NASA CONTRACTOR REPORT

NASA CR-367

FLUID AMPLIFIER DIGITAL INTEGRATOR

by R. K. Rose

Prepared under Contract No. NAS 8-5408 *by*
GENERAL ELECTRIC COMPANY
Schenectady, N. Y.
for George C. Marshall Space Flight Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • MARCH 1966

FLUID AMPLIFIER DIGITAL INTEGRATOR

By R. K. Rose

Prepared under Contract No. 8-5408 by
GENERAL ELECTRIC COMPANY
Schenectady, N. Y.

for George C. Marshall Space Flight Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151 – Price $3.00

ABSTRACT

The digital integrator is a basic building block in digital computation and control systems which can be used to solve non-linear differential equations, multiply, and generate functions. Integration can be performed with respect to time or any other variable.

The fluid amplifier implementation of the applicable digital logic equations is presented. Thirty-seven OR-NOR, flip-flop, and digital amplifier elements were assembled into a circuit comprising an adder, shift register, and complementer. The experimental digital integrator used serial implementation of 5 bit words and was operated at a nominal 100 cps clock frequency.

Experimental results are presented which show the integrator's response to a step input as well as all of the detailed arithmetic operations within the integrator. An application showing the interconnection of integrators into a digital-differential-analyzer navigation system is included.

FOREWORD

The fluid amplifier digital-integrator development work described in this report was carried out as Task 2, Phase III of a NASA fluid amplifier program "Research and Development - Fluid Amplifiers and Logic" (contract NAS 8-5408). The work was sponsored by the Astrionics Laboratory at the George C. Marshall Space Flight Center, Huntsville, Alabama. The digital-integrator development program was jointly selected by Mr. R. E. Currie and Advanced Technology Laboratories personnel as a key project for future application of fluid amplifiers for digital computation and control systems.

The development work was carried out at the Advanced Technology Laboratories in Schenectady, New York. This report was prepared by Mr. R. K. Rose who was the principal development engineer for this work. Other major contributors to the program were Messrs. H. W. Avery, E. P. Kexel, E. B. Krulewich and R. E. Otten.

TABLE OF CONTENTS

Section 1

SUMMARY

1.1 BACKGROUND

The digital integrator is the basic building block in digital differential analyzer (DDA) systems just as the operational amplifier serves as the basic building block in analog computational systems. Appendix B describes typical applications of digital integrators. The use of digital computation has two major advantages over analog computation: accuracy and application flexibility. Practically any desired accuracy can be obtained by increasing the number of bits in the digital number. Digital computers can handle the multiplication of independent variables and nonlinear terms much more conveniently and accurately than analog computers. The major disadvantage of digital computation is the larger number of functional elements required.

Digital computation circuits may be either serial or parallel in implementation, depending on the method of handling the bits which make up a number. The circuits may be synchronously controlled by a "clock", or may be asynchronously controlled such that each operation triggers the next. The equipment developed under this program is the synchronously controlled serial word type.

The use of fluid amplifiers with no moving parts for digital computation systems is of interest primarily because of their environmental tolerance and their expected long life (shelf and operating). Their tolerance to both nuclear and electromagnetic radiation and to extreme temperature ranges appears limited only by the fabrication material. Shock and vibration tolerance also appear excellent; fluid amplifiers have withstood vibration levels as high as 50 g at 5000 cycles per second. In addition to these advantages it appears that production costs of fluid amplifier components can be low because of the lack of close fitting moving parts and bearing surfaces.

1.2 PROJECT SPECIFICATIONS

Since the digital integrator (DI) is the basic building block for digital computation systems, it was chosen as the most appropriate component to

study the feasibility of fluid amplifier implementation for such systems. The development plan selected consisted of two phases; 1) design of the fluid amplifier DI, and 2) fabrication of experimental hardware. The requirements of the DI were selected to permit use of existing fluid amplifier devices where possible. The resulting specifications were established as follows:

word capacity	5-bit
clock rate	100 pulses per second
iteration rate	20 per second
fluid amplifier size	0.040" x 0.040" power nozzles
operating fluid	air
supply pressures	5 psig

The selection of the 5-bit word represents a tradeoff between development cost and accomplishment of the project objectives. The 5-bit word is sufficient to obtain all the feasibility and application information expected from the project without the extra expense associated with a large amount of hardware complexity. The 5 bits produce a resolution of 1 part in 32; 10 bits would give resolution of 1 part in 1024.

The clock speed of 100 pulses per second is a reasonable response range for the available elements which had been selected. Higher speeds would probably be desired in specific applications but this is basically a problem in selection of suitable elements rather than a question of fundamental feasibility of the fluid digital integrator.

One reason for selecting the rather large elements (0.040" x 0.040") was their availability. Another reason was the problem of obtaining suitable test instrumentation for use with smaller elements. Instrumentation is certain to become a problem in future applications where small size and the need for monitoring and test instrumentation become critical.

1.3 DESCRIPTION OF DIGITAL INTEGRATOR

The basic computation performed by the digital integrator is integration of one variable with respect to a second variable, i. e.,

$$z = \int y \, dx$$

This computation is approximated by summing small discrete changes,

$$z = \Sigma\ y\ \Delta x$$

At each iteration the change in z is added to the previous value to obtain the latest value of z. The change in z is simply the incremental area represented by the latest value of y times Δx. The value of y must be continually updated to have on hand the latest value. Thus, inputs to the DI are changes in the y value (Δy) and Δx. A clock signal also is necessary to synchronize the operation of the DI. The output of the DI is a pulse train, Δz, which represents the change in the integral that has occurred. A detailed discussion of the theory of the DI is presented in Section 2.

The major components which make up the fluid amplifier DI are shown in Figure 1-1. The y and R registers are identical 5-bit shift registers for information storage. The y and R adder-control blocks operate in a very similar fashion and were made identical at the cost of a small amount of redundancy. Since the upper and lower half of the integrator (as shown in Figure 1-1) are so nearly identical it was decided to fabricate only the lower half. This provides a DI which can integrate one variable with respect to time and covers all the development areas which affect feasibility.

Several circuit possibilities for the shift register and the adder-control circuits were reviewed before arriving at the selected circuit designs. The resulting circuit is made up of 37 elements (see Fig. 2-8 for circuit), but only three element types are required. The three devices used were wall attachment OR-NOR elements, digital amplifiers, and flip-flops which were used where information storage was required, such as in the shift register. The digital amplifiers had high gain (with no memory) and were used to gate signals. Characteristic curves were obtained for each of the elements used in the DI; sample characteristics are presented in Appendix A.

The elements and circuits for the experimental equipment were fabricated from a photosensitive plastic material. The resulting hardware is described in Section 3 and is illustrated with photographs in Figure 3-2. Supply pressures for the various fluid amplifier elements ranged from 1.6 to 5.0 psig.

In order to test the DI, a signal generator is necessary to provide an input signal and clock signals. A pneumatic signal generator was fabricated

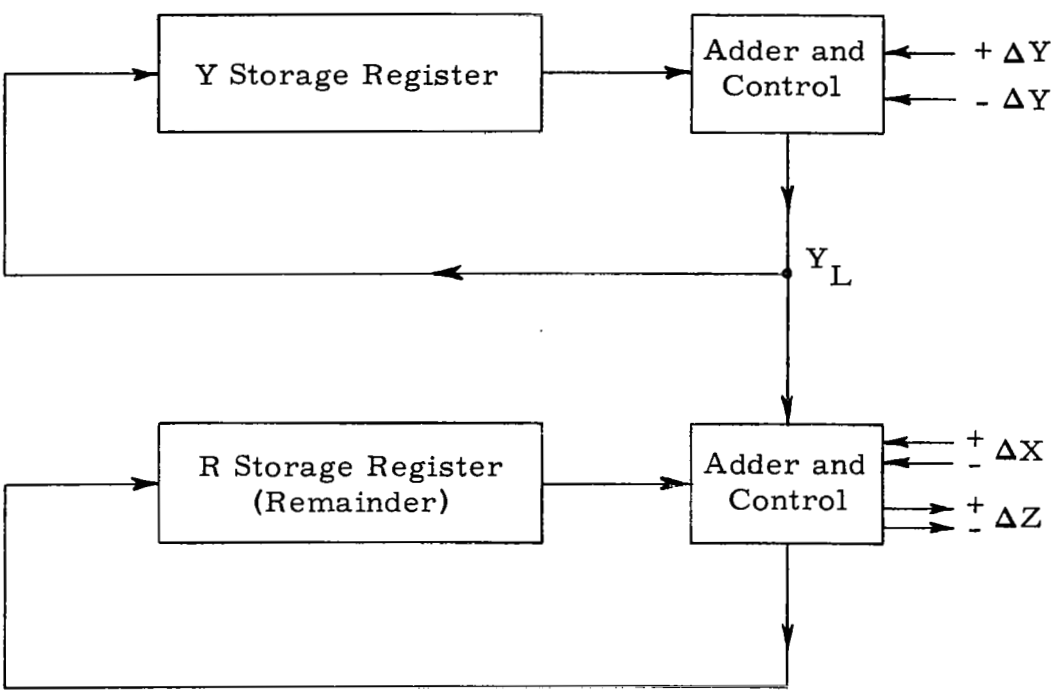


Figure 1.1. Simplified Digital Integrator Block Diagram.

for testing the equipment and is described in Section 3. This signal generator provides a 5-bit binary number for the value of y injected into the half DI (See Figure 1-1); it also generates the 100 pps clock signal, a reset signal pulse and a read pulse. In a DDA system these clock signals could be supplied by a fluid amplifier multivibrator. The binary number y, applied by the signal generator, of course would be generated by the other half of the DI(y) register and adder. The Δy and Δx input pulses would be generated by other DI's in a computation system. With the pneumatic signal generator it was convenient to supply a constant value of y, resulting in a simple test to determine the performance of the integrator half. Since the half DI provides integration with respect to time, a constant value of y results in the time integration of a constant producing a ramp output signal. Results of this and other tests carried out on the experimental equipment are described in Section 4. Several "open-loop" tests first were performed to confirm proper operation of the register and adder-control components. Then the complete system was tested "closed loop" to verify operation of the complete circuit. Verification of proper operation was established by comparing the test results with the results from a digital computer program which simulated operation of the DI.

1.4 CONCLUSIONS AND RECOMMENDATIONS

The results of this program are convincing that digital computation systems using fluid amplifiers are practical. The response speed of the fluid systems is adequate for many space applications; the potential for reliability in adverse environments such as nuclear radiation, heat and vibration is superior to electronic circuitry. Typical applications which have been considered are a satellite attitude control, and a guidance computer for an escape "lifeboat" for manned orbital stations.

The work done on this project has dealt only with the digital integrator since it was considered a key feasibility problem for digital systems. For any specific application it will be necessary to consider other parts of the system such as power supply, sensors, displays and digital/analog converters. Investigation of these areas can be pursued most economically by considering specific application requirements.

As far as the digital integrator itself is concerned, the following represent areas where additional effort should be applied:

1. miniaturization
2. speed
3. power consumption
4. packaging design and fabrication techniques
5. instrumentation.

The work in these areas is closely interrelated. Miniaturization is required to improve speed and to reduce power consumption; the degree of miniaturization will affect the instrumentation requirements for monitoring and test, and will dictate certain fabrication methods. The package design must permit very close coupling of elements to achieve high operating speeds. These recommended work areas are discussed further in Section 5.

Section 2

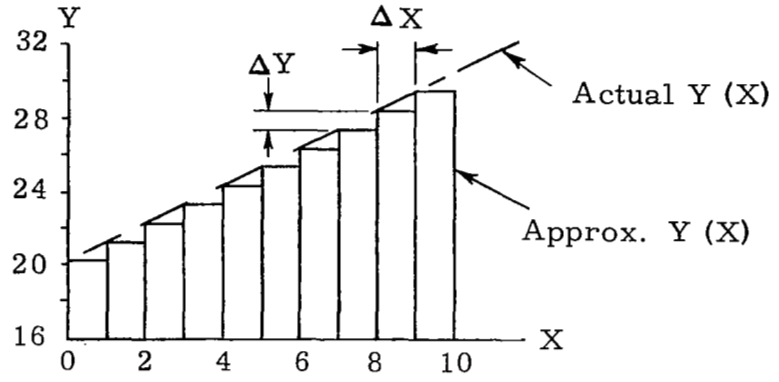
DIGITAL INTEGRATOR THEORY

2.1 BASIC THEORY

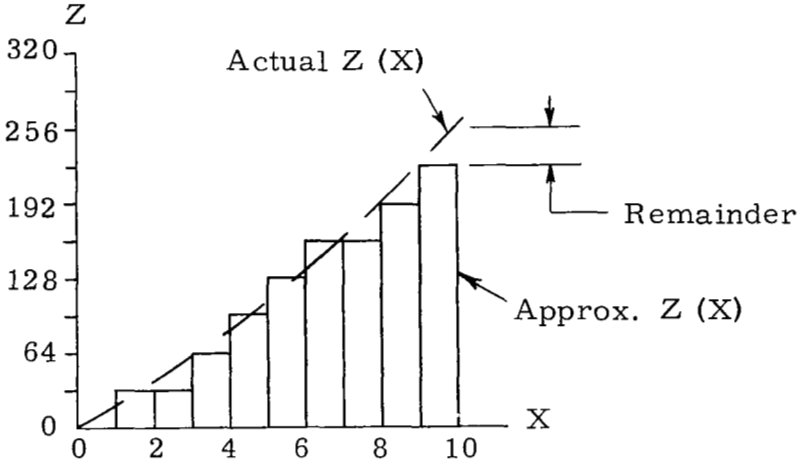
To illustrate the operation of the digital integrator, first consider the variables x, y, and z to be related by the function

$$z = \int_{x_1}^{x_2} y \, dx \qquad (2\text{-}1)$$

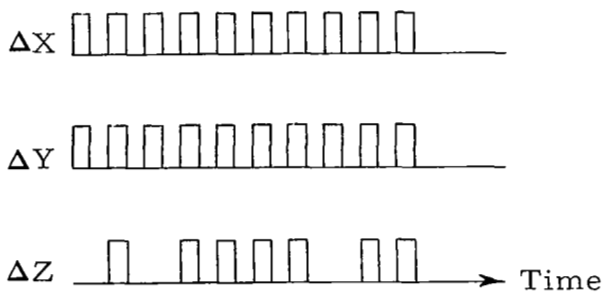
In order to accomplish this integration by numerical means the above integral is replaced with the summation $z = \Sigma y \, \Delta x$ for finite values of x. As an example, to illustrate the numerical integration assume y = f (x) is a straight line as shown in Figure 2-1a. Assume the accuracy with which x and y are known are integral units; i.e., x = 1, 2, 3 --y = 21, 22, 23 . Further assume that the integral is totalized in integral increments of 32; i.e., z = 0, 32, 64 . Under these restrictions the function and its integral would appear as illustrated in Figure 2-1a and Figure 2-1b, respectively. In this example Δx has been chosen to be unity for convenience. Integration then is performed by adding successive incremental areas $y \, \Delta x$ or simply successive values of the ordinate of y ($\Delta x = 1$). Δx, Δy and Δz in the integrator are represented by a series of pulses and in this example one Δx pulse has the value of 1, one Δy pulse has the value 1 and a Δz pulse has the value of 32. In the integrator these variables would be represented by pulse trains. For the example chosen actual pulse trains representing the variables are shown in Figure 2-1c. When the integrator is used in a system, the output pulses, Δz, would be accumulated on a counter and multiplied by a conversion factor to obtain the value of the integral or would be used as inputs to other digital integrators to perform programmed calculations. A variable can be integrated with respect to any other variable (the usual analog computer operational amplifier can integrate only with respect to time). Note that the digital integrator works with changes in quantities rather than their absolute values (e.g., Δz instead of z; Δx instead of x).



a. Assumed Function y = f (x).



b. Resulting integral versus x.



c. Pulse train representing variables.

Figure 2. 1. Example Numerical Integration

A block diagram of the digital integrator which performs the numerical integration process described above is illustrated in Figure 2-2. The upper half of the diagram (y storage register and adder-control) continually computes the current value of y. This computation is simply increasing or decreasing the previous value of y (stored in the register) by an amount $+\Delta y$ or $-\Delta y$ on every iteration to obtain the current value of y which is then stored back in the y register. The adder-control block adds the quantity Δy to y as the latter is circulated through it. The lower half of the block diagram sums the incremental areas $y\,\Delta x$ and thus indicates the value of integral. The summation of the incremental areas is done in the lower adder-control which on the application of a Δx pulse adds the value of y (y is the value of the incremental area since $\Delta x = 1$) to the previous value of the integral stored in the register. The incremental area is either added or subtracted from the value in the R register depending on the sign of Δx. The new value of the integral is returned to the R register for computation in the next iteration. Since the capacity of the R register is some finite value, the integral value stored in it may exceed the register capacity. When this occurs the register "overflows" and the overflow pulse represents an incremental value of area, viz. a value of 32. In the example the R register capacity then is 31. It is desirable to scale the digital integrator so that the R register overflows many times during computation since only the overflows are accounted for in arriving at a value of the integral. The error in the computation is represented by the value which is the remainder, R, retained by the R register. Numerically R represents the total value of the integral minus the algebraic accumulation of the number of Δz (overflow) pulses.

Since Δy and Δx may be negative as well as positive, the adder-control block must be capable of subtracting and adding. Using the lower adder-control block as an example, if Δx is positive y is added to R; if Δx is negative y is subtracted from R. The addition process is accomplished by applying y and R directly to the adder when Δx is positive. When Δx is negative the complement of y is first obtained and then y complement is added to R. Thus, subtraction is performed by adding a complemented number. An example of subtraction

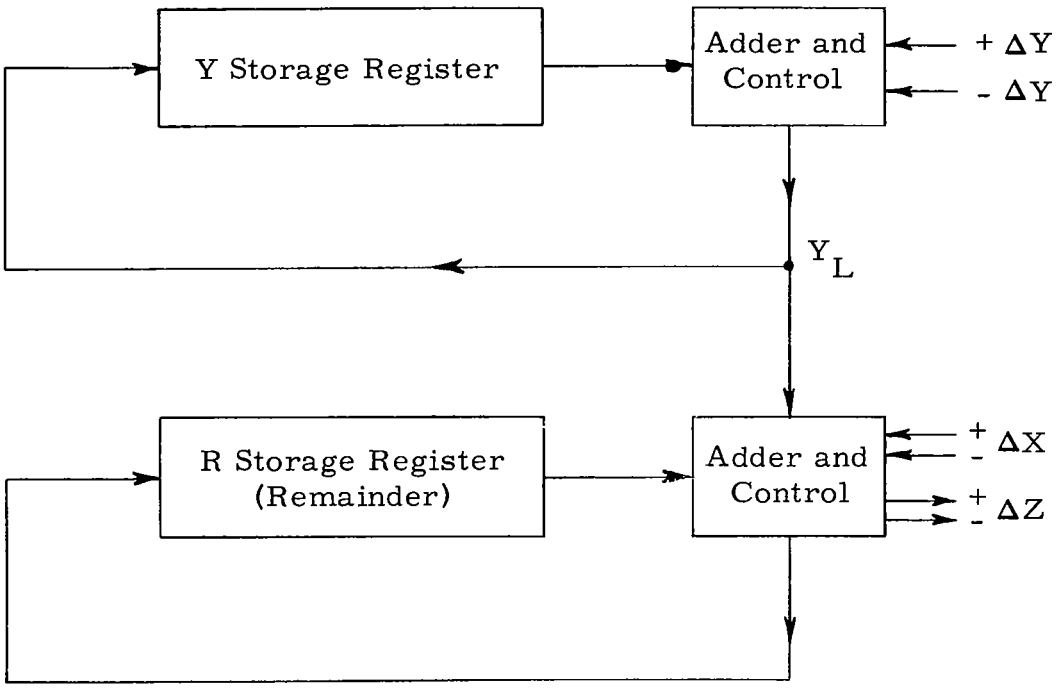


Figure 2.2. Simplified Digital Integrator Block Diagram.

is as follows: since "two's complement" arithmetic is used in the integrator, subtraction of a binary number, N, is accomplished by adding its complement which is $2^B - N$ where B is the number of bits in the word. For example let

$$N = 21 = 1 \quad 0 \quad 1 \quad 0 \quad 1$$

Then the complemented value of N is

$$2^B - N = 2^5 - 21 = 11 = 0 \quad 1 \quad 0 \quad 1 \quad 1$$

Assume 21 is to be subtracted from 29

	1	1	1	0	1	(29)
+	0	1	0	1	1	(complement of 21)
1	0	1	0	0	0	(8)

The sixth digit in this case is truncated (this digit will later be employed as overflow) leaving the binary number 8 (0 1 0 0 0) as the answer. A feature of two's complementing (to be explained in further detail later) is illustrated by comparing the number 21 with its complement and forming the following general rule: to compute the two's complement, pass all leading (starting from units column) zeros and the first 1 of the number; thereafter invert each digit. For the number 12 as an example

	12 =	0	1	1	0	0
Complement of	12 =	1	0	1	0	0

Succeeding Digits Inverted ← | → First Zeros and 1 Passed

When the original block diagram (Figure 2-2) is enlarged to include the complementing function it appears as shown in Figure 2-3. The inputs at the left side of the diagram are Δx and $\overline{\Delta y}$. Each of these inputs physically represents one input port, and if an input is not present its inversion is assumed. That is, if a positive Δx is absent the integrator assumes that Δx is negative. In addition, in the process of integrating if a particular variable is to be held constant, an alternating $\pm$ input (supplied by external circuitry) is sent to the integrator, thus approximating the no-change condition.

Again the logic operation of the integrator is simply one of addition or subtraction of an updated value of Y with a previous value of a remainder R, the choice of addition or subtraction being controlled by the complementer.

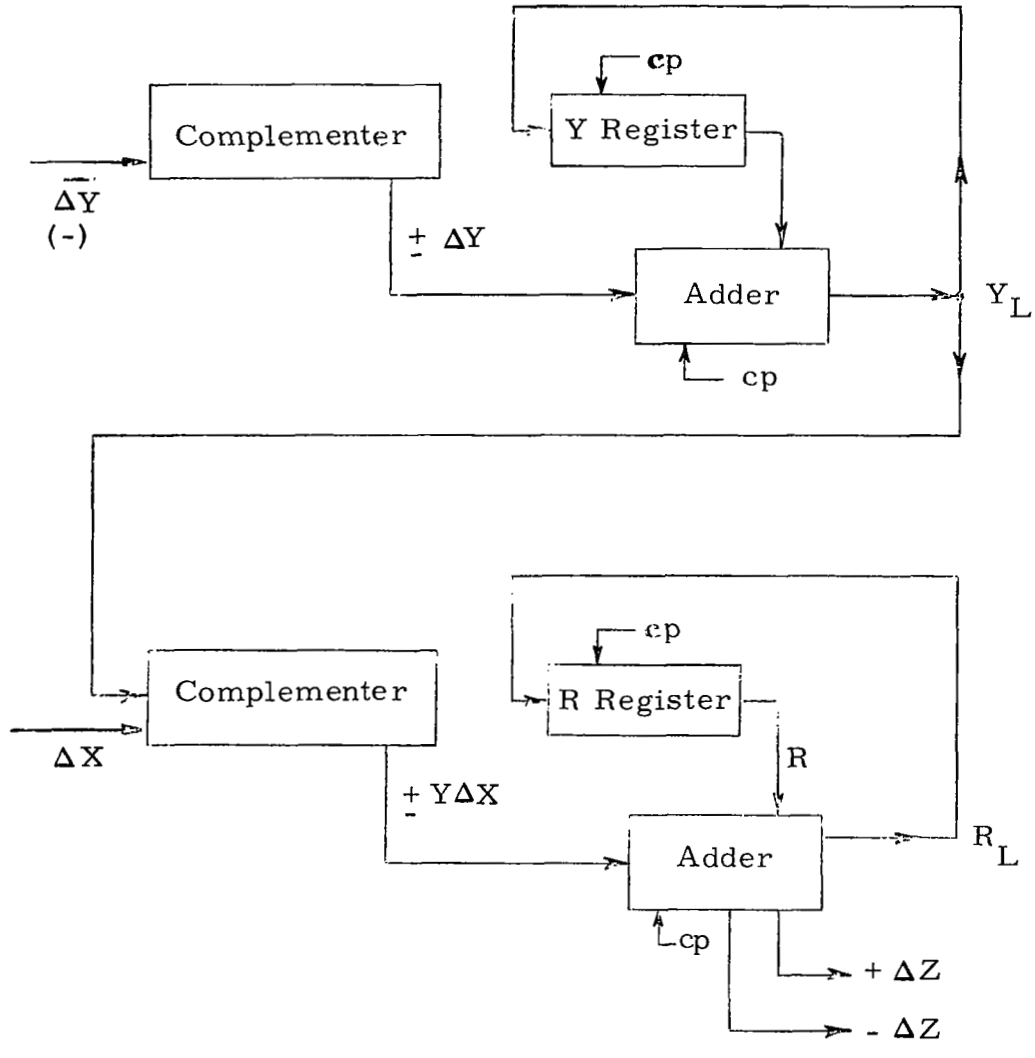


Figure 2. 3. Complete Digital Integrator Block Diagram.

For instance, in the lower part of the integrator, if Δx is present (implying a positive value) y is passed directly through the complementer with no change. If Δx is not present (implying a negative value) y is complemented before being sent to the adder. To maintain synchronization of information throughout the circuit a central source of timing pulses (indicated T in the diagram) is applied by an external clock. The frequency of these pulses divided by the number of bits in a word is the number of computations the integrator can perform in one second. The present fluid digital integrator employs 5-bit words and a 100 cps clock frequency which results in 20 iterations per second.

2.2 INTEGRATOR SUBSYSTEMS

Reference to Figure 2-2 shows that the integrator is divided into two identical halves (excluding $\pm \Delta z$ output channels), each of which integrate as a function of time. The time integrators are made up of three circuits: an adder, a complementer, and a register. The logic design for each of these three circuits will be discussed in detail in the following paragraphs.

2.2.1 ADDER CIRCUIT

The logic diagram for the adder circuit is shown below and consists of two half-adders with associated carry functions.

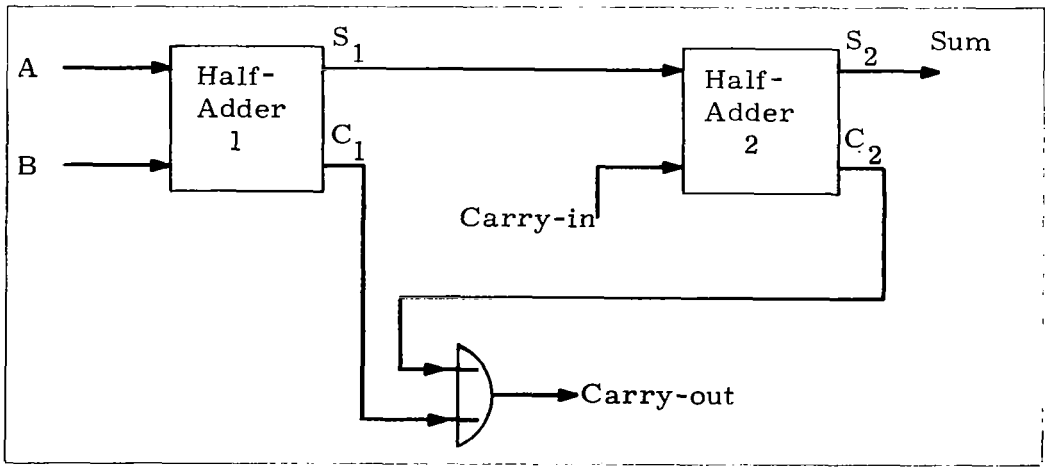


Figure 2.4. Adder Circuit.

In the first half-adder, a sum and carry is produced by adding A and B. In the second half-adder, a sum and carry is produced by adding the partial sum output of half-adder 1 with a carry resulting from a preceding addition. The truth table for half-adder 1 is

A	0	1	0	1
B	0	0	1	1
S_1	0	1	1	0
C_1	0	0	0	1

and the Boolean equations are:

$$S_1 = A\bar{B} + \bar{A}B \tag{2-2}$$

$$\text{and } C_1 = AB \tag{2-3}$$

Because of the active nature of NOR logic (each OR-NOR fluid amplifier has its own power supply) as well as the desirability of standardizing on one type of logic element, adder functions and all other logic functions associated with the integrator were designed to react to the absence of signals. Returning to Equation 2-2 and negating both sides of the equation

$$\bar{S}_1 = \overline{A\bar{B} + \bar{A}B} \tag{2-4}$$

which by DeMorgan's laws $\left[\begin{array}{ll} \text{Law I} & \overline{ab} = \bar{a} + \bar{b} \\ \text{Law II} & \overline{a + b} = \bar{a}\,\bar{b} \end{array}\right]$ becomes

$$\bar{S}_1 = \overline{(A\bar{B})}\ \overline{(\bar{A}B)} \qquad \text{(Law II)} \tag{2-5}$$

and employing Law I:

$$\bar{S}_1 = (\bar{A} + \bar{\bar{B}})\ (\bar{\bar{A}} + \bar{B})$$

$$= (\bar{A} + B)\ (A + \bar{B}) \quad (\text{Since } \bar{\bar{A}} = A) \tag{2-6}$$

Expanding Equation 2-6:

$$S_1 = \cancel{0}\,\overline{A}A + \overline{A}\overline{B} + AB + B\overline{B}\,\cancel{0}$$

$$= \overline{A}\overline{B} + AB \qquad (2\text{-}7)$$

Now from Law II

$$\overline{A}\overline{B} = \overline{A + B}$$

$$\text{and } AB = \overline{\overline{A}\overline{B}}$$

$$= \overline{\overline{A} + \overline{B}}$$

so that the negated sum is

$$\overline{S}_1 = \overline{A + B} + \overline{\overline{A} + \overline{B}} \qquad (2\text{-}8)$$

Also from Law I, the carry (given by Equation 2-3) is

$$\overline{C}_1 = \overline{A} + \overline{B} \qquad (2\text{-}9)$$

Equations 2-8 and 2-9 may be implemented by employing fluid amplifier NOR elements. For the single NOR element shown in Figure 2-5 an input at a or b produces an output at the right leg of the element.

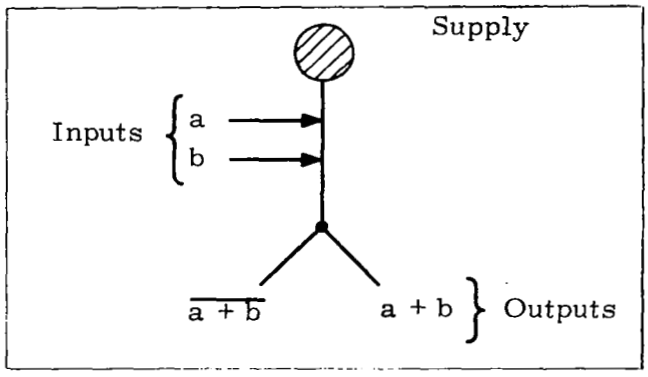


Figure 2.5. OR-NOR Element.

An absence of both a and b produces a high output at the left output port. More complete element characteristics obtained experimentally are presented in Appendix A. Three NOR elements are required to implement Equations 2-8 and 2-9, and the resulting circuit, shown in Figure 2-6, is a half-adder stage.

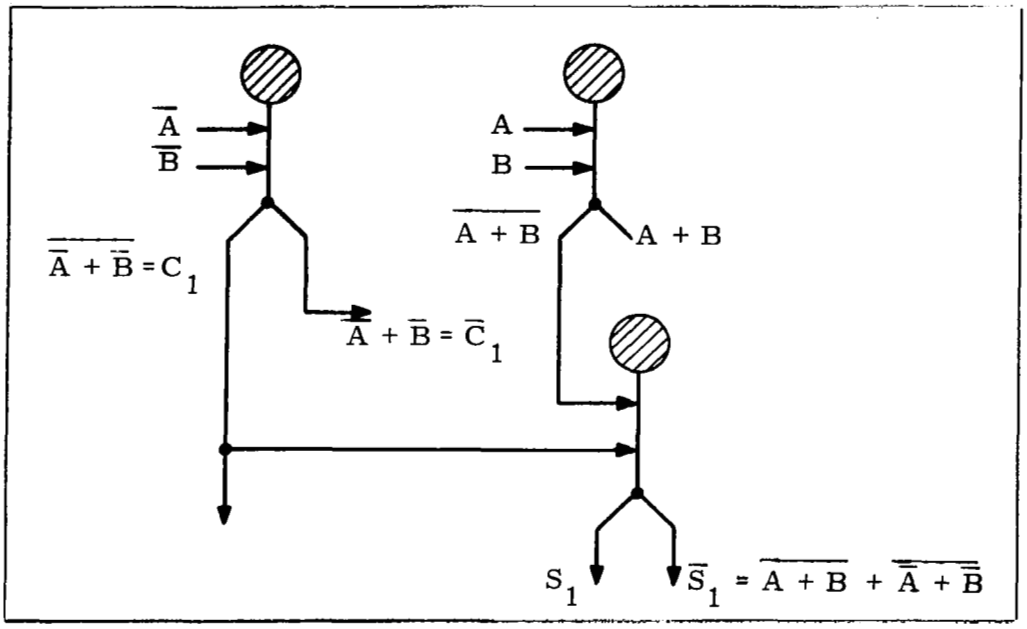


Figure 2.6. Half–Adder Circuit.

Both half-adders as well as half-adder functions required elsewhere in the integrator are implemented as shown in the latter diagram.

A carry may be generated at either half-adder 1 or half-adder 2 (see Figure 2-4), and since the addition is serial, these two carries, summed in an OR element, must be stored until the next two digits A and B are available. This storage is accomplished by applying the carry-out signal to the input of a shift register stage (used as one-bit storage) and passing the carry-out forward at clock frequency; that is, the carry-out becomes a carry-in one clock pulse later. The single shift register stage is shown in Figure 2-7.

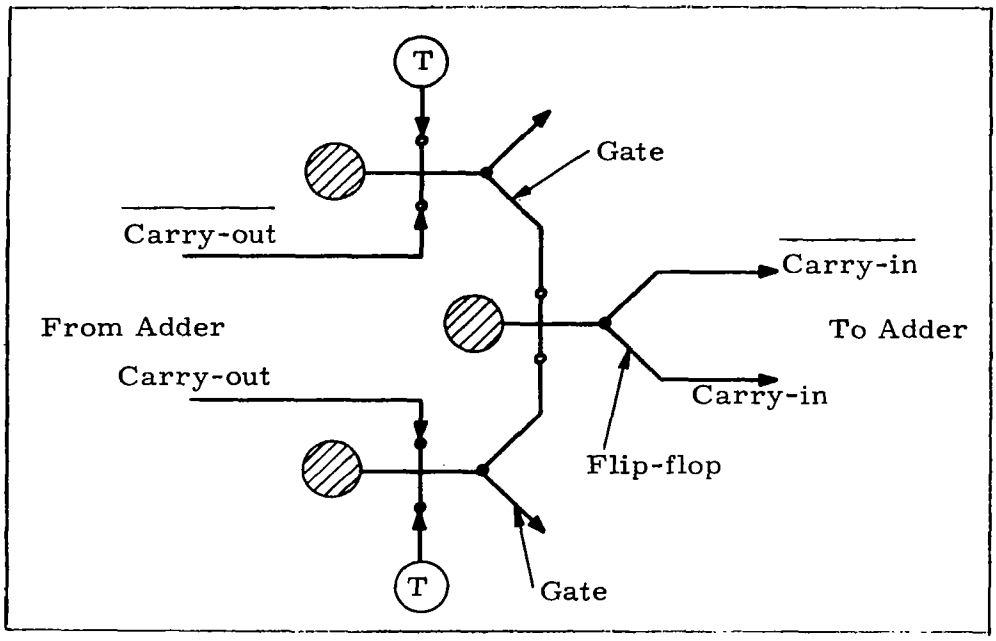


Figure 2.7. Shift Register Stage
(Used for Carry Storage)

The two gates shown at the top and bottom of the diagram are digital amplifiers having no memory and requiring some input signal to provide a stable output. The clock D.C. signal level is set slightly below the interior element pressure so that when the clock pulse T is absent the gate outputs are held out and vented to the atmosphere. The flip-flop is a bi-stable device and has memory in that the power jet initially deflected either up or down will attach to the wall toward which it is deflected and will remain so even after the control is removed. (Complete performance curves for these two elements are also given in Appendix A.) Operation of the register stage circuit is illustrated by the following example: assume a carry-out is present; that is, the pressure is high at the lower input and zero at the upper input. Upon application of a positive clock pulse T, a differential is developed across the upper gate (zero differential is developed across the lower gate since carry-out is present) thus applying an input signal to the upper control of the flip-flop thereby switching the flip-flop to the lower (carry-in) position. Thus, the overall action of the circuit is to pass information from the input (carry-out) side to the output (carry-

in) side on application of a clock pulse T and store this information (in the state of the flip-flop) until application of the next pulse T.

A complete adder circuit, comprising two half-adder circuits and a register stage, is shown in the complete circuit schematic (Figure 2-8). Discussion of the operation of the overflow stage, activated by the READ pulses shown on the diagram, will be deferred until later. As a further example of operation of the adder, consider the timing diagram shown in Figure 2-9 and assume the numbers A = 5 and B = 21 are to be added. Prior to T_1 (the first clock pulse) A and B produce a zero output of R_L and a carry-out; application of T passes the carry forward to the carry-in position where it is combined with zero inputs from A and B to produce a high output at R_L. The rest of the addition proceeds from T_1 through T_5 in direct analogy to the binary addition shown above the timing diagram. In an actual test of equipment the numbers A and B were generated with a pneumatic signal generator consisting of a binary coded disk rotating in synchronization with the clock frequency. This equipment is described in Section 3.

2.2.2 COMPLEMENTER OPERATION

If a subtract signal is present, the subtrahend is complemented and then added to the minuend. If a substract signal is absent (implying addition) the number is passed unmodified through the complementer. As shown previously (Section 2.1), to form the two's complement all leading zeros as well as the first one of the subtrahend are passed intact. Then all succeeding digits are inverted. Thus the logical requirements of the complementer can be summarized as shown in Figure 2-10. E is assumed to be the number passing into the complementer.

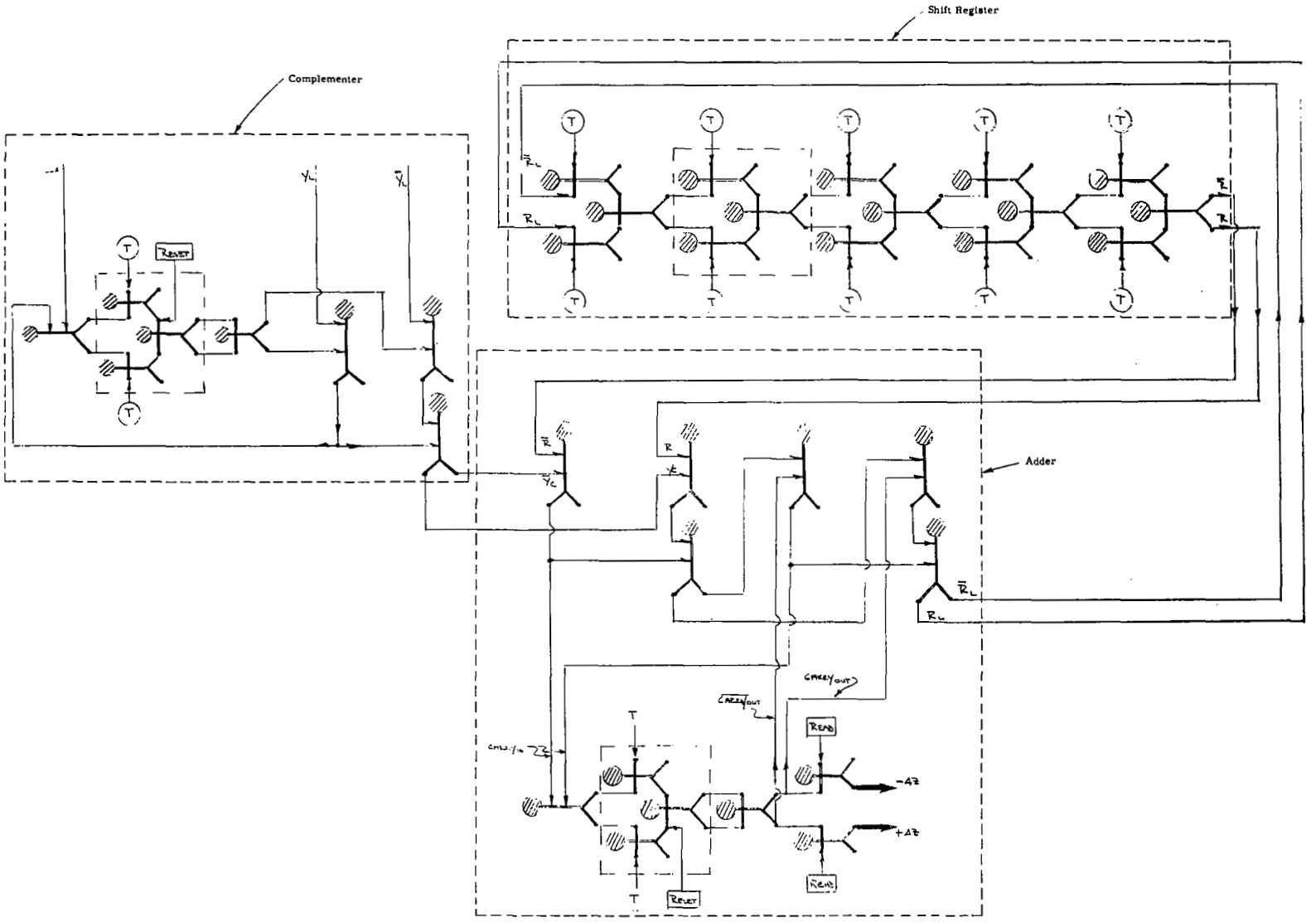


Figure 2-8. Circuit Schematic.

	(16)	(8)	(4)	(2)	(1)
Carry in		1		1	
A (=5)	0	0	1	0	1
B (=21)	1	0	1	0	1
R_L (=26)	1	1	0	1	0

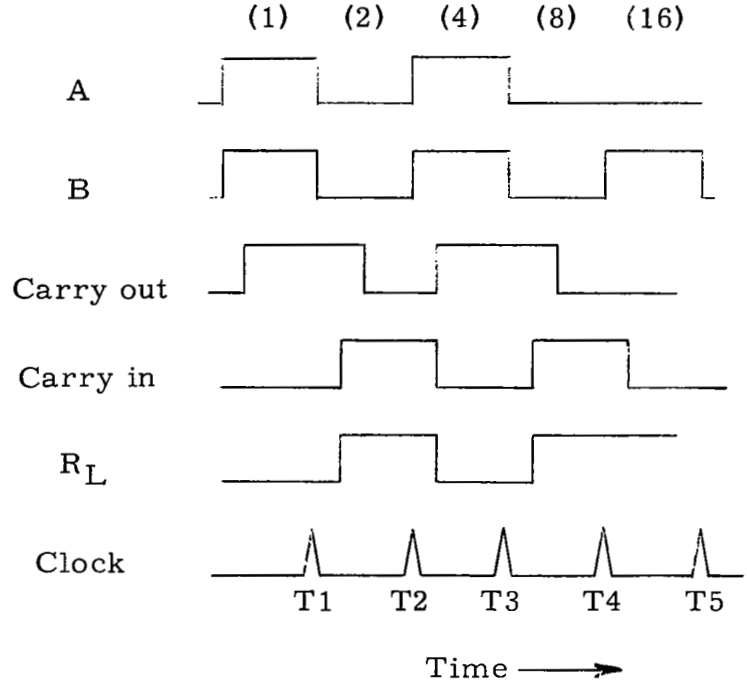


Figure 2.9. Adder Timing Diagram

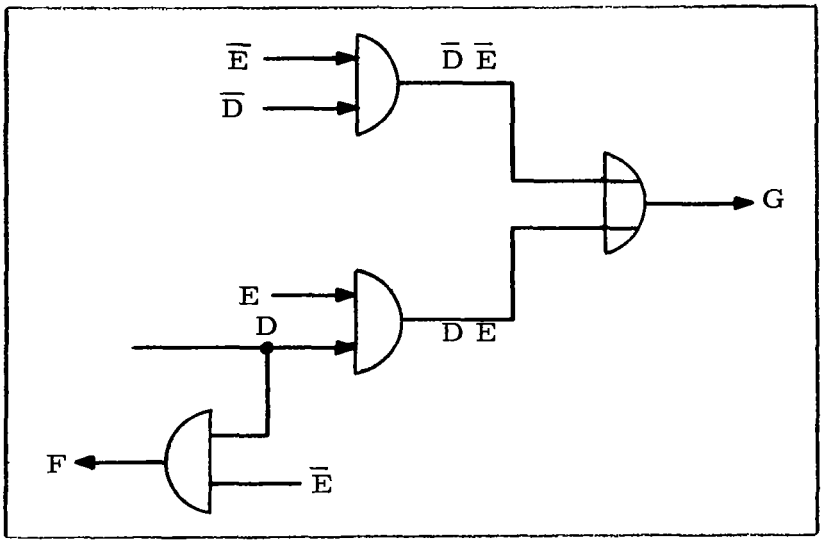


Figure 2.10. Complementer Logical Diagram.

Applicable equations from the above Figure 2-10 are:

$$G = \overline{D}\overline{E} + DE \qquad (2\text{-}10)$$

and

$$F = D\overline{E} \qquad (2\text{-}11)$$

Assume initially that D is present, in which case E is passed through directly to G. This requirement must be met since in the two's complementer at least the first digit is passed directly through the complementer. Further assume that the complement level is high. Since, following the first digit of E, succeeding digits of E are to be inverted, control must be passed to the upper AND gate. This is accomplished by applying D and $\overline{E}$ to the lower AND gate thereby producing F; F is then stored for one-bit time (to allow the first E digit to pass) and reappears as the absence of D ($\overline{D}$ present) at the next bit time. Notice that F then drops out since D drops out. The one-bit storage is accomplished in the same manner as in the previously described adder circuit by a shift register stage. Also as in the adder circuit, it is advantageous to convert Equations 2-10 and 2-11 to NOR logic as described in the following set of Boolean equations:

Negating Equation 2-10

$$\overline{G} = \overline{\overline{D}\overline{E} + DE} \tag{2-11}$$

Employing DeMorgan's laws

$$\overline{G} = (\overline{\overline{D}\overline{E}})\ (\overline{DE}) \tag{2-12}$$

$$= (\overline{\overline{D}} + \overline{\overline{E}})\ (\overline{D} + \overline{E}) \tag{2-13}$$

$$= (D + E)\ (\overline{D} + \overline{E}) \tag{2-14}$$

Expanding Equation 2-14

$$\overline{G} = D\cancel{\overline{D}}^{\,0} + D\overline{E} + E\overline{D} + \cancel{E\overline{E}}^{\,0}$$

Since

$$D = \overline{\overline{D}} \text{ and } E = \overline{\overline{E}}$$

$$\overline{G} = \overline{\overline{D}}\,\overline{E} + \overline{\overline{E}}\,\overline{D} \tag{2-15}$$

Again by DeMorgan's laws

$$\overline{G} = (\overline{\overline{D} + E}) + (\overline{D + \overline{E}}) \tag{2-16}$$

The function F in the previous diagram can also be inverted by employing DeMorgan's laws as follows:

$$\overline{\overline{F}} = \overline{D\overline{E}} \tag{2-17}$$

$$= \overline{D} + \overline{\overline{E}} \tag{2-18}$$

$$\overline{F} = \overline{D} + E \tag{2-19}$$

Equations 2-16 and 2-19 were implemented as shown in Figure 2-11 by employing fluid amplifier OR-NOR elements. Note that proper selection of inputs makes the latter array of elements identical to the half-adder stage used in the adder. The complete complementer including the one-bit storage is shown in Figure 2-8. Note that the action of the flip-flop RESET which is applied at the end of an iteration is to return the storage flip-flop to the lower position thereby insuring that D (Figures 2-10 and 2-11) will initially be present as assumed in the development of the logic equations. In Figure 2-8, Δx controls the complementer and Y_L corresponds to E of Equations 2-10 through 2-19. Examination of the circuit shows that if Δx is present Y_L is passed directly

through the complementer. If Δx is absent Y_L is complemented before being passed to the adder.

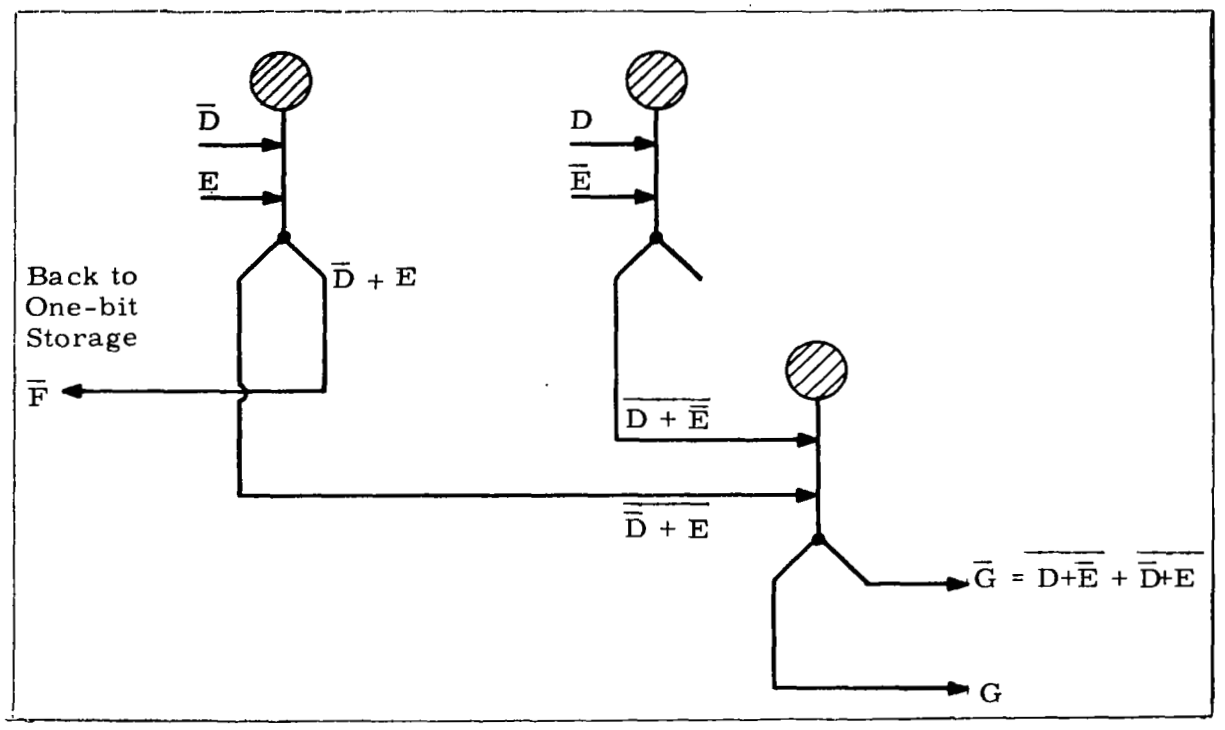


Figure 2.11. Partial Complementer Circuit.

A typical complementer timing diagram for complementing the number 10101 is shown in Figure 2-12.

2.2.3 SHIFT REGISTER

Five of the one-bit storage stages (shown in Figure 2-7) are connected in series as shown in Figure 2-8 to obtain the five-stage shift register. The most significant bit (MSB) is contained in the left-hand flip-flop at the beginning of an iteration, and the least significant bit (LSB) is contained in the right-hand flip-flop. Figure 2-13 presents a typical timing diagram. To illustrate operation, assume a periodic 5-bit word, 11010, is applied to the register

input. The register is assumed to be cleared initially. The states of the five flip-flops show that the output (from Figure 2-5) is shifted by five clock pulses from the input. In addition if, after any integral multiple of five clock pulses, the clock is removed, the states of the five flip-flops indicate the original input number as shown in Figure 2-13.

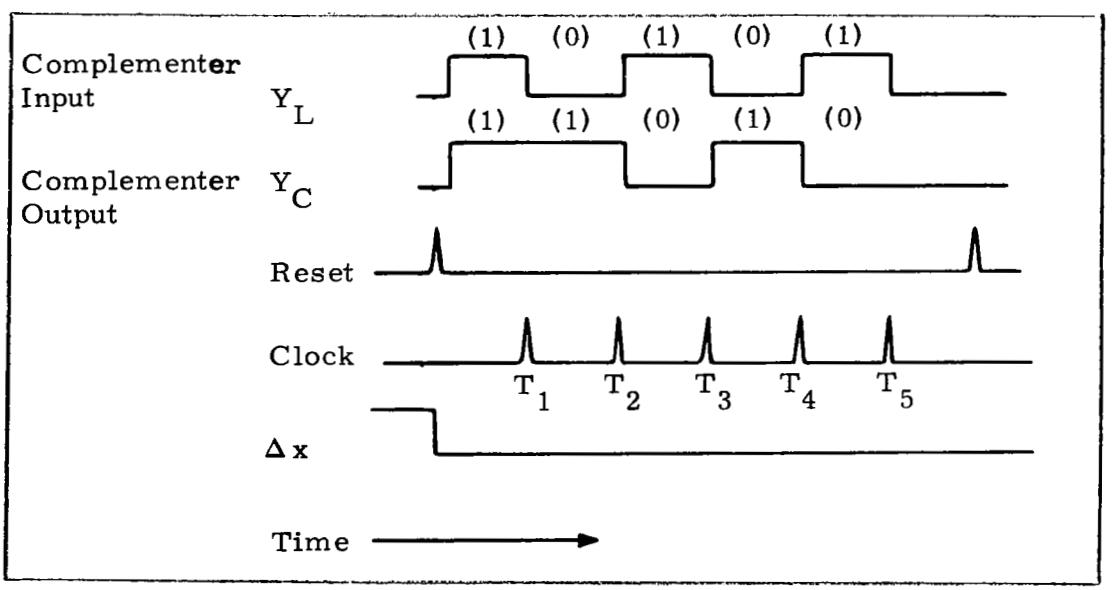


Figure 2.12. Complementer Timing Diagram.

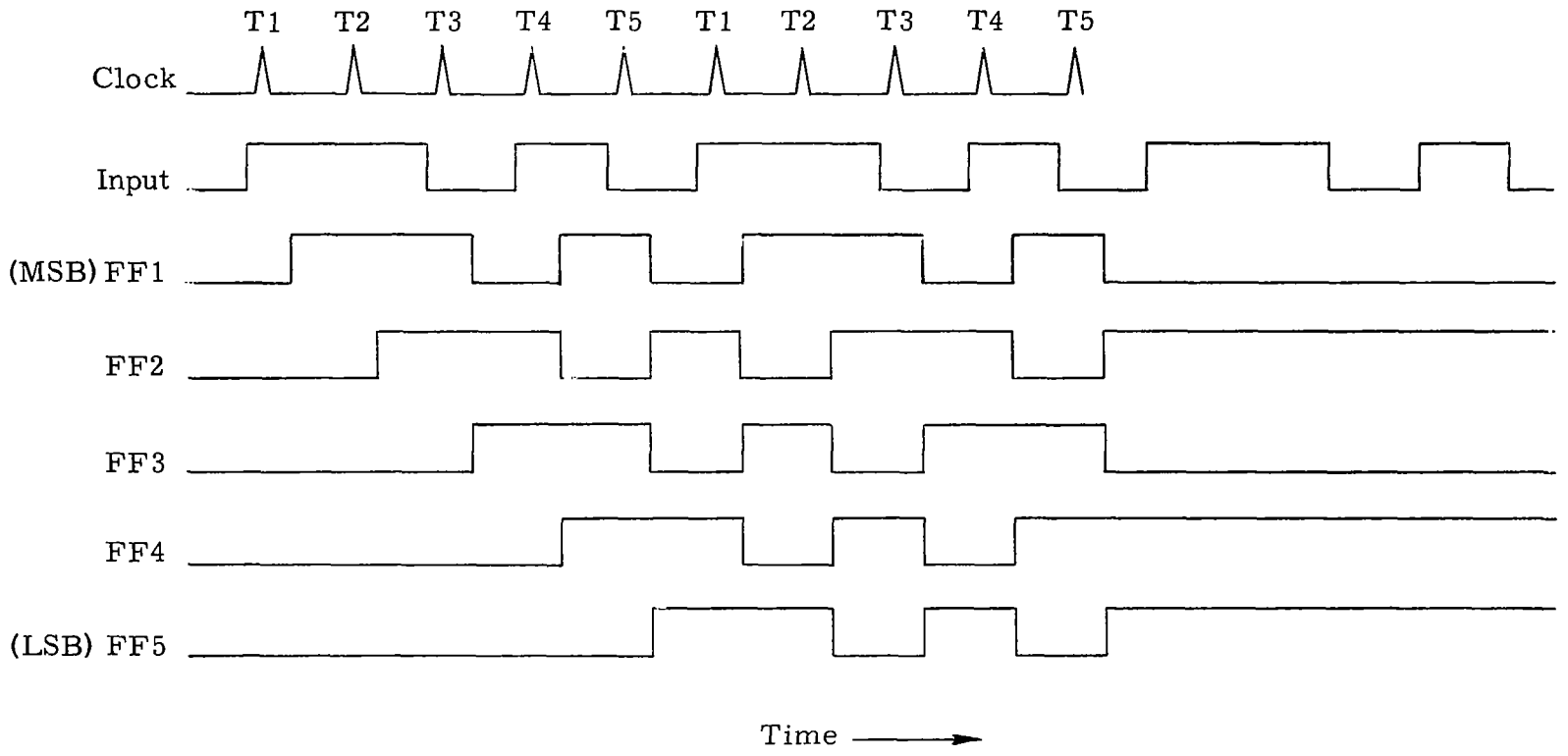


Figure 2.13. Shift Register Timing Diagram.

2.3 INTEGRATOR OPERATION

As previously outlined, the integrator operates by receiving an input Y_L at the complementer. If Δx is absent Y_L is complemented; if Δx is present Y_L is passed directly to the adder. At the adder the remainder of a previous addition and the latest value of Y_C are summed, producing a new remainder and a value of Δz (a change in the value of the integral).

Operation of the adder reset and overflow stage now can be illustrated. Assume as an example that the two numbers Y_C = 20 (1 0 1 0 0) and R_L = 30 (1 1 1 1 0) are to be summed. The addition will produce the value 50 which is too large to be contained as a five-bit number, and a sixth bit will be formed; this bit will be sensed as overflow and of course has the binary value 32, i.e.,

	1	0	1	0	0	(Y_C)
	1	1	1	1	0	(R_L)
1	1	0	0	1	0	
Overflow	Remainder					

The presence of an overflow is sensed by determining if a carry has occurred after the addition of the fifth and most significant bits (occurs at the fifth clock pulse). The sensing is done by applying READ pulses on two shift register gates driven by the carry signal (see Figure 2-8). The READ pulses are applied only after each fifth clock pulse so that only carry pulses with a value of 32, i.e., overflows, are sensed. Since each bit generated by the adder is normally sent back to the register to be employed as remainder in a succeeding iteration, the flip-flop which generates an overflow must be reset to zero before the next iteration commences. Operation of the overflow stage is similar to that previously described for the shift register in that the two gates shown are held open until a READ pulse is applied. A timing diagram, illustrating the operation of the READ and RESET signals is shown in Figure 2-14. Note that the overflow will always be either positive or negative and never zero.

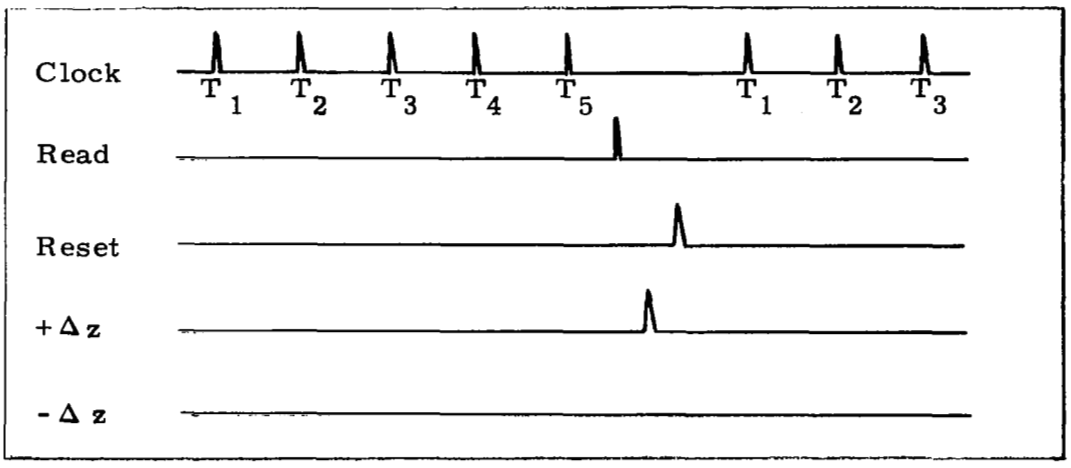


Figure 2.14. Read and Reset Timing Diagrams.

2.4 INTEGRATOR SCALING AND BIAS REQUIREMENTS

The digital integrator is constructed so that an overflow (either positive or negative, $\pm \Delta z$) is produced at each READ time. For a 5-bit binary number system the magnitude of a number must lie between a minimum of 0 0 0 0 0 and a maximum of 1 1 1 1 1. The largest number of negative overflows are obtained when 0 0 0 0 0 is integrated, and the largest number of positive overflows are obtained when the number 1 1 1 1 1 is integrated. A zero output from the integrator is defined as the level when the total number of positive overflows equals the total number of negative overflows. Thus in converting from binary to decimal equivalents the decimal numbering system must be biased about zero so that binary 0 0 0 0 0 represents a negative number and 1 1 1 1 1 represents a positive number. This requirement is illustrated by noting the overflow output for integrating the minimum and maximum 5-bit binary numbers (0 0 0 0 0 and 1 1 1 1 1) as functions of time. Assume both initial values to be 0 0 0 0 0 corresponding to a cleared register.

```
  0  0  0  0  0
+ 0  0  0  0  0
-------------------
  0  0  0  0  0    = 0 + neg ovf
+ 0  0  0  0  0
-------------------
  0  0  0  0  0    = 0 + neg ovf
+ 0  0  0  0  0
-------------------
  0  0  0  0  0    = 0 + neg ovf
        .
        .
        .
        .
        .
        .
      etc
```

```
    0  0  0  0  0
+   1  1  1  1  1
-------------------
    1  1  1  1  1  = 31 + neg ovf
+   1  1  1  1  1
-------------------
1   1  1  1  1  0  = 30 + pos ovf
+   1  1  1  1  1
-------------------
1   1  1  1  0  1  = 29 + pos ovf
        .
        .
    -   .
        .
        -
1   0  0  0  1  0  = 2 + pos ovf
+   1  1  1  1  1
-------------------
1   0  0  0  0  1  = 1 + pos ovf
```

For the left-hand example after 31 additions, 31 negative overflows are obtained. For the right-hand example 30 positive and one negative overflows are obtained with a remainder of 1. The two numbers just chosen, since both represent the largest number of negative and positive overflows, represent, of course, the maximum negative and positive numbers.

Now, bias each of the latter binary numbers by the value of the most significant bit (MSB = 16) so that

$$0\ 0\ 0\ 0\ 0 = -16$$

$$\text{and}\quad 1\ 1\ 1\ 1\ 1 = +15$$

Also evaluate each overflow as being equal to the MSB. Then for the examples previously chosen, in decimal, the value of the left-hand integral is simply the number of additions times the value of the number or 31(-16) = -496; also, 31 negative overflows were obtained for a value of 31(-16) = -496. For the right-hand integral the value is 31(15) = 465. This value may be obtained by algebraically summing the overflows and any remainder left after the additions have been completed.

```
pos ovf =  + 30
neg ovf =  -  1
-----------------
net ovf =   29(+16) + 1  = 465
            Value    Remainder
            of ovf
```

In a practical application, since it is inconvenient to inspect the remainder contained within the shift register, a particular problem should be scaled to produce a rapid overflow rate, thereby making the register contents small compared with the number of overflows (recall that two overflows are equivalent to the entire contents of the register).

The computations performed by the fluid amplifier digital integrator were simulated with a digital computer. The results of the computer runs are shown in Tables 2-1 through 2-5. These runs show the expected performance of the fluid amplifier hardware and thus are used as a "standard" for comparison of the experimental results (presented in Section 4). The tables list values such as positive, negative and net overflows and overflow ratios which are measured outputs from the experimental studies. Specifically, the computer results presented in Table 2-2 and the experimental results presented in Section 4 can be compared since the same value of Y (1 0 1 0 1) is used in both. In the computer results, Tables 2-1 through 2-5, several typical five-bit numbers are integrated as a function of time. The Y value in each run is held constant, and thus is analogous to applying a step input to the integrator and obtaining a ramp response. The approximate integral is obtained by merely multiplying the net overflows by 16. All of the first twenty iterations are shown (corresponding to approximately 1 second of integrator operation); thereafter, only data every 20 iterations is presented. On the average, the approximate integral is 16 units, or one-half of the register contents (same as Absolute Error) less than the exact integral. Also notice that the binary number associated with the number -5 is merely the complement of the binary number associated with +5.

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 1

J=NUMBER OF PRINT INCREMENTS= 20

K=VALUE OF ORDINATE= 13 BINARY VALUE=11101

M=NUMBER OF BITS IN WORD= 5

TABLE 2-1
$Y_L = 13$

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
1	13	-16	29	223.077	0	1	-1	0.
2	26	0	26	100.000	1	1	0	1.000
3	39	16	23	58.974	2	1	1	2.000
4	52	32	20	38.462	3	1	2	3.000
5	65	48	17	26.154	4	1	3	4.000
6	78	64	14	17.949	5	1	4	5.000
7	91	80	11	12.088	6	1	5	6.000
8	104	96	8	7.692	7	1	6	7.000
9	117	112	5	4.274	8	1	7	8.000
10	130	128	2	1.538	9	1	8	9.000
11	143	112	31	21.678	9	2	7	4.500
12	156	128	28	17.949	10	2	8	5.000
13	169	144	25	14.793	11	2	9	5.500
14	182	160	22	12.088	12	2	10	6.000
15	195	176	19	9.744	13	2	11	6.500
16	208	192	16	7.692	14	2	12	7.000
17	221	208	13	5.882	15	2	13	7.500
18	234	224	10	4.274	16	2	14	8.000
19	247	240	7	2.834	17	2	15	8.500
20	260	256	4	1.538	18	2	16	9.000

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 20

J=NUMBER OF PRINT INCREMENTS= 50

K=VALUE OF ORDINATE= 13 BINARY VALUE=11101

M=NUMBER OF BITS IN WORD= 5

TABLE 2-1
- Continued

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
20	260	256	4	1.538	18	2	16	9.000
40	520	512	8	1.538	36	4	32	9.000
60	780	768	12	1.538	54	6	48	9.000
80	1040	1024	16	1.538	72	8	64	9.000
100	1300	1280	20	1.538	90	10	80	9.000
120	1560	1536	24	1.538	108	12	96	9.000
140	1820	1792	28	1.538	126	14	112	9.000
160	2080	2080	0	0.	145	15	130	9.667
180	2340	2336	4	0.171	163	17	146	9.588
200	2600	2592	8	0.308	181	19	162	9.526
220	2860	2848	12	0.420	199	21	178	9.476
240	3120	3104	16	0.513	217	23	194	9.435
260	3380	3360	20	0.592	235	25	210	9.400
280	3640	3616	24	0.659	253	27	226	9.370
300	3900	3872	28	0.718	271	29	242	9.345
320	4160	4160	0	0.	290	30	260	9.667
340	4420	4416	4	0.090	308	32	276	9.625
360	4680	4672	8	0.171	326	34	292	9.588
380	4940	4928	12	0.243	344	36	308	9.556
400	5200	5184	16	0.308	362	38	324	9.526
420	5460	5440	20	0.366	380	40	340	9.500
440	5720	5696	24	0.420	398	42	356	9.476
460	5980	5952	28	0.468	416	44	372	9.455
480	6240	6240	0	0.	435	45	390	9.667
500	6500	6496	4	0.062	453	47	406	9.638
520	6760	6752	8	0.118	471	49	422	9.612
540	7020	7008	12	0.171	489	51	438	9.588
560	7280	7264	16	0.220	507	53	454	9.566
580	7540	7520	20	0.265	525	55	470	9.545
600	7800	7776	24	0.308	543	57	486	9.526
620	8060	8032	28	0.347	561	59	502	9.508
640	8320	8320	0	0.	580	60	520	9.667
660	8580	8576	4	0.047	598	62	536	9.645
680	8840	8832	8	0.090	616	64	552	9.625
700	9100	9088	12	0.132	634	66	568	9.606
720	9360	9344	16	0.171	652	68	584	9.588
740	9620	9600	20	0.208	670	70	600	9.571
760	9880	9856	24	0.243	688	72	616	9.556
780	10140	10112	28	0.276	706	74	632	9.541
800	10400	10400	0	0.	725	75	650	9.667
820	10660	10656	4	0.038	743	77	666	9.649
840	10920	10912	8	0.073	761	79	682	9.633
860	11180	11168	12	0.107	779	81	698	9.617
880	11440	11424	16	0.140	797	83	714	9.602
900	11700	11680	20	0.171	815	85	730	9.588
920	11960	11936	24	0.201	833	87	746	9.575
940	12220	12192	28	0.229	851	89	762	9.562
960	12480	12480	0	0.	870	90	780	9.667
980	12740	12736	4	0.031	888	92	796	9.652
1000	13000	12992	8	0.062	906	94	812	9.638

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 1

J=NUMBER OF PRINT INCREMENTS= 20

K=VALUE OF ORDINATE= 5 BINARY VALUE=10101

M=NUMBER OF BITS IN WORD= 5

TABLE 2-2

$Y_L = 5$

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
1	5	-16	21	420.000	0	1	-1	0.
2	10	0	10	100.000	1	1	0	1.000
3	15	-16	31	206.667	1	2	-1	0.500
4	20	0	20	100.000	2	2	0	1.000
5	25	16	9	36.000	3	2	1	1.500
6	30	0	30	100.000	3	3	0	1.000
7	35	16	19	54.286	4	3	1	1.333
8	40	32	8	20.000	5	3	2	1.667
9	45	16	29	64.444	5	4	1	1.250
10	50	32	18	36.000	6	4	2	1.500
11	55	48	7	12.727	7	4	3	1.750
12	60	32	28	46.667	7	5	2	1.400
13	65	48	17	26.154	8	5	3	1.600
14	70	64	6	8.571	9	5	4	1.800
15	75	48	27	36.000	9	6	3	1.500
16	80	64	16	20.000	10	6	4	1.667
17	85	80	5	5.882	11	6	5	1.833
18	90	64	26	28.889	11	7	4	1.571
19	95	80	15	15.789	12	7	5	1.714
20	100	96	4	4.000	13	7	6	1.857

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 20

J=NUMBER OF PRINT INCREMENTS= 50

K=VALUE OF ORDINATE= 5 BINARY VALUE=10101

M=NUMBER OF BITS IN WORD= 5

TABLE 2-2
- Continued

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
20	100	96	4	4.000	13	7	6	1.857
40	200	192	8	4.000	26	14	12	1.857
60	300	288	12	4.000	39	21	18	1.857
80	400	384	16	4.000	52	28	24	1.857
100	500	480	20	4.000	65	35	30	1.857
120	600	576	24	4.000	78	42	36	1.857
140	700	672	28	4.000	91	49	42	1.857
160	800	800	0	0.	105	55	50	1.909
180	900	896	4	0.444	118	62	56	1.903
200	1000	992	8	0.800	131	69	62	1.899
220	1100	1088	12	1.091	144	76	68	1.895
240	1200	1184	16	1.333	157	83	74	1.892
260	1300	1280	20	1.538	170	90	80	1.889
280	1400	1376	24	1.714	183	97	86	1.887
300	1500	1472	28	1.867	196	104	92	1.885
320	1600	1600	0	0.	210	110	100	1.909
340	1700	1696	4	0.235	223	117	106	1.906
360	1800	1792	8	0.444	236	124	112	1.903
380	1900	1888	12	0.632	249	131	118	1.901
400	2000	1984	16	0.800	262	138	124	1.899
420	2100	2080	20	0.952	275	145	130	1.897
440	2200	2176	24	1.091	288	152	136	1.895
460	2300	2272	28	1.217	301	159	142	1.893
480	2400	2400	0	0.	315	165	150	1.909
500	2500	2496	4	0.160	328	172	156	1.907
520	2600	2592	8	0.308	341	179	162	1.905
540	2700	2688	12	0.444	354	186	168	1.903
560	2800	2784	16	0.571	367	193	174	1.902
580	2900	2880	20	0.690	380	200	180	1.900
600	3000	2976	24	0.800	393	207	186	1.899
620	3100	3072	28	0.903	406	214	192	1.897
640	3200	3200	0	0.	420	220	200	1.909
660	3300	3296	4	0.121	433	227	206	1.907
680	3400	3392	8	0.235	446	234	212	1.906
700	3500	3488	12	0.343	459	241	218	1.905
720	3600	3584	16	0.444	472	248	224	1.903
740	3700	3680	20	0.541	485	255	230	1.902
760	3800	3776	24	0.632	498	262	236	1.901
780	3900	3872	28	0.718	511	269	242	1.900
800	4000	4000	0	0.	525	275	250	1.909
820	4100	4096	4	0.098	538	282	256	1.908
840	4200	4192	8	0.190	551	289	262	1.907
860	4300	4288	12	0.279	564	296	268	1.905
880	4400	4384	16	0.364	577	303	274	1.904
900	4500	4480	20	0.444	590	310	280	1.903
920	4600	4576	24	0.522	603	317	286	1.902
940	4700	4672	28	0.596	616	324	292	1.901
960	4800	4800	0	0.	630	330	300	1.909
980	4900	4896	4	0.082	643	337	306	1.908
1000	5000	4992	8	0.160	656	344	312	1.907

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 1

J=NUMBER OF PRINT INCREMENTS= 20

K=VALUE OF ORDINATE= 0 BINARY VALUE=10000

M=NUMBER OF BITS IN WORD= 5

TABLE 2-3

$Y_L = 0$

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
1	0	-16	16	1600.000	0	1	-1	0.
2	0	0	0	0.	1	1	0	1.000
3	0	-16	16	1600.000	1	2	-1	0.500
4	0	0	0	0.	2	2	0	1.000
5	0	-16	16	1600.000	2	3	-1	0.667
6	0	0	0	0.	3	3	0	1.000
7	0	-16	16	1600.000	3	4	-1	0.750
8	0	0	0	0.	4	4	0	1.000
9	0	-16	16	1600.000	4	5	-1	0.800
10	0	0	0	0.	5	5	0	1.000
11	0	-16	16	1600.000	5	6	-1	0.833
12	0	0	0	0.	6	6	0	1.000
13	0	-16	16	1600.000	6	7	-1	0.857
14	0	0	0	0.	7	7	0	1.000
15	0	-16	16	1600.000	7	8	-1	0.875
16	0	0	0	0.	8	8	0	1.000
17	0	-16	16	1600.000	8	9	-1	0.889
18	0	0	0	0.	9	9	0	1.000
19	0	-16	16	1600.000	9	10	-1	0.900
20	0	0	0	0.	10	10	0	1.000

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 20

J=NUMBER OF PRINT INCREMENTS= 50

K=VALUE OF ORDINATE= 0 BINARY VALUE=10000

M=NUMBER OF BITS IN WORD= 5

TABLE 2-3
- Continued

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
20	0	0	0	0.	10	10	0	1.000
40	0	0	0	0.	20	20	0	1.000
60	0	0	0	0.	30	30	0	1.000
80	0	0	0	0.	40	40	0	1.000
100	0	0	0	0.	50	50	0	1.000
120	0	0	0	0.	60	60	0	1.000
140	0	0	0	0.	70	70	0	1.000
160	0	0	0	0.	80	80	0	1.000
180	0	0	0	0.	90	90	0	1.000
200	0	0	0	0.	100	100	0	1.000
220	0	0	0	0.	110	110	0	1.000
240	0	0	0	0.	120	120	0	1.000
260	0	0	0	0.	130	130	0	1.000
280	0	0	0	0.	140	140	0	1.000
300	0	0	0	0.	150	150	0	1.000
320	0	0	0	0.	160	160	0	1.000
340	0	0	0	0.	170	170	0	1.000
360	0	0	0	0.	180	180	0	1.000
380	0	0	0	0.	190	190	0	1.000
400	0	0	0	0.	200	200	0	1.000
420	0	0	0	0.	210	210	0	1.000
440	0	0	0	0.	220	220	0	1.000
460	0	0	0	0.	230	230	0	1.000
480	0	0	0	0.	240	240	0	1.000
500	0	0	0	0.	250	250	0	1.000
520	0	0	0	0.	260	260	0	1.000
540	0	0	0	0.	270	270	0	1.000
560	0	0	0	0.	280	280	0	1.000
580	0	0	0	0.	290	290	0	1.000
600	0	0	0	0.	300	300	0	1.000
620	0	0	0	0.	310	310	0	1.000
640	0	0	0	0.	320	320	0	1.000
660	0	0	0	0.	330	330	0	1.000
680	0	0	0	0.	340	340	0	1.000
700	0	0	0	0.	350	350	0	1.000
720	0	0	0	0.	360	360	0	1.000
740	0	0	0	0.	370	370	0	1.000
760	0	0	0	0.	380	380	0	1.000
780	0	0	0	0.	390	390	0	1.000
800	0	0	0	0.	400	400	0	1.000
820	0	0	0	0.	410	410	0	1.000
840	0	0	0	0.	420	420	0	1.000
860	0	0	0	0.	430	430	0	1.000
880	0	0	0	0.	440	440	0	1.000
900	0	0	0	0.	450	450	0	1.000
920	0	0	0	0.	460	460	0	1.000
940	0	0	0	0.	470	470	0	1.000
960	0	0	0	0.	480	480	0	1.000
980	0	0	0	0.	490	490	0	1.000
1000	0	0	0	0.	500	500	0	1.000

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 1

J=NUMBER OF PRINT INCREMENTS= 20

K=VALUE OF ORDINATE= -5 BINARY VALUE=01011

M=NUMBER OF BITS IN WORD= 5

TABLE 2-4
$Y_L = -5$

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
1	-5	-16	11	-220.000	0	1	-1	0.
2	-10	-32	22	-220.000	0	2	-2	0.
3	-15	-16	1	-6.667	1	2	-1	0.500
4	-20	-32	12	-60.000	1	3	-2	0.333
5	-25	-48	23	-92.000	1	4	-3	0.250
6	-30	-32	2	-6.667	2	4	-2	0.500
7	-35	-48	13	-37.143	2	5	-3	0.400
8	-40	-64	24	-60.000	2	6	-4	0.333
9	-45	-48	3	-6.667	3	6	-3	0.500
10	-50	-64	14	-28.000	3	7	-4	0.429
11	-55	-80	25	-45.455	3	8	-5	0.375
12	-60	-64	4	-6.667	4	8	-4	0.500
13	-65	-80	15	-23.077	4	9	-5	0.444
14	-70	-96	26	-37.143	4	10	-6	0.400
15	-75	-80	5	-6.667	5	10	-5	0.500
16	-80	-96	16	-20.000	5	11	-6	0.455
17	-85	-112	27	-31.765	5	12	-7	0.417
18	-90	-96	6	-6.667	6	12	-6	0.500
19	-95	-112	17	-17.895	6	13	-7	0.462
20	-100	-128	28	-28.000	6	14	-8	0.429

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 20

J=NUMBER OF PRINT INCREMENTS= 50

K=VALUE OF ORDINATE= -5 BINARY VALUE=01011

M=NUMBER OF BITS IN WORD= 5

TABLE 2-4
- Continued

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
20	-100	-128	28	-28.000	6	14	-8	0.429
40	-200	-224	24	-12.000	13	27	-14	0.481
60	-300	-320	20	-6.667	20	40	-20	0.500
80	-400	-416	16	-4.000	27	53	-26	0.509
100	-500	-512	12	-2.400	34	66	-32	0.515
120	-600	-608	8	-1.333	41	79	-38	0.519
140	-700	-704	4	-0.571	48	92	-44	0.522
160	-800	-800	0	0.	55	105	-50	0.524
180	-900	-928	28	-3.111	61	119	-58	0.513
200	-1000	-1024	24	-2.400	68	132	-64	0.515
220	-1100	-1120	20	-1.818	75	145	-70	0.517
240	-1200	-1216	16	-1.333	82	158	-76	0.519
260	-1300	-1312	12	-0.923	89	171	-82	0.520
280	-1400	-1408	8	-0.571	96	184	-88	0.522
300	-1500	-1504	4	-0.267	103	197	-94	0.523
320	-1600	-1600	0	0.	110	210	-100	0.524
340	-1700	-1728	28	-1.647	116	224	-108	0.518
360	-1800	-1824	24	-1.333	123	237	-114	0.519
380	-1900	-1920	20	-1.053	130	250	-120	0.520
400	-2000	-2016	16	-0.800	137	263	-126	0.521
420	-2100	-2112	12	-0.571	144	276	-132	0.522
440	-2200	-2208	8	-0.364	151	289	-138	0.522
460	-2300	-2304	4	-0.174	158	302	-144	0.523
480	-2400	-2400	0	0.	165	315	-150	0.524
500	-2500	-2528	28	-1.120	171	329	-158	0.520
520	-2600	-2624	24	-0.923	178	342	-164	0.520
540	-2700	-2720	20	-0.741	185	355	-170	0.521
560	-2800	-2816	16	-0.571	192	368	-176	0.522
580	-2900	-2912	12	-0.414	199	381	-182	0.522
600	-3000	-3008	8	-0.267	206	394	-188	0.523
620	-3100	-3104	4	-0.129	213	407	-194	0.523
640	-3200	-3200	0	0.	220	420	-200	0.524
660	-3300	-3328	28	-0.848	226	434	-208	0.521
680	-3400	-3424	24	-0.706	233	447	-214	0.521
700	-3500	-3520	20	-0.571	240	460	-220	0.522
720	-3600	-3616	16	-0.444	247	473	-226	0.522
740	-3700	-3712	12	-0.324	254	486	-232	0.523
760	-3800	-3808	8	-0.211	261	499	-238	0.523
780	-3900	-3904	4	-0.103	268	512	-244	0.523
800	-4000	-4000	0	0.	275	525	-250	0.524
820	-4100	-4128	28	-0.683	281	539	-258	0.521
840	-4200	-4224	24	-0.571	288	552	-264	0.522
860	-4300	-4320	20	-0.465	295	565	-270	0.522
880	-4400	-4416	16	-0.364	302	578	-276	0.522
900	-4500	-4512	12	-0.267	309	591	-282	0.523
920	-4600	-4608	8	-0.174	316	604	-288	0.523
940	-4700	-4704	4	-0.085	323	617	-294	0.524
960	-4800	-4800	0	0.	330	630	-300	0.524
980	-4900	-4928	28	-0.571	336	644	-308	0.522
1000	-5000	-5024	24	-0.480	343	657	-314	0.522

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 1

J=NUMBER OF PRINT INCREMENTS= 20

K=VALUE OF ORDINATE= -16 BINARY VALUE=00000

M=NUMBER OF BITS IN WORD= 5

TABLE 2-5
$Y_L = -16$

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
1	-16	-16	0	0.	0	1	-1	0.
2	-32	-32	0	0.	0	2	-2	0.
3	-48	-48	0	0.	0	3	-3	0.
4	-64	-64	0	0.	0	4	-4	0.
5	-80	-80	0	0.	0	5	-5	0.
6	-96	-96	0	0.	0	6	-6	0.
7	-112	-112	0	0.	0	7	-7	0.
8	-128	-128	0	0.	0	8	-8	0.
9	-144	-144	0	0.	0	9	-9	0.
10	-160	-160	0	0.	0	10	-10	0.
11	-176	-176	0	0.	0	11	-11	0.
12	-192	-192	0	0.	0	12	-12	0.
13	-208	-208	0	0.	0	13	-13	0.
14	-224	-224	0	0.	0	14	-14	0.
15	-240	-240	0	0.	0	15	-15	0.
16	-256	-256	0	0.	0	16	-16	0.
17	-272	-272	0	0.	0	17	-17	0.
18	-288	-288	0	0.	0	18	-18	0.
19	-304	-304	0	0.	0	19	-19	0.
20	-320	-320	0	0.	0	20	-20	0.

DIGITAL INTEGRATOR SIMULATION

I=NUMBER OF ITERATIONS PER PRINT INCREMENT= 20

J=NUMBER OF PRINT INCREMENTS= 50

K=VALUE OF ORDINATE= -16 BINARY VALUE=00000

M=NUMBER OF BITS IN WORD= 5

TABLE 2-5
- Continued

ITERATION NUMBER	EXACT INTEGRAL	APPROX INTEGRAL	ABSOLUTE ERROR	PERCENT ERROR	POSITIVE OVERFLOW	NEGATIVE OVERFLOW	NET OVERFLOW	OVERFLOW RATIO
20	-320	-320	0	0.	0	20	-20	0.
40	-640	-640	0	0.	0	40	-40	0.
60	-960	-960	0	0.	0	60	-60	0.
80	-1280	-1280	0	0.	0	80	-80	0.
100	-1600	-1600	0	0.	0	100	-100	0.
120	-1920	-1920	0	0.	0	120	-120	0.
140	-2240	-2240	0	0.	0	140	-140	0.
160	-2560	-2560	0	0.	0	160	-160	0.
180	-2880	-2880	0	0.	0	180	-180	0.
200	-3200	-3200	0	0.	0	200	-200	0.
220	-3520	-3520	0	0.	0	220	-220	0.
240	-3840	-3840	0	0.	0	240	-240	0.
260	-4160	-4160	0	0.	0	260	-260	0.
280	-4480	-4480	0	0.	0	280	-280	0.
300	-4800	-4800	0	0.	0	300	-300	0.
320	-5120	-5120	0	0.	0	320	-320	0.
340	-5440	-5440	0	0.	0	340	-340	0.
360	-5760	-5760	0	0.	0	360	-360	0.
380	-6080	-6080	0	0.	0	380	-380	0.
400	-6400	-6400	0	0.	0	400	-400	0.
420	-6720	-6720	0	0.	0	420	-420	0.
440	-7040	-7040	0	0.	0	440	-440	0.
460	-7360	-7360	0	0.	0	460	-460	0.
480	-7680	-7680	0	0.	0	480	-480	0.
500	-8000	-8000	0	0.	0	500	-500	0.
520	-8320	-8320	0	0.	0	520	-520	0.
540	-8640	-8640	0	0.	0	540	-540	0.
560	-8960	-8960	0	0.	0	560	-560	0.
580	-9280	-9280	0	0.	0	580	-580	0.
600	-9600	-9600	0	0.	0	600	-600	0.
620	-9920	-9920	0	0.	0	620	-620	0.
640	-10240	-10240	0	0.	0	640	-640	0.
660	-10560	-10560	0	0.	0	660	-660	0.
680	-10880	-10880	0	0.	0	680	-680	0.
700	-11200	-11200	0	0.	0	700	-700	0.
720	-11520	-11520	0	0.	0	720	-720	0.
740	-11840	-11840	0	0.	0	740	-740	0.
760	-12160	-12160	0	0.	0	760	-760	0.
780	-12480	-12480	0	0.	0	780	-780	0.
800	-12800	-12800	0	0.	0	800	-800	0.
820	-13120	-13120	0	0.	0	820	-820	0.
840	-13440	-13440	0	0.	0	840	-840	0.
860	-13760	-13760	0	0.	0	860	-860	0.
880	-14080	-14080	0	0.	0	880	-880	0.
900	-14400	-14400	0	0.	0	900	-900	0.
920	-14720	-14720	0	0.	0	920	-920	0.
940	-15040	-15040	0	0.	0	940	-940	0.
960	-15360	-15360	0	0.	0	960	-960	0.
980	-15680	-15680	0	0.	0	980	-980	0.
1000	-16000	-16000	0	0.	0	1000	-1000	0.

Section 3
EXPERIMENTAL EQUIPMENT

Figures 3-1 through 3-4 illustrate the digital integrator, the pneumatic signal generator, and the test setup. As outlined in the Summary (Section 1) the test equipment fabrication was limited to one time integrator; two identical time integrators made up a complete integrator capable of integrating one variable with respect to a second variable. As will be discussed later, the input from the eliminated time integrator was obtained from a signal generator.

Figure 3-1, prepared from the photo negatives used to fabricate the Dycril integrator elements, is 25% full-size and illustrates the circuit packaging assembly as well as the gas flow paths for the complementer, adder, and shift register circuits. These circuits were described in Section 2 and shown in Figure 2-3. The various input and output channels (e.g., Y_L, R, Δx, etc.) are the same as those shown in Figure 2-3. In Figure 3-1 the signal inputs to the adder and complementer circuits are applied to the left NOR element of each. These input elements are at the top of the respective assemblies as shown in Figure 3-2. Supply ports, reset, and clock ports appear on the far right in Figure 3-1 and are physically located at the bottom of the assemblies as shown in Figure 3-2. In both the adder and complementer assemblies, the outputs (Y_C, $\overline{Y}_C$, R_L, $\overline{R}_L$) are taken from the center of the stacks. In the adder the outputs + Δz and - Δz are sensed at the bottom manifold plate. Three stages of the five-stage shift register are shown at the bottom of the diagram (Figure 3-1). The adder output R_L and $\overline{R}_L$ is applied at the point labeled Register Input at the right of the diagram, and the Register Output (which connects to the adder at R and $\overline{R}$) is at the left side of the diagram. Figure 3-2 is four views of the breadboard digital integrator showing the interconnection between the various assemblies. The pressure settings used for operations of the integrator are listed in Table 3-1.

The input to the integrator was supplied by the signal generator shown in Figure 3-3. The signal generator consists of two disks mounted on a common shaft which is driven by a d.c. motor. Each disk passes between a nozzle

and receiver; thus a pressure signal is generated when a slot on the disk uncovers a receiver. The top disk was employed to generate the integrator input, Y_L, and its inversion, $\overline{Y}_L$. Any 5-bit binary number can be used for Y_L by proper arrangement of the open slots on the disk. The bottom disk provided the clock, read, and reset signals necessary for integrator operation. Mounting of the two disks on a single shaft insured exact synchronization of the input and clock, read, and reset signals. The proper "d.c." level for the clock pressure was established by using an aspirator mounted on the generator.

Figure 3-4 summarizes the salient features of the experimental test setup. The digital integrator and signal generator previously described are shown in the right portion of the Figure. The various summing, differencing, and filtering operations shown in the left portion of the Figure were accomplished with a small electronic analog computer to conveniently supply test signals. The waveshaping amplifiers were employed to remove d.c. components from the overflow signals ($+\Delta z$ and $-\Delta z$) from the transducer amplifiers prior to applying these signals to the summing and differencing amplifiers and then to an electronic counter. Both negative, positive, and total overflow could be counted. Since an overflow is produced every iteration, the signal generator frequency could be quite accurately set by counting the total overflow against a known time base. Overflow difference was available at the left-hand amplifier for display on an oscillograph and oscilloscope. As will be subsequently described in Section 4, other instrument channels were employed as required to illustrate various intermediate steps during a computation. Typical waveforms are also shown in Section 4.

TABLE 3-1

DIGITAL INTEGRATOR PRESSURE SETTINGS

Complementer Circuit	
Main Supply	5.0 psig
Gate Supply	1.6
Main Clock	+ 1.35 - .25
Reset	3.0
Adder Circuit	
Main Supply	3.5
Gate Supply	1.6
Main Clock	+ 1.35 - .25
Reset	3.0
Overflow Clock	+ .85 - 1.0
Shift Register Circuit	
Flip-Flop Supply	3.3
Gate Supply	4.9
Clock	+ .85 - 1.0
Buffer Amplifier	
Main Supply	3.5 psig

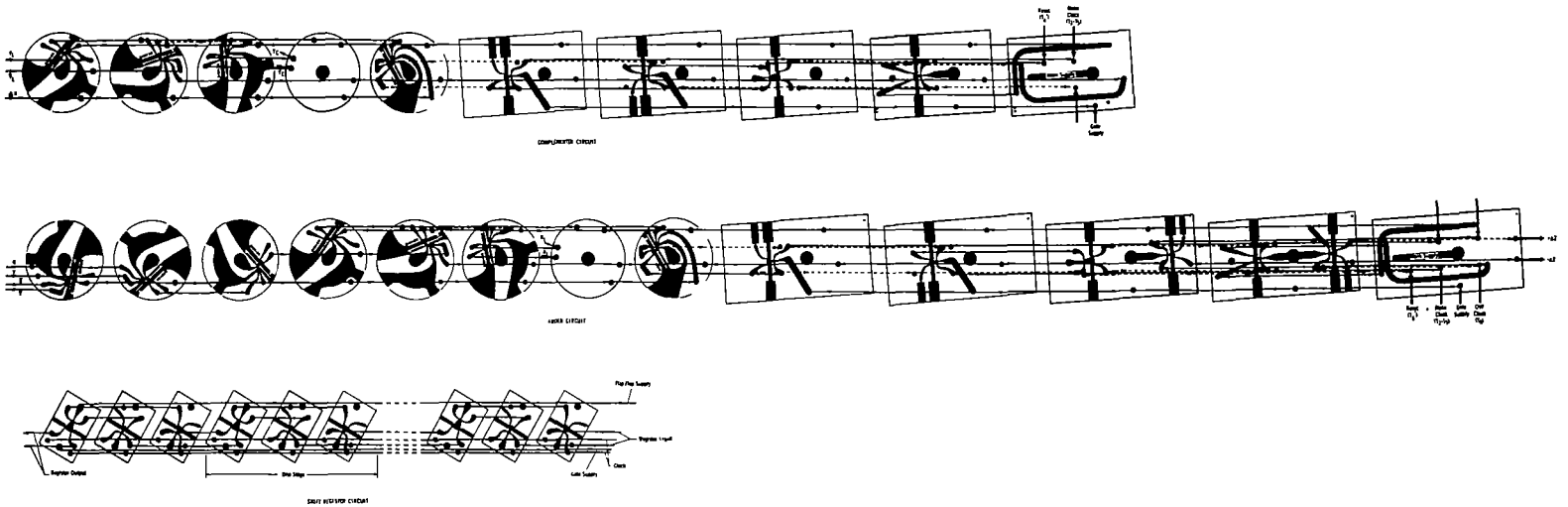
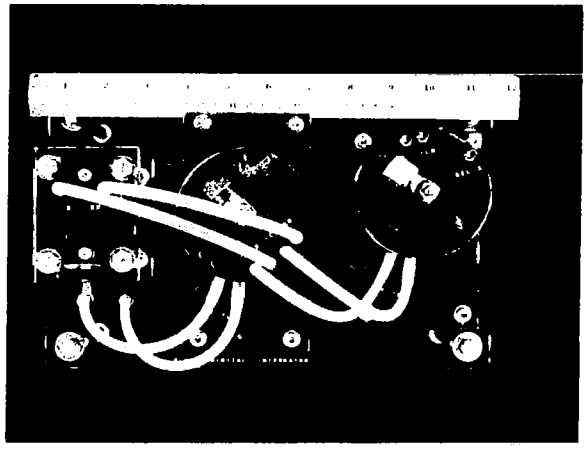
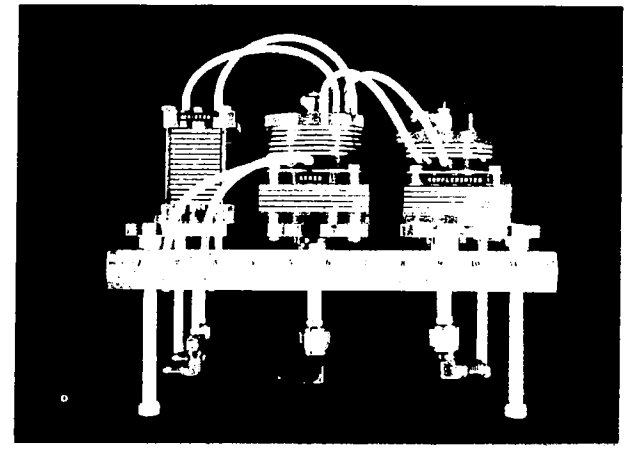


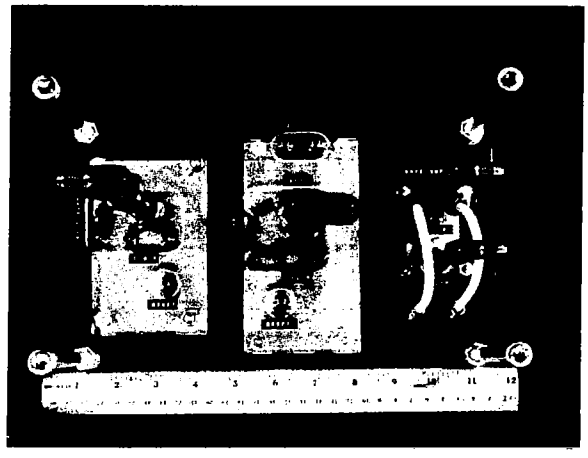
Figure 3-1. Digital Integrator Assembly.
(scale - 0.25 x full size)



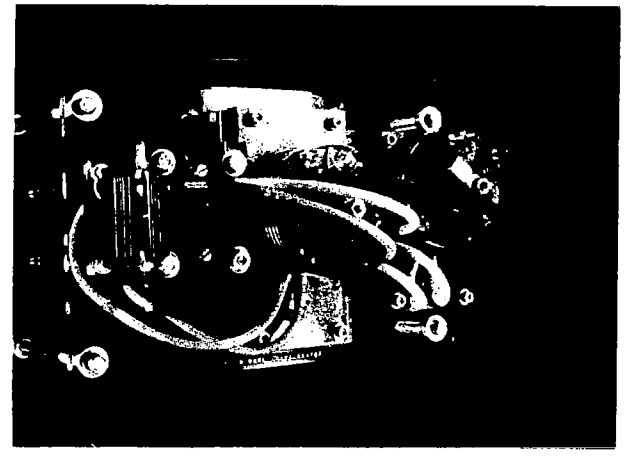
TOP VIEW



FRONT VIEW



BOTTOM VIEW



SIDE VIEW

FLUID DIGITAL INTEGRATOR

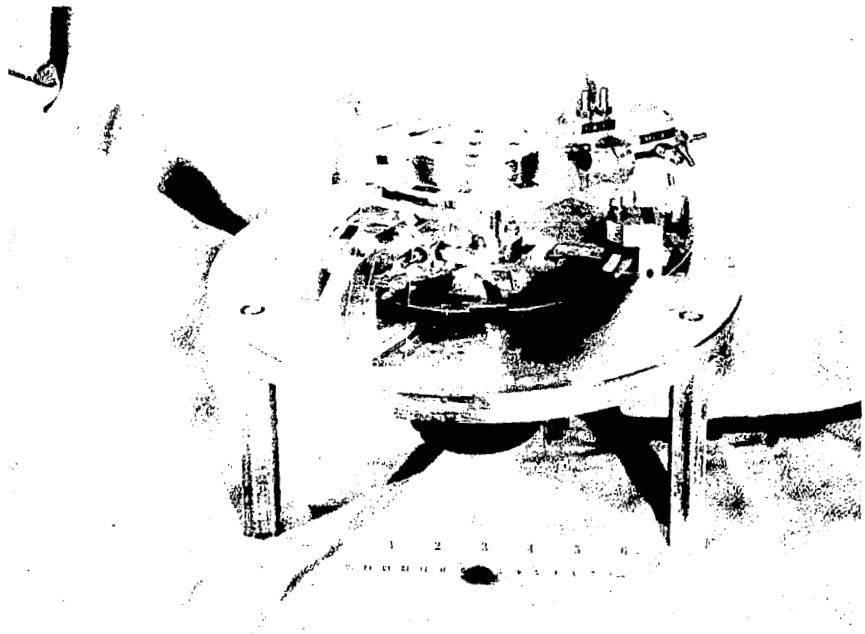


Figure 3-3. Fluid-Signal Generator

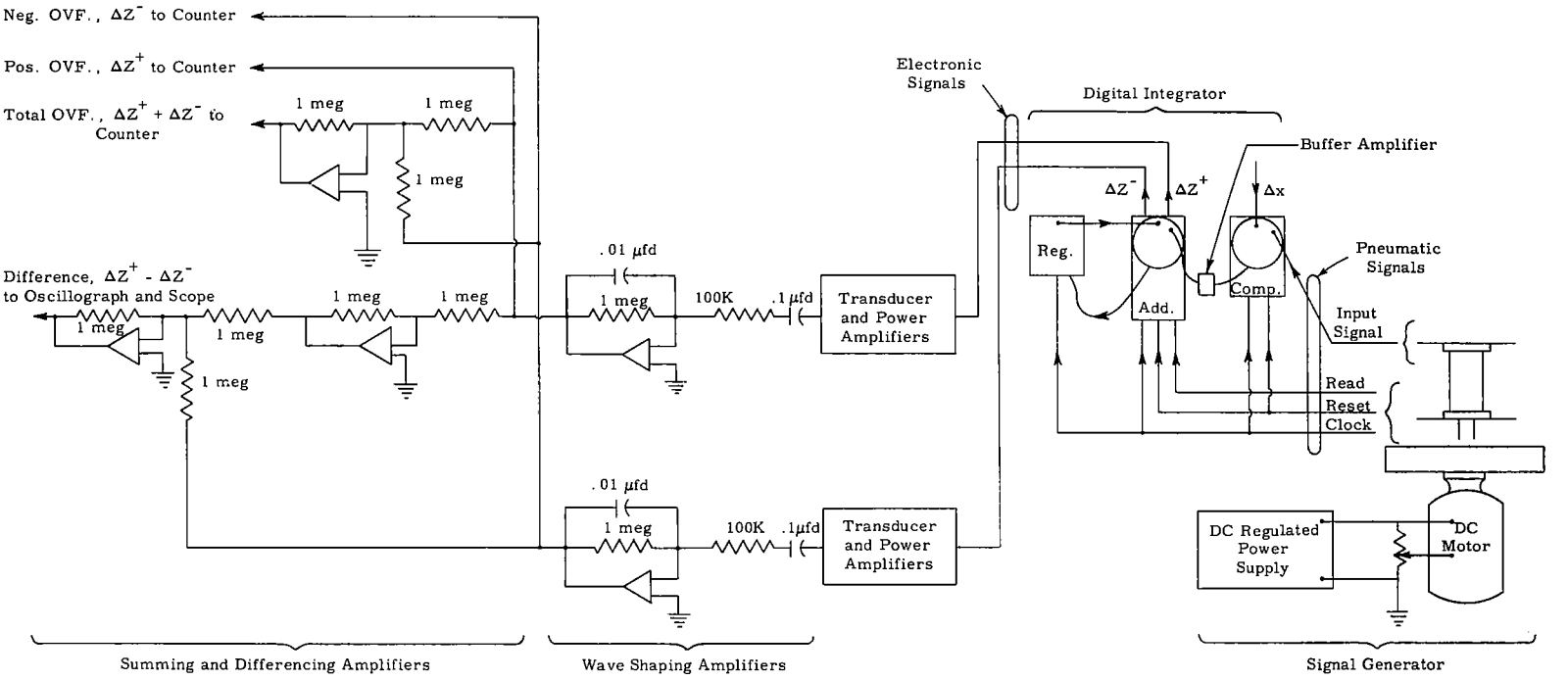


Figure 3.4. Test Setup Schematic

Section 4

EXPERIMENTAL RESULTS

Figures 4-3 through 4-12 and Table 4-1 present typical data taken during check-out of the integrator. In all tests the clock frequency was maintained at approximately 100 cycles per second and supply pressures were maintained at the values listed in Section 3.

The first four runs were conducted open-loop in that the shift register output was not fed back to the adder as shown in Figure 4-1. In these tests the signal applied at R was 0 0 0 0 0 or 1 1 1 1 1 so that the input test signal (Y_L), after passing through the complementer, could either be added to or subtracted from these numbers. Thus, the integrator without feedback was operated as an adder or subtractor with the adder output being shifted serially through the register. These runs served to illustrate some typical wave forms as well as a check-out of interconnected integrator components. The last 8 runs were conducted in a cloosed-loop configuration with the circuit connected as shown in Figure 4-2. The input, Y_L, was generated by means of a binary-coded disk as previously described in Section 3. The value of Y_L arbitrarily was chosen to be 1 0 1 0 1 and the results to be presented may be compared with the computer runs of Table 2-1. A short description of each run is presented in the following paragraphs.

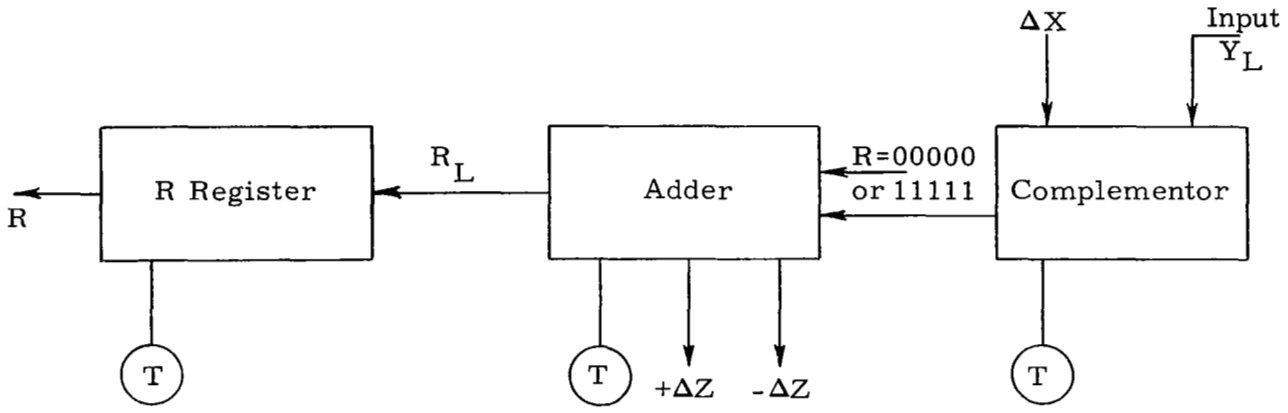


Figure 4.1. Open-Loop Test Arrangement

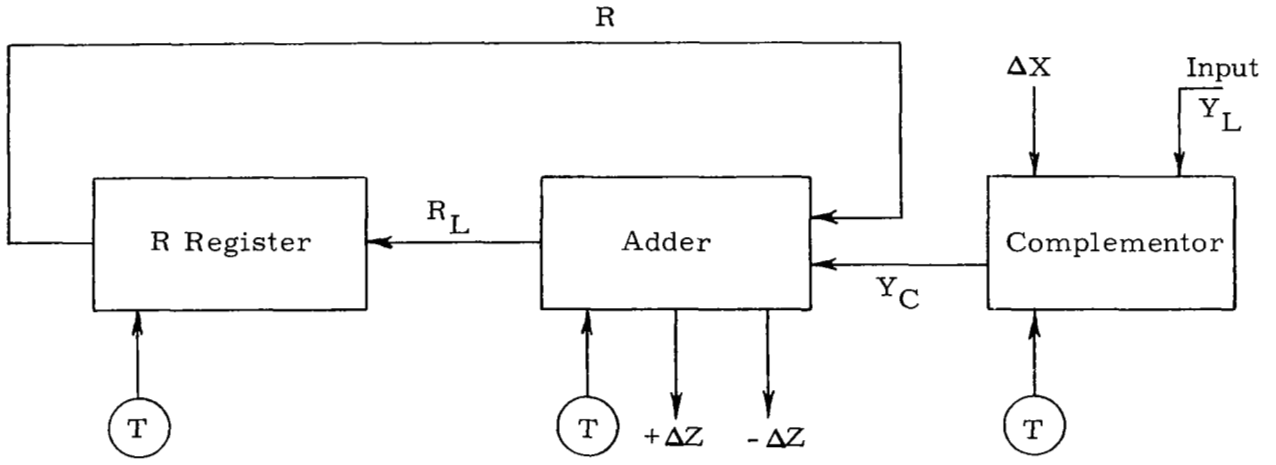
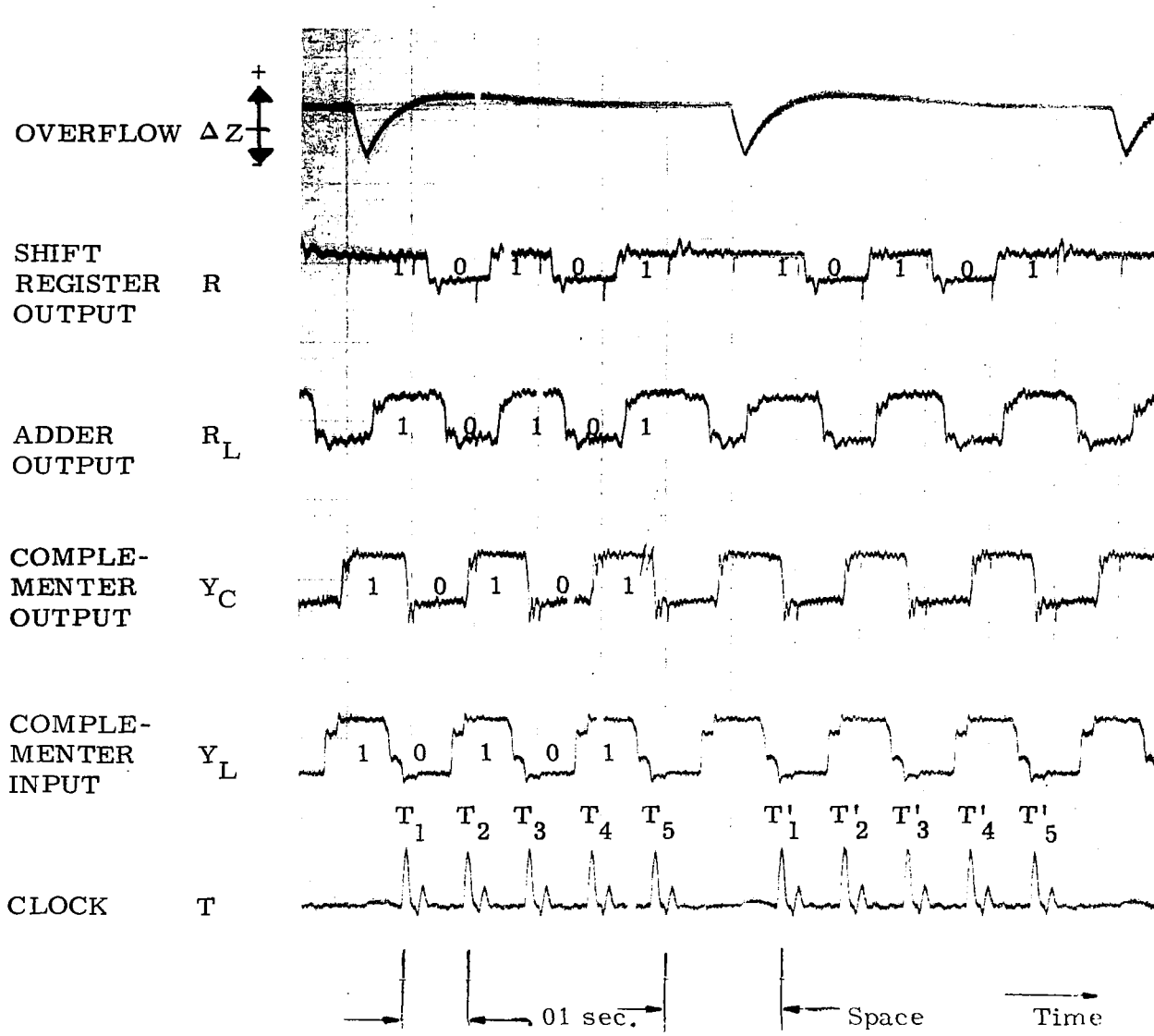


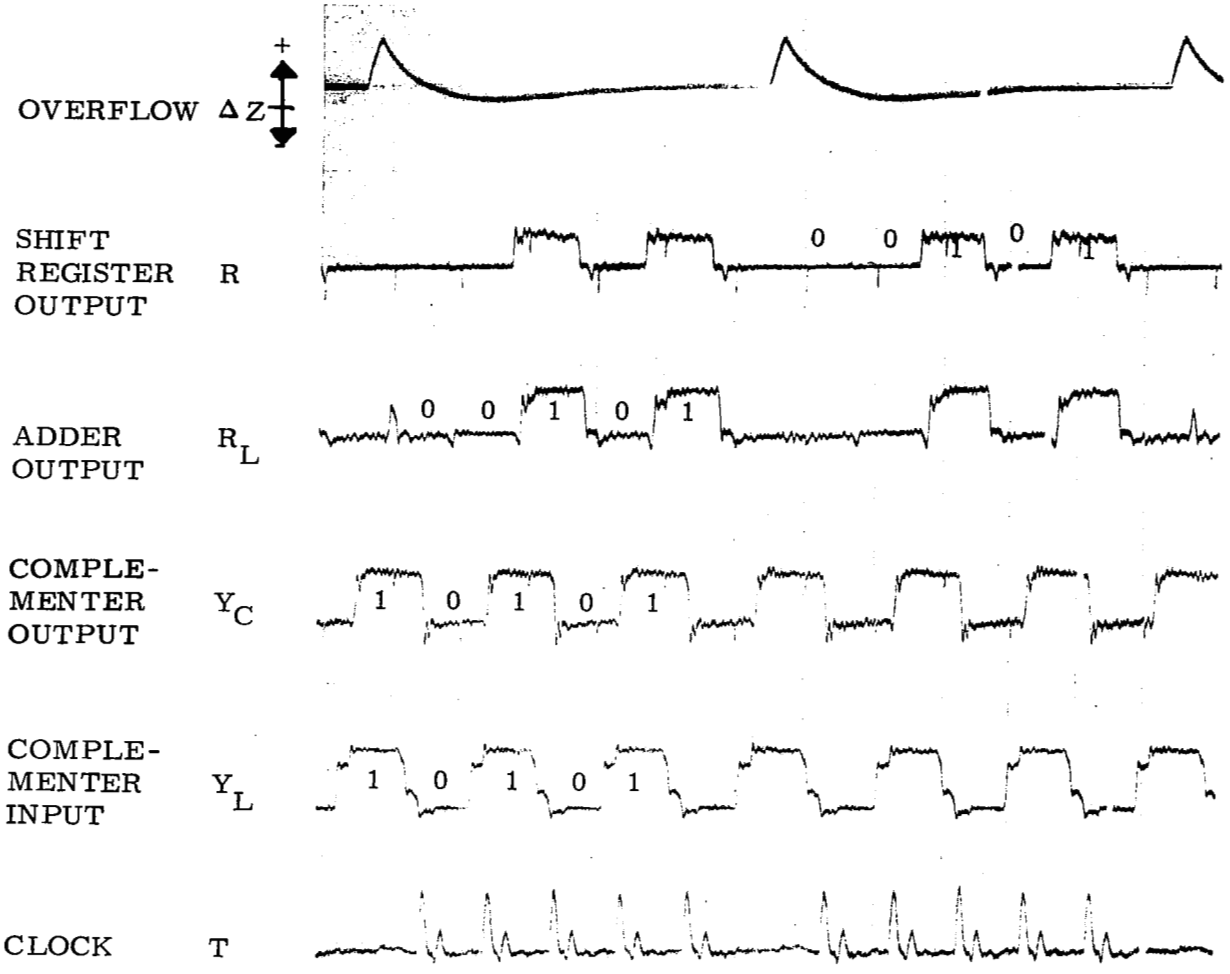
Figure 4.2. Closed-Loop Test Arrangement.



	T_5	T_4	T_3	T_2	T_1
Y_C =	1	0	1	0	1
+	0	0	0	0	0
R_L =	1	0	1	0	1

Figure 4.3
Run 1

Run 1 - In this run Y_L was added to R = 00000 (R = 00000 was produced by reducing the pressure at the "R" port to zero). Y_L, applied at the complementer input, was passed directly through the complementer since a steady pressure was applied at the Δx port indicating a + Δx. Note that the complementer input, Y_L, complementer output, Y_C, and adder output, R_L, waveforms are all similar but shifted in phase due to signal delay in the elements making up these circuits. For convenience in interpreting the data, a space is allowed between the fifth pulse of one iteration and the first pulse of the following iteration (see lower trace Figure 4-3). The shift-register output-waveform is shifted in phase by one entire iteration since the input signal appears at the output after 5 clock pulses. Its output waveform is slightly different from the input since the register retains the last information (in this case a "1") during the space between iterations. The one iteration delay in the shift register can be seen in Figure 4-3; at time T_1 the adder output (R_L) is 1 which appears as R = 1 at time T_1' (one iteration later) at the register output. The adder output is 0 which appears at the register output as a 0 at time T_2' and so forth through T_5 and T_5'. Thus in this test register output is the same as the input to the system. Since the sum of 10101 and 00000 do not produce a carry at time T_5, all negative overflows are produced (top channel of Figure 4-3).

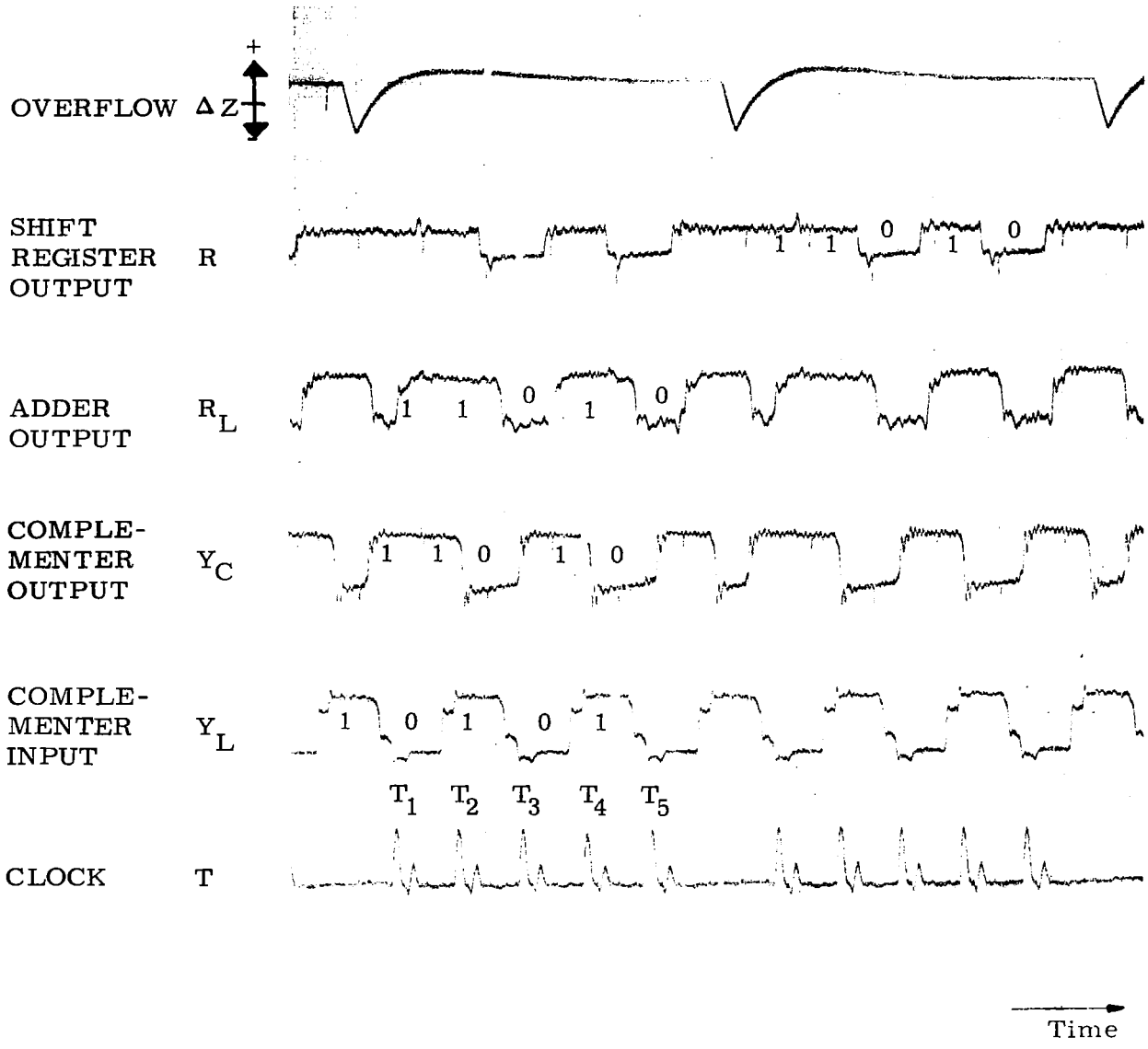


		T_5	T_4	T_3	T_2	T_1
Y_C	=	1	0	1	0	1
	+	1	1	1	1	1
R_L		1	0	1	0	0

1 positive overflow

Figure 4.4
Run 2

Run 2 - This run was similar to Run 1; the input (Y_L = 10101) was passed directly through the complementer (by maintaining a pressure at Δx to provide a $+\Delta x$) to the adder where it was summed with 11111 (steady pressure held at port R) producing the sum, 10100, at the adder output. This number again was shifted through the register to provide the output. As shown in Figure 4-4, the addition of the two numbers produces a sum larger than 5 bits; thus, a positive overflow is generated at the upper channel of Figure 4-4.



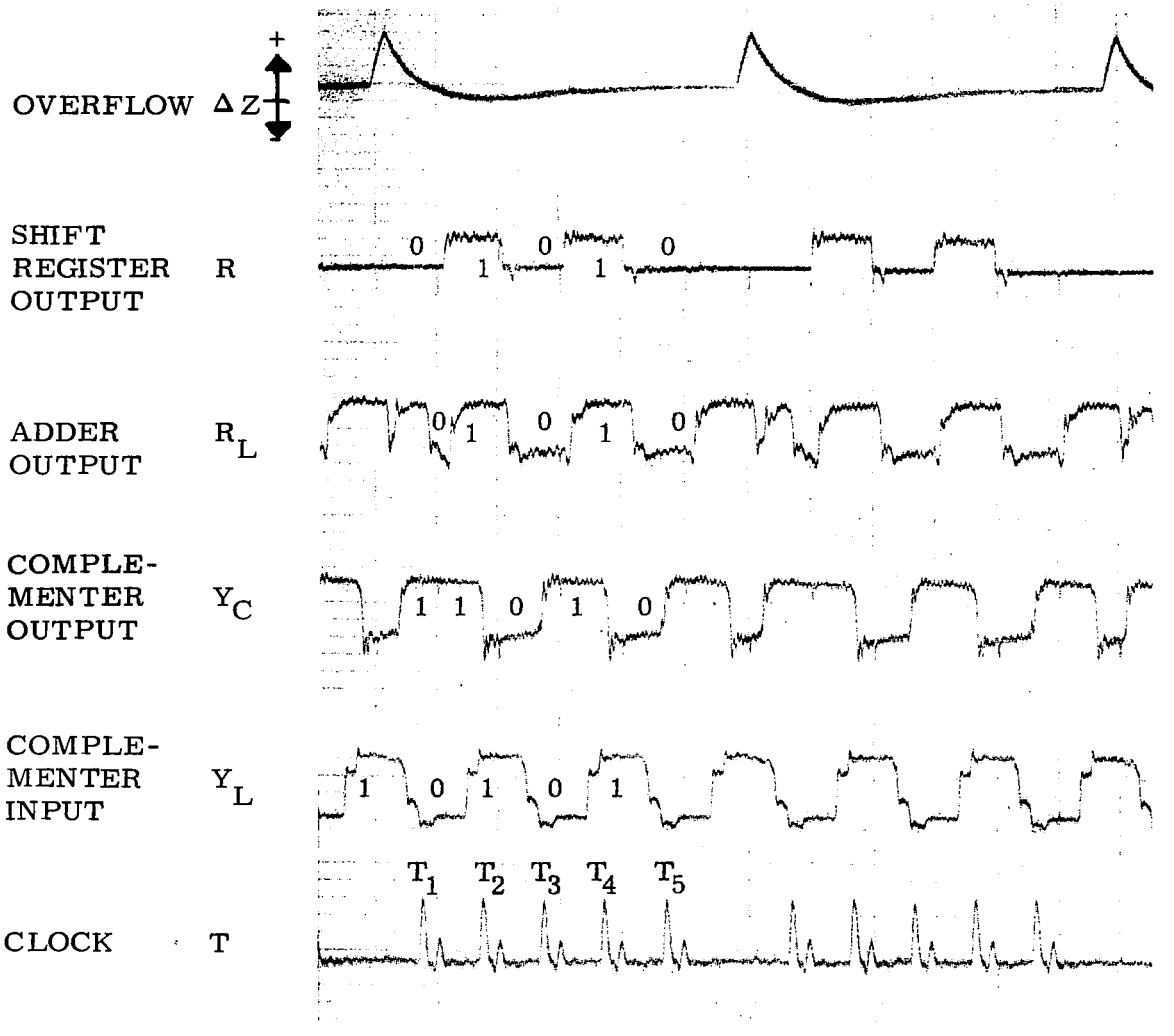
Y_C	=	0	1	0	1	1
	+	0	0	0	0	0
R_L	=	0	1	0	1	1

Figure 4.5
Run 3

Run 3 - In this run as well as Run 4, the open-loop integrator was operated simply as a subtractor. An input of 00000 was applied at R (zero pressure at port "R"), and the input Y_L = 10101 was complemented in passing through the complementer to produce Y_C = 01011 which was then fed to the adder. The results are shown in Figure 4-5.

The waveforms shown illustrate the subtraction of +5 from -16 to yield the correct result of -21. +5 is represented by the input, Y_L = 10101* and -16 is represented by R = 00000. The number -21 is displayed at the adder output, R_L, as R_L = 11010 = -5 and one negative overflow which represents the correct value -21 (-5 -16). As shown on the top channel of Figure 4-5 and also as indicated in the arithmetic shown in the Figure, no carry was generated after the last digits were added corresponding to negative overflow.

* Each binary number must be biased by -16 units as stated in Section 2-3.

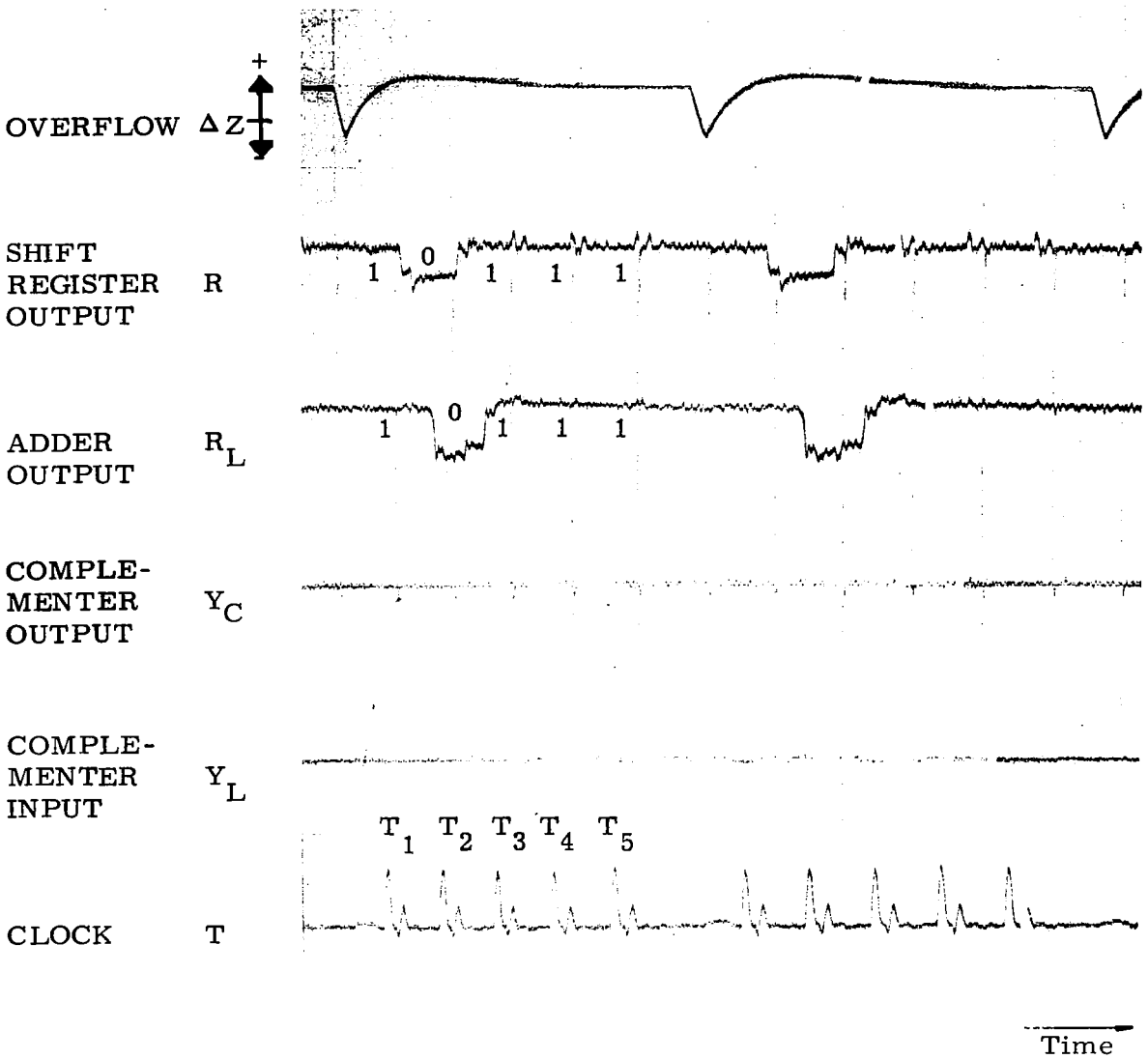


$$
\begin{array}{rccccc}
Y_C = & 0 & 1 & 0 & 1 & 1 \\
+ & 1 & 1 & 1 & 1 & 1 \\
\hline
R_L = 1 & 0 & 1 & 0 & 1 & 0
\end{array}
$$

positive overflow

Figure 4.6
Run 4

Run 4 - In this run the input 10101 was subtracted from 11111 by complementing the input before applying it to the adder. All positive overflows as well as the correct sum are produced at the adder output and then clocked forward into the register.



$$
\begin{array}{rccccc}
R_L = & 1 & 1 & 1 & 0 & 1 \\
+ & 0 & 0 & 0 & 0 & 0 \\
\hline
 & 1 & 1 & 1 & 0 & 1
\end{array}
$$

Figure 4.7
Run 5

Run 5 - This run as well as all succeeding runs was conducted closed-loop. The shift register output was connected back to the adder input as shown in Figure 4-2 to result in the complete circuit. If a number is placed in the register and the integrator input, Y_L, is reduced to zero, this is equivalent to adding zero to the register contents, producing the sum at the adder output, and clocking this sum back into the register. Thus, the same number continues to circulate through the integrator at clock frequency. Figure 4-7, Run 5, presents results obtained in a test of this type. Since no carry is produced after T_5, all negative overflows are generated as shown by the top oscillograph channel.

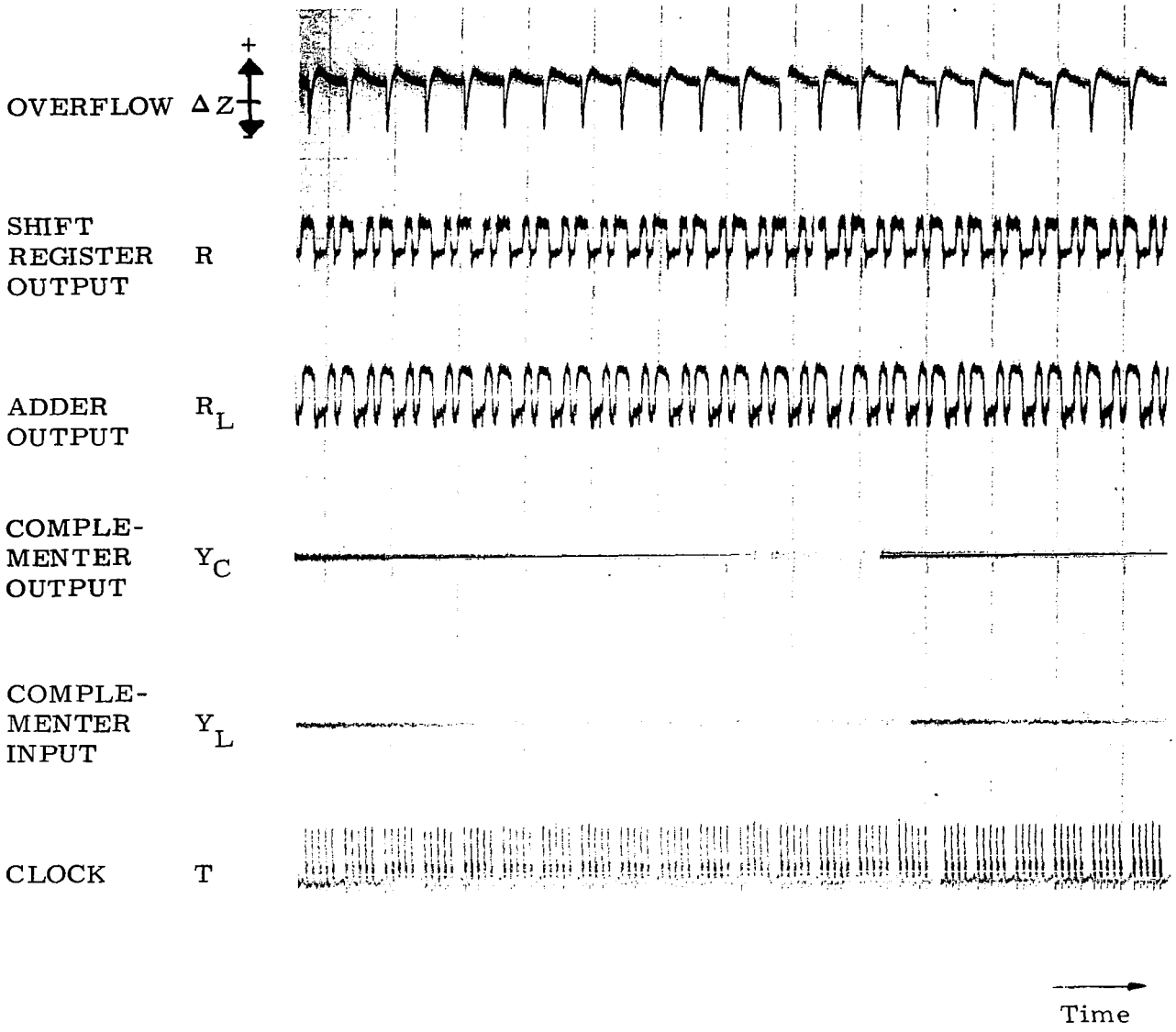
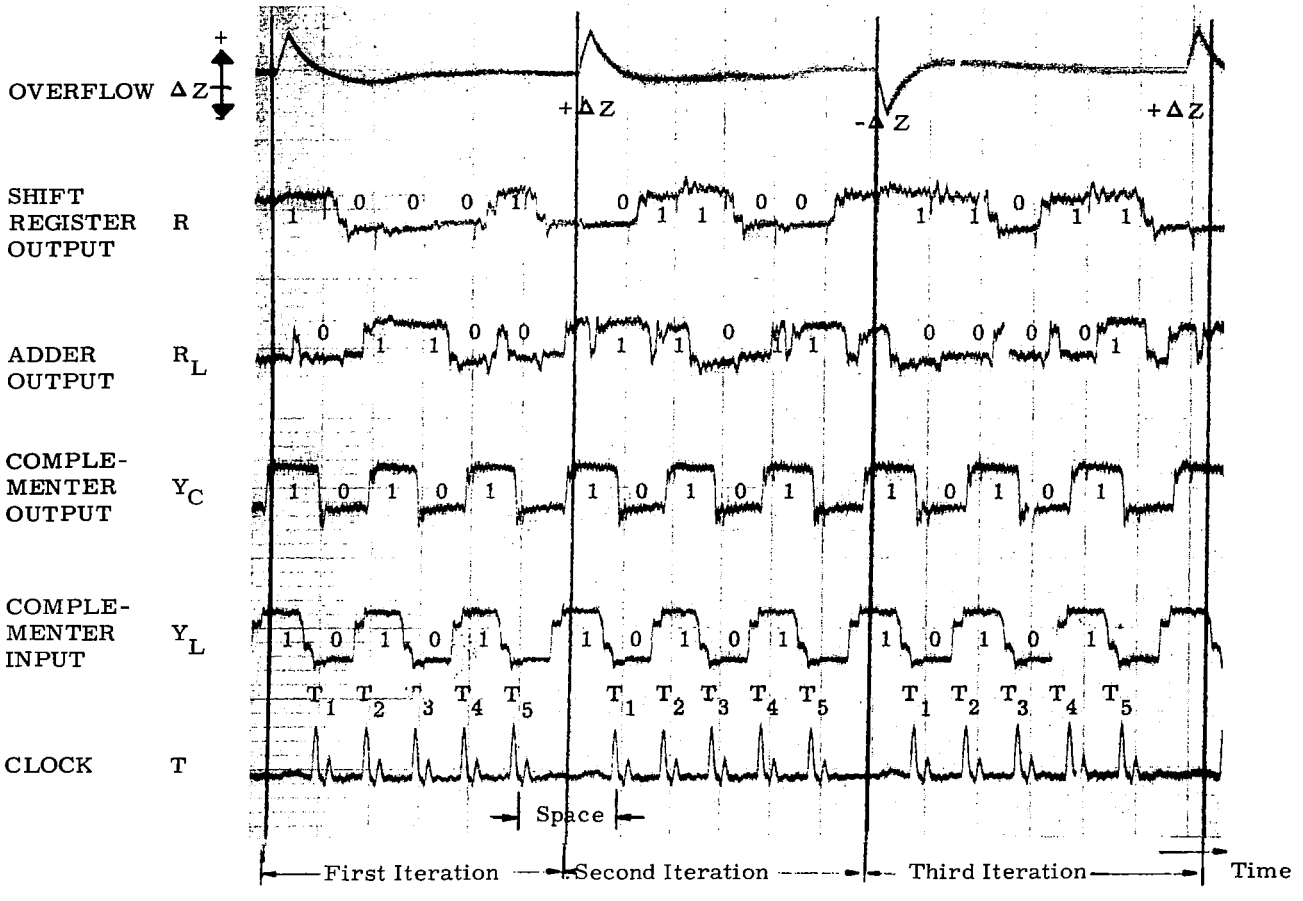


Figure 4.8
Run 6

Run 6 - This Run (Figure 4-8) was similar to Run 5 except that a different number was inserted in the register and the test was carried out for a longer period of time. The data of both Runs 5 and 6 illustrate the memory retention of the integrator.

The remainder of the runs presented show data obtained while operating the circuit as an integrator and connected as shown in Figure 4-2. Inputs of Y_L and Δx are applied to the complementer. In all cases Y_L was the binary number 10101 generated by the pneumatic signal generator running in synchronization with clock frequency. Thus, the signal generator replaced the output of the upper "Y" register of Figure 2-2 and supplied the constant number $Y_L = 10101$. If a pressure is applied at Δx (corresponds to $+\Delta x$), Y_L is passed directly through the complementer and a positive ramp output should be obtained as expected from the time integration of a constant. If Δx is absent (corresponds to $-\Delta x$), Y_L is complemented before being sent to the adder and a negative ramp output of the same slope should be obtained. The ramp outputs are represented by the algebraic accumulation of the $+\Delta z$ and $-\Delta z$ pulses as shown by NET OVERFLOW column in Figure 2-1. The input binary number 10101 produces 1.9 positive Δz overflows for each negative overflow (see OVERFLOW RATIO column, Figure 2-1). An input binary number of 01011 (complement of 10101) should produce 0.526 positive overflows (reciprocal of 1.9) for each negative overflow.



			T_5	T_4	T_3	T_2	T_1	
First Iteration	Comp. Input	Y_L	1	0	1	0	1	
	Comp. Output	Y_C	1	0	1	0	1	(Since ΔX = 1)
	+ Register Output	R	+1	0	0	0	1	
	Adder Output = R_L	1 ($+\Delta Z$)	0	0	1	1	0	(This number fed into shift register to become register output for succeeding iteration)
Second Iteration	Comp. Output	Y_C	1	0	1	0	1	
	+ Register Output	R	+0	0	1	1	0	(Also adder output from last iteration)
	Adder Output	R_L ($+\Delta Z$)	1	1	0	1	1	(No carry developed @ T_5; Therefore - ΔZ)
Third Iteration	Comp. Output	Y_C	1	0	1	0	1	
	+ Register Output	R	+1	1	0	1	1	
	Adder Output	R_L 1 ($+\Delta Z$)	1	0	0	0	0	

Figure 4.9
Run 7

Run 7 - Figure 4-9 presents three iterations by the integrator and the arithmetic processes are shown below the trace. In this test Y_L was not complemented, and the integration thus represented the function $z = \int (+5)\, dt$.

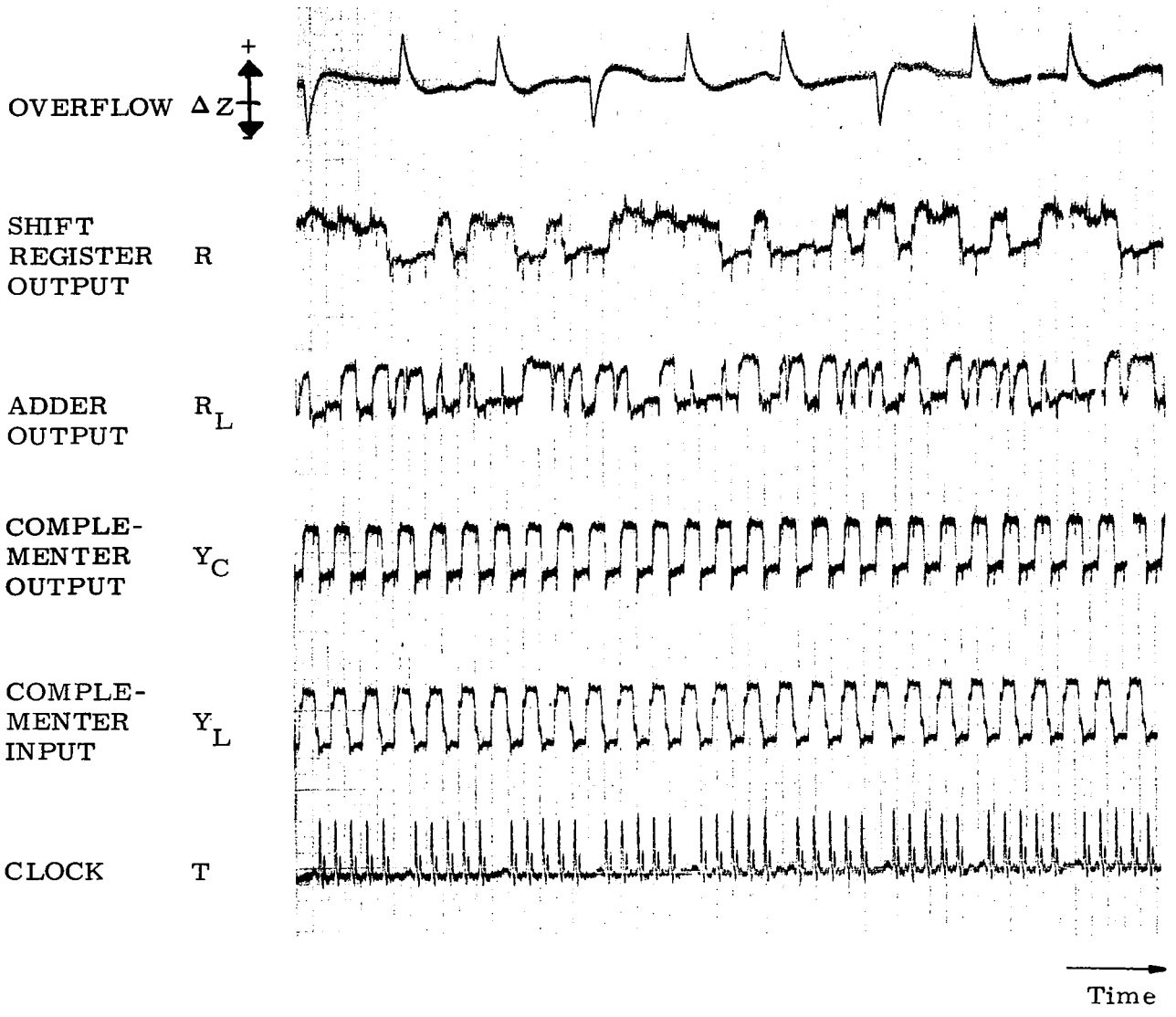
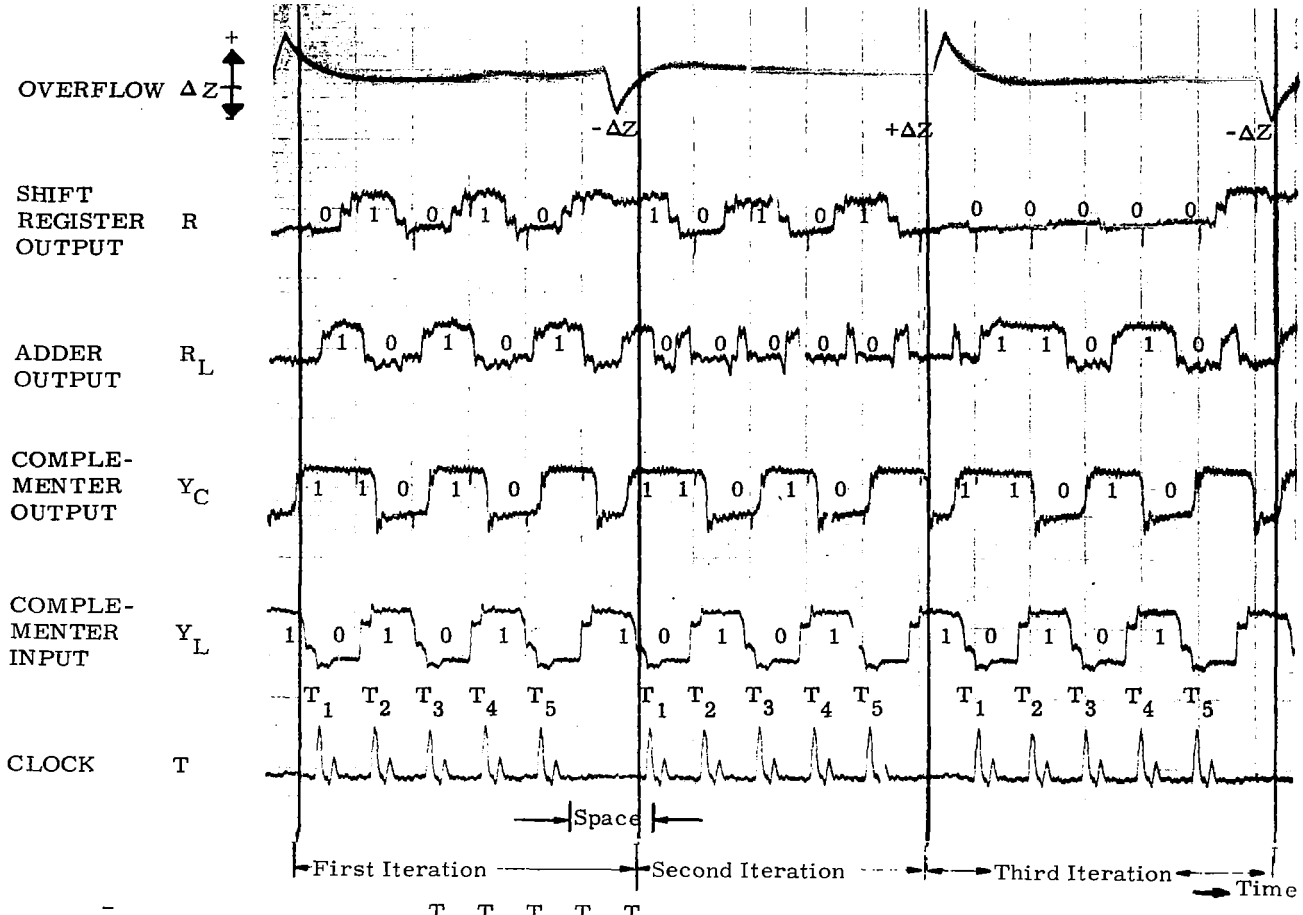


Figure 4.10
Run 8

Run 8 - Figure 4-10 shows a repeat of Run 7 made at a slower recording speed to illustrate the overflow action. Note that over the time interval shown 6 positive and 3 negative overflows are obtained. From Table 2-1 the theoretical value of overflow ratio ($+\Delta z / -\Delta z$) is 1.9.



				T_5	T_4	T_3	T_2	T_1	
First Iteration	Comp. Input	Y_L		1	0	1	0	1	
	Comp. Output	Y_C		0	1	0	1	1	Complement Since ΔX = 0
	+ Register Output	R	+	0	1	0	1	0	
	Adder Output	R_L		1	0	1	0	1	(No carry at T_5 ; hence -ΔZ is generated)
				-ΔZ					
Second Iteration	Comp. Output	Y_C		0	1	0	1	1	
	+ Register Output	R	+	1	0	1	0	1	(Adder output from first iteration)
	Adder Output	R_L		0	0	0	0	0	
				+ΔZ					
Third Iteration	Comp. Output	Y_C		0	1	0	1	1	
	+ Register Output	R	+	0	0	0	0	0	
	Adder Output	R_L		0	1	0	1	1	
				-ΔZ					

Figure 4.11
Run 9

Run 9 - Figure 4-11 is an oscillograph record and detailed arithmetic computation for performing $\int(-5)\,dt$; 3 iterations are shown. To obtain the number -5 for integration Y_L (10101 = +5) is first complemented before passing to the adder; otherwise, the comments of Run 8 apply for this run.

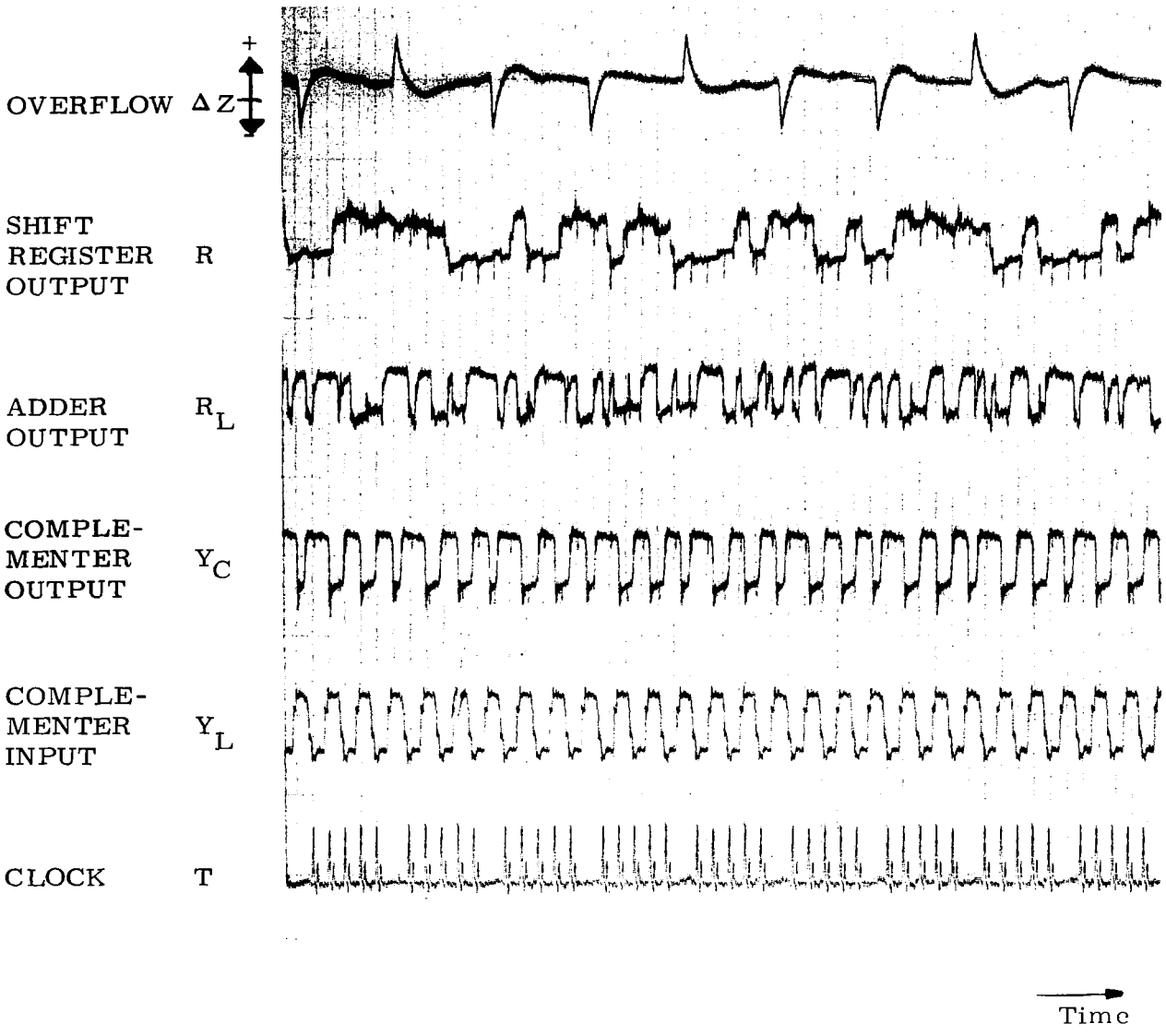


Figure 4.12
Run 10

Run 10 - Figure 4-12 shows a repeat of Run 9 except that the data are taken at a lower recording speed to illustrate the overflow action. As expected approximately 2 negative overflows are obtained for each positive overflow.

TABLE 4-1

SUMMARY OF DIGITAL INTEGRATOR COUNTER RUNS

RUN 11 (Y_L not complemented)				RUN 12 (Y_L complemented)			
Δx	+Δz	-Δz	+Δz/-Δz	Δx	+Δz	-Δz	-Δz/+Δz
↑	111	59	1.88	↑	56	110	1.97
Present	109	60	1.82	Absent	58	112	1.93
	110	57	1.93		60	112	1.87
	110	59	1.87		58	112	1.93
	110	61	1.81		56	111	1.98
	110	59	1.87		58	110	1.90
	109	60	1.82		56	110	1.97
	111	61	1.82		58	113	1.95
	111	59	1.88		60	113	1.88
↓	110	58	1.90	↓	57	114	2.00
Avg + Δz/-Δz =			1.86	Avg - Δz/+Δz =			1.94

Run 11 and Run 12 - All of the runs presented thus far represent data taken over a relatively short time period to illustrate waveforms and the detailed arithmetic operation within the integrator. As indicated earlier, the contents of the register and even the detailed arithmetic operations are of little interest in a practical application. The overflow ($\pm \Delta z$) produced by these computations is of primary importance. To determine "long-time" performance, i.e., accuracy, 2 runs (Runs 11 and 12) were conducted a relatively long period of time and the overflow was counted with an electronic counter. During each of the runs the average clock frequency was 102 cps and the counting period was 10 seconds. Because of the space in clock waveform (to delineate beginning and end of an iteration) approximately 170 overflows instead of 200 are produced in 10 seconds.

Table 4-1 summarizes the data obtained in these "long-time" tests. In Run 11 a positive input $Y_L = 10101$ was integrated. In Run 12, the negative of this input was integrated by reducing the Δx pressure to zero (corresponds to $-\Delta x$) thus complementing Y_L. In both tests, results of 10 consecutive counting periods are presented. No effort was made either to initialize the register contents or to read the register contents before and after the 10 second period. Since the register is capable of holding biased numbers ranging from -16 to +15 (refer to Section 2, Digital Integrator Theory) and the absolute value of each overflow is 16, uncertainty in the remainder contained within the register is equivalent to $\pm$ 1 overflow. Since this uncertainty is present at both the beginning and end of the counting period, the value of the integral is accurate to $\pm$ 2 overflows. According to Table 2-1 for the number being integrated (10101) the overflow ratio should be 1.9. Typically for 171 total overflows and $\pm$2 overflows uncertainty, the nominal number of overflows and deviation associated with the $\pm$2 count tolerance is:

$$\text{Min} = \frac{110}{61} = 1.80$$

$$\text{Nominal} = \frac{112}{59} = 1.90$$

$$\text{Max} = \frac{114}{57} = 2.00$$

Notice that all of the overflow ratios presented in Table 4-1 fall within the above limits. For Run 11 the average overflow ratio is 1.86 and for Run 12 the average overflow ratio is 1.94.

Section 5
CONCLUSIONS AND RECOMMENDATIONS

The results and information obtained in the course of this program are convincing that fluid digital computation systems are feasible. The use of fluid amplifiers appears attractive for many systems because of high tolerance to hostile environments such as vibration, radiation and temperature, and expected reliability. There are inherent disadvantages, however, in the application of these fluid devices to digital computation for space systems. The speed of the fluid amplifier is several orders of magnitude slower than electronic elements, and some form of a fluid power supply will be needed. Therefore, fluid amplifiers are not likely to replace most present electronic digital computation systems, but many specialized applications can use fluid amplifiers advantageously. In these latter applications fluid amplifier implementation should not be simply a one-for-one replacement of the electronic elements. In order to be used advantageously the fluid amplifier components must be designed to perform the required functions in the way most suited to their inherent characteristics. For example, due to the higher speed of electronic devices, the optimum electronic system may use serial implementation with time sharing of a single digital integrator while with fluid amplifiers the optimum system design may be parallel implementation with the use of several digital integrators. Even if there is no special need for reliability and environmental tolerance, the availability of a power source may be a major factor in determining that fluid amplifier computation should be used. For example, in vehicles such as boosters, large quantities of pressurized fluid are available and fluid amplifier computation could be used to eliminate the need for one electrical source. For long-life satellites no convenient source of fluid power is available although concepts for solar-powered fluid power sources have been advanced. Development of these power supplies will proceed only when the need for fluid amplifier computation in satellites becomes more acute.

The major technical accomplishments required in the digital integrator to achieve prototype performance are to increase speed and to reduce size and

and power consumption. Size reduction is especially important since it not only improves element packing density but also reduces power consumption and improves element operating speed. The elements used on this project have nozzles 0.040" x 0.040". It appears quite practical to reduce the element nozzle size to 0.010" x 0.010". Even smaller sizes are possible but do not appear practical from the contamination and cost considerations. Power consumption can also be reduced by operating at lower pressures; a 0.010" x 0.010" nozzle operating with 0.1 psig air represents only 2 milliwatts power consumption. The minimum operating pressures selected will require a compromise between speed and power consumption, however, since lowering supply pressure also decreases operating speed. It now appears that the optimum will exist in the 0.1 to 1.0 psig range.

Another problem related to smaller size elements is the difficulty of obtaining sufficiently reliable miniaturized instrumentation for test and monitoring purposes. Volumes required for introduction of pressure transducers represent capitances that can impair response; introduction of probes for flow measurements may restrict a channel or seriously alter circuit operation. The need for some form of fluid buffer to the monitoring equipment appears likely. The limitation of size reduction to the 0.010" nozzle will probably be a practical consideration for the near future applications because of the instrumentation problem even if there are no other restraints.

Operation of the DI with clock speeds of 1 kc to 5 kc appears practical using 0.010" nozzle elements. Analog fluid circuits using elements with 0.016" x 0.016" nozzles are operating at speeds over 4 kc. A system operating at this speed must be carefully designed, however, with concentrated attention not only to the element design but also to the interconnection techniques. A DI with a 5 kc clock speed capability will require careful package design to achieve the optimum layout from internal fluid dynamics considerations as well as requiring a practical configuration for interconnecting with other DI's.

As can be seen in the data of Appendix A, there is a significant amount of "noise" present in the output of fluid amplifiers. While it is difficult to

make a numerical evaluation of the seriousness of this noise without considering a specific application, there is little doubt that signal/noise characteristics of fluid elements are going to be a problem area in future applications. The digital computation systems should enjoy an advantage over the analog systems in this area since they are basically ON-OFF in nature while the analog systems are amplitude sensitive. The equipment developed for this project did not encounter any difficulty due to noise.

Summarizing these comments, there are five areas requiring additional effort for further development of the digital integrator. It is recommended that continued work be undertaken within the framework of a specific application to provide realistic and useful specifications and to identify some of the system interface problems. These five areas are:

1. miniaturization
2. higher speed
3. lower supply pressures
4. integrated design
5. instrumentation.

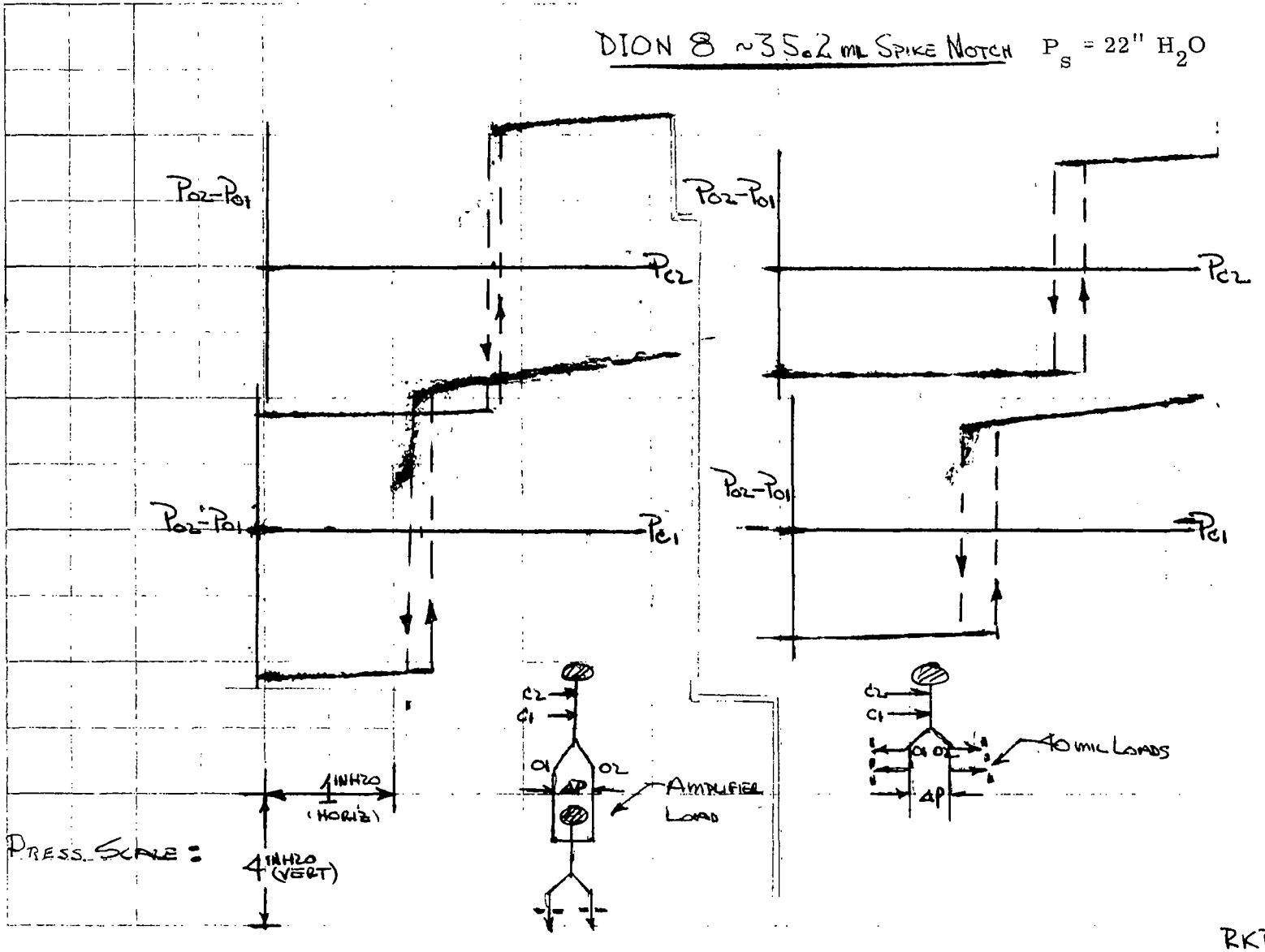
Appendix A
FLUID ELEMENT CHARACTERISTICS

The following x-y plotter traces are included to illustrate typical characteristics of the fluid amplifier elements that were used in the development of the digital integrator. Three elements were used: a 2-input OR-NOR, a digital amplifier, and a flip-flop. The outlines and the location of these elements in the circuit can be determined from Figure 3-1. The characteristic size of the elements was 0.040" x 0.040" nozzles.

The nomenclature used in the following x-y plots is as follows:

P_s	- supply pressure
P_{C1}, P_{C2}	- input signal pressures as identified on the schematics
P_{O1}, P_{O2}	- output signal pressures as identified on the schematics

The relatively large apparent noise present in the switching region in Figure A-2 results from the x-y recorder's inability to follow random deflections of the jet. In this region the jet position is indeterminate; a larger input signal must be used to attach the jet to a sidewall. The recorder noise at other regions in Figure A-2 and in Figure A-3 is both instrumentation noise and fluid signal noise.



RK Rose
2 Nov 64

Figure A-1. OR-NOR Switching Characteristics.

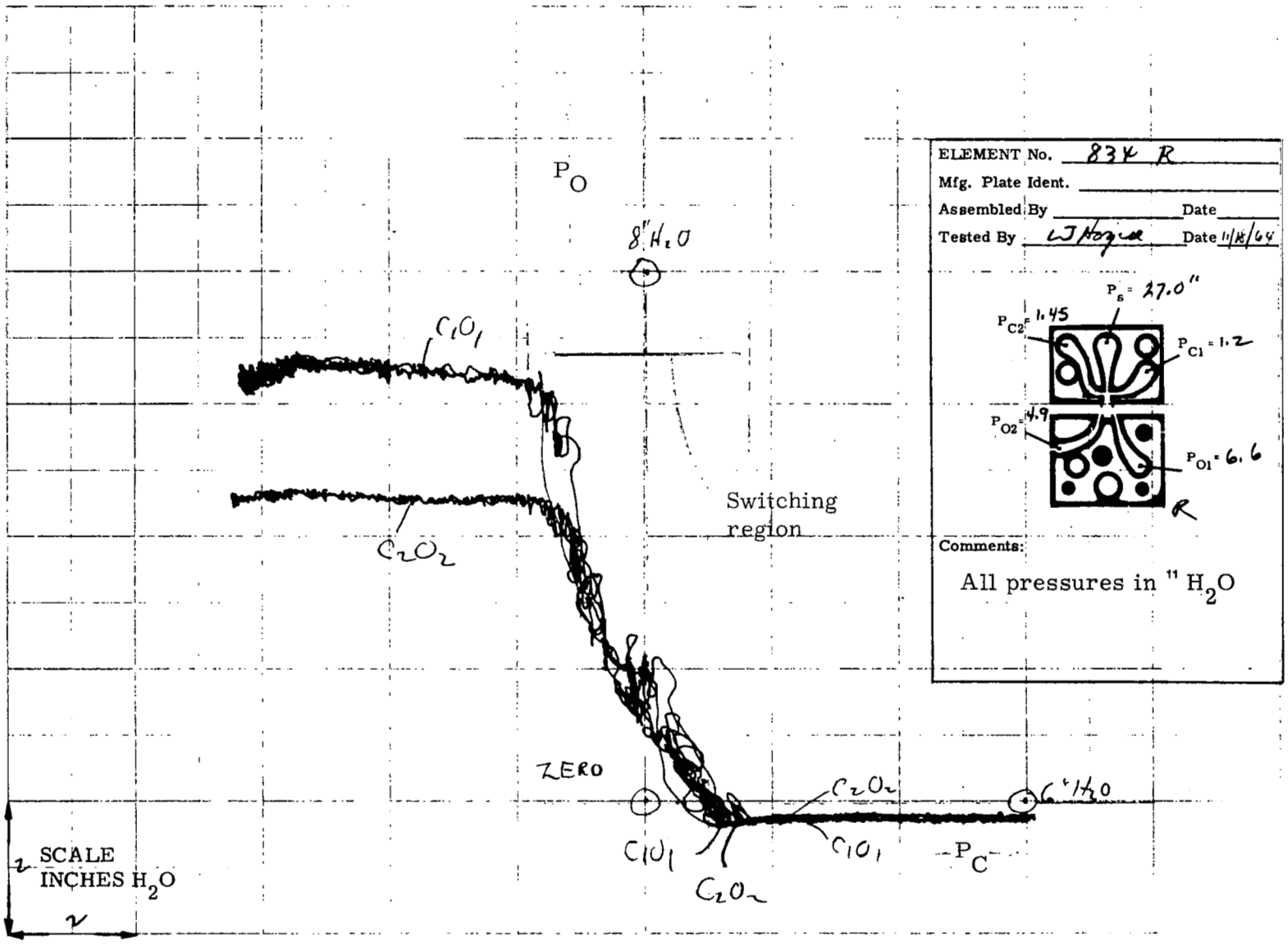


Figure A-2. Digital Amplifier Switching Characteristics.

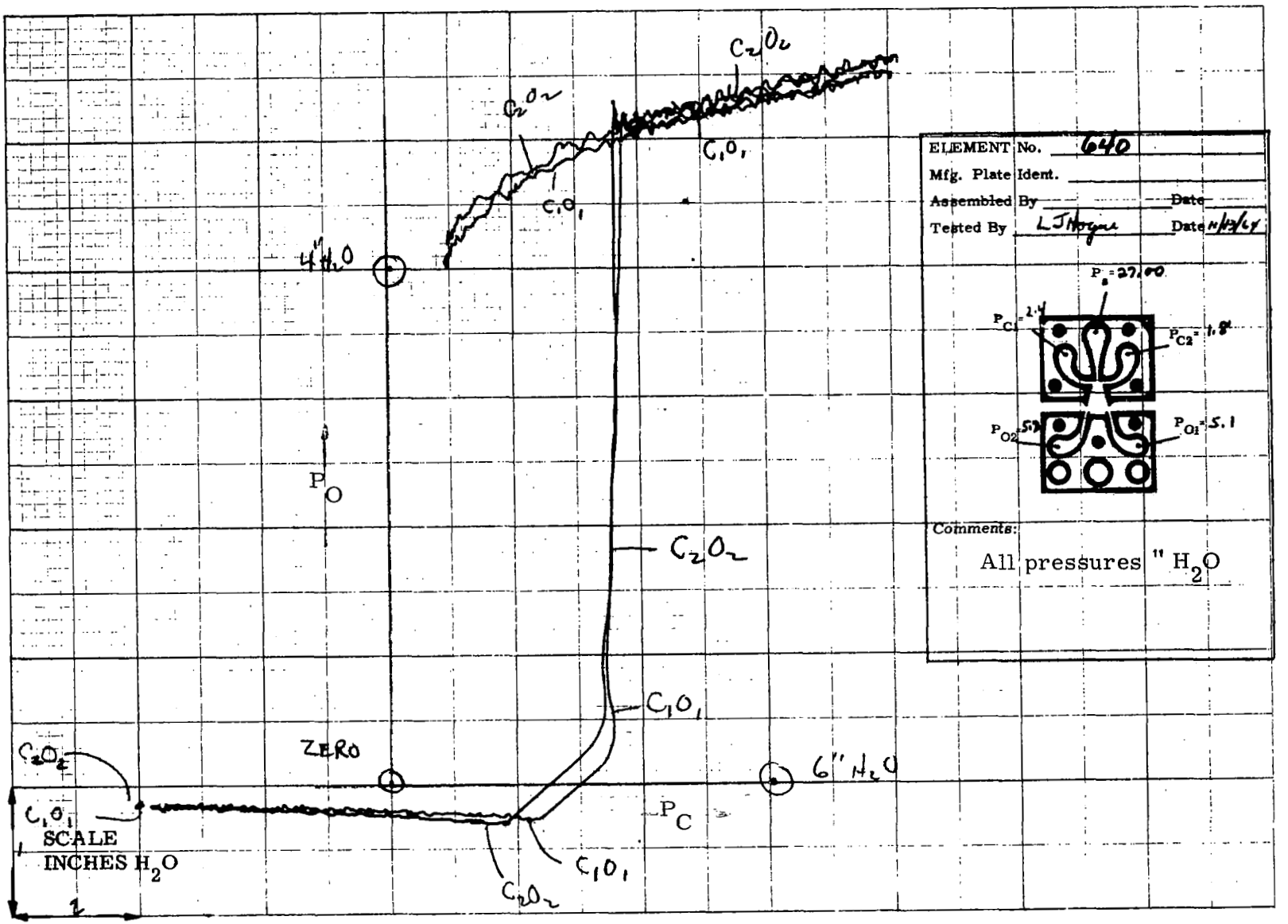


Figure A-3. Flip-Flop Switching Characteristics.

Appendix B

DIGITAL INTEGRATOR APPLICATIONS

The digital integrator (DI) can be interconnected to generate functions and to solve linear and nonlinear differential equations. Figure B-1 is a listing of some general applications and block diagrams to illustrate function generation.

It is convenient when designing systems using DI's to use schematic representations. Figure B-2 illustrates the commonly used schematic for the DI. Increments of the dependent variable are entered in the lower leg of the schematic; this lower leg represents the y register. The independent variable is entered in the upper leg which represents the remainder (R) register. Since the remainder register is of a finite length, continued addition of the integrand (y) into the R register results in periodic overflows representing changes in the integral (Δz). The apex of the schematic represents the Δz output of the integrator.

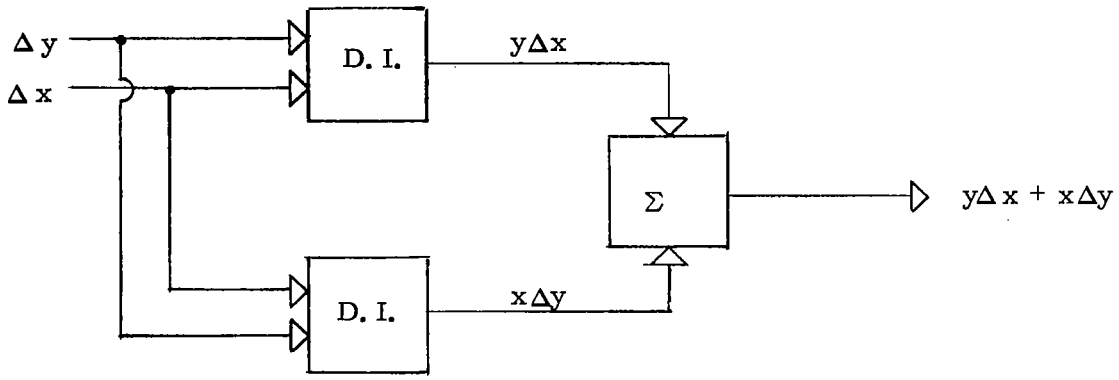
The interconnection of integrators is illustrated schematically in Figure B-3 where two integrators are interconnected to form the sine and cosine of an angle. This represents the first terms of a Taylor series for sin θ or cos θ. The input to the integrators are changes in an angle ($\Delta\theta$) supplied by an encoder which emits a pulse for a unit change in angle. Outputs of the integrators are changes in sin θ and cos θ. The negative of the change in sin θ must be inserted into the cosine integrator.

A typical use for this function (sin θ - cos θ) is illustrated in Figure B-4. This map represents the mechanization of one of the three control equations required in a typical satellite attitude control. Integrators 1 and 2 generate the sine and cosine of angle y. Multiplication by a constant (K) is obtained by inserting an initial condition into the y register of integrator three and not allowing any further changes in y to occur. Generation of sine and cosine of angle z is accomplished in integrators 4 and 5. Integrator 6 sums the outputs from the various integrators and also serves as part of the loop which implicitly performs a division operation. The summation into the y register of integrator

6 is accomplished by either accumulating the increments by a counter or adding them in sequence (through the adder) into the y register. The output of integrator 6 represents the change of the desired control angle (Δz).

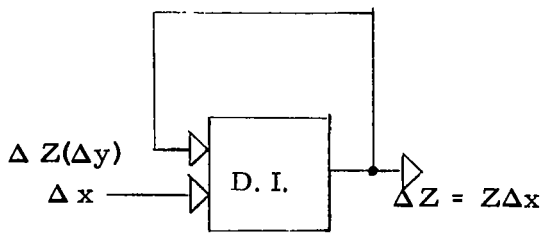
Integrator, Adder, Subtractor, Counter

Multiplier; $\int ydx + \int xdy = xy$



Function Generation

$x = \int \frac{dZ}{Z}$; $Z=e^x$



$\frac{d^2y}{dx^2} + y = 0$; $y = \cos x$; $\frac{dy}{dx} = -\sin x$

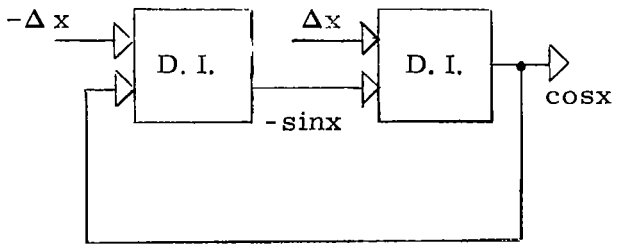


Figure B-1. Digital Integrator Applications

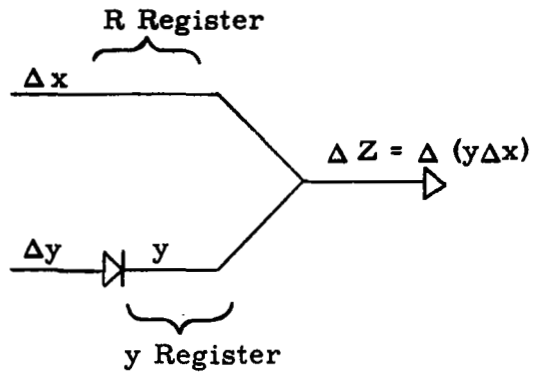


Figure B-2. Integrator Schematic

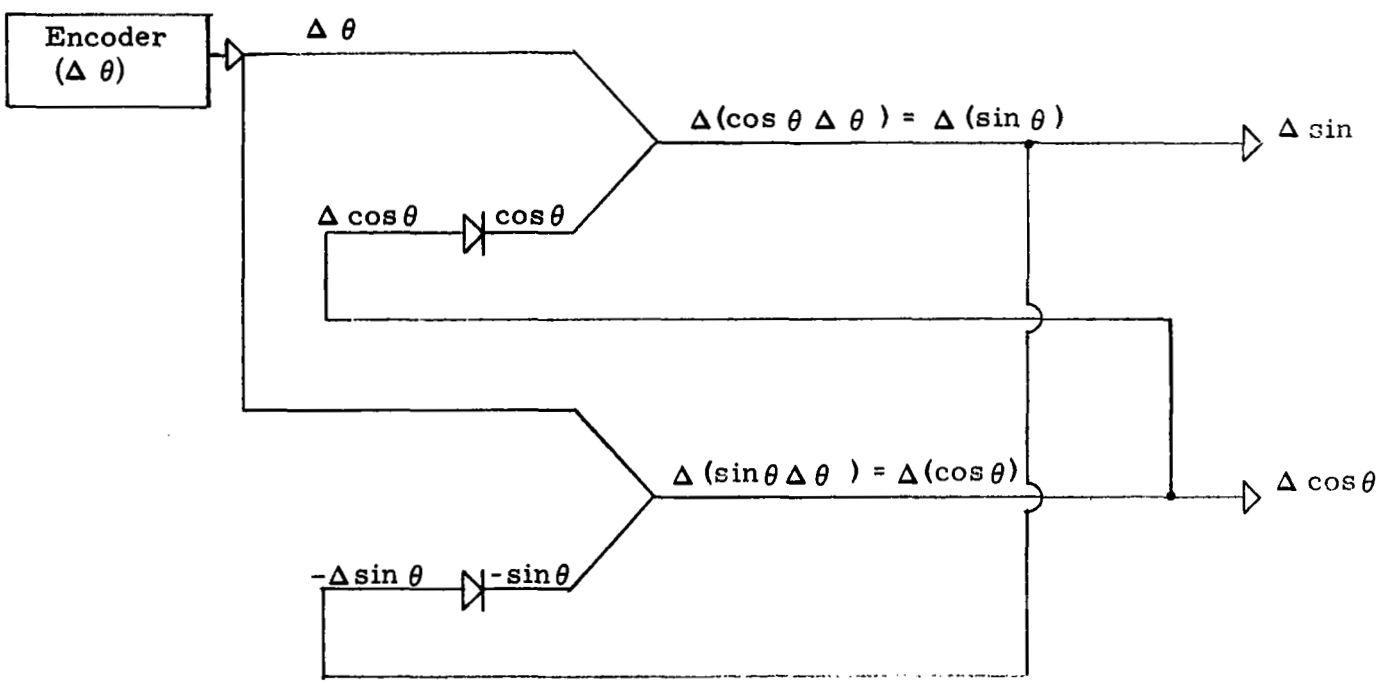


Figure B-3. Integrator Map for Sin θ - Cos θ

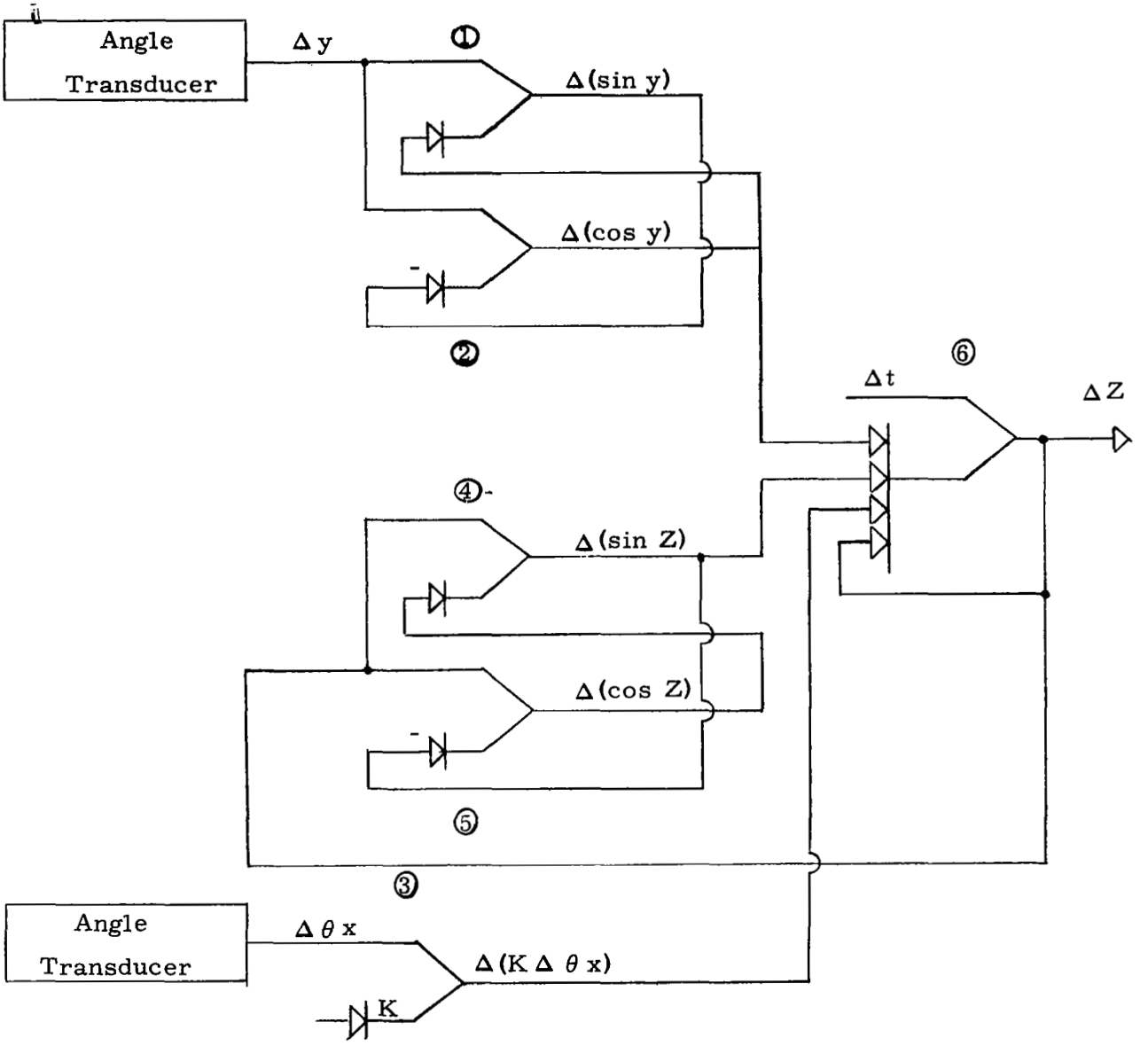


Figure B-4. Integrator Map for Satellite Attitude Control Equation