZ·$\overline{ZC}$· GO·T28) the output of the two "and" gates during T28 bit time. In other words, the outputs from Zi gates are $ZE_s$ and $ZS_s$ signals during T28 of integrator (I+1).

3.5.12 OUTPUT REGISTER (Reference Figure 28; Timing Diagram, No. 14, 15, 18, 19; and Paragraph 3.4.9.)

The output of Zi gate during T28 of word I + 1 sets the ZS and ZE flip-flops. If the output of the Zi gate is positive, there will not be any $ZS_s$ signal, therefore, ZS will remain reset. The four important flip-flops are discussed separately in the following.

3.5.12.1 ZE FLIP-FLOP

$$\boxed{ZE_s = Go \cdot T28\ (\overline{KdZ} \cdot ZC + kdZ \cdot \overline{ZC})}$$

ZE is set by two "and" gates which are qualified by (Go and T28). At each output ZE will be set by T28 whether it is positive or negative.

$$\boxed{ZE_r = ZE}$$

$ZE_r$ is the resetting pulse which is high during T29 bit time. Therefore, ZE is reset unconditionally by ZE (the true output of ZE). The false output of ZE ($\overline{ZE}$) will be high during T1 through T28, and low during T29 (provided $ZE_s$ is high).

3.5.12.2 ZF FLIP-FLOP

ZF is the associate of ZE flip-flop, the output of which is recorded in 4 word short line 21 (M21). The setting and resetting is done through a buffer inverter.

$$\boxed{ZF_s = Go\ \overline{T28}\ (M21 + ZE)}$$

The preceding equation is not clear from the schematics, but the proof given below does clarify the situation. It is clear from the "AND" gate that the following signals set and reset ZF.

$$
\begin{aligned}
ZF_s &= GO\ \left[\overline{(T28 \cdot Go\ +\ (\overline{ZE} \cdot \overline{M21} \cdot Go)}\right] \\
&= GO\ \left[(\overline{T28}+\overline{Go}) \cdot (ZE + M21 + \overline{Go})\right] \\
&= (GO \cdot \overline{T28}) \cdot (ZE + M21) \\
ZF_r &= GO\ (\overline{ZE} \cdot \overline{M21} + T28) \\
&= GO\ (\overline{ZE + M21} + T28)
\end{aligned}
$$

Both $ZF_s$ and $ZF_r$ show that ZF is reset by T28 and set by $\overline{T28}$ and GO. The output could be high from T1 to T27. Signal GO·T28 is used to block old information from recirculation. The output of ZF is written into Line 21 (M21W) through an "AND" gate when qualified by GO. The information written into Line 21 is precessed one bit every four word time because of ZF flip-flop.

3.5.12.3 OE (OUTPUT EXISTS)

$$
\begin{aligned}
OE &= dZE_s + Go \cdot \overline{P2} \cdot ZF \\
\overline{P2} &= \overline{T28 + T29} = \overline{T28} \cdot \overline{T29}
\end{aligned}
$$

OE (output exists) is the same signal which was discussed with the dY counters. $dZE_s$ is high during T28. Therefore, OE can be high during T28. OE is used as input to dX and dY counters. The output of the "AND" gate qualified by $\overline{P2}$·GO·ZF can be high from T1 through T27. Therefore, OE can have information from T1 through T28.

3.5.12.4 ZS FLIP-FLOP

ZS is the true output of sign flip-flop which is set by the signal $ZS_s$ (the output of the Zi gate). If the output of the Zi gate is negative, $ZS_s$ signal will be high during T28. It will set ZS and the true output resets itself. Therefore, $\overline{ZS}$ is low during T29.

3.5.12.5 ZT FLIP-FLOP

ZT is associated with ZS flip-flop for handling the sign of the output.

The output of the "AND" gate qualified by $\overline{ZS}\cdot GO\cdot\overline{M22}$ resets ZT flip-flop through a buffer inverter.

$$ZT_s = GO\ (\overline{\overline{ZS}\cdot\overline{M22}\cdot\overline{GO}} + T28)$$
$$= GO\left[(\overline{T28})\cdot(ZS + M22 + \overline{GO})\right] = Go\cdot\overline{T28}\cdot(ZS + M22)$$
$$ZT_r = GO\ (T28 + \overline{M22}\cdot\overline{ZS})$$

ZT can be set at any time except during T28 and is reset by T28. The output of ZT will be high at T1.(ZT • GO) qualify the "AND" gate and that writes the information into Line 22. Also, ZT qualifies the "AND" gate which gives ON (output negative) signal. Information written in Line 22 when circulating is precessed one bit for every four word time.

3.5.12.6 ON (OUTPUT NEGATIVE)

ON signal is used as input to dy counter along with OE. ON can have information from T1 through T28 because of $dZS_s$ signal.

$$ON = dZS_s + Go\cdot\overline{P2}\cdot ZT$$

$dZS_s$ is high during T28 only, because ($dZ_s$ = GO•T28•YNdx) sets the sign flip-flop when the output is negative. The output of the "AND" gate could be high from T1 through T27. So, ON can have information from T1 through T28.

3.5.12.7 OP (OUTPUT POSITIVE)

When ZT is reset by T28, it will stop the recirculation of old information already in Line 22. The false output of ZT ($\overline{ZT}$) with ($Go\cdot\overline{P2}$) qualifies the "AND" gate, the output of which is OP. Also, $dZS_r$ is high at T28 when the output is positive.

$$OP = dZS_r + \overline{ZT}\cdot GO\cdot\overline{P2}$$

Both signals give output OP from T1 through T28 which is used in dx and dy counters along with the coincidence pulses to repeat the flow

of information through the DA-1 again.

3.5.13 PLOTTER AND FOLLOWER (Reference Figure 28.)

There are two receptacles in the rear panel of the DA-1 marked "Plotter" and "Plotter Follower". Both are used for graphical input to and output from the machine.

3.5.13.1 The receptacle marked "Plotter" receives the contents of dx counters #3 and #4. Each counter should be set before T0 and remain set throughout word 00, although it may be possible in some cases to set the counter in the first few bits of word 00.

The four "AND" gates for the Plotter are qualified by G0•W0 (control signal). The four outputs ( $\pm\Delta x_2$ and $\pm\Delta y_2$) of the four "AND" gates are recorded by the Plotter in the form of a graph.

$$+\Delta x_2 = G0 \cdot W0 \cdot X3 \cdot XC$$
$$-\Delta X_2 = G0 \cdot W0 \cdot X3 \cdot \overline{XC}$$
$$+\Delta Y2 = G0 \cdot W0 \cdot X4 \cdot XD$$
$$-\Delta Y2 = G0 \cdot W0 \cdot X4 \cdot \overline{XD}$$

Line 6 contains the information which with OE signal sets X4 flip-flop of the dx counter. Dump pulses are located at 2 mod. 4 in Line 6. The associated flip-flop XD gives the output for the sign of $\Delta Y$ to the Plotter. The output of any integrator in the series (1 mod. 4) 1, 5, 9, 13.....105 may provide the $\Delta Y$ output to the plotter. The integrator will provide $\Delta Y$ if it is coded to be the dx input into integrator 03. Line 7 has the information which with OE sets X3. Line 7 has dump pulses located at 1 mod. 4. The associated flip-flop XC gives the output for the sign of $\Delta X$ to the plotter. The output of any integrator of the two series (0 mod. 4 and 1 mod. 4), may provide the

0, 4, 8, 12, 16, .....104
1, 5, 9, 13, 17, .....105

$\Delta X$ output to the plotter. The integrator will provide $\Delta X$ if it is

coded to be the dx input into integrator 02.

3.5.13.2 The receptacle marked "Plotter/Follower" gets the input from four "AND" gates qualified by GO•WO control signal. The four outputs are $\pm$ ΔY1 and $\pm$ ΔX1.

$$+\Delta Y1 = Go \cdot WO \cdot X2 \cdot XB$$
$$-\Delta Y1 = Go \cdot WO \cdot X2 \cdot \overline{XB}$$
$$+\Delta X1 = Go \cdot WO \cdot dZE \cdot \overline{dZS}$$
$$-\Delta X1 = Go \cdot WO \cdot dZE \cdot dZS$$

Information in Line 8 with OE sets X2. Dump pulses in Line 8 are located at 0 mod. 4. The sign flip-flop XB supplies the sign bit to the output. If the plotter is plugged into Plotter/Follower receptacle, the output of any integrator may provide the ΔY output, except integrators in the series: (2 mod. 4)

2, 6, 10, 14, 18, .....106 .

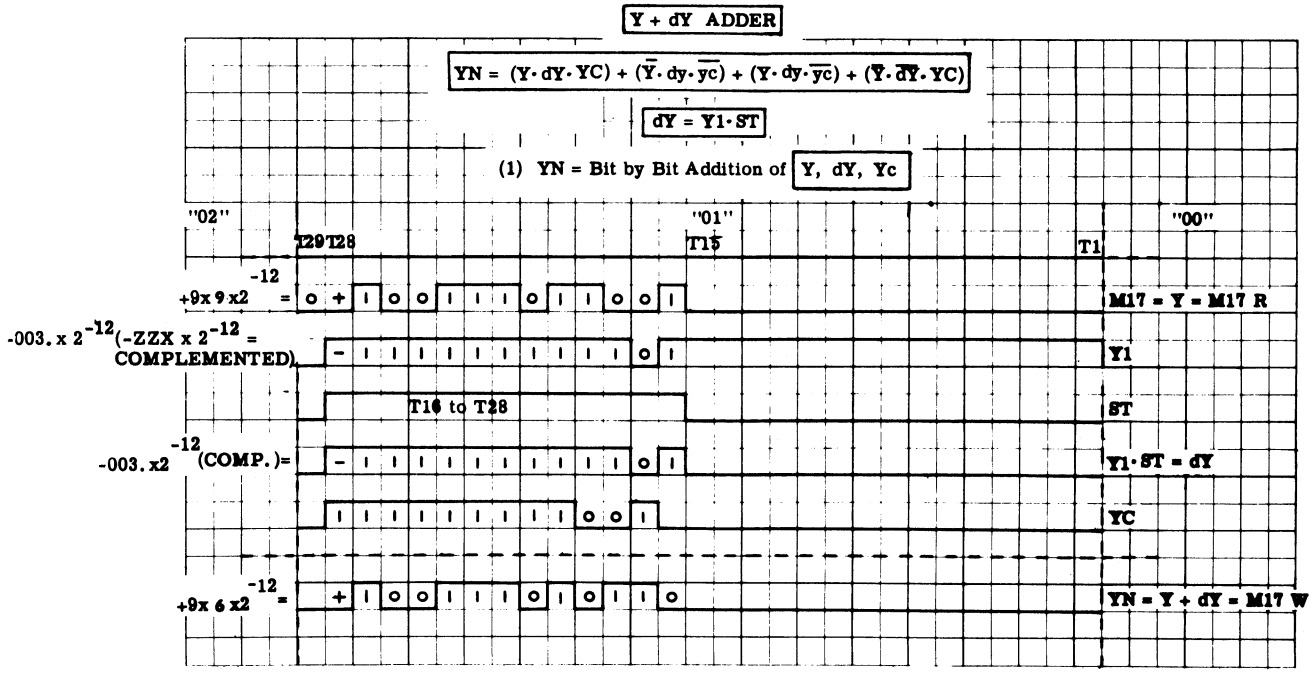
The integrator will provide ΔY if it is coded to be the dX input to integrator 01.

$\pm$ΔX is the output of two "AND" gates which are qualified by the unmultiplied output of the integrator. The unmultiplied output of integrator 107 will provide the ΔX output to the plotter and no additional coding is necessary.

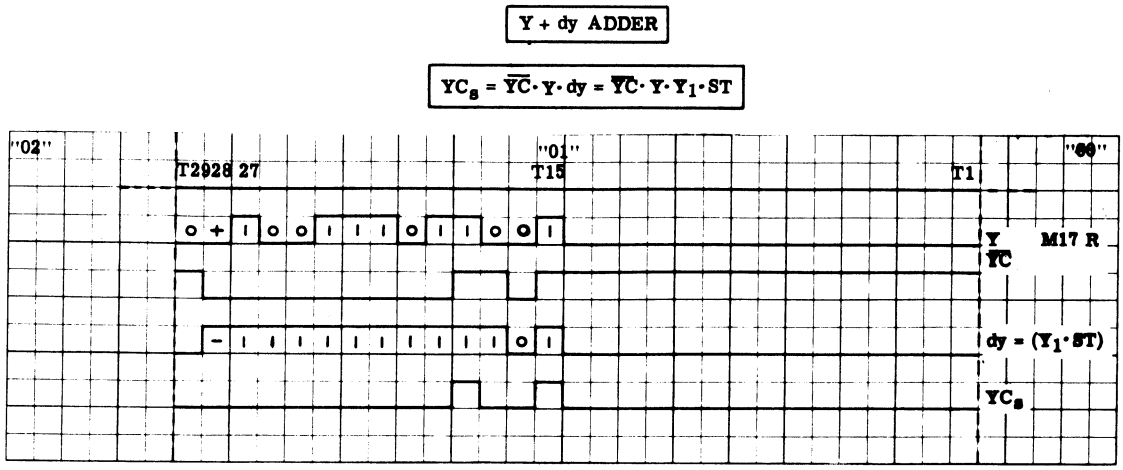
If the follower is plugged into this receptacle, the unmultiplied output of integrator 107 will provide the ΔX output to the follower and no additional coding is necessary. The ΔY increments from the follower will enter as the dx input into integrator 04 and no coding is necessary.

From the above discussion it is clear that when the plotter is plugged into the plotter receptacle, all the 108 integrators are being used and, thus, all the long (108 word) lines are occupied and programmer does not have a free memory line for general purpose use. On the

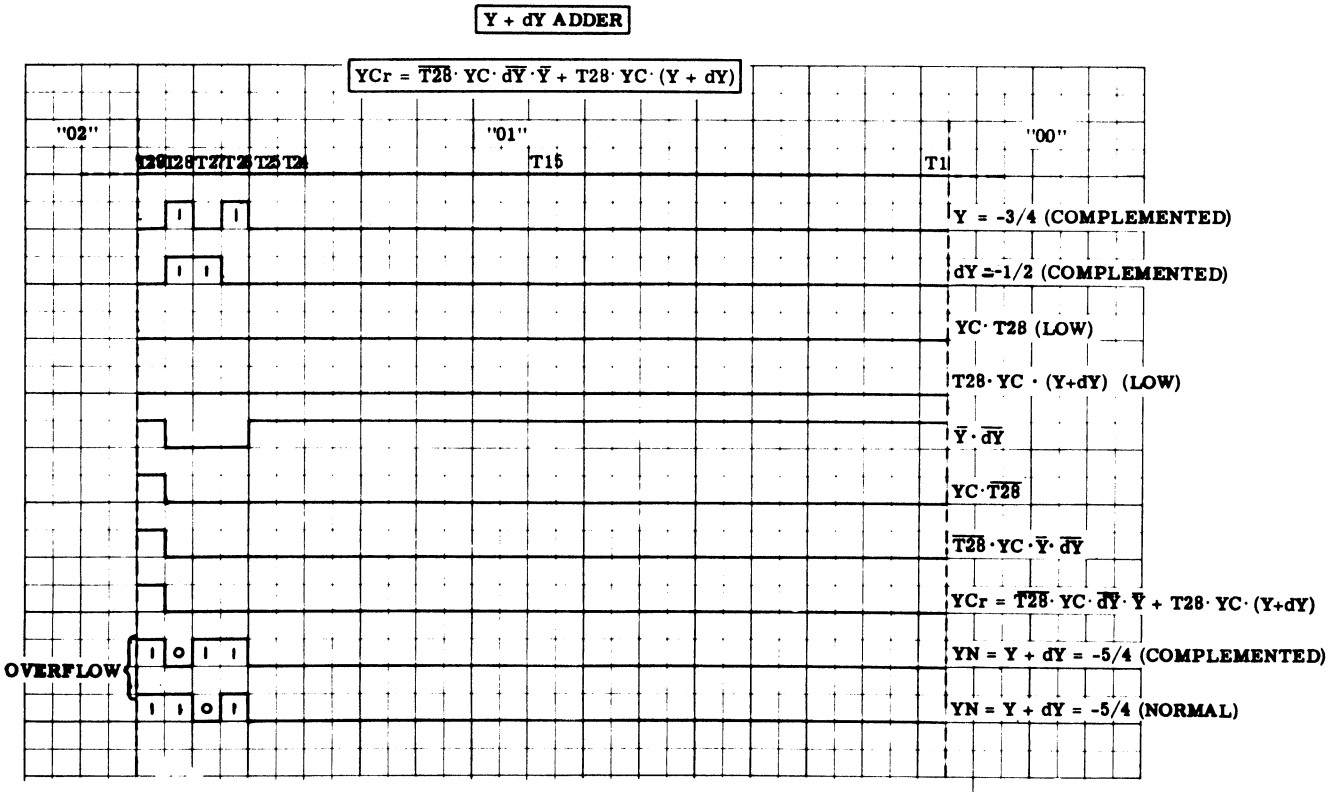
other hand, when the follower is plugged into the Plotter/Follower receptacle, only 81 integrators are necessary and, thus, leaving the Line 5 and 6 free. The Followers are not in use at the present time. Therefore, if +ΔY2 and -ΔY2 are exchanged with +ΔY1 and -ΔY1 respectively, by exchanging the plug pins PLF 4-3 and 4-4 with PLF 3-3 and 3-4 respectively, the plotter can be used with 81 integrators, leaving line 5 and 6 free. Line 5 is used for storing the YO (initial) value. This is a simple modification and can be executed in the field whenever need arises.



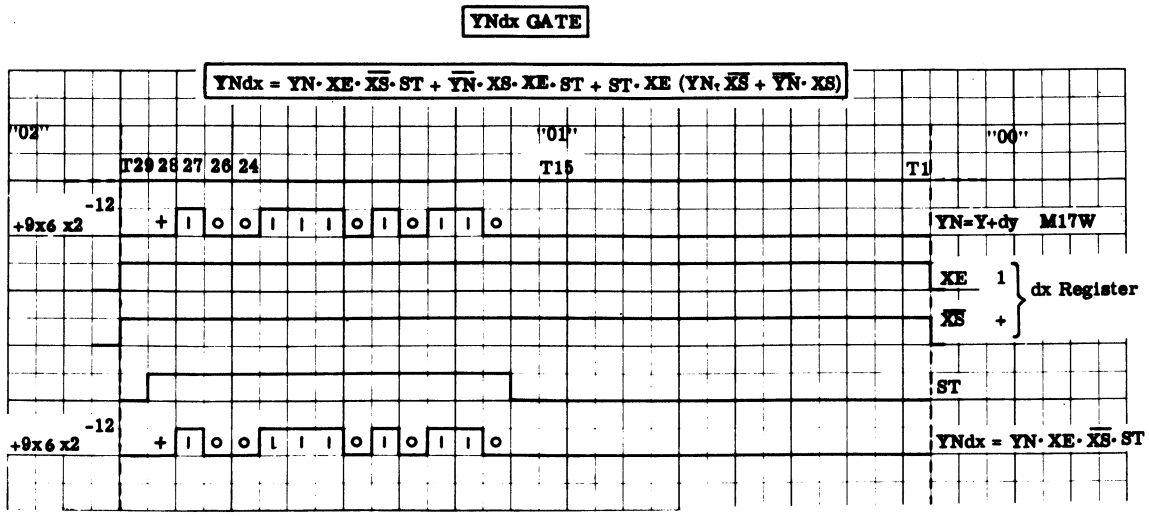

TIMING DIAGRAM, NO. 1 (INTEGRATOR "01")



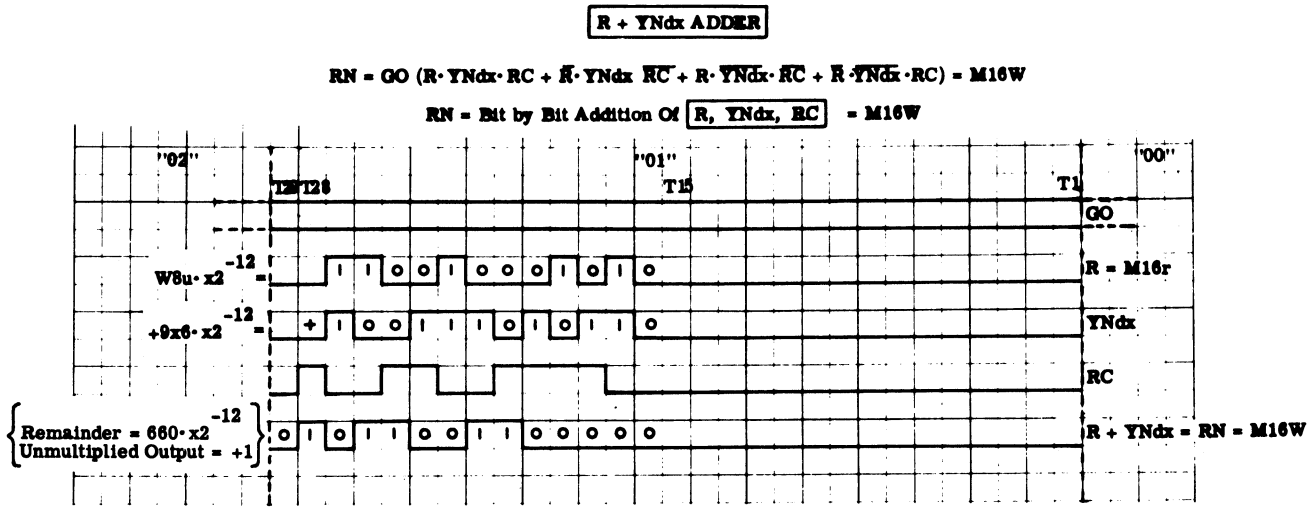

TIMING DIAGRAM, NO. 2 (INTEGRATOR "01")



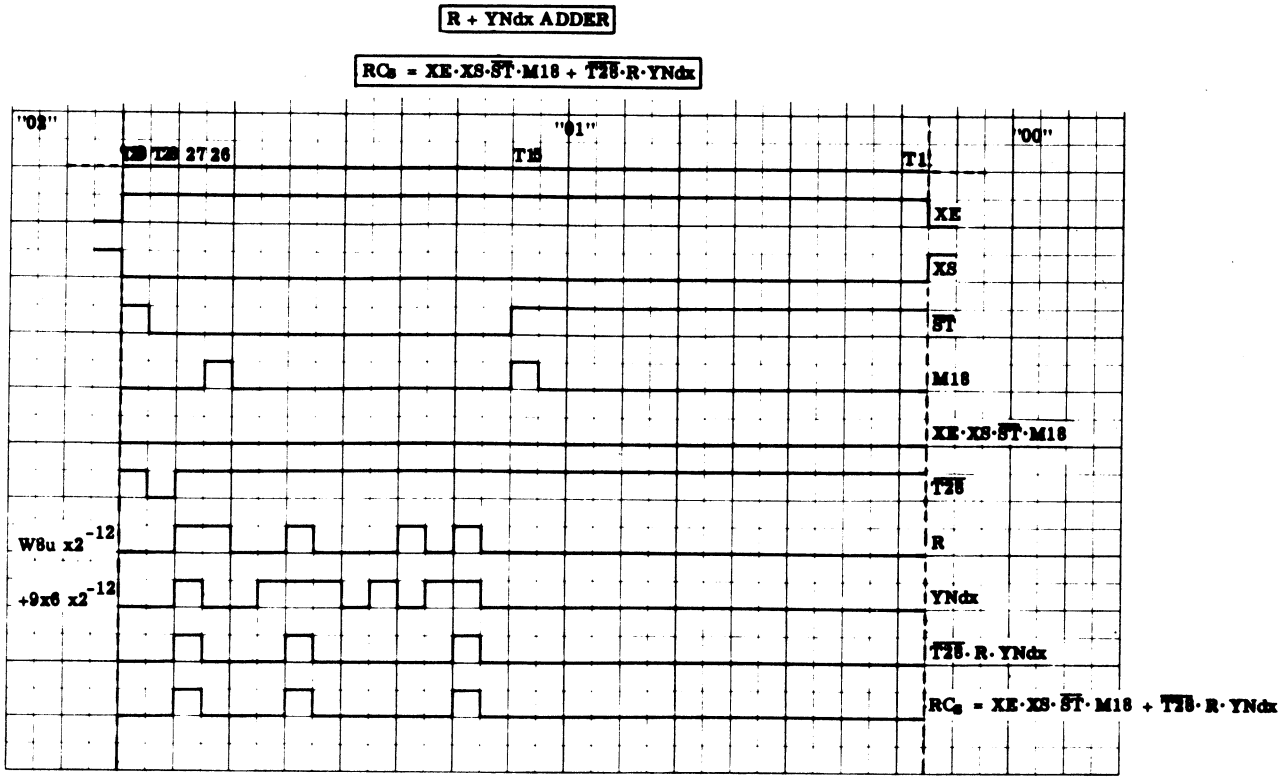

TIMING DIAGRAM, NO. 3 (INTEGRATOR "01" - EXAMPLE "E")



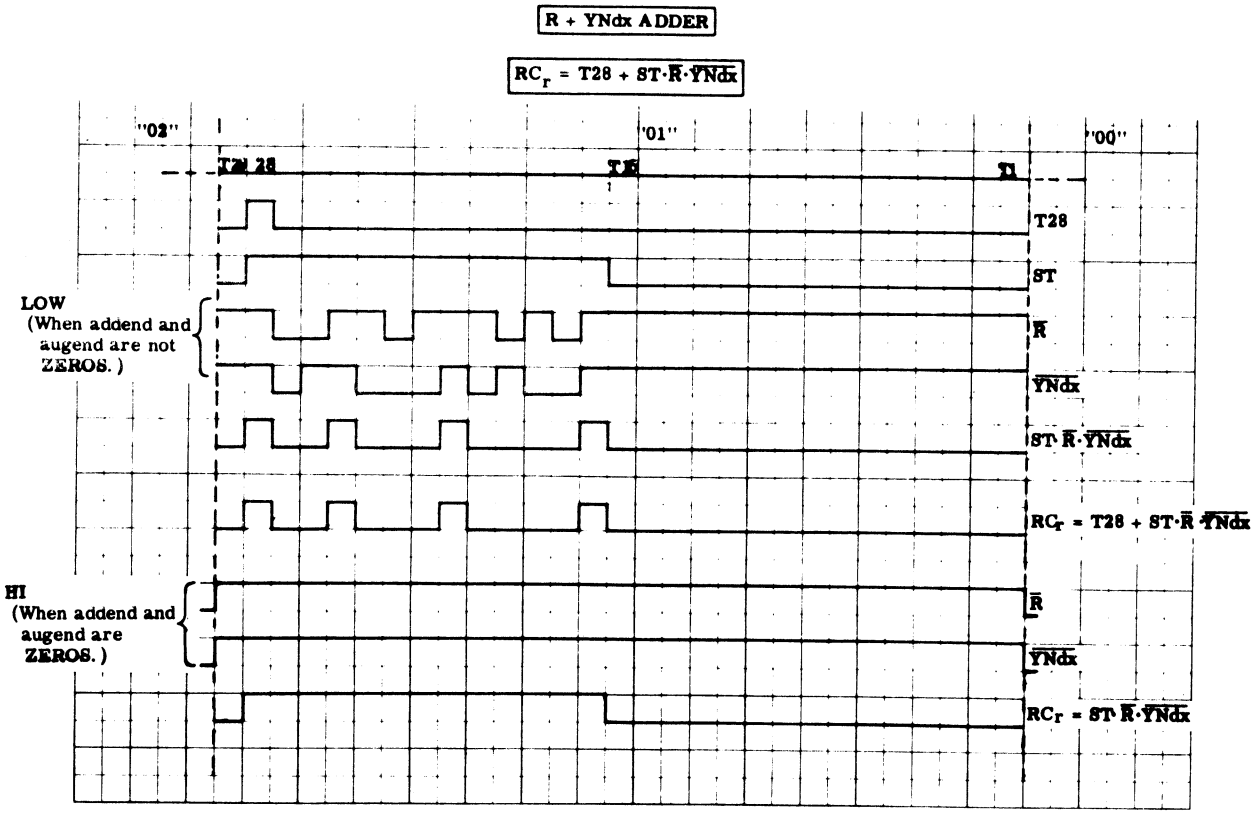

TIMING DIAGRAM, NO. 4 (INTEGRATOR "01")



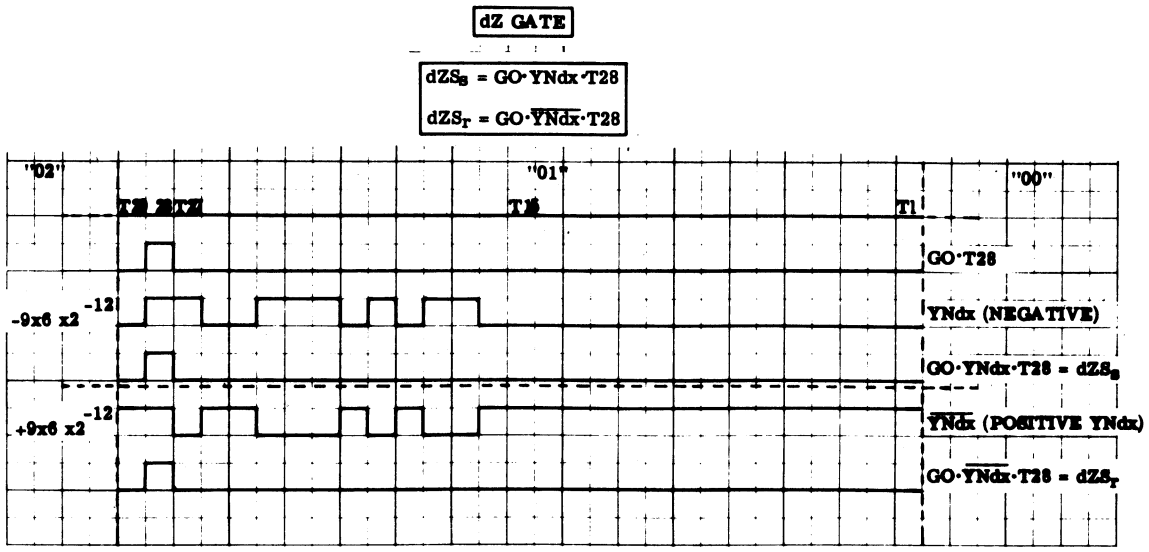

TIMING DIAGRAM, NO. 5 (INTEGRATOR "01"



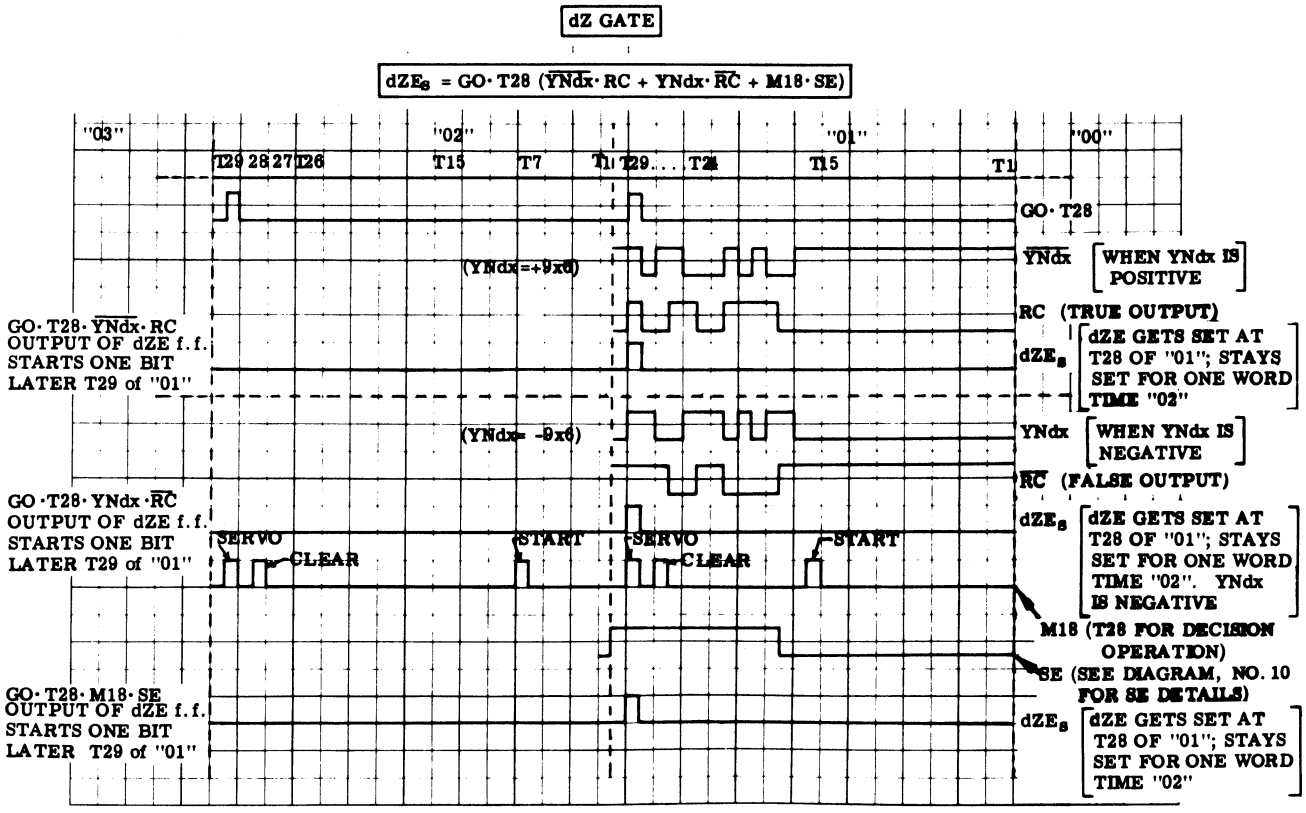

TIMING DIAGRAM, NO. 6 (INTEGRATOR "01"




TIMING DIAGRAM, NO. 7 (INTEGRATOR "01")



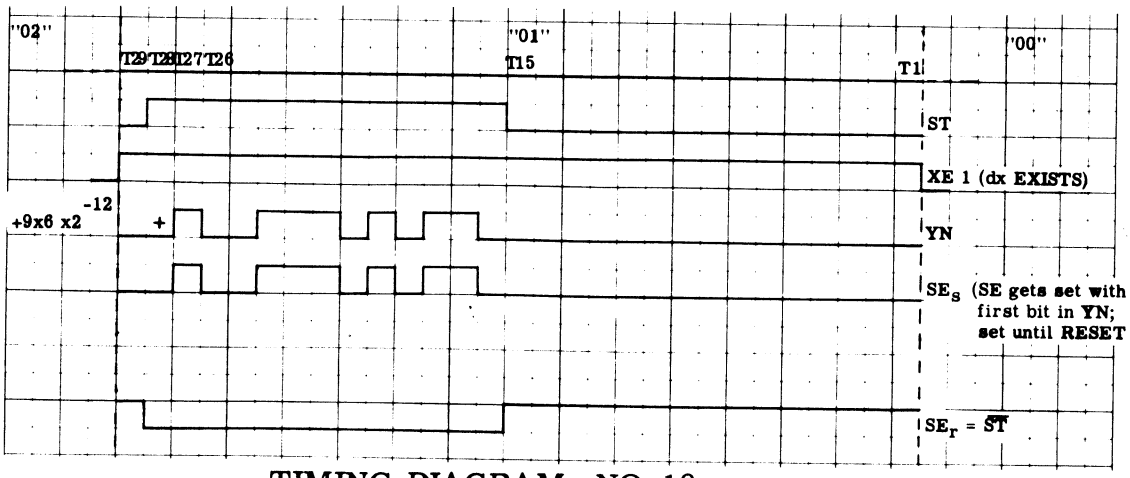

TIMING DIAGRAM, NO. 8 (INTEGRATOR "01" - TOP PART ONLY)




TIMING DIAGRAM, NO. 9

dZ GATE

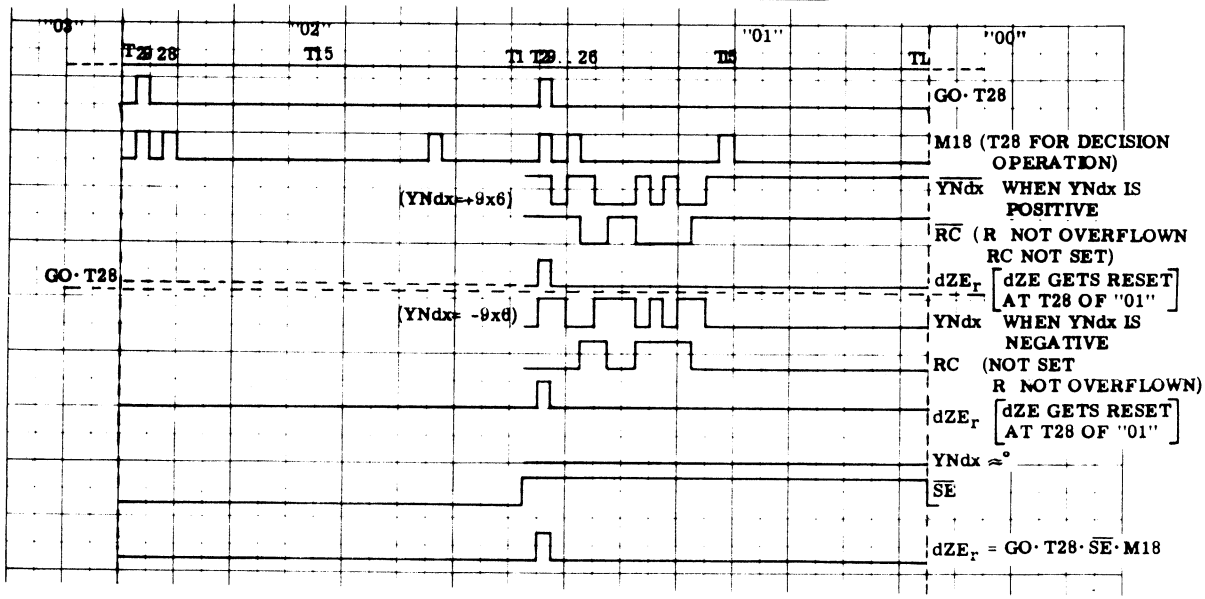
$SE_S = \overline{ST} \cdot YN \cdot XE$
$SE_r = \overline{ST}$




TIMING DIAGRAM, NO. 10

dZ GATE

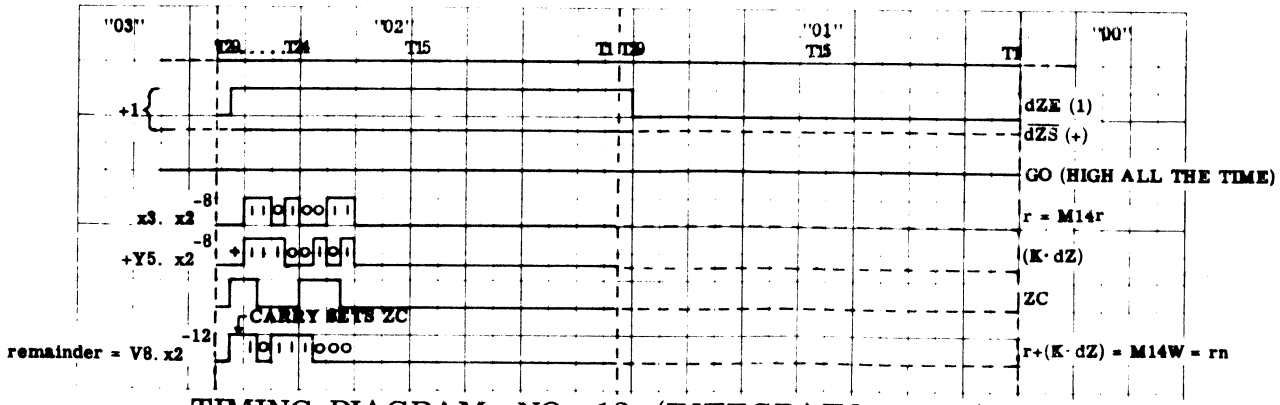
$dZE_r = GO \cdot T28 \cdot \overline{M18}\ (\overline{YNdx} \cdot \overline{RC} + YNdx \cdot RC) + GO \cdot T28 \cdot \overline{SE} \cdot M18$



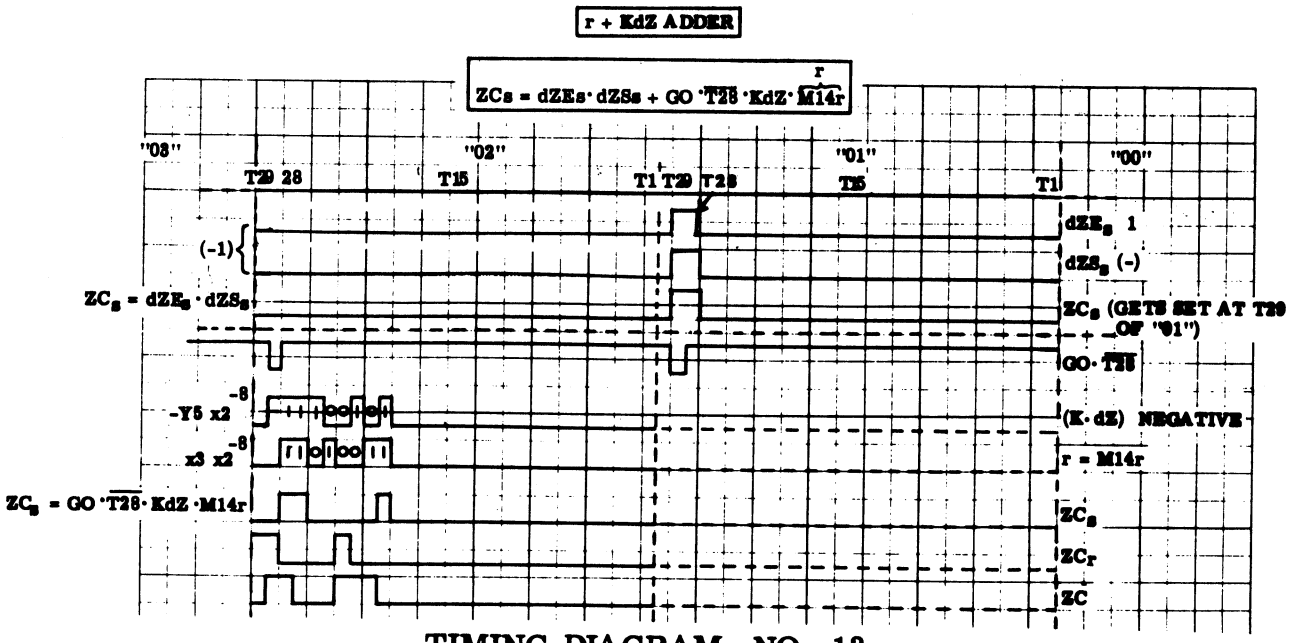

TIMING DIAGRAM, NO 11

r + KdZ ADDER

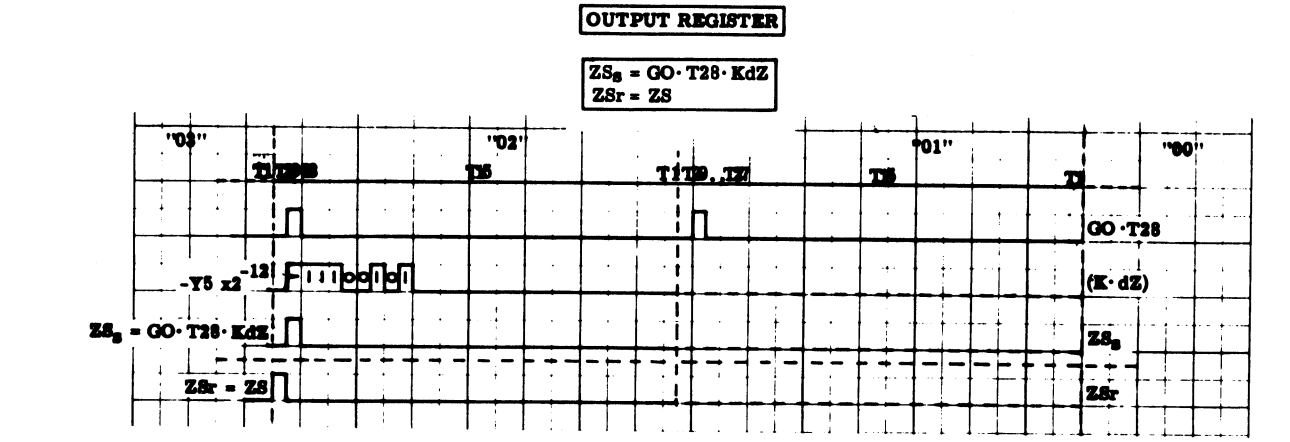
$rn = GO\ [M14r \cdot (K \cdot dZ) \cdot ZC + \overline{M14r} \cdot (K \cdot dZ) \cdot \overline{ZC} + M14r \cdot (\overline{K \cdot dZ}) \cdot \overline{ZC} + \overline{M14} \cdot (\overline{K \cdot dZ}) \cdot ZC] = M14W$



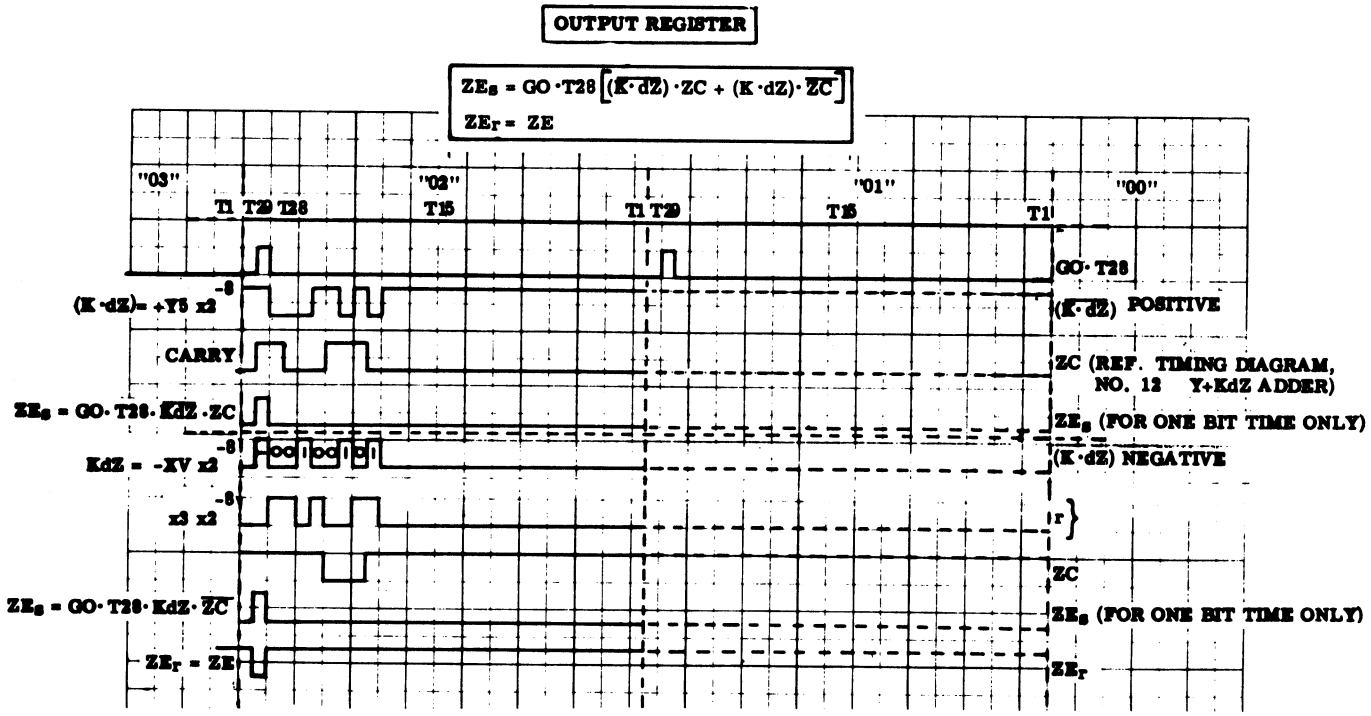

TIMING DIAGRAM, NO. 12 (INTEGRATOR "01")



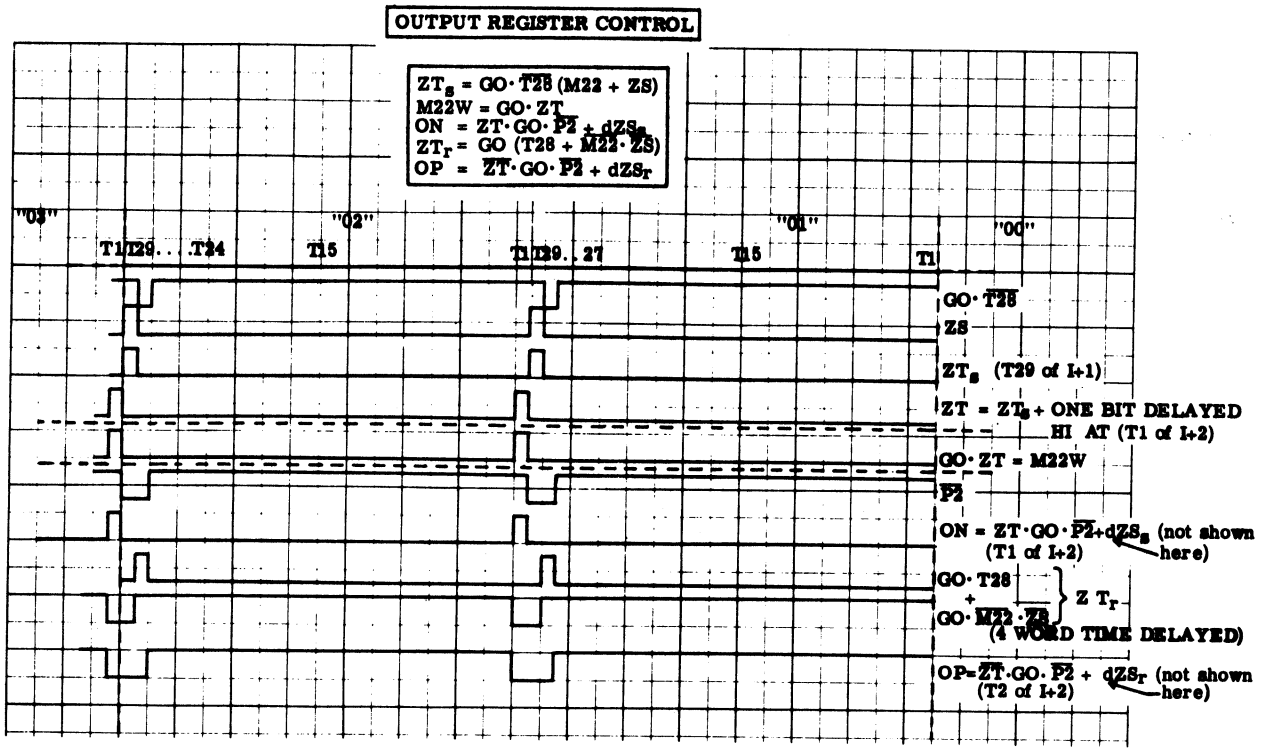

TIMING DIAGRAM, NO. 13



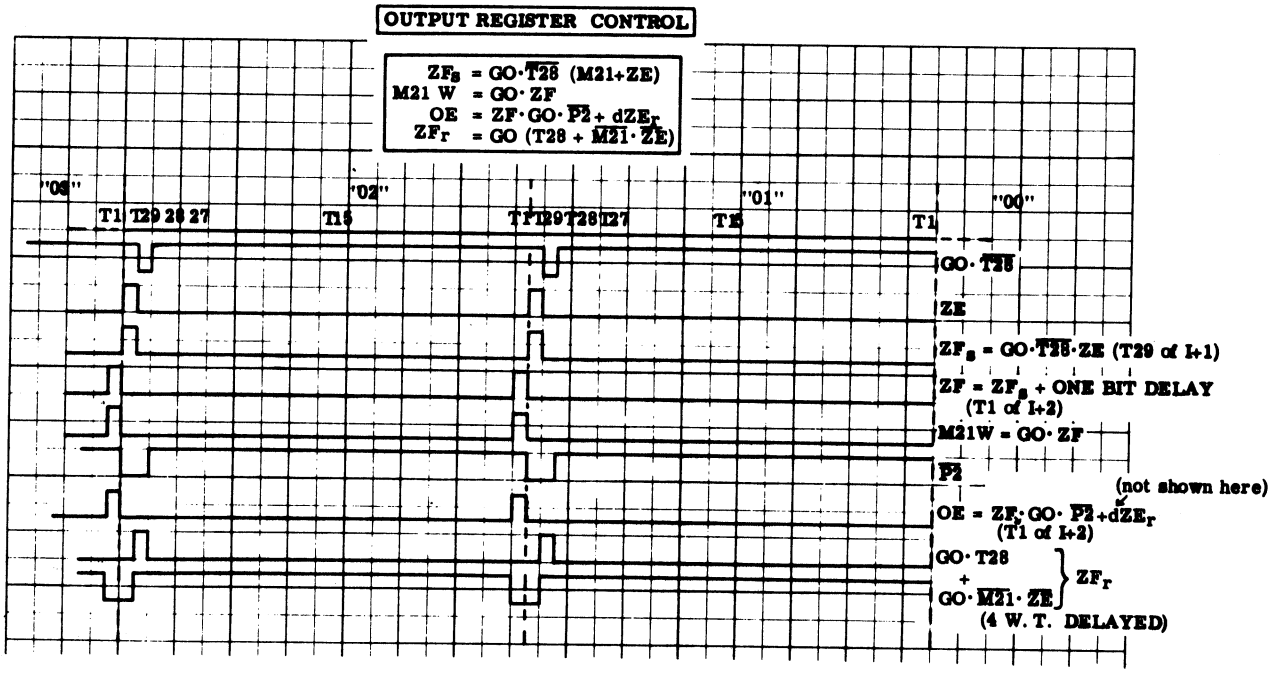

TIMING DIAGRAM, NO. 14



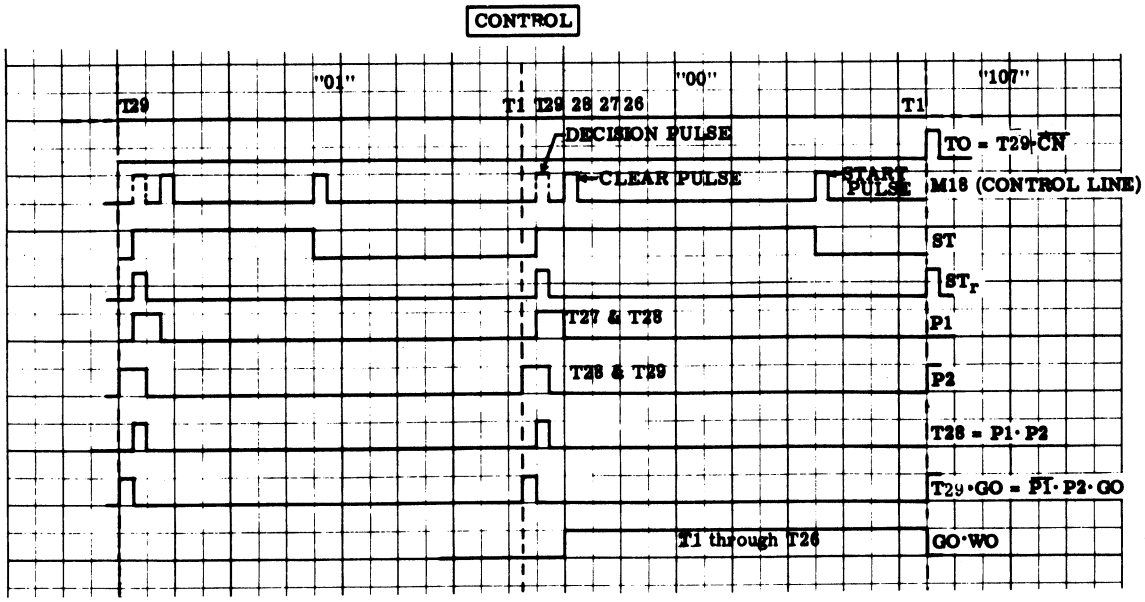

TIMING DIAGRAM, NO. 15



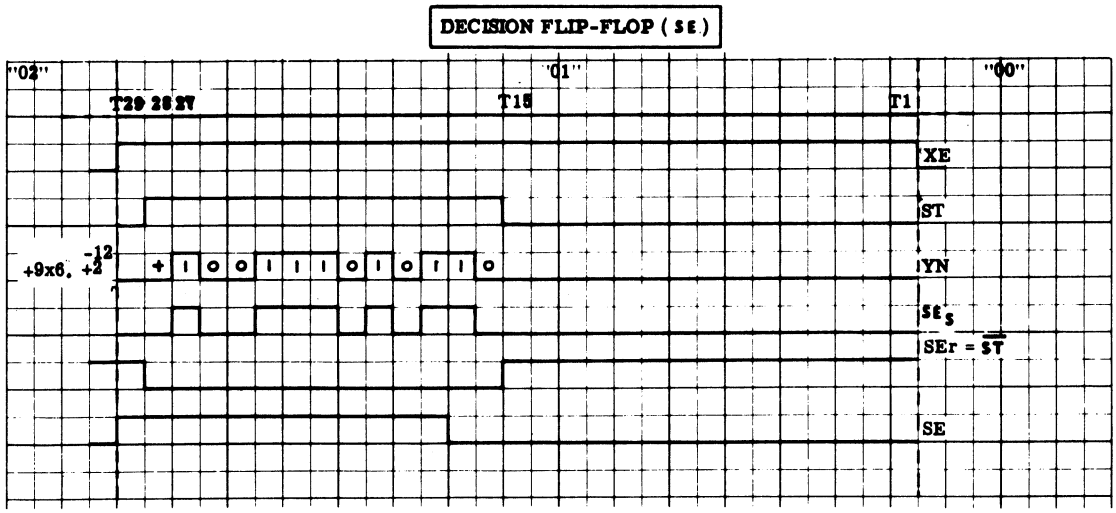

TIMING DIAGRAM, NO. 16




TIMING DIAGRAM, NO. 17




TIMING DIAGRAM, NO. 18




TIMING DIAGRAM, NO. 19

# SECTION IV

# POWER SUPPLY

4.0 POWER SUPPLY

4.1 GENERAL (Reference Figure No. 35, 36, 37 and 38).

The DA-1 has a part of its power supply built into the unit. The +250, -13 and +160 volt supplies are brought from the G-15 because of the low power requirement. +250 and -13 volts are used in the clock repeater package (black handle) while +160 volts supply is used in the plotter (PA-2 only). -20 volts supply from G-15 and DA-1 are tied together through switches 1 and 2 and R-13. +100 volts, -20, -160 and the filament voltages are generated in the DA-1. All the potentials from DA-1 along with +250 volts from G-15 can be tested by the percent DC voltmeter. There are six relays which switch the power in a sequence similar to G-15.

The power supply section (lower front hinged panel) can be opened by taking out four sheet metal screws and dropping down the hinged panel. (See Figure 37). The plug connections are shown in Figure 6. The back panels of DA-1 and G-15 are also shown for various plug connections which are discussed below:

4.2 PLF - 1 (DIFFERENTIAL ANALYZER)

With reference to Figure 6, the receptacle on the left, (PLF-1), is labelled "Differential Analyzer". This plug has all the logic pin connections. The pin connections are provided along with the associated signals in the continuity chart, Figure 39.

4.3 PLF-2 (DA-POWER)

This plug (Reference Figure 40) connects the G-15 supplied power to DA-1. The pin connections are given in the Continuity Chart, Figure 40.

4.4 PLF-3 (GRAPH PLOTTER AND FOLLOWER)

This plug is used mainly for the Follower. It can also be used for

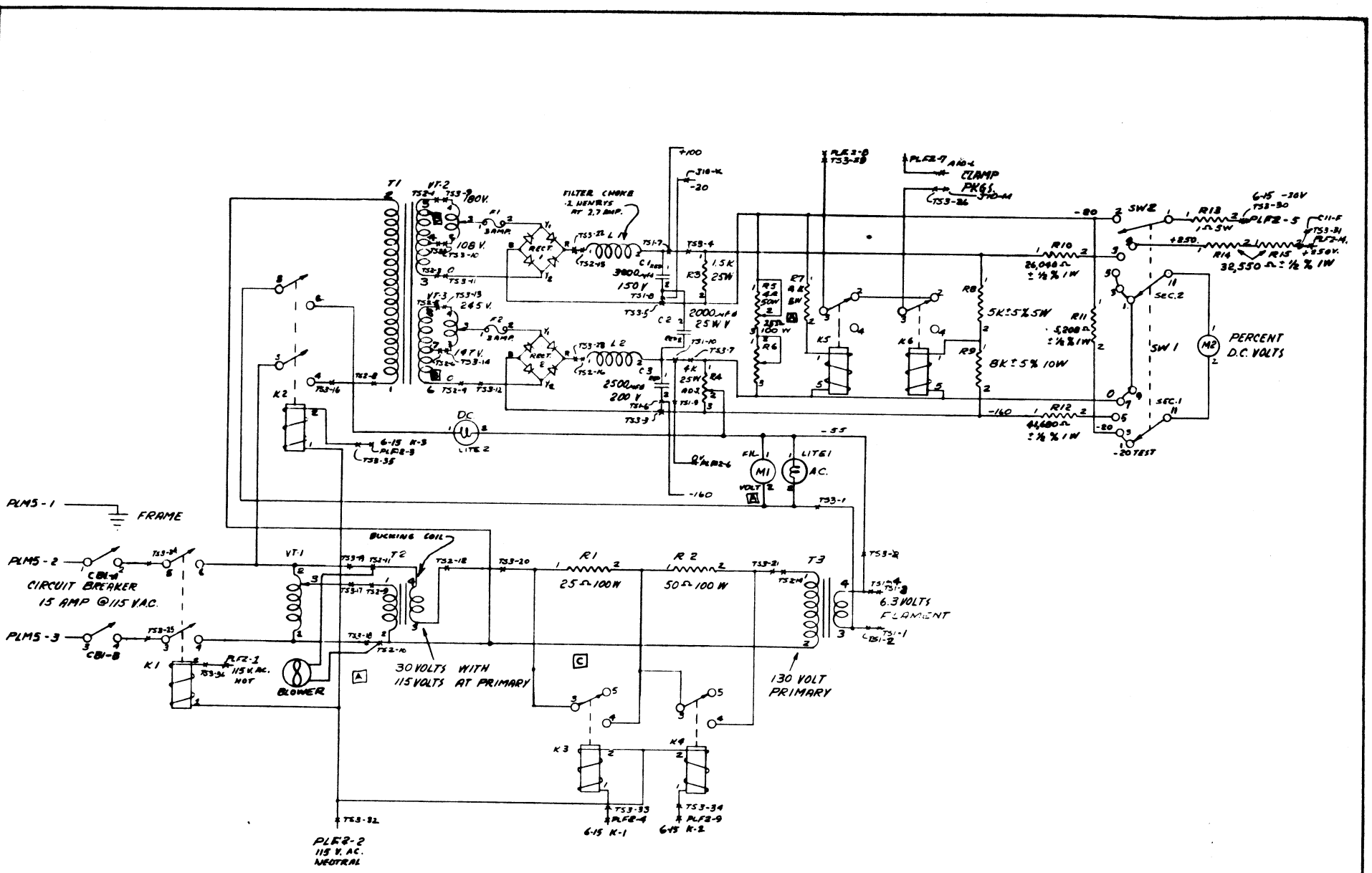



Figure 35, Schematic, Power Supply

1) PLF2 ON DA-1 CONNECTS TO PLF4 ON GSD

NOTES: UNLESS SPECIFIED

Plotter after some minor changes in the program. See Figure 40 for pin connections.

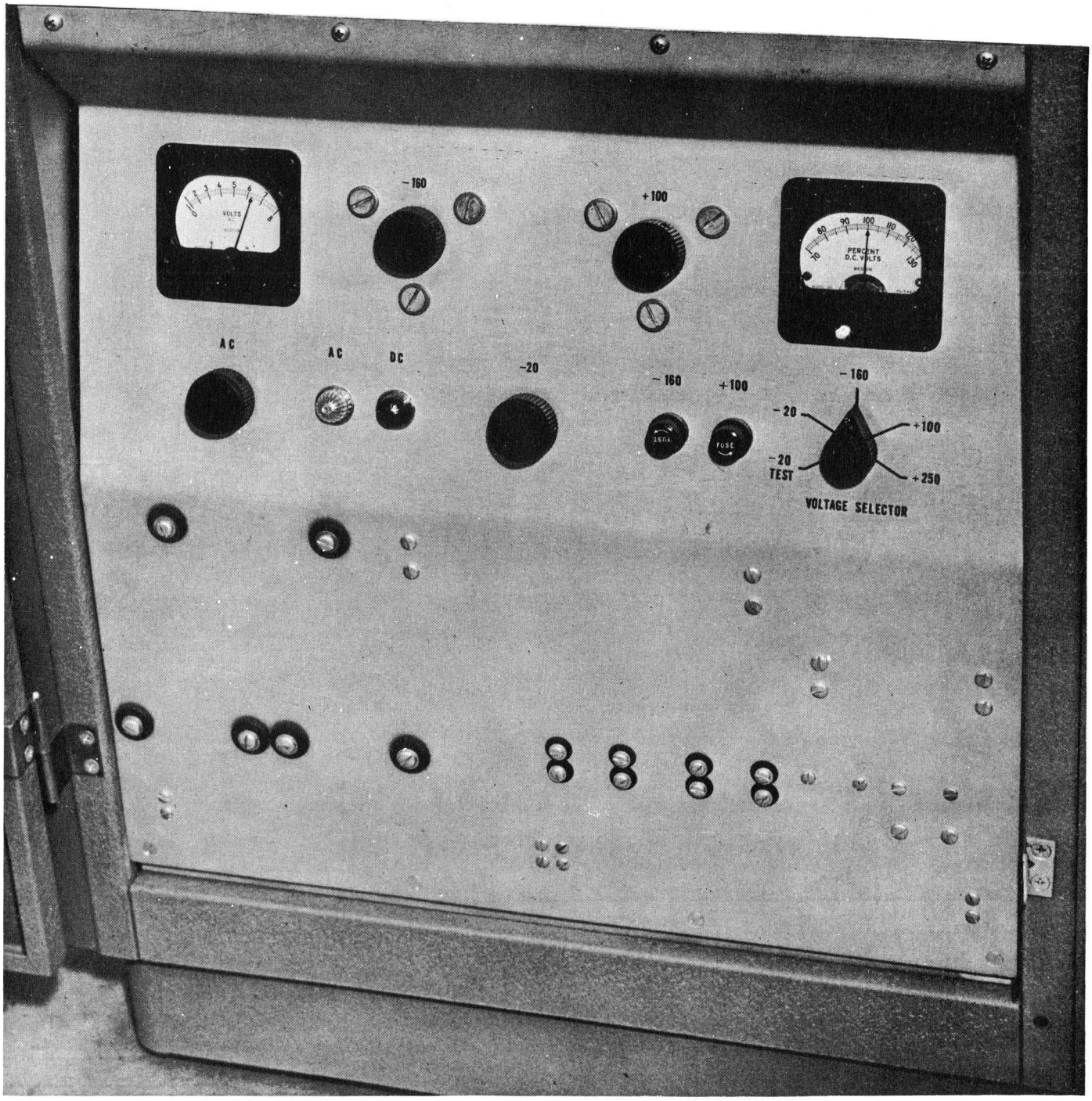



Figure 36. Power Supply, Front View

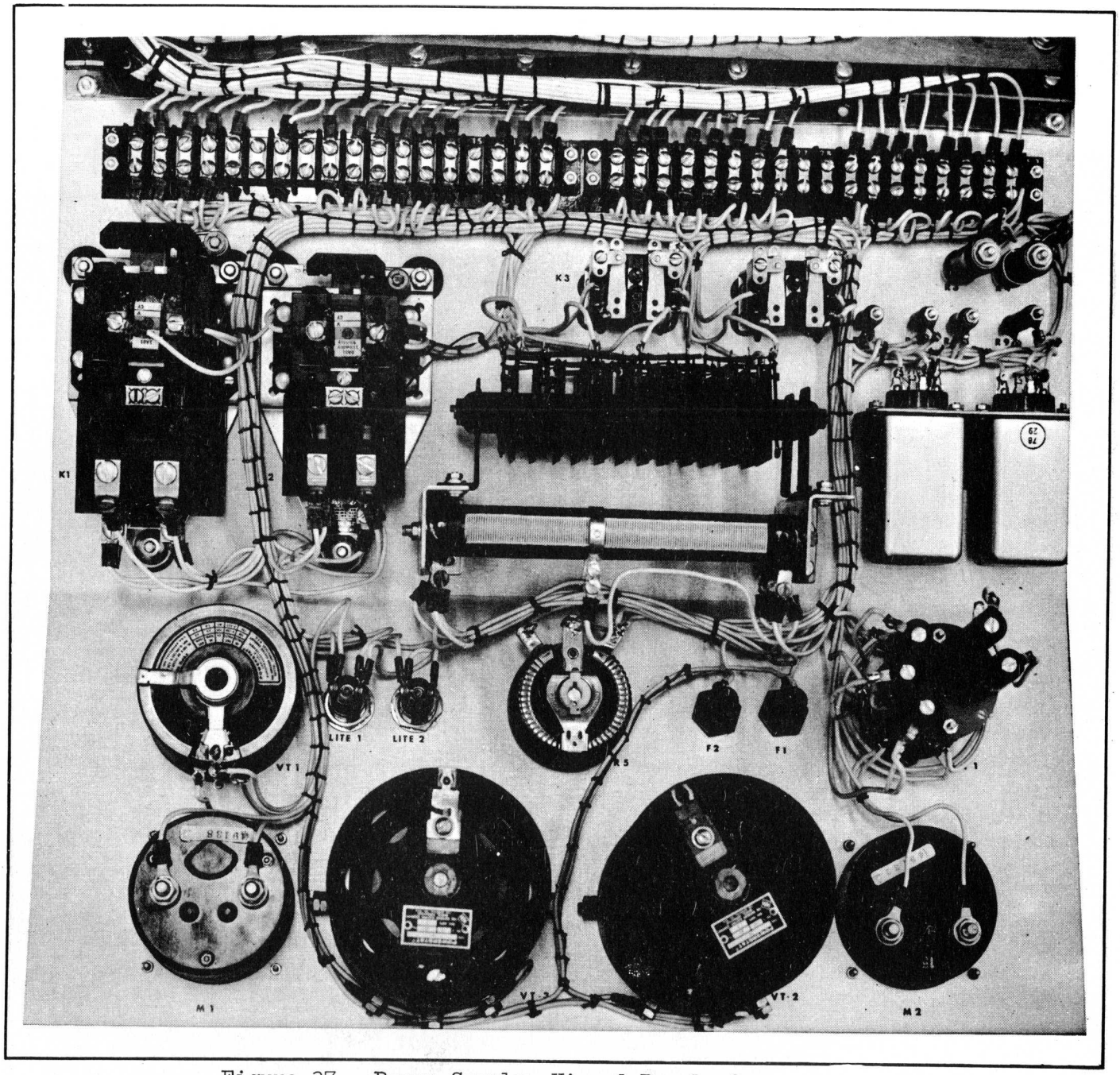



Figure 37. Power Supply, Hinged Panel, Open

## 4.5 PLF-4 (GRAPH PLOTTER)

The Graph Plotter is connected into PLF-4. For pin connections, refer to the chart given in Figure 40.

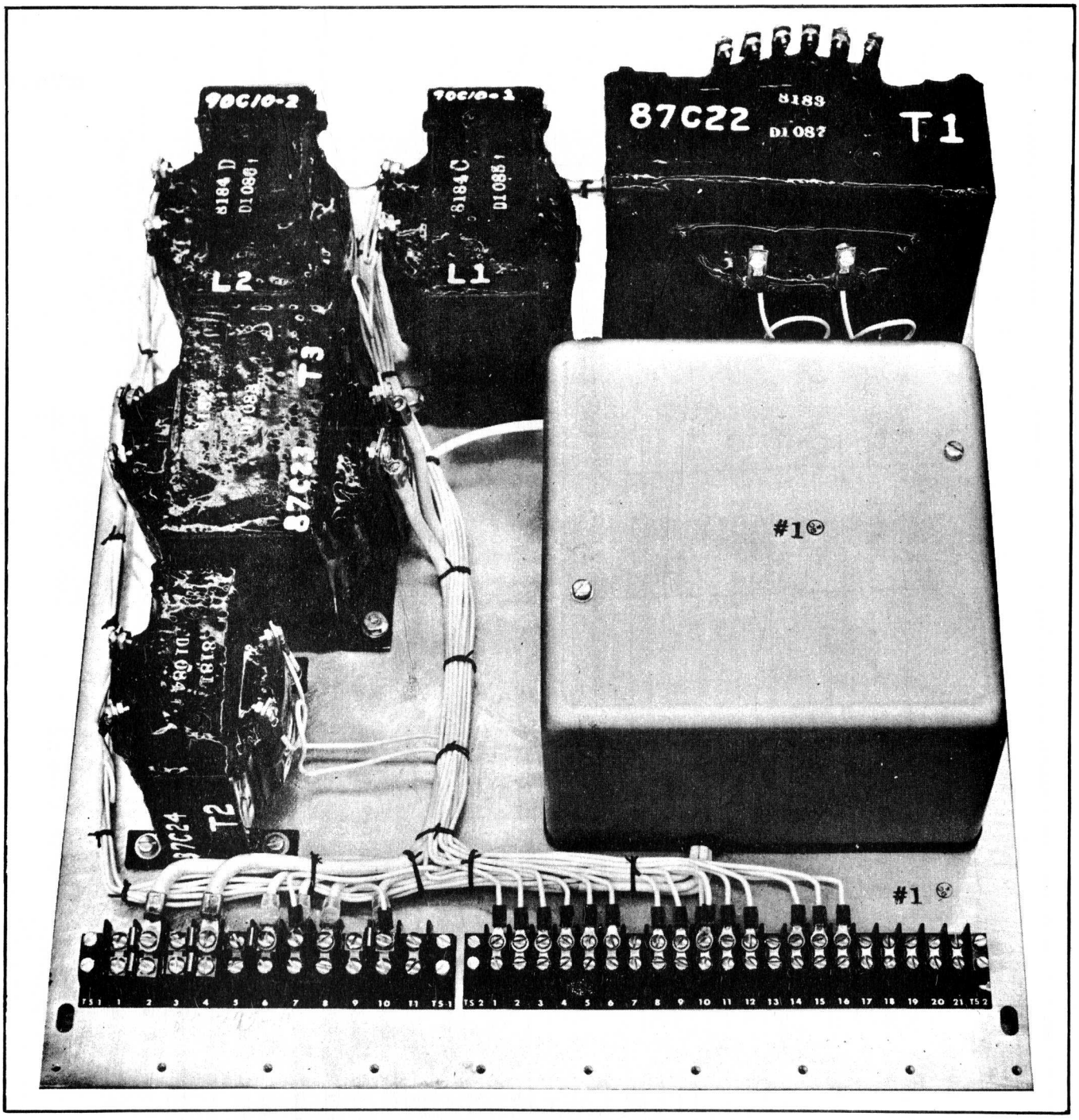



Figure 38. Power Supply, Transformer Section

## 4.6 PLM-5 (A.C. PLUG)

The PLM-5 connects the A.C power line to K-1 relay through the circuit breaker. K-1, when closed, supplies A.C. power to the primary

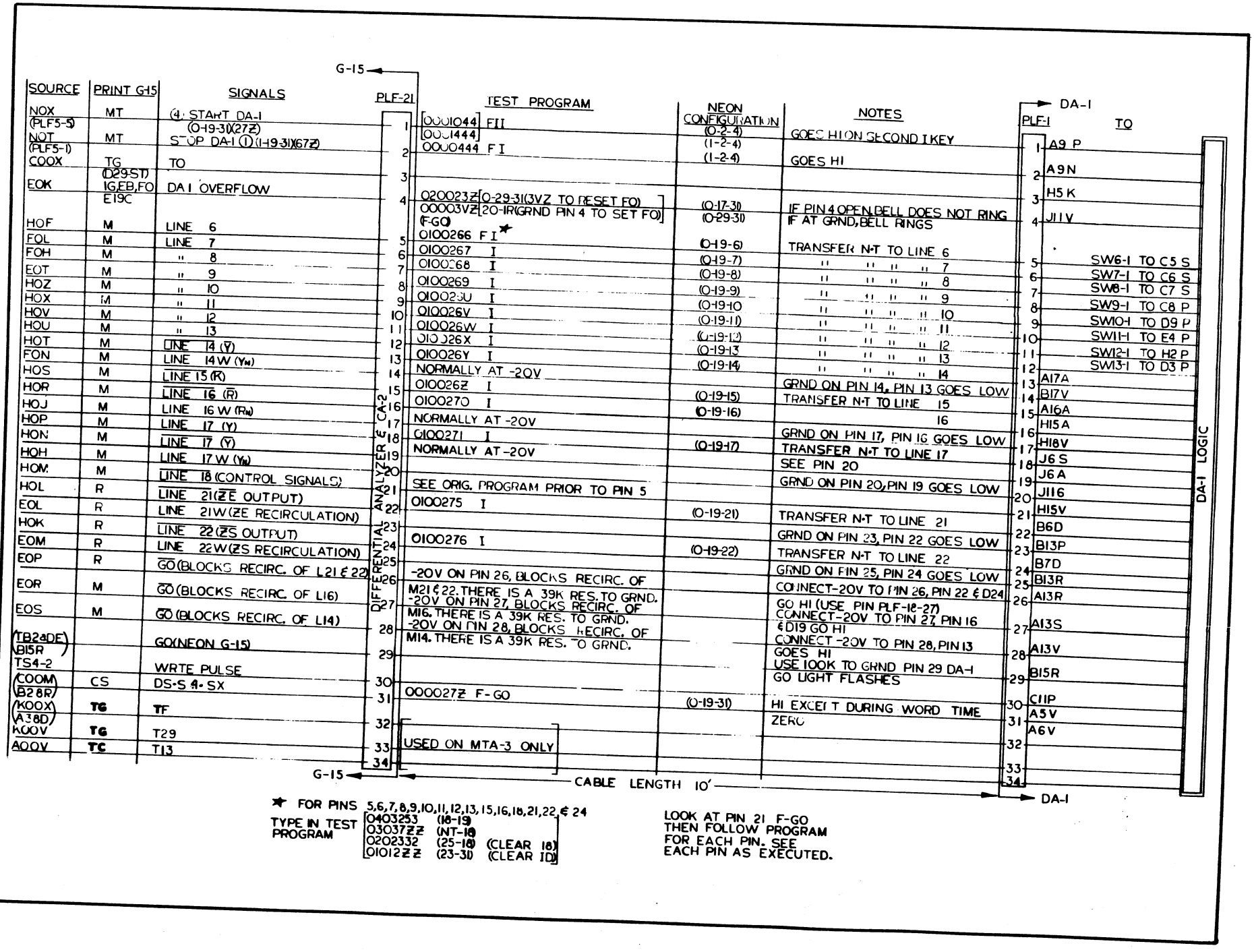



Figure 39. G-15 - DA-1 Continuity Chart

of the (VT-1) variable transformer. Refer to Figure 40 for interconnections.

4.7 POWER TURN - ON CYCLE (Reference Figure 35, 41 and 42).

4.7.1 The DA-1 is connected to G-15 with proper plug connections which are shown in Figure 6. The circuit breaker of DA-1 is turned on. As soon as the A.C. power of G-15 is turned on, K-1 relay of DA-1 pulls in and power from the line is supplied to the blower and the primary of the (VT-1) variable transformer. The output of VT-1 is stepped down by T2. The secondary of T2 (the bucking coil) is connected to the primary of T3 (filament transformer) with resistance (R1 and R2) in series. K-3 and K-4 relays increment the filament voltage as shown in Figure 42. K-3, K-4 and K-2 of DA-1 are in parallel with K-1, K-2 and K-3 of G-15 respectively.

4.7.2 When power is turned ON, some small filament voltage will appear in the secondary of T3. K-3 of DA-1 will pull in to short out R1 when K-1 of G-15 has pulled in. K-4 will pull in when K-2 of G-15 has pulled in, and K-3 has dropped out; thus, shunting out R2 and keeping R1 in series with the primary. After a short time (controlled by the timing motor, TM1, in G-15) K-3 will also pull in to short out R1 again, and thus, stepping up the output to 6.3 volts in steps. A simplified schematic of power relays of G-15 and DA-1 is provided in Figure 41.

4.7.3 The A.C. incandescent lamp will light dimly immediately after the power switch is turned ON, but will get brighter as the voltage steps up. Watch the indicator of filament A.C. voltmeter (M-1) after turning on the power switch, and see how the filament voltage steps up. A filament voltage incrementation graph is shown in Figure 42 to give the relationship of different relays of DA-1 and G-15.

4.7.4 Immediately after the reset button has been pressed, K-3 of G-15 will pull in along with K-2 of DA-1, thus, completing primary circuit

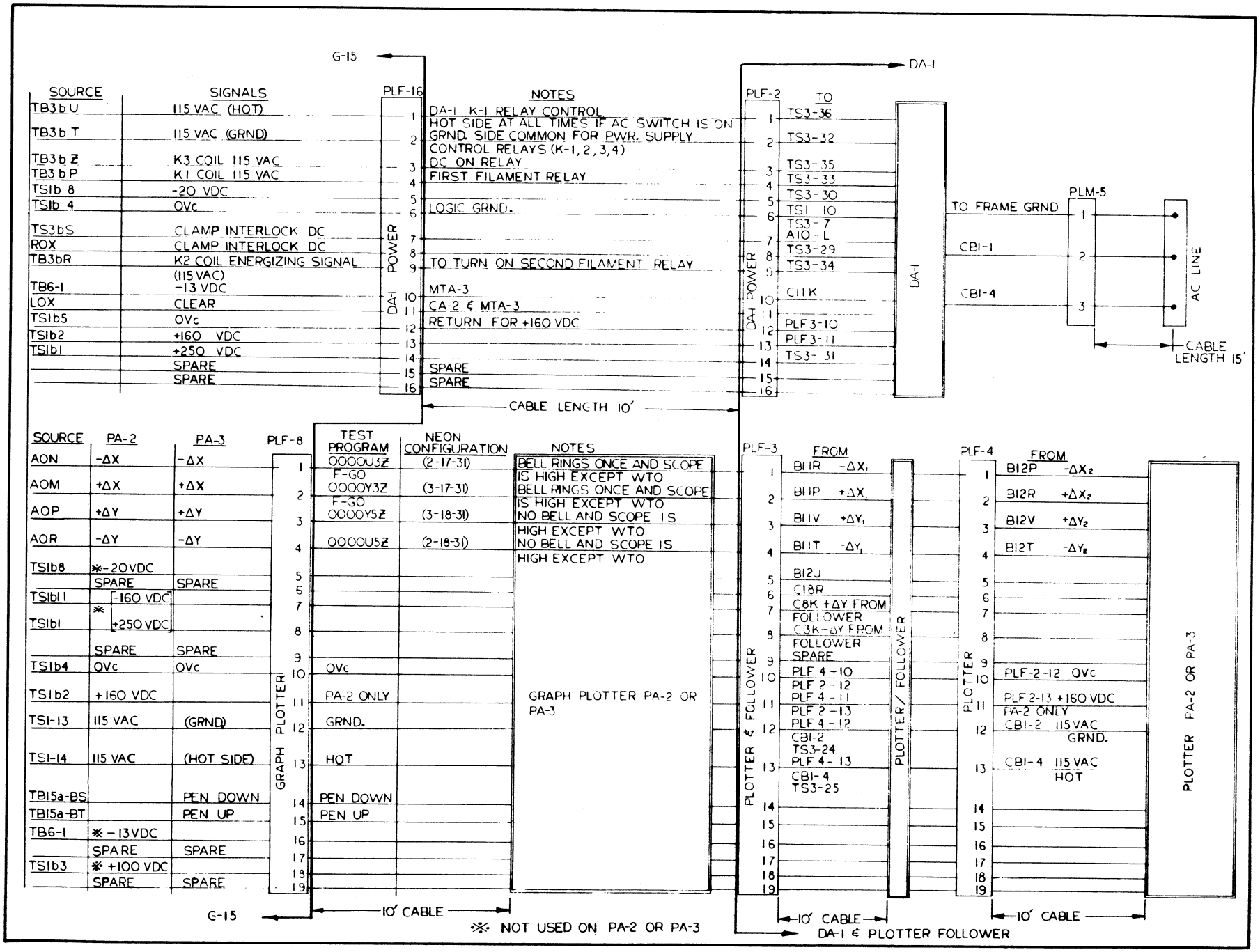



Figure 40. G-15 - DA-1 Graph Plotter and Follower Continuity Chart

of T1. The secondary of T1 through VT2 and VT3, supply power to the conventional full wave bridge rectifiers (rect. 1 and 2) through two 3 amp fuses (F1 and F2). The output is terminated with choke input filters (L1 and L2). The outputs are -20, +100, -160 and 0 reference potentials. Sufficient amounts of filter capacitance ($C_1$, $C_2$, $C_3$) shunt every supply, rendering them tolerant of changing load requirements and relatively immune to line voltage transients unless they are of unreasonable proportions.

4.7.5 The variable transformers can be manually adjusted by the control knobs on the front panel, see Figure 36. The filament voltage floats on -55 VDC, which is obtained from variable resistor (R4). D.C. incandescent lamp will light immediately after K-2 has pulled in.

4.7.6 K-5 and K-6 of DA-1 are the interlocking relays for K-3 of G-15. Under normal conditions, both are in 2-3 position (not pulled in) as shown in Figure 41.

K-5 is tied to -20V through R7. If there is a large variation in -20V supply, K-5 will pull in and break the K-3 (G-15) interlock safety circuit and shut off the D.C.

4.7.7 Similarly, K-6 is tied to +100 and -160 volt supplies through R8 and R9. Under normal conditions, terminal 1 of R9 and 2 of R8 are at zero potential. But, if there is an unreasonable fluctuation in +100 or -160 or both supplies, K-6 will pull in and again the interlock circuit will be broken to shut off the D.C. The same thing can also happen if diode clamp package is pulled out without shutting the D.C. off.

4.7.8 A percent reading meter is available to monitor all the potentials. Switches 1 and 2 (Sw 1, Sw 2) are ganged together so as to connect the two -20V supplies in all positions except "-20 Test". The position of the switch shown on the schematic is only for testing the -20V supply of the DA-1, while -20V supply of G-15 is disconnected. For any other check and, also, for normal operation,

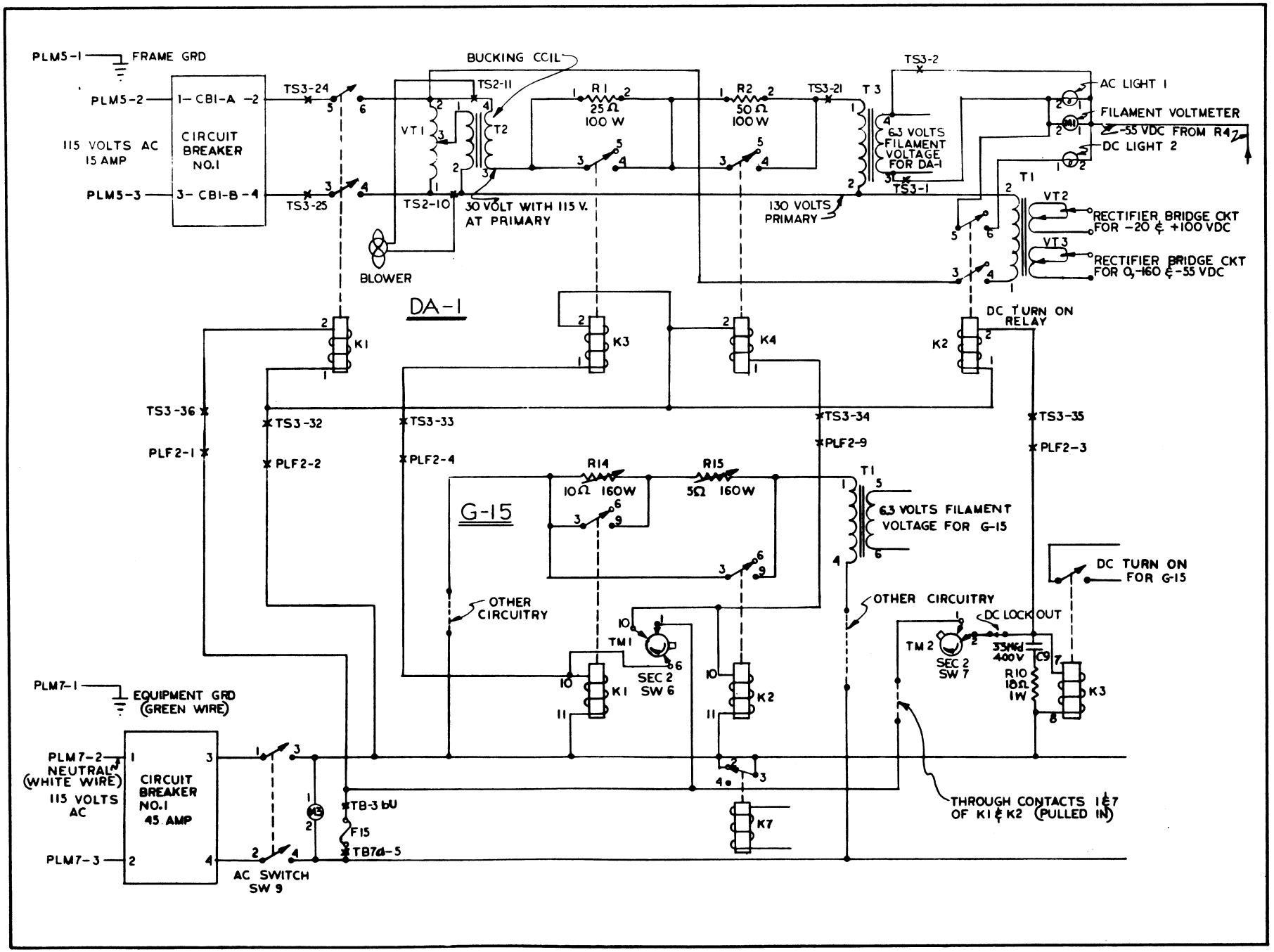



Figure 41. Filament and DC Voltage Relays, G-15 and DA-1

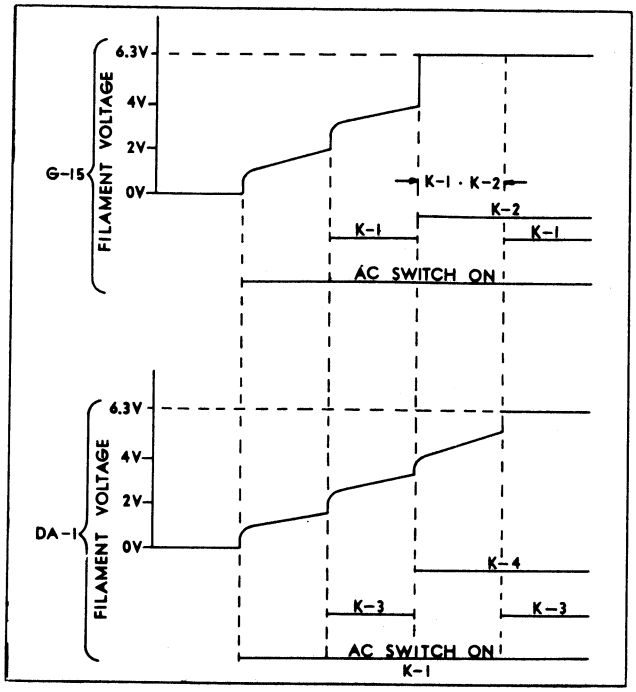



Figure 42. Filament Voltage Incrementation

-20 volts of G-15 is left tied to -20 volts of DA-1 to keep the signals of G-15 and DA-1 at the same level. +250 volts supply from G-15 is also tested with the same switch.

## 4.8 PACKAGES (Reference Figure 43)

There are 144 sockets for different packages. Eighty (80) of them are for diodes and 64 are for tube type packages. Diode and tube package sections are separated by a baffle running longitudinally on the Logic panel. 134 are occupied and 10 are left as spares. The packages list follows:

| PACKAGES | TYPE | HANDLE COLOR | NO. | TOTAL PACKAGES |
|---|---|---|---|---|
| Diode | D-1 | | 64 | 74 |
| | D-C | | 10 | |
| Spare Sockets | | | 6 | |
| Tube Type | C-F-1 | Red | 17 | 60 |
| | C-F-2 | Silver | 13 | |
| | F.F. | Gold | 23 | |
| | B.I. | Blue | 6 | |
| | C.R.(Clk.Rep.) | Black | 1 | |
| Spare Sockets | | | 4 | |
| | | Total | 144 | 134 |

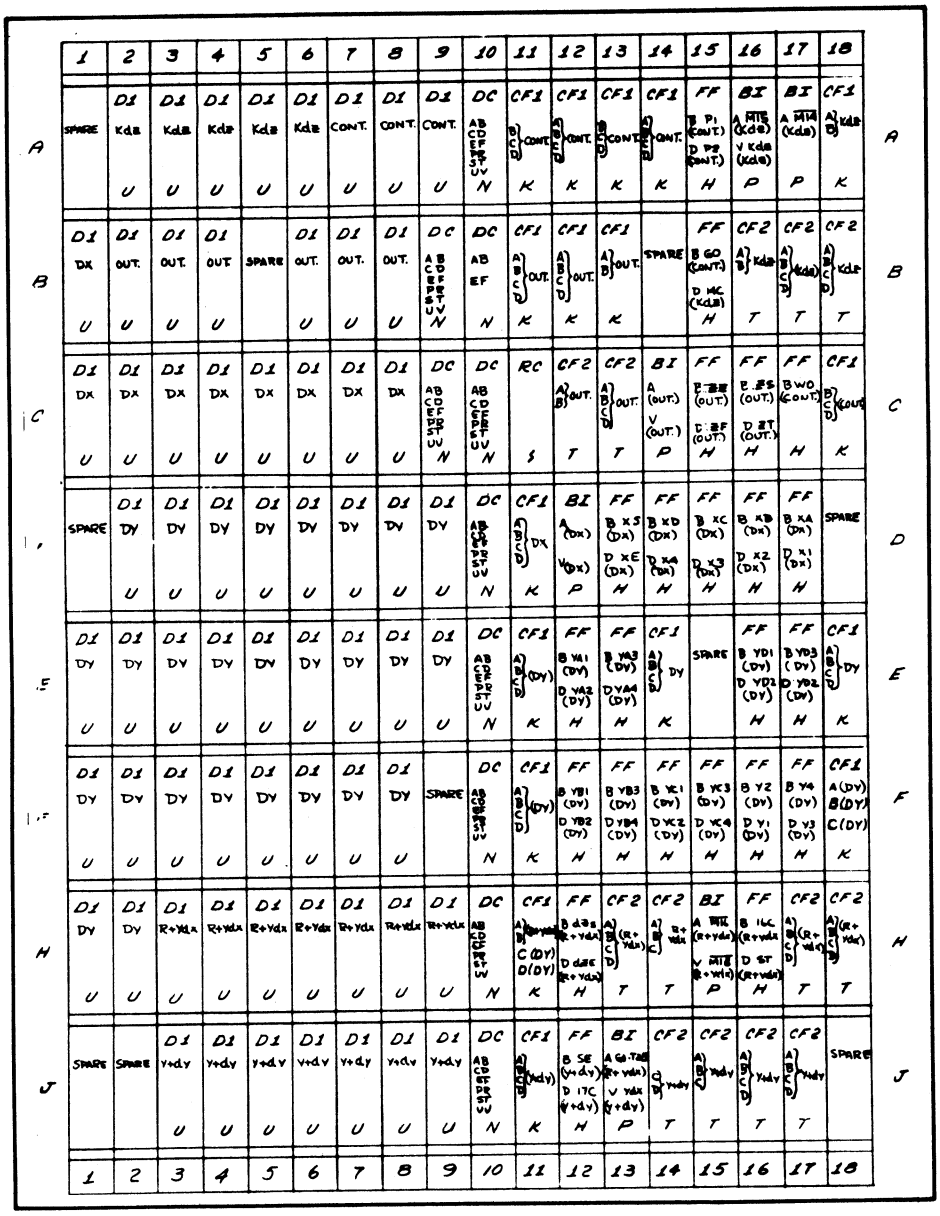


Figure 43. Diagram, Package Location

A package location diagram is provided in Figure 43. All the packages except C.R. (clock repeater) and their circuits are discussed in detail in the G-15 Technical Manual.

4.8.1 CLOCK REPEATER PACKAGE (C.R.)

The clock repeater package (C.R; black handle) is needed in the DA-1 to provide the clock pulse. It is driven from the G-15 "Write Pulse". Reference Figure 44 and 45.

4.8.1.2 WRITE PULSE

4.8.1.2.1 Write Pulse is two micro-second positive going square wave. The pre-amplifier output (a series of sine waves) is squared by V1 and V2 located in the Clock Chassis of G-15 (Reference Figure 46). The V2 output, differentiated, triggers multivibrator V3, which yields a

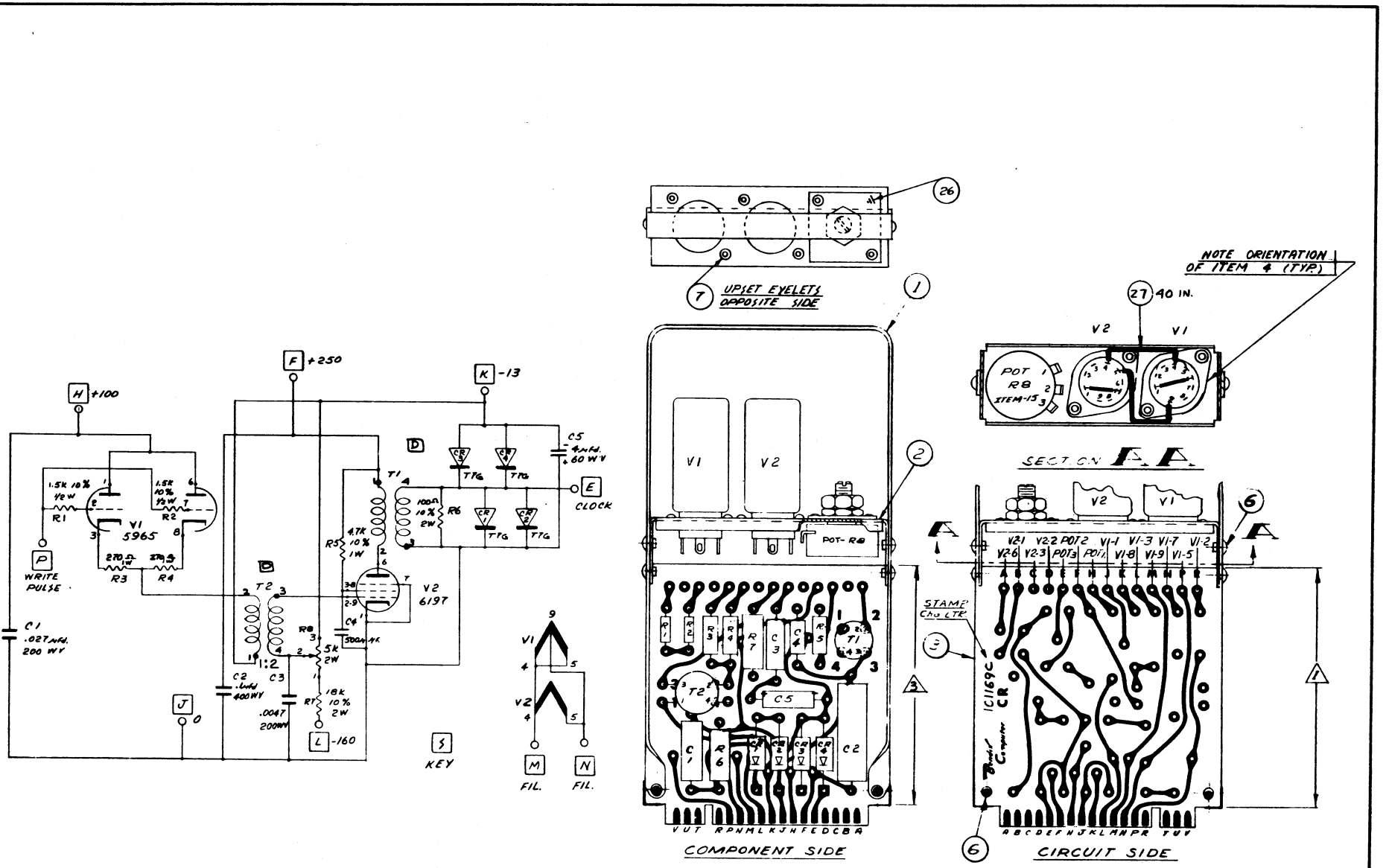



Figure 44. Final Assembly - Clock Repeater Package

2 micro-second positive going square wave at V3, pin 1. This square wave causes power pentode (V7) to conduct through its plate transformer (T3). The negative end of T3's secondary (term. 4) is returned to -20V; the positive end (term. 3; WRITE PULSE) is appropriately terminated to -20V through 680 ohms and clamped at OV. The secondary is terminated by the load and clock clamp packages.

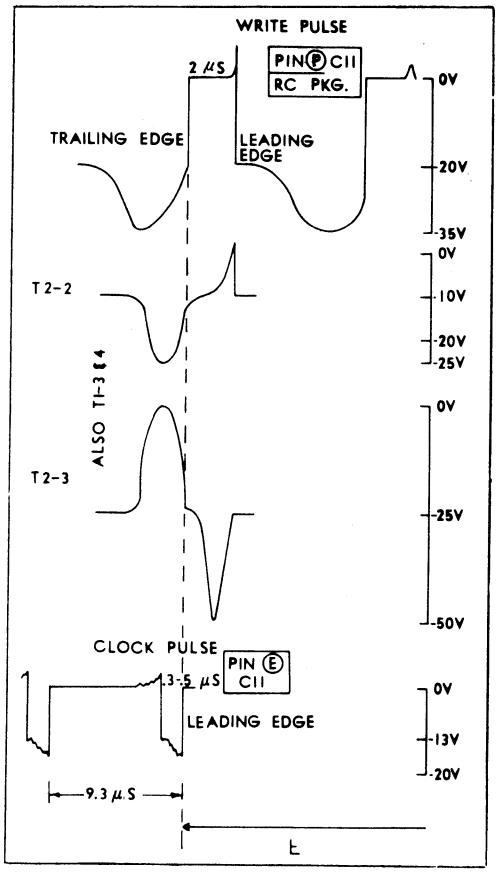



Figure 45. Clock Pulse

4.8.1.2.2 This WRITE PULSE is the input to the (C.R.) clock repeater package at pin P. Both cathode followers (V1) differentiate the pulse which is going positive when the leading edge of the write pulse approaches OV level. The secondary of pulse transformer (T2, term. 3) yields a negative pulse. V2 (power pentode) is biased through the secondary of T2. The plate goes positive and the secondary of T1 (term. 4) stays clamped at OV. The trailing edge of the write pulse yields a positive going pulse at the secondary of T2 (term. 3). This positive going pulse causes V2 to conduct and the plate goes negative. This will give a negative going pulse at the secondary of T1 (term. 4). The output is clamped at -13V and terminated by R6 (100 ohms) to ground.

This .3 to .5 micro-second duration negative pulse has its leading edge right after the trailing edge of the write pulse. The wave shape at different points is shown in Figure 45. The -13 volt amplitude can be varied by the variable resistor (R8). Two consecutive clock pulses are 9.3 micro-seconds apart.

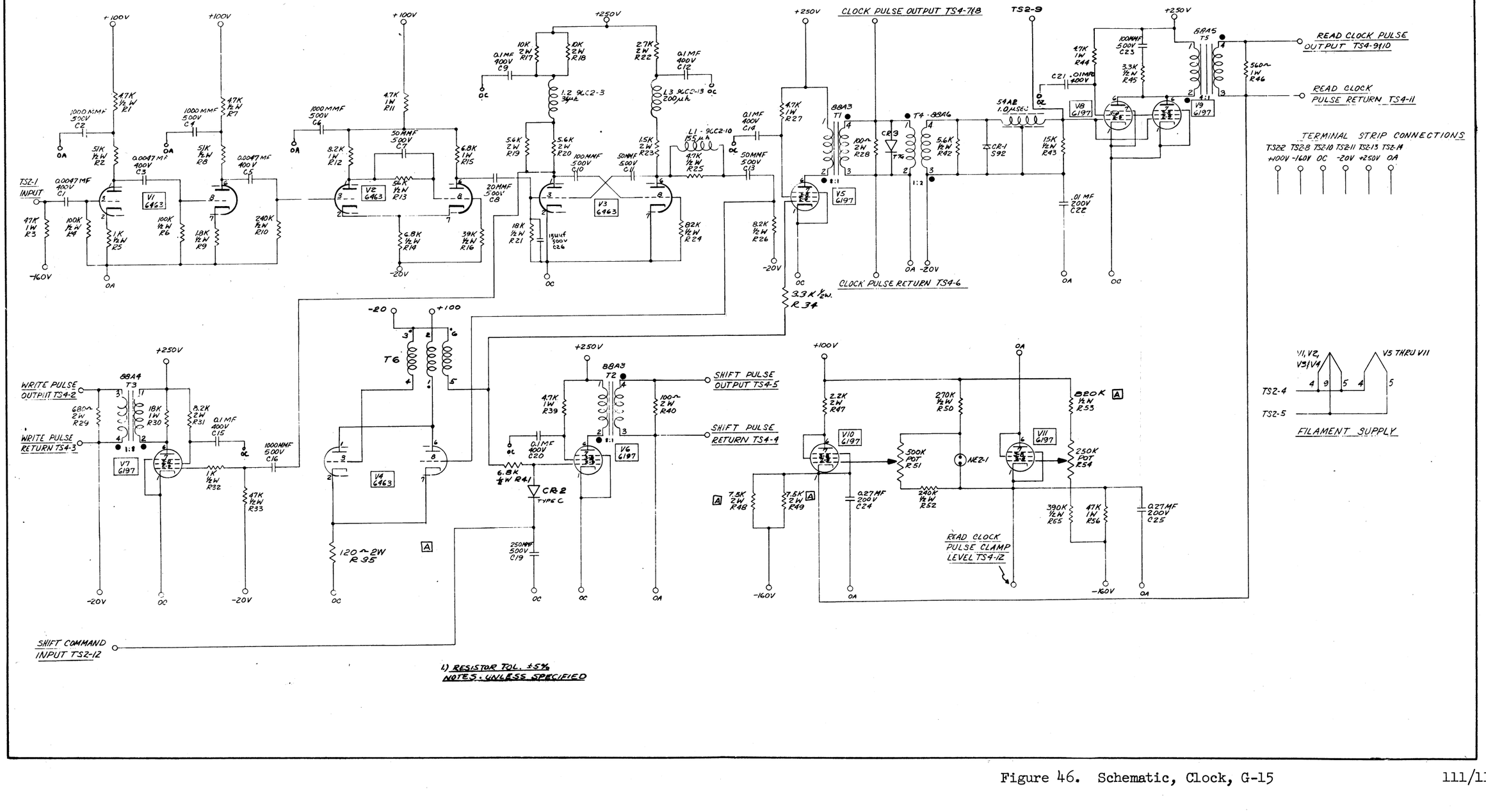



Figure 46. Schematic, Clock, G-15

# SECTION V

# PROGRAMMING

5.0 PROGRAMMING (Reference: Programming Manual)

The subject of programming is much too large to cover in a few short paragraphs, and, indeed, many of the facets must come by experience. Moreover, there is a separate programming manual specifically for DA-1. A DAPPER 2 manual may also be used, since it discusses the operation of the program preparation routine for the DA-1. These two manuals should be available to the reader. However, some brief discussion about programming is presented here.

There are three basic steps in preparing a problem for solution.

(1) Mapping
(2) Scaling
(3) Coding.

5.1 MAPPING (Reference: Programming Manual)

5.1.1 The connections between integrators were briefly discussed in paragraph 2.3.3. Schematic, Block Diagram represents the information flow between integrators. The process of mapping is the drawing of this information flow diagram for all the integrators used in the problem. This information flow diagram will be determined by the particular problem being solved, and the relations which express the operation of various integrators. The following example will give some idea how a second order non-linear differential equation is mapped.

$$\frac{d^2Y}{dx^2} = -\frac{dY}{dx} + Y^2 + \text{Sin } Y + A$$

5.1.2 The highest order derivitive is separated from the rest of the equation. Assume that $\frac{d^2Y}{dx^2}$ is in integrator 25 in order to form d $(\frac{dy}{dx})$. Accumulate $\frac{dy}{dx}$ in integrator 28 and generate dy. Accumulate y in integrator 31 and form $d(y^2)$; integrator 34 and 35 generated (Sin Y). The terms are added together and fed back into integrator 25 to close the loop. The mapping

diagram is given in Figure 47. This problem can also be mapped by other arrangements of integrators.

Bendix Computer

**DA-1**

**INTEGRATOR MAP**

DATE _____ PAGE _____

OF _____

PROBLEM: $\frac{d^2Y}{dx^2} = -\frac{dy}{dx} + y^2 + \sin Y + A$

PROGRAMMER: _____

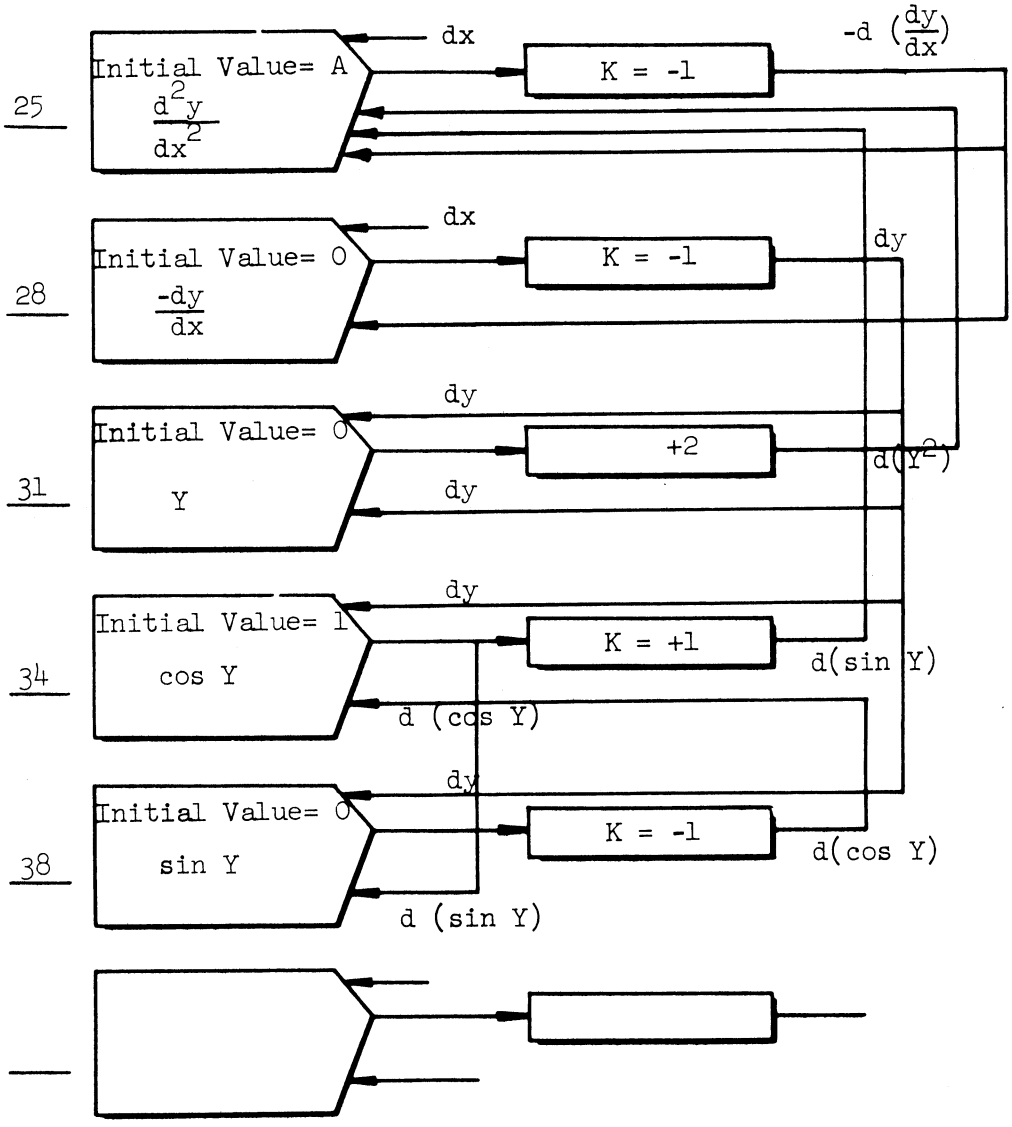



Figure 47. DA-1 Integrator Map

## 5.2 SCALING

In general, scaling consists in assigning mathematical values to each increment of all of the series of increments being transferred from integrator to integrator. For any series of increments the scale value is shown as the exponent of the value of any one increment and is expressed as a power of 2. This scale value is reciprocal of the number of increments which, if accumulated, would give one unit of the variable concerned.

As a result, associated with the dx and the dy inputs to each integrator is a scale factor. Scaling for each individual integrator consists in calculating the scale factor for the dZ output increments. In case of dy increments, the scale factor gives the number of increments necessary to produce a change of one unit in the y value.

There are three binary scale exponents, or scale factors:

Sy is the scale of the integrand and it is determined by the expected maximum value of the integrand Y. Sy is the power of 2 such that $2^{Sy}$ is larger than this maximum value. During computation, if the value of integrand reaches or exceeds $2^{Sy}$, the DA-1 will halt and overflow neon will light.

Example: y variable is the speed of an automobile which under normal conditions is expected to remain less than 100 miles per hour. Since $2^6 = 64$ and $2^7 = 128$, the smallest scale factor (Sy) to be used is 7.

$S_K$ is the power of 2 such that $2^{SK}$ is larger than the constant multiplier.

If k = 296, $S_k = 9$ because $2^9 = 512$ and $2^8 = 256$.

Sdy represents the increments of the Y variable. It specifies the value of a single dy input. Sdy is chosen so that $2^{Sdy}$ is the nearest power of 2 to the desired smallest variation (accuracy). In the automobile case, the speed is required to the nearest tenth of a mile or better. One tenth is between one-eighth and one-sixteenth.

Since 1/16 mph = $2^{-4}$, set Sdy = -4. If an integrator is used as a multiplier constant, make Sdy = Sy-26. All exponents in DA-1 programming are expressed in excess-fifty form. In the preceding examples, therefore, Sy would be written as 57, Sk as 59 and Sdy as 46.

## 5.2.1 RELATIONSHIPS BETWEEN SCALING EXPONENTS

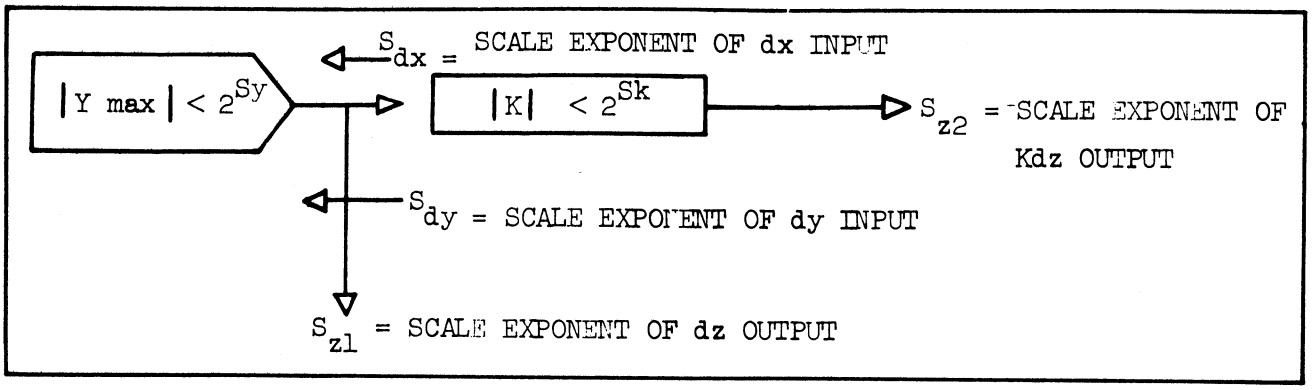



Figure 48. Scaling Exponent Relationships

$$SZ_1 = Sy + S_{dx}$$
$$SZ_2 = S_k + S_y + S_{dx}$$

Approximate the Sy, Sdy and $S_k$ for each integrator and its multiplier. Determine the compatibility of the approximated scale exponents with those of the other integrators. Calculate $S_{z1}$, $S_{Z2}$ as shown above. These exponents $S_{z1}$ and $S_{Z2}$ must be equal to the Sdy or Sdx for the integrator to which they are input. It may be necessary to change some of the estimated scaling in order to obtain compatibility.

## 5.3 CODING

In coding, the scaled flow diagram is transcribed into a set of numerical codes, understandable by the computer. The assignment of integrator numbers, other aspects of coding, the procedure to enter information into the computer, the operational codes for modifying the program, and the error indications during the type-in of the program are all discussed in detail in the Programming Manual, and they will not be repeated in this manual.

# SECTION VI

# MAINTENANCE

## 6.0 MAINTENANCE

The justification for the maintenance of a machine lies in its use to ensure availability and for the performance of its function at the optimum rate. Maintenance of a machine is as important as is its design and manufacturing. The previous discussion has been about the design and logic of the DA-1 so as to enable the reader to be able to maintain or restore it into working order.

## 6.1 PREVENTIVE MAINTENANCE

The general rules about preventive maintenance of the DA-1 are not different from that of G-15. Past experience and statistics prove that hours of uptime vary directly with hours of preventive maintenance performed. Good preventive maintenance consistently performed, can eliminate many costly emergency calls. It is also experienced that a specific procedure should be followed at preventive maintenance time for efficient procedure. The following will be considered standard procedure during each preventive maintenance period.

### 6.1.1 PRIOR TO TURNING ON EQUIPMENT

The preventive maintenance should be scheduled in such a manner that insures getting a "cold" machine at least every other preventive maintenance call. It is necessary to eliminate the warm-up problems.

Consult the User's Log on machine operation for troubles since the last call. If there is no log kept, check with the user. Do not eliminate any portion of the procedure to speed up preventive maintenance for any reason. The portion eliminated may be the emergency call of tomorrow. A quick check of the fuses for proper seating may save a rectifier.

(1) See that the circuit-breaker of DA-1 is ON before turning on the G-15.

(2) Make sure that the A.C. cycle is complete before the D.C. reset button of G-15 is pressed. DA-1 is one step behind the G-15. Watch the A.C. meter on DA-1.

6.1.2 AFTER MACHINE ON

(1) Check the Clock Pulse before running the Test Routine, and turn the 8 toggle switches ON in the back of DA-1.

(2) Read in the DA-1 Test Routine, and check all the tests of the routine with 10% margins.

Plug in the plotter (PA-2 or PA-3) and read in the DAPPER (1A or 2) and the Plotter calibration routines according to the procedure given. Check the plotter with all the margins. This will give a check of the circuitry concerned with the output to the plotter.

These checks should be made as soon as possible after turning ON the machine.

6.1.3 MISCELLANEOUS

(1) Check the air filter and clean or replace, if necessary.

(2) Check the meter accuracy of DA-1 with an accurate voltmeter.

(3) Put the dummy plug in PLF-16 after disconnecting the DA-1 from the G-15.

6.2 POWER SUPPLY CHECK (Reference: Figure 35, 41 and 42).

Power Supply usually does not give much trouble in the field, if it is once thoroughly checked in the Systems Test. If it does give more than usual trouble, the following procedure may be employed to give it a thorough check. This check is usually employed in the Systems Test before all the packages are plugged in.

6.2.1 PROCEDURE

(1) Remove both fuses F1, F2 and pull out the D.C. interlock relays K-5 and K-6. Connect pin K5-3 to K6-3 and A10L to J10M by using jumpers. Take all the packages out and with PLF-1 out, PLF-2 in, and PLM-5 out, check the A.C. turn-on cycle as follows:

K-1 will pull in immediately when A.C. of G-15 is turned ON.

Then, K-3 will also pull in, i.e. (K-1•K-3).

Then, K-4 will pull in and K-3 will drop out, i.e. (K-1•$\overline{K\text{-}3}$•K-4)

Then, K-3 will also pull in again, i.e. (K-1•K-3•K-4).

(The above relays are controlled by the timing motor TM-1 of G-15).

Check the simplified schematic, Figure 41, to see the connections between G-15 and DA-1 power relays.

(2) Follow the same procedure as described in "(1)" after plugging in PLF-5 and turning on the Circuit Breaker -1 (CB-1).

(3) Same as "(2)", but with D.C. on. Check the secondary voltages of VT-2 and VT-3. The relays K-1, K-3, K-4 and K-2 will be pulled in.

(4) Same as "(3)", except K5-3 to K6-3 jumper is removed. Fuses F-1 and F-2 and relays K-5 and K-6 are plugged back in.

(5) Remove jumper (A1OL to J1OM) and visually inspect and install the packages according to the package location, see Figure 43. Plug in PLF-1.

(6) Check -55 volt D.C. filament reference voltage and -20 volts D.C., and adjust, if necessary.

(7) Check all the voltages on the percent voltmeter by using the voltage selector switch. The switch should never be left at -20V test position. This position is only for checking the -20 volts supply of DA-1 while the -20 volts from G-15 is disconnected.

## 6.3 LOGIC CHECK

The logic circuitry of DA-1 is checked by the DA-1 Test Routine. The new DA-1 Test Routine is almost the same, except a few modifications are incorporated to check the logic more thoroughly. It is pointed out that the cathode followers in G-15, the outputs of which are brought to the DA-1, are not checked with Test Routine 1 & 2 of G-15. So, it will be of great convenience if a quick check of the output signals of these cathode followers be made first.

The check list of the plug pins, signals they carry, and the source

of the signals from the G-15, is given in Figure 39 and 40 in the form of a continuity chart.

Before running the Test Routine, check the Clock Pulse, and turn on the 8 toggle switches in the back of the DA-1 to connect the memory lines to the DA-1.

# SECTION VII

# TEST ROUTINE

7.0 TEST ROUTINE

7.1 INTRODUCTION (Reference: Figure 49).

A total number of 45 integrators are used by this test routine, (00 through 46, with the exception of integrator number 37 and 38). The first 16 integrators are used to develop 8 negative KdZ outputs (integrators 01 - 08), and 8 positive KdZ outputs (integrators 09 - 15, and 00). In developing these 16 KdZ outputs, the arithmetic circuits are checked for their ability to properly handle all possible combinations of positive and negative dx inputs, initial y values, all possible combinations of dZ outputs and constant multiplier K values. Integrators 16 through 29 are used to check the dy counters and registers, and the ability of the arithmetic circuits to handle all possible combinations of positive and negative values of y and dy inputs. The next 6 integrators are used to check a programmed halt and the ability of integrators to function properly as servo or decision integrators.

Integrator 35 is a special case, where the initial values of y is 0 and as such, there would never be a dz output. Integrators 39, 42, 44 and 45 give negative outputs, while integrators 40, 41, 43 and 46 have positive outputs. The outputs of the following pairs of integrators go to the dy inputs of integrator 26 through 29:

39 and 40 -- 26
41 and 42 -- 27
43 and 44 -- 28
45 and 46 -- 29

Integrator 39 through 42 do not yield a pulse every drum cycle. Integrators 39 and 40 check the value of K, and integrator 41 and 42 check the y register when it is less than full. Integrators 43 and 44 check the -dx, -dy, and -y, while integrators 45 and 46 check -dx, -dy and +y. The ability of the DA-1 to detect and halt as a result of an overflow in a Y register is checked with integrator 36. Integrators 37 and 38 are left out purposely to show the freedom of choice of integrators. Numbers appearing in circles on the integrator map are the binary scale factors.

Bendix Computer

# DA-1
## INTEGRATOR MAP

DATE _____ PAGE 1 OF 6

PROBLEM: DA-1 Test

PROGRAMMER: Thad Lee

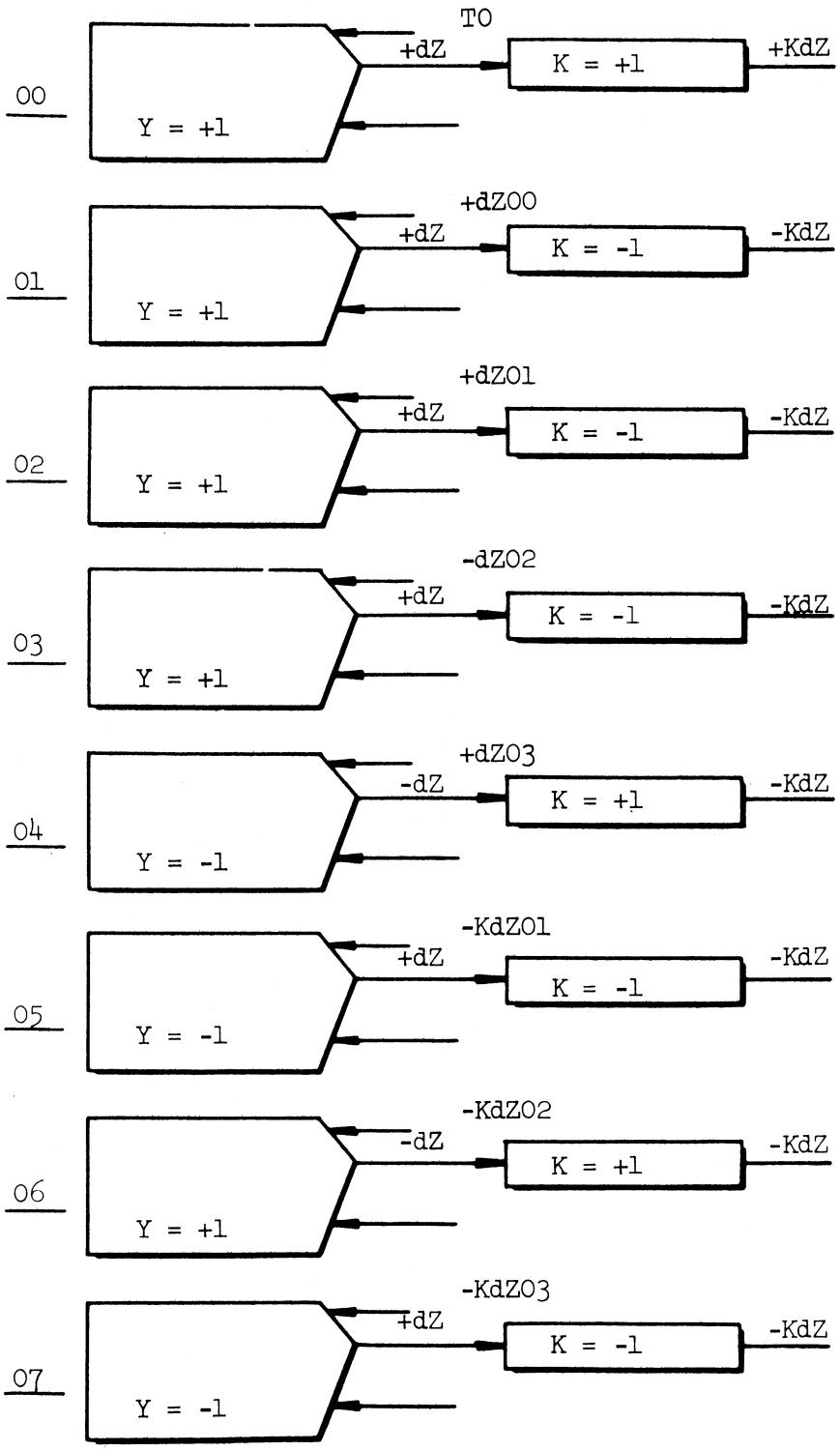



Figure 49. DA-1 Test Routine

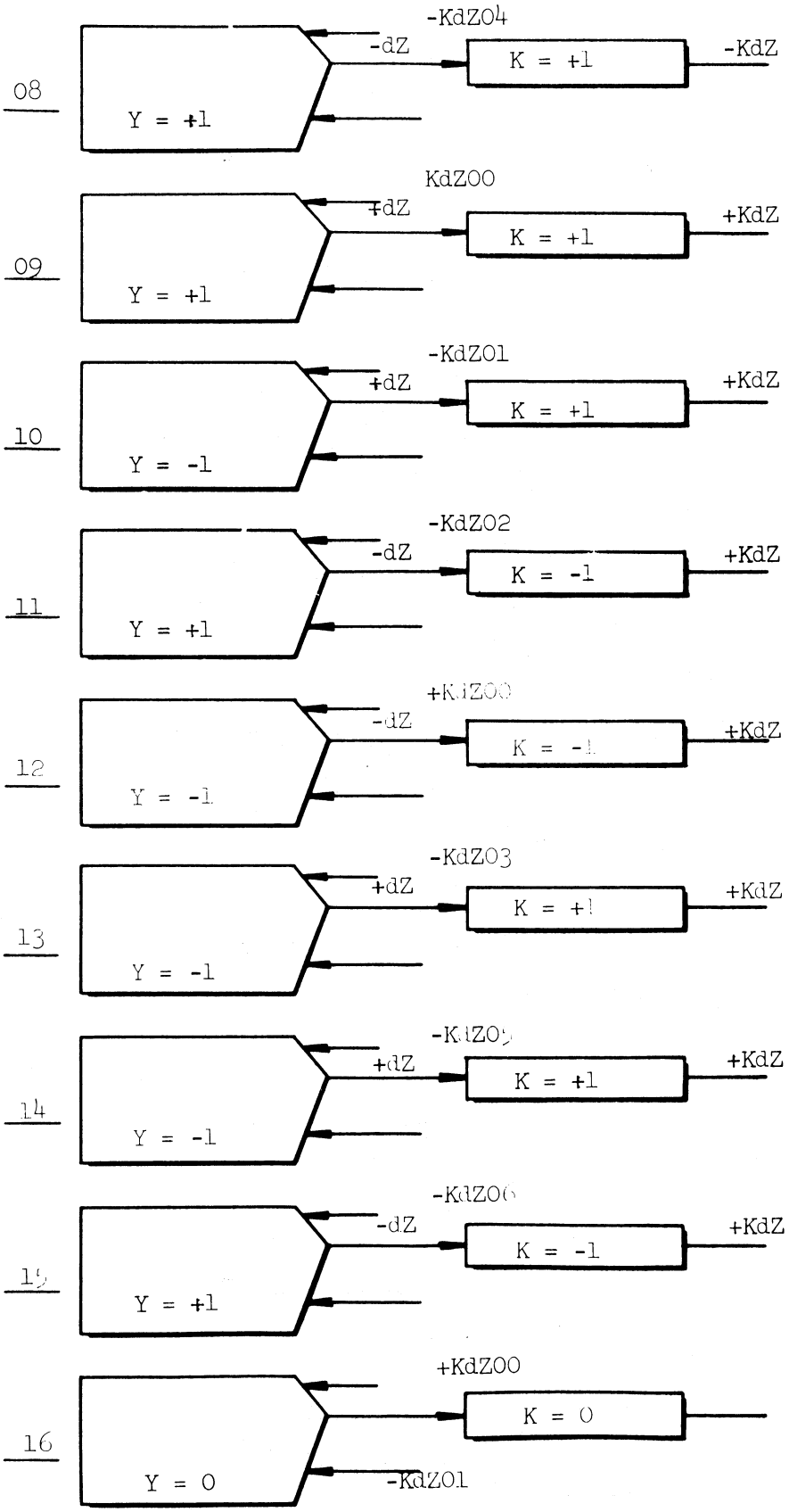
Bendix Computer

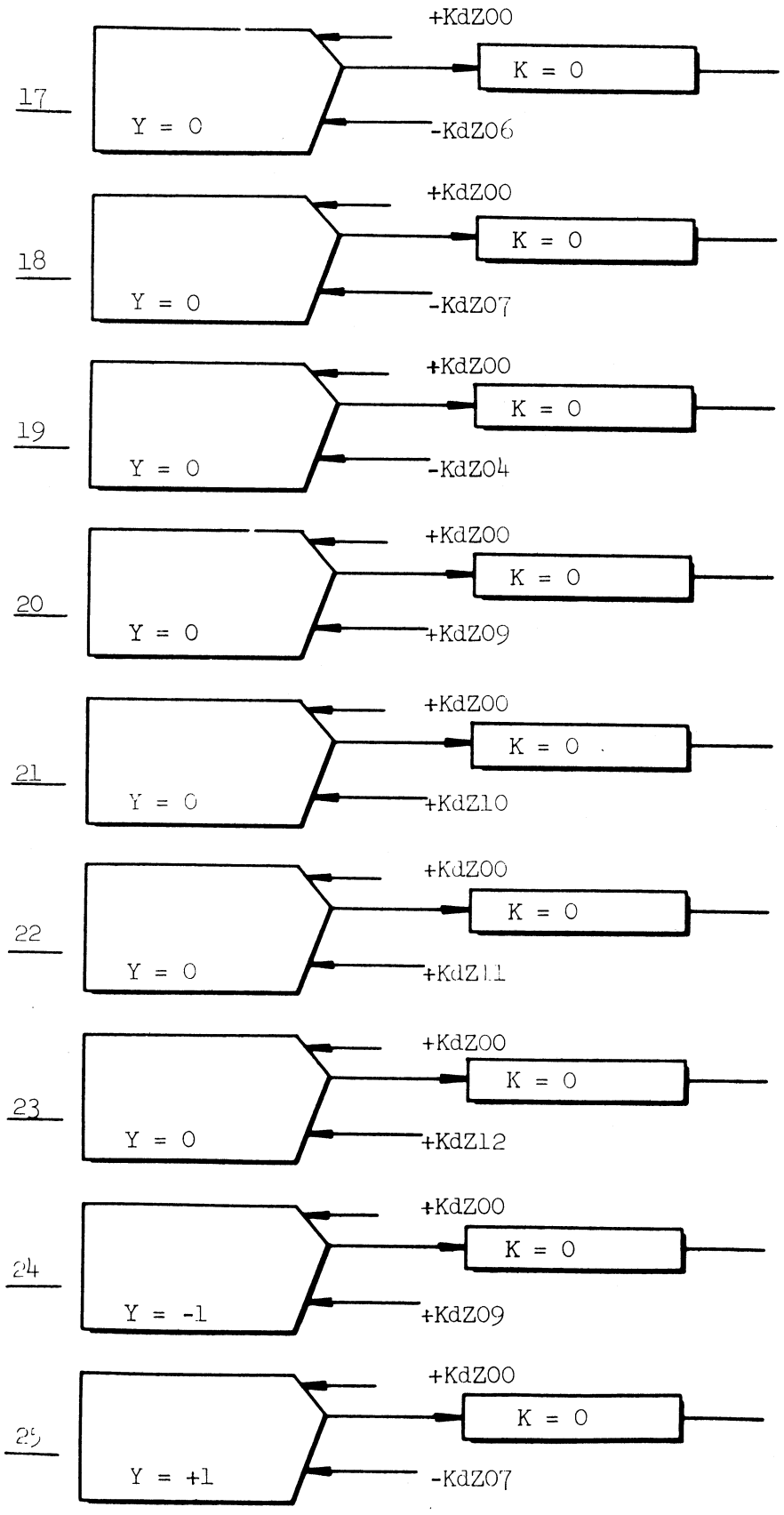
# DA-1
## INTEGRATOR MAP

DATE _____ PAGE 2 OF 6

PROBLEM: DA-1 Test

PROGRAMMER: _____




17
Y = 0
+KdZ00
K = 0
-KdZ06
18
Y = 0
+KdZ00
K = 0
-KdZ07
19
Y = 0
+KdZ00
K = 0
-KdZ04
20
Y = 0
+KdZ00
K = 0
+KdZ09
21
Y = 0
+KdZ00
K = 0
+KdZ10
22
Y = 0
+KdZ00
K = 0
+KdZ11
23
Y = 0
+KdZ00
K = 0
+KdZ12
24
Y = -1
+KdZ00
K = 0
+KdZ09
25
Y = +1
+KdZ00
K = 0
-KdZ07

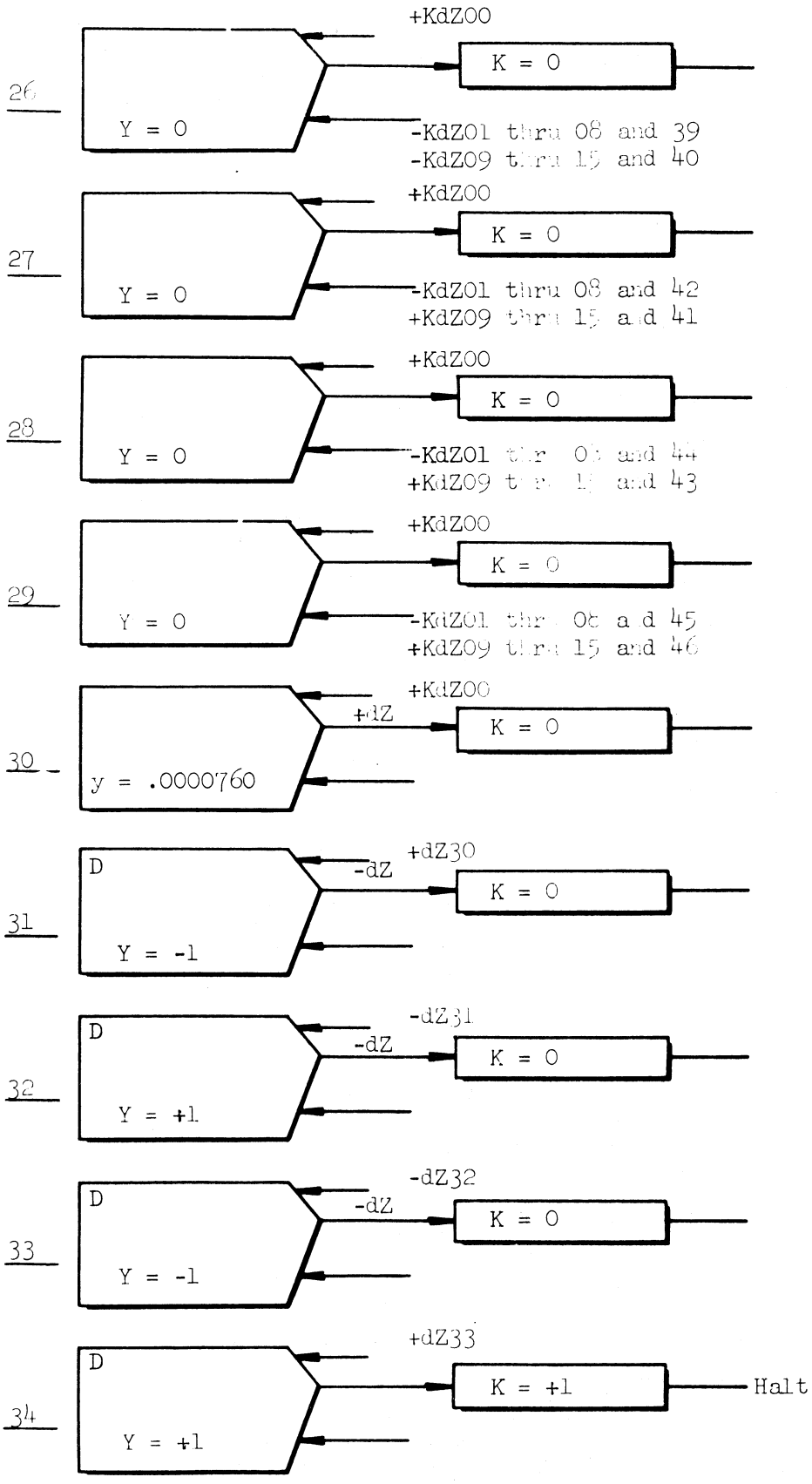
Bendix Computer

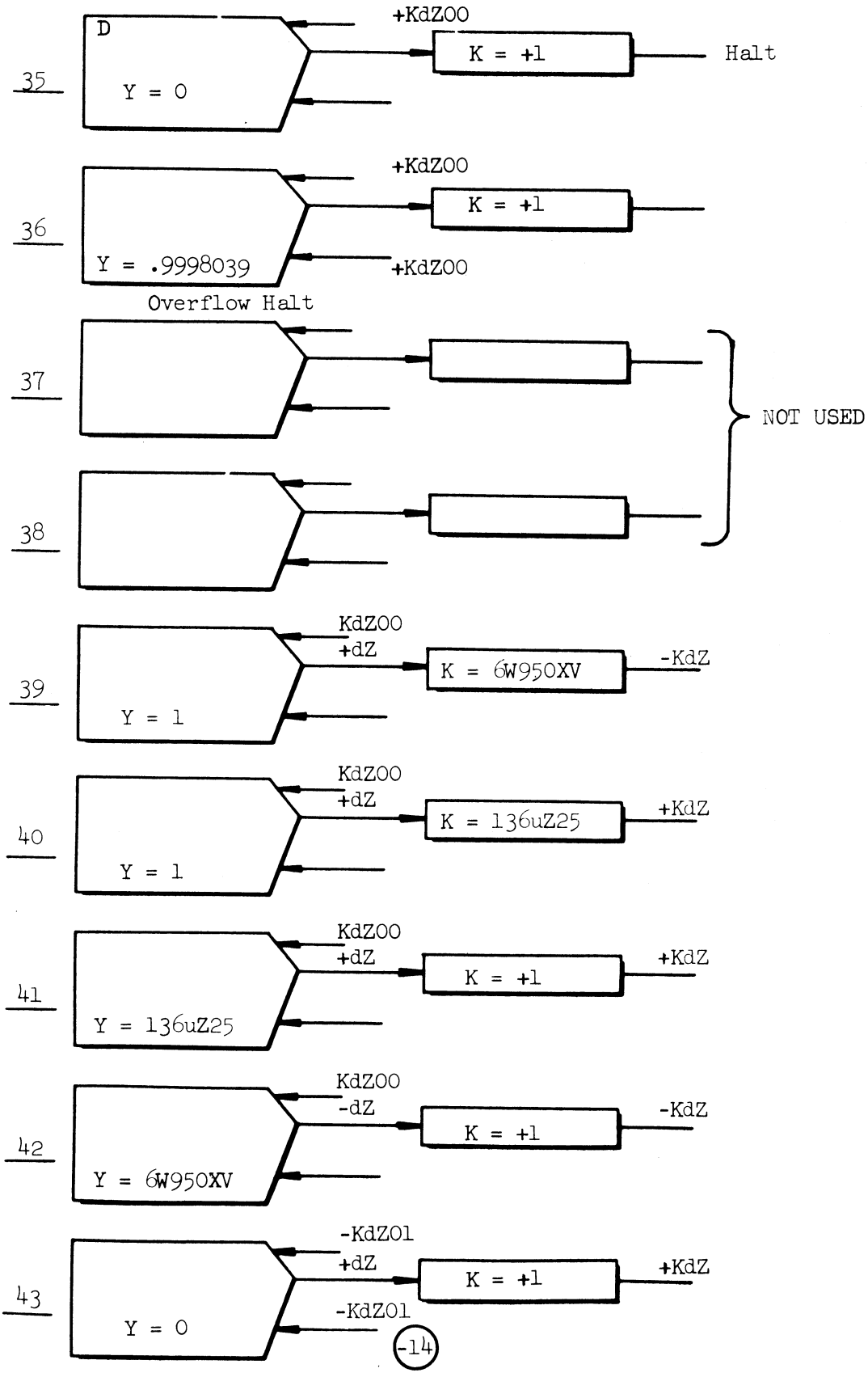
# DA-1 INTEGRATOR MAP

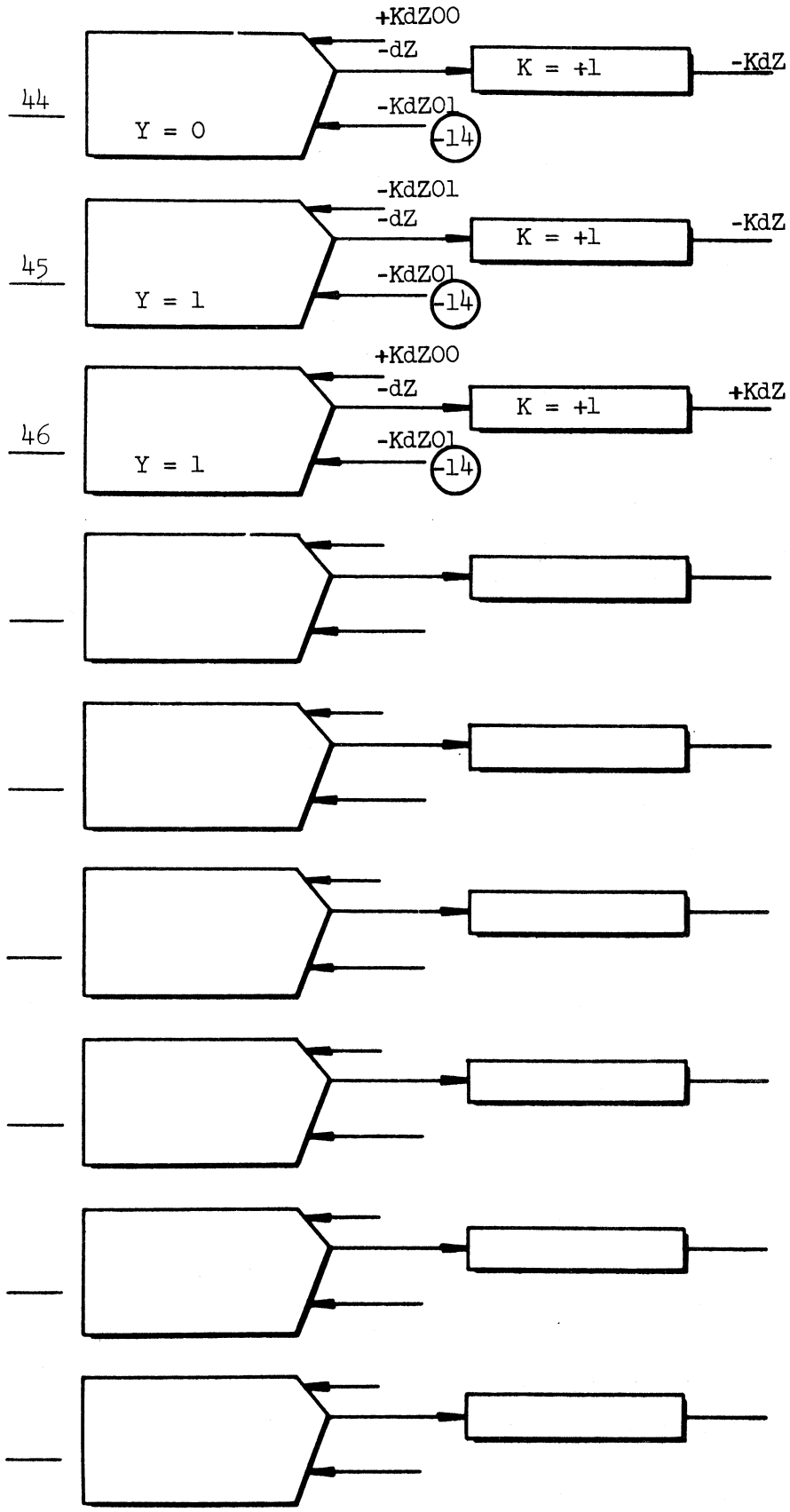
DATE _____ PAGE 4 OF 6

PROBLEM: DA-1 Test

PROGRAMMER: _____




35
D
Y = 0
+KdZ00
K = +1
Halt
36
Y = .9998039
+KdZ00
K = +1
+KdZ00
Overflow Halt
37
38
NOT USED
39
Y = 1
KdZ00
+dZ
K = 6W950XV
-KdZ
40
Y = 1
KdZ00
+dZ
K = 136uZ25
+KdZ
41
Y = 136uZ25
KdZ00
+dZ
K = +1
+KdZ
42
Y = 6W950XV
KdZ00
-dZ
K = +1
-KdZ
43
Y = 0
-KdZ01
+dZ
K = +1
+KdZ
-KdZ01
-14


+KdZOO
-dZ
K = +1
-KdZ
44
Y = 0
-KdZO1
-14
-KdZO1
-dZ
K = +1
-KdZ
45
Y = 1
-KdZO1
-14
+KdZOO
-dZ
K = +1
+KdZ
46
Y = 1
-KdZO1
-14

The outputs of I-34 and I-35 could not both be sent to the dx of I-00 with any of the **DAPPER** routines.

## 7.2 OPERATING INSTRUCTIONS

It is assumed that the G-15 has been checked for proper operation prior to the use of this test. The first block of this test routine is the number track (.147 x 86w), and the second is the loader (-.WW7y4zy). Read in these first two blocks with P key. Move the compute switch to Go. This will load lines 6 through 18, beginning with line 18, and will type out the check sums as follows:

-.20037WO
-.3zzy05u
-.6000000
.zzzzzwu
-.8000000
-.840031y
-.84002vu
-.860026w
-.840017w
.0020382
.0000302
.0000322
.0000282

When the last block has been read in, and check sum typed out, a sequence of operations will be executed which will clear the dx and dy counters and registers to prepare for the DA-1 operation.

If at this time xxxxxxx is typed out and the computer halts, it indicates that the DA-1 did not obey a stop DA-1 command from the G-15. This trouble must be cleared before the program can be continued. Take the compute switch out of Go to idle, and then back to Go. It will restart the DA-1 clearing sequence, and again check for the proper operation of the stop DA-1 command.

Initial values of Y (M17) and K (M15) are also stored in lines 5 and 4 respectively. The clearing sequence will end with a gate type-in ready configuration. At this point there is a choice of three modes of operation depending upon whether 1 tab (S), -1 tab (S), or 0 tab (S) is typed in.

## 7.3 NORMAL TEST

Type in 1 tab (S).

This will start the DA-1 and allow it to run for approximately six and one-half minutes, after which DA-1 will halt and the contents of y registers of integrators 16 through 29 will be typed out in sequence. If the test was successful, the following typeout will occur:

| | | |
|---|---|---|
| .7zzww99 | For I-16 | These 4 should be the same for correct typeout. |
| .7zzww99 | " I-17 | |
| .7zzww99 | " I-18 | |
| .7zzww99 | " I-19 | |
| .0003367 | For I-20 | These 4 should be the same for correct typeout. |
| .0003367 | " I-21 | |
| .0003367 | " I-22 | |
| .0003367 | " I-23 | |
| .4003368 | For I-24 | |
| .3zzww98 | For I-25 | |
| .7zzww99 | For I-26 | These 4 should be the same for correct typeout. |
| .7zzww99 | " I-27 | |
| .7zzww99 | " I-28 | |
| .7zzww99 | " I-29 | |

After the typeout has been completed, the computer will halt. Now test for an overflow in the y register by taking the compute switch out of Go and back to Go.

If this test is successful, a .0000002 will be typed out. To restart the test, the compute switch is turned back to idle, and then to Go again. This will clear the DA-1, set up the initial conditions, and

gate the typein so that either 1 tab (S), -1 tab (S) or 0 tab (S) may be typed in.

If the typeouts of I-28 and I-29 are .7zzww93 instead of .7zzww99, check the y and dy adder.

If the typeout of I-26 is .7zzv2y6 instead of .7zzww99, check the Zi gate and r + KdZ adder.

## 7.4 SHORT TEST

Type in -1 tab (S):

This will start the DA-1 and allow it to run for approximately 15 seconds. DA-1 will halt and the contents of integrators 16 through 29 will be typed out in sequence. This test is useful in checking margins. If the test is successful, the following typeout will occur:

| | | |
|---|---|---|
| .7zzzy00 | For I-16 | These 4 should be the same for correct typeout. |
| .7zzzy00 | " I-17 | |
| .7zzzy00 | " I-18 | |
| .7zzzy00 | " I-19 | |
| .0000200 | For I-20 | These 4 should be the same for correct typeout. |
| .0000200 | " I-21 | |
| .0000200 | " I-22 | |
| .0000200 | " I-23 | |
| .4000201 | For I-24 | |
| .3zzzxzz | For I-25 | |
| .7zzzy00 | For I-26 | These 4 should be the same for correct typeout. |
| .7zzzy00 | " I-27 | |
| .7zzzy00 | " I-28 | |
| .7zzzy00 | " I-29 | |

The computer will halt after the typeout. Now, the overflow in the Y register can be tested by moving the compute switch to idle and back to Go. If this test is successful, .0000002 will be typed out immediately following the movement of the compute switch to Go. To restart the test, the compute switch is moved back to idle and back

to Go again. This will clear the DA-1, set up the initial conditions, and gate the typein. It is ready again for one of the three tests (1 tab (S), -1 tab (S), or 0 tab (S)).

If -1 tab S test is repeated a number of times, the correct typeout may change slightly due to the fact that the R and the r registers are not reset to their initial values. However, the form of typeout for each group should be the same. It is pointed out that there is some circuitry which is not checked by this 15 seconds test unless it is repeated five or six times. If there is something wrong in the y and dy adder, the typeout for integrator 28 or 29 will be different (maybe by unity) from the I-26 and I-27.

If r and KdZ adder is malfunctioning, the typeout for I-26 will be .7zzzx01 instead of .7zzzy00.

## 7.5 UNLIMITED TEST

Typein 0, tab (S).

This starts the DA-1 and allows it to run almost indefinitely. This test is used primarily for troubleshooting. The DA-1 may be stopped at any time by moving the compute switch to BP and the test can be restarted by moving the switch from BP to Go. This will clear the DA-1, set up initial conditions, and gate the typein so that any one of the three tests can be performed.

The commands in this routine are executed from line 0. At any time after the DA-1 is off, a typeout of the contents of the y registers of integrators 16 through 29 may be accomplished by manually entering line 0 at word time 41. The DA-1 may be cleared, initial conditions set, and typing gated for a restart of the test by entering into line 0 at word 22.

## 7.6 PA-2 and PA-3 PLOTTER CALIBRATION TEST ROUTINES

This program assumes that the DA-1 is connected to the G-15 and that the DA-1 Test Routine has been run successfully. It, also, assumes

that the **PA**-2 or **PA**-3 is plugged into the **DA**-1, turned on, and the pen is down.

Step 1: Load the **DAPPER** 1-A or -2 routine as follows:

The first block of **DAPPER** 1-A is the Number Track for the G-15. With enable switch on, hit "**P**" key P to read in one block, then read in another one. **DAPPER**-2 does not have the Number Track, so read in just one block with P. For **DAPPER**-2 it is assumed that the number track is already in the G-15.

**A**fter this for both routines, turn off the enable switch and set the compute switch to **GO**. A few blocks of tape will be read in. The photo reader light goes off and the computer assumes **TYPE IN** configuration on the neon panel.

This program calls for all the integrators, so type in "(4 tab (S))" with compute switch still in **GO**. 108 will be typed out and a few more blocks of tape will be read in. The computer will assume **TYPE IN** configuration again. **A**t this stage, new program can be typed in or read in from the prepared tape or both.

Step 2: Remove **DAPPER** magazine (do not rewind) and mount the ***Plotter** Calibration Routine on the photo reader. Take the compute switch out of **GO** and type P (Enable **ON**). When the photo reader light turns off, put the compute switch to **GO**. The whole routine will be read in.

Step 3: When the photo reader light turns off, rewind the tape and remove the magazine. Replace **DAPPER** magazine in the same condition as it was removed.

Step 4: The computer is in the **TYPE IN** configuration. To check the **PA**-2 (not **PA**-3) at the lower speed, type in the following information leaving the compute switch to **GO**.

50.5000000 tab 5000 2 tab (S).

* If no Plotter Calibration Routine tape is available, the Plotter Calibration Routine program may be typed in place of Step 2 and 3.

.2000000 2 will be typed out.

This step is to check **PA-2**. It will work on PA-3 also, but at a slower speed than **PA-3** is made for.

Step 5: Type Y Tab (S). Y will be typed out and carriage will return.

Step 6: Now take the compute switch out of GO and turn to BP.

Step 7: Type X - tab (S). The photo reader will read a few blocks of the **DAPPER** routine and turns off.

The plotter will take off and draw a diagonal line for 2 inch square and then 1 inch square, and then it will continue and draw 2 inch square as shown in Figure 50. The 2 inch square will be retraced indefinitely. The pen will follow the arrows marked with serial numbers from 1 to 12.

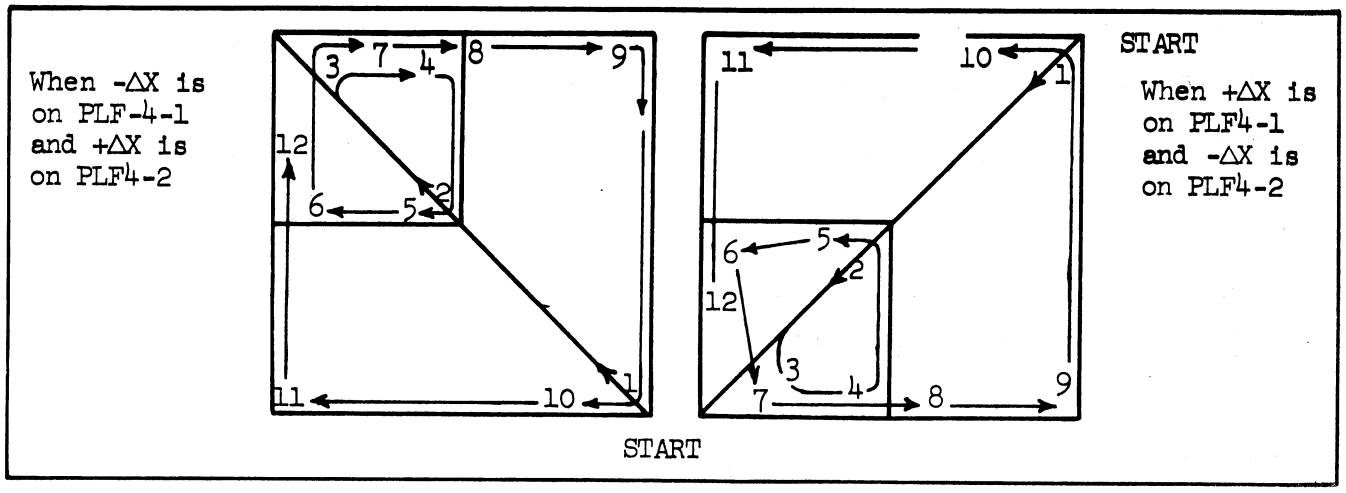



Figure 50. PA-2 and PA-3 Calibration Plot

Step 8: To stop the plotter, move the compute switch to **OFF**, type scf and back to BP or GO. To start it again from the same point, type x-tab (S) with compute in BP. To start it again from the beginning, type Y tab (S) with compute switch to GO and then x-tab (S) with compute switch to BP.

## 7.7 DA-1 DEBUGGING

DA-1 debugging is usually done with the "DA-1 Test Routine" discussed in the previous section. A punched tape copy of this test routine should be in the Maintenance Kit of every G-15 that is coupled to the DA-1. The write up of this test, including operating instructions, test type-outs, integrator map, and coding sheets have been discussed in the previous section.

### 7.7.1 FIRST METHOD

Working with the DA-1 Test can be time consuming and patience depleting if there is an error that stops the test almost as soon as it is started. You have to wait for fourteen or fifteen words to type out for every time you look at a point. To avoid this confusion, use the following program along with the test routine.

Step 1: Fully load the DA-1 Test and have the computer setting on the gate type in, waiting for -1, 1 or 0 (tab) (S) tests.

Step 2: Take out of GO ($\overline{GO}$) and with enable on, type in (S) C7q. Now, type in the following program for 15 seconds test, i.e. (-1 tab (S)):

-0000001 (tab) 242501w (tab) 0000000 (tab)
-U2012Y0 (tab) FI w0002y0 (tab) 0028000 (tab)
I x01601w (tab) U9162y0 (tab) I (S) C GO.

This will cause the DA-1 Test to operate repetitiously without gating typein or typeout. The above example is for the 15 seconds test, i.e. -1 (tab) (S) . Changing the first word of this program typed in from -0000001 to 0000001 or 0000000 will operate the 6 1/2 minute, i.e. 1 (tab) (S) or the indefinite 0 (tab) (S) test respectively. This test is prepared because of its brevity and its frequent start with known fresh data.

### 7.7.2 SECOND METHOD

Clipleading the reset term of the GO flip-flop to -20 VDC is another

quick and dangerous method that can be used with success. It is dangerous because -20 VDC is located on pin K of every diode clamp package, but inconveniently bracketed by ground on pin J and 115 VAC (interlock) on pin L. This method keeps using the same data over and over, right or wrong. It is quick and easy to remember, but does not lend itself to a close examination of the data in process.

## 7.8 SYNC PULSE

### 7.8.1 FIRST METHOD

Defining an iteration as a drum cycle and an integrator as a word time or location, it is frequently required to look at an integrator or word time located one third or one fourth of the way up in a drum cycle. Clearing line 19 completely and typing in a solitary pulse (usually a T29 of the word just before the one of interest) will give a good scope sync. To look at word 73 for instance, put the Sync Pulse in T29 of word 72 of line 19. To make sure that the Sync Pulse is in the right place on line 19, look at the Number Track on the test panel and see: w u 4 u 0 0 0 for word 73, i.e. in the binary form:

1100, 1010, 0100, 1010, 0000, 0000, 0000
W U 4 U 0 0 0

Remember that in all words of the number track except 107, T and N is equal to the location plus one. Syncing on a solitary pulse in line 19 is an old G-15 technique, but it is repeated here because of its usefulness in looking at an integrator.

### 7.8.2 SECOND METHOD

There is another method of syncing the scope for looking at the desired integrator. Type a deferred command into line 23 and sync the scope on the (TR) transfer jack on the test panel of G-15. For example, if word 39 of line 0 is in question, type in the following command:

(S) C7q U700000 (tab) (S) and compute to GO.
Sync the scope on TR jack and probe the circuitry.

Bendix Computer — Los Angeles 45, California

G-15 D
PROGRAM PROBLEM: SYNC PULSE ROUTINE

Prepared by _____

Page 1 of 1
Date: 10-10-60
Line 05

| | | | |
|---|---|---|---|
| ✕ | ✕ | 2 | ✕ |
| ✕ | ✕ | ✕ | ✕ |
| ✕ | ✕ | 10 | 11 |
| ✕ | ✕ | ✕ | ✕ |
| 16 | ✕ | ✕ | ✕ |
| 20 | ✕ | ✕ | ✕ |
| 24 | 25 | 26 | 27 |
| 28 | 29 | 30 | 31 |
| 32 | 33 | 34 | 35 |
| 36 | 37 | 38 | 39 |
| 40 | 41 | 42 | 43 |
| 44 | 45 | 46 | 47 |
| 48 | 49 | 50 | 51 |
| 52 | 53 | 54 | 55 |
| 56 | 57 | 58 | 59 |
| 60 | 61 | 62 | 63 |
| 64 | 65 | 66 | 67 |
| 68 | 69 | 70 | 71 |
| 72 | 73 | 74 | 75 |
| 76 | 77 | 78 | 79 |
| 80 | 81 | 82 | 83 |
| 84 | 85 | 86 | 87 |
| 88 | 89 | 90 | 91 |
| 92 | 93 | 94 | 95 |
| 96 | 97 | 98 | 99 |
| u0 | u1 | u2 | u3 |

| L | P | T or $L_k$ | N | C | S | D | BP | NOTES |
|---|---|---|---|---|---|---|---|---|
| 00 | | 02 | 05 | 0 | 12 | 31 | | Gate type-in= |
| 01 | U | 06 | 03 | 0 | 19 | 05 | | Line 19 → line $05^1$ |
| 03 | | 05 | 05 | $5^1$ | 21 | 31 | | N.C. $05^1$.05 |
| 05 | U | 10 | 08 | 0 | 29 | 23 | | Clear line 23 |
| 07 | U | 08 | 18 | 0 | 29 | 19 | | Clear line 19 |
| 08 | | 02 | 04 | 0 | 05 | 02 | | Extractor → 02.02 |
| 02 | | | | | | | | 00000Z0 |
| 04 | | 10 | 12 | 4 | $05^1$ | 24 | | Markers → MQ |
| 10 | | | | | | | | 8000000 |
| 11 | | | | | | | | 8000000 |
| 12 | | 14 | 12 | 0 | 28 | 31 | | Ready? |
| 13 | | 16 | 07 | 4 | 23 | 26 | | 23.0,1 → PN |
| 18 | W | 02 | 06 | 3 | 23 | 31 | | Extract |
| 06 | | 08 | 09 | 0 | 26 | 30 | | $PN_o$ left 1 |
| 09 | | 12 | 15 | 0 | 25 | 30 | | $ID_o \rightarrow PN_{o+}$ |
| 15 | U | 19 | 19 | 4 | 26 | 30 | | $PN_o$ left 2 |
| 19 | | 20 | 21 | 0 | 25 | 30 | | $ID_o \rightarrow PN_{o+}$ |
| 21 | | 22 | 23 | 0 | 26 | 28 | | $PN_o \rightarrow AR_c$ |
| 23 | U | 41 | 14 | 2 | 28 | 29 | | AR left 17 |
| 14 | | 16 | 17 | 3 | $05^1$ | 29 | | Subtract 1 |
| 16 | | | | | | | | 0100000 |
| 17 | | 20 | 22 | 0 | $05^1$ | 29 | | Dummy → $AR_+$ |
| 20 | < | 00 | 00 | 0 | 24 | 19 | > | |
| 22 | | 24 | 24 | 0 | 31 | 31 | | N.C. AR |
| AR | | N-1 | 00 | 0 | 24 | 19 | | Marker → line 19 |

Figure 51. Sync Pulse Routine

### 7.8.3 THIRD METHOD

Both methods discussed above to type in a sync pulse into the computer are tedious to work with. You have to change the command almost every time to look at a different integrator. To eliminate most of the trouble, a sync pulse routine is provided so that the reader can prepare his own tape and keep it handy for future use (See Figure 51).

The procedure for preparing the tape of the routine is given. After reading in this routine, put the compute switch to GO and type in the number (in decimal) of integrator (word) in question (tab) (S). Sync the scope on line 19.

To check if the sync pulse is in the right place, look at the number track and see the configuration of the word. Sync the scope on this pulse in line 19 and check the circuitry. You can return to the program by typing (S)C5f and compute switch to GO. Type u0 -- u7 for words 100 to 107 respectively. For word 00, type in U8 or you can sync on T0. For further information, reference may be made to Policies and Procedures Memo #220.

### 7.8.4 PROGRAM FOR SYNC PULSE

Read in one block of PPR with Enable P key.

Turn the Compute Switch to GO. Four blocks of tape will be read in. Computer halts and gate type in is established. Now, type in the following program, the coding sheet of which is given in the Appendix IV.

x00 (tab) (S) y00 (tab) (S)
.00 020501231 (tab) (S)
.05 u100802923 (tab) (S)
.08 020400502 (tab) (S)
.04 101240524 (tab) (S)
.12 141202831 (tab) (S)
.13 160742326 (tab) (S)
.07 u081802919 (tab) (S)

.18 w020632331 (tab) (S)
.06 080902630 (tab) (S)
.09 121502530 (tab) (S)
.15 u191942630 (tab) (S)
.19 202102530 (tab) (S)
.21 222302628 (tab) (S)
.23 u411422829 (tab) (S)
.14 161730529 (tab) (S)
.17 202200529 (tab) (S)
.22 242403131 (tab) (S)
.24 y20- (tab) (S)
.20 000002419 (tab) (S)
.00 .02.05.0.12.31 y01-(tab) (S)
.01 u060301905 (tab) (S)
.03 050552131 (tab) (S)
.05 u.10.08.0.29.23 z02 (tab) 00000z0 (tab) (S)
.02 z10 (tab) 8000000 (tab) (S)
.10 z11 (tab) 8000000 (tab) (S)
.11 z16 (tab) 0100000 (tab) (S)

.16. Now for punching the tape, type x06 (tab) (S). (.20ulz2y) check sum will be typed out and the tape will be punched. Read this tape in with P key and compute to GO. Sync the scope on T0 and look in Line 19.

Now, type in the integrator number in decimal and see the sync pulse as T29 of the previous word. Sync the scope on this pulse and probe the circuitry in question. If the program is left, type in (S)c5f and compute to GO. The computer will set into gate typein. Type in u0 -- u7 for words 100 to 107 accordingly. For word 00 type u8.

# SECTION VIII

# POINTS TO REMEMBER

8.0 **POINTS TO REMEMBER**

(1) In turning ON the G-15 and DA-1 combination, be sure that the warm-up cycle (A.C. cycle) in both units is completed before turning ON the D.C. power. D.C. will not come ON as long as A.C. cycle is not completed.

(2) If the computer (G-15) is used with the DA-1 disconnected (i.e. both plugs disconnected from the G-15), a dummy plug must be in place in "DA-1 Power" receptacle (PLF-16). Do not leave the DA-1 cable hanging in the back of G-15.

(3) Whenever the computer is in use and DA-1 is connected, the circuit breaker in the DA-1 must be ON.

(4) Computer power must be OFF while the DA-1 is being connected or disconnected; when diode clamp packages or Clock Repeater package are being taken out; or when PA-2 is being connected or disconnected to the DA-1.

(5) DA-1 circuit breaker should be ON before turning on the G-15.

(6) Do not vary the voltages more than 10 percent for marginal checks.

(7) Check the CLOCK PULSE before running the Test Routine.

(8) In case of power failure, the power supply should be checked according to the procedure given under MAINTENANCE, Section VI, paragraph 6.0.

(9) An extensive check of DA-1, PA-3 or PA-2 may be made, using the CALIBRATION ROUTINE.

(10) The value in the Y register, of an integrator coded as an adder, must remain close to zero throughout computation for the results to be accurate.

(11) If the value in a K register is to be +1, that value need not be inserted by the programmer. Programming routine (DAPPER) puts +1 in all K registers unless some other value is typed in.

(12) If a mistake occurs during the type-in of dx or dy inputs, remove the inputs in error by using the proper operation code from the table given in **APPENDIX IV** before retyping the correct input information.

(13) To halt and return to the "permit type-in" state in case of program difficulties, type (S)cf with the enable switch **ON** and the compute switch **OFF**.

(14) Programming Routines (DAPPER 1-A and 2) put initial values of zero in the Y register. There can be no output from integrator 00 unless it is programmed. Accordingly, it is necessary to enter a constant, normally +1, in the integrand of integrator 00 to get an output.

PAGES 141/142 INTENTIONALLY

LEFT BLANK

# APPENDIX

| PIN | SIGNAL | SOURCE | UNIT | INFORMATION | |
|---|---|---|---|---|---|
| 1 | Start DA-1 | NOX | | 0-19-31 (27Z) (4) | |
| 2 | Stop DA-1 | NOT | | 1-19-31 (67Z) (1) | |
| 3 | TO | COOX | TG | | |
| 4 | DA-1 Overflow | EOK | FO | 0-29-31 (3VZ to reset FO) 20 . IR Gnd pin 4 to set FO | |
| 5 | M6 | HOF | Memory | Line 6 information | dx |
| 6 | M7 | FOL | Memory | Line 7 information | |
| 7 | M8 | FOH | Memory | Line 8 information | |
| 8 | M9 | EOT | Memory | Line 9 information | |
| 9 | M10 | HOZ | Memory | Line 10 information | dy |
| 10 | M11 | HOX | Memory | Line 11 information | |
| 11 | M12 | HOV | Memory | Line 12 information | |
| 12 | M13 | HOU | Memory | Line 13 information | |
| 13 | $\overline{M14}$ | HOT | Memory | Line $\overline{14}$ information | r |
| 14 | M14w | FON | Memory | Normally at -20V. End on pin 14 fills M14 | rn |
| 15 | $\overline{M15}$ | HOS | Memory | Line $\overline{15}$ information | K |
| 16 | $\overline{M16}$ | HOR | Memory | | R |
| 17 | M16w | HOJ | Memory | See pin 14 | Rn |
| 18 | M17 | HOP | Memory | | Y |
| 19 | $\overline{M17}$ | HON | Memory | | $\overline{Y}$ |
| 20 | M17w | HOH | Memory | See pin 14 | Yn |
| 21 | $\overline{M18}$ | HOM | Memory | | Control Line |
| 22 | $\overline{M21}$ | HOL | Register | | ZE Line (output) |
| 23 | M21w | EOL | Register | See pin 14 | ZE |
| 24 | $\overline{M22}$ | HOK | Register | | ZS |
| 25 | $\overline{M22w}$ | EOM | Register | See pin 14 | ZS |
| 26 | $\overline{GO}$ | EOP | Register | -20V. on pin 26 blocks recirculation of M21 and M22. There is a 39K to gnd. | |

Appendix I. Signal From G-15 to DA-1 (PLF-21)

SIGNAL FROM G-15 TO DA-1 (PLF-21) - (CONTINUED)

| PIN | SIGNAL | SOURCE | UNIT | INFORMATION |
|---|---|---|---|---|
| 27 | $\overline{GO}$ * | EOR | Memory | -20V. on pin 27 blocks recirculation of M16. There is a 39K to gnd. |
| 28 | $\overline{GO}$ ** | EOS | Memory | -20V. on pin 28 blocks M14. There is a 39K to gnd. |
| 29 | GO | B15-R<br>TB2a DE | | |
| 30 | Write Pulse | TS4-2 | | |
| 31 | DS·S4·SX | B28R, COOM | CS | |
| 32 | | A38D, KOOX | | |
| 33 | | KOOV | TG | Not used in the DA-1. |
| 34 | | AOOV | | |

# DA-1 CONTROL PANEL PARTS LIST

I. Meters
- A. M1 - 59C6-004B - Voltmeter, 0-8 VAC.
- B. M2 - 59A10 - Meter, PRCT. VDC.

II. Variacs
- A. VT2-VT3 - 84C6A - EP1803
  - 1) Knobs, control modified (round) - 39A22A
  - 2) Screws, bind head, 1/4 - 20NC x 3/8 in. lg. - 13C44-003B
- B. VT1 - 84A3B - Superior, type 10
  - 1) Knob ( round) 39A19B

III. Lites
- A. Lite 1 - 95A3-002B - Green - dialco, 521310-992 )
- B. Lite 2 - 95A3-003B - Amber - dialco, 521310-993 ) 2 1/4 x 11/16"

Lugs equipped with bind head screws.

IV. Fuses
- A. F1, F2 - 68A6B - Fuse holder - Bussman #HCM-2 .453" x .750"
  - 1) Fuse - 68A1-014B-3AG250V 3A.

V. Switch Assembly - 1C1163
- A. Sw 1 (2) - 63C50B - Rotary
  - 1) Resistor mounting board - 8A327A
    - a) Spacers - 18A76A - 3/4"
- B. Resistors - E.L. type 100LE, 1W. ± .5%
  - 1) R10-81A10-002A - 26040 ohm
  - 2) R11-81A10-001A - 5208 ohm
  - 3) R12-81A10-004A - 41680 ohm
  - 4) R14, R15 - 81A10-003A - 32550 ohm

VI. Relays
- A. K1 - 57C24A - Allen-Bradley, size 1 )
- B. K2 - 57C23A - Allen-Bradley, size 0 ) shock mounted
- C. K3 - K4 - 57A22A - DOSX - 7T.
- D. K5 - K6 - 57C20B - SIGMO - 4R-ZOOS-SIL

D Socket - 62A6-002B - 5 pin

VII. Resistors
- A. Bracket mounted
  - 1) R1 - 81B12-001A-100W, 25 ohm, fixed
  - 2) R2 - 81B12-002A-100W, 50 ohm, fixed
  - 3) R6 - 81C15-007B-100W, 25 ohm, variable
- B. Mounted vertical to Chassis
  - 1) R3 - 81C13-019B - 25W, 1.5 K ohm, fixed
  - 2) R4 - 81A14A - 25W, 4K ohm, variable
  - 3) R7 - 81C6-0458 - 5W, 4K ohm, fixed
  - 4) R8 - 81C6 - 047B - 5W, 5K ohm, fixed
  - 5) R9 - 81C7 - 050B - 10W, 8K ohm, fixed
  - 6) R13 - 81C6 - 001B - 5W, 1 ohm, fixed
- C. R5 - Rheostat - 84A4B - 50W, 4 ohm

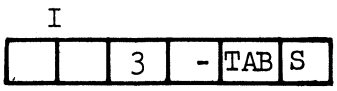
VIII. TS-3 - 8C273 - 018B - two 18 terminal barrier strips.

Appendix II. DA-1 Control Panel Parts List

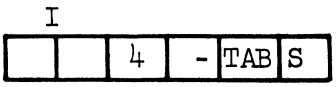
| TYPED-OUT INDICATOR | OPERATION IN ERROR | TYPE OF ERROR |
|---|---|---|
| 59 | 1+ Fill $Y_0$<br>2+ Fill $K_0$ | (50+N) exceeds limits |
| 58<br>or<br>57 | 1+ Fill $Y_0$<br>2+ Fill $K_0$ | $S_y$ too large or too small or scaling in error |
| 56 | 1+ Fill $Y_0$ | $S_y$ - $S_{dy}$ beyond limits |
| 53 | dx or dy (not coded 1+, 2+, 3+ or 4+) | Integrator number greater than 107 |
| 54<br>or<br>53 | 3+ dx input<br>4+ dy input | Integrator number of source not legitimate number |
| 52 | 3+ dx input<br>4+ dy input | Integrator number of destination greater than 107 |
| 51 | 1+ Fill $Y_0$<br>2+ Fill $K_0$ | Integrator number greater than 108 |
| 49 | 3+ dx input<br>4+ dy input | Machine error |
| zzzzzzz | u+<br>x+<br>x- | Overflow in computation preceding readout |
| z000000<br>or<br>000000z | u+<br>x+<br>z- | Sy in readout lists is too large or too small |
| No typeout after "u TAB s"<br>or<br>"x TAB s" | u+ Typeout<br>x+ Typeout<br>z+ Typeout | (50+N) in readout list exceeds limits |

Appendix III. Error Indicators During Type-in

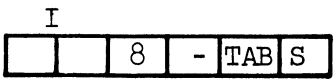
## OPERATION CODES FOR PROGRAM CONTROL




Remove dx input from Integrator I.

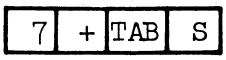



Remove dy inputs from Integrator I.

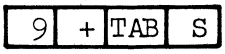



Change Integrator I from decision use to normal use.


Remove list of integrators coded for readout.



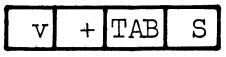

Type out integrand (hexadecimal notation) and integrator number (decimal notation) which caused overflow. Clear overflow integrand to zero. Type out integrands of readout integrators. Turn off overflow neon.




Punch out contents of DA-1 memory on tape.

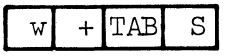

Type out decimal portions of readout integrands.


Type out integrator numbers of readout integrators.

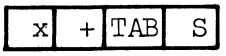



Type out tens exponent of readout integrands in excess-fifty form.


Type out $S_y$ of readout integrands in excess-fifty form.

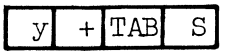



Recall DAPPER-1A loading routine.



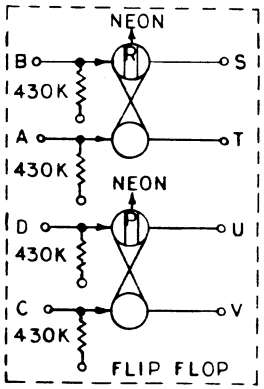

Type out decimal portions of readout integrands and compute, typing out at programmed intervals.

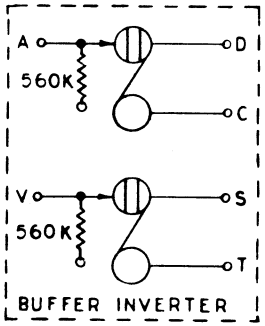

Compute, typing out decimal portions of readout integrands every iteration.

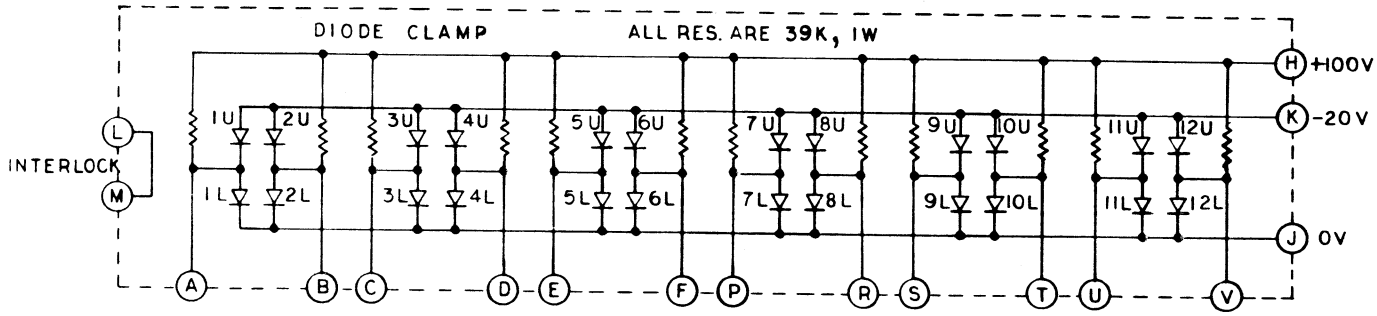


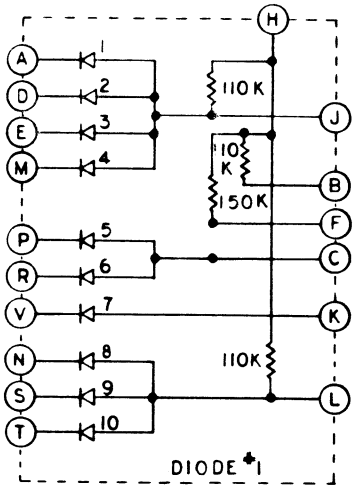

Insert initial values of integrands in all integrators.

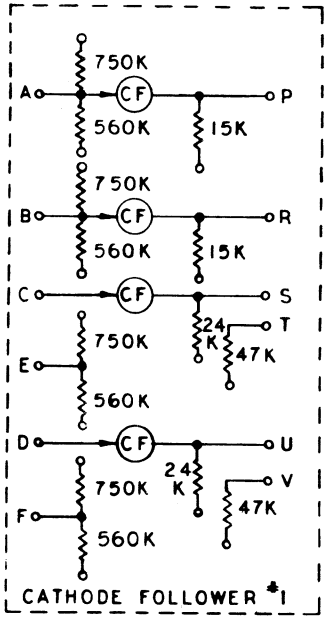
| PART NO. | PACKAGE | PKG SYM | HANDLE COLOR | A | B | C | D | E | F | H | J | K | L | M | N | P | R | S | T | U | V |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IC 998 | FLIP FLOP | F-F | YELLOW | IN-1B | IN. 1A | IN. 2B | IN. 2A | CLOCK | OPEN | KEY | OVA | OPEN | -160V | HTR-55V | HTR-55V | NEON-2 | NEON-1 | OUT-1A | OUT-1B | OUT-2A | OUT-2B |
| IC1140 | BUFF. INV. | BI | BLUE | IN.-1 | OPEN | OUT. 1B | OUT. 1A | OPEN | OPEN | +100V | OVA | OPEN | -160V | HTR.-55V | HTR.-55V | KEY | OPEN | OUT-2A | OUT-2B | OPEN | IN.-2 |
| IC1144 | CATHODE FOLL. NO.1 | CF-1 | RED | IN.-1 | IN.-2 | IN.-3 | IN.-4 | GRID DIV. | GRID DIV. | +100V | OVA | KEY | -160V | HTR.-55V | HTR.-55V | OUT.-1 | OUT.-2 | OUT.-3 | 47K TO B- | OUT-4 | 47K TO B- |
| IC 2061 | WRITE AMP | WA | BLACK | IN-1 | IN-2 | IN-3 | IN.-4 | NEON 1 | NEON 2 | OPEN | KEY | -20 | -160V | HTR.-55V | HTR-55V | NEON 3 | NEON 4 | OUT-1 | OUT-2 | OUT-3 | OUT-4 |
| IC 993 | DIODE CL. | DC | | $CL_1$ | $CL_2$ | $CL_3$ | $CL_4$ | $CL_5$ | $CL_6$ | +100V | OVA | -20V | JUMPER | | KEY | $CL_7$ | $CL_8$ | $CL_9$ | $CL_{10}$ | $CL_{11}$ | $CL_{12}$ |
| IC 964 | DIODE 1 | D1 | | $C_1$ | 110 K TO B+ | $P_{5,6}$ | $C_2$ | $C_3$ | 150 K TO B+ | +100V | $P_{1,2,3,4}$ | $P_7$ | $P_{8,9,10}$ | $C_4$ | $C_8$ | $C_5$ | $C_6$ | $C_9$ | $C_{10}$ | KEY | $C_7$ |

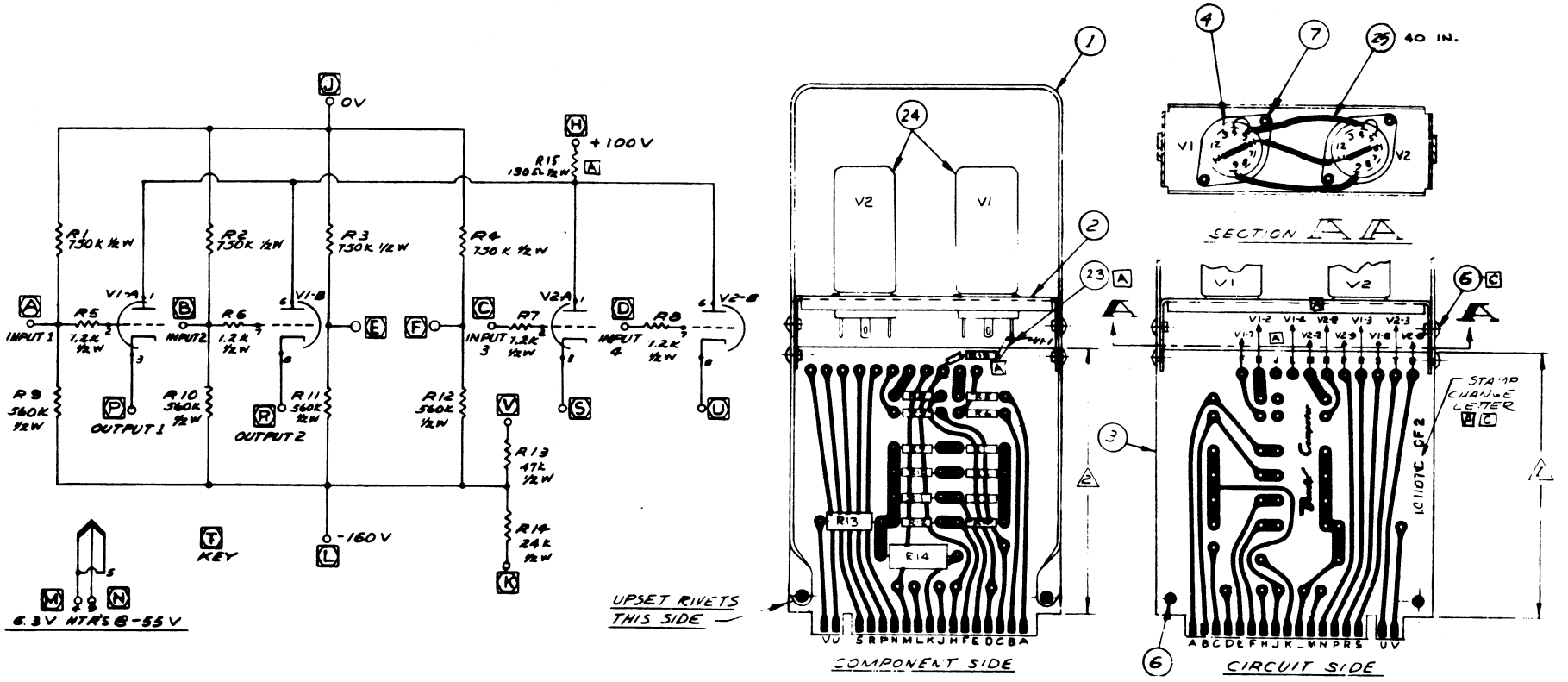


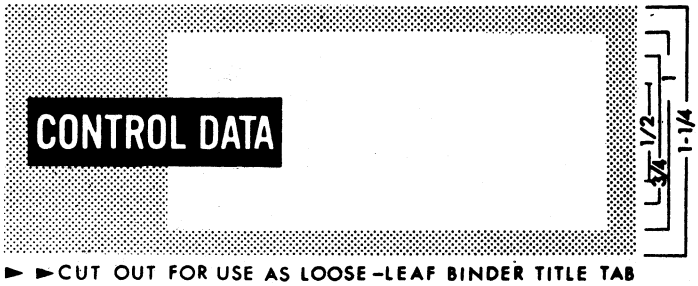



3) A INDICATES PIN LETTER ON CONNECTOR

2 DIP SOLDER SHOWN SIDE WITHIN DIM. NOTED BEFORE COMPONENT INSTALLATION

1 DIP SOLDER SIDE WITHIN DIM. NOTED AFTER COMPONENT INSTALLATION

NOTES ~




► ►CUT OUT FOR USE AS LOOSE-LEAF BINDER TITLE TAB

CONTROL DATA
CORPORATION

COMPUTER DIVISION

4201 NORTH LEXINGTON AVENUE, ST. PAUL, MINNESOTA