ON-LINE COMPUTATION OF CARDIAC OUTPUT FROM THERMO-DILUTION CURVES

BY

C. Narasimha Murthy

Submitted to the School for Graduate Studies
in partial fulfillment fo the requirements of
the degree of Masters of Applied Science

Department of Electrical Engineering
Faculty of Science and Engineering
University of Ottawa
Ottawa, Ontario
August, 1974

ABSTRACT

ON-LINE COMPUTATION OF CARDIAC OUTPUT FROM THERMO-DILUTION CURVES

A knowledge of cardiac output and other associated parameters is of vital importance in the management of critically ill patients. As a result of detailed investigations, a number of techniques have been suggested for their determination. Prior to the advent of compufing machines, the procedures for the determination of cardiac output and related homodynamic parameters were tedious and subject to inconsistancies. This restricted repeated measurements for their determination over a short interval of time and furthermore results were not immediately available.

However, recent advances in computing machines and computation algorithms have made it possible to determine the above parameters to a high degree of accuracy using computers.

The object of the research of this thesis is to use a mini digital computer for on-line computation of cardiac output, using an existing technique for its determination, called the thermo-dilution technique. A major advantage of this procedure, is that it enables repeated measurements of cardiac output to be made during therapy and results from each measurement is available immediately.

The computer used is a NOVA-1220 mini digital computer, a member of the NOVA family of computers. The computer program for data acquisition and processing is developed in the assembly language of the NOVA computers.

The scope of the thesis includes, verifying a mathematical procedure for fitting the family of curves represented by the equation

$$f(t) = k\,(t - t_a)^{a} \; e^{-\beta/(t-t_a)}$$

to the experimental curve; design of system software and comparison of results obtained to those determined from standard procedures.

ACKNOWLEDGEMENT

I wish to thank Dr. George M. White for his help and guidance during the course of this study. Also, I wish to express my sincere thanks and appreciation to the Bio-medical Engineering department, Ottawa Civic Hospital for their co-operation in making this project a success, and to the Department of Computer Sciences and Electrical Engineering for their encouragement and support during my stay.

Finally I wish to thank Mme L. LeBlanc for typing the manuscript of the thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE

TABLE PAGE NO

CHAPTER I

INTRODUCTION

The necessity for accurate information about cardiac output and related homodynamic parameters as aids during cardiovascular investigations has prompted development of sophisticated techniques for their determination. The advent of Digital / Analog. Computers has helped the computation of these parameters to a high degree of accuracy. It has also circumvented the tediousness and inexact results obtained from calculating the cardiac output with the use of a planimeter or replotting on logpaper.

Skinner, R.L., and Gehmlick, D.K. [1] have investigated the use of analog computers as aids for heart ailment diagnosis. Following this the development of a special purpose cardiac output computer for the rapid analysis of indicator dilution curves has been reported by Moody, N.E., Barber, H.D., and Holmlund, B.A. [2]. Further successful attempts have been made by Hirashi, H. Hara and J. Welson Bellville [3], for on-line computation of cardiac output from dyedilution curves using a general purpose analog computer. In the above report the authors use a Colson cuvette densitometer which yields an output voltage proportional to the dye concentration in the blood to obtain the dye-dilution curve. This is reshaped by removing the undesirable recirculation hump and the cardiac output is then determined by computing the area under this reshaped curve. However, Hepner, G.V., J.E. Jacobs et. al. [4] have described a suitable method for automatic computation of the area under the indicator dilution curves in the presence of recirculation. The method is based on an assumed exponential form for the downslope which allows the total area under the primary curve to be computed as the sum of two areas.

Hans, U. Wessel, Charles, F. Hepner, et. al. [5] have designed an analog computer for on-line computation of the area under the indicator dilution curves using the method presented by Hepner, C.F., et. al. [4]. Furthermore, the above report defines the accuracy and limitations of the instrument from evaluating the performance of the analog computer by digital computer analysis of dye-dilution curves. The drawback of such a system lies in the fact that, accurate results can be expected only if the downslope of the dilution curve decays exponentially before the onset of recirculation. Several other methods which use the analog computer for on-line computation of the cardiac output are available [6], [7].

Advances in software have widened the applicability of digital computers for the analysis of real-time problems. A study of dye-dilution curve using a digital computer has been presented by P. Sekel, J., G.R. Tait and M.M. Nathanson [8]. In the above report, the computer was programmed to analyse the recorded dye-dilution curve and estimate the area under the curve using Dow's formula [10]. More recent studies [8], [29], treat the indicator concentration as a Gamma variate, and use a digital computer for estimation of the cardiac output. C. Frank, Starmer and David O. Clark [9] suggest a numerical technique for analysis of the indicator-dilution curve.

THE PROBLEM

The development of an efficient algorithm and system software for real-time processing of thermal dilution curves, to determine the cardiac output and cardiac index during therapy, using a mini digital computer is the basis of this study.

Hitherto, the appearance time of the thermal indicator following injection, in the temperature-time curve, also called the thermo-dilution curve, has been treated as an observed quantity, introduced into the computer program as input or estimated by the method of moments. This study treats the appearance time as a variate and its value is determined during the real-time processing of the thermo-dilution curve. This technique lessens the necessity for a perfectly flat baseline. The reproducibility of measurements over short intervals of time makes the above technique highly useful during cardiovascular investigations.

The software has been developed in the assembly language of the NOVA family of mini computers. The physiological aspects for the problem are discussed in Chapter II. The algorithm for on-line computation of cardiac output from thermo-dilution curves and the necessary software are discussed in Chapter III and Chapter IV of this thesis. These are followed by discussions about system testing and system validation, in Chapter V and Chapter VI respectively. Finally conclusions drawn from studies in vitro are reported.

CHAPTER II

PHYSIOLOGICAL CONSIDERATIONS

Techniques for the Determination of Cardiac Output

Cardiac output is defined as the amount of blood flow per unit time usually expressed in liters/min. An associated parameter known as the cardiac index is defined as the ratio of the cardiac output to the body surface area. The body surface area is determined from nomograms or from the following Dubois and Dubois height-weight equation

$$\text{Body Surface area} = 71.84\, w^{0.425}\, H^{0.725} \qquad \ldots\ldots (1)$$

where

w is expressed in Kg.

H in cm and BSA in m^2.

Invasive and non-invasive methods are the primary means for determining the cardiac output. One of the non-invasive methods is explained by Chen F. Lam and Arno R. Hohn [28], where the rate of change of impedance between two electrodes placed around the neck and xyphoid process is recorded with an impedance plethysmograph. Base upon certain decision rules, a computer is programmed to determine the minimum value of the rate of change of impedance and the choice of the cardiac cycle for the calculation of cardiac output using the following equation :

$$C.0 = \frac{RL^2}{1000\, z_{10}^2}\, TH \left(\frac{dz}{dt}\right)_{min} \qquad \ldots\ldots (2)$$

where

R = resistivity of blood

L = mean distance in cms between the recording electrodes

T = the ventricular ejection time in seconds

H = Heart rate

$(\frac{dz}{dt})_{min}$ = is the minimum value of $\frac{dz}{dt}$ during a cardiac cycle

Z_0 = the mean body impedance between recording electrodes

The disadvantage with the above non-invasive technique for determination of cardiac output during therapy lies in the calibration of the impedance determination. Also the values of R, L and z_o vary from person to person and therefore the number of unknowns to be input to an on-line computing system is more, there by causing a delay in the availability of the final results. Accordingly non-invasive techniques are not frequently used during therapy for determination of cardiac output.

Several invasive methods have been described elsewhere [10] for the determination of the cardiac output. One of the methods uses an indirect respiratory procedure, and is called the DIRECT FICK method. In this method the cardiac output is expressed as a ratio of the total oxygen consumption (in mg/min) to the difference in O_2 Content between the arterial and mixed venous blood, usually expressed in volume percent. The inherent disadvantages of this technique is overcome in the Injection Method. In this method, a known quantity of a foreign substance is introduced into the vascular system and a time-concentration curve is recorded. The cardiac output is then determined by computing the area under the curve using a planimeter or by numerical techniques.

Several studies using the indicator and dye solutions in the injection method have been done [11], [20]. The disadvantage of this technique for on-line computation is in the recirculation of the injectate, which makes the determination of the end of disappearance the a uncertainty. This problem is overcome by a more recent development called the Thermodilution technique.

Principle of Thermodilution Technique:

A known quantity of a cold indicator solution is injected into the superior vena cava or the right atrium and the blood temperature measured in the pulmonary artery as a function of time. From this information the cardiac output is calculated as follows.

Let V_I be the volume of injectate with specific heat SH_I and specific gravity SG_I at temperature R_I. In accordance with these, let SH_B, SG_B and T_B represent the specific heat, specific gravity and the temperature of the blood. Assuming that there is no transfer of heat between the surrounding tissues, we can write the following heat equation

Heat Injected = Heat detected.

$$V_I \times SH_I \times SG_I \times (T_B - T_I)$$

$$= C.0 \times \Delta T \times t \times SH_B \times SG_B \times \frac{1000}{60} \qquad \ldots\ldots (3)$$

where ΔT = Average change in temperature
t = duration of temperature change
and $C.0$ = Cardiac output.

The term $\Delta T \times t$ is the area under thermodilution curve given by $\int_0^t \Delta T_B(t)\, dt$. Substituting this in equation (3) and solving for C.0 we get,

$$C.0 = \frac{V_I \times SH_I \times SG_I \times (T_B - T_I)}{SH_B \times SG_B \times \int_0^t \Delta T_B(t)\, dt} \times \frac{60}{1000} \ \mathrm{L/min} \qquad \ldots\ldots (4)$$

Technique:

The method involves the placement of a single flow directed catheter into the pulmonary artery. The catheter is positioned using pressure measurements. This permits measurements of the

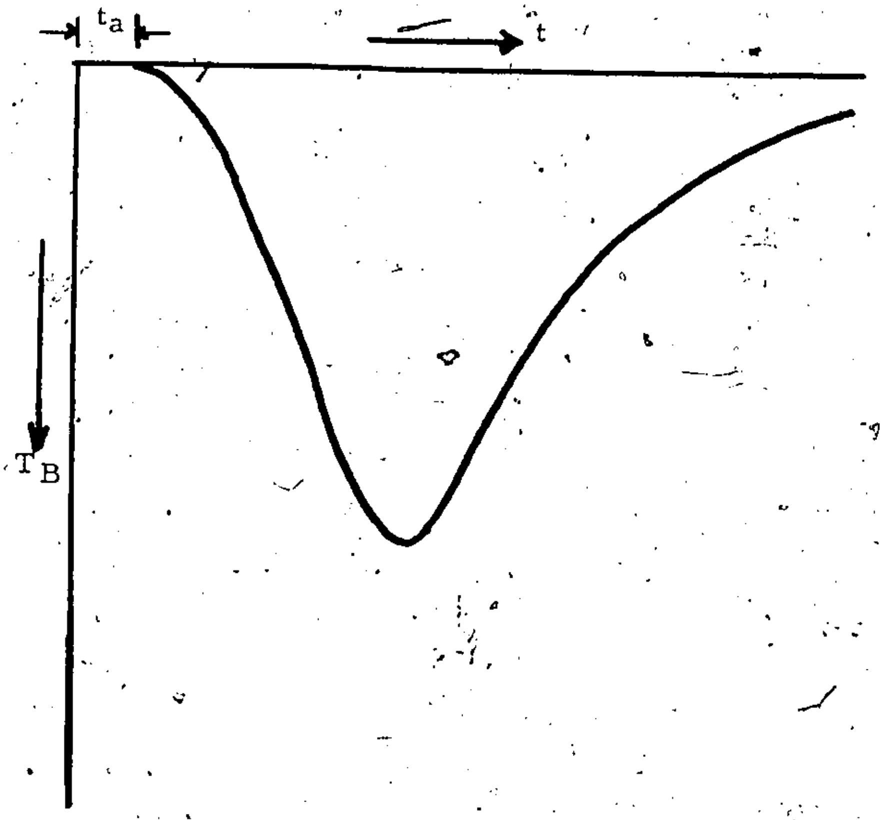


Figure 1

Thermo-dilution Curve

temperature change of the pulmonary blood following injection of the cold dextrose solution, and the pulmonary artery pressure.

A standard No. 7F catheter is made of polyvinyl chloride and measures 110 cm the length and 2.3 mm in outer diameter. It is constructed in such a way that the body is divided into four lumina along the long axis. One lumen connects directly to a small latex balloon fastened 1 mm from the catheter tip. When inflated with air, the balloon extends just beyond the catheter tip, converting the tip into a broad soft surface. A thermistor is placed at the tip of the second lumen just proximal to the balloon, and its fine insulated wires extend the length of the lumen. The thermistor measures the change in pulmonary blood temperature produced by the injection of the cold fluid into the right alrium. The third lumen terminates about 30 cm from the tip of the catheter, such that, when the catheter tip is in the pulmonary artery, the end of the third lumen is in the right alrium. This lumen is also called the proximal lumen. It is used for injecting cold solution and obtaining right alrium pressure. The fourth lumen called the Distal lumen, extends the entire length of the catheter to its tip. This is used for monitoring pulmonary artery pressure and for the withdrawl of blood samples.

James S. Forrester et. al. [21], describe in detail the catheterization technique. After the catheter tip is properly positioned in the pulmonary artery, a known amount of cold indicator solution, usually 0.5 % dextrose is injected into the right alrium through the proximal lumen of the catheter. The difference between the pulmonary artery blood temperature and the normal body temperature is continuously recorded. The cardiac output is inversely proportional to the integral of the temperature difference.

The above technique was introduced by Fegler [22] as a method for the measurement of volumetric blood flow-rate .

Since then a number of investigators [23] , [27] have made series of measurements of cardiac output using cold and room temperature injectates and report close agreement with those obtained from dye-dilution and direct Fick methods.

Some of the advantages of the thermo-dilution technique are: the absence of recirculation, which makes the method suitable for on-line computations, and also enables repeated measurements over short intervals of time. It does not require withdrawl of blood during measurements and removal of blood for calibration. The use of the flow directed catheter allows for simultanious measurements of related hemodynamic parameters. Even with low flow rates, accurate cardiac output determination may be obtained.

A major criticism of the technique is based on the uncertainty in the injectate temperature and heat transfer to and from the surrounding tissues. The former is minimized by keeping the distance between the site of injection and sampling as short as possible without affecting the complete mixing of the injectate and blood. Fluctuations of the base line temperature in the pulmonary artery with respiration is an associated problem with the determination of cardiac output by thermo-dilution. It represents a physiologic noise. This is partly overcome by the injection of more thermal indicator.

Repeated measurements made on animals have proved the consistancy of the thermo-dilution technique and has become a routine method for the determination of cardiac output in man .

CHAPTER III

ALGORITHM FOR ON-LINE COMPUTATION OF CARDIAC OUTPUT

The procedure is developed on the hypothesis that the change in the pulmonary blood temperature will be a gamma variate. The family of curves represented by the following function.

$$c(t_i) = k(t_i - t_a)^{\alpha} e^{-(t_i - t_a)/\beta} \qquad (t_i \geq t_a) \qquad \ldots\ldots (5)$$

where k, α and β are arbitrary parameters, t_a is the appearance time and $c(t_i)$ is the indicator concentration at time t_i; has been shown [29] to give excellent curve fits to indicator dilution curves. The virtual absence of recirculation in the thermo-dilution method helps to simplify similar techniques for fitting equation (5) to the thermo-dilution curves. A weighted least square method [Appendix I], is employed to fit equation (5) to the thermo-dilution curves. The parameters of the function are determined by computing the following expressions:

$$X1S = A_{11} = SX1S - \frac{(SX1)(SX1)}{SW} \qquad \ldots\ldots (6)$$

$$X12 = A_{12} = A_{21} = SX12 - \frac{(SX1)(SX2)}{SW} \qquad \ldots\ldots (7)$$

$$X2S = A_{22} = SX2S - \frac{(SX2)(SX2)}{SW} \qquad \ldots\ldots (8)$$

$$X1Y = A_{1Y} = SX1Y - \frac{(SX1)(SY)}{SW} \qquad \ldots\ldots (9)$$

$$X2Y = A_{2Y} = SX2Y - \frac{(SX2)(SY)}{SW} \qquad \ldots\ldots (10)$$

where,

$$SX1 = \sum_{i=1}^{n} w_i x_{i1} \qquad \ldots\ldots (11)$$

$$SX1S = \sum_{i=1}^{n} w_i \, x_{i1}^{2} \qquad \ldots\ldots (12)$$

$$SX12 = \sum_{i=1}^{n} w_i \, x_{i1} \, x_{i2} \qquad \ldots\ldots (13)$$

$$SX2 = \sum_{i=1}^{n} w_i \, x_{i2} \qquad \ldots\ldots (14)$$

$$SX2S = \sum_{i=1}^{n} w_i \, x_{i2}^{2} \qquad \ldots\ldots (15)$$

$$SX1Y = \sum_{i=1}^{n} w_i \, x_{i1} \, y_i \qquad \ldots\ldots (16)$$

$$SX2Y = \sum_{i=1}^{n} w_i \, x_{i2} \, y_i \qquad \ldots\ldots (17)$$

$$SY = \sum_{i=1}^{n} w_i \, y_i \qquad \ldots\ldots (18)$$

$$SW = \sum_{i=1}^{n} w_i \qquad \ldots\ldots (19)$$

From the above expressions the parameters are determined using the following equalities :

$$D = (X1S)\,(X2S) - (X12)\,(X12) \qquad \ldots\ldots (20)$$

$$ALPHA = \alpha = \frac{(X2S)\,(X1Y) - (X12)\,(X2Y)}{D} \qquad \ldots\ldots (21)$$

$$-\frac{1}{B} = \frac{(X1S)\,(X2Y) - (X12)\,(X1Y)}{D} \qquad \ldots\ldots (22)$$

$$XM = \mu = (SY - (ALPHA)\,(X1S) - \frac{1}{B}\,(X2S))/SW \qquad \ldots\ldots (23)$$

$$k = e^{XM} \qquad \ldots\ldots (24)$$

with these values of k, α and β, the area under the thermo-dilution curve is computed using the relation

$$\int_{o}^{\infty} c(t_i)\,dt = k\ \beta^{\alpha+1}\,\Gamma(\alpha + 1) \qquad \ldots\ldots (25)$$

The following steps of the algorithm are invoked after positioning the catheter tip in the pulmonary artery.

The initial blood temperature is recorded. Then 10 cc of ice cold dextrose solution is injected into the pulmonary circulation system through the proximal lumen of the catheter. The temperature of the blood in the pulmonary artery is sampled at $\frac{1}{10}$ th of a second interval in time. This sampling rate is determined by the available frequencies of the real time clock, of the NOVA family of computers, for I/O transfers and on the memory requirements for storing the sampled data. Since higher sampling rates would necessitate large memory requirements, a low sampling rate is chosen.

The change in temperature of the blood in the pulmonary artery with respect to its initial temperature, over the sampling duration in time is computed. A plot of this temperature difference vs time interval of sampling is the thermo-dilution curve. The appearance time of the thermal indicator is then determined by first computing the standard deviation σ, of the initial signal which corresponds to the normal temperature of the blood. A point on the thermo-dilution curve whose standard deviation is $4(\sigma + 1)$ is determined. From this point the sampled data is retraced back till the standard deviation is less than or equal to σ. This point is assumed to be the beginning of the appearance of the thermal indicator, and the interval between this point and the origin is taken to be the appearance time t_a.

A weighted least squares method is then used to determine the values of the parameters of the equation (4), and using the relation given by equation (25), the area under the thermo-dilution curve is computed. The cardiac output is inversely proportional to the area under the curve, and is determined from the following equation

$$C.O = \frac{1.08 \times 0.825 \times 60 \times V_I \times (T_B - T_I)}{\int_0^\infty \Delta T_B(t)\, dt} \; mL/min \quad \ldots\ldots (28)$$

where

$1.08 = \frac{SG_I \times SH_I}{SG_B \times SH_B}$

0.825 = correction factor for injectate temperature raise after injection.

V_I = 10 mL volume of injectate

T_B = Initial blood temperature ($^\circ$c)

T_I = Initial injectate temperature ($^\circ$c)

and

$\int_0^\infty \Delta\, T_B(t)$ = Area under the thermo-dilution curve.

The cardiac index is computed by dividing the cardiac output by the body surface area (BSA), which is obtained from nomograms or calculated using equation (1). A flow diagram of the above algorithm is given in figure 2. For repetitive measurements of cardiac output and cardiac index, the steps following injection of the cold indicator solution are repeated.

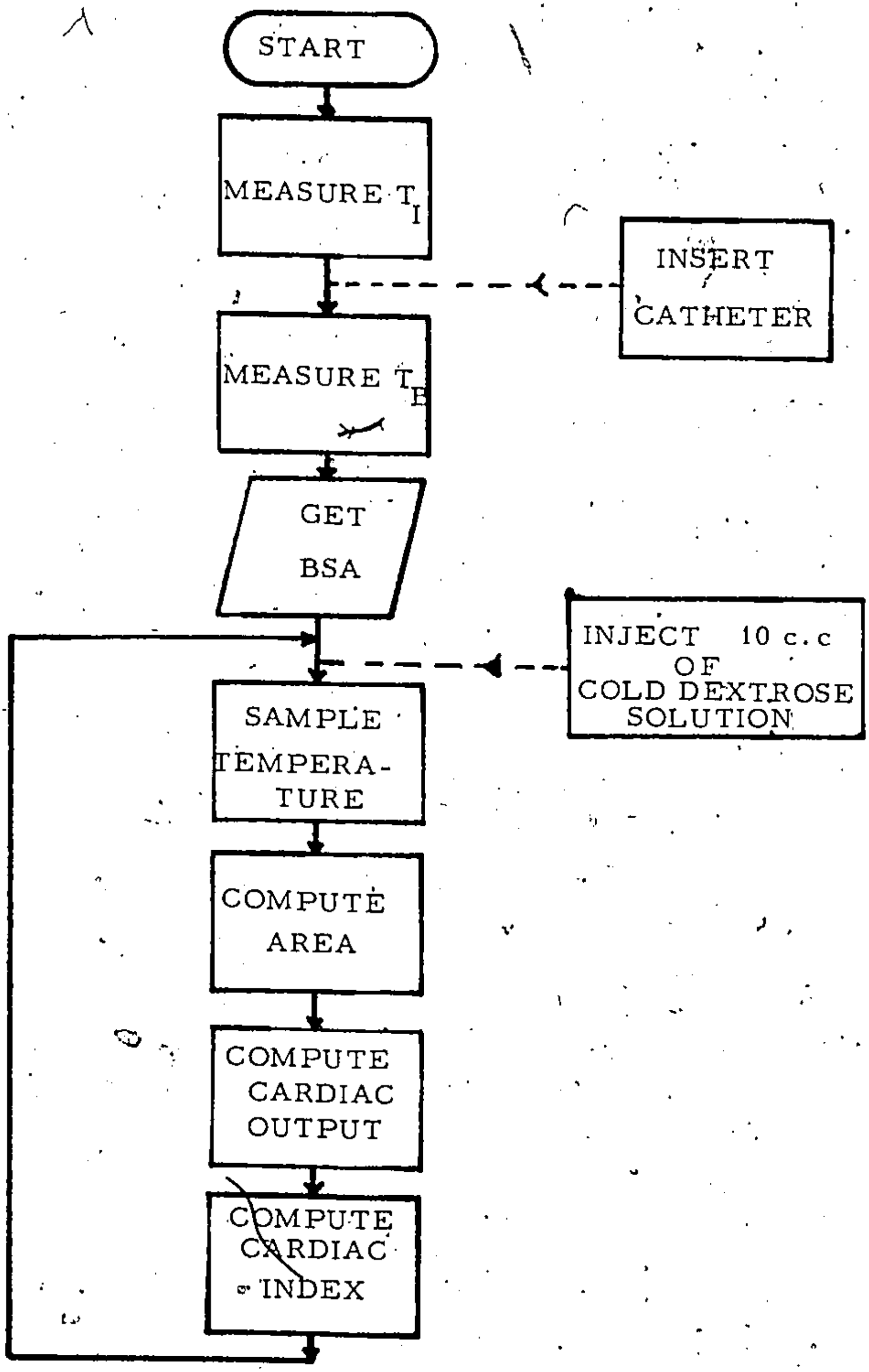


Figure 2

Flow Diagram for thq On-Line Computation of Cardiac Output From Thermo-Dilution Curves.

CHAPTER IV

SOFTWARE DEVELOPMENT

An on-line system may be defined as a system in which the operation of its components are under the control of the central processing unit of the computer. If the time scale of the computer is far less than that of the system, the term 'Real-time' is sometimes used. The system for determining the cardiac output and related hemodynamic parameters, from thermo-dilution curves, is developed in accordance with the above definitions. As in any on-line system, the computer is treated as a unit in the system which can be modified or taken out and replaced by a bigger and faster unit to increase the capacity of the system. Use of the NOVA computer and development of the software in its assembly language makes such a flexibility feasible, because of the availability of a range of compatible NOVA computers.

In programming an on-line or real-time computer system, as in any complex computer installation, three kinds of programs are needed; supervisory programs, application programs and support programs.

The supervisory programs co-ordinate and schedule the work of the applications program and carryout service functions for them. In a complex system, the many application programs can be considered as subroutines of the main supervisory program. The supervisory programs handle I /O operations and service interrupt requests. They also carryout queueing of messages and data, and deal with error conditions.

In a single user system dedicated to a particular task, as is the case for the on-line computation of cardiac output from thermo-dilution curves, the above function of the supervisory programs are minimal and can be treated as part of the application program of the system. Figure 3 is a flow diagram of the application programs for on-line

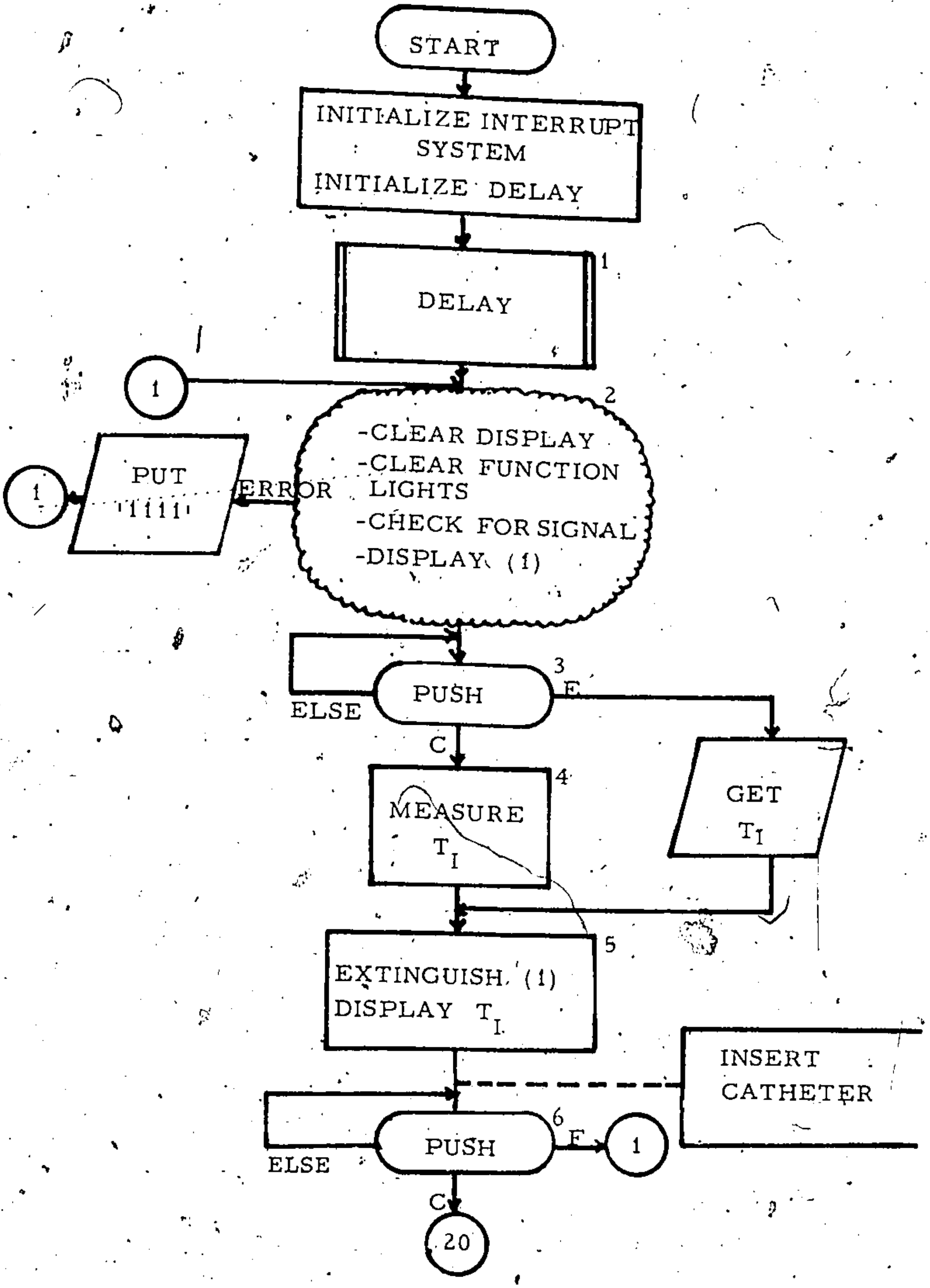


Figure 3

Flow Chart For On-Line Computation Of Cardiac Output From Thermodilution Curve.

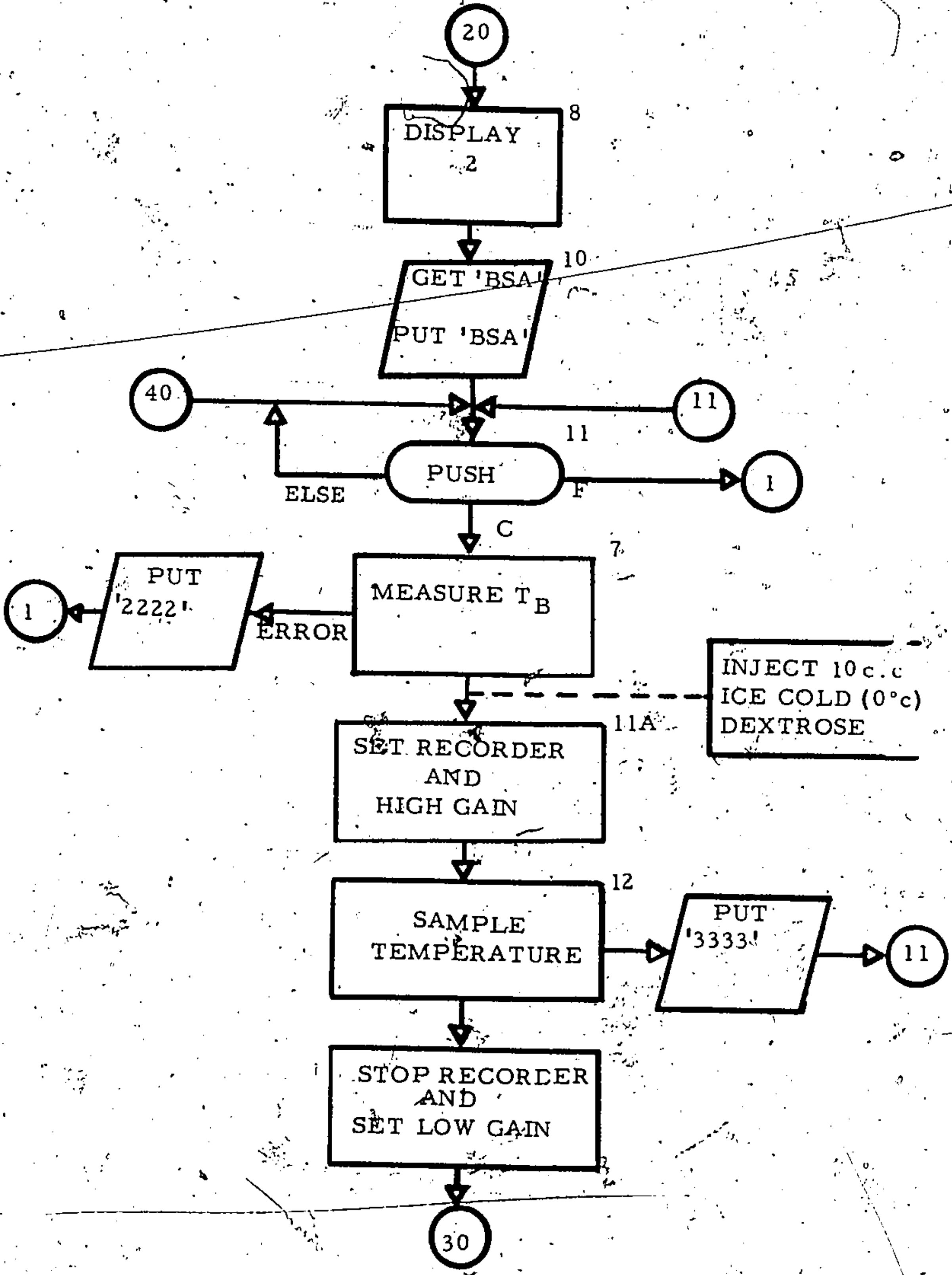


Figure 3 Continued

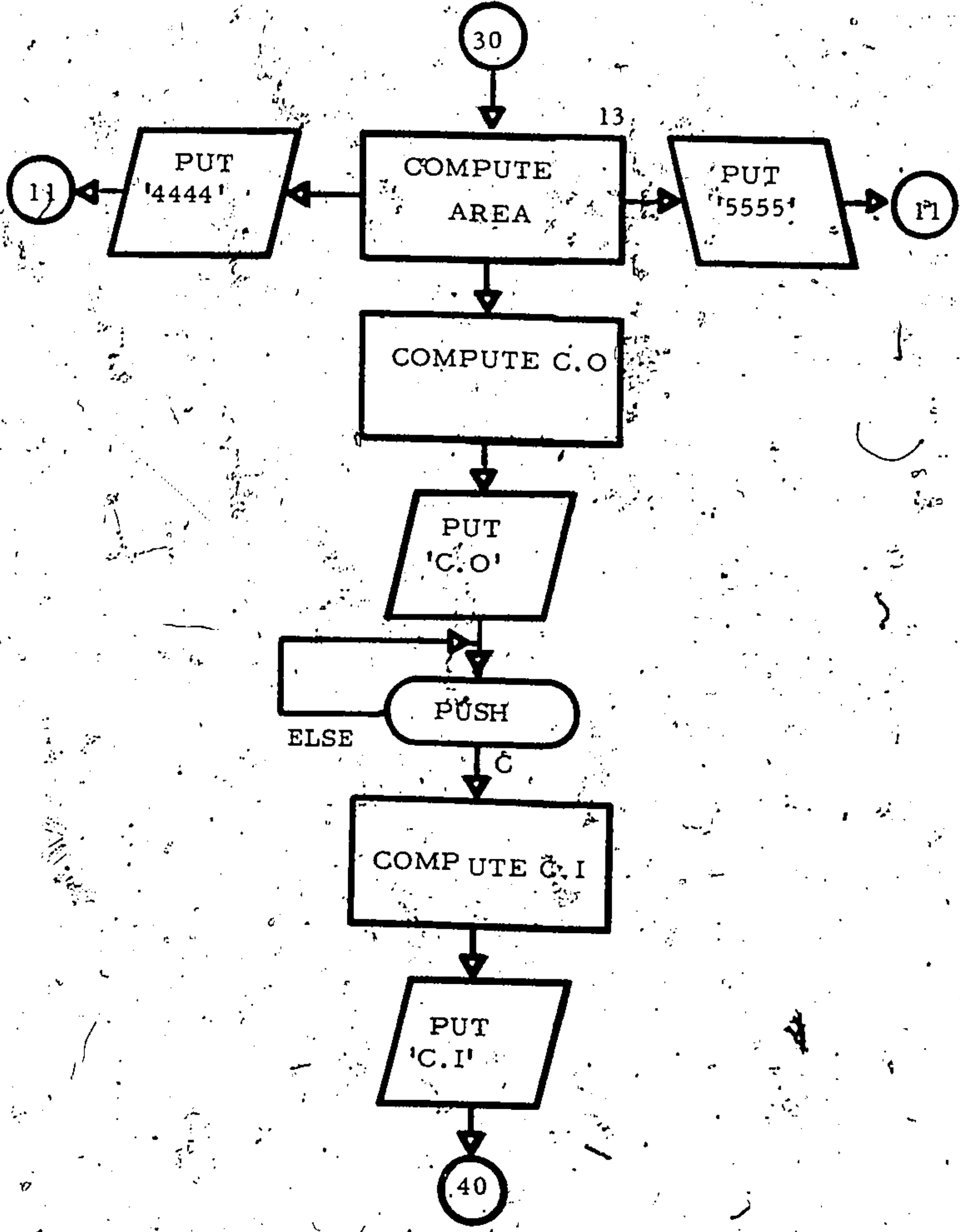


Figure 3 Continued

computation of cardiac output from thermo-dilution curves.

The execution of the procedure is initiated by enabling the power supply switch of the system. To overcome the initial transients in the power supply, a function called "DELAY" [Box 2], is executed everytime the power supply switch is enabled. A flowchart of this function is shown in Figure 4. The duration of the delay is determined by the number of interrupt requests from the real time clock of the system. Then the display of the I/O device, and the functional lights of the functional key board [FKB], are cleared, and the output signal from the thermistor of the catheter is checked. The presence of a signal is indicated by displaying a '1' on the display of the I/O device; otherwise a '1111' is displayed to indicate an error condition. If an error has occurred, the program will wait till the error is corrected. The next step in the procedure is to measure the voltage corresponding to the injectate temperature (Box 4, Figure 5), or if measured before, to input the corresponding voltage to the program. This is done by enabling the 'C' key of the I/O device if the temperature has to be measured [Box 3], or enable the 'E' key if the voltage corresponding to the injectate temperature has to be input to the program. Following normal execution of either of the above two functions, the voltage corresponding to the injectate temperature is displayed. However, if an error has occurred during the measurement of the injectate temperature, a '2222' is displayed and the procedure has to be reinitiated starting from Box, 2. After the measurement of the injectate temperature, if the 'F' key of the FKB is enabled, the procedure is reinitiated from Box 2, Figure 3. However, if the 'C' key is enabled, then the procedure continues and a '2' is displayed on the FKB. Following this, the body surface area 'BSA' is input to the program and then displayed on the FKB. At this stage, the catheter is positioned in the right atrium of the heart; and now the system is ready for processing the thermodilution curve. The procedure is continued by enabling the 'C'

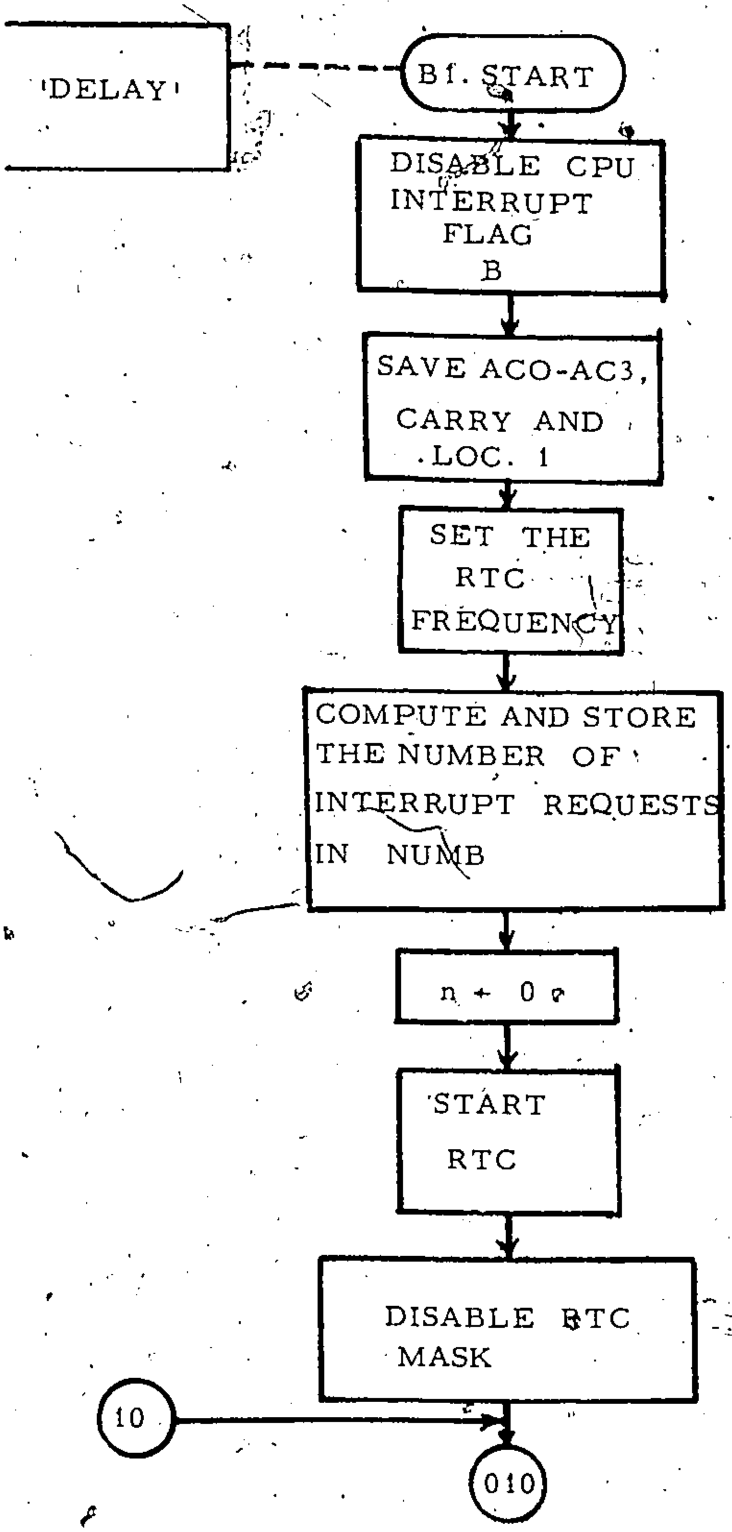


Figure 4

Flow Chart Of Function 'Delay'

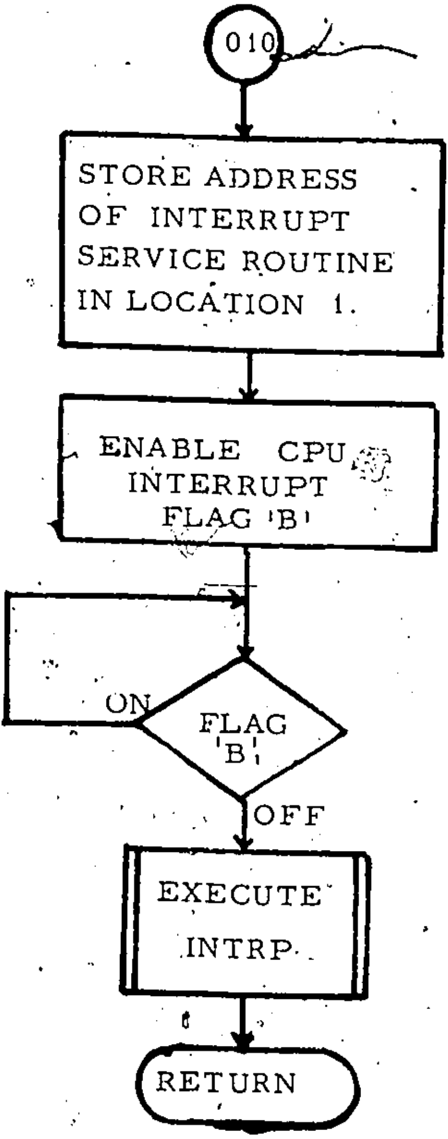


Figure 4 Continued

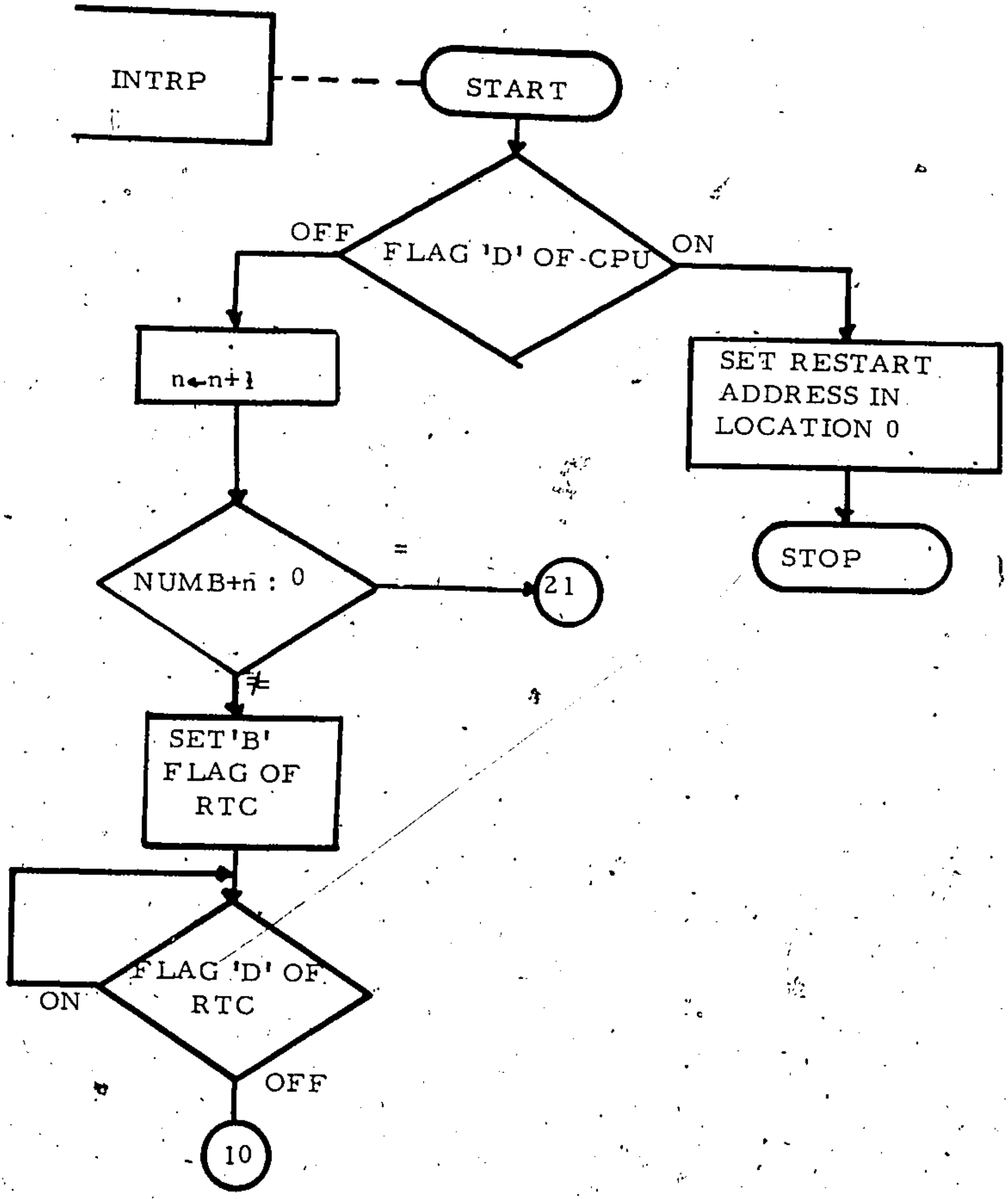


Figure 4 Continued

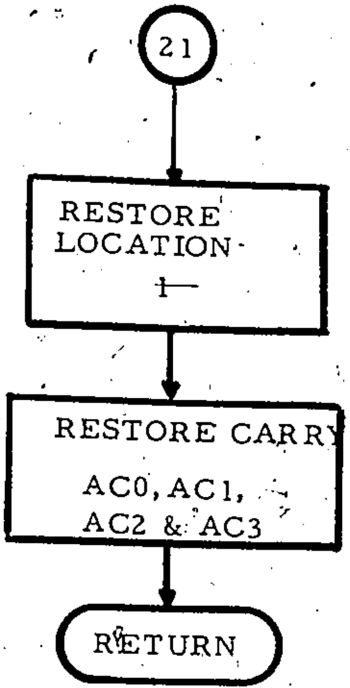


Figure 4 Continued

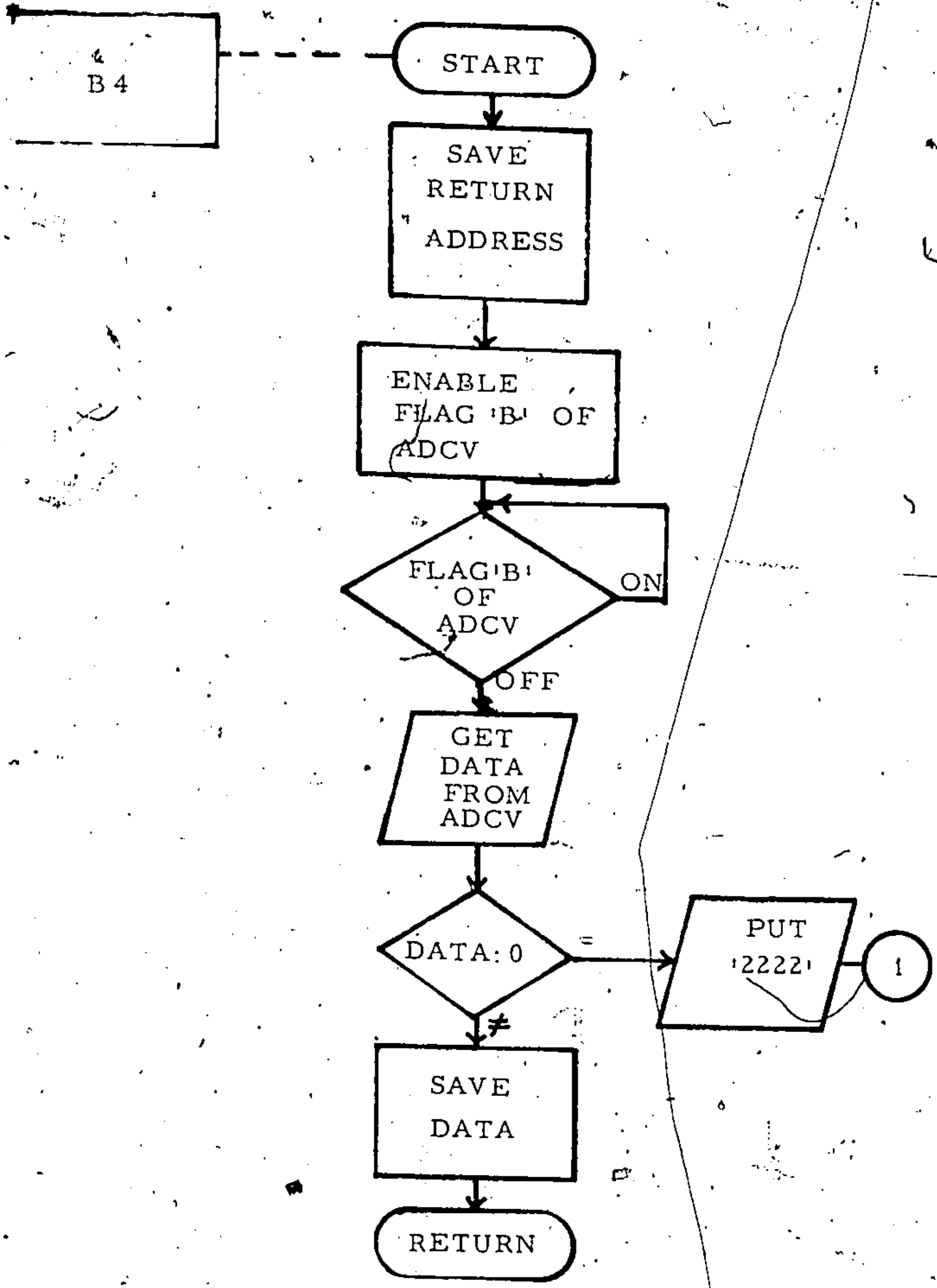


Figure 5

Flow Chart For Measurement Of Injectate And Blood Temperature.

key of the FKB [Box 11], and simultaniously injecting 10 c.c. of ice cold dextrose solution into the pulmonary circulation system. However, if the 'F' key is enabled, the procedure is restarted from Box, 2.

Following the injection of the dextrose solution, the voltage corresponding to the normal blood temperature is measured, [Box 7, Figure 3]. An error in the measurement is indicated by displaying a '2222'. If such an error condition does not occur, the procedure continues and the recorder is turned on to record the thermo-dilution curve; and the thermistor circuit HI/LO gain amplifier is switched to HI gain [Box 11A]. The signal which corresponds to the varying pulmonary blood temperature, is sampled every 10th of a second for 20 seconds, and the values are stored in floating point format. Figure 6 is a flow chart for sampling pulmonary blood temperature [Box 12]. If the sampled temperature is below the minimum a '3333' code is displayed to indicate that condition. The procedure can be restarted from Box 11 or Box 2 as shown in Figure 3.

After sampling the temperature, the recorder is turned off and the thermistor circuit HI/LO gain amplifier is switched to LO gain [Box, 12A]. The weighted least square method is used for fitting equation (5) to the sampled data and the area under the thermo-dilution curve is then computed [Box, 13]. Figure 7 is a flow chart for computation of the area under the thermo-dilution curve. If the computed result is too large resulting in an overflow, such a condition is indicated by displaying '5555' and the absence of the peak of the sampled thermo-dilution curve is indicated by displaying '4444'. In both cases, the procedure can be restarted either from Box 2 or Box 11. If neither of the above two errors occur, the cardiac output (CO) is computed using equation (3) [Box 14], and then displayed on the FKB. An overflow in the computed result is indicated by displaying '6666', and the procedure can be restarted from Box 11 or Box 2. Finally the normalized cardiac output, often called the Cardiac Index, is computed

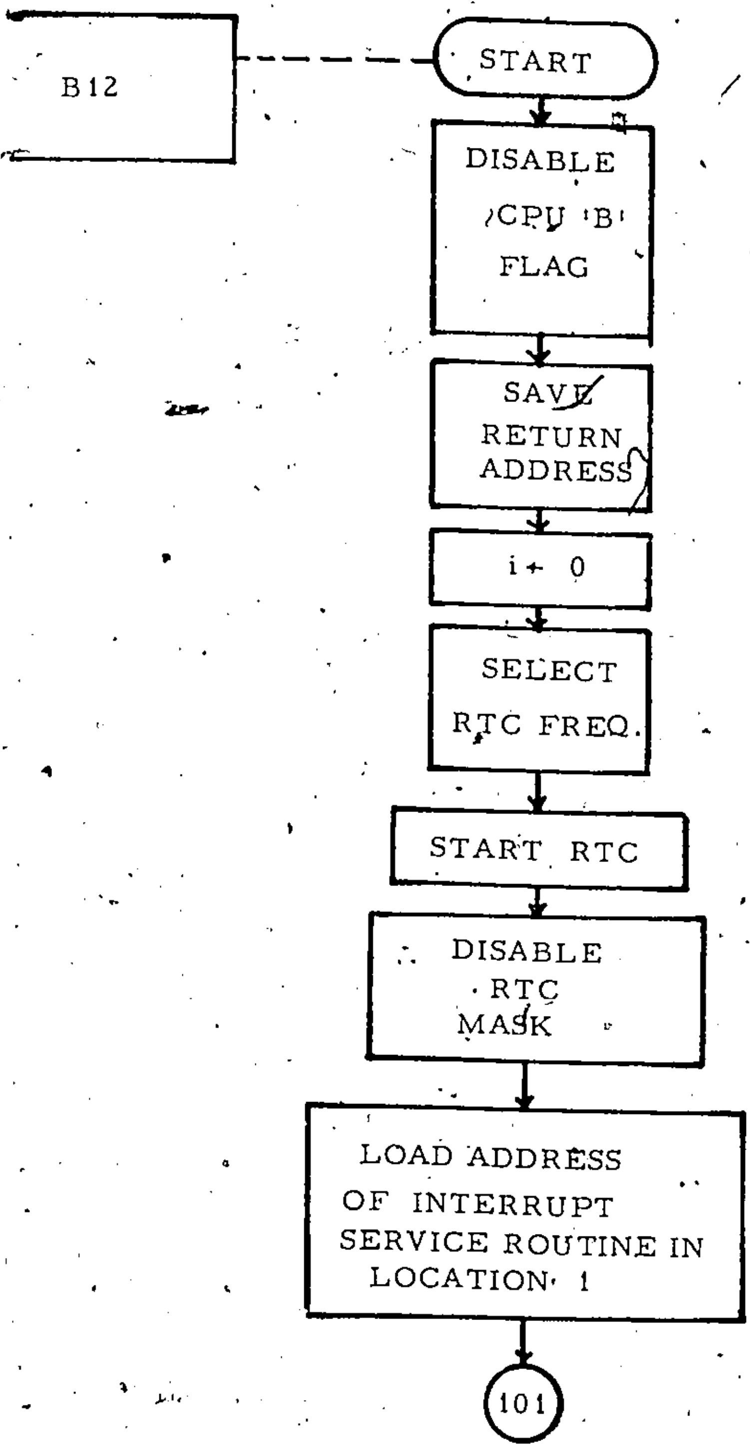


Figure 6

Flow Chart For Sampling Pulmonary Blood Temperature.

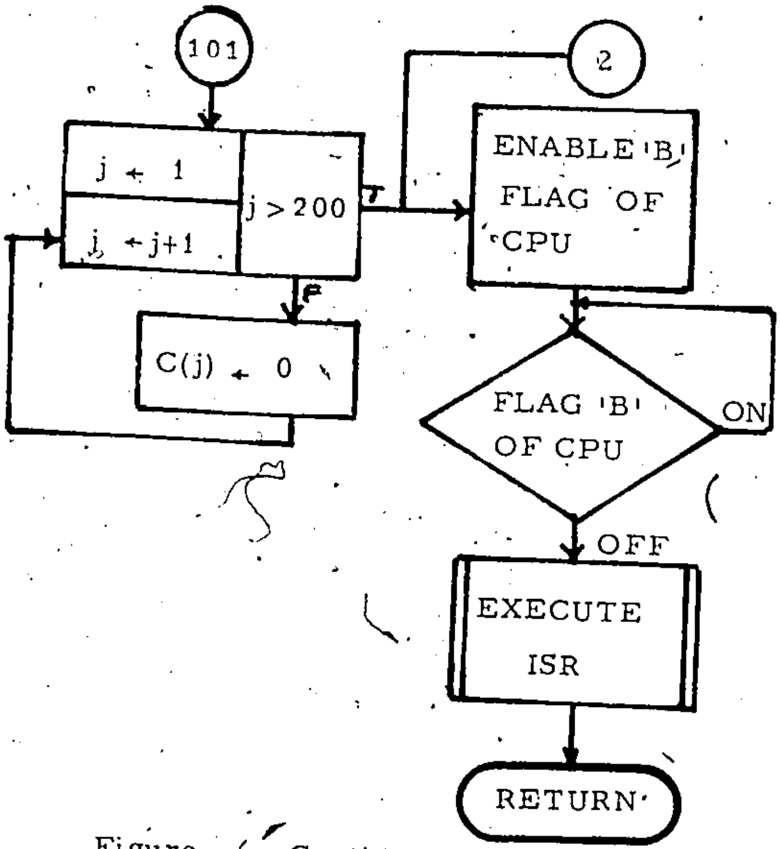


Figure 6 Continued

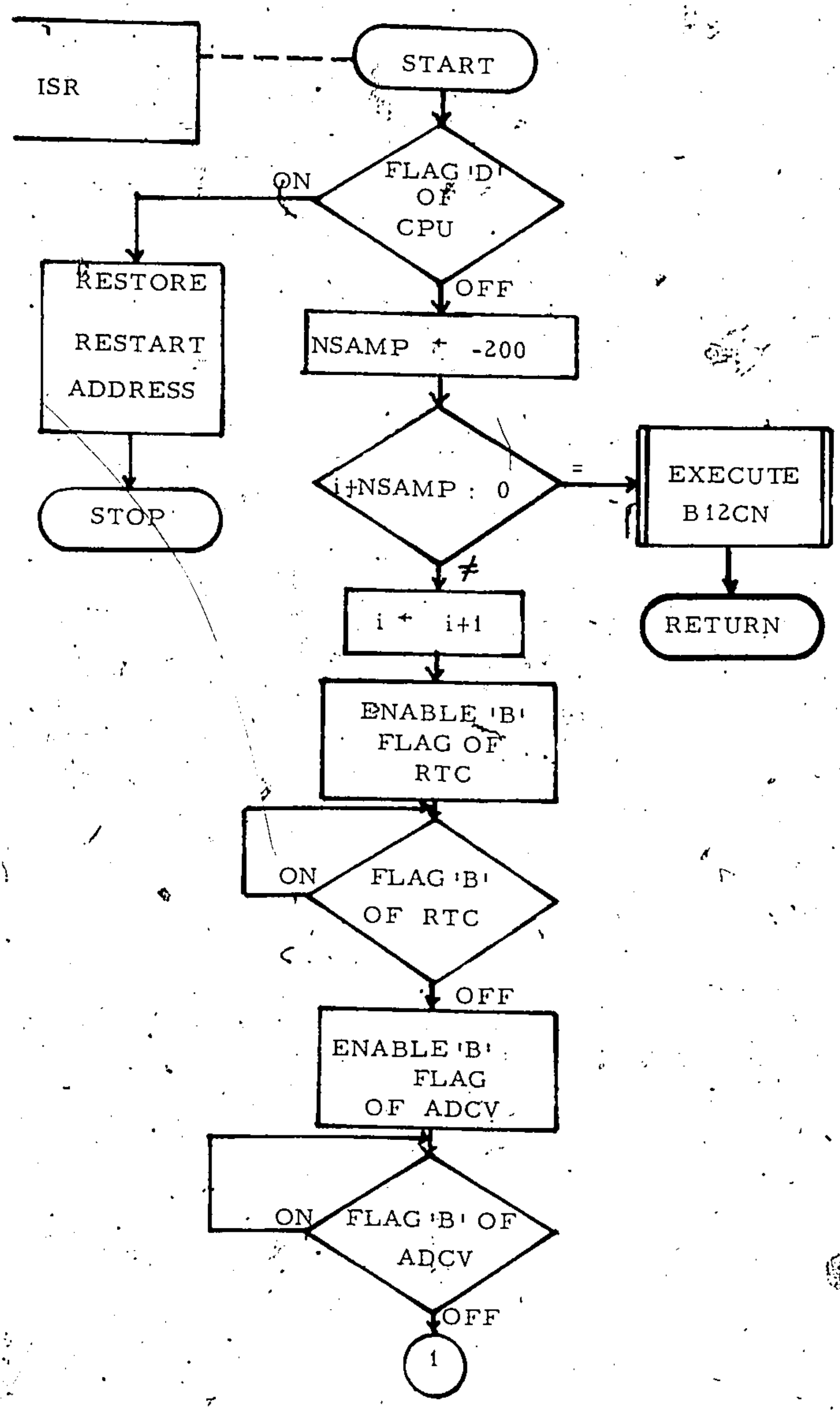


Figure 6 Continued

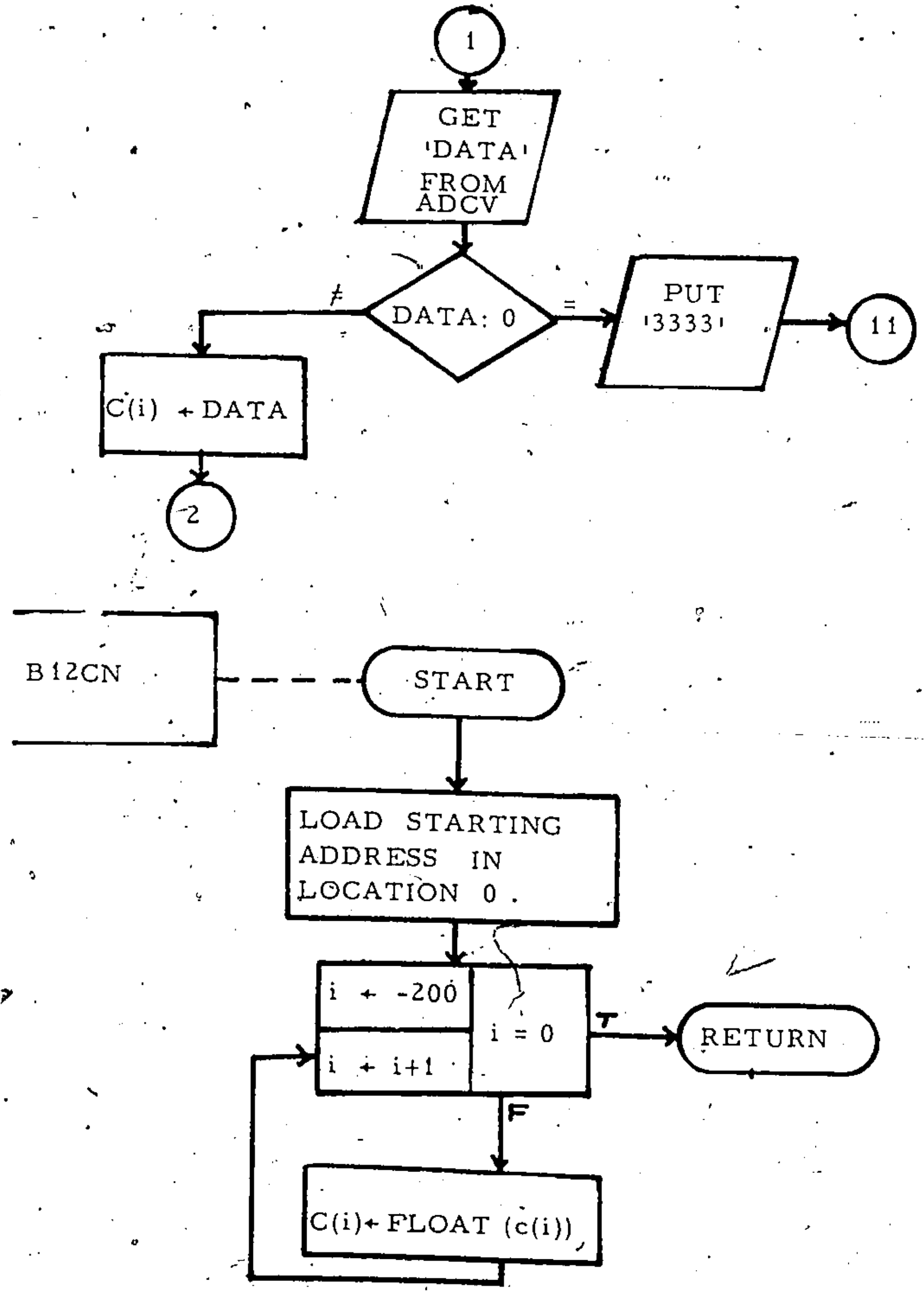


Figure 6 Continued

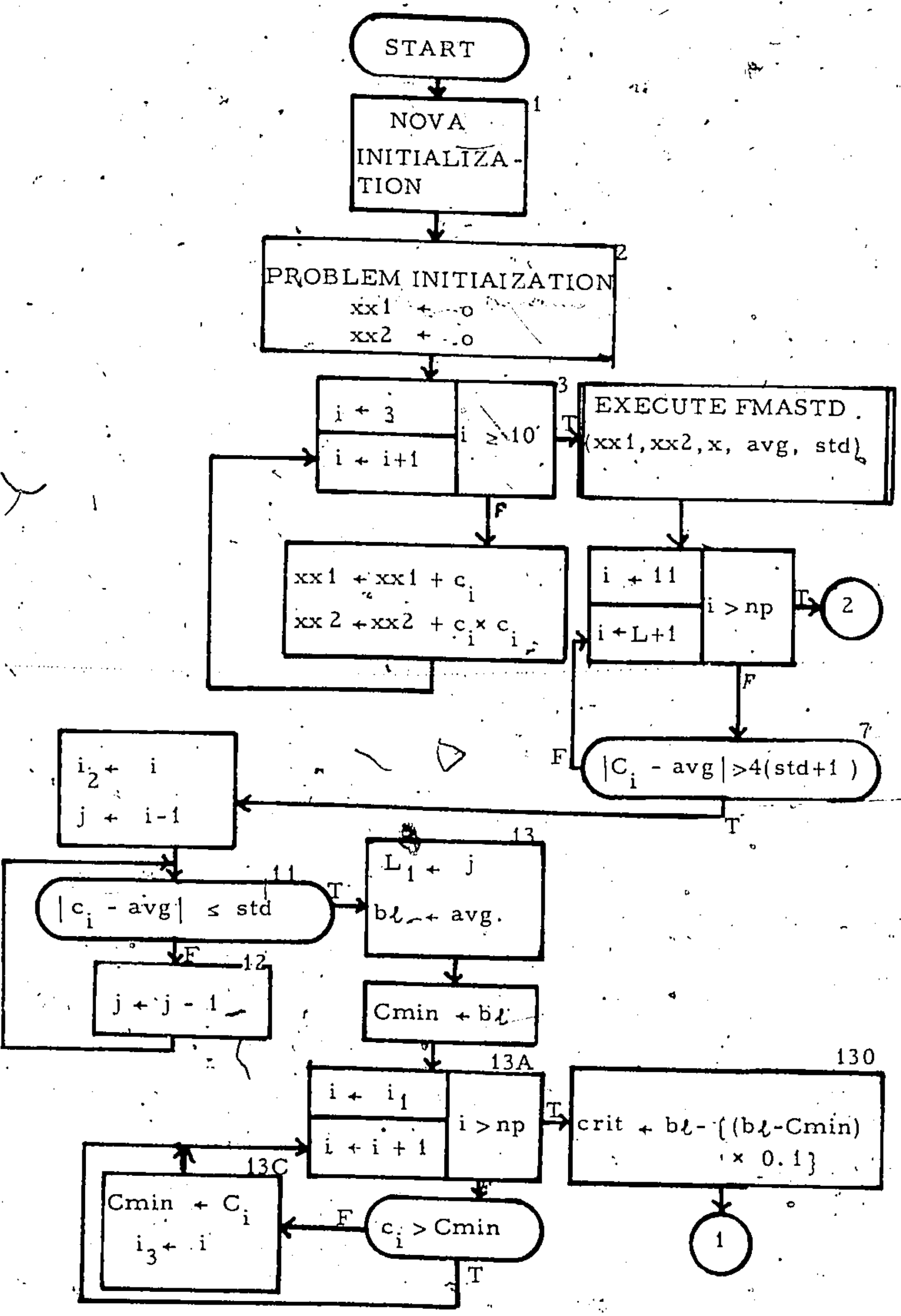


Figure 7

Flow Chart For On-Line Computation Of The Area Under The Thermo-Dilution Curve.

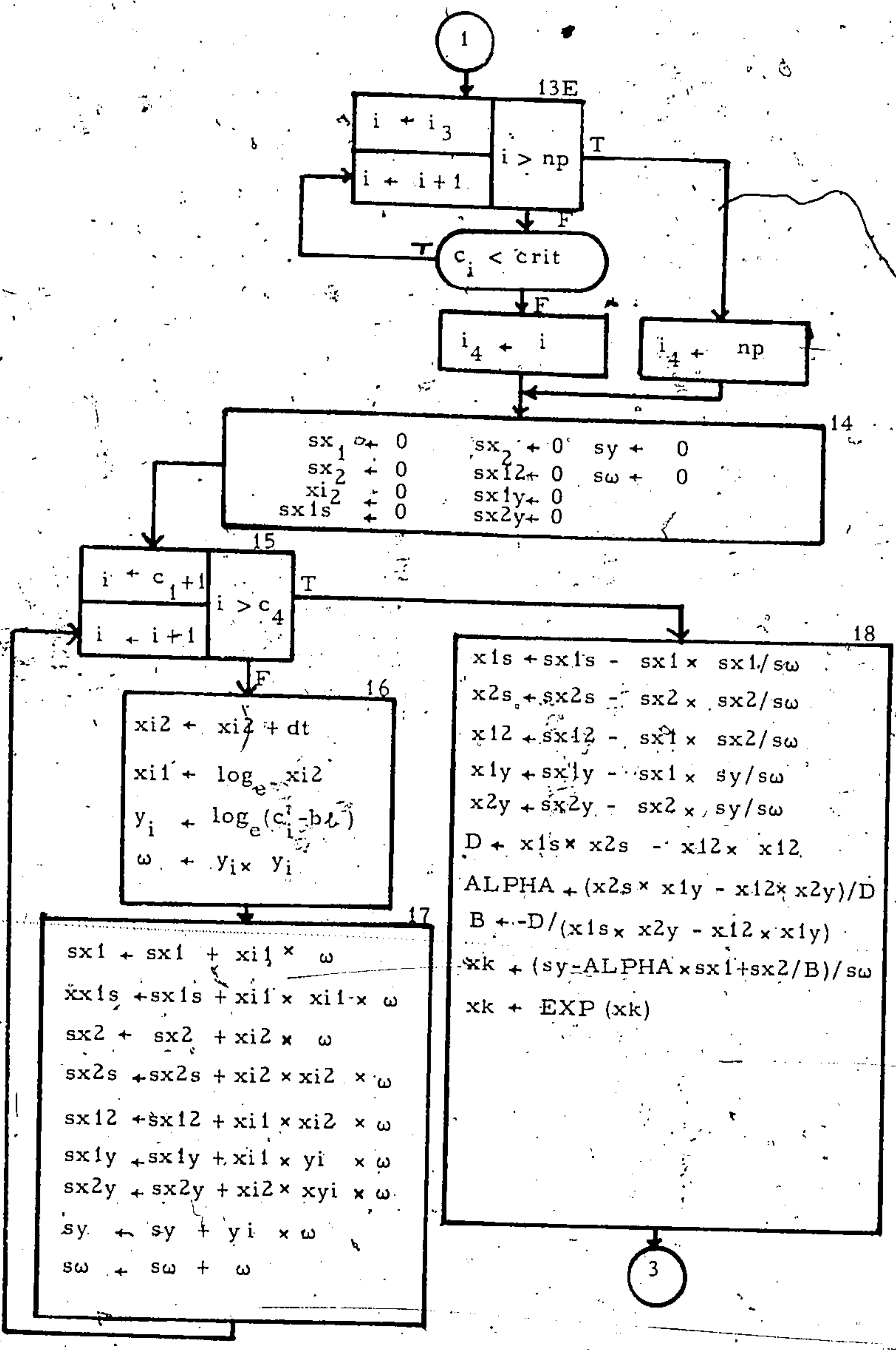


Figure 7 Continued

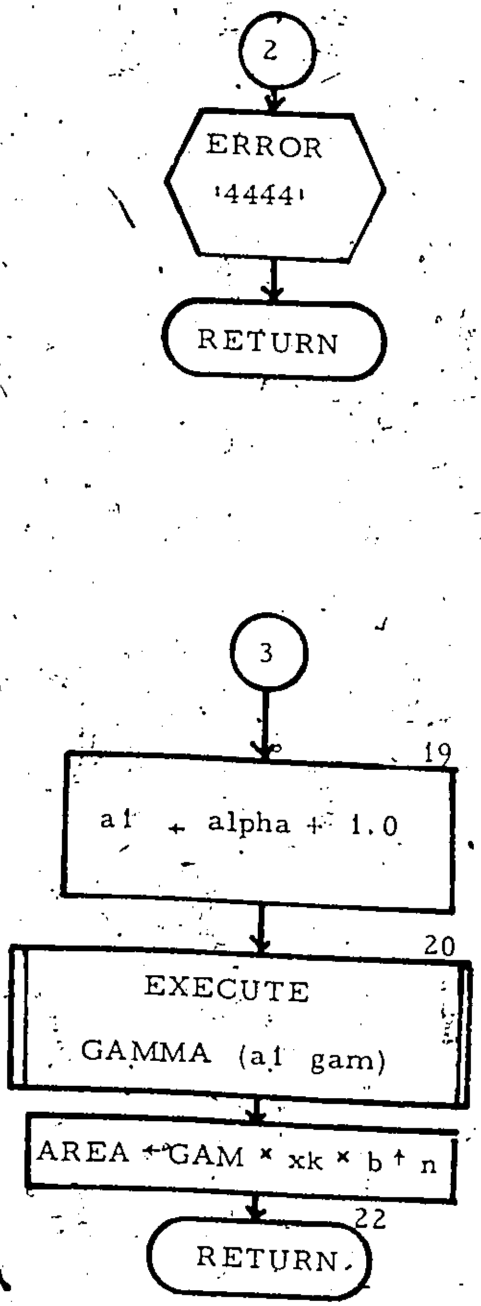
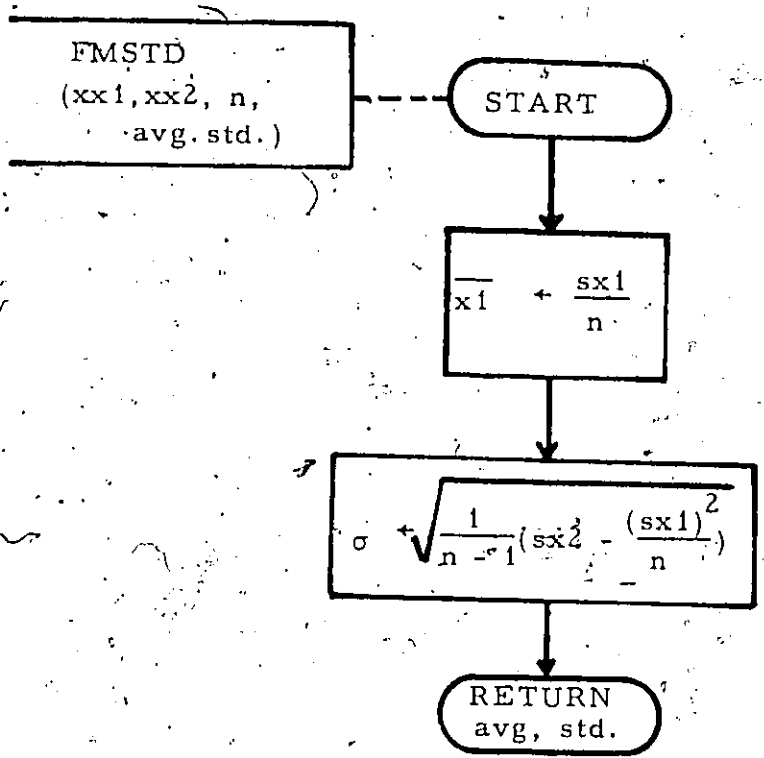


Figure 7 Continued



where $sx1 = \sum_{i=1}^{N} x_i$

$sx2 = \sum_{i=1}^{n} x_i^2$

n = No of Samples.

avg = $\overline{x1}$

std. = σ

Figure 7 Continued

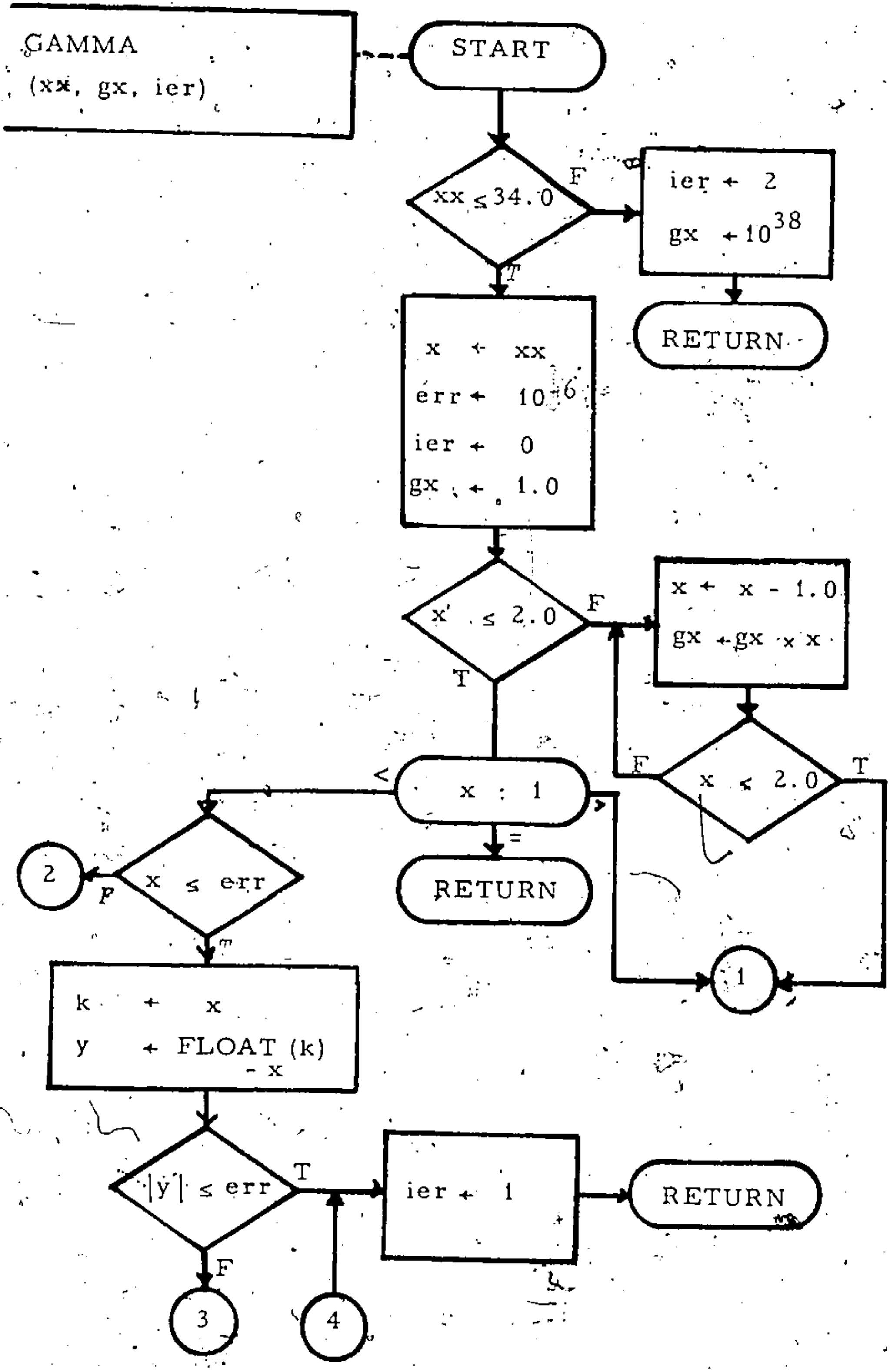


Figure 7 Continued

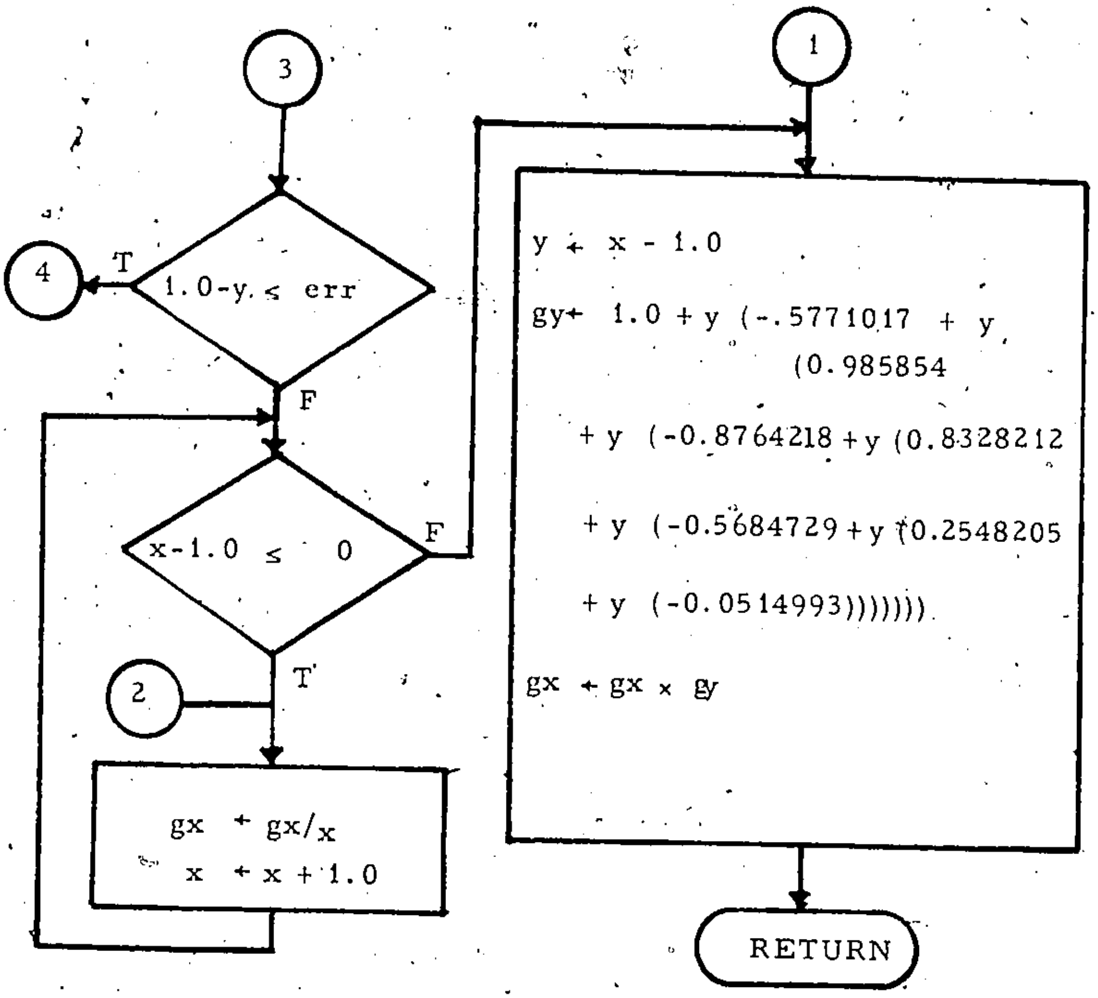


Figure 7 Continued

and subsequently displayed by enabling the 'C' key of the FKB. This completes the procedure for processing a thermo-dilution curve. Repeated measurements can be made by initiating the procedure from Box, 11 or Box, 2 as required.

The support software necessary to install the system for on-line computation of cardiac output from thermo-dilution curves, and keep it running smoothly, are the system loading and initialization programs, debugging aids and the floating point interpreter. A listing of some of these programs along with the software for system initialization is included in Appendix III.

CHAPTER V

SYSTEM TESTING

The technique for testing real-time systems has been found to be more advanced and complex in nature compared to conventional systems. The procedure of first testing the programs and then the overall system is usually adopted, except on large systems, in which case it has been proved to be extremely difficult.

The first phase in testing of the programs involved verifying the procedure for fitting equation (3) to the thermo-dilution curve. Figure 8 shows an algorithm for digital simulation of thermo-dilution curves. The algorithm takes into account, the noise levels in sampled data by appropriate addition of noise signals assumed to be Gaussian in nature. The procedure for fitting equation (3) to the thermo-dilution curve is given in Figure 9. A listing of the codes for the above procedures are included in Appendix III. After several successful tests, the procedure was adopted for on-line computation of cardiac output from the thermo-dilution curve. The procedure was again checked for correctness by comparing the results obtained from tests made on simulated thermo-dilution curves. The next step in testing of the program involved verifying the procedure for processing thermo-dilution curves. This required an analog simulation of the thermo-dilution curve. Figure 10 shows an experimental set-up corresponding to the central blood circulation. The rate of flow of the liquid in the system is determined by the operating speed of the pump. The temperature of the circulating liquid is kept constant by the heat exchanger. RE and MC represent the reservoir and the mixing chamber respectively. The catheter is positioned such that its tip carrying the thermistor lies beyond the mixing chamber along the direction of flow of the liquid. The temperature vs. time curve

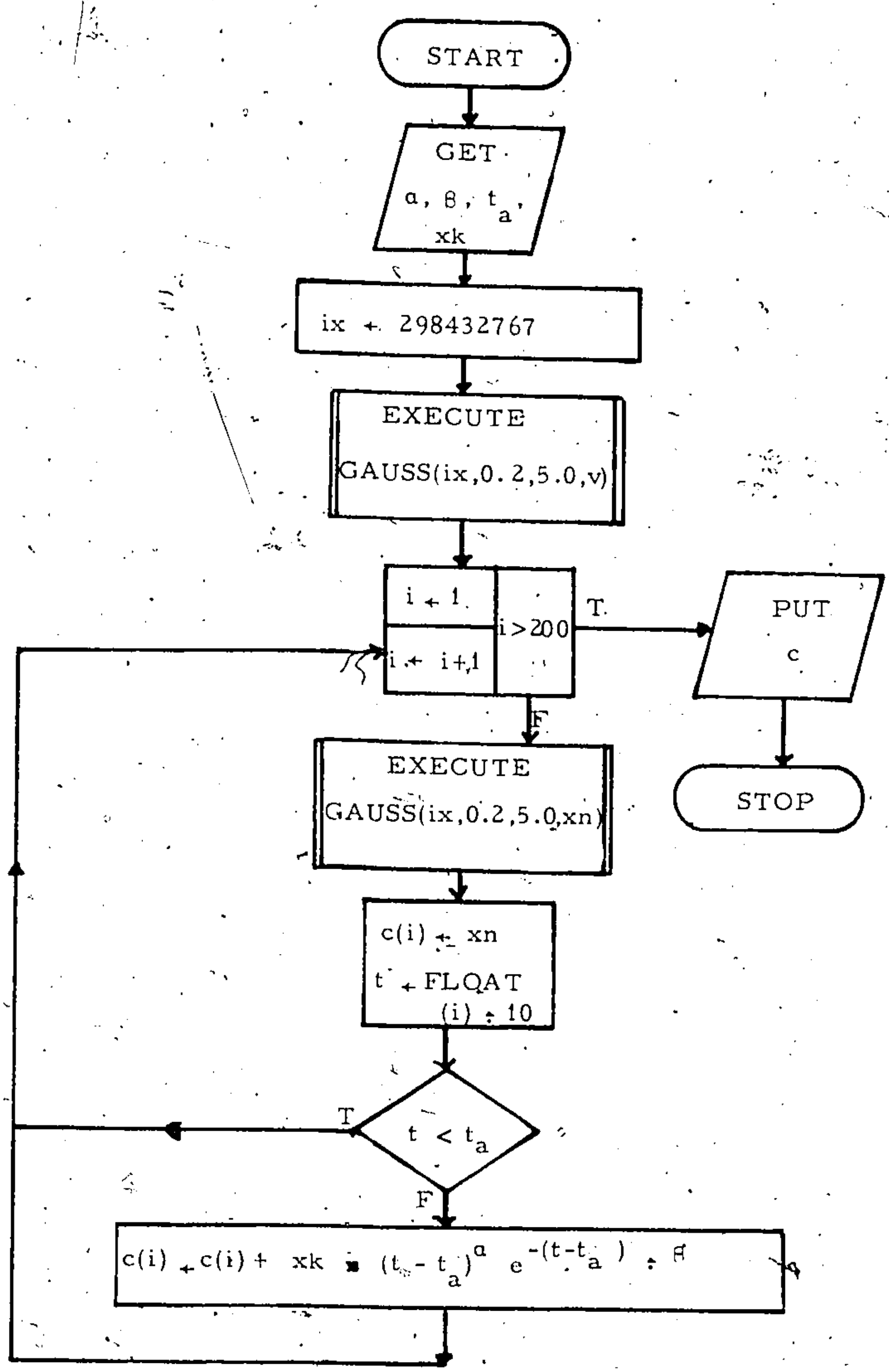


Figure 8
Flowchart For Digital Simulation Of Thermo-Dilution Curve.

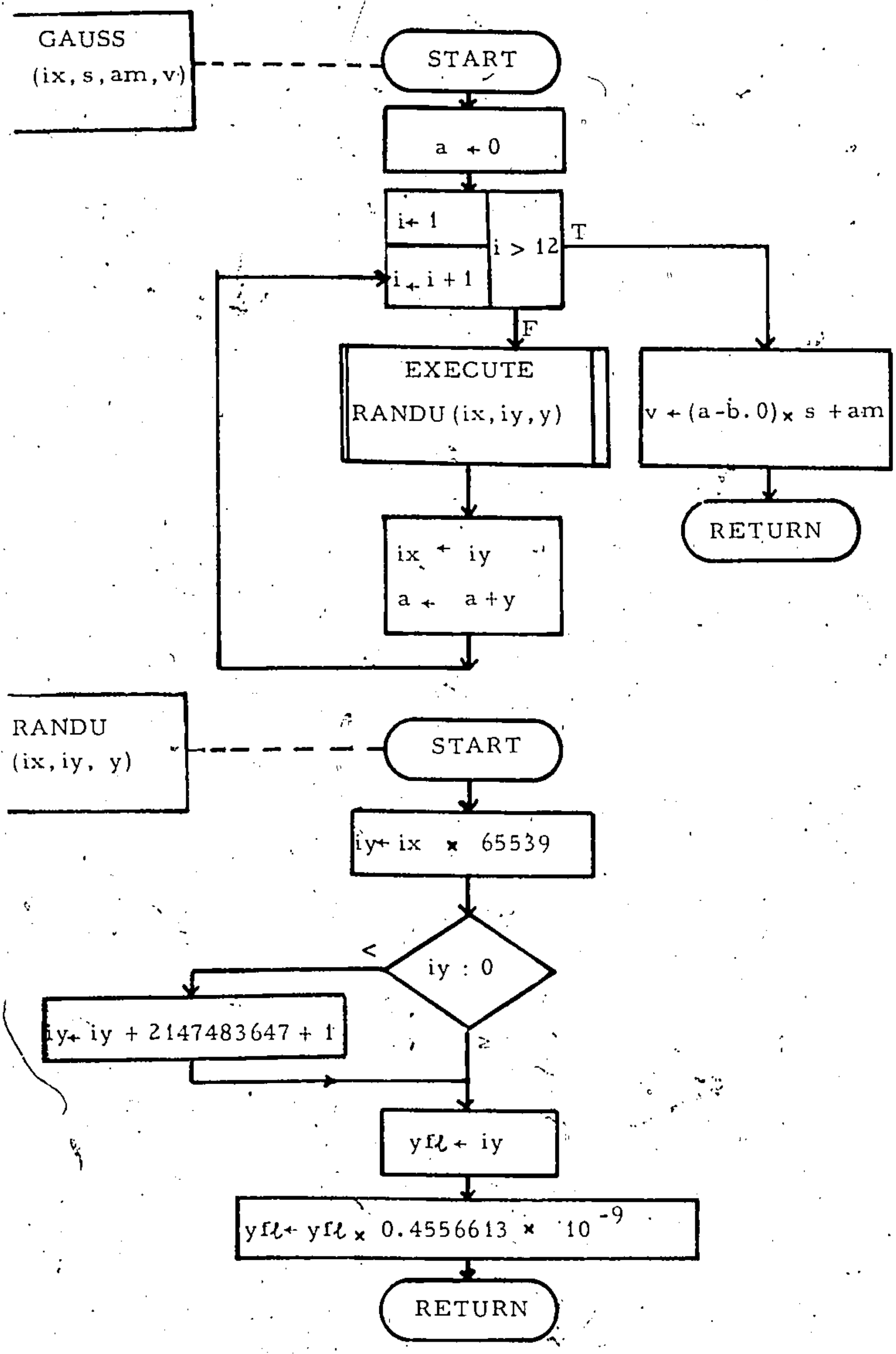


Figure 8 Continued

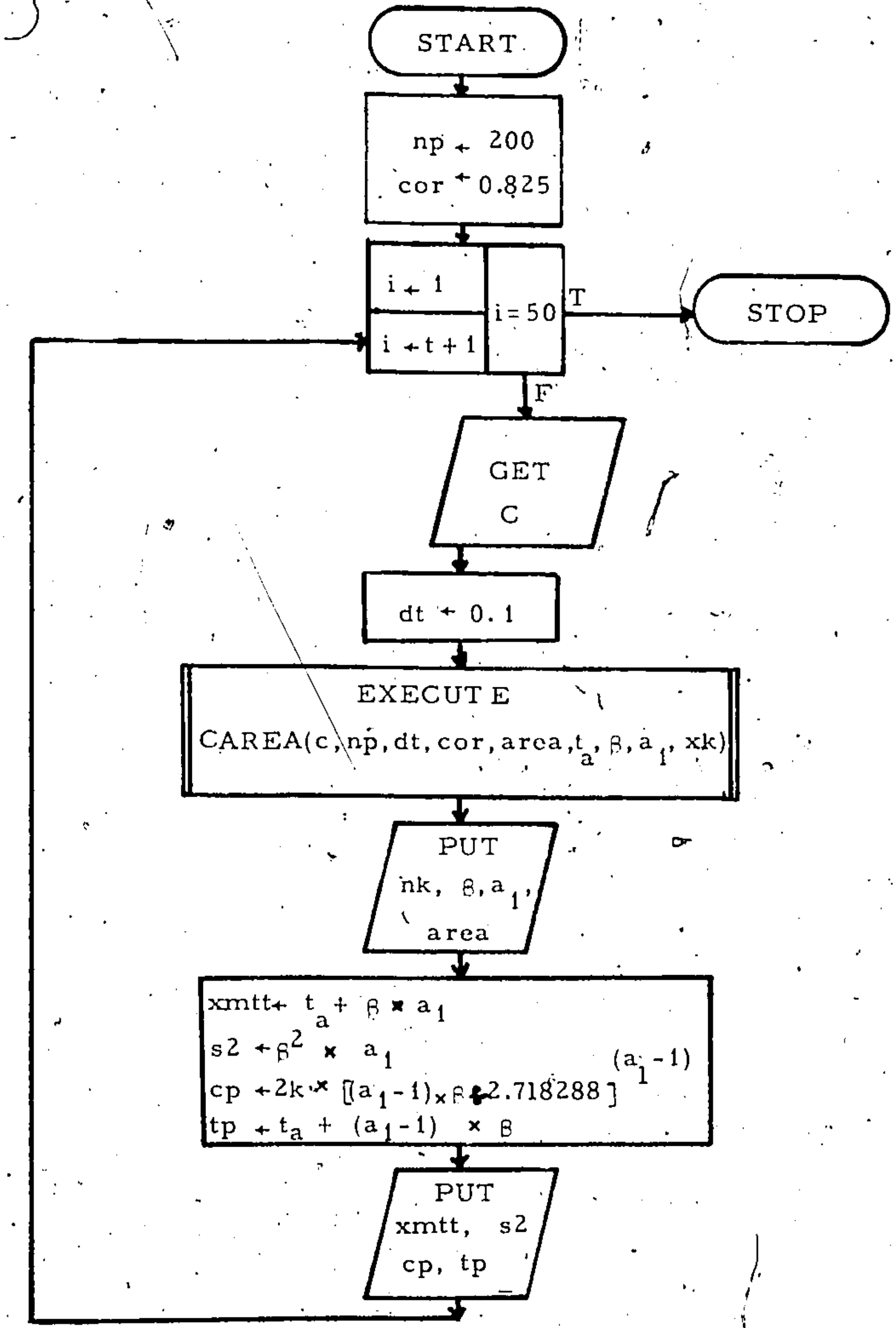


Figure 9

Flowchart For The Analysis Of Simulated Thermo-Dilution Curves

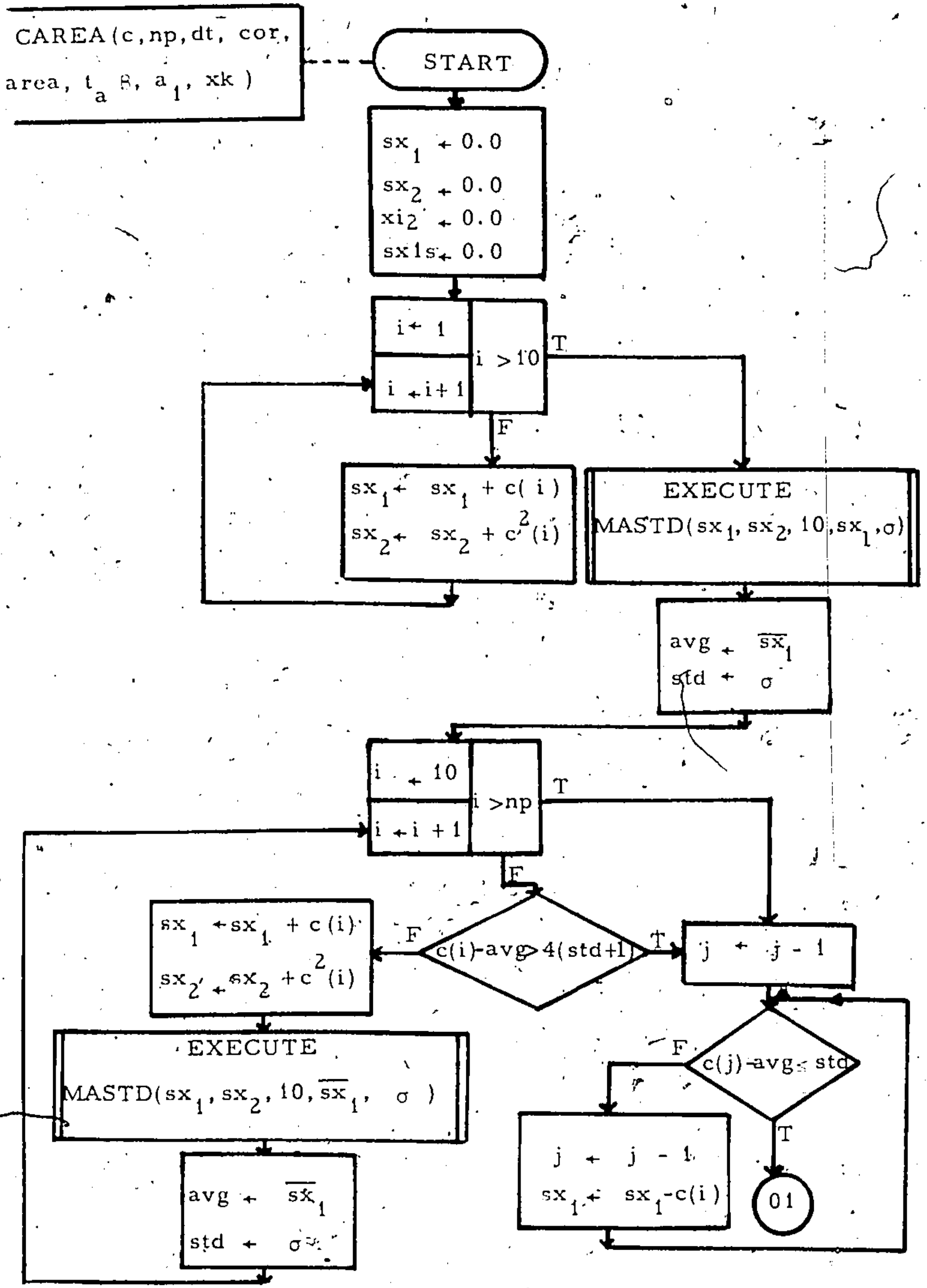


Figure 9 Continued

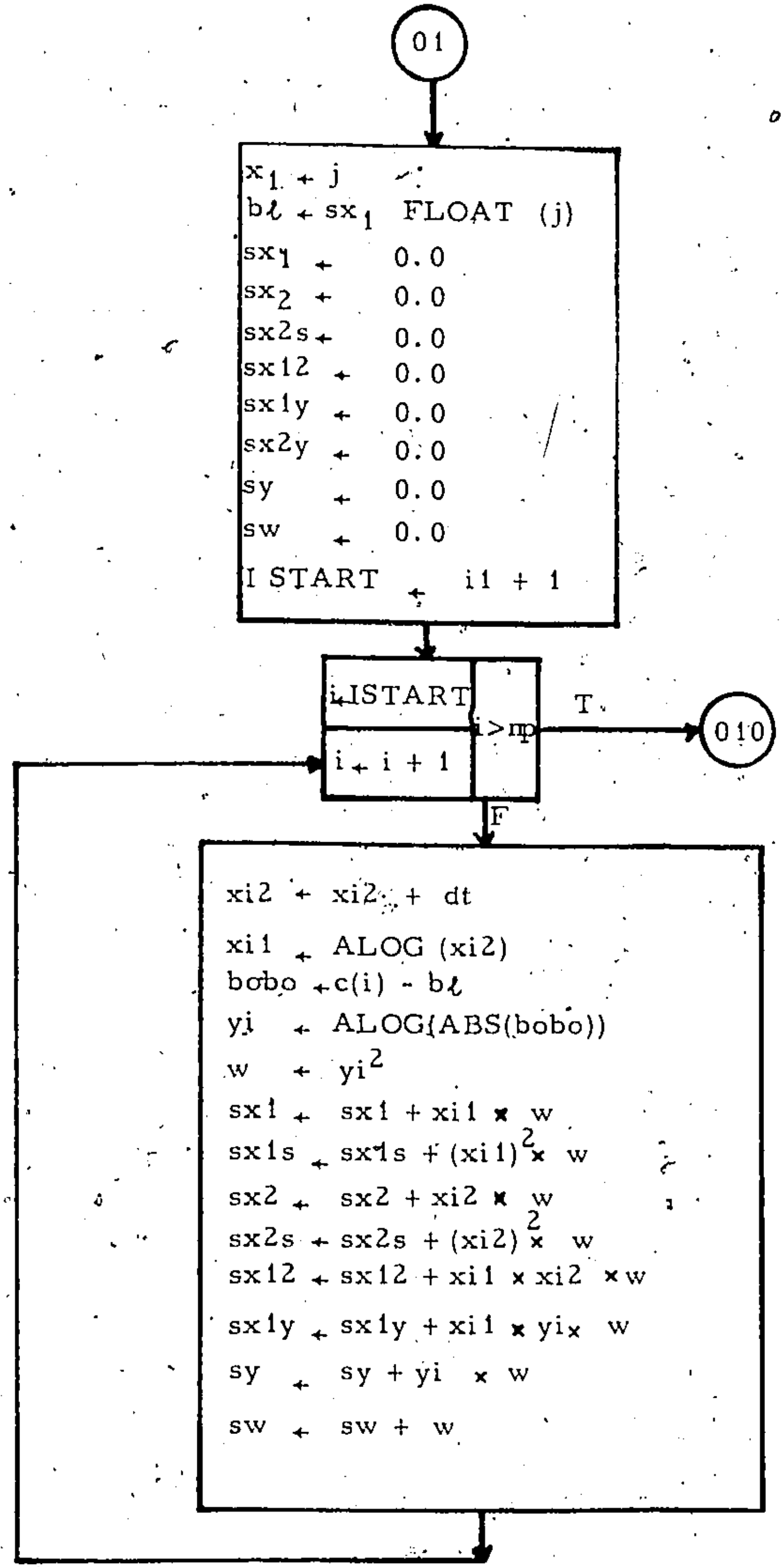


Figure 9 Continued

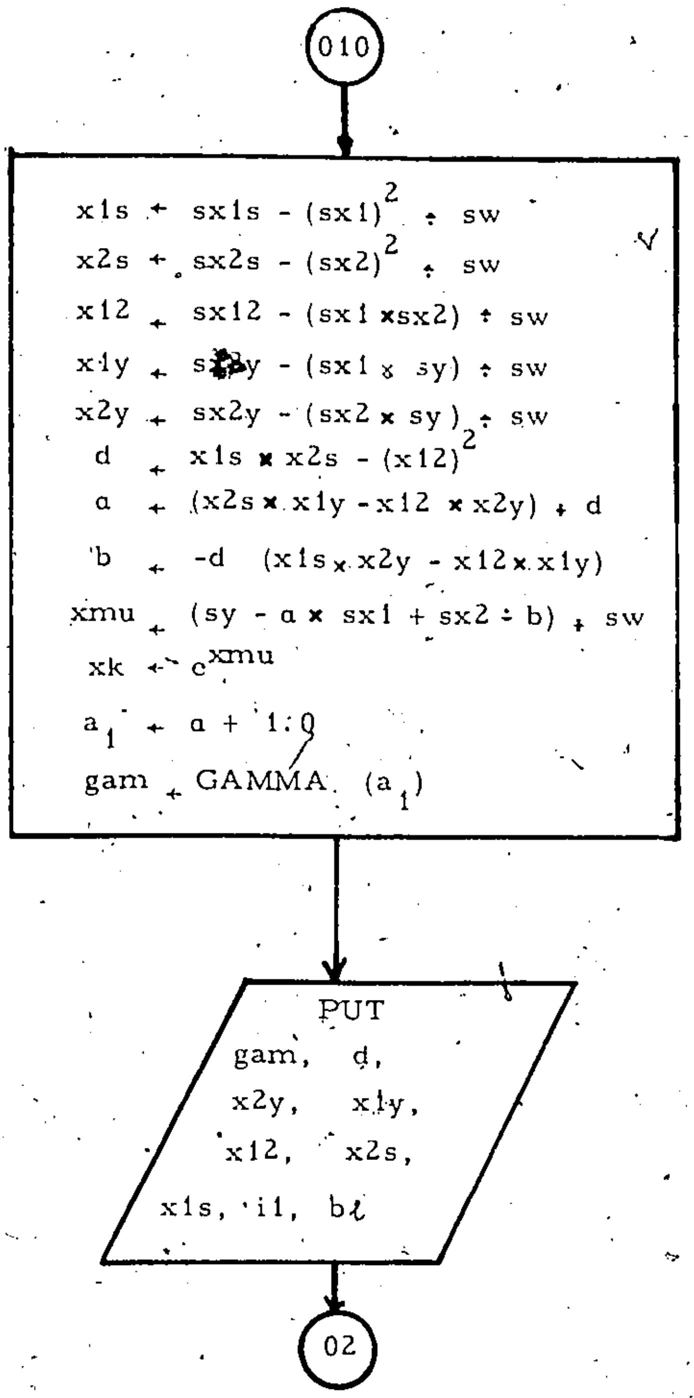


Figure 9 Continued

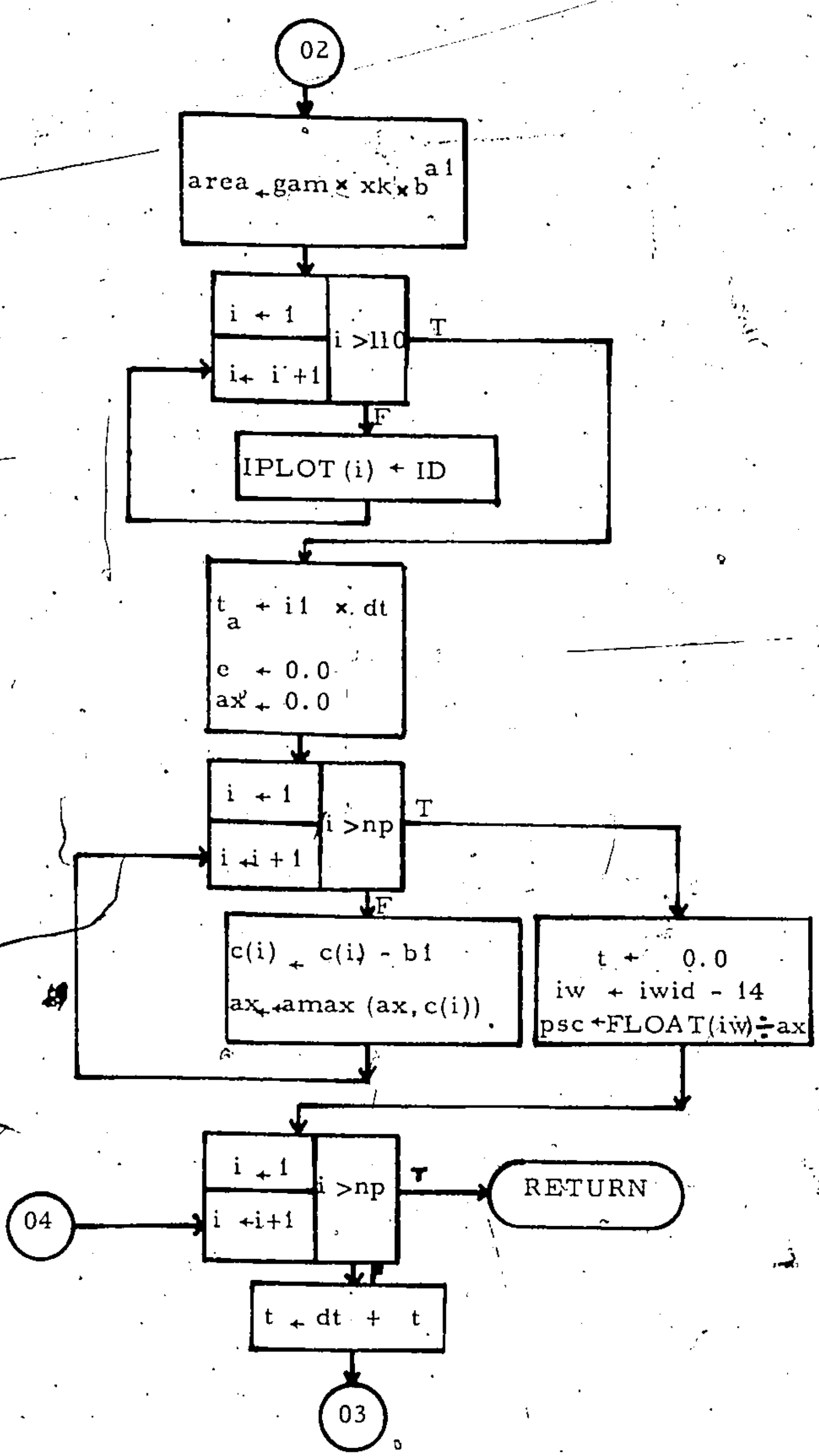


Figure 9 Continued

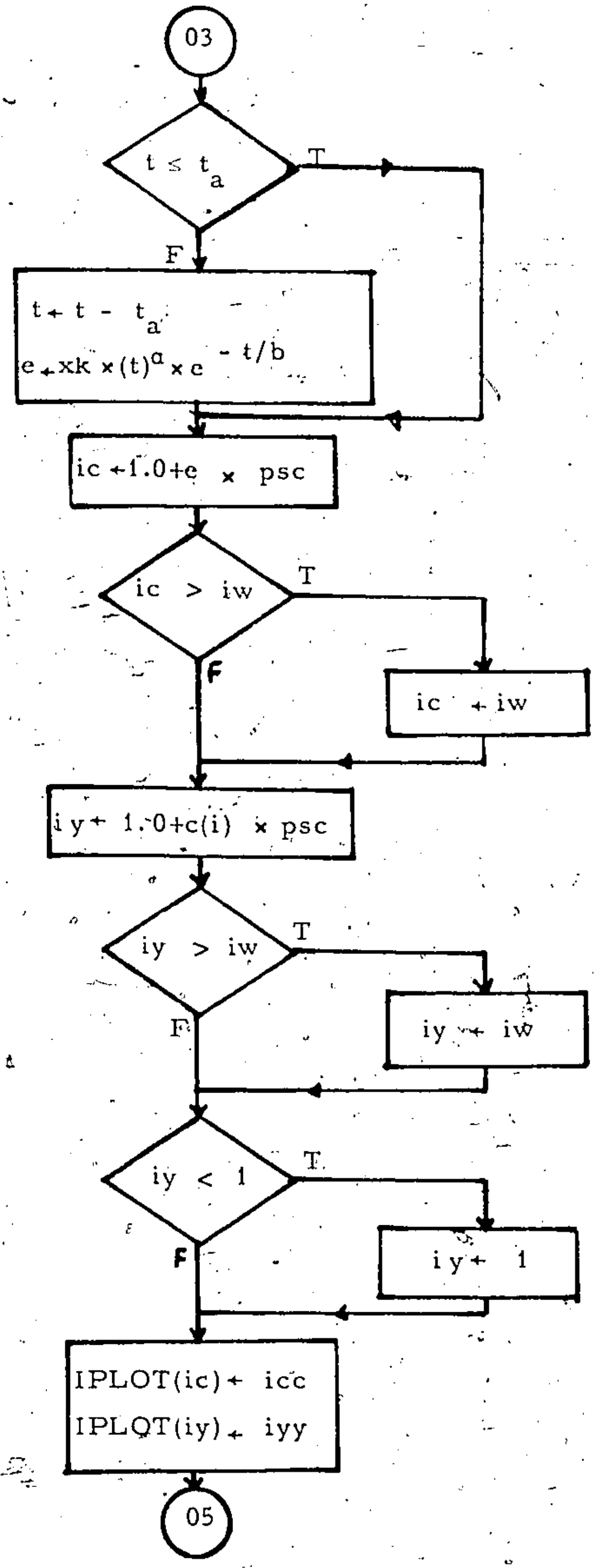


Figure 9 Continued

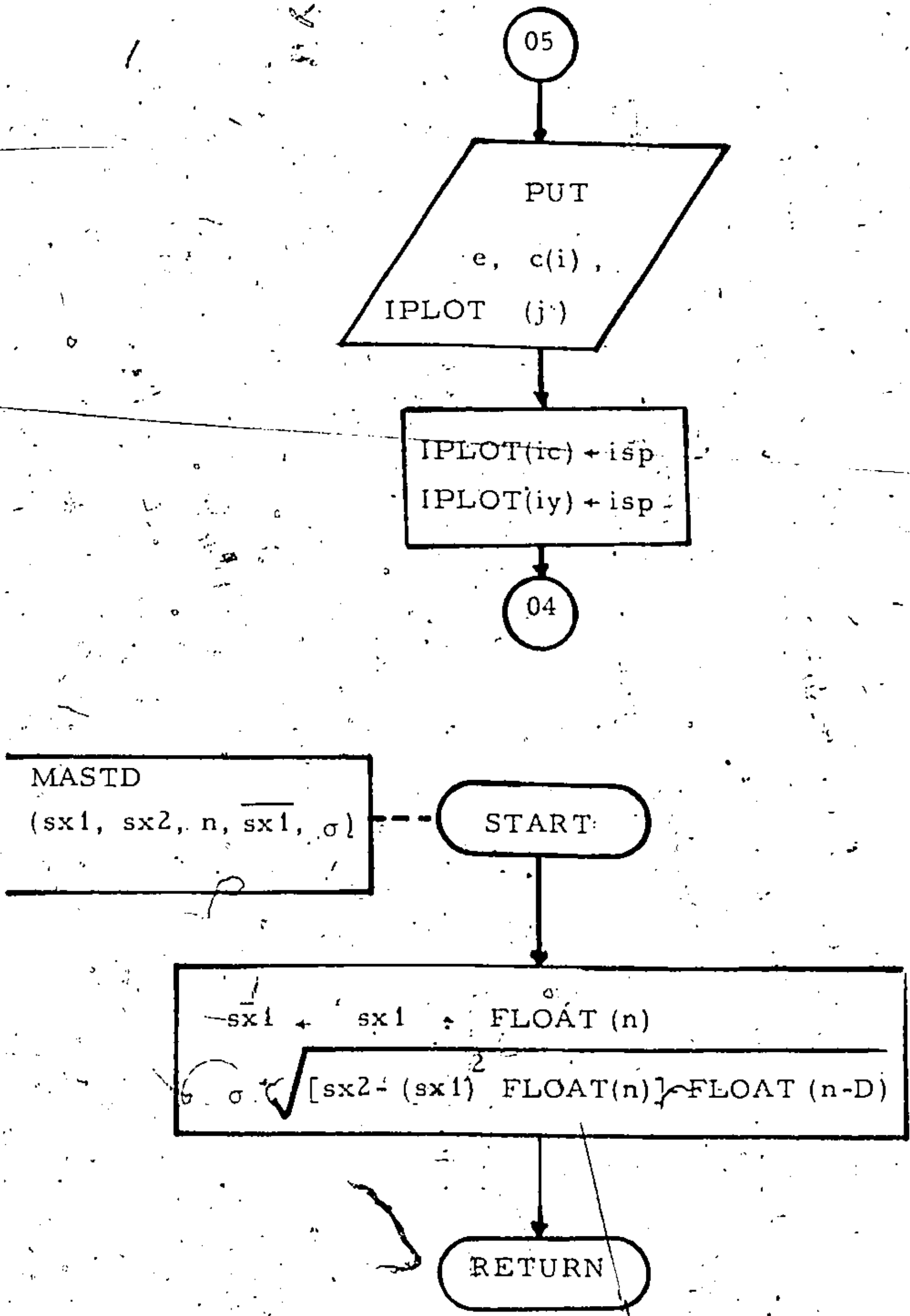


Figure 9 Continued

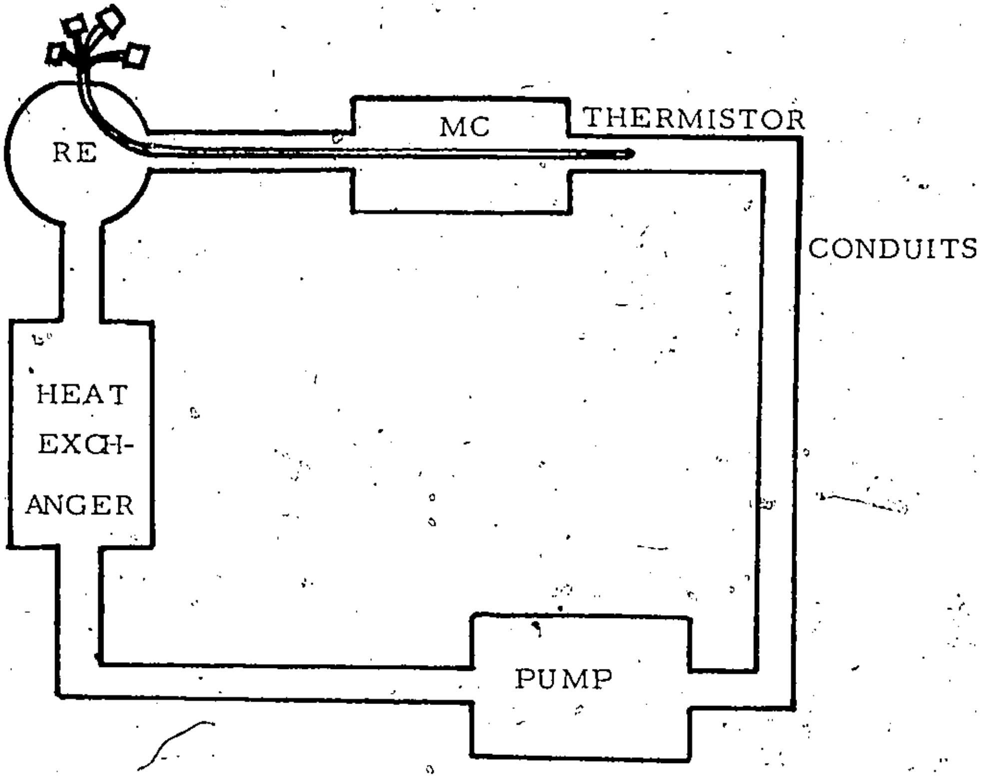


Figure 10

Schematic Diagram of the Experimental Setup For Simulating the Central Blood Circulation obtained following injection of 10 cc of cold dextrose solution was processed and the rate of flow of the liquid in the system was then computed. The hardware interface necessary for processing thermo-dilution curves is shown in Figure 11. It was designed by the Bio-engineering department of the Ottawa Civic Hospital, to interface the catheter thermistor to the A/D converter of the computer.

$R(T)_c$ is the resistance corresponding to the catheter thermistor. The change in resistance of the thermistor as a function of temperature is approximately given by

$$R \propto e^{-1/T}$$

Accordingly, the voltage across the thermistor is proportional to the resistance. The thermistor is supplied by a constant current source and the voltage across the thermistor is amplified using an isolation amplifier. The thermistor circuit employs a logarithmic amplifier to obtain a voltage proportional to the temperature change in the pulmonary blood following injection of the cold dextrose solution. This voltage is made available at the output of a HI/LO gain amplifier. The output of this amplifier serves as input to the A/D converter of the computer and the recorder, which records the thermo-dilution curve. The gain ratio which is defined as the ratio of the slopes of the temperature vs voltage curves of the thermistor circuit at low gain and at high gain is approximately 1/12. The choice of the amplifier gain as well as the operation of the recorder is under computer control.

The software necessary for proper selection of the amplifier gain and the operation of the recorder was tested for correctness. The procedures for measurement of temperature of the injectate, normal temperature of the blood, and the change in pulmonary blood temperature following injection of the cold dextrose solution were individually tested.

The sources of error during development and testing of the

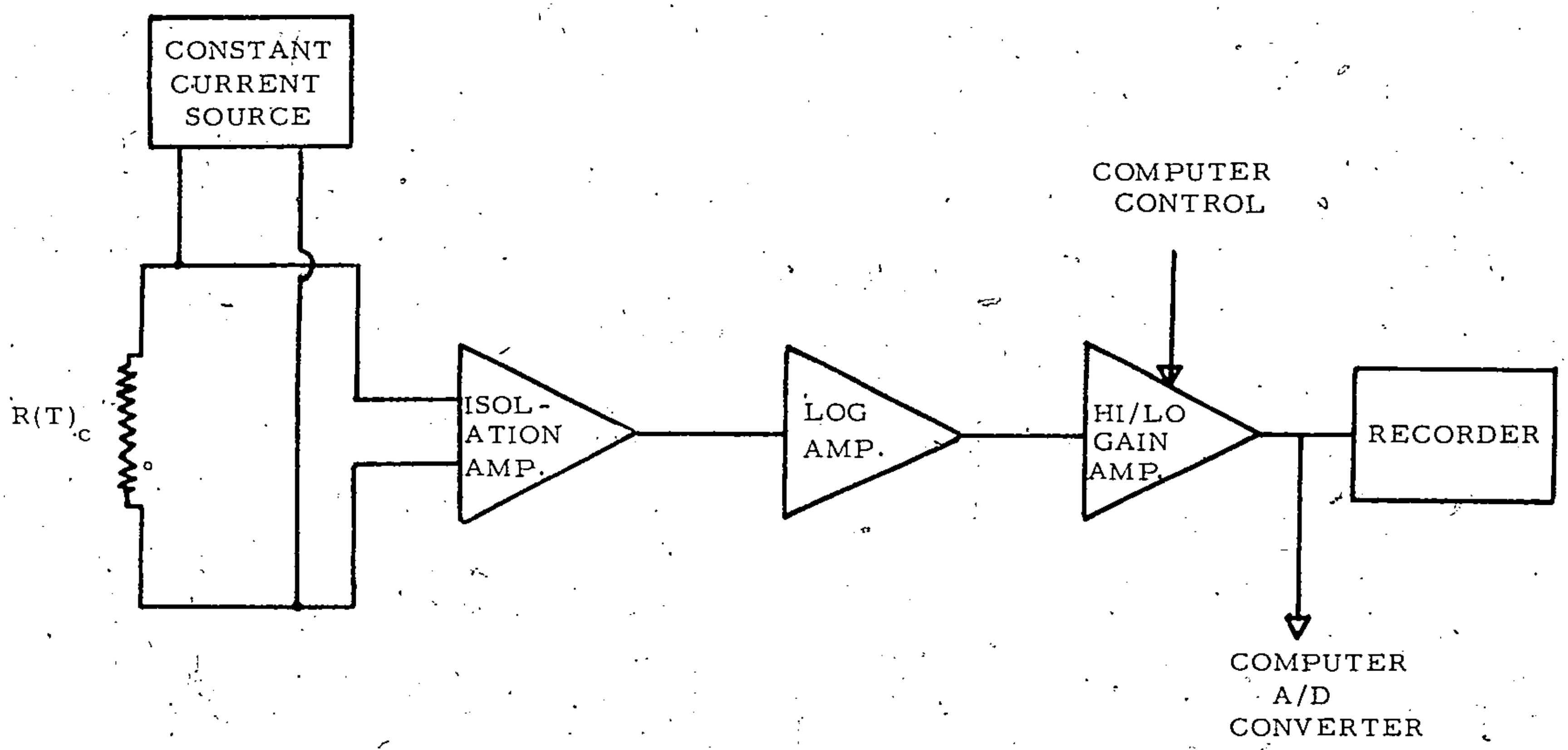


Figure 11

Schematic Diagram of the Thermistor Circuit Interface Unit overall system for on-line computation of cardiac output from thermodilution curves were similar to those found in other real-time systems. Series of tests conducted on the system have shown that the results depended on the rate of injection of the cold dextrose solution and also on the length of the catheter exposed to the atmosphere. Also varying input rates seriously affected the response of the system. Approximations made in fitting equation (3) to the thermo-dilution curve and the drifts in the baseline affect the accuracy of the computed values. Because of the large number of possible combinations of these errors, it has been found difficult to remove the last traces of error from the system.

CHAPTER VI

SYSTEM VALIDATION

Validation of real-time systems is an area requiring a good deal of judgment. To a large extent the problem is a complement to the formulation of the system. Validation of real-time systems are done by comparing histories of performance of the system under consideration with those available for existing systems; or in the absence of such information, the validity of the system is inferred by observing it meets the set requirements. Errors can, of course occur in the design and development of the system. If mathematical modelling is a part of the real-time system, ideally; the errors of the system and programming errors should be separated by validating the mathematical model before embarking on programming. This is not easy to do, however, because usually the mathematical model is intractable. It may be possible to solve special cases, for example by removing all randomness, but as a rule the validation is carried out by comparing computed results to expected standard results.

The validity of the real-time system for on-line computation of cardiac output was done by comparing computed results of cardiac output from tests on simulated thermo-dilution curves with standard measurements made concurrently. Table I shows computed results and standard measurements made from simulated thermo-dilution curves. Figure 12 is a plot of the computed results vs standard measurements. The solid line indicates the desired values of cardiac output.

Sl. No.	Computed Values of C.0, L/min	Standard Measurements of C.0, L/min	Desired Values of C.0, L/min
1	1.15	1.15	1.15
2	2.15	2	2.0
3	3.95	3.25	3.25
4	3.76	3.75	3.75
5	4.78	4.45	4.45
6	5.55	5.45	5.45
7	6.9	5.8	5.8
8	7.3	6.2	6.2

Table I

Computed, Measured and Desired Values of C.0.

CONCLUSIONS

The system for on-line computation of cardiac output from thermo-dilution curves, using a digital mini-computer has proved to be extremely useful as a tool in bio-medical research. The results obtained from tests conducted on simulated thermo-dilution curves are in close correspondence to the expected values.

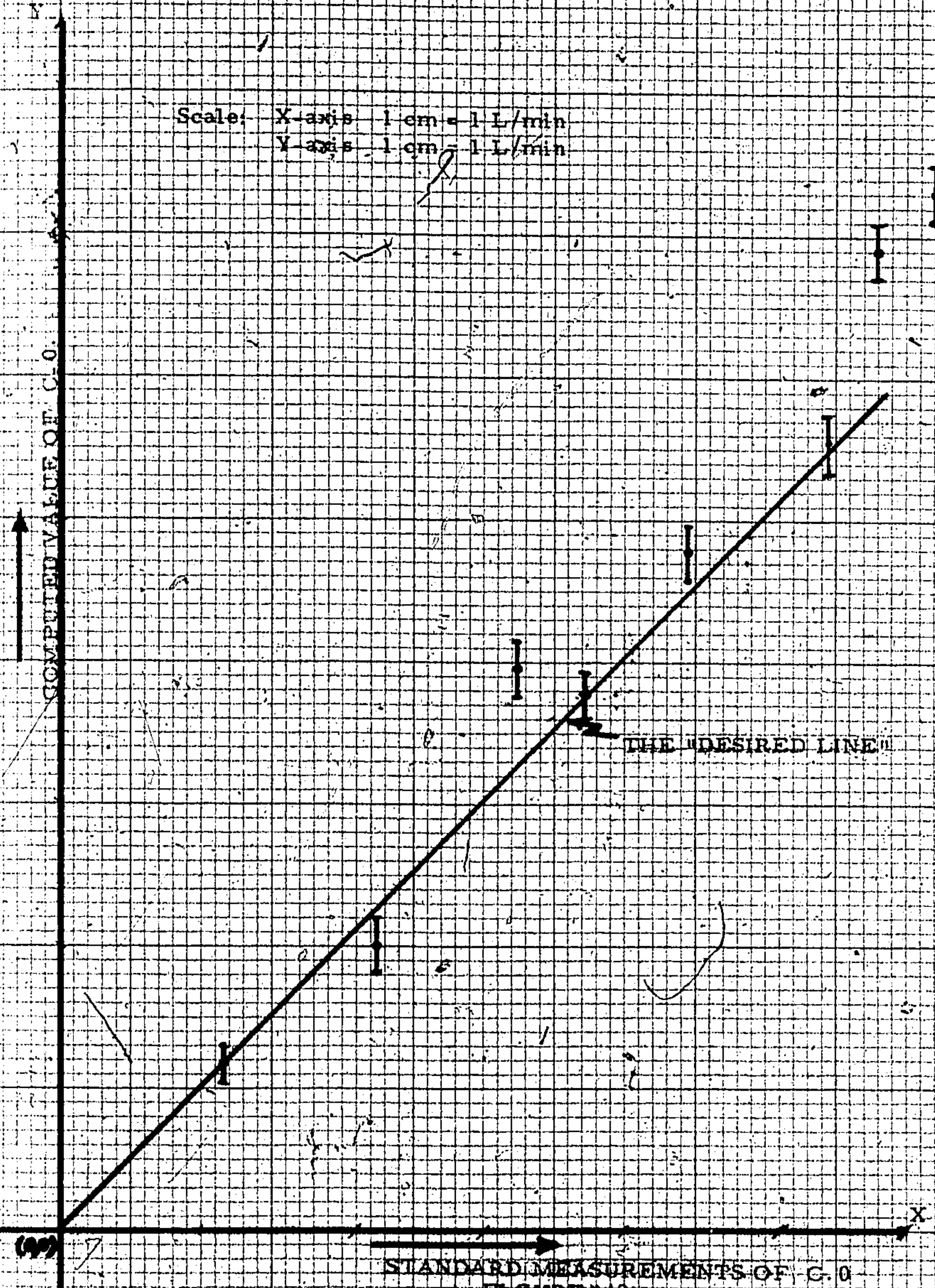


FIGURE 12

GRAPH OF COMPUTED VALUE OF C.O. VS STANDARD MEASUREMENTS OF C.O.

APPENDIX I

WEIGHTED LEAST SQUARE PROCEDURE FOR THE ANALYSIS OF INDICATOR DILUTION CURVES

The following equation is shown elsewhere [29], to represent a good fit for the indicator dilution curves.

$$C(t_i) = k(t_i - t_a)^{\alpha} e^{-(t_i - t_a)/\beta} \quad \ldots\ldots \text{(A1.1)}$$

where k = scale factor,

α, β = arbitrary parameters,

t_a = appearance time,

$C(t_i)$ = indicator concentration at time t_i.

Equation A1.1 is non-linear in k, α and β. This is made linear by taking logarithms, so that the linear least square procedure can be employed to determine the constants k, α and β

$$\ln C(t_i) = \ln k + \alpha \ln (t_i - t_a) - (t_i - t_a)/\beta$$

or

$$y_i = \mu + a_1 x_{i1} + a_2 x_{i2} \quad \ldots\ldots \text{(A1.2)}$$

where $y_i = \ln [C(t_i)]$ $\quad a_1 = \alpha \quad x_{i1} = \ln (t_i - t_a)$

$\mu = \ln (k) \quad a_2 = 1/\beta \quad x_{i2} = (t_i - t_a)$

The weighted sum of squares of the deviations (δ_i) of the fitted curve can be written as

$$S = \sum_{i=1}^{n} (\delta_i)^2 = \sum_{i=1}^{n} w_i (y_i - \mu - a_1 x_{i1} - a_2 x_{i2})^2 \ldots\ldots \text{(A1.3)}$$

Where w_i are the weights corresponding to each value of the polynomial given by A1.2, and are taken to be inversely proportional to the error variance in the determination of $C(t_i)$. The error in the determination of $C(t_i)$ is approximately constant. Howevor, since a logarithmic model is assumed, the error is no longer constant, but varies inversely with the square of the indicator concentration. Therefore

$$w_i = C^2(t_i)$$

Equation A1.3 can be expanded to the following form.

$$\begin{aligned} S = {} & \sum_{i=1}^{n} w_i y_i^2 + \sum_{i=1}^{n} w_i \mu^2 + \sum_{i=1}^{n} w_i a_1^2 x_{i1}^2 \\ & + \sum_{i=1}^{n} w_i a_2^2 x_{i2}^2 - 2 \sum_{i=1}^{n} w_i y_i \mu \\ & + 2 \sum_{i=1}^{n} w_i a_1 \mu x_{i1} + 2 \sum_{i=1}^{n} w_i a_1 a_2 x_{i1} x_{i2} \\ & - 2 \sum_{i=1}^{n} a_2 w_i x_{i2} y_i + 2 \sum_{i=1}^{n} w_i \mu a2 \, x_{i2} \\ & - 2 \sum_{i=1}^{n} w_i y_i a_1 x_{i1} \qquad \ldots\ldots \text{(A1.4)} \end{aligned}$$

To obtain a close fit it is necessary that the deviations be minimal. Minimizing the sum of squares of the deviations will discriminate against large deviations δ_i. In essence, it is necessary to minimize the right hand side of equation A1.3. Since the x_{i1} and x_{i2} are known functions which can be easily evaluated given t_i and t_a, and y_i is a measured quantity, the only unknowns in equation A1.3 are a_1 and a_2. μ is a constant which can be solved using a_1, a_2 and equation A1.2. Hence we can look on A1.3 as a function of the coefficients a_1 and a_2, with all other terms being known constants.

In elementary calculus whenever we wish to minimize a function, we set its first derivatives equal to zero and solve for the unknown variable. The equation A1.3 is minimized by setting the partial derivatives with respect to a_1 and a_2 each equal to zero

i.e $\frac{\partial s}{\partial a_1} = 0$ and $\frac{\partial s}{\partial a_2} = 0$

$$a_1 \sum_{i=1}^{n} w_i x_{i1}^2 + a_2 \sum_{i=1}^{n} w_i x_{i1} x_{i2} + \mu \sum_{i=1}^{n} w_i x_{i1} = \sum_{i=1}^{n} w_i y_i x_{i1} \quad \ldots (A1.5)$$

$$a_1 \sum_{i=1}^{n} w_i x_{i1} x_{i2} + a_2 \sum_{i=1}^{n} w_i x_{i2}^2 + \mu \sum_{i=1}^{n} w_i x_{i2} = \sum_{i=1}^{n} w_i y_i x_{i2} \quad \ldots (A1.6)$$

Equation A1.2 can be rewritten as follows,

$$a_1 \sum_{i=1}^{n} w_i x_{i1} + a_2 \sum_{i=1}^{n} w_i x_{i2} + \mu \sum_{i=1}^{n} w_i = \sum_{i=1}^{n} w_i y_i \quad \ldots (A1.7)$$

Equations A1.5, A1.6 and A1.7 are three linear equations in three unknowns. Solving for a_1 and a_2, we get the following expression

$$a_1 = \frac{A_{1y} A_{22} - A_{12} A_{24}}{A_{11} A_{22} - A_{12}^2}$$

and

$$a_2 = \frac{A_{11} A_{2y} - A_{1y} A_{21}}{A_{11} A_{22} - A_{12}^2}$$

where

$$A_{11} = \sum_{i=1}^{n} w_i x_{i1}^2 - \frac{\sum_{i=1}^{n} w_i x_{i1} \sum_{i=1}^{n} w_i x_{i1}}{\sum_{i=1}^{n} w_i}$$

$$A_{22} = \sum_{i=1}^{n} w_i x_{i2}^2 - \frac{\sum_{i=1}^{n} w_i x_{i2} \sum_{i=1}^{n} w_i x_{i2}}{\sum_{i=1}^{n} w_i}$$

$$A_{12} = A_{21} = \sum_{i=1}^{n} w_i x_{i1} x_{i2} - \frac{\sum_{i=1}^{n} w_i x_{i1} \sum_{i=1}^{n} w_i x_{i2}}{\sum_{i=1}^{n} w_i}$$

$$A_{1y} = \sum_{i=1}^{n} w_i y_i x_{i1} - \frac{\sum_{i=1}^{n} w_i y_i \sum_{i=1}^{n} w_i x_{i1}}{\sum_{i=1}^{n} w_i}$$

and

$$A_{2y} = \sum_{i=1}^{n} w_i y_i x_{i2} - \frac{\sum_{i=1}^{n} w_i y_i \sum_{i=1}^{n} w_i x_{i2}}{\sum_{i=1}^{n} w_i}$$

Substituting the above expressions for a_1 and a_2 in equation A1.7 and solving for μ, we get

$$\mu = \left(\sum_{i=1}^{n} w_i y_i - a_1 \sum_{i=1}^{n} w_i x_{i1} - a_2 \sum_{i=1}^{n} w_i x_{i2}\right) / \sum_{i=1}^{n} w_i$$

from which μ can be computed.

From the above equalities we can obtain k, α and β of equation A1.1 as follows,

$$k = e^{\mu}, \quad \alpha = a_1 \quad \text{and} \quad \beta = -{}^{1}/a_2 .$$

APPENDIX II

NOVA-1200 ASSEMBLER CODE, OF THE APPLICATION PROGRAMS FOR THE ON-LINE COMPUTATION OF CARDIAC OUTPUT FROM THERMODILUTION CURVES

```
        .LOC   377
        JMP    START
 START: LDA    1,A.STR          ;START HERE
        STA    1,0                ;LOAD STARTING ADDRESS AT LOCATION 0
        SUBZL  1,1             ;GENERATE 1 IN AC1  -  THIS WILL BE DELAY TIME
        JSR    @A.B1
 RST:   LDA    0,CCL           ;BOX 2  -  CLEAR DISPLAY
        JSR    @A.PTC
        LDA    0,CP                ;CLEAR STATUS INDICATORS
        JSR    @A.PTC
        FINI
        LDA    2,7
        JMP    .+2
        0C5                        ;WIDTH FIELD
        LDA    0,.-1
        STA    0,121,2             ;SET WIDTH FIELD
        JMP    .+2
        002                        ;DECIMAL PLACES FIELD
        LDA    0,.-1
        STA    0,122,2         ;SET DECIMAL PLACES FIELD
           NICS   ADCV         ;CHECK ADCV
           SKPBZ  ADCV
           JMP    .-1
           DIC    0,ADCV
        MOV    0,0,SZR
        JMP    L3
        LDA    0,CST1
        JSR    @A.PTC
        LDA    1,.+3
        JSR    @A.BINO
        JMP    @A.RST
        1111
 L3:    LDA    0,C1                ;DISPLAY 1
```

```
        JSR   @A.PTC
        JSR   @A.GTC        ;BOX 3
        LDA   1,CC
        SUB#  0,1,SNR
        JMP   .+5
        LDA   1,CE
        SUB#  0,1,SZR
        JMP   .-6
        JMP   L4A
        JSR   @A.B4          ;BOX 4
        LDA   0,VOUT
        STA   0,VI
        LDA   0,VOUT+1
        STA   0,VI+1
        JMP   L5
L4A:    JSR   @A.OBIN        ;BOX 4A
        STA   1,VI+1
        SUBO  1,1
        STA   1,VI
L5:     LDA   0,CCL          ;BOX 5
        JSR   @A.PTC
        LDA   1,VI+1
        JSR   @A.BINO
        JSR   @A.GTC        ;BOX 6
        LDA   1,CC
        SUB#  0,1,SNR
        JMP   .+5
        LDA   1,CF
        SUBW  0,1,SZR
        JMP   .-6
        JMP   @A.RST
        FETR
        FFLO    VI
        FEXT
        LDA   0,CCL   ;BOX 8
        JSR   @A.PTC
        LDA   0,C2
        JSR   @A.PTC
        JSR   @A.B10          ;BOX 10
L11:    JSR   @A.GTC       ;BOX 11
        LDA   1,CC
        SUB#  0,1,SNR
```

```
        JMP     .+5
        LDA     1,CF
        SUB#    0,1,SZR
        JMP     .-6
        JMP     @A.RST
        JSR         @A.B4           ; BOX 7
        LDA         0,VOUT
        STA         0,VB
        LDA         0,VOUT+1
        STA         0,VB+1
        FETR
        FFLD        VB
        FEXT
        LDA     0,CP            ;BOX 11A
        JSR     @A.PTC
        LDA     0,CST34
        JSR     @A.PTC
        JSR     @A.B12          ;BOX12
        LDA     0,CP            ;BOX12A
        JSR     @A.PTC
        JSR     @A.B13          ;BOX13
        JSR     @A.B14          ;BOX14
        JSR     @A.GTC        ;BOX 15
        LDA     1,CC
        SUB#    0,1,SZR
        JMP     .-3
        JSR     @A.B16          ;BOX 16
        JMP     L11
;=====================================================================
;ROUTINE TO READ CHARACTERS FROM THE TELETYPE. AS
;EACH CHARACTER IS READ . IT IS ECHOED. IF A CARRIAGE
;RETURN CHARACTER IS INPUT, THE ROUTINE AUTOMATICALLY
;GENERATES A LINE FEED.
;CALLING SEQUENCE:
;           JSR     TTYIN
;OUTPUT:
;           AC0=CHARACTER RIGHT JUSTIFIED
TTYIN:      STA     3,SGET          ;SAVE RETURN ADDRESS
            NIOS    TTI             ;START TELETYPE
            SKPDN   TTI             ;WAIT FOR INPUT
            JMP     .-1
            DIAC    0,TTI           ;GET CHAR AND CLEAR TTY
            LDA     1,MSK           ;AC1=177
            AND     1,0             ;KEEP RIGHT 7 BITS
            JSR     .+10            ;OUTPUT CHARACTER
            LDA     3,CR            ;CHECK FCR CR
            SUB     0,3,SZR         ;SKIP IF CR
            JMP     @SGET           ;NCT CR - RETURN
            LDA     0,LF            ;CR GENERATE LF
            JSR     .+3
            LCA     0,CR            ;RESTORE CR
            JMP     @SGET           ;RETURN
;ROUTINE TO OUTPUT CHARACTERS ON THE TELETYPE. IF
;THE CHARACTER OUTPUT IS NULL, THE ROUTINE WILL
;AUTCMATICALLY GENERATE A CR AND LF.
;CALLING SEGUENCE:
```

```
;       JSR             TTYOT
;INPUT:
;            ACO=CHARACTER RIGHT JUSTIFIED
TTYOT:       SKPBZ   TTC          ;WAIT UNTIL NOT BUSY
             JMP      .-1
             DOAS    0,TTO        ;OUTPUT CHARACTER
             MCV     0,0,SZR      ;SKIP IF NULL CHAR
             JMP     0,3          ;NULL RETURN
             STA     3,SFUT       ;SAVE RETURN
             LDA     0,CR         ;CUTPUT CR
             JSR      .-7         ;RECURSE
             LDA     0,LF         ;OUTPUT LF
             JSR      .-11        ;RECURSE
             SUB     0,0          ;RESTORE NULL
             JNP     @SPLT        ;NOT NULL- RETURN
SGET:         0
SFUT:         0
CR:          15
MSK:        177
LF:          12
;===========================================================================
      B1:       INTDS
                STA 3,E1RA      ; SAVE RETURN ADDRESS
                STA 0,CSAV
                STA 1,CSAV+1
                STA 2,CSAV+2    ;SAVE ACCUMULATORS 0-2
                MOVL 0,0
                STA 0,CARRY     ; SAVE CARRY
                LDA 0,C
                STA 0,STATE     ; SAVE CONTENTS OF LOCATION 0
                MOVZL  1,2
                ADDZL  2,1  ;MULTIPLY AC1 BY 6
                MOV    1,2
                ADDZL  1,1
                ADDZL  2,1  ;MULTIPLY AC1 BY 10
                NEG    1,1
                STA    1,NUMB
                SUBO 0,0
                STA 0,[illegible]
                JMP .+1
                DOAS 0,RTC      ; START THE RTC
                LDA 1,MASK
                MSKO 1          ; SET MASK
                JMP .+2
                INTRP
                LDA 1,.-1
                STA 1,1         ; STORE ADDRESS OF INTRUPT ROT. IN 1
                JMP .+1
```

```
                    SUBZL 1,1 ;GENERATE 1 IN AC1
                    LDA 3,N
                    INTEN
                    JMP .+1
                    JMP .+1
                    JMP .-2
          INTRP:    SKPDN CPU ;CHECK POWER MONITOR FLAG
                    JMP    .+4
                    LDA    0,A.STR
                    STA    0,0  ;SET USUAL RESTART ADDRESS
                    HALT        ;STOP HERE IF POWER FAILURE
                    LDA    0,NUMB
                    ADD 3,0,SNR
                    JMP B1RE           ; JUMP TO RETURN ROUTINE.
                    ADD    1,3
                    NIOS RTC
                    SKPDZ RTC
                    JMP .-1
                    JMP .+1
                    JMP .+1
                    INTEN
                    JMP @0
          B1RE:     LDA 0,STATE
                    STA 0,0            ; RESTORE LOCATION 0
                    LDA 0,CARRY
                    MOVR 0,0           ; RESTORE CARRY
                    LDA 0,DSAV
                    LDA 1,DSAV+1
                    LDA 2,DSAV+2
                    LDA 3,B1RA         ; RESTORE ACCUMULATORS 0-3
                    JMP 0,3
          B1RA:  0
          STATE: 0
          DSAV: 0
          .BLK  5
          NUMB: 0
          CARRY:0
          N:0
;=================================================================
;B 4 IS THE MODULE WHICH
;  1) MEASURES VOLTAGE CORRESPONDING TO TEMPERATURE (IN M UNITS)
;  2) STORES THE VALUE IN VCUT  (IN DPF FORMAT)
B4:      STA    3,B4RT             ;SAVE RETURN
         NICS   ADCV
         SKPBZ  ADCV
         JMP    .-1
         DIC    0,ADCV
         MOV    0,0,SZR
```

```
          JMP    .+7
          LDA    0,CST1
          JSR    @A.PTC
          LCA    1,.+3
          JSR    @A.BINC
          JMP    @A.RST
          2222
          STA    0.VOUT+1  ;BOX 4 - VALUE IS STORED IN DPF FORMAT
          SUBO   0,0
          STA    0,VOUT
          JMP    @B4RT
B4RT :    000
;=========================================================================
;B10 IS THE MODULE WHICH
;     1) GETS THE VALLE OF BSA FROM THE TTI
;     2) PRINTS THE VALUE OF BSA ON THE TTO
; IT IS NECESSARY TO USE THE FLOATING POINT
; INTERPRETER WITH THIS ROUTINE
;
B10:     STA    3,B10RT     ;STORE RETURN ADDRESS
         FINI               ;INITIALIZE INTERPRETER
         FFTR               ;ENTER INTERPRETER
         FDFC   0           ;GET BSA AND ENTER IN AC0
          FDFC   1
          FLDA   2,B10RT+1
          FDIV   2,1
          FADD   1,0
         FFDCF  0           ;PRINT THE VALUE JUST OBTAINED
         FSTA   0,BSA       ;STORE THE VALUE IN BSA
         FEXT               ;EXIT FRCM INTERPRETER
         JMP    @B10RT      ;RETURN
B10RT:   CC0
          041144                  ;100  F.P.
          000000
;=========================================================================
B12:     INTDS
        STA    3,B12RT     ;STORE RETURN ADDRESS
        SUBO    0,0
        STA    0,I         ;INITIALIZE I TO 0
        LCA    0,FREQ
        JMP    .+1
        DOAS   0,RTC       ;SET RTC WITH FREQUENCY FREQ
        LDA    0,C+1
        STA    0,21
        LDA    1,MASK
        MSKO   1           ;SET MASK
        JMP    .+2
        ISR
```

```
        LDA    1,.-1
        STA    1,1         ;LOAD ADDRESS OF ISR IN 000001
        JMP    .+1
        LDA    2,NSAMP
        MOV    2,3
        ADD    3,2
        STA    2,30
        LDA    2,C+1
        STA    2,20
        SUBO   0,0
        STA    0,@20          ;CLEAR ALL ELEMENTS OF C
        ISZ    30
        JMP    .-2
        SUBZL  1,1       ;AC1 CONTAINS 1
        INTEN
        JMP    .+1
        JMP    .+1
          JMP    .-2
ISR:    SKPDN  CPU            ;CHECK POWER MONITOR FLAG
        JMP    .+4
        LDA    0,A.STR
        STA    0,0            ;SET USUAL RESTART ADDRESS
        HALT                  ;STOP HERE IF POWER FAILURE
        LDA    3,I
        LDA   0,NSAMP
        ADD   3,0,SNR
        JMP    B12CN
        ADD    1,3
        STA   3,I
        NIOS    RTC
        SKPDZ   RTC
        JMP    .-1
        JMP   .+1
        JMP    .+1
        JMP    .+1
        NIOS    ADCV
        SKPBZ   ADCV
        JMP    .-1
        DIC    0,ADCV
        MOV    0,0,SZR
        JMP    .+11                 ;START OF ERROR ROUTINE
        LDA    0,CP
        JSR    @A.PTC
        LDA    0,CST1
        JSR    @A.PTC
        LDA    1,.+3
        JSR    @A.BINO
        JMP    @A.LIJ
```

```
        3333                            ;END OF ERROR ROUTINE
        LDA     2,@21
        STA     0,@21
        INTEN
        JMP     @0
I:        0
        .RDX   10
NSAMP:    -200
        .RDX     8
FREQ:   000001           ;FREQUENCY CODE OF RTC
B12RT:    000
B12CN:    LDA     3,A.STR
          STA     3,0                   ;LOAD STARTING ADDRESS IN .LOC 0
        LDA     2,NSAMP
        STA     2,I
        LDA     2,C
        FINI
        FETR                 ;ENTER F.P. INTERPRETER
L35:    FIC2                 ;INCREASE ADDRESS BY 2
        FFLO     0,2         ;FLOAT
        FISZ     I
        FJMP     L35
        FEXT
        JMP     @B12RT
C:        .
          .
        .BLK     620        ;RESERVE 400 LOCATIONS FOR C
        JMP     .+1
;===================================================================
B13:         STA     3,CRRET            ;SAVE RETURN ADDRESS
             FINI                       ;INITIALIZE INTERPRETER
             FETR                       ;ENTER INTERPRETER
             FSUB    0,0              ;PROGRAM INITIALIZATION  - BOX 2
             FSTA    0,XX1
             FSTA    0,XX2
             FLD3    @CADD+1
             FST3    TMP1+1            ;ADDRESS OF CI STORED IN TMP+1
             FLDA    0,CRM10            ;                     - BOX 3
             FLDA    1,CRONE
C[illegible]B3L:     FADD    1,0,FSLE
             FJMP    L1
             FLDA    2,XX1
             FISZ    TMP1+1
             FISZ    TMP1+1
             FLDA    3,@TMP1+1
             FJMP    .+1
             FJMP    .+1
             FADD    3,2
```

```
        FSTA    2,XX1
        FLDA    2,XX2
        FMPY    3,3
        FADD    3,2
        FSTA    2,XX2
        FJMP    CRB3L
L1:     FLDA    0,XX1           ;PREPARE FOR BOX 5
        FSTA    0,FMSTD+1
        FLDA    0,XX2
        FSTA    0,FMSTD+3
        FLDA    0,CRM10
        FPOS    0,1
        FSTA    1,FMSTD+5
        FJSR    FMSTD               ;BOX 5
        FJMP    B6
; FLOATING SUBROUTINE FMSTD---------------------------------
; THIS SUBROUTINE ASSUMES THAT THE PARAMETER LIST IS STORED
; IMMEDIATELY AFTER ITS FIRST LINE OF CODE THUS:
;   FMSTD:      FJMP  ETC.
;     SX1   -   FL
;
;     SX2   -   FL
;
;     N     -   FL
;
;     AVG   -   FL
;
;     STD   -   FL
;
FMSTD:   FJMP   .+13
         .BLK   12          ;RESERVE 10 LOCATIONS
         FLDA   3,FMSTD+1
         FMOV   3,2
         FLDA   1,FMSTD+5
         FDIV   1,3
         FSTA   3,FMSTD+7
         FMPY   2,3
         FLDA   2,FMSTD+3
         FSUB   3,2
         FLDA   3,.+6
         FSUB   3,1
         FDIV   1,2
         FSQR   2,2
         FSTA   2,FMSTD+11
         FJMP   0,3
         040420             ;FLOATING POINT 1
         0
; END OF SUBROUTINE FMSTD-----------------------------------
```

```
J6:      FLDA    0,CRM10        ;I IS STORED IN FACO  -  BOX 6
         FFCS    0,0
         FLDA    1,CRFNE
B6A:     FADD    1,0
         FLDA    2,NP
         FSUBW.   0,2,FSGE
         FJMP ERROR             ;ERROR EXIT FROM BOX 6
         FISZ    TMP1+1
         FISZ    TMP1+1         ;TMP1+1 CONTAINS ADD(C(I))
         FLDA    2,@TMP1+1      ;BOX 7
         FLDA    3,FMSTD+7      ;FAC3 CONTAINS AVG
         FSUB    3,2
         FPCS    2,2
         FLDA    3,FMSTD+11     ;FAC3 CONTAINS STD
         FADD    1,3
         FADD    3,3
         FADD    3,3
         FSUB    3,2,FSLT
         FJMP    L10
         FJMP    B6A
ERROR:   FEXT.
         LDA     0,CST1
         JSR     @A.PTC
         LDA     1,.+3
         JSR     @A.BINO
         JMP     @A.L11
         4444
CADD:    0                  ;ADDRESS OF FIRST ELEMENT OF C
         C
CRRET:   0                    ;SFX  -  RETURN ADDRESS
XX1:     0                    ;FL
         0
XX2:     0              ;FL
         0
CRM10:   140640                     ;FL  -  FLOATING POINT -10
         0
TMP1:    0
         0
I2:      0
         0
I1:      0
        0
BL:      0
         0
L10:     FSTA     0,I2
         FSUB    1,0          ;J IN FACO  -  BOX 10
         FDSZ     TMP1+1
         FDSZ     TMP1+1          ;TMP1+1 CONTAINS ADD(CJ)
```

```
B11:     FLDA   2,@TMP1+1   ;       BOX 11
         FLDA   3,FMSTD+7
         FSUB   3,2
         FPCS   2,2
         FLDA   3,FMSTD+11
         FSUB   3,2,FSGT
         FJMP   L13         ;TRUE EXIT FROM BOX 11
         FSUB   1,0         ;      BOX 12
         FDSZ   TMP1+1
         FDSZ   TMP1+1
         FJMP   B11
L13:     FSTA   0,I1        ;       BOX 13
         FLDA   2,FMSTD+7
         FSTA   2,BL
         FLD3   TMP1+1
         FST3   TMP2
         FSTA   2,CMIN
L13A:    FLDA   3,NP       ;BOX 13A  -  NP IN FAC3
         FSUB#  3,0,FSLE
         FJMP   L13D
         FLDA   3,@TMP1+1      ;TMP1+1 CONTAINS ADD(C(I)) - BOX 13B
         FSUB#  2,3,FSLE
         FJMP   INC13
         FMOV   3,2        ;BOX 13C
         FSTA   2,CMIN
         FSTA   0,I3
INC13:   FADD   1,0        ;BOX 13A
         FISZ   TMP1+1
         FISZ   TMP1+1
         FJMP   L13A
L13D:    FLDA   0,BL       ;BOX 13D
         FSUB   2,0
         FLDA   3,DT              ;DT = 0.1
         FMPY   0,3
         FLDA   0,BL
         FSUB   3,0
         FSTA   0,CRIT
         FLD3   CADD+1     ;BOX 13E
         FST3   TMP1+1
         FFLO   TMP1
         FLDA   0,TMP1
         FLDA   3,I3
         FADD   3,0
         FADD   3,0
         FSTA   0,TMP1
         FFIX   TMP1
         FLDA   2,CRIT
L13E:    FLDA   0,NP
```

```
        FSUB#   0,3,FSLE
        FJMP    L13H
        FLDA    0,@TMP1+1       ;BOX 13F
        FSUB#   2,0,FSLT
        FJMP    L13G
        FADD    1,3
        FISZ    TMP1+1
        FISZ    TMP1+1
        FJMP    L13E
L13G:   FSTA    3,I4       ;BOX 13G
        FJMP    L14
CMIN:   0
        0
I3:     0
        0
CRIT:   0
        0
I4:     0
        0
TMP2:   0
        0
L13H:   FLDA    3,NP       ;BOX 13H
        FSTA    3,I4
L14:    FLDS    TMP2
        FSTS    TMP1+1
        FSUB    3,3        ;       BOX 14
        FSTA    3,SX1
        FSTA    3,SX2
        FSTA    3,X12
        FSTA    3,SX1S
        FSTA    3,SX2S
        FSTA    3,SX12
        FSTA    3,SX1Y
        FSTA    3,SX2Y
        FSTA    3,SY
        FSTA    3,SW
        FLDA    0,I1           ;       BOX 15
L15:    FADD    1,0
        FSTA    0,TMP2
        FISZ    TMP1+1
        FISZ    TMP1+1
        FLDA    2,I4
        FSUB#   2,0,FSLE
        FJMP    L18             ;TRUE EXIT FROM BOX 15
        FLDA    0,X12      ;BOX 16
        FLDA    1,DT
        FADD    1,0
        FSTA    0,X12
```

```
FALG    0,1
FLDA    2,@TMP1+1
FLDA    3,BL
FSUB    3,2
FPCS    2,2
FJMP    .+1
FJMP    .+1
FALG    2,2
FMOV    2,3
FMPY    3,3
FSTA    2,YI
FMOV    3,2        ;      BOX 17
FMPY    1,2
FLDA    0,SX1
FADD    2,0
FSTA    0,SX1     ;CALC SX1
FMPY    2,1
FLDA    0,SX1S
FADD    1,0
FSTA    0,SX1S     ;CALC SX1S
FLDA    1,XI2
FMPY    2,1
FLDA    0,SX12
FADD    1,0
FSTA    0,SX12     ;CALC SX12
FLDA    1,YI
FMPY    2,1
FLDA    0,SX1Y
FADD    1,0
FSTA    0,SX1Y     ;CALC SX1Y
FLDA    2,XI2
FMOV    2,1
FMPY    3,2
FLDA    0,SX2
FADD    2,0
FSTA    0,SX2      ;CALC SX2
FMPY    2,1
FLDA    0,SX2S
FADD    1,0
FSTA    0,SX2S      ;CALC SX2S
FLDA    1,YI
FMPY    1,2
FLDA    0,SX2Y
FADD    2,0
FSTA    0,SX2Y      ;CALC SX2Y
FMPY    3,1
FLDA    0,SY
FADD    1,0
FSTA    0,SY       ;CALC SY
```

```
        FLCA    0.SW
        FADD    3.0
        FSTA    0.SW          ;CALC SW
        FLCA    0.TMF2
        FLDA    1.CRFNE
        FJMP    L15
YI:     0
        0
DT:     040031                   ;FL  0.1 SEC = TIME BETWEEN SAMPLES
        114631
SX1:    0
        0
SX2:    0
          0
SX1S:   0
        0
SX12:   0
        0
SX1Y:   0
        0
SX2S:   0
        0
SX2Y:   0
        0
SY:     0
        0
SW:     0
        0
X1S:    0
        0
X12:    0
        0
X1Y:    0
        0
X2S:    0
        0
X2Y:    0
        0
A1:     0
        0
GAM:    0
        0
D:      0
        0
ALPHA:  0
        0
B:      0
        0
```

```
XK:        0
           0
XI2:       0
           0
           040420    ;FL - FLOATING POINT 1
           0000
CRRT1:     CRRET
L18:       FLDA    3.SW       :      BOX 18
           FLDA    1.SX1
           FMOV    1.2
           FMPY    1.2
           FDIV    3.2
           FLDA    0.SX1S
           FSUB    2.0
           FSTA    0.X1S          ;CALC X1S
           FLDA    2.SX2
           FMPY    1.2
           FDIV    3.2
           FLDA    0.SX12
           FSUB    2.0
           FSTA    0.X12          ;CALC X12
           FLDA    2.SY
           FMPY    1.2
           FDIV    3.2
           FLDA    0.SX1Y
           FSUB    2.0
           FSTA    0.X1Y          ;CALC X1Y
           FLDA    1.SX2
           FMOV    1.2
           FMPY    2.2
           FDIV    3.2
           FLDA    0.SX2S
           FSUB    2.0
           FSTA    0.X2S          ;CALC X2S
           FLDA    2.SY
           FMPY    1.2
           FDIV    3.2
           FLDA    0.SX2Y
           FSUB    2.0
           FSTA    0.X2Y          ;CALC X2Y
           FLDA    0.X1S
           FLDA    1.X2S
           FLDA    2.X12
           FMPY    2.2
           FMPY    0.1
           FSUB    2.1
           FSTA    1.D            ;CALC D
           FLDA    0.X2S
```

```
FLDA    2,X1Y
FMPY    0,2
FLDA    0,X12
FLDA    3,X2Y
FMPY    0,3
FSUB    3,2
FDIV    1,2
FSTA    2,ALPHA     ;CALC ALPHA
FLDA    2,X1Y
FMPY    0,2
FLDA    0,X1S
FLDA    3,X2Y
FMPY    0,3
FSUB    2,3
FDIV    3,1
FFCS    1,1
FSTA    1,B         ;CALC B
FLDA    0,SX2
FDIV    1,0
FLDA    1,ALPHA
FLDA    2,SX1
FMPY    2,1
FLDA    2,SY
FSUB    1,2
FADD    0,2
FLDA    3,SW
FDIV    3,2
FEXP    2,2
FSTA    2,XK
FLDA    0,ALPHA     ;BOX 19
FLDA    1,X12+2      ;LOAD 1.0 IN FAC1
FADD    1,0
FSTA    0,A1
FJSR    @A,GAM      ;BOX 20
FSTA    0,GAM
FLDA    2,B         ;BOX 21
FLDA    1,A1
FALG    2,2
FMPY    2,1
FEXP    1,1
FLDA    2,XK
FMPY    0,2
FMPY    1,2
FSTA    2,AREA
FEXT
LDA     3,CRFT1
LDA     2,@7
MOV     2,2,SNR
```

```
          JMP    @0,3
          LDA    0,CST1
          JSR    @A.FTC
          LDA    1,.+5
          JSR    @A.BINO
          SUBO   0,0
          STA    0,@7
          JMP    @A.L11
          5555              ;MEANS OVERFLOW OCCURRED IN THIS SUBROUTINE
;==========================================================================
B14:      STA    3,B14RT
          FETR
          FLDA   0,DTA      ;COMPUTE CARDIAC OUTPUT
          FLDA   1,DTA+2
          FLDA   2,DTA+4
          FLDA   3,DTA+6
          FMPY   1,0
          FMPY   2,0
          FMPY   3,0
          FLDA   2,VB
          FLDA   3,VI
          FSUB   3,2
          FMPY   2,0
          FLDA   1,AREA
          FDIV   1,0
          FLDA   3,GRAT
          FMPY   3,0
          FLDA   3,DTA+10
          FMPY   3,0
          FSTA   0,CO
          FFDCF  0
          FEXT
          LDA    3,B14RT
          LDA    2,@7
          MOV    2,2,SNR
          JMP    0,3
          LDA    0,CST1
          JSR    @A.PTC
          LDA    1,.+5
          JSR    @A.BINO
          SUBO   0,0
          STA    0,@7
          JMP    @A.L11
          6666              ;MEANS OVERFOLW OCCURRED IN THIS SUBROUTINE
B14RT:    000
;==========================================================================
;B16 IS THE MODULE THAT COMPUTES THE CARDIAC INDEX. FACO CONTAINS
;THE RESULT AND IT IS STORED IN CI. THE ROUTINE ALSO PRINTS THE RESULT ON THE
```

```
;TTO. IT IS NECESSARY TO USE THE F.P.I. WITH THIS ROUTINE.
B16:   STA    3,B16RT            ;STORE RETURN ADDRESS
        FINI                     ;INITIALIZE INTERPRETER
        FETR                     ;ENTER INTERPRETER
        FLDA   0,CO
        FLDA   1,BSA
        FDIV   1,0
        FSTA   0,CI
        FFDCF  0
        FEXT                     ;EXIT FROM INTERPRETER
        JMF    @B16RT            ;RETURN
B16RT:  000
;=====================================================================
 GAMMA:    FST3  TEMP; SAVE RETURN ADDR.
           FLD3  ADC7; ADDR. OF HIGH ORDER GY
 ; CCEF. IN HARCWARE AC3
           FLDA 1,MAXA; MAX ARG FOR GF,
         FSUB# 1,0,FSLE
           FJMP OVFL ; GO TO OVFL IF ARG>MAXA
        FSTA 0,X ; SAVE GF ARG IN X
        FLDA 1,ZERO
        FSTA 1,IER ; ICR=0.0
        FLDA 1,ONE
        FSTA 1,GX   ;GX=1.0
        FLDA 1,TWO
        FSUB# 1,0,FSLE ; SKIP IF X LESS THAN
                   ;  CP EQUAL 2.0
       FJMP GT2 ; JMF IF X>2.
        FLDA 1,ONE
        FSUE# 1,0,FSLF; SKIP IF X LESS THAN
 ; OR ECUAL 1.0
        FJMP SI10; X>1.C
        FSUE# 1,0,FSZR; SKIP IF X=1.0
        FJMF LT1  ; X<1.0
        FLDA 0,GX
           FJMP @TEMP;X=1,RETURN. CORR.TO
 ; STMT 120 OF FORT. FRCG.
LT1:   FLDA 1,ERR
        FSUE# 1,0,FSLE; SKIP IF X< OR = ERR.
        FJMP GTERR ;X>ERR. STMT 80
       FSTA 0,K
        FFIX K
        FFLC K
        FLDA 1,K
        FSUB 0,1  ;FLOAT(K)-X IN FAC1
        FSTA 1,Y ; STCRE IN Y
        FPOS 1,2: ABS(Y) IN FAC2
        FLDA 3,ERR
```

```
        FSUB# 3,2,FSLE; SKIP IF ABS(Y) < OR
; =ERR
        FJMP NEXT; ABS(Y)>ERR. STMT 64
BACK:   FLDA 1,ONE
       FSTA 1,IER; IER=1.0. STMT 130
        FLDA 0,GX
           FJMP @TEMP; RETURN
NEXT:    FLDA 2,ONE
         FSUB 1,2; 1.0-Y IN FAC2
                 ;ERR STILL IN FAC3
          FSUB# 3,2,FSLE ; SKIP IF 1.0-Y
; LESS THAN OR EQ. ERR
          FJMP S70 ; 1-Y>ERR. STMT 70
        FJMP BACK; STMT 130.IER=1.0
S70:      FLDA 1,ONE
S70A:    FSUB# 1,0,FSLE;SKIP IF X<OR EQ.1.0
        FJMP S110 ;X>1.0. JMP TO S110
GTERR:   FLDA 1,ONE
       FLDA 2,GX;STMT 80
         FDIV 0,2 ; GX/X IN FAC2
         FSTA 2,GX; GX=GX/X
         FADD 1,0 ;X=X+1.0
         FJMP S70A; JUMP TO TEST IF X<OR EQ 1.0
OVFL:    FLDA 1,TWO
        FSTA 1,IER   ;IER=2.
        FLDA 0,LARGE;10**38 IN FAC0
        FSTA 0,GX   ;GX=10**38
           FJMP @TEMP; RETURN WITH GX IN FAC0
GT2:    FLDA 1,ONE
LOOP:   FSUB 1,0  ;X-1.0 IN FAC0
        FSTA 0,X  ;X=X-1.0
        FLDA 2,GX
        FMPY 0,2  ;GX*X IN FAC2
        FSTA 2,GX ;GX=GX*X
        FLDA 3,TWO
        FSUB# 3,0,FSLE; SKIP IF X<OR EQ 2.0
        FJMP LOOP ;X>2,JMP BACK AND DECR X
S110:    FLDA 1,ONE
        FSUB 1,0  ;X-1.0 IN FAC0
;POLYNOMIAL EVALUATION BEGINS. POLY VALUE
; WILL BE IN FAC0
;
        FMOV 0,2 ;Y=X-1.0 IN FAC2
        FLDA 0,ZERO ;CLEAR FAC0
FLPY1:     FLDA 1,0,3; GET COEF(HIGH ORDER FIRST)
        FMOV 1,1,FSNR; SKIP IF COEF NON ZERO
        FJMP POLY    ;COEF.0,END OF TABLE
        FMPY 2,0     ;SUM*Y IN FAC0
```

```
          FADD 1.0     ;SUM*Y+COEF
             FIC3   ; BUMP POINTER TO NEXT COEF.
        FJMP FLPY1
FCLY:     FMPY 2.0  ; MULT BY Y AGAIN
          FLDA 1,ONE
          FADD 1.0     ;GY IN FAC0
          FLDA 1,GX
          FMPY 1.0     ;GY*GX IN FAC0
             FJMP @TEMP; RETURN WITH GX IN FAC0
MAXA:      041042
           000000; MAX ARG FOR GF(34.0)
TWO :      040440
           000000 ; 2.0
LARGE:     060113
           035514 ; 10**38
ERR:       036020
           143367 ; 10**-6
ZERO:      000000
           000000 ; 0.0
ONE:       040420
           000000 ; 1.0
GC7:       137722
           170355 ; -.0514993
           040101
           035752 ; .2548205
           140221
           103560 ; -.5664729
           040325
           031704 ; .8328212
           140340
            056455; -.8764218
           040374
           060355
           140223
           136357      ;-5771017
           000000
           000000      ;DUMMY VALUE TO INDICATE END OF TABLE
IER:       0000
           000000; FOR ERROR CODE.
GX:        000000
           000000 ; GF VALUE
X:         000000
           000000 ; FOR SAVING GF ARG
K:         000000
           000000 ; FOR FIXED PT X
Y:         000000
           000000 ; FOR STORING Y
ADC7:       GC7; ADDR. OF HIGH ORDER GY COEF.
TEMP:       000000; FOR SAVING RETURN ADDR
```

```
     FKI = 50
FKIN: STA   3,FKRET                 ;GETC
     NIOS   FKI
     SKPDN  FKI
     JMP    .-1
     DIAC   0,FKI
     JSR    @A.PTC
     JMP    @FKRET
     FKC = 51
FKCT: SKPBZ FKO                     ;PUTC
     JMP    .-1
     DOA    0,FKO
     MOV    0,0,SZR
     NIOS   FKO
     JMP    0,3
FKRET: 000
;$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
;PROGRAM TO PRINT THE VALUE OF FLOATING POINT NUMBERS
        000000
         JSR    @A.OBIN
        STA    1,.-2
        FINI
         FETR
        FLDA   0,@.-5
         FFDC   0
        FEXT
        JMP    .-7
;========================================================================
;CONVERT AN ASCII OCTAL STRING TO A BINARY NUMBER
;INPUT: CALLS A GET CHARACTER ROUTINE WHOSE ADDRESS MUST BE STORED
;IN LOCATION A0 OF PAGE 0.
;CHARACTERS MUST BE RETURNED, RIGHT ADJUSTED IN AC0 WITH BIT 8 = 0
;INPUT OF FORM:      OO...O(BREAK)     WHERE 'O' REPRESENTS AN OCTAL
;DIGIT AND BREAK IS ANY OTHER CHARACTER
;OUTPUT: AC0 CONTAINS  BREAK CHARACTER
;AC1 CONTAINS THE BINARY NUMBER (MOD 200000 OCTAL)
;
;CALLING SEQUENCE:
;    JSR    .OBIN
;      RETURN
;
;IF AN INDICATION IS DESIRED TO SIGNAL CHARACTERS ARE REQUESTED
;CALLING SEQUENCE :
;   JSR    .CONI
;     RETURN
;
;AN ASCII 'C' FOLLOWED BY A NULL CHARACTER WILL BE TRANSMITTED VIA AC0
;TO USER PUT CHARACTER ROUTINE WHOSE ADDRESS MUST BE STORED IN
```

```
;LOCATION 41 OF PAGE 0
;CAUTION: RESULT IS N MOD 200000 (OCTAL)
; E.G.  576452* CONVERTS TO 176452
;
; DESTROYED: AC0,AC1,AC3,CARRY
; UNCHANGED: AC2
;
.CBNI:  STA    3,.EE03          ;SAVE RETURN
        STA    2,.EE02          ;SAVE AC2
        LDA    0,.EE22
        JSR    @.EE41           ;SEND '0'
        SUB    0,0
        JSR    @.FE41           ;SEND NULL
        JMP    .+3
.OBIN:  STA    3,.EE03          ;SAVE RETURN
        STA    2,.EE02          ;SAVE AC2
        SUB    1,1              ;CLEAR RESULT WORD
        STA    1,.EE10
.EE98:  JSR    @.EE40           ;GET A DIGIT
        LDA    2,.EE20          ;OCTAL 60
        LDA    3,.EE21          ;OCTAL 67
        ADCZ#  3,0,SNC          ;TEST FOR 60 <=N<=67
        ADCZ#  0,2,SZC
        JMP    .EE99            ;NO - MUST BE BREAK CHARACTER
        SUB    2,0              ;PUT N IN RANGE 0-7
        LDA    1,.EE10
        MOVZL  1,1              ;SHIFT SUM
        MOVZL  1,1
        MOVZL  1,1
        ADC    0,1
        STA    1,.EE10
        JMP    .EE98            ;LOOP TILL BREAK RECEIVED
.EE99:  LDA    2,.EE02          ;RESTORE AC2
        LDA    1,.EE10          ;ANSWER TO AC1
        JMP    @.EE03           ;AND RETURN
.EE02:  0                       ;SAVE AC2
.EE03:  0                       ;SAVE RETURN
.EE10:  0                       ;STORAGE FOR RESULTS
.EE20:  60                      ;ASCII '0'
.EE21:  67                      ;ASCII '7'
.EE22:  "0                      ;ASCII '0'
        .EE40=40                ;PAGE 0 ADDRESS OF GET A CHARRACTER ROUTINE
        .EE41=41                ;PAGE 0 ADDRESS OF PUT CHARACTER ROUTINE
;=========================================================================
;BINARY TO OCTAL ACSII CONVERT
;CONVERTS A 16-BIT BINARY WORD TO AN OCTAL ASCII CHARACTER STRING
;
;INPUTS:  N IN AC1
```

```
;
;OUTPUT:  ASCII CHARACTER STRING, TERMINATED BY A NULL CHARACTER
;CHARACTERS PASSED RIGHT ADJUSTED IN AC0 TO THE USER ROUTINE
;WHOSE ADDRESS MUST BE STORED IN LOCATION 41 OF PAGE 0
;
;STRING OF FORM:     OOOOOO(NULL)
;WHERE "O'S" REPRESENT OCTAL DIGITS
;
; CALLING SEQUENCE:
;    JSR    .EINO
;    RETURN
;
;DESTROYED:AC0, AC1, AC3, CARRY
;UNCHANGED: AC2
;
.EINC:  STA    3,.EF03          ;SAVE RETURN
        STA    2,.EF02          ; *SAVE AC2
        SUBZR  2,2,SKP          ;100000 TO AC2
.EF99:  SUB    2,1,SKP          ;DECREASE CURRENT DIGIT BY 1
.EF98:  LDA    0,.EF20          ;GET OCTAL 57
        INC    0,0              ;FORM ASCII OUTPUT DIGIT
        SUBZL# 2,1,SNC          ; IMPLIES DIGIT COMPLETE
        JMP    .EF99            ;NOT DONE, SUBTRACT 1 FROM CURRENT DIGIT
        STA    2,.EF1C          ;SAVE SUBTRACT CONSTANT
        JSR    @.EF40           ;PUT OUT A DIGIT
        LDA    2,.EF1C          ;RESTORE SUBTRACT CONSTANT
        MOVZR  2,2              ;POSITION "1" FOR NEXT OCTAL DIGIT
        MOVZR    2,2
        MOVZR    2,2,SZR
        JMP    .EF98
        MOV    2,0
        JSR    @.EF40
        LDA    2,.EF02          ; *RESORE AC2
        JMP    @.EF03           ;RETURN
.EF02:  0                       ; *SAVE AC2
.EF03:  0                       ;SAVE RETURN
.EF1C:  0                       ;SAVE LOCATION FOR SUBTRACT CONSTANT
.EF20:  57                      ;ASCII CONSTANT
       .EF40=41                 ;PAGE 0 ADDRESS OF PUT CHARACTER ADDRESS
;=====================================================================
;PROGRAM TO OUTPUT 200 SAMPLES OF DATA COLLECTED
      LDA    1,PT200+4
      STA    1,PT200
      LDA    2,PT200+3
      STA    2,PT200+5
      LDA    2,PT200+6    ;OUTPUT TO TTY
      STA    2,40
      LDA    2,PT200+7
```

```
        STA    2,41
RETN:   FETR
        FLDA   0,@PT200
        FSTA   0,PT200+1
        FFIX   PT200+1
        FEXT
        LDA    1,PT200+2
        JSR    @A.BINO
        ISZ    PT200
        ISZ    PT200
        DSZ    PT200+5
        JMP    RETN
        LDA    2,PT200+10     ;OUTPUT TO FUNCTIONAL KEYBOARD
        STA    2,40
        LDA    2,PT200+11
        STA    2,41
        HALT
PT200:   000                  ;ADDRESS OF NUMBER TO BE PRINTED
         000                  ;VALUE OF F.P. NUMBER TO BE PRINTED
         000
         310                  ;310 = 200 (DECIMAL)
         C+2
        000
         TTYIN
         TTYCT
         FKIN
         FKCT
          .END
        .END
```

NCVA-1200 SYSTEM SUPPORT PROGRAMS

I NCVA-1200 SYSTEM INITIALIZATION

```
        377
        .LOC    7
        3730
        .LOC 40
A.GTC:  FKIN
A.PTC:  FKOT
        .LOC    50
ESA:    000                             ;BODY SURFACE AREA
        000
DTA:    040421    ;FL - PCP(5% DEXTROSE)/PCP(BLOOD)
        043656    ; = 1.08
        040323    ;FL - CCRRECTION FOR INJECTATE TEMPERATURE RISE
        031463    ; = C.825
        041074    ;FL - SECONDS/MINUTE
        000000    ; = 60
        C40640    ;FL - VOLUME OF INJECTATE
        000000    ; = 1C ML
        C37101    ;FL - LITRES/ML
        104467    ; =0.CC1
AREA:   0
        0
NP:     041310    ;NUMBER OF SAMPLES = 200
        000000
CO:     C00       ;CARDIAC OUTPUT
        000
CI:     000       ;CARDIAC INDEX
      0C0
VOUT:   C00
        000
VI:     000
        000
VB:     000
        000
GRAT:   C40664    ;GAIN RATIO = 11.3
        146315
SN:     040031                  ;SMALL NUMBER = 0.1
        114631
CRFNE:    040420               ;FL  -  FLOATING POINT 1
          0
```

```
MASK:  177773
A.STR: START-1                ;STARTING ADDRESS - 1
A.RST: RST
A.B1:  B1
A.B4:  B4
A.B10: B10
A.B12: B12
A.B13: B13
A.B14: B14
A.B16: B16
A.GAM: GAMMA
A.CBIN: .CBIN
A.BINO: .BINO
A.L11: L11
; FUNCTIONAL KEYBOARD (RT01) DISPLAY CONTROL CHARACTERS
C0:     "0                    ;LOAD AND DISPLAY 0
C1:     "1                    ;LOAD AND DISPLAY 1
C2:     "2                    ;LOAD AND DISPLAY 2
C3:     "3                    ;LOAD AND DISPLAY 3
C4:     "4                    ;LOAD AND DISPLAY 4
C5:     "5                    ;LOAD AND DISPLAY 5
C6:     "6                    ;LOAD AND DISPLAY 6
C7:     "7                    ;LOAD AND DISPLAY 7
C8:     "8                    ;LOAD AND DISPLAY 8
C9:     "9                    ;LOAD AND DISPLAY 9
CBL:    ":                    ;LOAD AND DISPLAY A BLANK
CCL:    "@                    ;CLEAR DISPLAY
CCP:    ".                    ;ENTER AND DISPLAY A DECIMAL POINT
; FUNCTIONAL KEYBOARD (RT01) STATUS INDICATOR CONTROL CHARACTERS
CP:     "P                    ;CLEAR STATUS INDICATORS
CST1:   "Q                    ;SET STATUS INDICATOR 1
CST2:   "R                    ;SET STATUS INDICATOR 2
CST3:   "T                    ;SET STATUS INDICATOR 3
CST4:   "X                    ;SET STATUS INDICATOR 4
CST34:   134                  ;SET STATUS INDICATORS 3 AND 4
; FUNCTIONAL KEYBOARD (RT01) TRANSMIT CHARACTERS
CA:     "A
CB:     "B
CC:     "C
CD:     "D
CE:     "E
CF:     "F
        CPU=77
```

DEBUG PROGRAM

```
.LOC  3260
 CPU=77
 A:DSZ BPSWT;START HERE
 STA  0,ACS;SAVE ACS
 STA 1,ACS+1
 STA 2,ACS+2
 STA 3,ACS+3
 LDA 0,INST;SETUP BREAK INSTRUCTION
 STA 0,PINST
 LDA 0,BADDR;BREAK MAY BE CHANGED.
 STA 0,PADDR
 SUBCR 2,2;SAVE CARRY
 SKPBZ CPU;REMEMBER STATE OF
 INC 2,2;INTERRUPT FLOP
 C77:NIOC CPU;TURN OF INTERRUPT
 STA 2,ACS+4;CARRY SAVE
 SKPBZ TTO;REMEMBER THE STATE OF
 JMP .-1;TTO DONE FLAG.
 SKPDZ TTO
 ADC 2,2
 STA 2,TFLAG
 LDA 1,BADDR;IF ENTERED VIA
 ISZ BPSWT;BREAK POINT TYPE THE
 AAA:JSR POCT;BREAK POINT ADDRESS.
 LDA 0,C15
 JSR TYPE;ECHO CARRIAGE RETURN.
 AA:LDA 0,C12;TYPE LINE FEED
 JSR TYPE
 STA 0,OPEN;REGISTER OPEN/CLOSE
 SUB 2,2
 STA 2,BPSWT;BREAK POINT SWITCH.
 STA 2,DIN;DATA TYPED SWITCH
 SKPDN TTI
 JMP .-1
 DIAC 0,TTI;INPUT A CHAR
 BB:JSR TYPE;ECHO
 LDA 3,C177
 AND 3,0;MASK PARITY
 LDA 3,C70;CHECK FOR A DIGIT
```

```
ADCZ# 0,3,SZC
LDA 3,C60
ADDO 0,3,SEZ
JSP SERH;NOT A DIGIT 0-7
CC:ISZ DIN;DIGIT SWITCH
MOVZL 2,2;ASSEMBLE DIGITS
ADDZL 2,2
ADD 3,2,SNC;SKIP IF EXTRA DIGIT.
JMP BB-3
GOOF:LDA 0,C77;OPERATOR ERROR
JMP AA-1
SERH:LDA 1,X,3;SPECIAL CHARACTOR SEARCH
MOVL# 1,1,SZC
JMP GOOF;OPERATOR GOOF
C40:INCO 3,3
SUBZ 0,1,SZR;CHECK CHAR WITH TABLE
JMP SERH;NOT YET FOUND
JSR @7+X,3;A FIND
ACS: 0;AC STORAGE AREA
0
0
0
0
BADDR:INST
INST:JMP GOOF
BINST:0
LINE:MOVZ 0,0;LINE FEED TYPED
CARR:LDA 0,C15;CARRIAGE TYPED
JSR TYPE;TIME DELAY FOR TTY.
JSR TYPE
LDA 3,EXAM
LDA 1,OPEN
LDA 0,DIN
ADDZ# 0,1,SZC;TEST FOR OPEN REGISTER DATA
STA 2,0,3
INC 3,1,SZC
JMP AA;CARRIAGE EXIT
STA 1,EXAM;LINE FEED CONT
JSR POCT
LDA 3,EXAM
ATYP:ADD 3,2;LETTER "A" TYPED
SLASH:STA 2,EXAM;SLASH TYPED
LDA 0,BADDR;IF OPENED REGISTER
LDA 1,INST;IS LAST BREAK POINT
SUB# 0,2,SNR;DON'T OPEN PUT
JMP AAA;PRINT CORRECT INST.
LDA 1,0,2
JSR POCT
```

```
ADC 0,0;OPEN A REGISTER
JMP A/+2
POCT:STA 3,DIN;OCTAL PRINT C(1)
LDA 0,C40
JSR TYPE;PRINT SPACE.
SUBZR 2,2,SKP
SUB 2,1,SKP
LDA 0,C57
C1[illegible]:INC 0,0
SUBZ# 2,1,SZC
JMP POCT+4
JSR TYPE
MOVZR 2,2
MOVZR 2,2
MOVZR 2,2,SZR
JMP POCT+5;ANOTHER DIGIT
LDA 0,C40;EXIT WITH SPACE
JSR TYPE
JMP @DIN
TYPE:DOAS 0,TTO;SEND CHAR TO TYPE
SKPBZ TTO
C777:JMP .-1
JMP 0,3
CA:A+1
PBSWT:0
CJMP4:JMP @10
C70:70
BADDR:0
BIN:0
TFLAG:0
OPEN:0
C12:12;LINE FEED
C15:15;CARRIAGE
101;A
102;BREAK
120;PROCEED
C57:57;SLASH
122;RUN
CM60:-60;TERMINATOR
LINE
CARR
ATYP
B
PRC
SLASH
R
B:LDA 0,INST;BREAK LOGIC
STA 0,@BADDR;RESTORE PREVIOUS BREAK
      0
```

```
LDA 0,0,2
STA 0,INST;INST AT BREAK POINT
STA 2,BADDR
LDA 0,CJMP4
STA 0,@BADDR
JMP AA-2
X=C12+CC
LDA 2,BADDR
INCZL 1,1,SKP;C(1)=+2
R:LDA 0,CEX;GO COMMAND
ADC 1,2,SNR
JMP GOBF;RUN NEEDS A ADDRESS
STA 2,DIN
STA 0,EXAM
LDA 0,CA
STA 0,10;SET BREAK POINT RETURN
LDA 0,ACS;RESTORE AC 0-3
LDA 1,ACS+1
LDA 2,ACS+2
LDA 3,ACS+4
ISZ TFLAG
NIOC TTO
MOVOL 3,3,SKP
RELADR:0;ADDRESS FOR PC RELATIVE.
STA 3,ACS+4
LDA 3,ACS+3;RESTORE AC-3
DSZ ACS+4;PERHAPS TURN ON
C177:NIOS CPU;INTERRUPT SYSTEM
EXAM:0;JMP IF G,INST IF P
DSZ DIN;-1 ADDRESS IF NO SKIP
EXIT:JMP @DIN
CEX:DIN-EXAM&2777;A JMP @DIN INSTRUCTION
PRC:LDA 0,PINST;PROCEED FROM BREAK
LDA 2,C1400;CHECK IF INST IS
LDA 3,C777;;RELATIVE TO PC
ANDZL 3,2,SNC
ADC 3,2,SZR
JMP R-2;NOT PC RELATIVE
PRC1:MOVZR 3,1;DISPLACEMENT SIGN
AND 1,0;EXTENSION.
INCZR 1,1;C(1)=200,=BIT B(1)
ANDZL 0,1,SZR;C(3)=777,C(1)=400
SUB 1,0;C(0)=+,- DISPLACEMENT
LDA 2,BADDR;PROCEED ADDRESS
ADDL 0,2
PRC2:LDA 0,PINST;MAKE PROCEED INST
COM 3,1;INDIRECT AND RELATIVE
INCZL 3,3;TO CALCULATED ADDRESS.
```

```
ANDO 1,[illegible]
ANC# 0,3,SNF
ADDZ 3,0;INSERT @ EIT IN INST.
MCVR 2,2;IF @ INSERT @ IN
STA 2,RELACR;THE EFFECTIVE ADDRESS
LCA 2,CM5
ADC 2,0;THE ACDRESS PART TO INST.
SUF 1,1
JMF R-2
CMS:773
A
JMF @.-1;ANCTHER PLACE TC START.
```

```
ANDO 1,0
AND# 0,3,SNR
ADDZ 3,0;INSERT @ BIT IN INST
MOVR 2,2;IF @ INSERT @ IN
STA 2,RELADR;THE EFFECTIVE ADDRESS.
LDA 2,CM5
ADD 2,0;THE ADDRESS PART TO INST.
SUB 1,1
JMP R-2
CM5:773
A
JMP @.-1;ANOTHER PLACE TO START.
```

PROGRAM FOR RELOCATING PROGRAMS IN MEMORY

```
HALT
READS  0
HALT
READS  1
MOV    1,3
HALT
READS  2
SUB    0,1
SUBZ#  0,2,SNC
JMP    .+15
STA    3,30
STA    1,31
ADD    1,2
STA    2,32
ISZ    30
ISZ    31
ISZ    32
LDA    0,@30
STA    0,@32
DSZ    31
JMP    .-3
HALT
STA    0,20
STA    1,31
STA    2,22
DSZ    20
ISZ    31
DSZ    22
LDA    0,@20
STA    0,@22
DSZ    31
JMP    .-3
HALT
       .END
```

AUXILIARY LOADER

```
;AUXILIARY LOADER
GET:    SUBZL  1,1
        MOVS   1,1
        SKPDN  TTI
        JMP    .-1
        DIAS   0,TTI
        ADD    0,1,SZC
        JMP    0,3
        ADDL   1,1
        ADDL   1,1
        ADDL   1,1
        ADDL   1,1
        JMP    GET+2
        NIOS   TTI          ;START HERE
        LDA    2,ADDR
        STA    2,20
        JSR    GET
        STA    1,@20
        JMP    .-2
ADDR:   1256
```

APPENDIX III

PROGRAM FOR DIGITAL SIMULATION OF THERMO-DILUTION CURVES

```
      DIMENSION C(200)
      IOUT=3
      READ(1,104)XK,ALPHA,BETA,AT
      IX=298432767
      CALL GAUSS(IX,0.2,5.0,XN)
      C(I)=XN
      T=FLOAT(I)/10.0
      IF(T.LE.AT)GO TO 50
      C(I)=C(I)+XK*(T-AT)**ALPHA*EXP(-1.*(T-AT)/BETA)
   50 CONTINUE
      WRITE(3,101)C
      WRITE(2,102)C
      A1=ALPHA+1.0
      GAM=GAMMA(A1)
      AREA=XK*BETA**A1*GAM
      WRITE(3,103)AREA
      XMTT=AT+BETA*A1
      S2=BETA*BETA*A1
      CP=XK*(ALPHA*BETA/2.71828)**ALPHA
      TP=AT+(A1-1)*BETA
      WRITE(IOUT,3)XMTT
      WRITE(IOUT,4)S2
      WRITE(IOUT,5)CP
      WRITE(IOUT,6)TP
      STOP
  101 FORMAT(' ',10F9.2)
  102 FORMAT(10F8.2)
  103 FORMAT(//' AREA= ',F14.6)
  104 FORMAT(4F10.4)
    3 FORMAT(//' MEAN TRANSIT TIME= ',F8.2)
    4 FORMAT(//' S SQUARED= ',F8.2)
    5 FORMAT(//' PEAK CONCENTRATION= ',F8.2)
    6 FORMAT(//' PEAK TIME= ',F8.2)
      END
C     THIS IS THE IBM SSP SUBROUTINE GAUSS
```

```
      SUBROUTINE GAUSS(IX,S,AM,V)
      A=0.0
      DO 50 I=1,12
      CALL RANDU(IX,IY,Y)
      IX=IY
   50 A=A+Y
      V=(A-6.0)*S+AM
      RETURN
      END
C     THE IBM - SSP SUBROUTINE  R A N D U  FOR GENERATING UNIFORM
C     PSEUDO-RANDOM NUMBERS
      SUBROUTINE RANDU(IX,IY,YFL)
      IY=IX*65539
      IF(IY)5,6,6
    5 IY=IY+2147483647+1
    6 YFL=YFL*.4656613E-9
      RETURN
      END
```

PROGRAM FOR FITTING EQUATION 3 TO THE THERMO-DILUTION CURVES

```
      DIMENSION C(200)
      NAMELIST/L12/XK,BETA,A1,AREA
      COMMON IOUT,ICARD,IWID
C     THE FOLLOWING PARAMETERS SPECIFY I/O DEVICES
      IOUT=3
      ICARD=1
      IWID=110
      NP=200
      COR=0.825
   10 READ(ICARD,1,END=50)C
      DT=0.1
      CALL CAREA(C,NP,DT,CCR,AREA,AT,BETA,A1,XK)
      WRITE(IOUT,L12)
      WRITE(IOUT,2)AREA
      XMTT=AT+BETA*A1
      S2=BETA*BETA*A1
      CP=XK*((A1-1)*BETA/2.71828)**(A1-1)
      TP=AT+(A1-1)*BETA
      WRITE(IOUT,3)XMTT
      WRITE(IOUT,4)S2
      WRITE(IOUT,5)CP
      WRITE(IOUT,6)TP
      GO TO 10
   50 STOP
    1 FORMAT(10F8.2)
    2 FORMAT(//' AREA= ',F8.2)
    3 FORMAT(//' MEAN TRANSIT TIME= ',F8.2)
    4 FORMAT(//' S SQUARED= ',F8.2)
    5 FORMAT(//' PEAK CONCENTRATION= ',F8.2)
    6 FORMAT(//' PEAK TIME= ',F8.2)
      END
      SUBROUTINE CAREA(C,NP,DT,CCR,AREA,AT,B,A1,XK)
      DIMENSION C(NP)
      DIMENSION IPLOT(110)
      COMMON IOUT,ICARD,IWID
      DATA ISP/1H /
      DATA ID/1H /,ICC/1HC/,IYY/1HX/
      NAMELIST/L57/GAM,D,X2Y,X1Y,X12,X2S,X1S  ,I1,BL
      SX1=0.0
```

```
      SX2=0.0
      XI2=0.0
      SX1S=0.0
      DO 10 I=1,10
      SX1=SX1+C(I)
   10 SX2=SX2+C(I)*C(I)
      CALL MASTD(SX1,SX2,10,AVG,STD)
      DO 20 I=11,NP
      IF(C(I)-AVG.GT.3.0*STD)GO TO 30
      SX1=SX1+C(I)
      SX2=SX2+C(I)*C(I)
   20 CALL MASTD(SX1,SX2,I,AVG,STD)
   30 J=I-1
   40 IF(C(J)-AVG.LE.STD)GO TO 50
      J=J-1
      SX1=SX1-C(J)
      GO TO 40
   50 I1=J
      BL=SX1/FLOAT(J)
      SX1=0.0
      SX2=0.0
      SX2S=0.0
      SX12=0.0
      SX1Y=0.0
      SX2Y=0.0
      SY=0.0
      SW=0.0
      ISTART=I1+1
      DO 60 I=ISTART,NP
      XI2=XI2+DT
      XI1=ALOG(XI2)
      BOBO=C(I)-BL
      YI=ALOG(ABS(BOBO))
      W=YI*YI
      SX1=SX1+XI1*W
      SX1S=SX1S+XI1*XI1*W
      SX2=SX2+XI2*W
      SX2S=SX2S+XI2*XI2*W
      SX12=SX12+XI1*XI2*W
      SX1Y=SX1Y+XI1*YI*W
      SX2Y=SX2Y+XI2*YI*W
      SY=SY+YI*W
      SW=SW+W
   60 CONTINUE
      X1S=SX1S-SX1*SX1/SW
      X2S=SX2S-SX2*SX2/SW
      X12=SX12-SX1*SX2/SW
      X1Y=SX1Y-SX1*SY/SW
```

```
      X2Y=SX2Y-SX2*SY/SW
      D=X1S*X2S-X12*X12
      ALPHA=(X2S*X1Y-X12*X2Y)/D
      B=-D/(X1S*X2Y-X12*X1Y)
      XMU=(SY-ALPHA*SX1+SX2/B)/SW
      XK=EXP(XMU)
C
C     COMPUTE CARDIAC OUTPUT
C
      A1=ALPHA+1.0
      GAM=GAMMA(A1)
      WRITE(3,L57)
      AREA=GAM*XK*(B**A1)
C
C     PLOT RESULTS
C
      DO 2 I=1,110
    2 IPLOT(I)=ID
      AT=I1*DT
      E=0.0
      AX=0.0
      DO 4 I=1,NP
      C(I)=C(I)-BL
    4 AX=AMAX1(AX,C(I))
      WRITE(LOUT,7)
      T=0.0
      IW=IWID-14
      PSC=FLOAT(IW)/AX
      DO 31 I=1,NP
      T=DT+T
      IF(T.LE.AT)GO TO 35
      T=T-AT
      E=XK*(T**ALPHA)*EXP(-T/B)
      T=T+AT
   35 IC=1.0+E*PSC
      IF(IC.GT.IW)IC=IW
      IY=1.0+C(I)*PSC
      IF(IY.GT.IW)IY=IW
      IF(IY.LT.1)IY=1
   37 IPLOT(IC)=ICC
      IPLOT(IY)=IYY
      WRITE(IOUT,8)E,C(I),(IPLOT(J),J=1,IW)
      IPLOT(IC)=ISP
      IPLOT(IY)=ISP
   31 CONTINUE
      RETURN
    7 FORMAT(' PLOT OF ORIGINAL AND EXTRAPOLATED CURVE --X # ORIGINAL
     C # EXTRAPOLATED',/,/,'EST/OBSERVED',/,/)
```

```
8 FORMAT(' ',2F7.2,'.1',101A1,'1')
  END
  SUBROUTINE MASTD(SX1,SX2,N,AVG,STD)
  AVG=SX1/FLOAT(N)
  STD=SQRT((SX2-SX1*SX1/FLOAT(N))/FLOAT(N-1))
  RETURN
  END
```

REFERENCES

1. Skinner, R. L. and Gehmlick, D. K., Analog Computer Aids heart oilment diagnosis. <u>Electronics</u> 32, 56, 1963.

2. Moody, N. E., Barber, H. D., Holmund, B. A., and Herriman, J. E. A Cardiac output Computer for the rapid analysis of indicator dilution curves. <u>Digest of the 1960 Conference on Medical Electronics New York,</u> July 1961, p 179.

3. Hirashi, H. Hara, J. Weldon Bellville. On-line Computation of Cardiac Output from dye dilution curves. <u>Circulation Res.</u> 12: 379, 1963.

4. Hepner, C. V., J. E. Jacobs, H. U. Wessel, and P. Kezdi. An improved method of automatically computing dilution curves. <u>Conference paper</u> 62-390, 1962. Winter Meeting fo American Institute of Electrical Engineers, New York, N. Y.

5. Wessel, H. V., C. F. Hepner, G. W. Jamer, and P. Kezdi. Performance of a new on-line Computer for indicator - dilution curves. <u>J. Appl. Physiol.</u> 19: 1024-1027, 1964.

6. Stanley Sinclair, John, H. Duft, and Lloyd O. Maclean. Use of a Computer for Calculating Cardiac output. <u>Surgery</u>: Vol 57, No. 3, p 414, March 1965.

7. Albert L. Kunz and Charles W. Smith. Theory and design of an on-line Cardiac output Computer <u>Journal of Applied Physiology</u>: Vol. 23, No. 5, Nov. 1967.

8. P. Sekelj, G. R. Tait, and M. M. Nathanson. Studies on dye dilution curves using a digital computer. IEEE transactions on Bio-medical Engineering. Vol : BME -13, No. 1, Jan. 1966, p. 32.

9. C. Frank Stormer and David O. Clark. Computer Computations of Cardiac output using the Gamma function. Journal of Applied Physiology: Vol. 28, No. 2, Feb., 1970. p. 219.

10. Sing San Yang, M. D. et. al. From Cardiac Catheterization data to the hemodynamic parameters - p. 21 - p. 36. at 1972 by F. A. Davis Company, Philadelphia.

11. Hetzel, P. S., Swan, H. J. C., Ramirez de Arellano, A. A., and Wood, E. H., Estimation of Cardiac output from first part of asterial dye-dilution curves. Journal of Applied Physiology 13; 92, 1958

12. Hamilton, W. F., Riley, R. L., A Hayh, A. M., Cournand, A., Fowell, D. M., Himmelstein, A., Noble, R. P., Ramington, J. W., Richards, D. W., Wheeler, N. C., and Witham, A. C., Comparison of the Fick and dye injection methods of measuring the cardiac output in man. American Journal of Physiology 153 : 309, 1948.

13. Kinsman, J. M., Moore, J. W., and Hamilton, W. F. Studies on the Circulation. I. Injection Method: physical and mathematical considerations. *American Journal of Physiology.* 89: 322, 1929.

14. Stephenson, J. L. Theory of the measurement of blood flow by the dilution of an indicator. *Bull. Math. Biophys.* 10: 117, 1948.

15. Newman, E. V., Merell, M., Genecin, A., Monge, C., Milnor, W. R., and Mckoever, W. P. The dye-dilution method for describing the central circulation. An analysis of factors shaping the time-concentration curves *Circulation.* 4: 735, 1951.

16. Sheppard, C. W. Synthesis of dye-dilution curves (abst.). *American Journal of Physiology.* 171: 767, 1952.

17. Sheppard, C. W. Mathematical Considerations of indicator dilution techniques. *Minn. Med.* 37: 93, 1954.

18. Meier, P. and K. L. [illegible]erler. On the theory of the indicator dilution method for measurement of blood flow and volume. *Journal of Applied physiology.* 12: 731-744, 1954.

19. Stephenson, J. L. Theory of measurement of blood flow by dye-dilution techniques. *IRE Trans. Med. Elect.* P.G. M.E. 12: 82, 1958

20. Sherman, H. On the theory of indicator dilution methods under varying blood flow conditions. *Bull. Math. Biophys.* 22: 417, 1960.

21. James S. Forrester, William Ganz, et. al. Thermo-dilution Cardiac output determination with a Single flow directed Catheter. American Heart Journal. Vol. 83, No. 3, pp. 306-311, March, 1972.

22. Fegler, G. A Thermocouple method of determination of heart output in anaesthetised dogs (abst.). Nineteenth international physiological congress., August 31 to September 4, 1953, Montreal, Canada, p. 341.

23. Rapaport E, Ketterer S. G. The measurement of Cardiac output by the thermo-dilution method. Clin. Res. 6: 214, 1958.

24. Fornek, A., Ganz, V. Measurement of flow in a single blood vessel including Cardiac output by local thermo-dilution. Circ. Res. 8: 175-182, 1960.

25. Evonuk E, Imig C J, Greenfield, W., et. al. Cardiac output measured by thermal dilution of room temperature injectate. Journal of Applied physiology. 16: 271-275, 1961.

26. Hoise, K. F. Thermal dilution Technics. Circ. Res. 10: 491-504, 1962.

27. Brauthwaite, M. A. Bradley R. D: Measurement of Cardiac output by thermal dilution in man. Journal of Applied physiology. 24: 434-438, 1968.

28. Chan F. Lam and Arner, Hohn. Computer determination of Cardiac output by means of impedance plethysmography. 24th ACEMB-international Hotel, Los Vegas, Nevada, October 31sc. - November 4th, 971. page 41.

29. Howard K. Thompson, Jr., C. Frank Stormer., et. al. Indicator transit time Considered a Gamma variate. Circulation Research. Volume XIV, June, 1964, pp 502-515.

VITA

NAME : C. Narasimha Murthy

BORN : Bhadravathi, Mysore State India
December 30th 1947.

EDUCATION: Bachelor of Electrical Engg. 1969
Malnad College of Engg.
Mysore University
Hassan, India.

Master of Arts (Solid State Physics) 1973
Indiana State University
Terre Haute Indiana.

EXPERIENCE: Mysore State Electricity Board 1970
Govt. of India
Bangalore

Systems Engineer I 1974
CAE Electronics
Montreal (Since June 1974).