In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institute shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the Dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

3/17/65
b

A STUDY OF CONTROL SYSTEM DESIGN

AND HARDWARE SELECTION BY COMPUTER

A THESIS

Presented to

The Faculty of the Graduate Division

by

Baxter Lee Thorman

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Mechanical Engineering

Georgia Institute of Technology

November, 1965

A STUDY OF CONTROL SYSTEM DESIGN

AND HARDWARE SELECTION BY COMPUTER

Approved:

Chairman

Date approved by Chairman: May 10, 1966

ACKNOWLEDGMENTS

The author would like to express his gratitude to all those who have contributed to the completion of this study. Dr. Eugene Harrison, of the School of Mechanical Engineering, the faculty advisor, provided the problem and guidance throughout the study. Appreciation is extended to Dr. G. P. Francis, of the School of Mechanical Engineering, and Dr. R. P. Webb, of the School of Electrical Engineering, for their critical review and service on the reading committee, to Mr. J. E. Jones and Mr. V. S. Robyn of Honeywell, Incorporated for their unlimited cooperation in the furnishing of information on Honeywell components, and to Mr. Frank Speckhart for his aid in setting up the numerical integration procedure used in this study. Finally, the author would like to express his gratitude to his wife, without whose programming aid and moral support this work would not have been possible.

TABLE OF CONTENTS

Page

ACKNOWLEDGMENTS ii

LIST OF TABLES v

LIST OF ILLUSTRATIONS vi

SUMMARY vii

NOMENCLATURE TABLE ix

Chapter

I. INTRODUCTION 1

II. PROCEDURE USED IN FEASIBILITY STUDY 4

Comparison Method
Assumptions
Control System Considerations

III. COMPUTER PROGRAM DESCRIPTION 8

General Description
General Design Information
Program Results for Two Examples

IV. RESULTS OF THE COMPARISON OF COST FOR THE TWO DESIGN METHODS 24

Computer Design Cost Considerations
Conventional Design Cost Considerations
Determination of Program Development Cost

V. CONCLUSIONS 33

VI. RECOMMENDATIONS 40

APPENDICES

A. PROGRAM SYMBOL TABLE 44

B. DERIVATION OF SYSTEM FLOW EQUATIONS 54

C. DEVELOPMENT OF CONTROL EQUATIONS 77

APPENDICES (Continued) Page

D. THEORY OF DIFFERENTIAL EQUATION SOLUTION 86

E. COMPUTER PROGRAM FLOW CHARTS 90

F. COMPUTER PROGRAM 115

LITERATURE CITED 155

LIST OF TABLES

Table		Page
1.	Systems Considered for Study	5
2.	Outline of Conventional Design Method for Liquid Level Control System	28
3.	Value of Constant K_4	29
4.	Equivalent Lengths of Valves and Fittings	71
5.	Results of First Flow Equation Test	73
6.	Results of Second Flow Equation Test	76
7.	System Types	77

LIST OF ILLUSTRATIONS

Figure		Page
1.	Diagram of Tank Configuration Two . . . . (Figure 1 is repeated on p. 52 for convenience.)	14, 55
2.	Plot of Level Versus Time for a Low Cost System and a 72-25, Series 1 System	18
3.	Plot of Level Versus Time for a Proportional System and for the First Proportional Plus Integral System Selected by the Program	23
4.	Plot of Design Cost Versus Run Time	31
5.	Flow Versus Valve Lift for an Equal-Percentage Control Valve	58
6.	Free-Body of Fluid in Pipe "L2"	60
7.	Straight-Line Approximation of Assumed Variation of CVI with Reynolds Number	67
8.	On-Off Control System Diagram	79
9.	Plot of Motor Voltage Versus Error	80
10.	Plot of Valve Opening Versus Control Pressure	84
11.	Plot of Control Pressure Versus Error in Level	84

SUMMARY

The objective of the reported study was to determine the feasibility of writing a digital computer program which will, given certain required parameters, select control system components, compute system cost, and analyze system performance with the selected hardware. This objective was achieved by writing a computer program which met the above requirements and comparing its cost (development cost plus cost per run) with the cost of a conventional design of the same control system. The term "conventional" refers to a method whereby the engineer achieves the system design using tabulated data and past experience. The control system selected for this study was a tank liquid level control system.

It was found that even if program development costs were neglected, some computer designs would cost more than equivalent conventional designs, depending on the amount of required run time (the time needed to determine the response of the control system). This result is brought about because the cost per unit run time on the computer is greater than ten times the cost per unit run time for the conventional design method.

If an experienced programmer had been assigned to develop the program, it is estimated that the development cost would be approximately $4,580. This cost is made up of two components, a cost corresponding to a programming time of 500 hours and a cost corresponding to a computer time of 250 minutes.

Because the computer design method cannot compete with the conventional design method on a cost per design basis and because of the high program development cost required, it is felt that computer design of liquid level control systems using a digital computer program of the type developed for this study is not economically feasible.

Even if the results of this study had indicated that a design program of this nature would be slightly profitable, the wisdom of putting such a program into production would be questionable, because of the conservatism of the customers who buy this type of control system.

Three possible means of reducing the required computer time are discussed. The first deals with the selection of a faster numerical integration method, the second is concerned with the use of analog computational units for determining the system response curve, and the third suggests the possibility of using approximate flow equations.

NOMENCLATURE TABLE

A_1	Internal cross sectional area of pipe at D_1
A_3	Internal cross sectional area of pipe at D_3
A_8	Tank cross sectional area
CR	Control range
CV	Valve size coefficient
CVI	Inlet valve size coefficient
CVO	Outlet valve size coefficient
$Cost_{c.d.}$	Cost of computer design
$Cost_{m.d.}$	Cost of manual design
d	Nominal pipe diameter
D	Inside diameter of pipe
D_1	See Figure 1
D_3	See Figure 1
D_8	See Figure 1
E	Measure of roughness of inside pipe surface
F_e	External force
F	Friction factor
F_f	Friction force in pipe
F2	Friction factor—L2 section of pipe
F3	Friction factor — L3 section of pipe
F4	Friction factor—L4 section of pipe
g	Acceleration due to gravity

g_c	Gravitational constant
G	Specific weight
h_f	Head loss due to friction
H_I	Initial value of head
H	Head of liquid in tank
H1	Required head
H2	See Figure 1
H3	See Figure 1
K_1	Constant $0.04/sec.
K_2	Constant whose value is a function of the system type
K_3	Constant—Cost per second for keeping an engineer in the field
K_4	Constant whose value depends on the system type
KI	Ratio of flow through inlet control valve to maximum flow through the same control valve
K2O	Ratio of flow through outlet control valve to maximum flow through the same control valve
L	Pipe length
L1	Tank height
L2, L3, L4, L5	Equivalent pipe lengths (See Figure 1)
L2T, L3T, L4T, L5T	True pipe lengths

M	Mass of fluid in pipe
N	Number of moles of gas
P	Control pressure
PO	See Figure 1
POO	Initial tank charging pressure
P1	See Figure 1
P2	
P3	
P4	
P5	
P6	
P7	
P13	Constant pressure above liquid in tank
PB	Proportional band
P_{mi}	See Figure 10
P_{ma}	
Q	Flow rate
Q_I	Flow rate into tank
Q_m	Maximum flow rate
Q_o	Flow rate from tank
QOT	Quick opening time
r	Radius of pipe bend
R	Valve rangeability
R_e	Reynolds number
R_o	Universal gas content
RR	Repeat rate

S.G.	Specific gravity
T	Time
T_I	Initial temperature
T1	Time interval
T7	Temperature at D_7 section
TRR	Required rise time
U	Fluid viscosity
V1	See Figure 9
V	Velocity of fluid in pipe
V_m	Motor voltage
V_o	Tank volume
V_{oi}	Initial tank volume
V_r	Liquid velocity in control volume
w	Fluid specific weight
w_w	Specific weight of water
x	Position of valve stem
X	Maximum position of valve stem
ΔH	See Figure 8
ΔP	Pressure drop across the control valve
ρr	Density of liquid in region R
ρs	Density of liquid at surface S

CHAPTER I

INTRODUCTION

The object of the study described in this thesis is to determine the feasibility of writing a digital computer program which will, given certain required parameters such as maximum allowable rise time, maximum overshoot, maximum settling time, and maximum allowable steady state error, select system components for a control system with a known block diagram, compute system cost, and analyze the system performance with the selected hardware.

Since the inception of modern high speed digital computers, attempts have been made to apply them to various engineering design problems. Computer programs have been written for finding thermal stresses in piping, for calculating pressure drops in piping systems, for calculating stresses in structures, and for calculating power plant heat balances.

In every case listed above, the digital computer was ideally suited to the problem because of the large number of repetitive calculations required for solution.

Recently, attempts have been made to enlarge the field of computer applications. The IBM "COMMEND" program is one such attempt (1). This program attacks design problems previously done by graphical or experimental methods, and short duration calculations performed occasionally by an engineer. The following mechanical design areas are included in

COMMEND I: kinematic synthesis, kinematic analysis, component design, physical properties calculations, dynamic analysis, detail drawings, numerical control machining instructions, and engineering information retrieval.

The computer-aided design project at the Massachusetts Institute of Technology is an example of work in another area of computer application (2). This effort centers around the use of a light pen and a cathode-ray oscilloscope. With this concept, the engineer draws his design on the oscilloscope and the computer analyzes the design almost concurrently.

During and after World War II, tremendous advances in automatic control design theory took place. Control design changed from an art to a science with the increased application of the mathematical approach to control problems.

The use of both analog and digital computers for determining the time response of automatic control systems is common today (3). Gain settings which will give the required system response are the end product of most present computer solutions. Most of these programs require frequent communication between the computer and the engineer during the design process. If the control system is to be made up of purchased hardware, as is often the case in process control work, the designer must still select components with the required parameters from various catalogs.

The objective of this study, outlined in the first paragraph of this section, will be achieved by writing a program meeting the above requirements and comparing its cost (computer cost plus programming cost) with an estimated cost for a conventional design. This cost comparison must be augmented by consideration of the ability of the two methods to accomplish the design.

Some of the information obtained from the specific control system design study above can be extrapolated to provide information on the feasibility of computer-aided control system design in general.

CHAPTER II

PROCEDURE USED IN FEASIBILITY STUDY

Comparison Method

One method of studying the feasibility of control system design by digital computer would involve selecting a control system, writing a computer program capable of carrying out the specified design, determining the total cost (program development cost plus cost per run) and comparing this cost to an estimated cost for a conventional design of the same system. This was the method used in the reported study.

Assumptions

The above comparison method makes the following assumptions:

1. The control system requires design methods of sufficient complexity to warrant computer solution.

2. The results of the conventional design and the computer design are equivalent.

3. There is no advantage of one design method over the other that cannot be evaluated numerically.

Control System Considerations

Any control system that is eligible for complete design by computer must meet several requirements. The block diagrams for the various configurations of the system must be fairly well standardized.

The subject study is concerned with control systems having known block diagrams. The design consists of finding the right components and the correct gain adjustments, if applicable, for those components. As long as the forms of the block diagrams are known, several different forms can be tried sequentially until an acceptable design is found. No component design will be required. The dynamic equations representing the plant and the control system must be available. Finally, there must be a systematic procedure available for determining the required gain settings.

A partial list of systems for possible use in this study was compiled from the literature. This list is given in Table 1.

Table 1. Systems Considered for Study

<u>PROCESS</u>
Distillate Column Control
Exothermic Reactor Temperature Control
Liquid Level Control
<u>SERVOMECHANISM</u>
Position Control
Speed Control
<u>POWER GENERATION</u>
Combustion Control
Feedwater Flow Control
Superheater and Reheater Temperature Control
Air Heater Temperature Control

It was felt that the control system for this study should be selected from the industrial controls field rather than the military or aerospace areas, primarily because of the larger market for industrial control systems. Other considerations included the special design requirements connected with most military applications and the uncertainty of the military market.

The power generation control systems and the distillate column control listed in Table 1 were eliminated from consideration in the study because of the lack of usable information on dynamic equations describing the systems and due to the general complexity of the systems. Position and speed controls were eliminated because no complete listing of components was available. A tremendous amount of research would have been required to compile such a list. The exothermic reactor temperature control system was given considerable serious thought, but was eliminated on the basis of discussions with local consulting engineers and representatives of control system component manufacturers. These discussions indicated that the market for exothermic reactor controls was fairly small.

These considerations resulted in the selection of the tank liquid level control system as the system to be used in this study. Liquid level control systems find application where a constant liquid level or head is required. They are widely used in the textile, chemical, petroleum, food, and metals industries. Because liquid level control systems are so widely used, it was felt that there might be enough demand for new liquid level control systems to warrant development of a computer program specifically for their design.

Level control systems require varying degrees of design sophistication. Level tolerances in industry range from fractions of an inch to several feet. Therefore, component and system costs will vary widely. Overdesign of the system can be fairly expensive.

Another reason for the selection of the liquid level control system was the availability of the dynamic equations representing the response of the system components. Campbell (4), Buckley (5), and Eckman (6) concern themselves to varying degrees with liquid level control.

The amount of available information on components was another factor in the selection of the liquid level control system. One of the key components of a liquid level control system is the control valve; and most valve manufacturers supply publications on the characteristics of their particular product line. Another source is a paper by Forman and Oriolo (7) which contains a large amount of information on control valves.

CHAPTER III

COMPUTER PROGRAM DESCRIPTION

General Description

From a list of available components, the computer program written for this study selects the two most economical sets of components that will meet the given response requirements. This is accomplished by determining a trial CV (valve size coefficient) on the basis of the initial flow into the tank, setting the gain adjustments to predetermined values, and plotting a trial response curve. If the resulting response curve does not meet the design criteria satisfactorily, the trial system is adjusted, either by changing component gains or by changing the type of component used, and the data for the response curve is calculated again. If the response curve is satisfactory, control valves, level sensors, and controllers with approximately the same characteristics as those which resulted in the acceptable response curve found above are selected from a list of components stored on a tape. The characteristics of the two most economical sets of selected components are used in the response curve calculations to verify the acceptability of the resulting control system and, if the system is acceptable, cost, component ordering information, and gain settings are printed as the program output.

If one or both of the two selected sets of components fail to achieve the required response curve characteristics, gain adjustments are made if possible, or the system or systems are discarded and other sets of components are selected. If no components with the required characteristics are available, a message to that effect is printed, and work on the program stops.

The program simulates the types of control systems available (low cost proportional, general proportional, and proportional plus integral) in order of increasing cost. The first control system to be simulated (the low cost proportional system) requires a proportional band setting of 30 per cent. If the response curve corresponding to a representative low cost system is acceptable, the two most economical sets of hardware that are compatible with the low cost system are selected, and the response curves are run again to verify the acceptability of the two resulting low cost proportional control systems. If the low cost proportional system is not acceptable, the general proportional system (proportional band equal five per cent) is simulated and, if its response is adequate, two sets of components compatible with the general proportional control system are selected. The same procedure is used for the proportional plus integral control systems. In every case, after the response of a representative of one of the three types of control systems is acceptable, the two most economical sets of hardware (control valve, level sensor, and controller) compatible with that type control system are selected and checked in the response calculations. The coordinates of the response curves can be printed if desired.

Component Information

Honeywell components were used exclusively throughout the study as it was felt that program capability could be demonstrated adequately with only one manufacturer's components.

The following types of control valves are listed on the tape used with the program: angle, double seated, single seated, low flow, Saunders, split body, and cage. Two types of pneumatic actuators and one type of electric actuator are listed. The valve size range covered in the program is from three-fourths to four inches nominal diameter. Foreman (8) states that 96 per cent of all valves used in general chemical or petrochemical facilities are four inches or under. The same reference indicates that butterfly valves are not generally used in sizes less than four inches and none was considered in the study.

A positioner is sometimes used in connection with a valve actuator if the static friction forces which result when the valve is opened are large or when the actuator response is too slow. Basically, the positioner is a form of pilot valve. When a positioner is used, the control air pressure is not applied to the actuator diaphragm as it is when no positioner is used, but is used to vary a nozzle opening which controls the flow of supply air (at pressures of 20 to 100 psig) to the actuator diaphragm. A feedback linkage is required to re-position the flapper when the valve opening corresponding to the input pressure is reached.

Forman also states that positioners are used on all four inch and larger control valves. Therefore, the devised program was set up so that selection of a four inch valve automatically calls for the use of a positioner. Saunders valves used in throttling service require the use of a positioner on all sizes, and the program provides these positioners automatically. All valves fail shut.

Information on 81 full capacity and reduced capacity v-port and quick-opening control valves is stored on the tape.

Based on the Corrosion Table on pages 74 and 75 of reference (9), 161 possible flowing mediums can be handled by this program. A "Proceed with Caution" message can be seen on the program output (page 16 of this study) after the body material entry of the first system selected in the first example included in this study. Certain combinations of flowing fluids and body materials will cause this message to be printed. The "Proceed with Caution" message indicates that corrosion will result with this combination of fluid and body material, but that it is not dangerously rapid.

Cast iron, bronze, cast steel, and 316 stainless steel are the available body materials for valves included in the study. Available trim materials are stellite, 440C stainless steel, Hastelloy B and Hastelloy C, Durimet 20, Monel, and 316 stainless steel. Radiating fin, bellows seal, and extension column bonnets can be selected by this program.

Honeywell offers a wide range of level sensors, controllers, and displacer-controller combinations. Four types of displacer-controller combinations are available. Since the major difference in these four types of displacer-controllers is the method of mounting, only one type, the top mounted Series 1 unit was included on the component tape. Slight price increases must be added to the cost listed on the program output if one of the more expensive displacer-controllers is desired. Other types of level sensors included are static pressure sensors and a differential pressure-to-current transmitter.

Controllers with three types of control action are included in this study. These three types are on-off, proportional, and proportional plus integral.

The on-off controller, which is the least expensive of the three types, can only maintain a liquid level between two points; it cannot hold a given level. Furthermore, the on-off system must be used with an electrically conductive flowing medium.

The proportional systems used in the study can be divided into two types; a type in which a general controller is used which has an infinite number of gain settings, and a type called the low cost system. Two low cost systems are available from Honeywell: System A and System B. System A is made up of a static pressure transmitter, a positioner-controller which mounts on the control valve actuator, and the control valve. Only control valves with Honeywell 05 actuators can be used in System A. Three possible proportional band settings are available with System A. The value of these three settings is determined by the control valve travel. System B is slightly higher in price than System A, and is made up of a static pressure transmitter, a controller which mounts on the static pressure transmitter, and the control valve. Any Honeywell valve can be used in System B, but only three proportional band settings (30, 50, and 100 per cent) are available.

The most sophisticated controller used in this study is the proportional plus integral controller. Proportional plus integral control is available with a displacer-controller combination (72-25, Series 1) or a general purpose two mode controller used in conjunction with a static pressure transmitter.

Since controller selection is largely a matter of individual preference after the number of control modes required is determined, no attempt was made to list all available controllers on the component tape for this feasibility study. The controller prices listed in the program must be adjusted if special controller features are desired.

Two tank configurations, Configuration One and Configuration Two are used in this study. Configuration Two (Figure 1) requires that flow out of the tank be through a discharge pipe and is the more general configuration. Configuration One, which requires that flow out be from the top of the tank, would be used in a dipping or dyeing process in which material is placed in the tank. When this material leaves the tank, some of the liquid in the tank is removed with it.

The complete computer program is given in Appendix F.

Program Results for Two Examples

Input information and program results are given below for two examples.

Example One

It is desired to design a system having the following known physical characteristics. Configuration Two was selected because it illustrates more of the program capabilities. The tank used is a 125-lb., vented tank. Water is to be the system fluid, pneumatic control is desired, and a full capacity valve is needed. For the location of the following variables, see Figure 1.

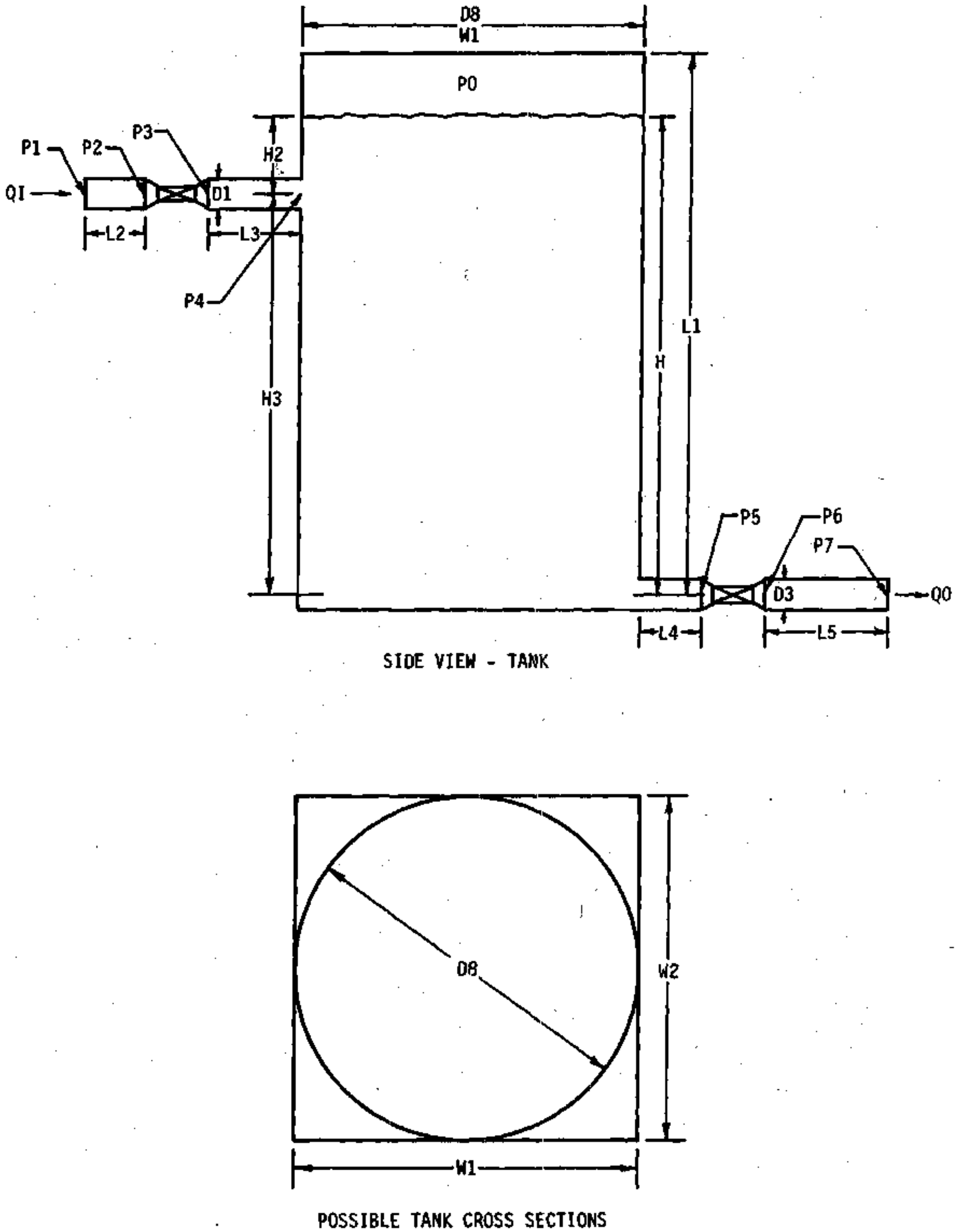


Figure 1. Diagram of Configuration Two.

Pressures—psia	
P0	15
P1	30
P7	17.25

Temperatures—F	
T7	80

Dimensions—Inches	
D1	4.0
D3	4.0
DV3	4.0
D8	36.0
H3	12.0
L1	200.0
L2T	150.0
L3T	100.0
L4T	100.0
L5T	150.0

Outlet Valve Size Coefficient—Inches^4/Lb. Seconds	
CVO	654

Pipe Roughness—Inches	
E	0.00085

Fluid Characteristics	
U	1×10^{-7} lb sec/in^2
G	0.036 lb/in^3

The following fittings are included in the indicated pipe sections:

L2 section—1 conventional swing check valve

L3 section—1 conventional disc gate valve, 2-90° pipe bends (r/d = 4.0)

L4 section—1 in-line ball check valve

L5 section—1 conventional disc gate valve

This system is to meet the following criteria if the required head H1 is changed by a step change from 100 inches to 125 inches with the outlet valve fully opened, (K20 = 1.00).

Required rise time	45 sec.
Maximum overshoot	2.0 in.
Maximum settling time	100 sec.
Maximum steady state error	0.500 in.

The initial conditions are:

$$QI = 768\ in^3/sec.$$

$$QO = 768\ in^3/sec.$$

$$H = 100\ in.$$

$$XT = 0.82$$

$$H1 = 125\ in.$$

The program output (excluding response curve points) for Example One is given below.

Proportional Band = 0.05

System	Component	Cost
ONE	Type 72-25 Series 1	$ 260.00
	Valve Model No. 1101	
	Size 3.00 in.	
	Type double seated	
	CV 112.00	
	Actuator No. 13	0
	Body Material cast iron	
	(Proceed with caution)	

System	Component	Cost
ONE (Continued)	Body rating 125 lb.	$ 326.00
	(Flanged end)	
	Plain bonnet	
	Trim material 316 S.S.	0
	TOTAL COST	$ 586.00
TWO	Type 72-25 Series 1	$ 260.00
	Valve Model No. 1001	
	Size 3.00 in.	
	Type Angle	
	CV 115.00	
	Actuator No. 13	0
	Body Material 316 S.S.	
	Body Rating 150 lb.	914.00
	(Flanged end)	
	Plain bonnet	
	Trim material 316 S.S.	0
	TOTAL COST	$1,174.00

Figure 2 shows plots of level versus time for a low cost system and the first of the systems selected by the program, a 72-25 Series 1 system using a Model 1101, three inch, double seated control valve.

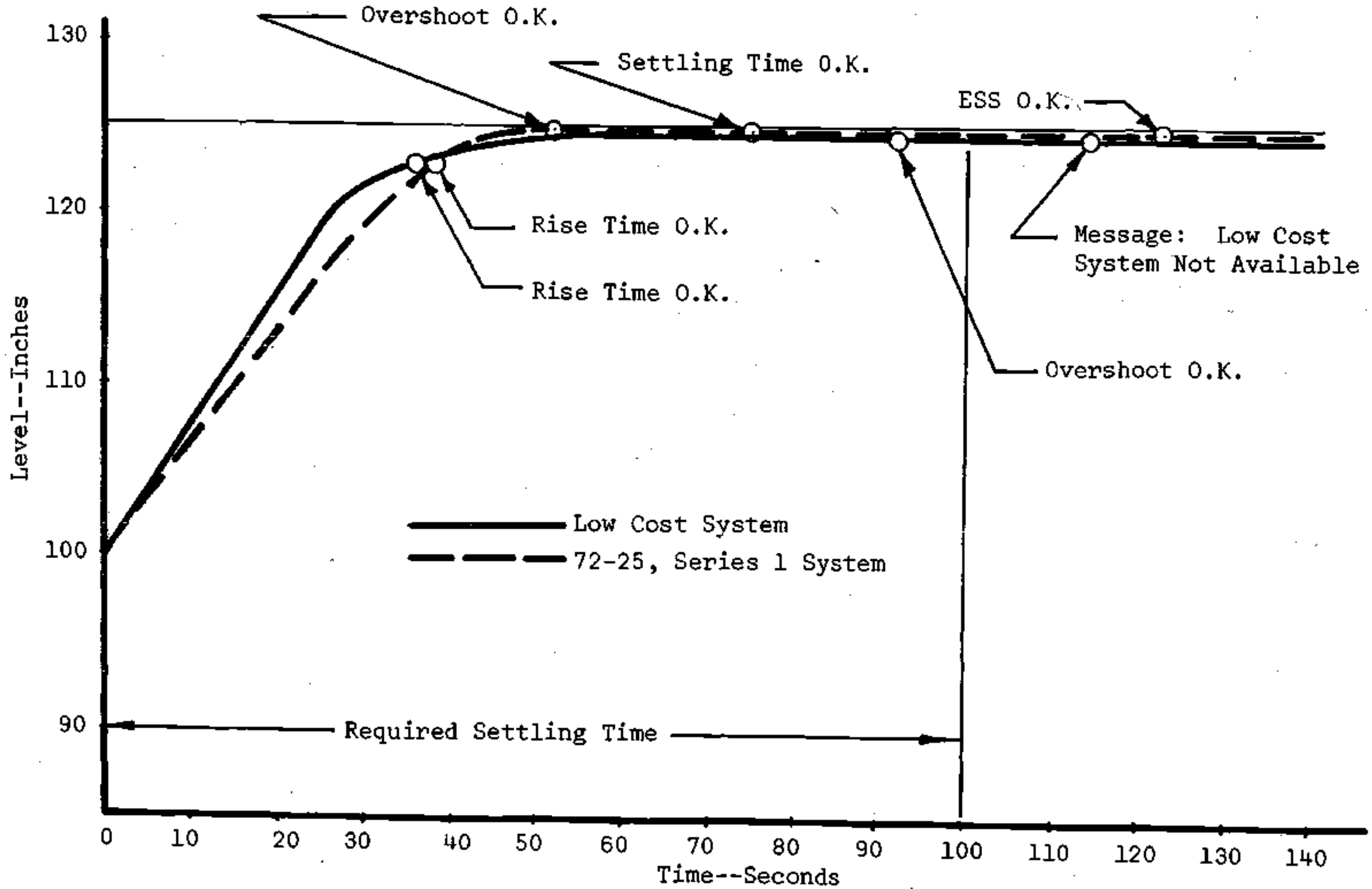


Figure 2. Plot of Level Versus Time for a Low Cost System and a 72-25, Series 1 System.

The low cost system was eliminated by the program because it did not meet the settling time requirement. Inspection of Figure 2 indicates that the low cost system does meet this requirement if it is defined as the time required for the response to stay within a band, centered on the desired level, whose width equals 10 per cent of the initial error. The problem is caused by the fact that the program is set up so initiation of the settling time check must wait until the overshoot check is completed. Because of the response sluggishness caused by the high proportional band requirement for the low cost system, the overshoot check consumed so much time that the maximum allowable settling time was exceeded. It is felt that the program method of checking settling time is acceptable for program demonstration, but, for production use, the program should be changed to allow the overshoot and the settling time checks to run concurrently.

Had the low cost system been selected, an example of the program output would appear as shown below.

Proportional Band = 0.30

System	Component	Cost
ONE	System B	$ 147.00
	Valve Model No. 1101	
	Size 3.00 in.	
	Type double seated	
	CV 112.00	

System	Component	Cost
	Actuator No. 13	
	Body Material cast iron	
	(Proceed with Caution)	
	Body Rating 125 lb.	$ 326.00
	(Flanged end)	
	Plain Bonnet	
	Trim Material 316 S.S.	0
	TOTAL COST	$ 473.00

Figure 2 clearly illustrates the effect of CV and proportional band on the system response. The level of the low cost system, with a CV of 485.80 rises faster than that of the 72-25 system which used a CV of 431.00. Because of its smaller proportional band, 0.05 versus 0.30, the 72-25 system reached its final position much faster. Figure 2 shows that the larger the proportional band used, the more sluggish is the resulting response.

Example Two

It is desired to design a system having the same physical characteristics, initial conditions, desired level change, required rise time, overshoot, and settling time. However, the allowable steady state error is now 0.07 inches.

The program output is given below.

Proportional Band = 0.05

Repeat Rate = 0.50

System	Component	Cost
ONE	Type 72-25 Series 1	$ 285.00
	Valve Model No. 1101	
	Size 3.00 in.	
	Type double seated	
	CV 112.00	
	Actuator No. 13	0
	Body Material cast iron	
	(Proceed with caution)	
	Body Rating 125 lb.	326.00
	(Flanged end)	
	Plain Bonnet	
	Trim Material 316 S.S.	0
	TOTAL COST	$ 611.00
TWO	Type 72-25 Series 1	$ 285.00
	Valve Model No. 1001	
	Size 3.00 in.	
	Type Angle	
	CV 115.00	
	Actuator No. 13	0
	Body Material 316 S.S.	
	Body Rating 150 lb.	914.00

System	Component	Cost
TWO (Continued)	(Flanged End)	
	Plain Bonnet	
	Trim Material 316 S.S.	0
	TOTAL COST	$1,199.00

Plots of level versus time for a proportional system and for the first proportional plus integral system selected by the program are given in Figure 3, which indicates that the addition of the integral mode makes the transient response less stable (the curve is now oscillatory) but it forces the steady state error to approach zero. As in Example One, the difference in slopes during the transient response for the two systems is caused by a difference in the CV used.

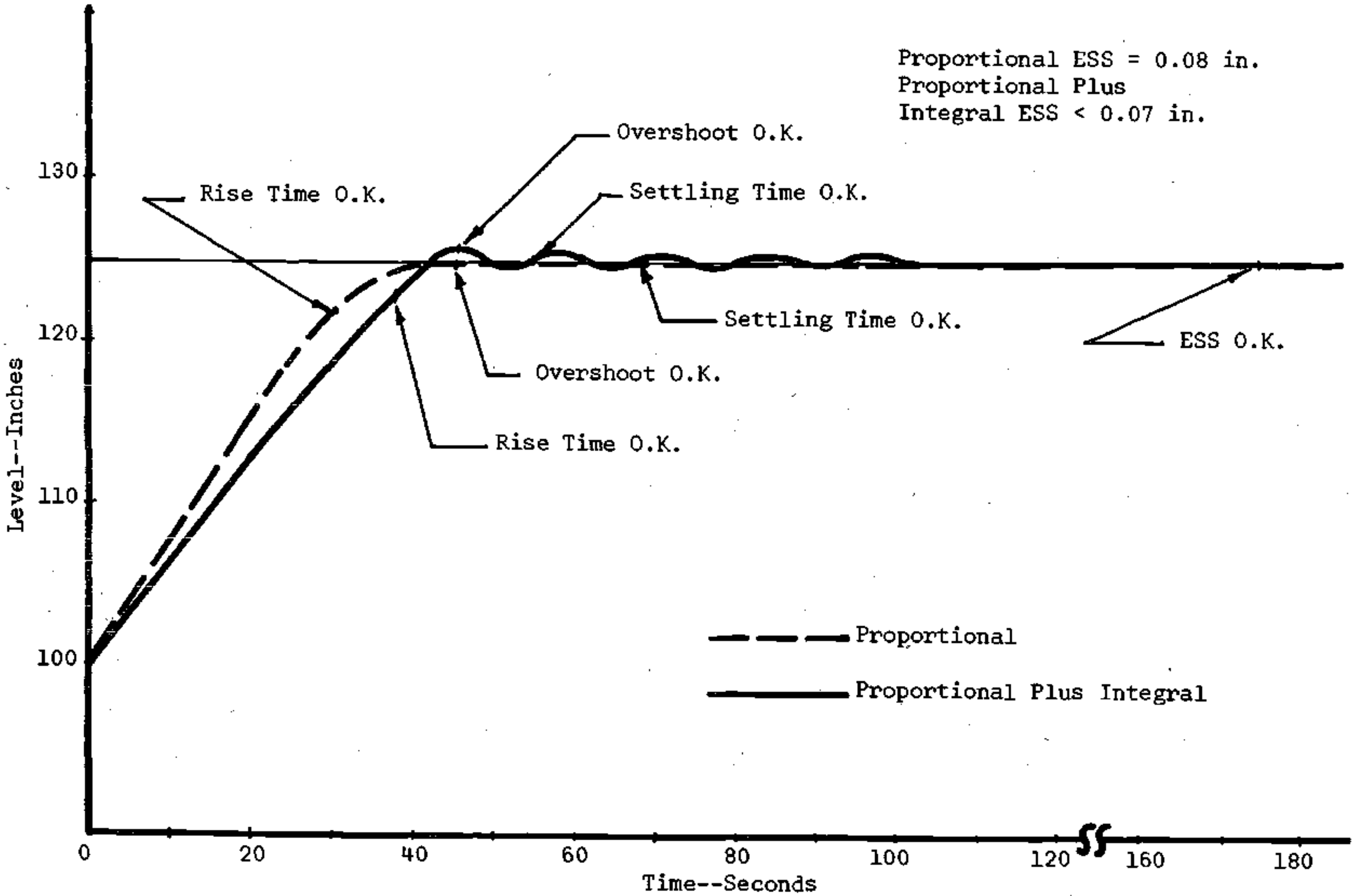


Figure 3. Plot of Level Versus Time for a Proportional System and for the First Proportional Plus Integral System Selected by the Program.

CHAPTER IV

RESULTS OF THE COMPARISON OF COST FOR THE TWO DESIGN METHODS

Neglecting development cost and any costs common to both conventional and computer designs, the cost per design completed on the computer and the cost per conventional design are made up of two quantities.

Computer Design Cost Considerations

The computer design cost consists of a constant cost which covers compilation time, component selection time, input time, and any other miscellaneous times that can be considered constant for all runs, and a variable cost which is a function of the type system used and the time required to test the system response on the computer. For this program, the constant time was found to be approximately 170 seconds.

The variable cost was determined as follows: The cost of problem time on the Burrough's B5500 computer was taken as $0.04 per problem second. This cost was obtained by dividing the computer cost per hour by 3600 since one problem second took approximately one real time second.

The present program requires that the system response curve coordinates be calculated several times. The number of times that the curve must be determined is a function of the type system necessary to meet the design requirements since the program always tries the different types of control systems in order of increasing cost.

Three calculations of the response curve are required if the low cost proportional system can be used; one general plot which determines the applicability of a representative low cost system, and one run through the response curve for each of the two low cost systems (sets of hardware) selected.

If the low cost system cannot meet any one of the required design criteria, a general proportional control system is simulated. If this general proportional system will satisfy the design criteria, the system response curve will have to be calculated four times; once for the low cost system which could not meet the design criteria, once for the general proportional system, and once for each of the specific sets of proportional system components selected by the program.

In both cases above, no time need be required for setting system gains. The response curve for most practical systems having proportional control is exponential and there is no problem with stability. The proportional band is set at a predetermined minimum and no further adjustment is possible. Gain adjustment is necessary only when simulating a proportional plus integral control system.

Finally, if a proportional plus integral system is required, and it is assumed that the system gains can be adjusted in two trials, seven plots of the system response curves are required. The first checks the response of the low cost proportional system, which is not acceptable. The second calculation of the response curve coordinates is made for a general proportional system which is also not acceptable. The third, fourth, and fifth determinations of the system's response are required to test a general proportional plus integral control system and to adjust its gains, and the sixth and seventh runs are made with the specific sets of components to be used in the final systems.

It is assumed that the person in charge of installation of the equipment can make the gain settings at the time of installation, since these gains are given on the computer output. If this is true, very little design time is required for tuning the system in the field.

The length of time required per run is assumed to be M, an arbitrary constant found in the program by multiplying the required rise time by four. Although the program is set up so that run time does not reach M in all cases because calculation of the response curve points is stopped when the design requirements are met, no estimate of the shorter time is possible and M must be used. This fact will result in calculated computer design costs slightly higher than true computer design costs.

Mathematically, the above information can be expressed by the following equation:

$$\text{Cost}_{\text{c.d.}} = K_1(\text{Constant time}) + K_1 K_2 M \qquad (1)$$

where

$M = 4(\text{TRR})$

$K_1 = \$0.04/\text{second}$

$$K_2 = \begin{cases} 3 & \text{Low cost system} \\ 4 & \text{Proportional system} \\ 7 & \text{Proportional plus integral system} \end{cases}$$

or

$$\text{Cost}_{c.d.} = \$6.80 + \$0.04\ K_2M \qquad (2)$$

Conventional Design Cost Considerations

Conventional design cost is made up of a constant component selection cost and a variable cost due to the time spent tuning the completed control system in the field.

The component selection cost was arrived at as follows: It is assumed that the designer's salary works out to $4.00 per hour, that the overhead per hour in the office is 100 per cent, and that two hours are required to complete the design using the outline given in Table 2. Therefore the constant design cost will be

$$(2.00)(2\ \text{hours})(\$4.00/\text{hour}) = \$16.00$$

Table 2. Outline of Conventional Design Method for Liquid Level Control System

I. Find Maximum Flow into Tank, QI

A. Use maximum flow from tank plus safety factor

B. Find the flow rate required to fill the tank in a specified time

II. Find Pressure Drop Across the Control Valve

A. Assume $\Delta P = P1 - P0$ (P1, P0 are both known)

III. Find CV

Table 2. Outline of Conventional Design Method for Liquid Level Control System (Continued)

A. $CV = \frac{QI}{\sqrt{\frac{\Delta P}{S.G.}}}$

IV. <u>Select Control Valves on the Basis of</u>

A. CV

B. Body material

C. Trim material

D. Maximum allowable pressure drop across the control valve

E. Maximum allowable static pressure

F. Bonnet required

G. Body strength

H. Availability of Positioner

V. <u>Select Level Sensor on the Basis of</u>

A. Tank configuration

1. Open

2. Closed

B. Required control range

VI. <u>Select Controller</u>

A. Allowable steady state error

System tuning cost is estimated as follows: It is assumed that the field engineer's salary is equivalent to $4.00 per hour, that the overhead cost per hour in the field is 200 per cent of the engineer's salary, that the field engineer can tune the system in the same number of trials as used in the computer program, and that the average time required for the field engineer to travel to and from the equipment location is one hour. In contrast to the computer simulation of the control system, on-site calibration must take into consideration the time for the tank to return to its initial condition between trials. This time is taken as M. Therefore, one trial takes M seconds, two trials take 3M seconds, and three trials take 5M seconds. The coefficients of M just found are called K_4 and are listed below as a function of the system used.

Table 3. Value of Constant K_4

System	K_4
Low cost	1
Proportional	1
Proportional plus integral	5

The cost per second of keeping the field engineer on the job is a constant called K_3. This constant equals (\$4.00/hour)(3.0)(1/3600 sec.).

The above information can be summarized by the following equation for conventional design cost.

$$\text{Cost}_{m.d.} = \$16.00 + K_3(K_4M + 3600) \qquad (3)$$

where K_4 and M are as previously defined.

A plot of Design Cost versus M for the three types of control systems used in this study is shown in Figure 4.

Determination of Program Development Cost

The engineering time required for writing the computer program included here was 1041 hours. The computer time used was 495.82 minutes. It is felt that both the time required for writing the program and the computer time could be cut in half if an experienced person had done the work. This would result in a programming time of approximately 500 hours and a computer time of 250 minutes. If computer time cost \$140 per hour, the engineer's salary was equivalent to \$4.00 per hour, and overhead per hour is 100 per cent of the engineer's salary, the program development cost would be \$4,583.

It is obvious from Figure 4 that even neglecting program development costs, using the present numerical integration technique will result in some computer designs costing more than equivalent conventional designs, since many practical systems will require run times of several hundred seconds. This is due to the fact that the cost per unit of run time (which is one component of the slope of the lines in Figure 4) for the computer design is greater than ten times the cost per unit time for the conventional design.

Since almost all of the run time is spent in performing the numerical integration necessary for solution of the system of differ-

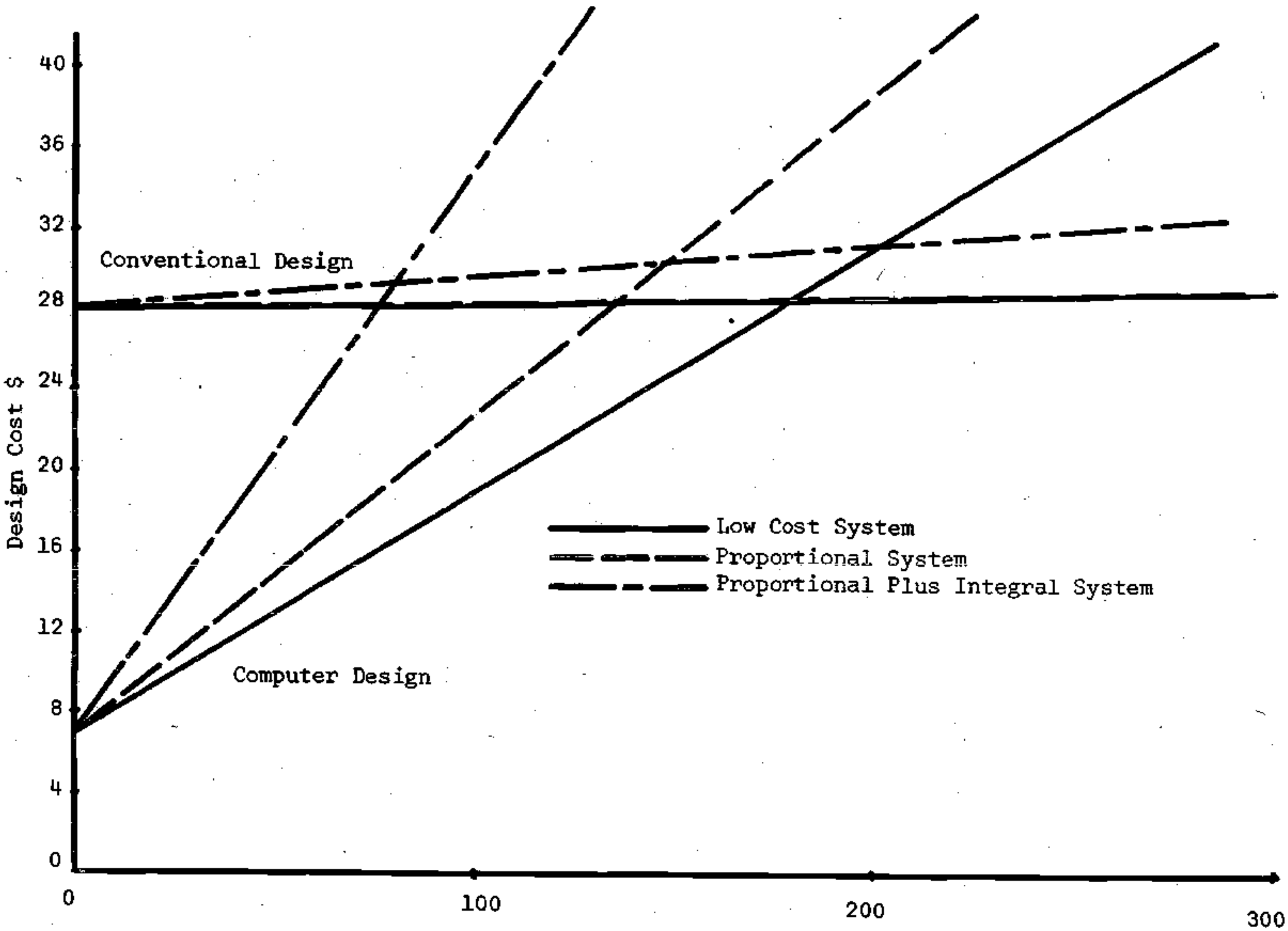


Figure 4. Plot of Design Cost Versus Run Time.

ential equations, the use of any numerical method which requires less computer time than the Runge-Kutta method will make the computer design more competitive.

Because of the lack of clear cut savings in design cost and because of the high program development cost, any justification for computer design of tank liquid level control system must come from consideration of advantages of the computer design not available with the conventional design. That there are several of these advantages will be brought out in Chapter V. Unfortunately, it is extremely difficult to place these advantages on a numerical basis.

CHAPTER V

CONCLUSIONS

Based only on comparative values of cost per design for the two design methods, the computer design cannot compete with the conventional design if the numerical integration method in the present program is used.

This conclusion must be tempered by consideration of other information. Although a large number of levels of design sophistication are possible, all liquid level control systems can be divided into three categories. The first category contains control systems that are only required to maintain a liquid level within a fairly wide band. On-off and proportional controllers are used in this category. The second category is made up of systems that maintain the liquid level fairly accurately, say within a band of one inch or less. The third and most sophisticated of the level control design categories contains control systems for which the transient response as well as close maintenance of the required liquid level is important. Proportional and proportional plus integral controllers are used for the last two categories.

The conventional design method is much less sophisticated than the computer design method. Liquid level control systems are not considered as systems when designed conventionally. The control valve is sized on the basis of the estimated flow necessary to maintain the desired liquid level. The assumptions of steady flow and constant CV are made.

Controller selection is made strictly on the basis of past experience. If no stand-off error is allowable, it is known that a proportional plus integral controller must be used. If some stand-off error is allowable, the applications engineer must decide whether to use a low cost proportional system, with its limited proportional band range, or a general proportional system.

The average applications engineer is not concerned with determining rise time, amount of overshoot, or settling time. He hopes that the system can be tuned to meet the desired criteria after it is installed in the field.

The computer solution considers the level control system as a system; the group of differential equations representing the control system are solved simultaneously. The control valve is sized on the basis of required rise time and the computer solution takes unsteady flow and the variation of CV with Reynolds number into consideration. The system response is simulated on the computer. Therefore, a system can be selected that will meet any reasonable rise time, overshoot, settling time, and/or steady state error requirements. Since potential systems are tested in order of increasing system cost, the customer is guaranteed that he isn't buying more control system than he needs. Very little time is required to select gains after the system is installed, since approximate gains are obtained from the computer simulation.

Comparison of the three levels of design sophistication mentioned earlier with the level of sophistication available from the two design methods covered here reveals that the conventional design is all that is required for the first category. Customers interested in control systems in this category are not the least bit interested in rise time and overshoot and there is generally little question about what type of controller to use. The computer design method becomes more attractive if the level of design sophistication is that of category two; not because of any need to find approximate gain settings or any interest in rise time, overshoot, or settling time, but because the computer design guarantees that the least expensive system will be selected. If there is any question at all in the applications engineer's mind about whether to use a proportional or a proportional plus integral system because of a small allowable steady-state error, he will call for a proportional plus integral controller. This is because he never knows exactly what steady-state error the proportional system will have until the system is installed. The customer would be extremely unhappy if the control system had been installed and it could not meet his steady-state error requirements. No problem like this would arise if the system were simulated on the computer. The proper controller can be selected with a high degree of accuracy. System cost for a low cost proportional system can be as much as $100 less than the cost of a proportional plus integral system.

If the level of design sophistication required corresponds to Category Three, there is no guarantee that the conventional design method will be entirely satisfactory in many cases. If the level control system is part of a large process, the prospect of tuning the level control system after installation is not pleasant, particularly when adjusting the level control system's gains might necessitate recalibration of the rest of the process. Total system characteristics cannot be determined until the system is installed if the conventional design method is used.

There has been considerable interest generated recently about computer control of processes. Conventional design of control systems included in larger systems controlled by computer is out of the question. Accurate data on system response is necessary for the design of the computer control circuits.

It appears that computer design can only compete with conventional design if the level of sophistication required is that of Category Two or Three. Discussions with manufacturer's representatives indicate that the bulk of the liquid level control system market is in the first category. Although the demand for systems of the second and third level of design sophistication is increasing, it is felt that this demand will fall far short of the number of systems required to justify the large development cost for a program of this nature.

Reference to an on-off control system can be found throughout this study. The original study outline called for the inclusion of an on-off, a proportional, and a proportional plus integral control system in the computer program.

When an example problem using the on-off control system was introduced into the program, the Runge-Kutta method of solution of the equation representing flow into the tank required a time increment of 0.001 seconds for stability when the control valve was in the closed position. Calculation of response curve points for any normal liquid level control system would be extremely expensive with this small time increment. Since the comparison of available levels of sophistication for the computer and the conventional design methods and the required level of design sophistication for the three categories listed above indicates that on-off control system design by computer is not feasible, it was decided to discontinue development of the on-off control section of the computer program.

Weighing these considerations with the original conclusion; that on a cost per design basis the computer design cannot compete with the conventional design, can only lead to the conclusion that computer design of liquid level control systems using a digital computer program of the type developed for this study is not economically feasible.

If the computer run time could have been reduced by a factor of ten, use of such a computer program might become feasible on a cost per run basis. Three possible methods of decreasing the time required to obtain system response are discussed in Chapter VI

Based on the results of this study, component selection by computer for certain classes of components can be profitable. Honeywell's method of selecting components and finding their cost involves searching through two different sets of books, one listing the components and their characteristics and the other giving their cost. It is estimated that component selection for the program included in this study took approximately one minute of computer time. Equivalent manual selection would have taken approximately one hour. Only classes of components which can be selected on a logical basis would yield such savings. Control valves are excellent examples of this type of component.

Control system design by computer requires a knowledge of the process dynamic equations. The number of assumptions required in developing the flow equations for the simple system used in this study brought out the lack of basic knowledge about transient flow through valves and fittings.

A systematic method of gain adjustment must be available if the computer is to determine acceptable system response for the simulated control system without adjustment between runs by the system designer. Gain adjustment for the liquid level system used in this study was no problem, but had a control system which included a rate mode been selected, a fairly elaborate trial and error method of gain variation would have been required in the computer program.

Even if the results of this study had indicated that a design program of this nature would have been slightly profitable, it might not be a wise idea to develop such a program for production. Customers for control systems of the type used as an example in this study are extremely conservative. An interview with an instrument engineer for a local consulting engineering firm revealed that his company's policy is to never incorporate any new design features into their designs until these new features have been proven in someone else's installation. Any cost advantage due to computer design can be lost if upon completion of the design, the customer wanted to increase the control valve diameter one size to insure that he will have adequate flow capacity.

Many control systems are sold on the strength of considerations which cannot be evaluated on a numerical basis. One such consideration is the availability of maintenance and replacement parts. The sales ability of the applications engineer and the personal preference of the customer or his representative are two large considerations since purchases in the price range of most of the control systems considered here ($1,000 or less) do not have to be justified on a comparative cost basis to management.

CHAPTER VI

RECOMMENDATIONS

It has been found (see Chapter V) that if the time required for the computer to make the calculations necessary to plot the system response curve could be cut by a factor of ten, the computer design would be competitive with the conventional design on a cost per run basis. Three alternate approaches to reducing the computer time required per design are presented here and are recommended as areas in which further profitable research might be done.

The first approach concerns using a faster numerical integration technique for the response curve determination. One candidate for this application would be the modified Euler method. If the modified Euler method were substituted for the Runge-Kutta method, and only two cycles were required per point on the curve, the computer time required would be one half that required for the Runge-Kutta method, which requires four cycles per point.

It is believed that a second approach, concerned with utilization of analog computational techniques, promises greater reductions in computer time required for control system design. In general, the digital computer is slower and more accurate than the analog computer. The value of this increased accuracy is questionable in determination of control system response in many cases. On the other hand, component selection and involved decision making is impossible on the analog computer. These facts lead one to consider the possible applications of a hybrid digital-analog computer.

A large amount of research is now being directed toward determination of optimization methods. One approach to the determination of optimum gain settings for control systems involves trial and error methods of gain adjustment. The systematic method of gain variation used in the trial and error approach could be programmed into the digital part of a hybrid computer. The digital section would then control the gain settings of the analog part of the hybrid computer on which a large number of trial response curves could be run in a much shorter time than on a digital computer alone.

The third method of attempting to reduce the required computer time is suggested by the results of the tests of the system equations in Appendix B. It can be seen that for the chosen example and the short run time, the approximate method in which pipe friction is ignored resulted in values within 5 per cent of those found by the more rigorous computer solution. Further research could be directed toward determining the applicability of approximate expressions for fluid flow that could be solved in less computer time than the expressions used in this study.

The need for a large amount of experimental work in the field of transient fluid flow, particularly in connection with flow through valves and fittings was demonstrated by the number of assumptions required in the development of the flow equations used in this study. Several articles have been written on control valve selection, but none is satisfactory from an unsteady flow standpoint. Each of the flow assumptions mentioned above must be justified if the computer solution is to have any practical value. The use of system engineering concepts in process work requires much stronger definitions of some of the basic flow processes than are now available.

APPENDIX

APPENDIX A

SYMBOL TABLE

A	Number of fluid as read from compatibility table on input tape
A8	Area of tank
CAPCTY	Code indicating whether capacity is full or reduced (1--full; zero--reduced)
CHANGE	Value of sign of dH/dT at T-ΔT
COMPAT	Number of control valves which meet system requirements
COMPONENT	Type of low-cost system (A or B)
CONFIG	Code indicating whether system is of configuration 1 or 2 (1--flow out of top of tank; 2--flow out of discharge pipe)
COST	Temporary storage for cost of an item used during printing
CR	Control range
CV1	Value of valve size coefficient at a given Reynolds number (inlet side)
CV3	Value of valve size coefficient at a given Reynolds number (outlet side)
CVCN1	Catalog valve size coefficient (inlet side)
CVCN3	Catalog valve size coefficient (outlet side)
CVMAX	Maximum allowable C_v of control valve
CVMIN	Minimum allowable C_v of control valve
CVO	Outlet valve size coefficient
D1	Pipe diameter on inlet side
D3	Pipe diameter on outlet side
D8	Diameter of circular tank

P	Pressure drop
DELTPI	ΔP required by system
DELTT2	Time increment used in checking system response
ΔH DH	Desired band width x (1/2)
DIFF	H1-initial value of H
DIAP1	Parameter used in Runge-Kutta routine to represent D1
DIAP3	Parameter used in Runge-Kutta routine to represent D3
DIAV1	Parameter used in Runge-Kutta routine to represent diameter of control valve
DIAV3	Parameter used in Runge-Kutta routine to represent diameter of valve on outlet side
DT	ΔT used in Runge-Kutta
DUM	Dummy variable used to store absolute values
DV1	Control valve diameter
DV3	Valve diameter on outlet side
DX1	Used as an index
E	A measure of pipe roughness
ESS	Allowable steady state error
EXT	Code indicating whether extension column is required or not (1--yes; zero--no)
F2	Inlet pipe friction factor
F4	Outlet pipe friction factor
FAC0 FAC1 FAC10 FAC2 FAC20	Constant factors used in Runge-Kutta routine

FAC3 FAC30 FAC300 FAC33 FAC4 FAC40 FAC5 FAC50 FAC6	Constant factors used in Runge-Kutta routine
FLOUT	Constant flow out
FMAX	Factor by which catalog valve size coefficient (inlet) is multiplied to get CVMAX
FMIN	Factor by which catalog valve size coefficient (inlet) is multiplied to get CVMIN
G	Specific weight of fluid
H1	Desired level in tank
H1X2	H1-H(t) at a given time
H2	See Figure 3
H3	See Figure 3
I	Used as an index
IND	Used as an index
INITL	Initial value of an index used in a program loop
IFINAL	Final value of an index used in a program loop
J	Used as an index
J1	Used as an index
K	Used as an index
K1	Used as an index
K20	Measure of degree of opening of outlet valve
L	Used as an index
L1	Height of tank
M	Used as an index

MEDM	Number of fluid desired by customer
MONEL	Code indicating whether a monel bellows seal can be used by the system (1--yes; zero--no)
N	Number of differential equations being solved in Runge-Kutta routine
NEXTV	Code indicating whether next cheapest valve is to be checked for response characteristics or not (1--yes; zero--no)
NMRTR	Storage for numerator of initial CV equation
NVALVS	Total number of valves in valve table on input tape
ONOFF	Code indicating whether system is on-off or not (1--yes; zero--no)
OP	Code indicating whether manual operator is desired or not (1--yes; zero--no)
OS	Maximum allowable overshoot
OSOK	Code indicating whether overshoot has been checked or not (1--yes; zero--no)
P	Used as a factor in Runge-Kutta routine
PO	See Figure 1
P1	See Figure 1
P1MAX	Maximum value of P1
P4MIN	Minimum value of P4
P7	See Figure 1
P8	See Figure 1
P13	Pressure for regulated tank
P15	Allowable system pressure
P30	Allowable system pressure
P40	Allowable system pressure
P60	Allowable system pressure

PB	Proportional band
PBMAX	Limit on proportional band
PLAIN	Code indicating whether plain bonnet is required or not (1--yes; zero--no)
PNEU	Code indicating whether system is pneumatic or not (1--yes; zero--no)
QI	Flow in
QO	Flow out
QOM	Maximum flow out
QOTIM	Parameter used in Runge-Kutta to represent closing time of a quick-opening valve
QOT	Quick opening time
R	Valve rangeability Maximum controllable flow/Minimum controllable flow
RADFIN	Code indicating whether radiating fin is required or not (1--yes; zero--no)
RD	r/d for elbows
RE1	Storage for inlet pipe Reynolds number or control valve Reynolds number, depending upon whether valve has been chosen or not
RESTART	Parameter used in Runge-Kutta to indicate if next cheapest valve is to be checked for adequate response
RP1	Inlet pipe Reynolds number
RP3	Outlet pipe Reynolds number
RUNGKUTTA	Name of Runge-Kutta subroutine
RR	Integral gain, repeats per second
RV1	Control valve Reynolds number
RV3	Outlet valve Reynolds number
SAUNDERS	Code indicating whether Saunders valve is to be considered or not (1--yes; zero--no)

SEALI — Code indicating whether bellows seal is required or not (1--yes; zero--no)

SETTLNG — Code in Runge-Kutta indicating if settling time has been checked (1--yes; zero--no)

SLURRY — Code indicating whether fluid is a slurry or not (1--yes; zero--no)

SPECIAL — Code indicating whether special transducer is to be ordered or not (1--yes; zero--no)

STATIC — Code indicating if static pressure sensor is desired (1--yes; zero--no)

STEEL — Code indicating whether 316 stainless steel bellows seal can be used or not (1--yes; zero--no)

STOPX — Code indicating whether program is to be stopped at this point (1--yes; zero--no)

SUML — Sum of equivalent lengths of fittings in a length of pipe

T — Accumulated time (seconds) in Runge-Kutta

T1 — Time accumulated during calculation of on-off XT in Runge-Kutta

T3 — Time accumulated during checking of settling time during the Runge-Kutta

T4 — Time accumulated during checking of steady state error in Runge-Kutta

T7 — Temperature of fluid

TANK — Code indicating whether tank is vented, regulated, or trapped (1--vented; 2--regulated; 3--trapped)

TLAST — Time which is equal to present time minus DT in the Runge-Kutta

TNKRATNG — Tank rating

TOP — Code indicating whether a manual operator is mounted in top or side position (1--top; zero--side)

TOTAL — Total cost of a system in printing routine

TRO — Rise time or difference between TR(2) and TR(1) in Runge-Kutta

TRR	Required rise time
TSETR	Required settling time
U	Fluid viscosity
V1	Motor voltage used computing XT for on-off system
VLAST	Previous motor voltage
W1	See Figure 1
W2	See Figure 2
X1M	Design flow in
XT	x/X, per cent opening for control valve used in program
Z	Computer time stored at the beginning of the program

ARRAYS:

ACTCOST	Valve actuator or motor cost
ACTNO	Valve actuator or motor number
AIR	Air-to-diaphragm value of a valve
AVAIL	Number of a valve that is usable in the system so far
AVAIL	Temporary storage for AVAIL
B	Number of material that is compatible with fluid
BCOST	Cost of valve body material
BODY	Number of available body materials for a valve
BONCOST	Valve bonnet costs
CHEAP	Number of a valve that has been inserted in the list of valves in order of cost
COL	Location of the applicable CV of a valve
COSTX	Temperature storage for cost of valve body material
CV	Valve CV
CX	Factor used in computing rise time in Runge-Kutta

D	Derivative used in Runge-Kutta
DELTP	ΔP of valve
DISCOST	Level sensor cost
DISPL	Control range of level sensor
EXT	Code indicating whether valve has extension column or not
INDX1	Row location of least expensive body material and rating of a valve
INDX2	Column location of least expensive body rating of a valve
LD	Equivalent length of a fitting
LE	Equivalent length of a length of pipe
LIMVFC	Location of last available v-port CV of a full capacity valve
LIMVRC	Location of last available v-port CV of a reduced capacity valve
LT	True length
MAN	Row location of applicable valve manual operator cost
MCOST	Cost of a valve manual operator
MODEL	Model number of a valve
NAME	Alphanumeric name of a body or trim material
NB	Valve body material
NBI	Required body material
NBON	Body material corresponding to a valve bonnet cost
NBX	Temporary storage for cheapest valve body material
NFIT	Number of a type of fitting in one length of pipe
NT	Valve trim material
NTI	Required trim material
NTX	Trim material compatible with fluid
NUMCVS	Total number of available CV's of a valve

OMIT	Number of valves omitted from list of usable valves
ORIGX	Initial value of an element in the X array
POS	Location of a positioner on a valve
POSCOST	Cost of a positioner on a valve
QO	Code indicating whether valve is quick-opening or not
QOLINK	Linkage number of a quick-opening valve
QOTIME	Closing time for a quick-opening valve
RAD	Code indicating whether valve has radiating fin available
RATNG	Body rating of valve
RATNGI	Required body rating
RATNGX	Temporary storage for rating of cheapest body material
SEAL	Code indicating whether valve has bellows seal or not
SIZE	Diameter of a valve
SPRMAX	Maximum pressure to be balanced by valve actuator spring
SPRMIN	Minimum pressure to be balanced by valve actuator spring
TMAX	Maximum allowable ambient temperature for valve
TMCOL	Location of cheapest trim material of valve
TMCOST	Cost of a valve trim material
TMIN	Minimum allowable ambient temperature for valve
TOTCOST	Total cost of a useable valve
TR	Time stored for use in calculating rise time in Runge-Kutta
TRAVL	Valve travel
TSET	Time stored for use in calculating settling time
TYPE	Valve type
VPFC	Number of CVs available with v-port characteristics in a full capacity valve

VPLINK	Linkage number of a valve with v-port characteristics
VPORT	Code indicating that valve has v-port characteristics
VPRC	Number of CVs available with a reduced capacity valve with v-port characteristics
VPTIME	Closing time of a valve with v-port characteristics
X	Solution of differential equation in Runge-Kutta
XLAST	Solution at T-ΔT of differential equations in Runge_kutta

APPENDIX B

DERIVATION OF SYSTEM FLOW EQUATIONS

Two tank configurations are utilized in this study. The first, Configuration One, represents a system containing a tank discharging from the top. Configuration Two, shown in Figure 1 which is repeated here for convenience, contains a tank which discharges thru an outlet pipe. Configuration Two is illustrated because it is the more general of the two. For the same reason equations of motion will be developed for Configuration Two. Flow into the tank and the liquid level in the tank of Configuration One can be represented by the corresponding equations for Configuration Two if flow from the tank, Q_o, is assumed constant or varies only by step changes.

The tank shown in Figure 1 is typical of those found in industry. It has vertical sides and a round or a rectangular cross section. All pipes are assumed level, and L2, L3, L4, and L5 are equivalent lengths made up of the true pipe length and the equivalent length of any bends or fittings included in that run of piping. It is assumed that flow will never reverse direction in the systems covered by this derivation. To insure fully developed flow, all sections of pipe must have an L/D ratio greater than 20, where L is the pipe length and D is the inside diameter of the pipe.

The definitions of the symbols used in the following derivations are found in the Nomenclature Table.

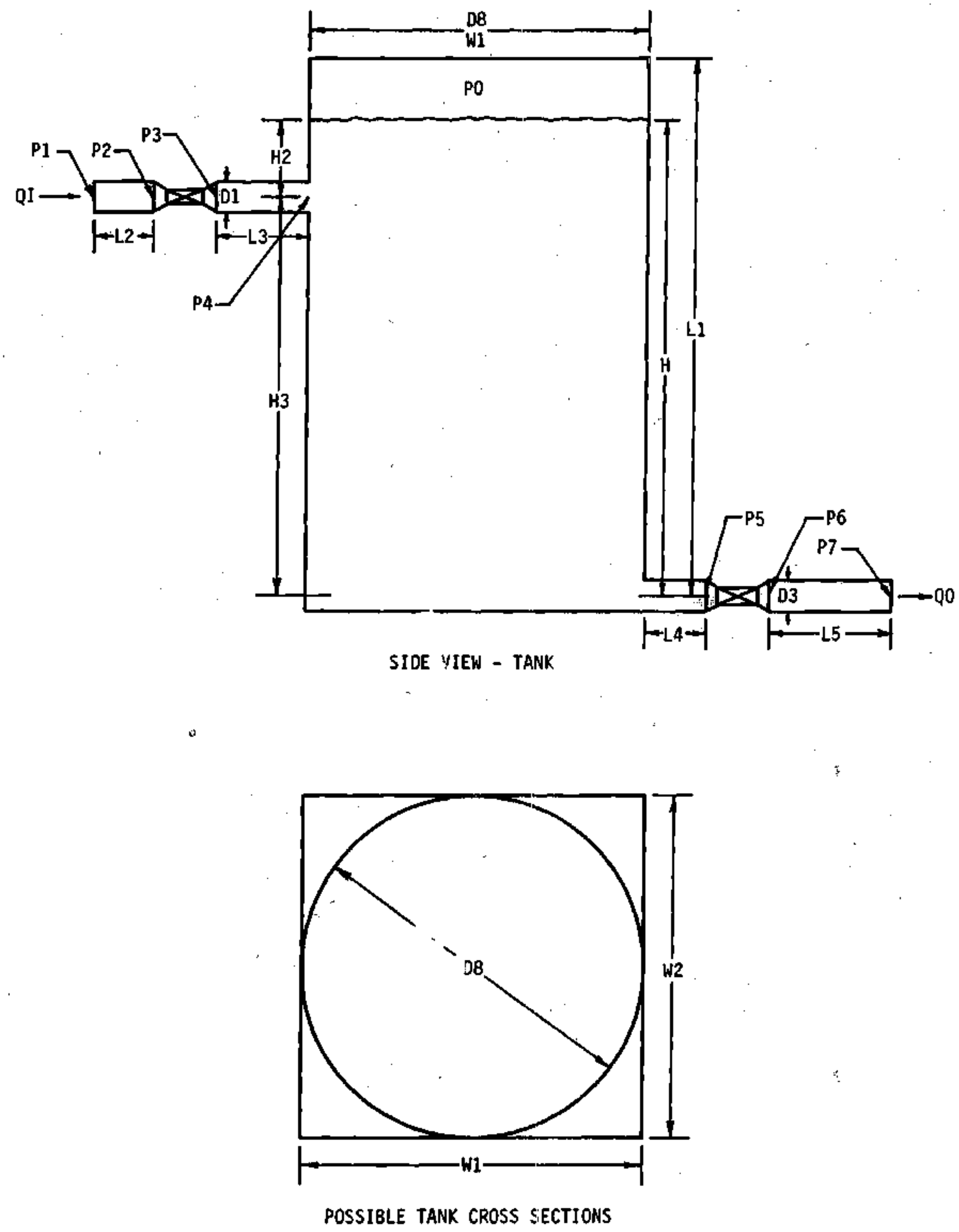


Figure 1. Diagram of Configuration Two.

According to Eckman (10), the flow rate of a liquid through a fully opened control valve is given by

$$Q = CV \sqrt{\frac{\Delta P}{S.G.}} \tag{4}$$

The specific gravity uses the specific weight of water at 60°F as the reference specific weight.

$$S.G. = w/w_w \tag{5}$$

The valve size coefficient CV is defined as the flow rate of water in gallons per minute provided by a pressure differential of one pound per square inch through a fully opened control valve.

Forman (11) recommends the equal-percentage spool characteristic for control valves used in most process situations, primarily on the basis of its good rangeability. Therefore, only control valves with equal-percentage characteristics will be used in the throttling systems of this study. With an equal-percentage spool characteristic, each increment of plug movement produces a change in flow which is proportional to the amount flowing before the change occurred. Rangeability is the ratio of maximum controllable flow to minimum controllable flow. The flow through a sliding stem equal-percentage control valve is given by Eckman (12) as

$$Q/Q_m = R^{(x/X-1)} \tag{6}$$

A plot of flow versus valve lift for an equal-percentage valve is given in Figure 5.

The flow-lift characteristic for Honeywell quick-opening control valves was not available at the time of this writing. For lack of better information, the quick-opening valves were assumed to have approximately linear characteristics described by Eckman (13) as

$$Q/Q_m = 1/R\ [1 + (R-1)x/X] \tag{7}$$

Let $Q/Q_m = KI$. Combining KI, Equation (4), and Equation (5) results in the following expression for flow through a control valve at any stem position.

$$Q = (KI)CV\sqrt{\frac{w_w \Delta P}{w}} \tag{8}$$

Applying this equation to the control valve on the inlet side of the tank in Figure 1 yields

$$Q_I = (KI)CVI\ \sqrt{\frac{w_w \Delta P_I}{w}} \tag{9}$$

Forman (14) states that control valves are sometimes one pipe size larger or smaller than the pipe in which they are connected. In these cases, reducers are used between the pipe and the control valve. If the friction loss in the reducers is neglected,

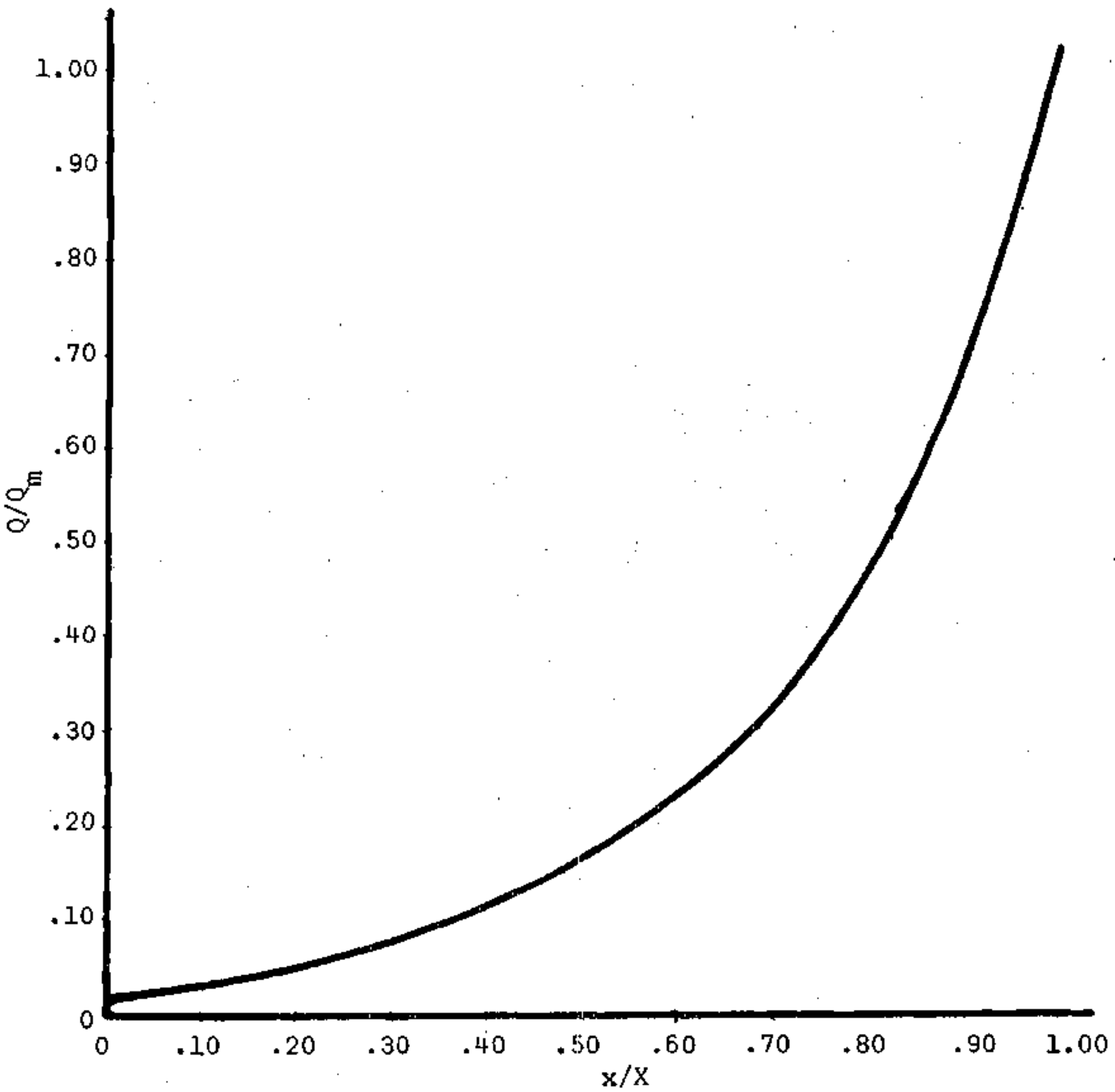


Figure 5. Flow Versus Valve Lift for an Equal-Percentage Control Valve.

$$\Delta P = P2 - P3 \tag{10}$$

It can be shown (15) that the momentum equation for fluid flow applied to a control volume R, bounded by a surface S, is

$$\Sigma \vec{F} = \frac{1}{g_c} [\frac{d}{dt} \int_R p_R \vec{V}_R d\vec{R} + \int_S (p_S \vec{V}_S) \vec{V}_S \cdot d\vec{S}] \tag{11}$$

where $\frac{d}{dt} \int_R p_R \vec{V}_R d\vec{R}$ is the time rate of change of momentum of the fluid within R and bounded by S and $\int_S (p_S \vec{V}_S)\vec{V}_S \cdot d\vec{S}$ is the time rate of net efflux of momentum across the control surface S. Since the flow is uniform and since $\vec{V}_1$ equals V_2 which equals $\vec{V}_R$ (the velocity in the region) if $\vec{V}_R$ is set equal to $\vec{V}$,

$$\Sigma \vec{F} = \frac{1}{g_c} [\frac{d}{dt} \int_R P_R V_R d_R + \dot{m}_2 \vec{V}_2 - \dot{m}_1 \vec{V}_1] \tag{12}$$

Since $\vec{V}_1 = \vec{V}_2 = \vec{V}_R$,

$$\Sigma \vec{F} = \frac{1}{g_c} [\frac{d}{dt} (M\vec{V}_R)] \tag{13}$$

Because the flow is incompressible,

$$\Sigma \vec{F}_E = M \frac{dV}{dT} \tag{14}$$

Three external forces act on the fluid flowing in the "L2" section of the pipe as shown in Figure 6. Since all the forces act along the same axis, the vector notation can be dropped and

$$\Sigma F_E = M \frac{dV}{dt} \tag{15}$$

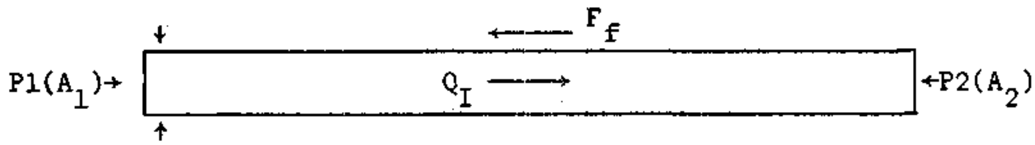


Figure 6. Free-Body of Fluid in Pipe "L2."

The friction force, F_f, is shown in Figure 6. Assuming that resistance to flow in unsteady flow is equal to steady flow resistance at the same velocity (16), the head loss due to resistance is

$$h_f = \frac{FLV^2}{2g_D} \tag{16}$$

For the case in Figure 6,

$$V = \frac{Q_I}{A_I} \tag{17}$$

where $A_1 = \frac{\pi}{4} D_1{}^2$.

The friction force F_f is the product of the head loss, the fluid specific weight, and the cross sectional area of the pipe as expressed by the equation

$$F_f = wA_1h_f \tag{18}$$

Substituting Equation (16) and Equation (17) into Equation (18) yields

$$F_f = \frac{wF2(L2)Q_T^{\,2}}{2gD_1A_1} \tag{19}$$

for the pipe in Figure 6.

The mass of the fluid in the pipe, M, is given by

$$M = \frac{w(L2)A_1}{g} \tag{20}$$

Since

$$V = \frac{Q_1}{A_1}$$

$$\frac{dV}{dt} = \frac{1}{A_1}\frac{dQ_I}{dt} \tag{21}$$

Substituting Equations (19), (20), and (21) into Equation (15) and rearranging results in

$$P2 = P1 - \frac{wF2(L2)}{2gD_1\ A_1^{\,2}}Q_I^{\,2} - \frac{w(L2T)}{gA_1}\ \frac{dQ_I}{dt} \tag{22}$$

Likewise, for pipe length L3,

$$P3 = P4 + \frac{wF3(L3)}{2gD_1\ A_1^{\ 2}} Q_I^{\ 2} + \frac{w(L3T)dQ_I}{A_1 g\ dt} \qquad (23)$$

Combining Equations (22) and (23) and assuming the friction factor in L2 is equal to the friction factor in L3,

$$P2 - P3 = P1 - P4 - \frac{wF2}{2D_1 gA_1^{\ 2}} [L2 + L3]Q_I^{\ 2} - \frac{w}{gA_1} [L2T + L3T] \frac{dQ_I}{dt} \quad (24)$$

As can be seen from Figure 1, P4 may be made up of two components: the pressure due to a head of liquid above the pipe inlet in the tank, and the pressure of the gas above the liquid level. Three conditions are covered here. The tank can be vented, i.e., the pressure above the liquid level is taken as 15 psia; the pressure above the liquid in the tank is regulated or maintained at a constant pressure P13; or, the gas above the liquid level is trapped so that the pressure above the liquid is a function of the level. Representing P4 mathematically results in

$$P4 = P0 + (H2)w \qquad (25)$$

where

$$P0 = 15 \text{ psi} \qquad \text{Vented}$$

$$= P13 \qquad \text{Regulated}$$

$$= \frac{(P00)L1}{L1 - H} \qquad \text{Trapped}$$

H2 is defined as follows.

$$H2 = \begin{cases} 0 & H \le H3 \\ H - H3 & H > H3 \end{cases} \quad (26)$$

The expression for PO for the trapped tank is found as follows. It is assumed that the empty tank is charged to a pressure, POO. Using the perfect gas law,

$$(POO)V_{oi} = NR_cT_i \quad (27)$$

where i subscripts indicate initial conditions, and

$$(PO)V_o = NR_cT \quad (28)$$

Since the tanks covered by this derivation will generally not be insulated, it is assumed that the gas above the liquid will not experience any appreciable temperature change during level changes and the expansion and contraction of this gas will therefore be assumed isothermal. Therefore,

$$(PO)V_o = (POO)V_{oi} \quad (29)$$

where

$$V_o = A_8[L1 - H] \tag{30}$$

and

$$V_{oi} = A_8(L1) \tag{31}$$

Substituting Equation (30) and (31) into Equation (29) yields

$$PO = \frac{(POO)L1}{L1 - H} \tag{32}$$

Substituting Equation (24) into Equation (9) and rearranging yields the following equation for flow into the tank,

$$\frac{dQ_I}{dt} = \frac{gA_1}{(L2T + L3T)} [\frac{P1 - P4}{w} - [\frac{1}{((KI)CVI)^2 w_w} + \frac{F2(L2 + L3)}{2gD_1(A_1)^2}]Q_I{}^2] \tag{33}$$

The equation to be used for the friction factor, F2, depends on the Reynolds number for flow in the pipe where

$$Re = \frac{4Q_t w}{D_1 Ug}$$

For Re $\leq$ 2000,

$$F = \frac{64}{Re} \tag{34}$$

Therefore,

$$F2 = \frac{16_{\pi} D_1 Ug}{Q_I w} \tag{35}$$

From Potter (17), if the Reynolds number is greater than 4000, the friction factor F can be found from the equation,

$$F = 0.0055 \left[1 + \left(20{,}000 \frac{E}{D} + \frac{10^6}{Re}\right)^{\frac{1}{3}}\right] \tag{36}$$

Assuming that the friction factor corresponding to Reynolds numbers in the range between 2000 and 4000 can be found by Equation (36). For Re > 2000,

$$F2 = 0.0055 \left[1 + \left(20000 \frac{E}{D_1} + \frac{10^6}{Re}\right)^{\frac{1}{3}}\right] \tag{37}$$

As shown in Chapter IV, control valve selection is based in part on a valve size coefficient, CV. This size coefficient is a measure of the flow rate through a control valve as a function of the fluid and pressure drop across the valve. The CV is determined for the flow system under consideration by the following relationship

$$CV = \frac{Q_I}{\sqrt{\frac{\Delta P}{S.G.}}}$$

The size and type of control valve required to meet a specific application can be selected on the basis of a comparison of this calculated CV and the ta bulated CV's found in the valve manufacturer's literature. The valve size coefficients given in the manufacturer's literature have been determined experimentally for each type and size valve.

Although the valve size coefficient is generally given as a constant in the manufacturer's catalogs, in reality, its value varies somewhat with Reynolds number. Even though manufacturer's information on this variation was not available at the time of this writing, a correction to be applied to the manufacturer's constant CV values is included in the computer program for demonstration purposes. The variation of CV with Reynolds number was assumed to be that of a 200 x 400 Venturi meter as given by the curve in the 4th edition of "Fluid Meters: Their Theory and Application" published by the American Society of Mechanical Engineers in 1937. This curve should be replaced with the manufacturer's curves of CV versus Reynolds number for various valves when they are available if the computer program is to be used in practice. The 200 x 400 Venturi meter curve is approximated by straight line segments as shown in Figure 7. In Figure 7, CV is the manufacturer's published valve size coefficient, CVI is the valve size coefficient corrected for the effect of the Reynolds number, and the Reynolds number is the Reynolds number for flow through the valve.

From Figure 7, the following relationships were found. (The program is not applicable for RE < 200). For,

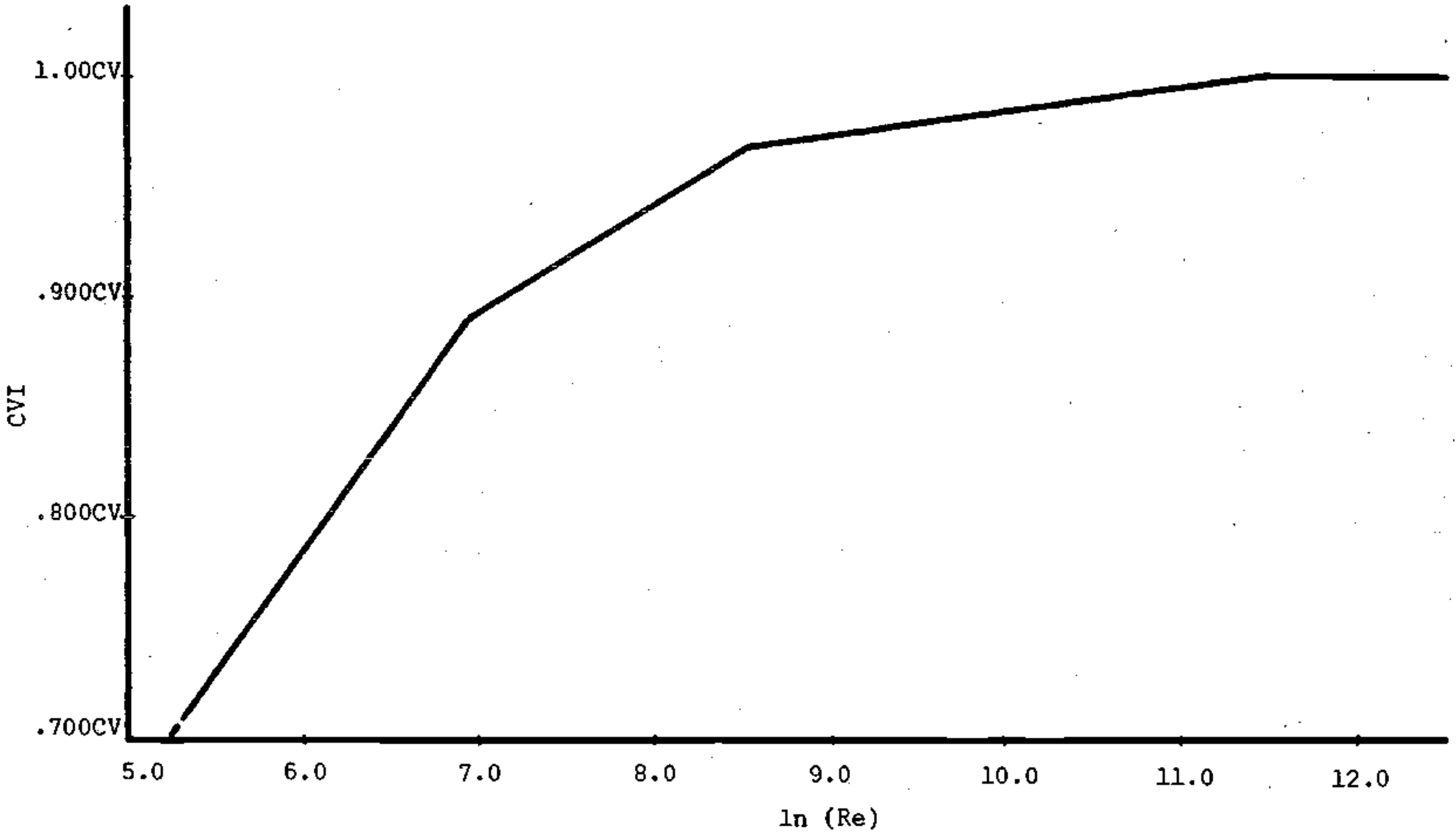


Figure 7. Straight-Line Approximation of Assumed Variation of CVI with Reynolds Number.

$200 \leq Re \leq 1{,}000$	CVI=[0.707+0.11 (ln Re-5.30)]CV	(38a)
$1{,}000 \leq Re \leq 5{,}000$	CVI=[0.884+0.0398 (ln Re-6.91)]CV	(b)
$5{,}000 \leq Re \leq 100{,}000$	CVI=[0.948+0.01734(ln Re-8.52)]CV	(c)
$100{,}000 \leq Re$	CVI=(1.00)CV	(d)

The pipe lengths used in the terms containing the friction factors in Equation (33) correspond to the true length of the pipe plus the equivalent length of any bends or fittings included in that length of pipe. For the purpose of the described study, it is assumed that the head loss effect of various fittings is the same in transient flow as in steady flow. It is felt that this assumption is justified for a feasibility study of this sort, but, in practical usage, the error caused by making this assumption should be determined. Table 4 contains a list of various valves and fittings covered by the program along with their equivalent lengths in pipe diameters. This list is taken from Crane (18). The equivalent length for a run of pipe is found by summing up the equivalent lengths in pipe diameters for all the fittings included in that run of pipe and multiplying that sum by the nominal pipe diameter. The equivalent length of a 90° pipe bend must be found from a "Chart for Resistance of 90 Degree Bends" on page A-27 in Crane. Assuming r/d will be greater than or equal to 3.0, but less than 20, the curve has been approximated by a polynomial curve fit computer program which resulted in the following equation,

$$L/D = 5.9997 \times 10^{1} - (5.9953 \times 10^{1})r/d + (2.8896 \times 10^{1})(r/d)^{2} \quad (39)$$

$$- (7.2291)(r/d)^{3} + (1.0614)(r/d)^{4} - (9.3532 \times 10^{-2})(r/d)^{5}$$

$$+ (4.8507 \times 10^{-3})(r/d)^{6} - (1.3621 \times 10^{-4})(r/d)^{7}$$

$$+ (1.5959 \times 10^{-6})(r/d)^{8}$$

If the tank is set up as in Configuration Two, the equation for flow from the tank is found in much the same way as that found for flow into the tank. This study assumes that K20, the opening of the outlet valve is a known constant greater than or equal to zero, but less than or equal to one. Then from Equation (8),

$$Q_o = (KO)CVO\sqrt{\frac{w_w \Delta P}{w}} \quad (40)$$

It is also assumed that the outlet valve CV varies with Reynolds number as shown in Figure 7, and that friction in the reducers can be neglected.

$$\Delta PO = P5 - P6 \quad (41)$$

Substituting values in Equation (15) yields the following equations for P5 and P6,

$$P5 = PO + Hw - \frac{wF4(L4)}{2g\ D_3(A_3)^2} Q_o^{\ 2} - \frac{w(L4T)}{gA_3} \frac{dQ_o}{dt} \tag{42}$$

$$P6 = P7 + \frac{wF4(L5)}{2g\ D_3(A_3)^2} Q_o^{\ 2} + \frac{w(L5T)}{A_3 g} \frac{dQ_o}{dt} \tag{43}$$

Combining Equations (40), (41), (42), and (43) results in

$$\frac{dQ_o}{dt} = \frac{gA_3}{(L4T+L5T)} \left[\frac{PO-P7}{w} + H - \left[\frac{1}{w_w((KO)CVO)^2} + \frac{F4(L4+L5)}{2\ D_3(A_3)^2 g}\right] Q_o^{\ 2}\right] \tag{44}$$

where F4, L4, L5, and CVO are found as on the inlet side.

The change in level in the tank is found by the following equation,

$$\frac{dH}{dt} = \frac{1}{A_8}(Q_I - Q_o) \tag{45}$$

Tests for the validity of the system flow equations derived above for use in the computer program were developed and are described on page 72.

Certain simplifying assumptions can be made to obtain approximate expressions for the time rate of change in head in the tank, for flow into the tank, and for flow from the tank. If the tank outlet valve is shut, ($Q_o = 0$) and the control valve is held open (KI = 1.0), an

Table 4. Equivalent Lengths of Valves and Fittings

Type	Sub-Type	VALVES AND FITTINGS Description	Equivalent Lengths L/D
Globe Valves	Conventional	With no obstruction in flat, bevel, or plug type seat.	Fully open 340
		With wing or pin guided disc.	Fully open 450
	Y-Pattern	(No obstruction in flat, bevel, or plug type seat.)	
		Stem 60 degrees from run of pipe line.	Fully open 175
		Stem 45 degrees from run of pipe line.	Fully open 145
Angle Valves	Conventional	With no obstruction in flat, bevel, or plug type seat.	Fully open 145
		With wing or pin guided disc.	Fully open 200
Gate Valves	Conventional		Fully open 13
Check Valves		Conventional swing.	Fully open 135
		Clearway swing.	Fully open 50
		Globe lift or stop.	Fully open 450
		Angle lift or stop.	Fully open 200
		In-line ball.	Fully open 150
Fittings		90 Degree standard elbow.	30
		45 Degree standard elbow.	16
		90 Degree long radius elbow.	20
		90 Degree street elbow.	50
		45 Degree street elbow.	26
		Square corner elbow.	57
	Standard Tee	With flow through run.	20
		With flow through branch.	60
Pipe		90 Degree pipe bends.	See Equation

approximate expression for head in the tank as a function of time can be found as follows:

$$\frac{dH}{dT} = \frac{1}{A_8} (Q_I)$$

For steady flow, neglecting friction, with the control valve open,

$$Q_I = (1.0)CVI \sqrt{\frac{(PI-PO)w_w}{w}}$$

For water,

$$\frac{w}{w_w} = 1.0$$

Therefore,

$$\frac{dH}{dT} = \frac{CVI}{A_8} \sqrt{PI-PO}$$

$$dH = [\frac{CVI}{A_8} \sqrt{PI-PO}\ ]dT$$

Integrating,

$$H]_0^H = [\frac{CVI}{A_8} \sqrt{PI-PO}\ ]_0^T\ T$$

If the tank is empty at T = 0,

$$H = \frac{CVI}{A_8} \sqrt{PI-PO}\ \ T$$

Setting

$$PO = 15.0$$

$$PI = 16.6$$

$$A_8 = 1000$$

$$CVI = 654$$

results in

$$H = (0.827)\ T$$

Table 5 shows the results obtained from the above expression and the corresponding computer solution and the per cent deviation for a five-second run. The per cent deviation uses the computer values as base.

Table 5. Results of the First Flow Equation Test

T	H_{Hand}	$H_{Computer}$	Per Cent Deviation
0.0	0	0	0
1.0	0.827	0.801578	3.12
1.5	1.240	1.20259	3.08
2.0	1.654	1.60360	3.11
2.5	2.070	2.00461	3.14
3.0	2.480	2.40563	3.08
4.0	3.310	3.02766	3.18
5.0	4.130	4.00968	3.00

Table 5 demonstrates the fairly close agreement between the results of the two methods of solution for the short time interval used. It can be seen that H_{hand} is consistently larger than $H_{computer}$. It is felt that this is due to the inclusion of the effects of pipe friction in the computer solution. This pipe friction is a retarding factor which reduces the flow rate into the tank. Therefore, the tank fills slower than when the pipe friction is ignored.

An approximate expression for the complete set of equations can be found for certain specific cases.

Using the head equation,

$$\frac{dH}{dT} = \frac{1}{A_8}(Q_I - Q_o)$$

If Q_I is held constant and

$$Q_o = (K20)CVO\sqrt{wH}$$

$$\frac{dH}{dT} = \frac{Q_I}{A_8} - \frac{(K20)CVO}{A_8}\sqrt{w}\ \sqrt{H}$$

Making the following substitutions,

$$u = \sqrt{H}$$

$$K = (K20)CVO\sqrt{w}$$

$$\frac{dH}{dT} = 2u\,\frac{du}{dT}$$

results in

$$2\,A_8\,u\,\frac{du}{dT} = Q_I - ku$$

Separating the variables,

$$2\,A_8\,\left[\frac{u\,du}{Q_I - ku}\right] = dT$$

Integrating the above equation and replacing u with H results in

$$T - 2A_8\left[\frac{\sqrt{H} - \sqrt{H_I}}{k} + \frac{Q_I}{k^2}\ln\left[\frac{\sqrt{H} - Q_I/k}{\sqrt{H_I} - Q_I/k}\right]\right]$$

Table 6 shows comparative values of head and flow out, and lists per cent deviation for the two methods of solution. As before, per cent deviation uses the computer values as base.

It can be seen that there is a very small per cent deviation in head resulting from the two methods of solution. Since the hand calculation neglects pipe friction and the computer solution does not, it can be assumed, that for the specific example used, the effect of the pipe friction on the inlet side of the tank is virtually cancelled by the effect of the pipe friction on the outlet side of the tank. Table 6 also indicates that Q_{oHand} is consistently larger than $Q_{oComputer}$. This can also be explained as the effect of the inclusion of the friction factor in the computer solution.

Table 6. Results of Second Flow Equation Test

T	H_{Hand}	$H_{Computer}$	Q_{oHand}	$Q_{oComputer}$	Per Cent Deviation H	Per Cent Deviation Q_o
0.	200	200.000	1755	1755.00	0	0
0.459	199	199.265	1750	1677.20	0.133	4.34
1.048	198	198.338	1745	1667.88	0.170	4.63
2.416	196	196.196	1735	1658.65	0.100	4.60
3.192	194	194.998	1728	1653.57	0.200	4.50
4.240	192	193.370	1718	1646.64	0.713	4.33

APPENDIX C

DEVELOPMENT OF CONTROL EQUATIONS

Based on information obtained from Honeywell catalogs and discussions with local Honeywell engineers, the following Table of System Types was prepared.

Table 7. System Types

Control	Pneumatic	Electric
On-Off		Open or Closed Tank Versa-Tran (Controller) Probes (Sensor)
Proportional	Open Tank Low Cost System A Low Cost System B Type 71-06 Level Control (Displacer) Closed Tank Type 71-06 Level Control (Displacer)	Open or Closed Tank R7165A Proportioning Relay (Controller) P/I Transmitter (Sensor)
Proportional Plus Integral	Open Tank Type 71-06 Level Control (Displacer) 2-Mode Indicating Controller, Statis Pressure Transmitter (Sensor)	
Proportional Plus Integral	Closed Tank Type 71-06 Level Control (Displacer)	

It should be noted that no systems containing three mode controllers are included. Eckman (19), Buckley (20), and Honeywell (21) indicate that the rate mode is rarely used in liquid level control.

It can be seen from Table 7 that only proportional systems have both electric and pneumatic components. This does not mean that Honeywell cannot provide components from which an electric proportional plus integral control system, for example, could be made. It does indicate that the frequency of demand for such a system is so low that applications engineers at the local level do not have full information on the system and considerable correspondence with the factory would be required for the purchase of such a system.

The set of equations governing the valve opening for an on-off control system is obtained as follows. Figure 8 is a diagram of the on-off control system used in this study. It can be seen that this type of liquid level control can only maintain a level between two points; it cannot hold the level at a specified location. Honeywell on-off liquid level control systems require that the liquid used be an electrical conductor. The relationship of motor voltage V versus the error (H1-H), in the liquid level is given in Figure 9.

A design requirement for the systems of this study was that all control valves fail shut. This means that all valves must be of the spring return type for which closing time is assumed negligible. On the other hand, opening time is a linear function of time. The motor-operated valves used in the study have opening times of 30 and 60 seconds.

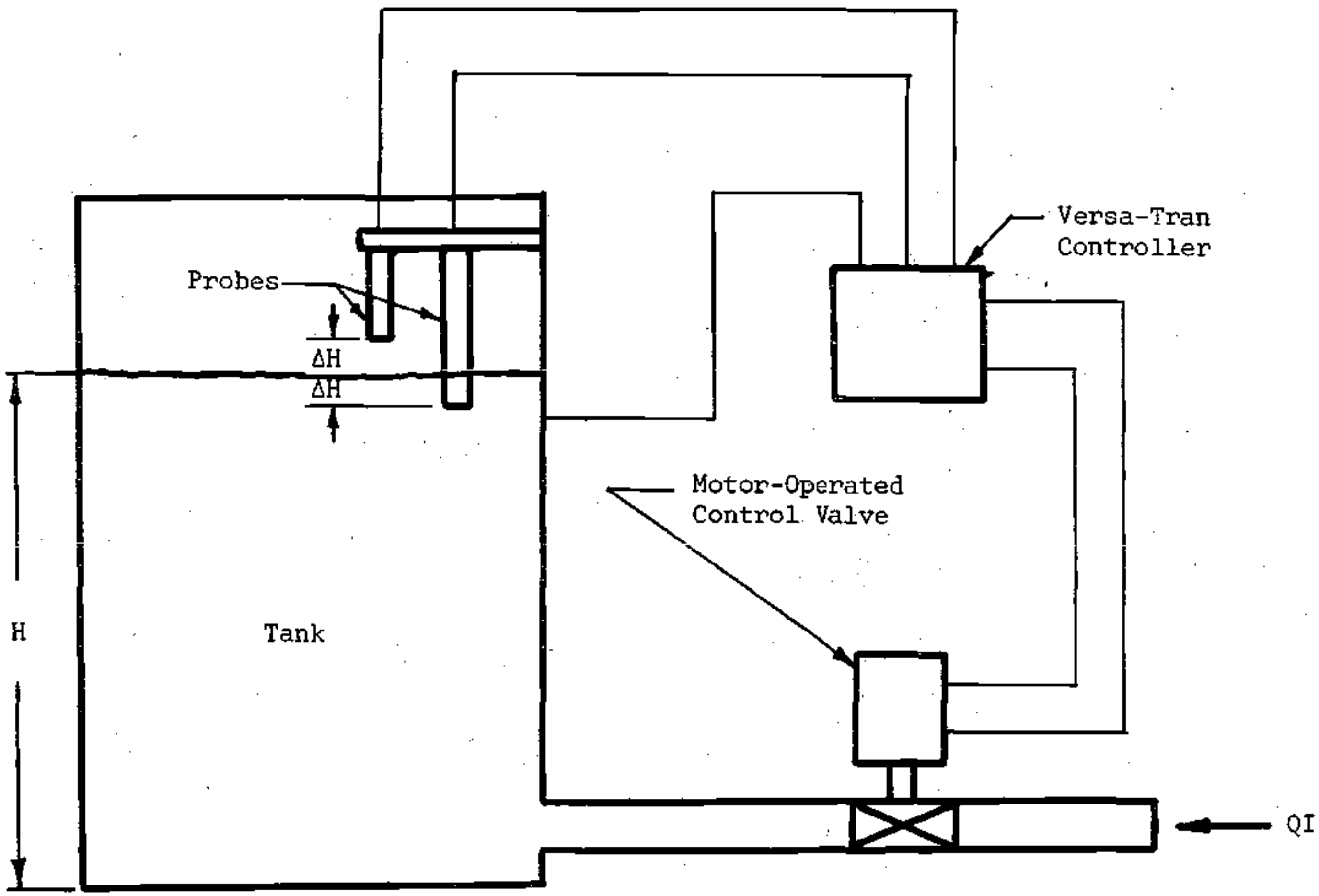


Figure 8. On-Off Control System Diagram.

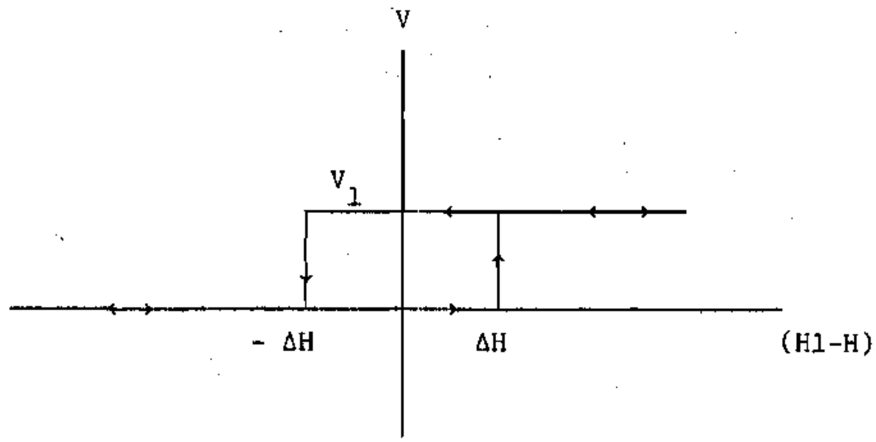


Figure 9. Plot of Motor Voltage Versus Error.

As shown in Figure 9, the voltage V, to the valve motor is not only a function of the error in the head (H1-H), but is also a function of the voltage past history. In addition, the opening of the valve is a function of the voltage applied to the valve and a function of the time elapsed since the voltage was applied. With this in mind, the following set of equations can be written.

For $(H1-H) \leq -\Delta H$,

$$V = 0$$

and

$$\frac{x}{X} = 0 \; . \tag{46}$$

For $-\Delta H < (H1-H) < \Delta H$

and $V(T-\Delta T) = 0$,

$$V = 0$$

and

$$\frac{x}{X} = 0 \tag{47}$$

For $-\Delta H < (H1-H) < \Delta H$

and $V(T-\Delta T) = V1$,

$$V = V1$$

and if $T1 \leq QOT$,

$$\frac{x}{X} = \frac{T1}{QOT} . \tag{48}$$

if $T1 > QOT$,

$$\frac{x}{X} = 1.0 \tag{49}$$

For $(H1-H) \geq \Delta H$,

$$V = V1$$

and if $T1 \leq QOT$,

$$\frac{x}{X} = \frac{T1}{QOT} . \tag{50}$$

or if $T1 > QOT$,

$$\frac{x}{X} = 1.0 \tag{51}$$

The control equation used for the proportional systems of this study is derived for a pneumatic system but can be used for an electric system.

According to Buckley (22), the natural frequency of a displacer type level sensor is one to three cycles per second. Since this frequency is much higher than the frequency of the tank level changes for any practical liquid level system, the transfer function for the displacer is represented by a constant. This same reasoning also holds for the other level sensors included in the study. For the purpose of the study, the system is assumed to be set up so that a zero error signal corresponds to flow through the control valve equal to one half the maximum flow for a given pressure drop across the valve.

From Figure 5 in Appendix B, when $Q/Q_m = \frac{1}{2}$, $x/X = 0.82$ for an equal-percentage control valve. The relationship between x/X and P is shown in Figure 10. From Figure 10,

$$P = P_{mi} + (P_{ma}-P_{mi})\frac{x}{X} \tag{52}$$

The relationship between P and (H1-H) is shown in Figure 11. Point P_c can be found by setting x/X equal to 0.82 in Equation (52). In Figure 11,

$$P = \frac{P_{ma}-P_{mi}}{PB(CR)}(H1-H) + P_c \qquad (53)$$

Eliminating P between Equation (52) and Equation (53) yields

$$\frac{x}{X} = \frac{(H1-H)}{PB(CR)} + 0.82 \qquad (54)$$

Because of the limiting nonlinearities in Figure 11, the expression for x/X for the entire range of allowable values is

$$\frac{x}{X} = \begin{cases} 0 & \frac{(H1-H)}{PB(CR)} + 0.82 \le 0 \\ \frac{(H1-H)}{PB(CR)} + 0.82 & 0 < \frac{(H1-H)}{PB(CR)} + 0.82 < 1 \\ 1 & \frac{(H1-H)}{PB(CR)} + 0.82 \ge 1 \end{cases} \qquad (55)$$

The most sophisticated control system used in this study uses a two mode controller incorporating a proportional mode and an integral or reset mode. The addition of the integral mode to the proportional controller results in the following control equation,

$$\frac{x}{X} = \frac{1}{PB(CR)}\left[RR\int_o^t (H1-H)dt + (H1-H)\right] + 0.82 \qquad (56)$$

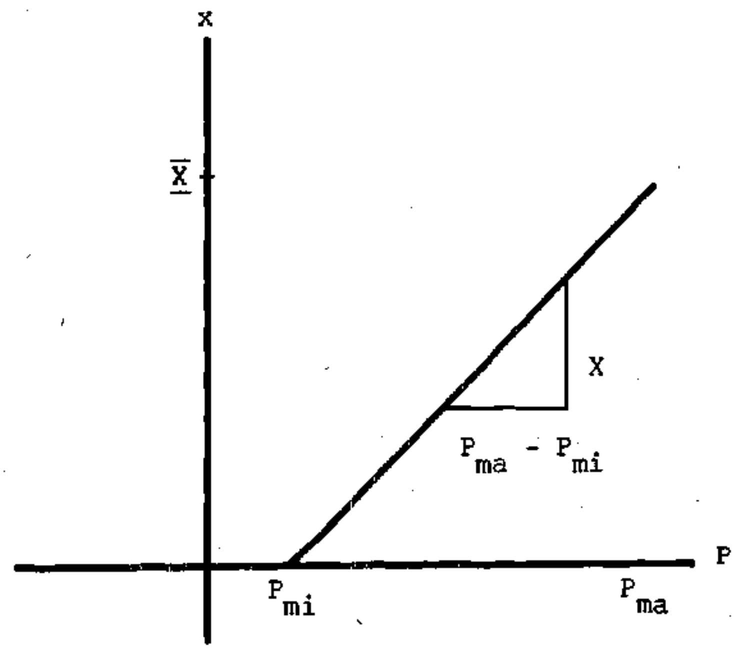


Figure 10. Plot of Valve Opening Versus Control Pressure.

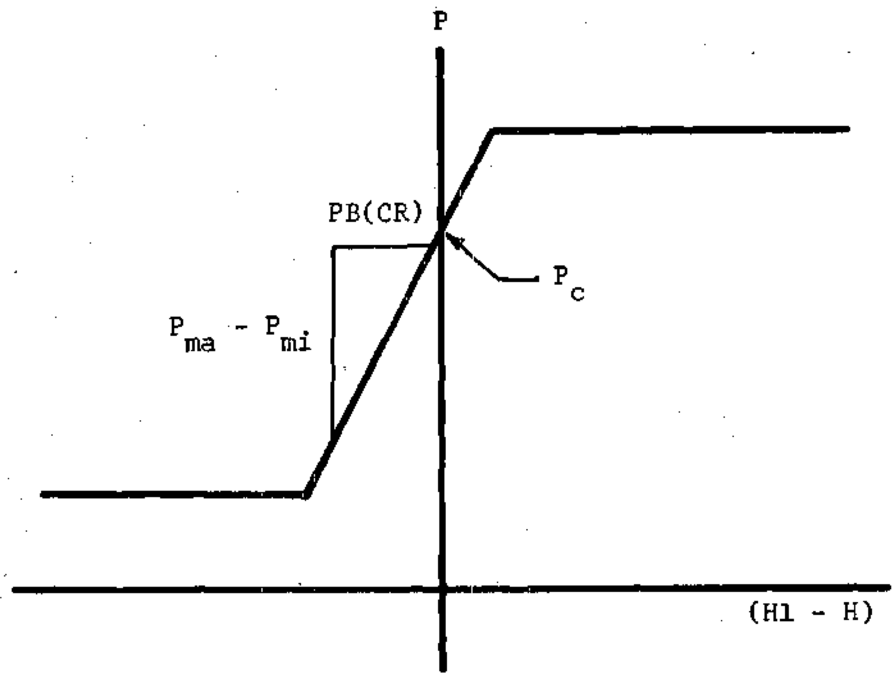


Figure 11. Plot of Control Pressure Versus Error in Level.

Again, the range of x/X is limited and, if

$$\frac{1}{PB(CR)}\left[RR\int_{o}^{t}(H1-H)dt + (H1-H)\right] + 0.82 = A \ ,$$

the complete proportional plus integral control equation is

$$\frac{x}{X} = \begin{cases} 0 & A \leq 0 \\ \frac{1}{PB(CR)}\left[RR\int_{o}^{t}(H1-H)dt + (H1-H)\right] + 0.82 & 0 < A < 1 \\ 1.0 & 1 \leq A \end{cases} \qquad (57)$$

APPENDIX D

THEORY OF DIFFERENTIAL EQUATION SOLUTION

Determination of the time response of the tank liquid level requires the simultaneous solution of a system of first order nonlinear differential equations. The number of differential equations making up the system depends on the tank configuration and on the type of control system used.

The Runge-Kutta method of numerical integration is used to obtain the required solutions. A brief explanation of this method follows (23). Given $y' = f(x,y)$ and (x_o,y_o), additional points (x_1,y_1) on the integral curve are desired. Let $\Delta x = h$ and $\Delta y = k$ so that

$$y_1 = y_o + k \tag{58}$$

and

$$x_1 = x_o + h \tag{59}$$

The problem is now reduced to finding values of k to the required degree of accuracy. There have been several methods presented for finding k based on the Taylor series expansion

$$k = y_o'h + y_o'' \frac{h^2}{2!} + y_o''' \frac{h^3}{3!} + y_o^{iv} \frac{h^4}{4!} + \cdots. \tag{60}$$

The easiest to calculate are the formulas of Runge, which have been modified by Heun and Kutta. The chief advantage of these formulas is that they can be written with the functional values only. Let k_1, k_2, $k_3, \ldots$, be the successive approximations of k, which will be weighted and averaged as indicated by Equation (65) below. The formulas as modified by Kutta are

$$k_1 = f(x_o, y_o)h \tag{61}$$

$$k_2 = f(x_o + \frac{h}{2}, y_o + \frac{k_1}{2})h \tag{62}$$

$$k_3 = f(x_o + \frac{h}{2}, y_o + \frac{k_2}{2})h \tag{63}$$

$$k_4 = f(x_o + h, y_o + k_3)h \tag{64}$$

and

$$k = \frac{1}{6} (k_1 + 2k_2 + 2k_3 + k_4) \tag{65}$$

If y does not appear on the right side of the differential equation, the problem reduces to simple quadrature and Kutta's equations take the form of Simpson's rule.

The method can be extended to systems of simultaneous differential equations. For example, consider two simultaneous equations

$$y' = f(x,y,z) \tag{66}$$

and

$$z' = g(x,y,z) \tag{67}$$

By defining q such that

$$z_1 = z_o + q \tag{68}$$

k and q can be written as follows:

$$k_1 = f(x_o,y_o,z_o)h \tag{69}$$

$$q_1 = g(x_o,y_o,z_o)h \tag{70}$$

$$k_2 = f(x_o + \frac{h}{2},\ y_o + \frac{k_1}{2},\ z_o + \frac{q_1}{2})h \tag{71}$$

$$q_2 = g(x_o + \frac{h}{2},\ y_o + \frac{k_1}{2},\ z_o + \frac{q_1}{2})h \tag{72}$$

$$k_3 = f(x_o + \frac{h}{2},\ y_o + \frac{k_2}{2},\ z_o + \frac{q_2}{2})h \tag{73}$$

$$q_3 = g(x_o + \frac{h}{2},\ y_o + \frac{k_2}{2},\ z_o + \frac{q_2}{2})h \tag{74}$$

$$k_4 = f(x_o + h,\ y_o + k_3,\ z_o + q_3)h \tag{75}$$

$$q_4 = g(x_o + h,\ y_o + k_3,\ z_o + q_3)h \tag{76}$$

and

$$k = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{77}$$

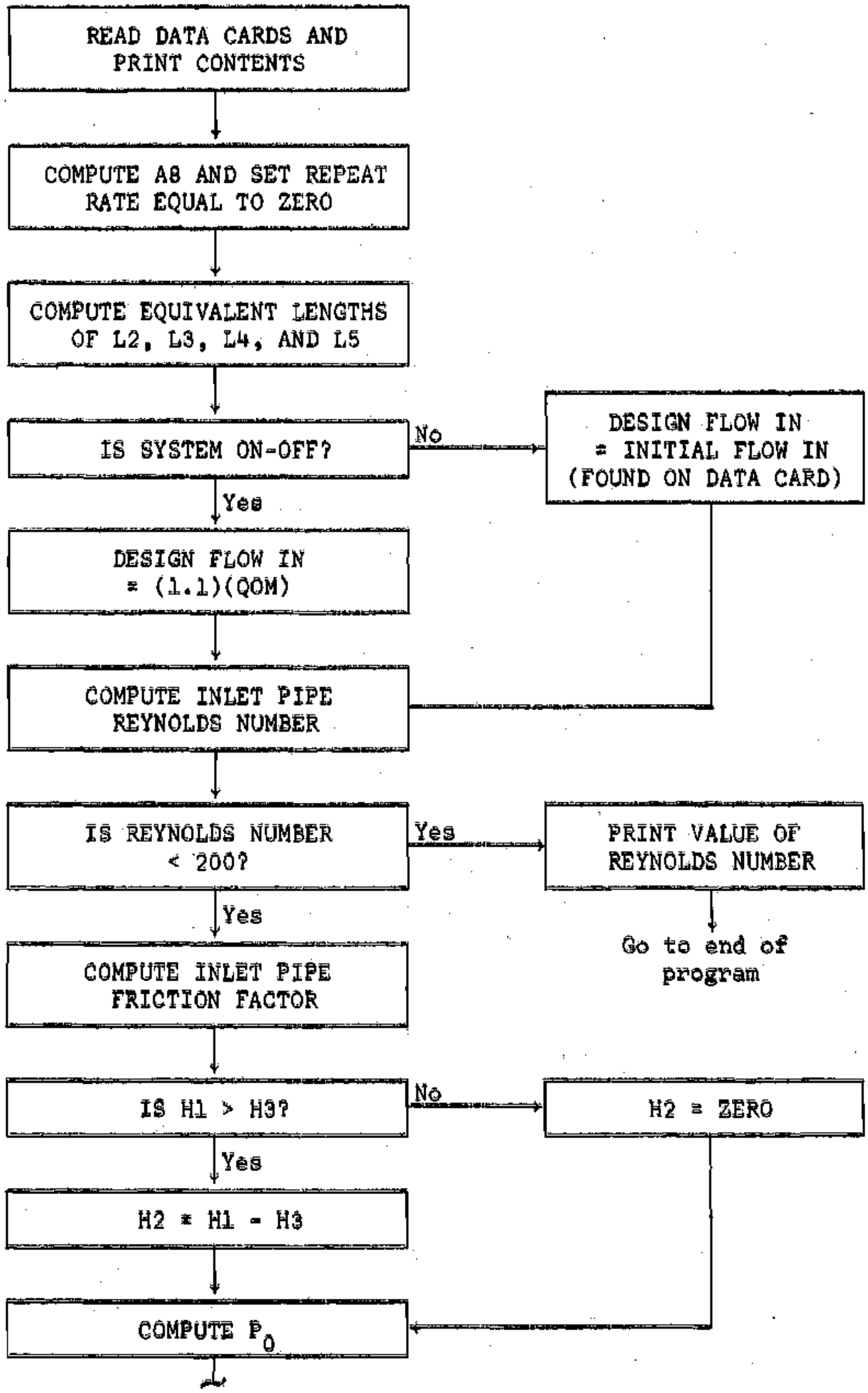
$$q = \frac{1}{6}(q_1 + 2q_2 + 2q_3 + q_4) \tag{78}$$

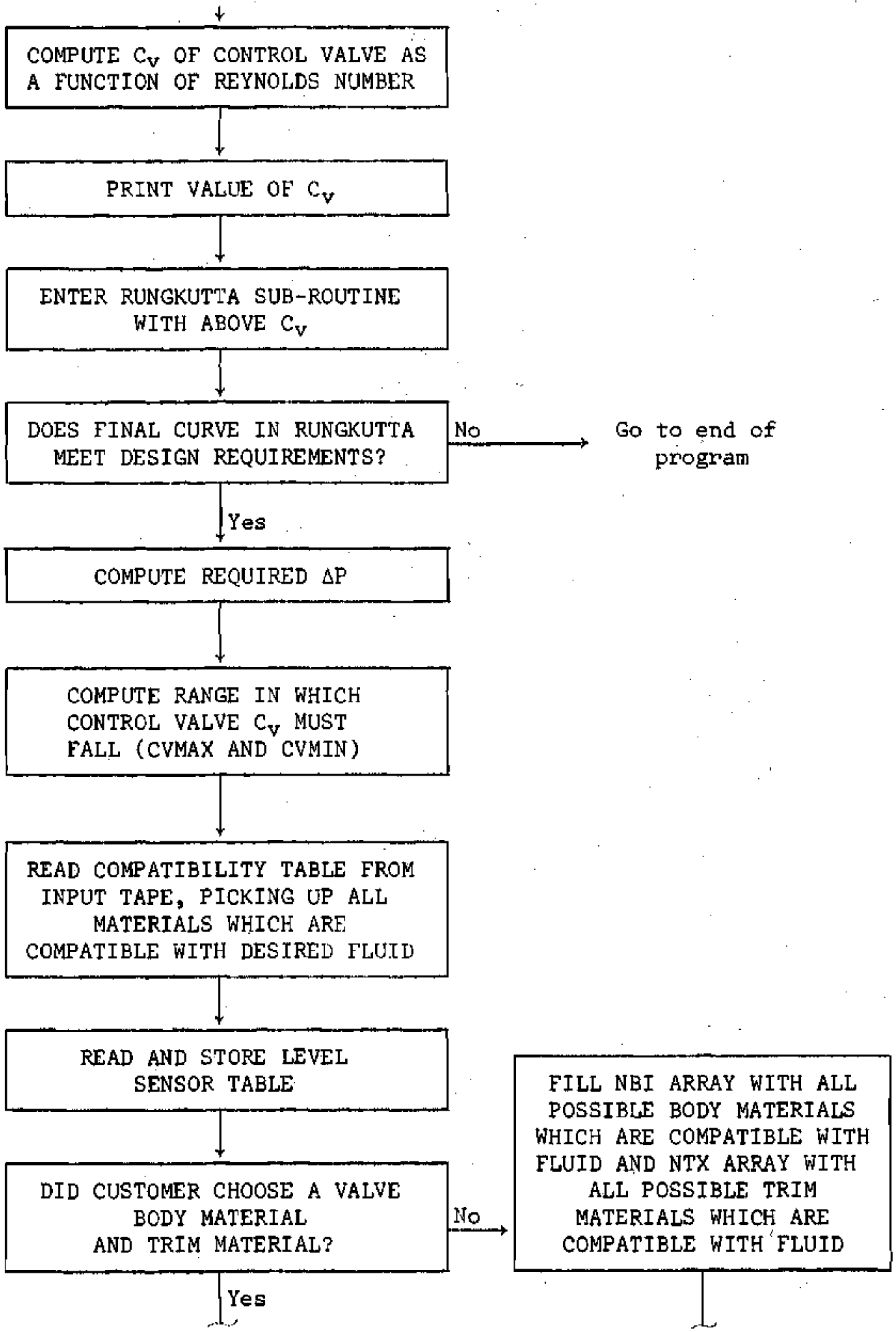
It can be shown that the error in these approximations is proportional to h to the fifth power (24). A simpler but less sophisticated method of determining the error involves re-running the calculation for a short interval of x at half the increment h. The number of significant figures found in agreement is an indication of the accuracy of the results.

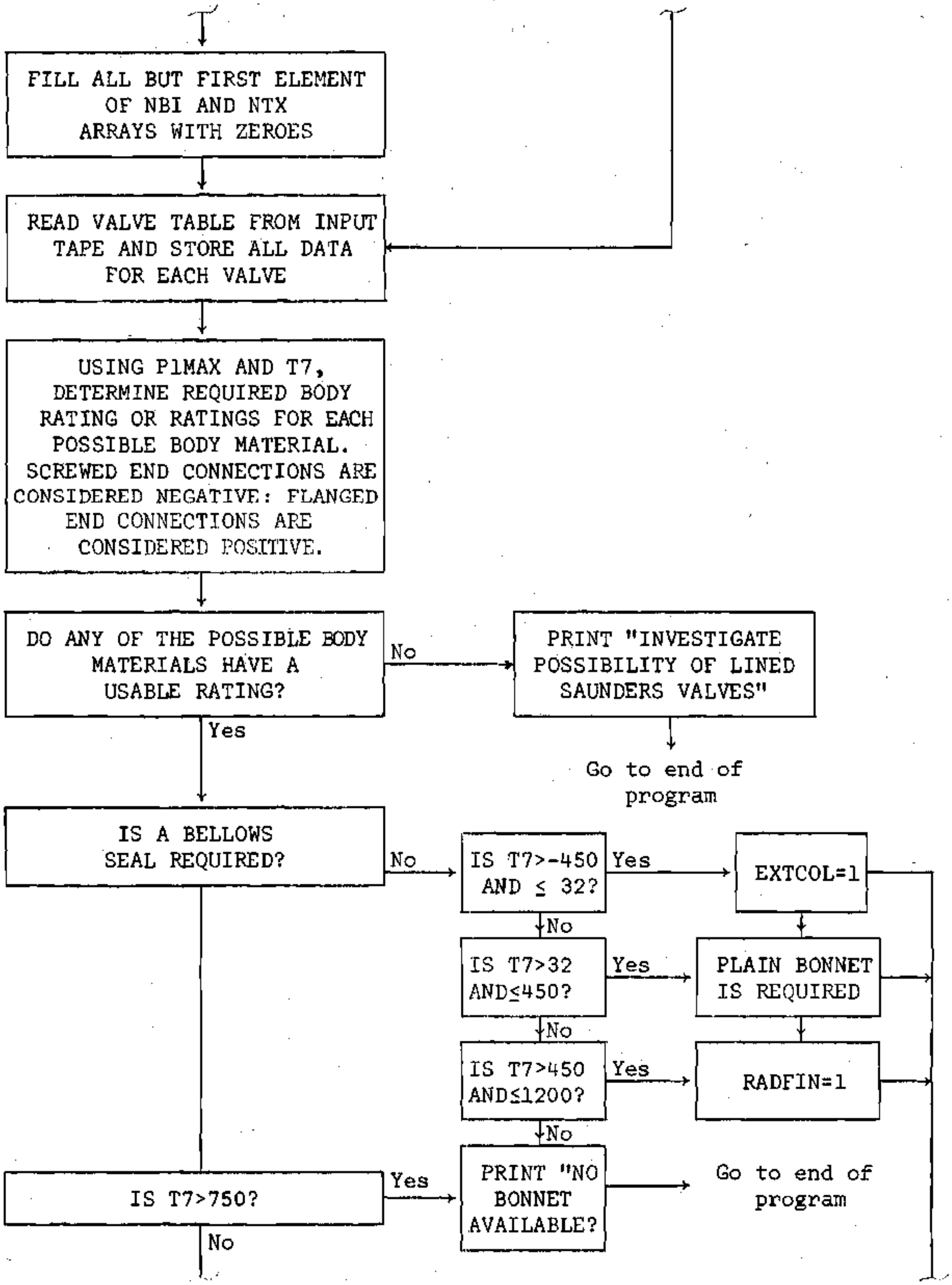
APPENDIX E

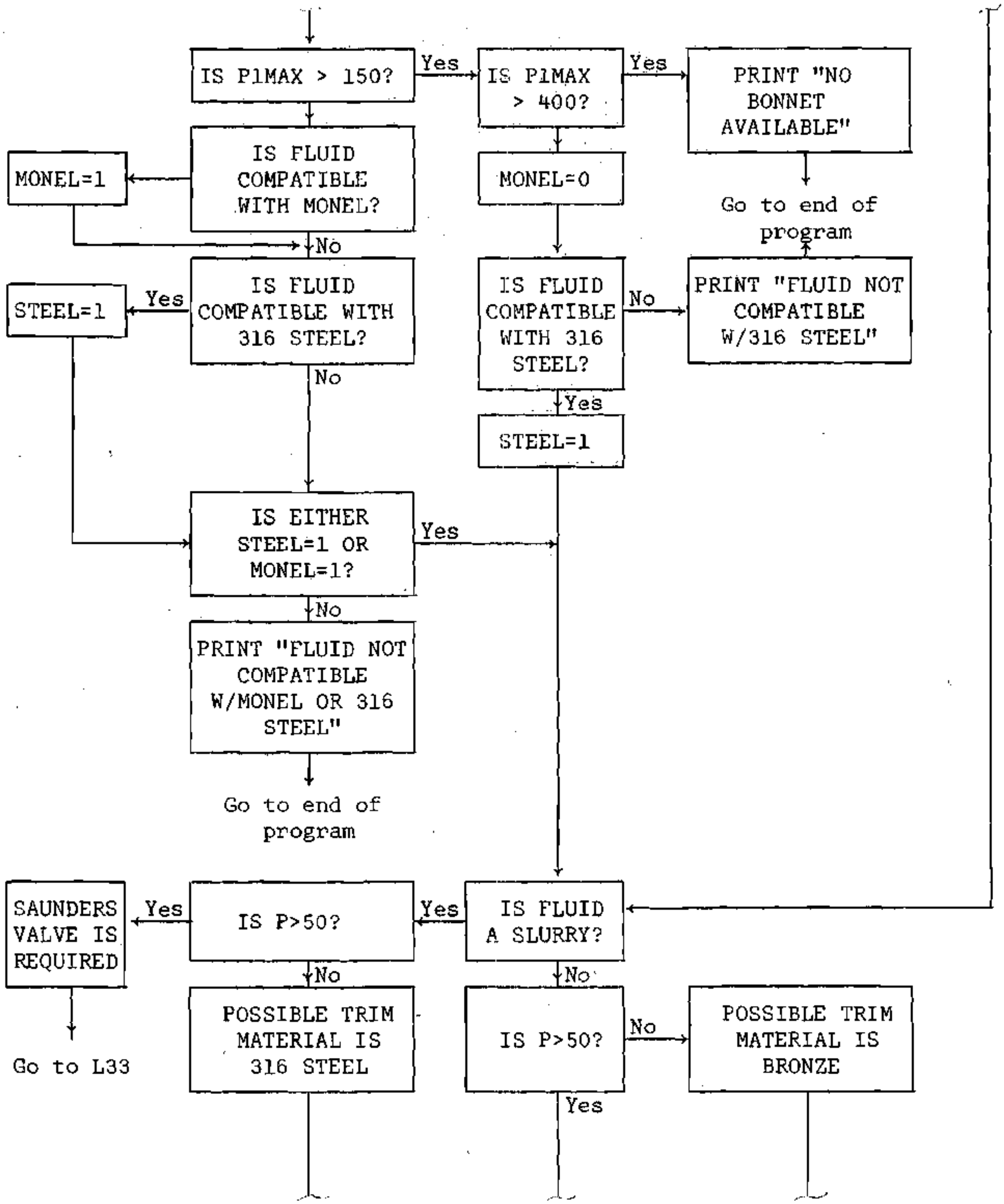
PROGRAM FLOW CHARTS

The Computer Program Flow Charts are given on the following pages. The Main Program Flow Chart starts on page 90 and the Flow Chart for the Runge-Kutta numerical integration method starts on page 103.

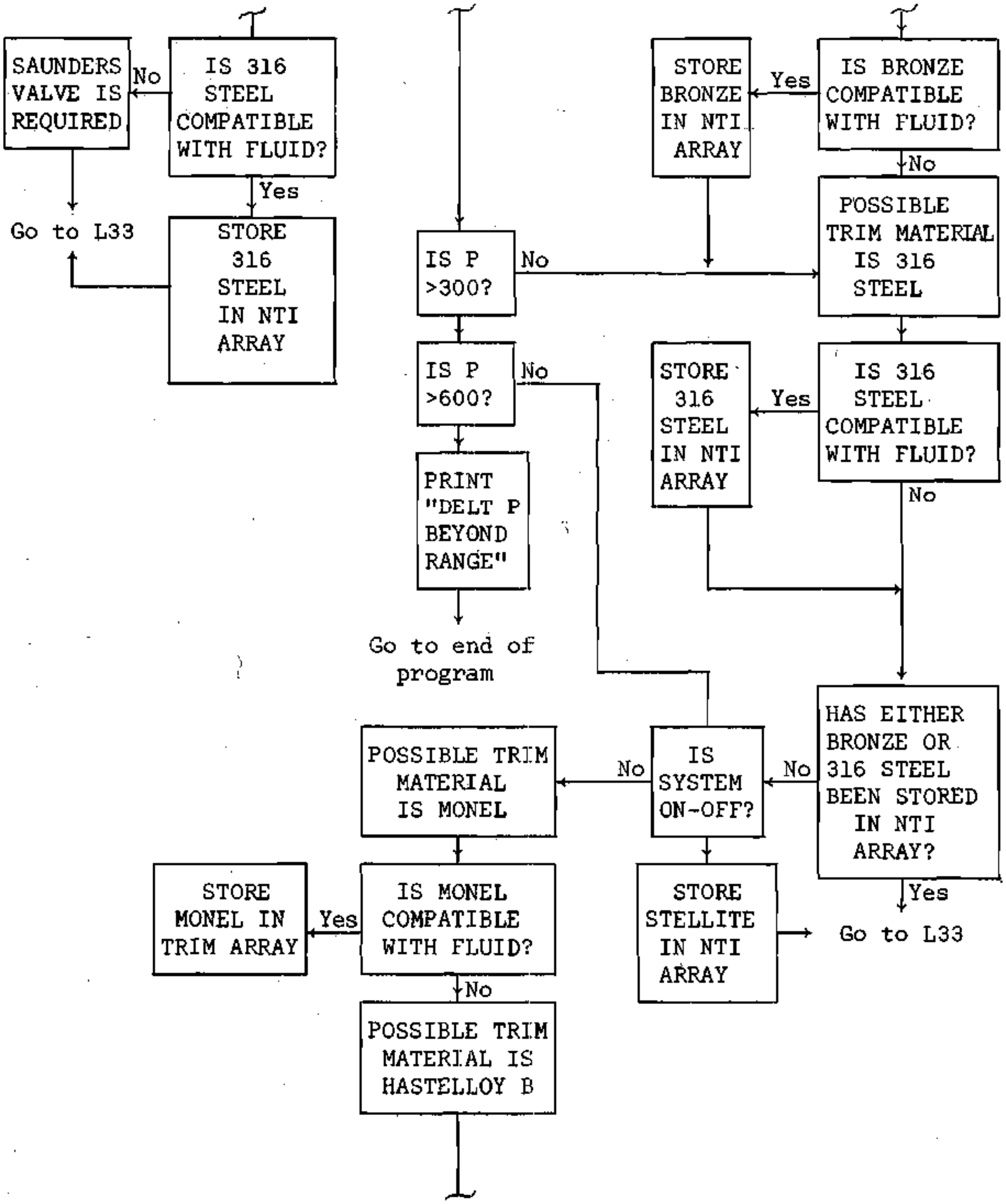

READ DATA CARDS AND PRINT CONTENTS
COMPUTE A8 AND SET REPEAT RATE EQUAL TO ZERO
COMPUTE EQUIVALENT LENGTHS OF L2, L3, L4, AND L5
IS SYSTEM ON-OFF?
No
DESIGN FLOW IN = INITIAL FLOW IN (FOUND ON DATA CARD)
Yes
DESIGN FLOW IN = (1.1)(QOM)
COMPUTE INLET PIPE REYNOLDS NUMBER
IS REYNOLDS NUMBER < 200?
Yes
PRINT VALUE OF REYNOLDS NUMBER
Go to end of program
Yes
COMPUTE INLET PIPE FRICTION FACTOR
IS H1 > H3?
No
H2 = ZERO
Yes
H2 = H1 - H3
COMPUTE P0

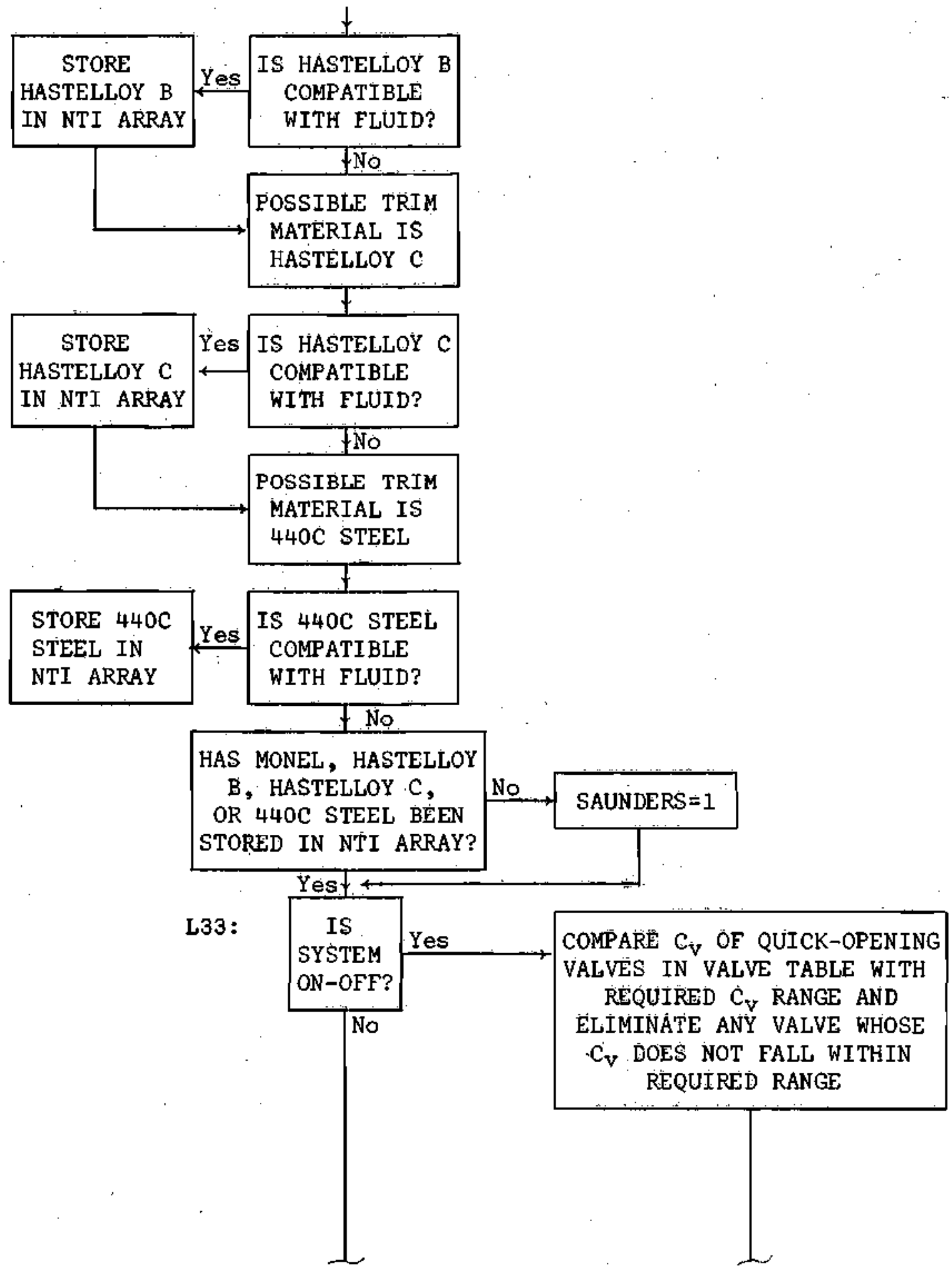

COMPUTE C_v OF CONTROL VALVE AS A FUNCTION OF REYNOLDS NUMBER
PRINT VALUE OF C_v
ENTER RUNGKUTTA SUB-ROUTINE WITH ABOVE C_v
DOES FINAL CURVE IN RUNGKUTTA MEET DESIGN REQUIREMENTS?
No
Go to end of program
Yes
COMPUTE REQUIRED ΔP
COMPUTE RANGE IN WHICH CONTROL VALVE C_v MUST FALL (CVMAX AND CVMIN)
READ COMPATIBILITY TABLE FROM INPUT TAPE, PICKING UP ALL MATERIALS WHICH ARE COMPATIBLE WITH DESIRED FLUID
READ AND STORE LEVEL SENSOR TABLE
DID CUSTOMER CHOOSE A VALVE BODY MATERIAL AND TRIM MATERIAL?
No
FILL NBI ARRAY WITH ALL POSSIBLE BODY MATERIALS WHICH ARE COMPATIBLE WITH FLUID AND NTX ARRAY WITH ALL POSSIBLE TRIM MATERIALS WHICH ARE COMPATIBLE WITH FLUID
Yes

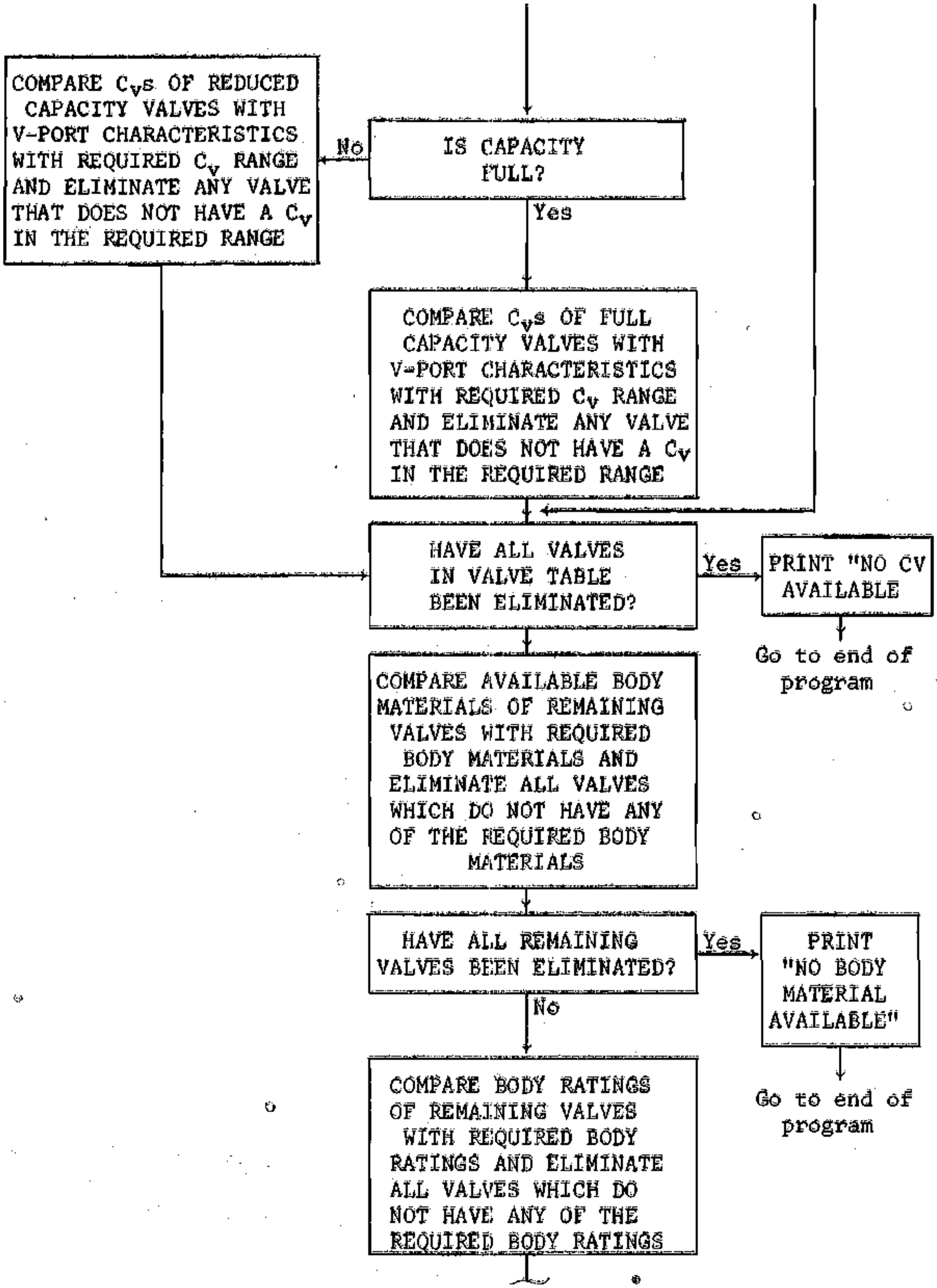

FILL ALL BUT FIRST ELEMENT OF NBI AND NTX ARRAYS WITH ZEROES
READ VALVE TABLE FROM INPUT TAPE AND STORE ALL DATA FOR EACH VALVE
USING P1MAX AND T7, DETERMINE REQUIRED BODY RATING OR RATINGS FOR EACH POSSIBLE BODY MATERIAL. SCREWED END CONNECTIONS ARE CONSIDERED NEGATIVE: FLANGED END CONNECTIONS ARE CONSIDERED POSITIVE.
DO ANY OF THE POSSIBLE BODY MATERIALS HAVE A USABLE RATING?
No
PRINT "INVESTIGATE POSSIBILITY OF LINED SAUNDERS VALVES"
Go to end of program
Yes
IS A BELLOWS SEAL REQUIRED?
No
IS T7>-450 AND ≤ 32?
Yes
EXTCOL=1
No
IS T7>32 AND≤450?
Yes
PLAIN BONNET IS REQUIRED
No
IS T7>450 AND≤1200?
Yes
RADFIN=1
No
IS T7>750?
Yes
PRINT "NO BONNET AVAILABLE?
Go to end of program
No

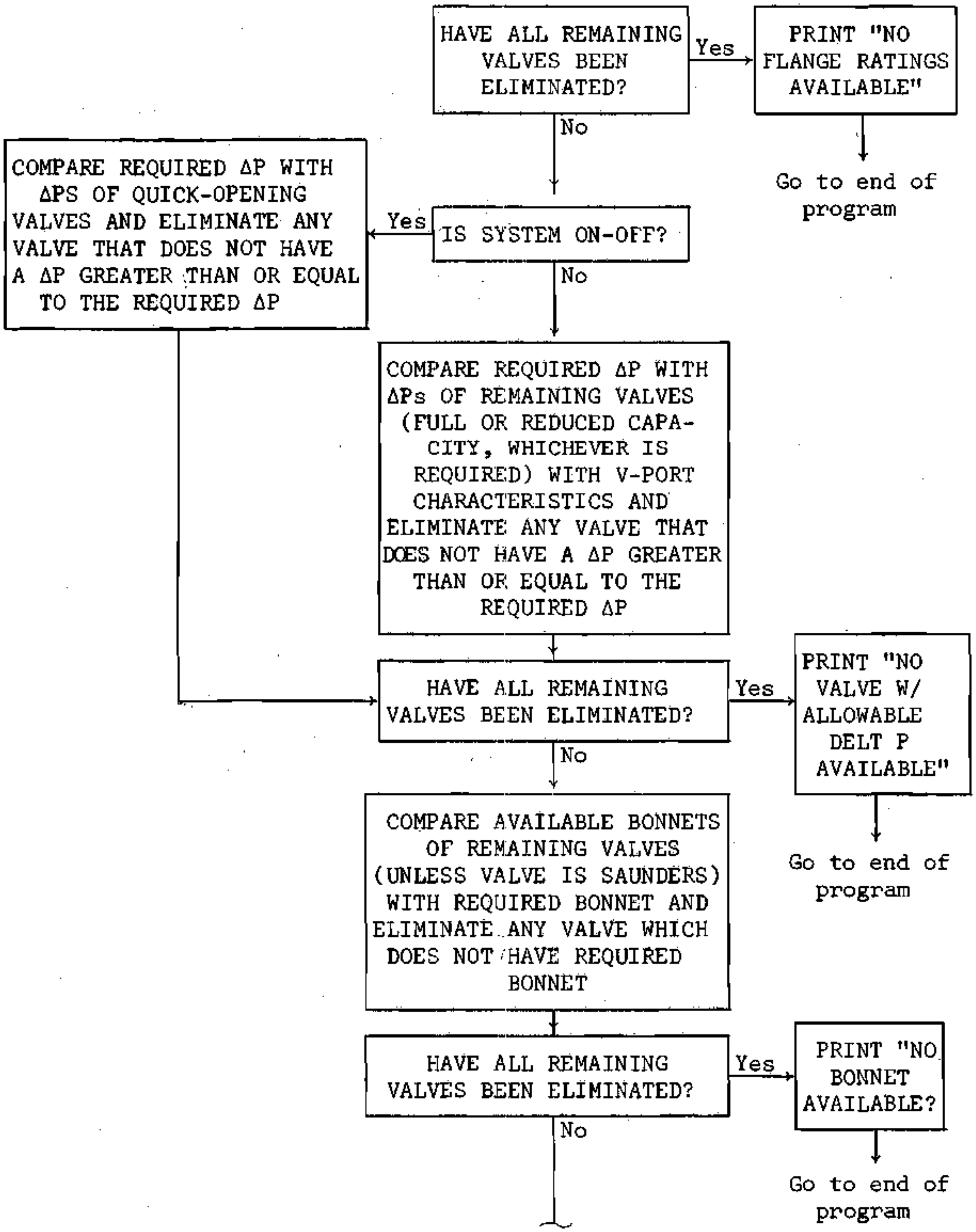


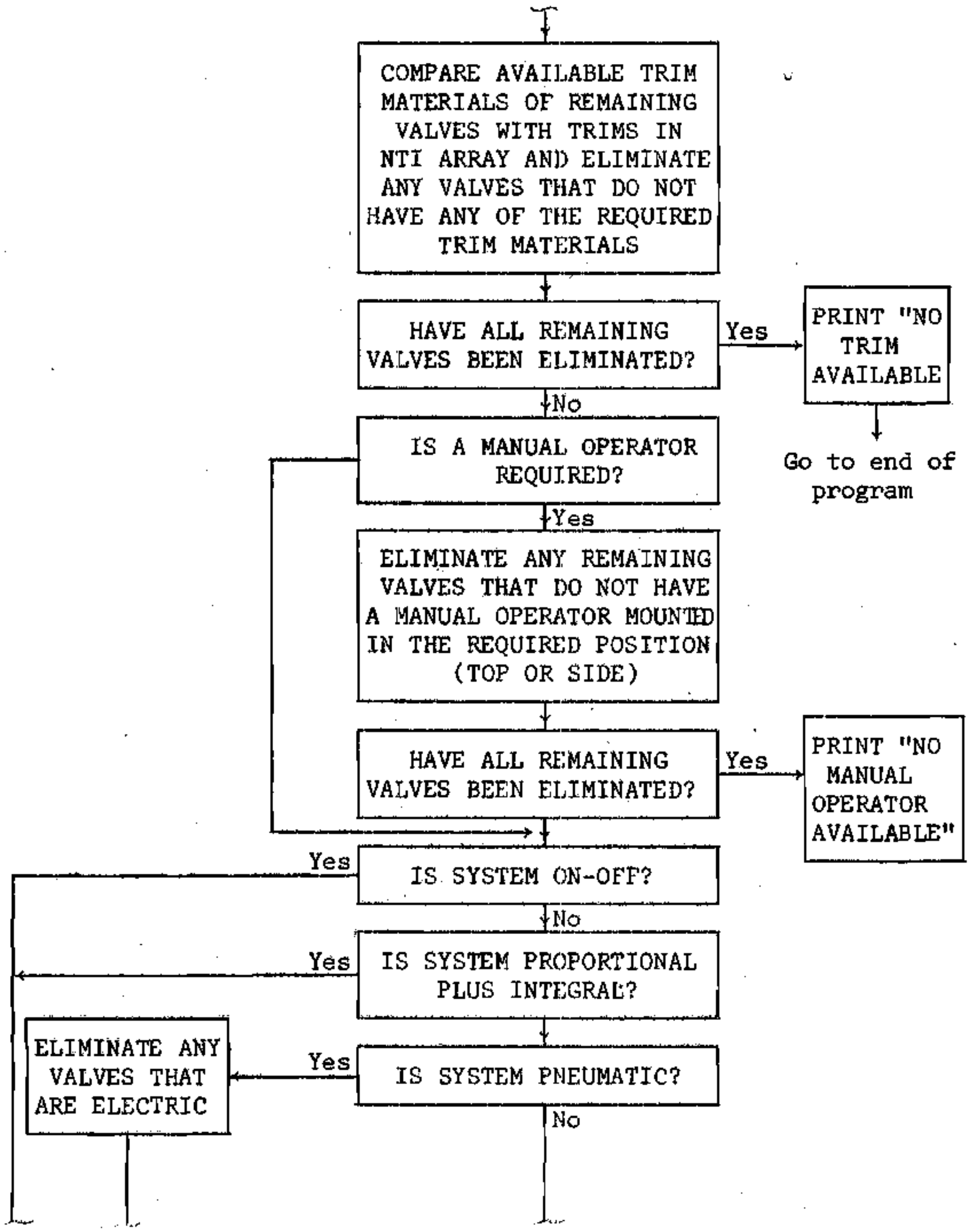
IS P1MAX > 150?
Yes
IS P1MAX > 400?
Yes
PRINT "NO BONNET AVAILABLE"
Go to end of program
MONEL=1
IS FLUID COMPATIBLE WITH MONEL?
MONEL=0
No
STEEL=1
Yes
IS FLUID COMPATIBLE WITH 316 STEEL?
No
IS FLUID COMPATIBLE WITH 316 STEEL?
No
PRINT "FLUID NOT COMPATIBLE W/316 STEEL"
Yes
STEEL=1
IS EITHER STEEL=1 OR MONEL=1?
Yes
No
PRINT "FLUID NOT COMPATIBLE W/MONEL OR 316 STEEL"
Go to end of program
SAUNDERS VALVE IS REQUIRED
Yes
IS P>50?
Yes
IS FLUID A SLURRY?
Go to L33
No
POSSIBLE TRIM MATERIAL IS 316 STEEL
No
IS P>50?
No
POSSIBLE TRIM MATERIAL IS BRONZE
Yes


SAUNDERS VALVE IS REQUIRED
No
IS 316 STEEL COMPATIBLE WITH FLUID?
Yes
Go to L33
STORE 316 STEEL IN NTI ARRAY
IS P >300?
No
IS P >600?
No
PRINT "DELT P BEYOND RANGE"
Go to end of program
STORE BRONZE IN NTI ARRAY
Yes
IS BRONZE COMPATIBLE WITH FLUID?
No
POSSIBLE TRIM MATERIAL IS 316 STEEL
STORE 316 STEEL IN NTI ARRAY
Yes
IS 316 STEEL COMPATIBLE WITH FLUID?
No
HAS EITHER BRONZE OR 316 STEEL BEEN STORED IN NTI ARRAY?
Yes
Go to L33
No
IS SYSTEM ON-OFF?
No
POSSIBLE TRIM MATERIAL IS MONEL
STORE STELLITE IN NTI ARRAY
STORE MONEL IN TRIM ARRAY
Yes
IS MONEL COMPATIBLE WITH FLUID?
No
POSSIBLE TRIM MATERIAL IS HASTELLOY B

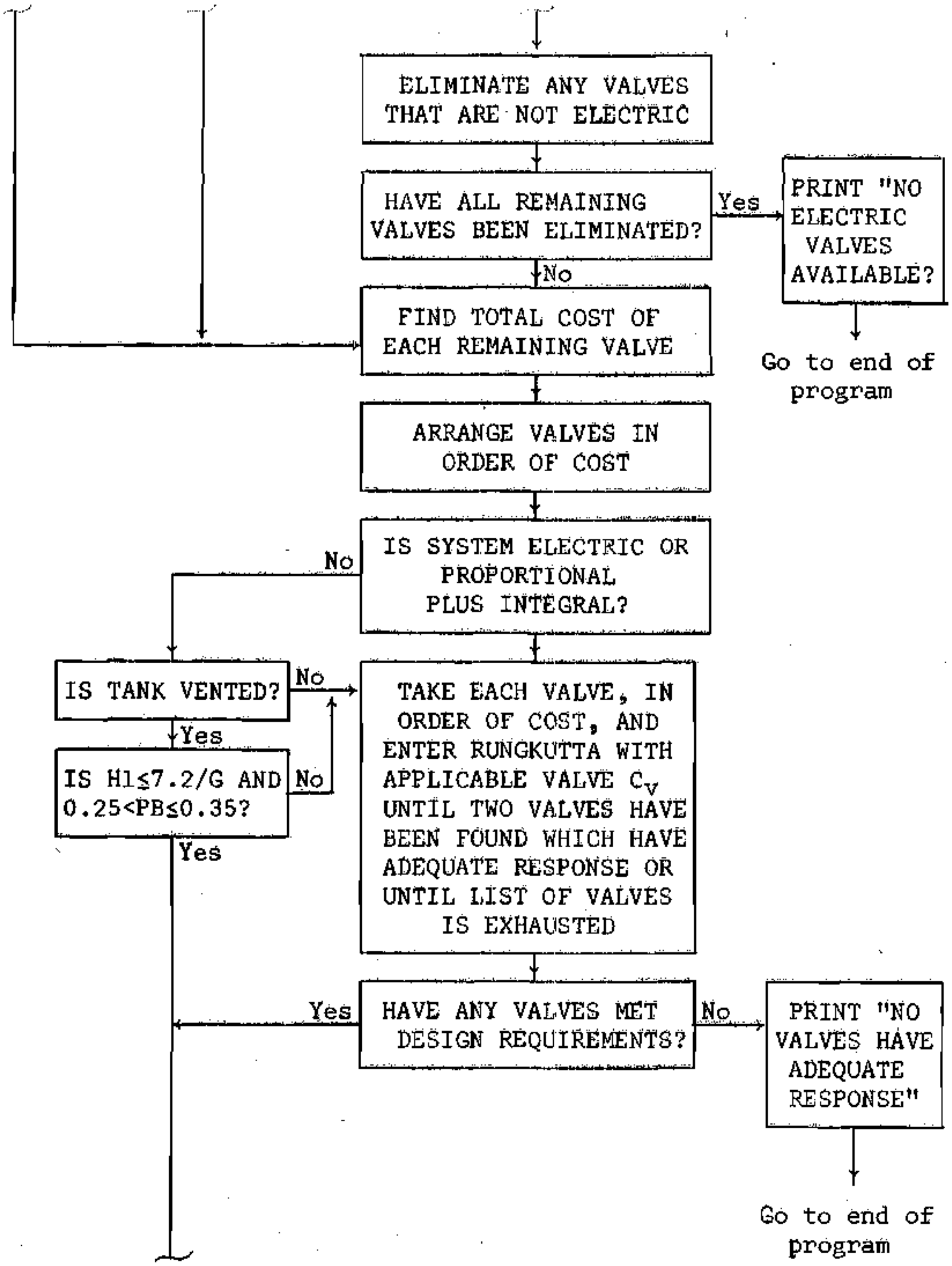

STORE HASTELLOY B IN NTI ARRAY
Yes
IS HASTELLOY B COMPATIBLE WITH FLUID?
No
POSSIBLE TRIM MATERIAL IS HASTELLOY C
STORE HASTELLOY C IN NTI ARRAY
Yes
IS HASTELLOY C COMPATIBLE WITH FLUID?
No
POSSIBLE TRIM MATERIAL IS 440C STEEL
STORE 440C STEEL IN NTI ARRAY
Yes
IS 440C STEEL COMPATIBLE WITH FLUID?
No
HAS MONEL, HASTELLOY B, HASTELLOY C, OR 440C STEEL BEEN STORED IN NTI ARRAY?
No
SAUNDERS=1
Yes
L33:
IS SYSTEM ON-OFF?
Yes
COMPARE C_V OF QUICK-OPENING VALVES IN VALVE TABLE WITH REQUIRED C_V RANGE AND ELIMINATE ANY VALVE WHOSE C_V DOES NOT FALL WITHIN REQUIRED RANGE
No

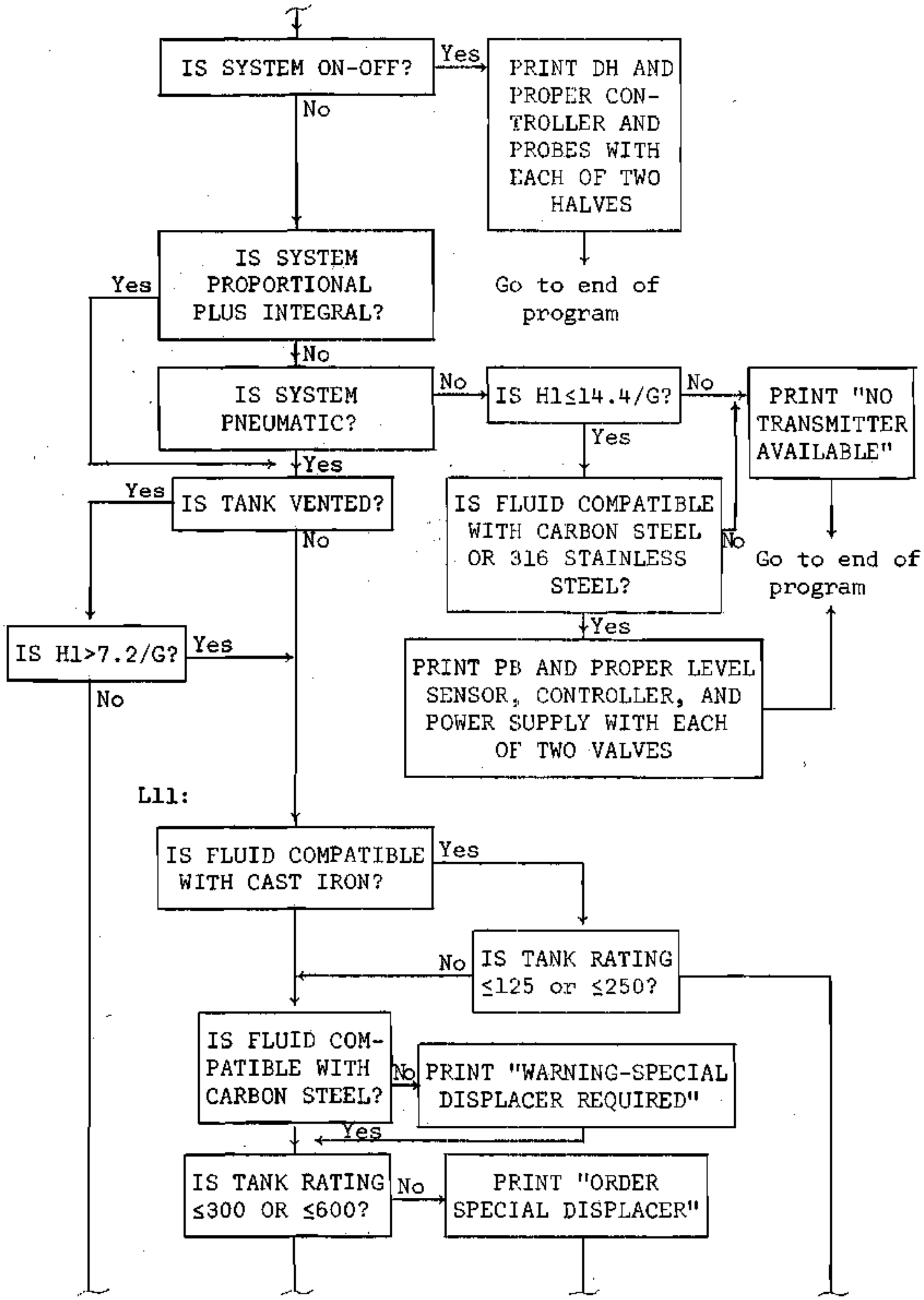

COMPARE C_vs OF REDUCED CAPACITY VALVES WITH V-PORT CHARACTERISTICS WITH REQUIRED C_v RANGE AND ELIMINATE ANY VALVE THAT DOES NOT HAVE A C_v IN THE REQUIRED RANGE
No
IS CAPACITY FULL?
Yes
COMPARE C_vs OF FULL CAPACITY VALVES WITH V-PORT CHARACTERISTICS WITH REQUIRED C_v RANGE AND ELIMINATE ANY VALVE THAT DOES NOT HAVE A C_v IN THE REQUIRED RANGE
HAVE ALL VALVES IN VALVE TABLE BEEN ELIMINATED?
Yes
PRINT "NO CV AVAILABLE
Go to end of program
COMPARE AVAILABLE BODY MATERIALS OF REMAINING VALVES WITH REQUIRED BODY MATERIALS AND ELIMINATE ALL VALVES WHICH DO NOT HAVE ANY OF THE REQUIRED BODY MATERIALS
HAVE ALL REMAINING VALVES BEEN ELIMINATED?
Yes
PRINT "NO BODY MATERIAL AVAILABLE"
Go to end of program
No
COMPARE BODY RATINGS OF REMAINING VALVES WITH REQUIRED BODY RATINGS AND ELIMINATE ALL VALVES WHICH DO NOT HAVE ANY OF THE REQUIRED BODY RATINGS

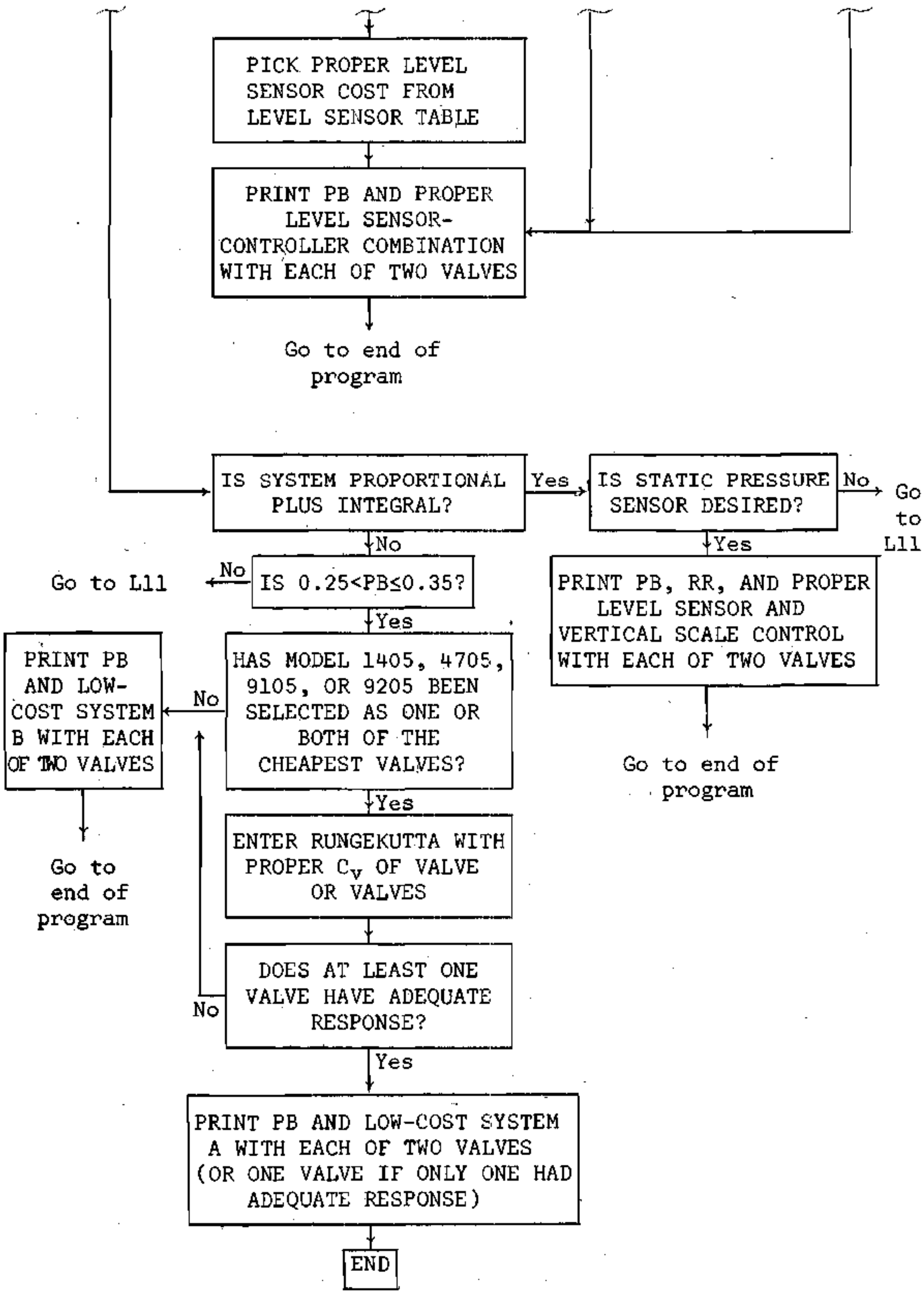

HAVE ALL REMAINING VALVES BEEN ELIMINATED?
Yes
PRINT "NO FLANGE RATINGS AVAILABLE"
Go to end of program
No
IS SYSTEM ON-OFF?
Yes
COMPARE REQUIRED ΔP WITH ΔPS OF QUICK-OPENING VALVES AND ELIMINATE ANY VALVE THAT DOES NOT HAVE A ΔP GREATER THAN OR EQUAL TO THE REQUIRED ΔP
No
COMPARE REQUIRED ΔP WITH ΔPs OF REMAINING VALVES (FULL OR REDUCED CAPA-CITY, WHICHEVER IS REQUIRED) WITH V-PORT CHARACTERISTICS AND ELIMINATE ANY VALVE THAT DOES NOT HAVE A ΔP GREATER THAN OR EQUAL TO THE REQUIRED ΔP
HAVE ALL REMAINING VALVES BEEN ELIMINATED?
Yes
PRINT "NO VALVE W/ ALLOWABLE DELT P AVAILABLE"
Go to end of program
No
COMPARE AVAILABLE BONNETS OF REMAINING VALVES (UNLESS VALVE IS SAUNDERS) WITH REQUIRED BONNET AND ELIMINATE ANY VALVE WHICH DOES NOT HAVE REQUIRED BONNET
HAVE ALL REMAINING VALVES BEEN ELIMINATED?
Yes
PRINT "NO BONNET AVAILABLE?
Go to end of program
No

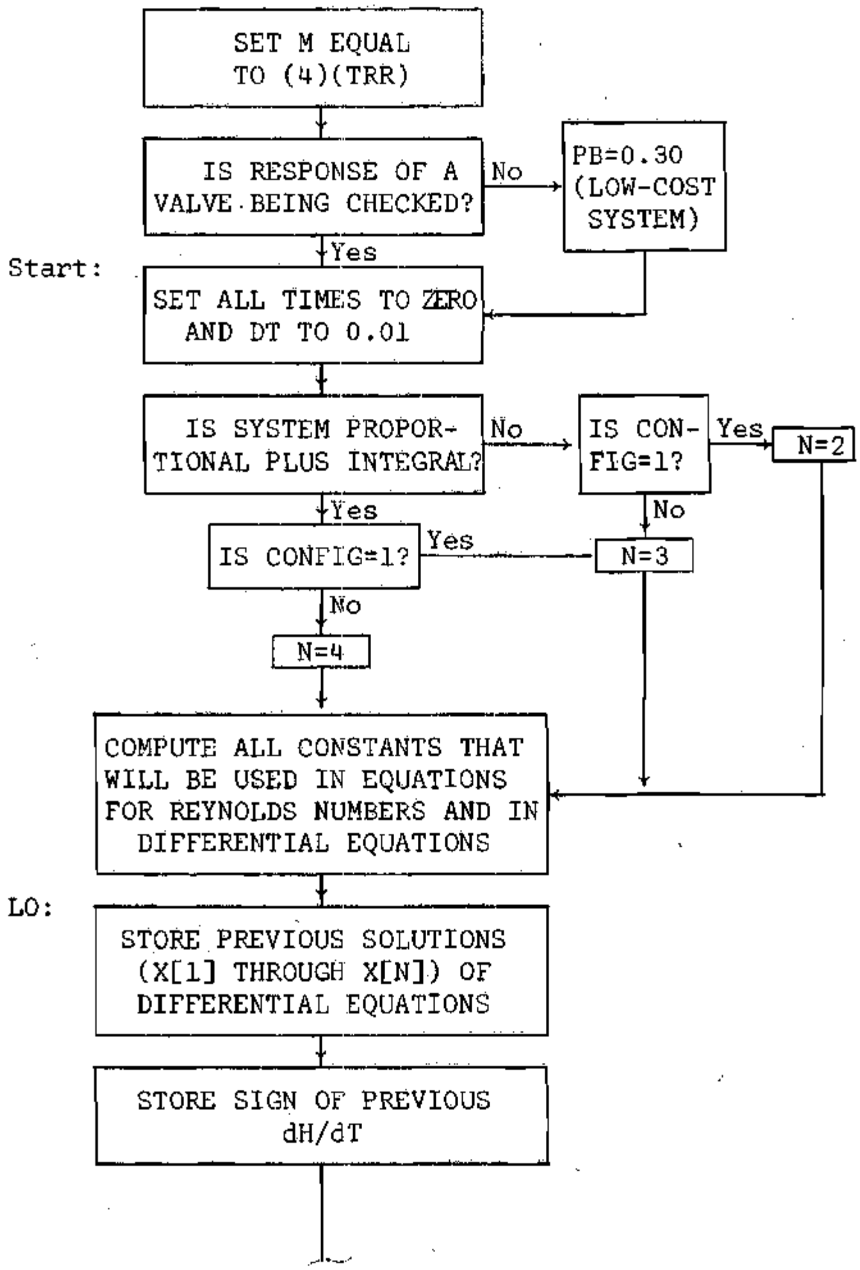


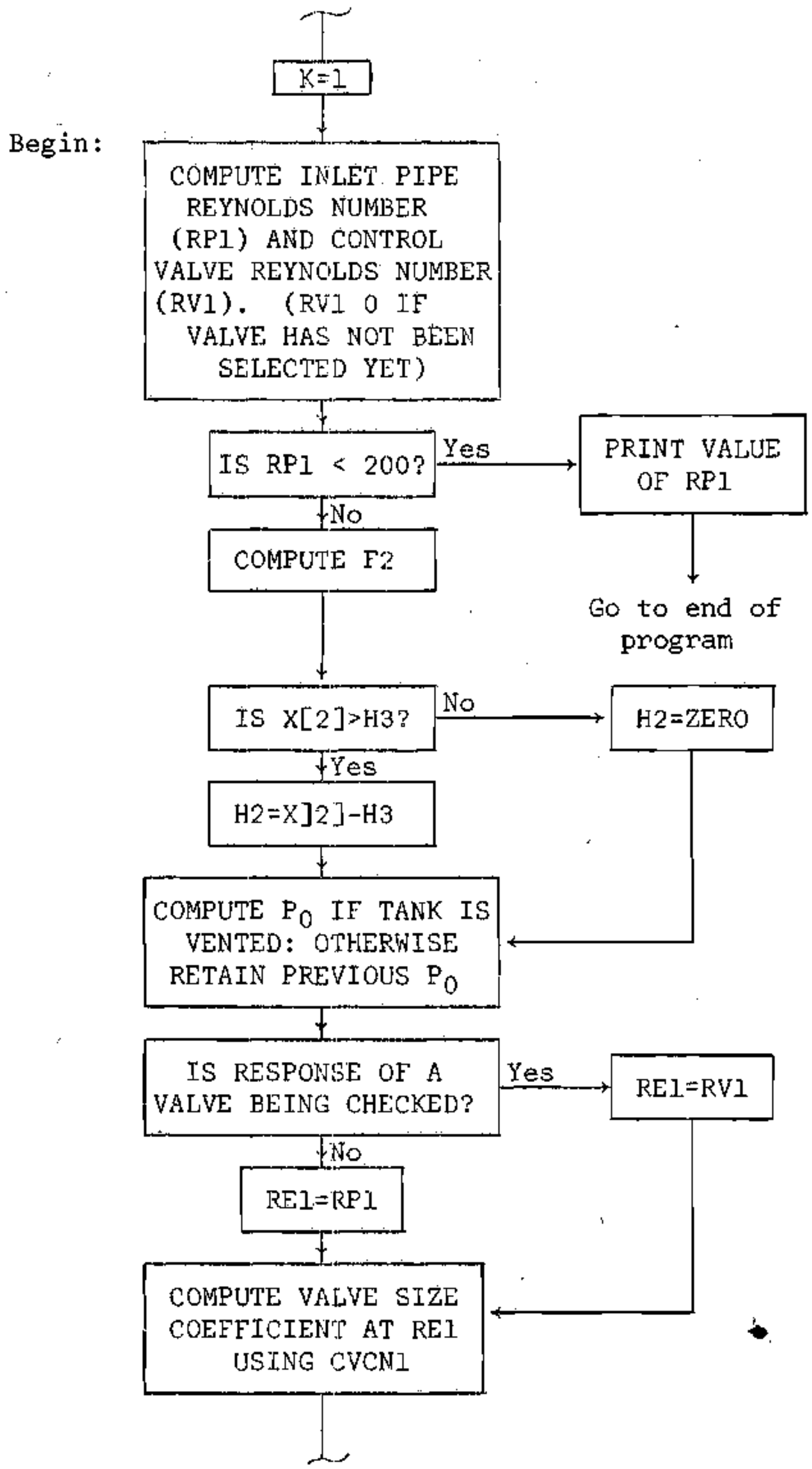
COMPARE AVAILABLE TRIM MATERIALS OF REMAINING VALVES WITH TRIMS IN NTI ARRAY AND ELIMINATE ANY VALVES THAT DO NOT HAVE ANY OF THE REQUIRED TRIM MATERIALS
HAVE ALL REMAINING VALVES BEEN ELIMINATED?
Yes
PRINT "NO TRIM AVAILABLE
Go to end of program
No
IS A MANUAL OPERATOR REQUIRED?
Yes
ELIMINATE ANY REMAINING VALVES THAT DO NOT HAVE A MANUAL OPERATOR MOUNTED IN THE REQUIRED POSITION (TOP OR SIDE)
HAVE ALL REMAINING VALVES BEEN ELIMINATED?
Yes
PRINT "NO MANUAL OPERATOR AVAILABLE"
IS SYSTEM ON-OFF?
Yes
No
IS SYSTEM PROPORTIONAL PLUS INTEGRAL?
Yes
IS SYSTEM PNEUMATIC?
Yes
ELIMINATE ANY VALVES THAT ARE ELECTRIC
No

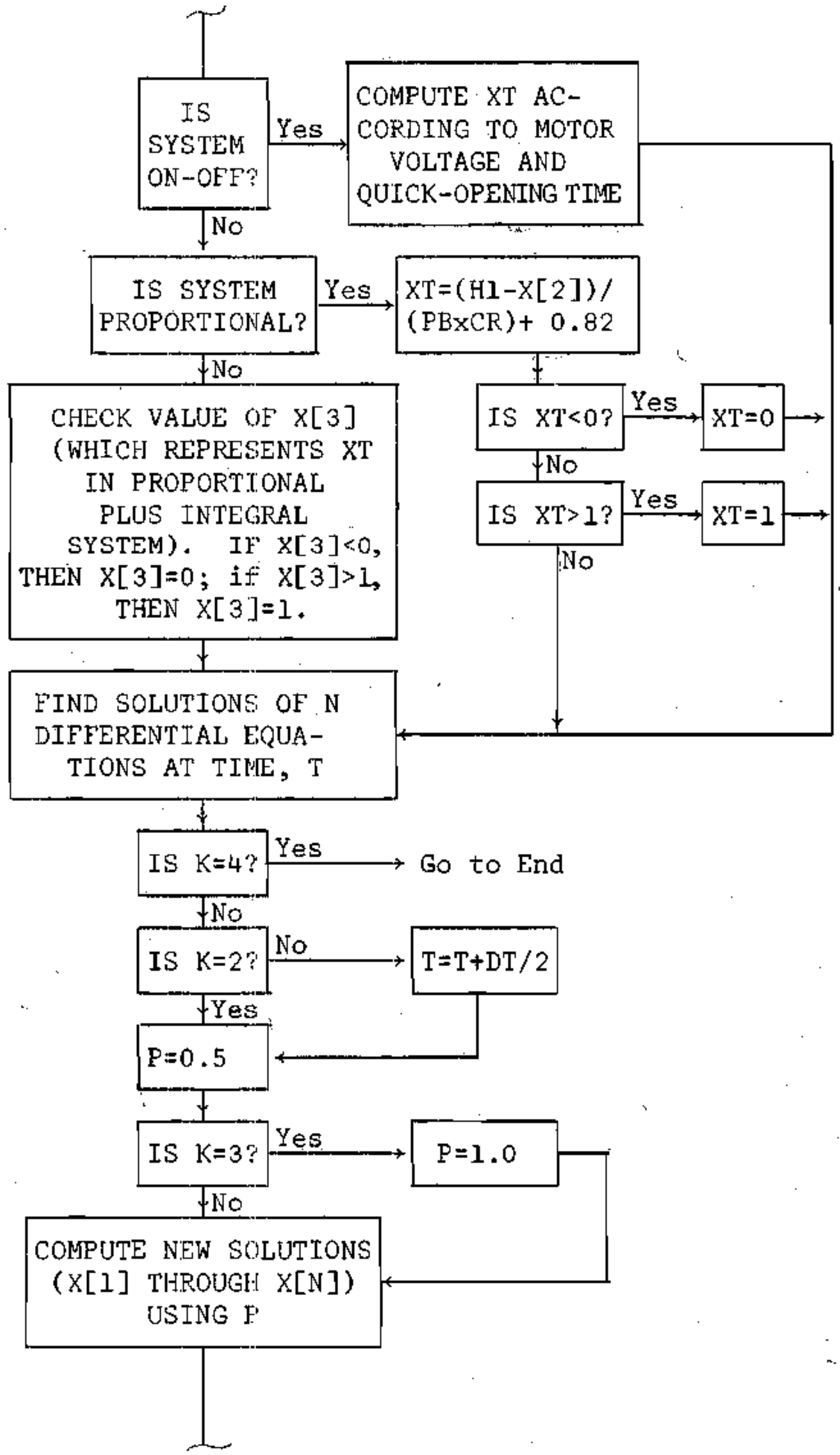

ELIMINATE ANY VALVES THAT ARE NOT ELECTRIC
HAVE ALL REMAINING VALVES BEEN ELIMINATED?
Yes
PRINT "NO ELECTRIC VALVES AVAILABLE?
Go to end of program
No
FIND TOTAL COST OF EACH REMAINING VALVE
ARRANGE VALVES IN ORDER OF COST
IS SYSTEM ELECTRIC OR PROPORTIONAL PLUS INTEGRAL?
No
IS TANK VENTED?
No
Yes
IS H1≤7.2/G AND 0.25<PB≤0.35?
No
Yes
TAKE EACH VALVE, IN ORDER OF COST, AND ENTER RUNGKUTTA WITH APPLICABLE VALVE C_V UNTIL TWO VALVES HAVE BEEN FOUND WHICH HAVE ADEQUATE RESPONSE OR UNTIL LIST OF VALVES IS EXHAUSTED
HAVE ANY VALVES MET DESIGN REQUIREMENTS?
Yes
No
PRINT "NO VALVES HAVE ADEQUATE RESPONSE"
Go to end of program

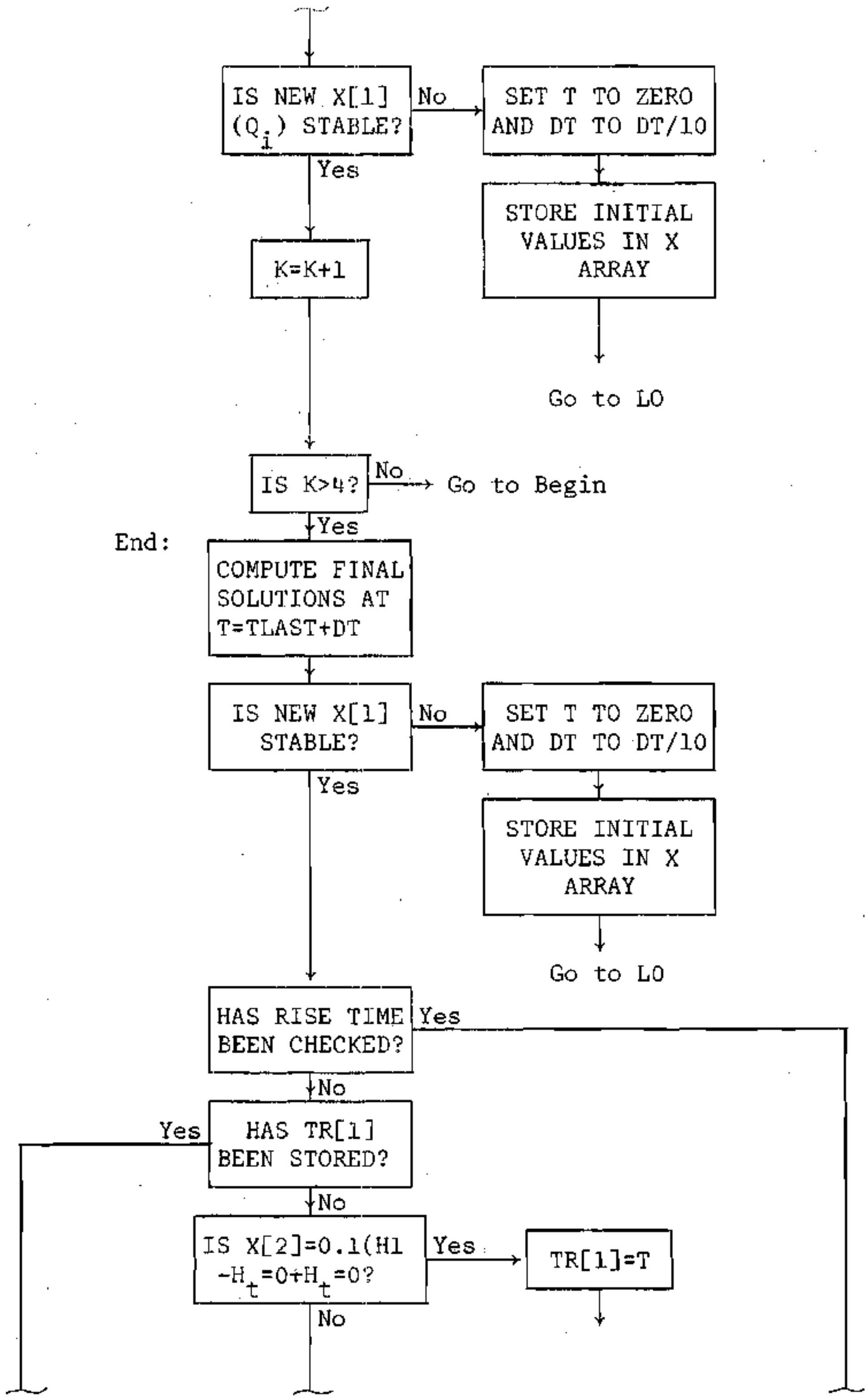

IS SYSTEM ON-OFF?
Yes
PRINT DH AND PROPER CON-TROLLER AND PROBES WITH EACH OF TWO HALVES
Go to end of program
No
IS SYSTEM PROPORTIONAL PLUS INTEGRAL?
Yes
No
IS SYSTEM PNEUMATIC?
No
IS H1≤14.4/G?
No
PRINT "NO TRANSMITTER AVAILABLE"
Go to end of program
Yes
IS FLUID COMPATIBLE WITH CARBON STEEL OR 316 STAINLESS STEEL?
No
Yes
PRINT PB AND PROPER LEVEL SENSOR, CONTROLLER, AND POWER SUPPLY WITH EACH OF TWO VALVES
Yes
IS TANK VENTED?
Yes
No
IS H1>7.2/G?
Yes
No
L11:
IS FLUID COMPATIBLE WITH CAST IRON?
Yes
IS TANK RATING ≤125 or ≤250?
No
IS FLUID COM-PATIBLE WITH CARBON STEEL?
No
PRINT "WARNING-SPECIAL DISPLACER REQUIRED"
Yes
IS TANK RATING ≤300 OR ≤600?
No
PRINT "ORDER SPECIAL DISPLACER"

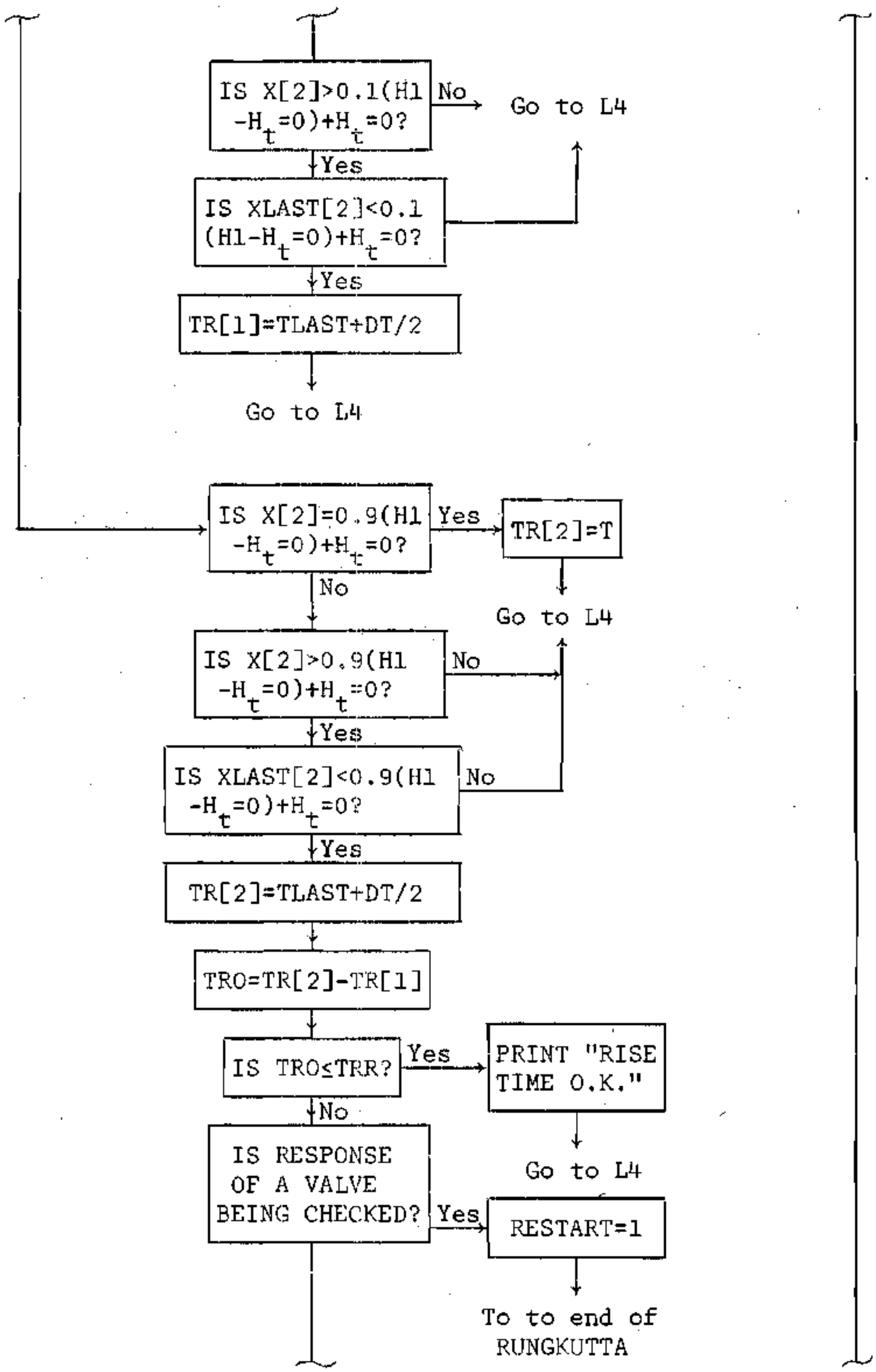

PICK PROPER LEVEL SENSOR COST FROM LEVEL SENSOR TABLE
PRINT PB AND PROPER LEVEL SENSOR-CONTROLLER COMBINATION WITH EACH OF TWO VALVES
Go to end of program
IS SYSTEM PROPORTIONAL PLUS INTEGRAL?
Yes
IS STATIC PRESSURE SENSOR DESIRED?
No
Go to L11
No
Yes
Go to L11
No
IS 0.25<PB≤0.35?
Yes
PRINT PB, RR, AND PROPER LEVEL SENSOR AND VERTICAL SCALE CONTROL WITH EACH OF TWO VALVES
PRINT PB AND LOW-COST SYSTEM B WITH EACH OF TWO VALVES
No
HAS MODEL 1405, 4705, 9105, OR 9205 BEEN SELECTED AS ONE OR BOTH OF THE CHEAPEST VALVES?
Go to end of program
Yes
Go to end of program
ENTER RUNGEKUTTA WITH PROPER CV OF VALVE OR VALVES
DOES AT LEAST ONE VALVE HAVE ADEQUATE RESPONSE?
No
Yes
PRINT PB AND LOW-COST SYSTEM A WITH EACH OF TWO VALVES (OR ONE VALVE IF ONLY ONE HAD ADEQUATE RESPONSE)
END

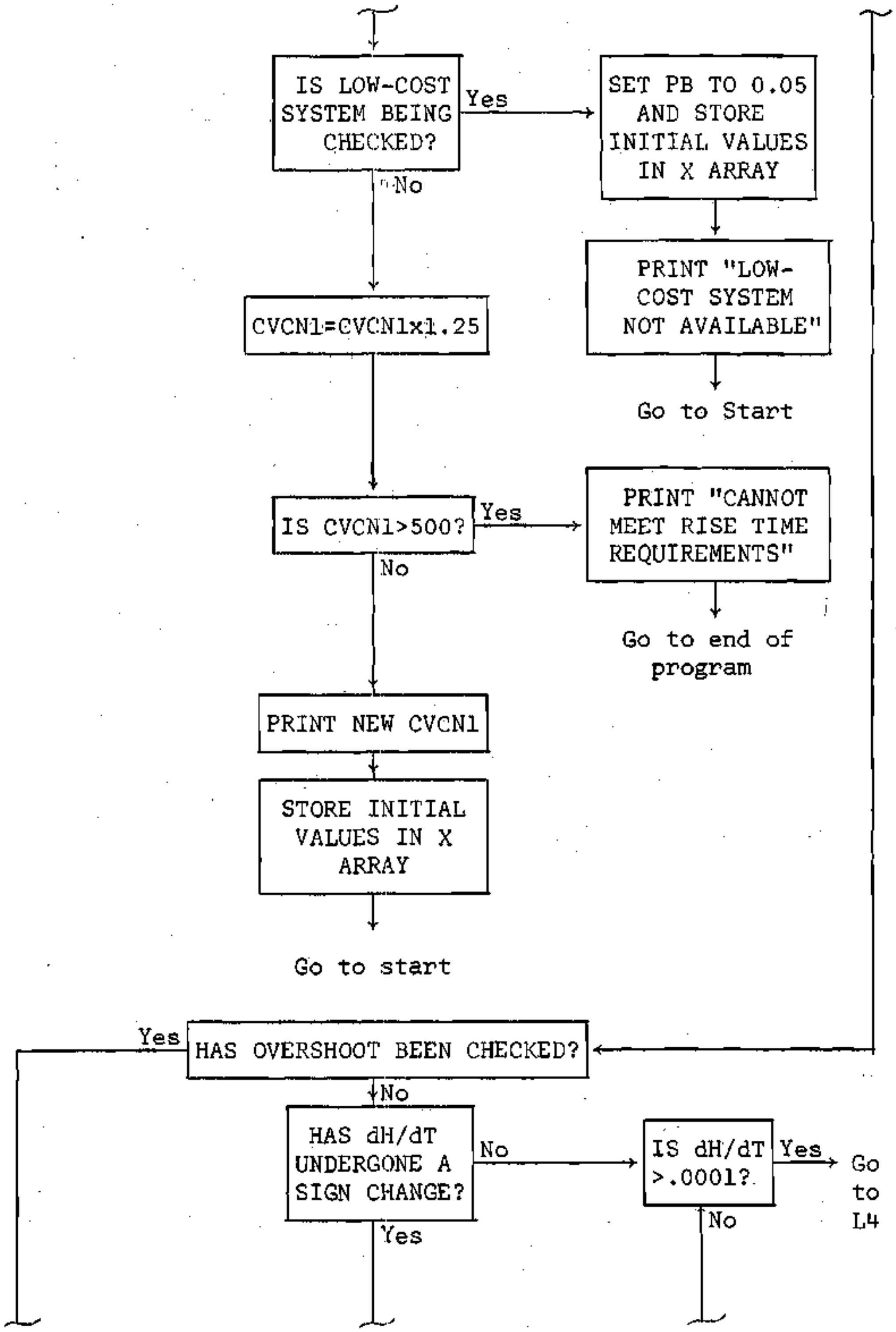

SET M EQUAL TO (4)(TRR)
IS RESPONSE OF A VALVE BEING CHECKED?
No
PB=0.30 (LOW-COST SYSTEM)
Yes
Start:
SET ALL TIMES TO ZERO AND DT TO 0.01
IS SYSTEM PROPORTIONAL PLUS INTEGRAL?
No
IS CONFIG=1?
Yes
N=2
Yes
IS CONFIG=1?
Yes
No
N=3
No
N=4
COMPUTE ALL CONSTANTS THAT WILL BE USED IN EQUATIONS FOR REYNOLDS NUMBERS AND IN DIFFERENTIAL EQUATIONS
LO:
STORE PREVIOUS SOLUTIONS (X[1] THROUGH X[N]) OF DIFFERENTIAL EQUATIONS
STORE SIGN OF PREVIOUS dH/dT

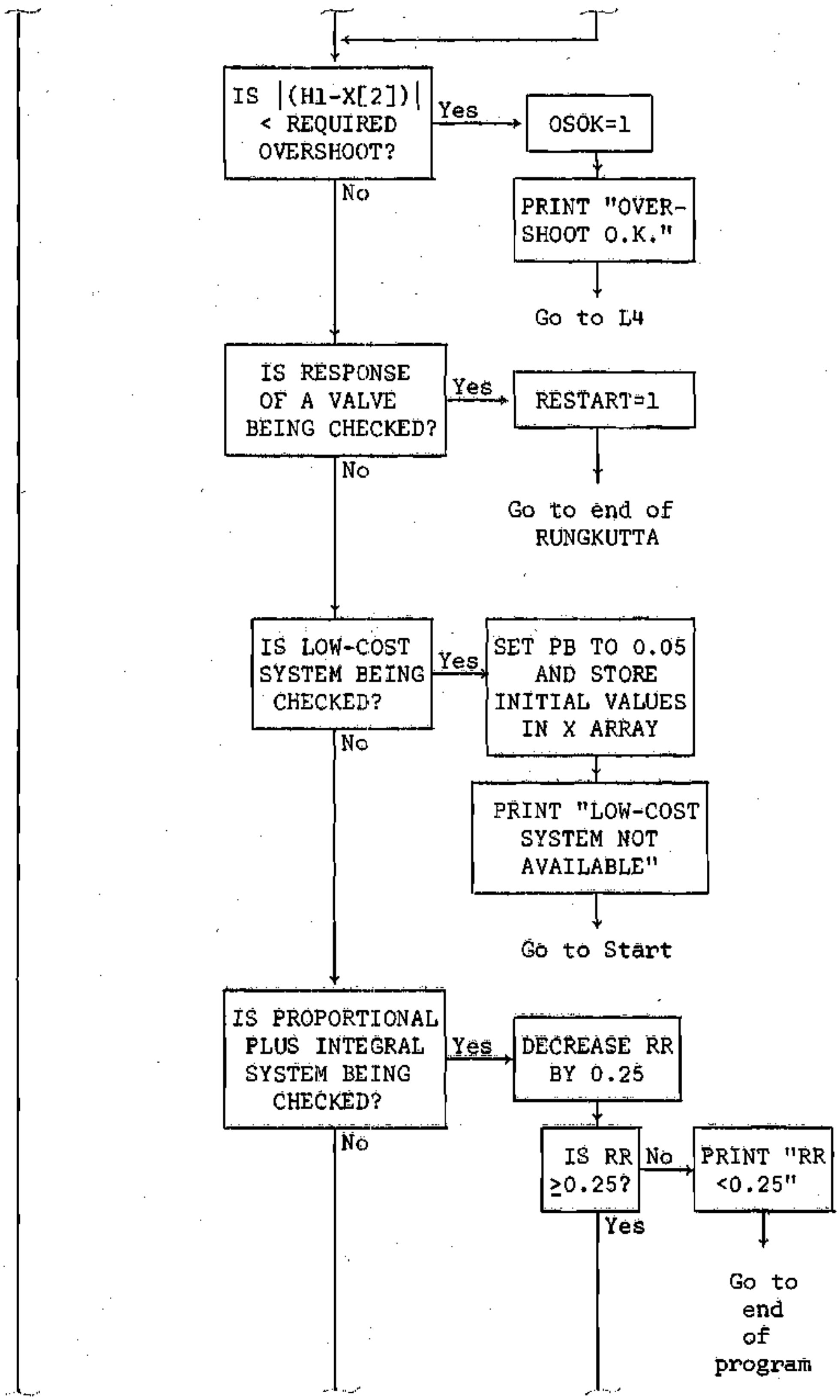

K=1
Begin:
COMPUTE INLET PIPE REYNOLDS NUMBER (RP1) AND CONTROL VALVE REYNOLDS NUMBER (RV1). (RV1 0 IF VALVE HAS NOT BEEN SELECTED YET)
IS RP1 < 200?
Yes
PRINT VALUE OF RP1
Go to end of program
No
COMPUTE F2
IS X[2]>H3?
No
H2=ZERO
Yes
H2=X]2]-H3
COMPUTE P_0 IF TANK IS VENTED: OTHERWISE RETAIN PREVIOUS P_0
IS RESPONSE OF A VALVE BEING CHECKED?
Yes
RE1=RV1
No
RE1=RP1
COMPUTE VALVE SIZE COEFFICIENT AT RE1 USING CVCN1

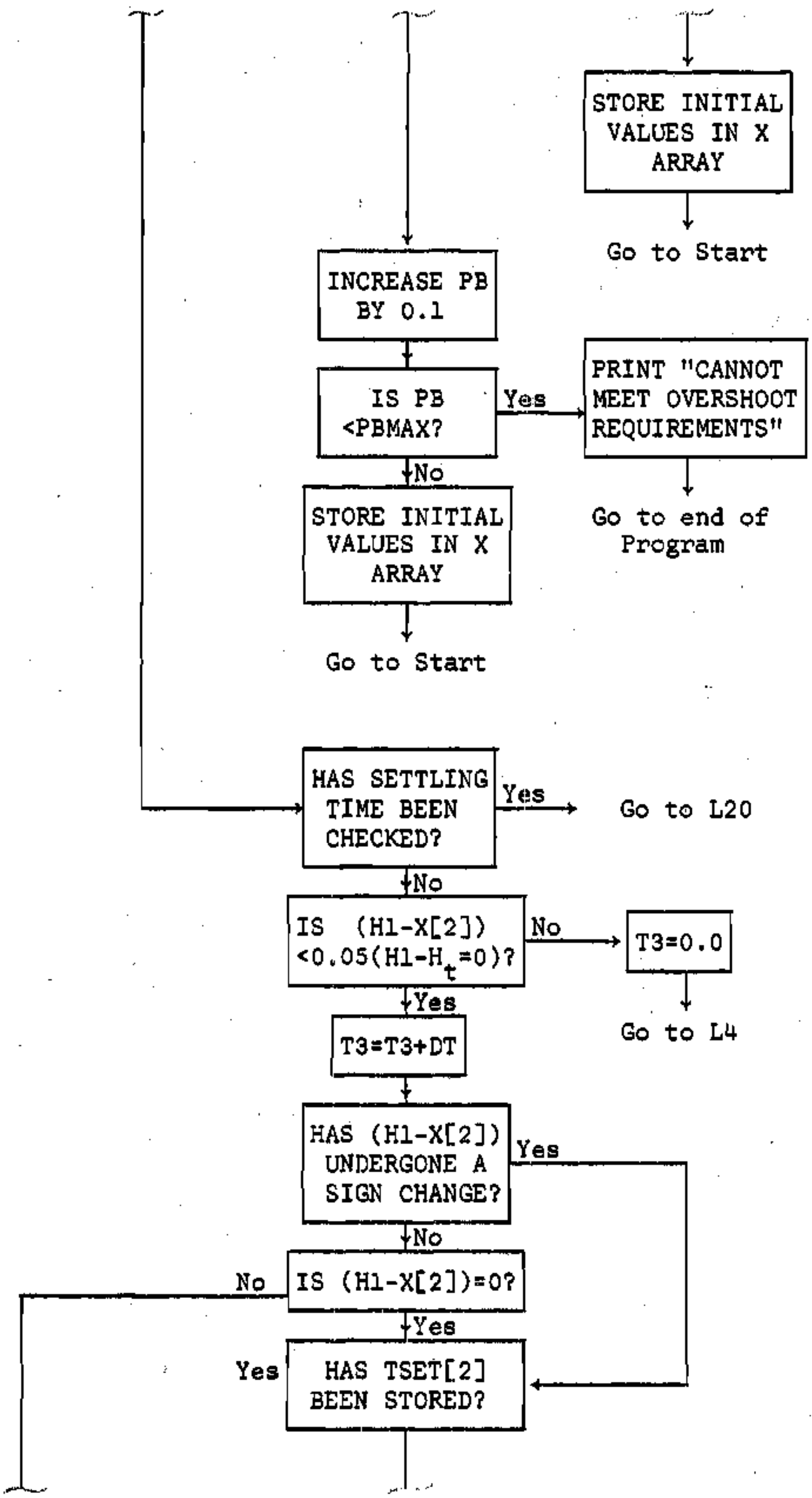

IS SYSTEM ON-OFF?
Yes
COMPUTE XT ACCORDING TO MOTOR VOLTAGE AND QUICK-OPENING TIME
No
IS SYSTEM PROPORTIONAL?
Yes
XT=(H1-X[2])/(PBxCR)+ 0.82
No
CHECK VALUE OF X[3] (WHICH REPRESENTS XT IN PROPORTIONAL PLUS INTEGRAL SYSTEM). IF X[3]<0, THEN X[3]=0; if X[3]>1, THEN X[3]=1.
IS XT<0?
Yes
XT=0
No
IS XT>1?
Yes
XT=1
No
FIND SOLUTIONS OF N DIFFERENTIAL EQUATIONS AT TIME, T
IS K=4?
Yes
Go to End
No
IS K=2?
No
T=T+DT/2
Yes
P=0.5
IS K=3?
Yes
P=1.0
No
COMPUTE NEW SOLUTIONS (X[1] THROUGH X[N]) USING P


IS NEW X[1] (Qi) STABLE?
No
SET T TO ZERO AND DT TO DT/10
Yes
STORE INITIAL VALUES IN X ARRAY
K=K+1
Go to L0
IS K>4?
No
Go to Begin
Yes
End:
COMPUTE FINAL SOLUTIONS AT T=TLAST+DT
IS NEW X[1] STABLE?
No
SET T TO ZERO AND DT TO DT/10
Yes
STORE INITIAL VALUES IN X ARRAY
Go to L0
HAS RISE TIME BEEN CHECKED?
Yes
No
Yes
HAS TR[1] BEEN STORED?
No
IS X[2]=0.1(H1 -Ht=0+Ht=0?
Yes
TR[1]=T
No

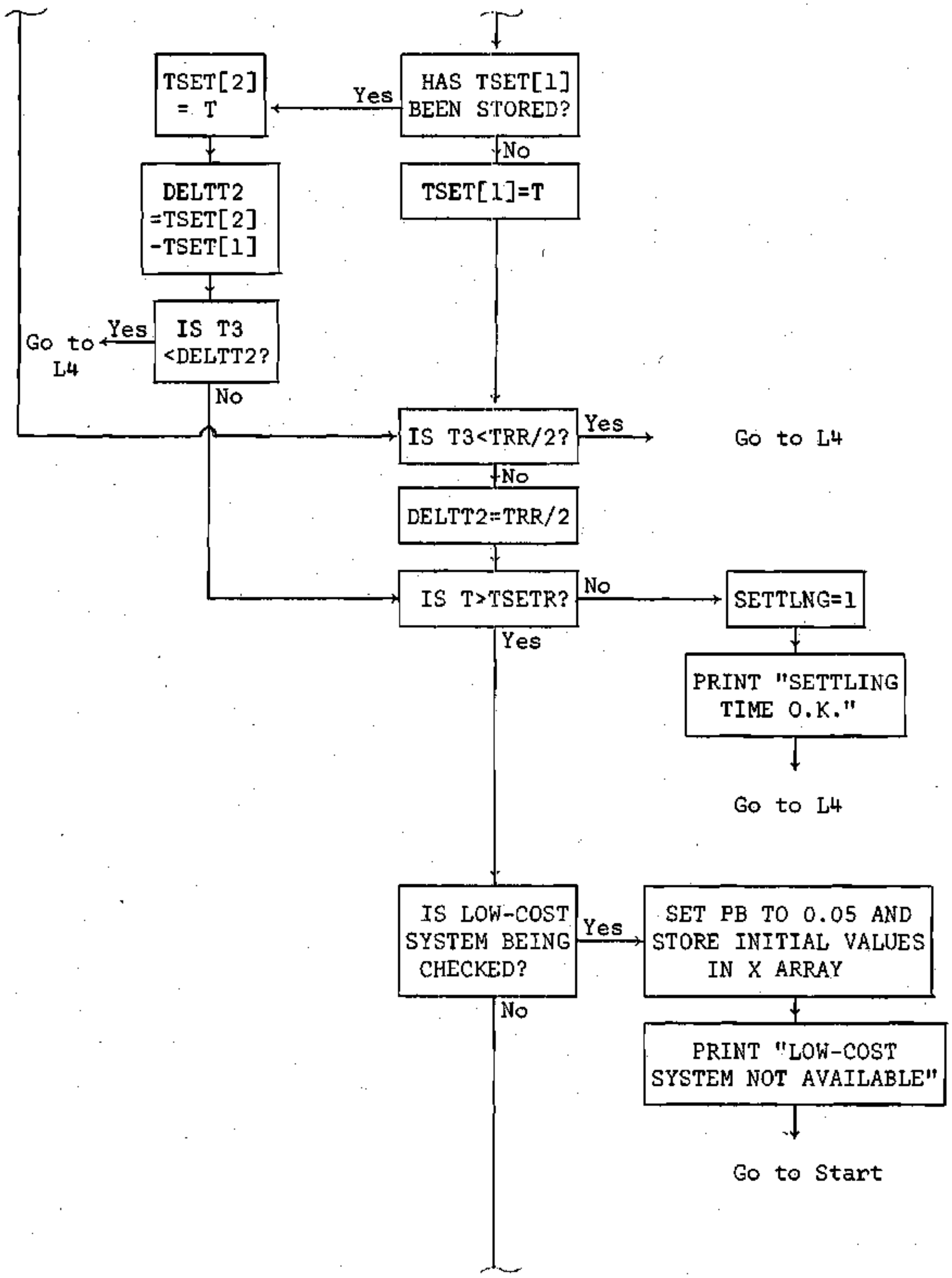

IS X[2]>0.1(H1 -Ht=0)+Ht=0?
No
Go to L4
Yes
IS XLAST[2]<0.1 (H1-Ht=0)+Ht=0?
Yes
TR[1]=TLAST+DT/2
Go to L4
IS X[2]=0.9(H1 -Ht=0)+Ht=0?
Yes
TR[2]=T
Go to L4
No
IS X[2]>0.9(H1 -Ht=0)+Ht=0?
No
Yes
IS XLAST[2]<0.9(H1 -Ht=0)+Ht=0?
No
Yes
TR[2]=TLAST+DT/2
TRO=TR[2]-TR[1]
IS TRO≤TRR?
Yes
PRINT "RISE TIME O.K."
Go to L4
No
IS RESPONSE OF A VALVE BEING CHECKED?
Yes
RESTART=1
To to end of RUNGKUTTA


IS LOW-COST SYSTEM BEING CHECKED?
Yes
SET PB TO 0.05 AND STORE INITIAL VALUES IN X ARRAY
PRINT "LOW-COST SYSTEM NOT AVAILABLE"
Go to Start
No
CVCN1=CVCN1x1.25
IS CVCN1>500?
Yes
PRINT "CANNOT MEET RISE TIME REQUIREMENTS"
Go to end of program
No
PRINT NEW CVCN1
STORE INITIAL VALUES IN X ARRAY
Go to start
HAS OVERSHOOT BEEN CHECKED?
Yes
No
HAS dH/dT UNDERGONE A SIGN CHANGE?
No
IS dH/dT >.0001?
Yes
Go to L4
No
Yes

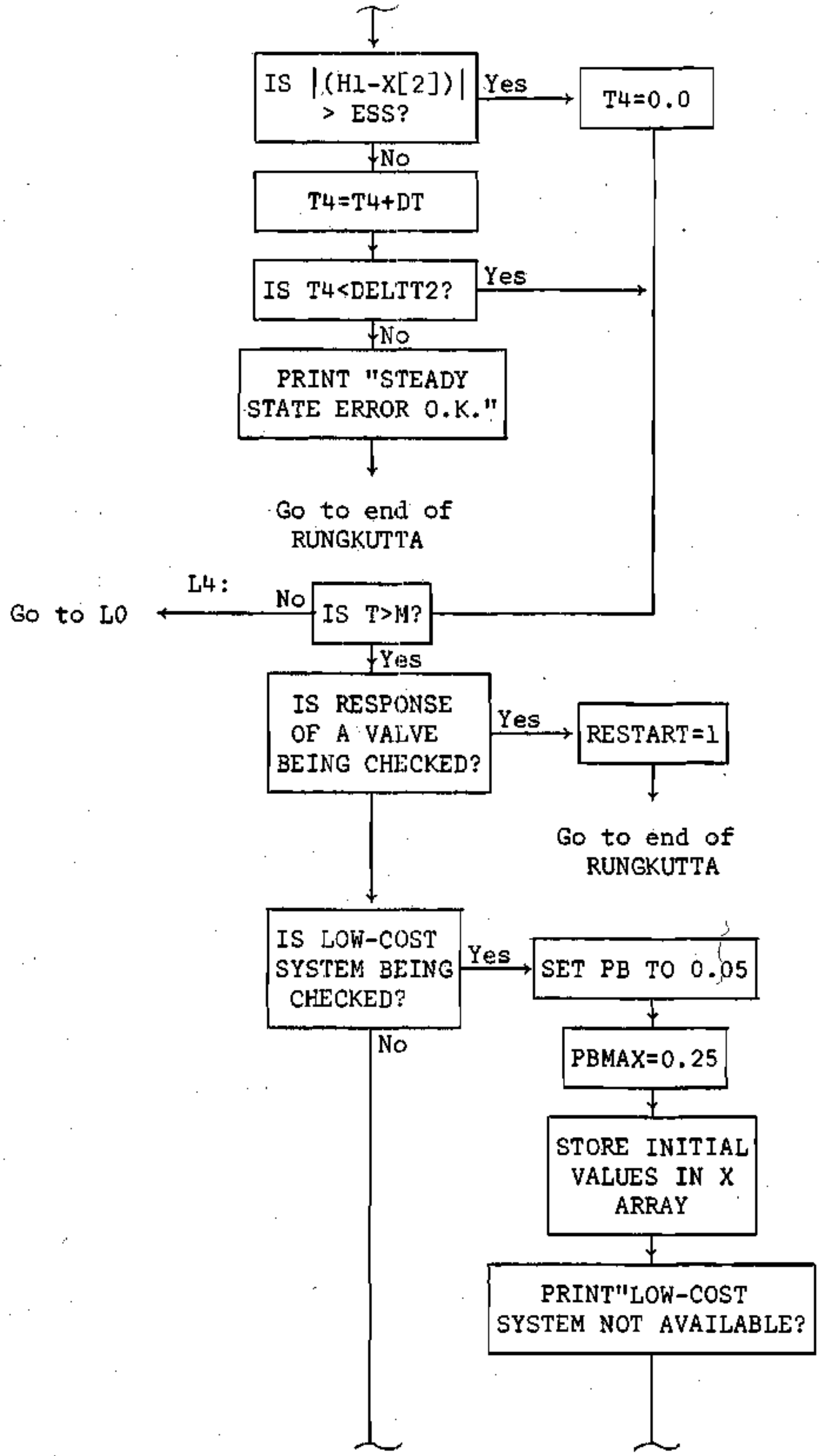

IS |(H1-X[2])| < REQUIRED OVERSHOOT?
Yes
No
OSOK=1
PRINT "OVER-SHOOT O.K."
Go to L4
IS RESPONSE OF A VALVE BEING CHECKED?
Yes
No
RESTART=1
Go to end of RUNGKUTTA
IS LOW-COST SYSTEM BEING CHECKED?
Yes
No
SET PB TO 0.05 AND STORE INITIAL VALUES IN X ARRAY
PRINT "LOW-COST SYSTEM NOT AVAILABLE"
Go to Start
IS PROPORTIONAL PLUS INTEGRAL SYSTEM BEING CHECKED?
Yes
No
DECREASE RR BY 0.25
IS RR ≥0.25?
No
Yes
PRINT "RR <0.25"
Go to end of program

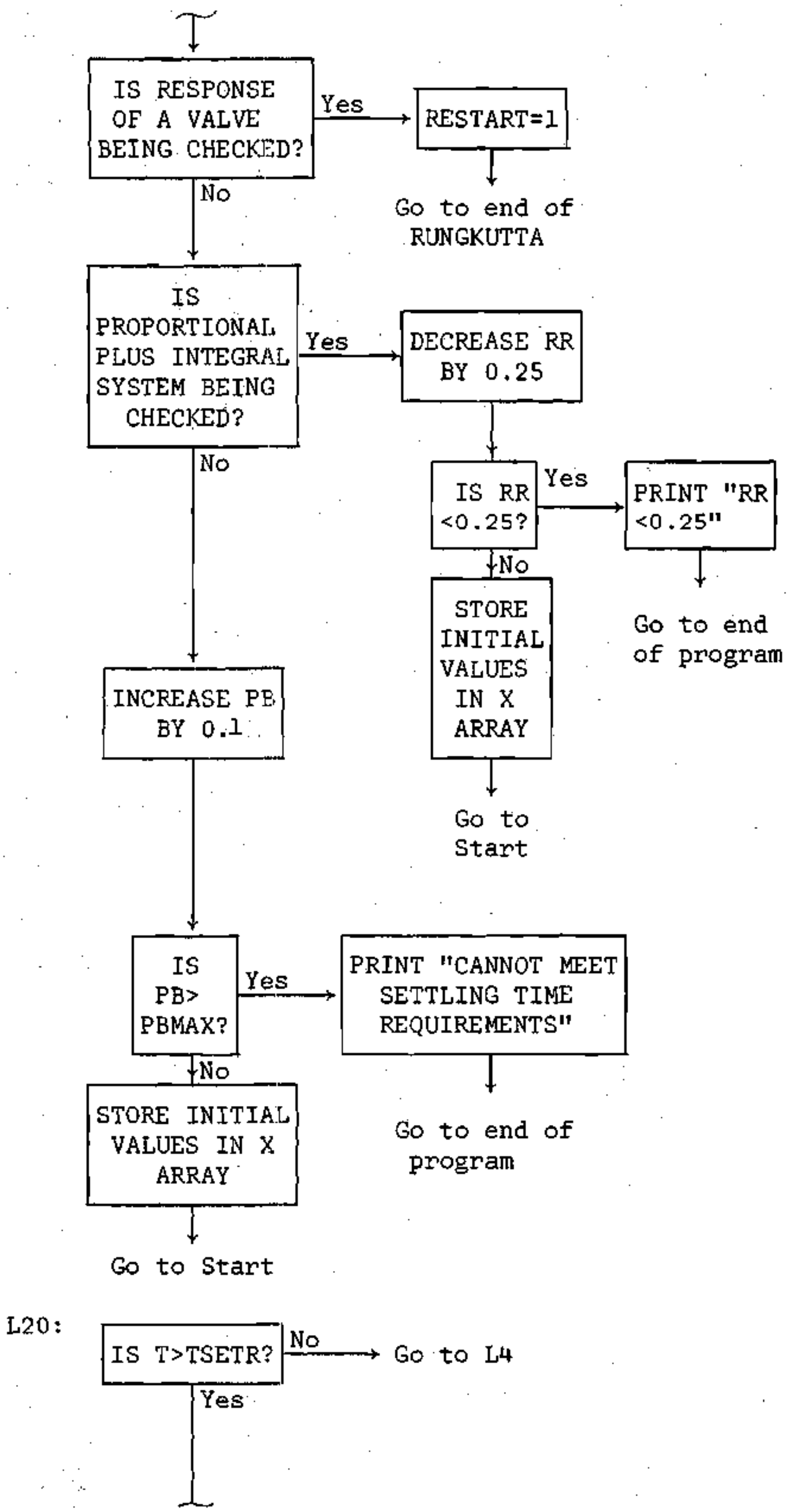


STORE INITIAL VALUES IN X ARRAY
Go to Start
INCREASE PB BY 0.1
IS PB <PBMAX?
Yes
PRINT "CANNOT MEET OVERSHOOT REQUIREMENTS"
Go to end of Program
No
STORE INITIAL VALUES IN X ARRAY
Go to Start
HAS SETTLING TIME BEEN CHECKED?
Yes
Go to L20
No
IS (H1-X[2]) <0.05(H1-Ht=0)?
No
T3=0.0
Go to L4
Yes
T3=T3+DT
HAS (H1-X[2]) UNDERGONE A SIGN CHANGE?
Yes
No
IS (H1-X[2])=0?
No
Yes
HAS TSET[2] BEEN STORED?
Yes

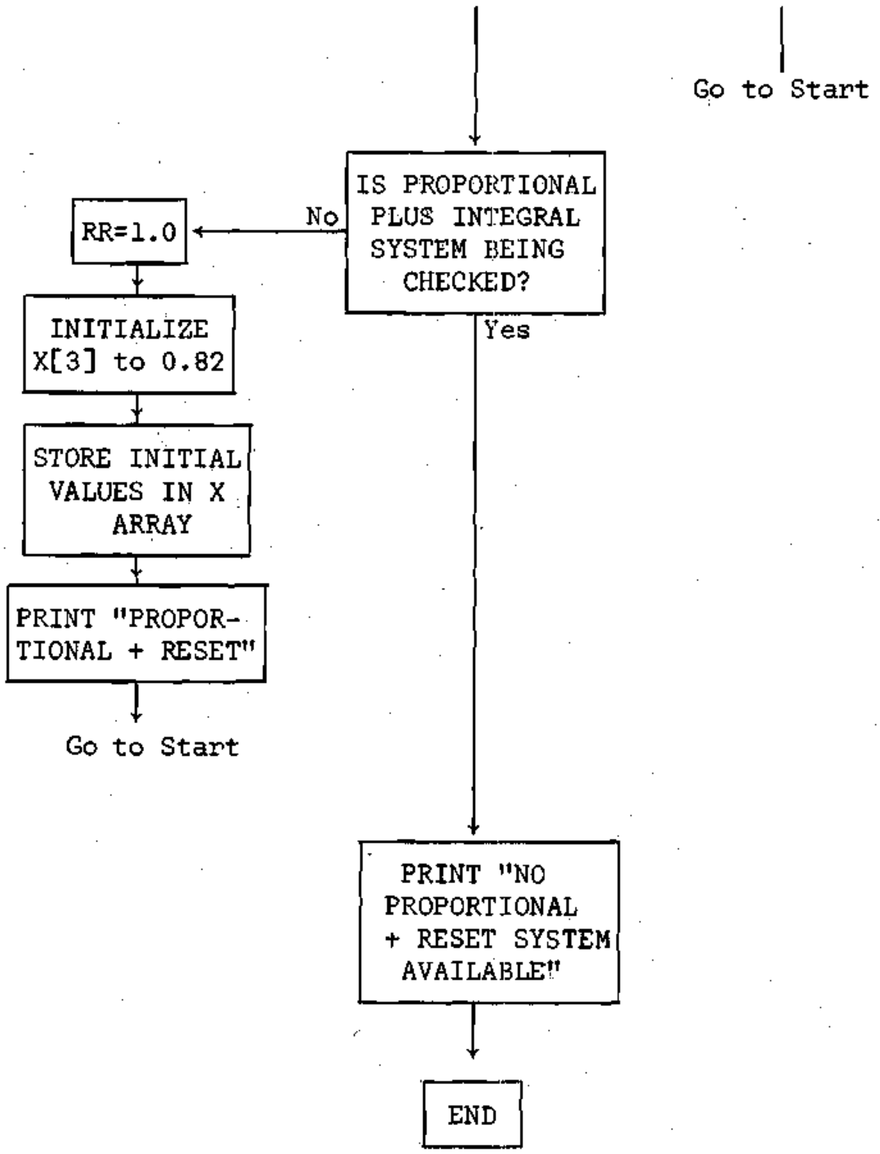


HAS TSET[1] BEEN STORED?
Yes
TSET[2] = T
No
TSET[1]=T
DELTT2 =TSET[2] -TSET[1]
IS T3 <DELTT2?
Yes
Go to L4
No
IS T3<TRR/2?
Yes
Go to L4
No
DELTT2=TRR/2
IS T>TSETR?
No
SETTLNG=1
PRINT "SETTLING TIME O.K."
Go to L4
Yes
IS LOW-COST SYSTEM BEING CHECKED?
Yes
SET PB TO 0.05 AND STORE INITIAL VALUES IN X ARRAY
PRINT "LOW-COST SYSTEM NOT AVAILABLE"
Go to Start
No


IS |(H1-X[2])| > ESS?
Yes
T4=0.0
No
T4=T4+DT
IS T4<DELTT2?
Yes
No
PRINT "STEADY STATE ERROR O.K."
Go to end of RUNGKUTTA
L4:
No
Go to L0
IS T>M?
Yes
IS RESPONSE OF A VALVE BEING CHECKED?
Yes
RESTART=1
Go to end of RUNGKUTTA
IS LOW-COST SYSTEM BEING CHECKED?
Yes
SET PB TO 0.05
No
PBMAX=0.25
STORE INITIAL VALUES IN X ARRAY
PRINT"LOW-COST SYSTEM NOT AVAILABLE?



IS RESPONSE OF A VALVE BEING CHECKED?
Yes
RESTART=1
Go to end of RUNGKUTTA
No
IS PROPORTIONAL PLUS INTEGRAL SYSTEM BEING CHECKED?
Yes
DECREASE RR BY 0.25
IS RR <0.25?
Yes
PRINT "RR <0.25"
Go to end of program
No
STORE INITIAL VALUES IN X ARRAY
Go to Start
No
INCREASE PB BY 0.1
IS PB> PBMAX?
Yes
PRINT "CANNOT MEET SETTLING TIME REQUIREMENTS"
Go to end of program
No
STORE INITIAL VALUES IN X ARRAY
Go to Start
L20:
IS T>TSETR?
No
Go to L4
Yes


Go to Start
IS PROPORTIONAL PLUS INTEGRAL SYSTEM BEING CHECKED?
No
Yes
RR=1.0
INITIALIZE X[3] to 0.82
STORE INITIAL VALUES IN X ARRAY
PRINT "PROPORTIONAL + RESET"
Go to Start
PRINT "NO PROPORTIONAL + RESET SYSTEM AVAILABLE"
END

APPENDIX F

COMPLETE COMPUTER PROGRAM

The complete computer program is given on the following pages.

```
BEGIN                                                                      0000
                                                    START OF SEGMENT ********** 0002
COMMENT TEST CASE;                                                         0000
FILE IN CARD (1,10), TAPE1 "VALVES" (2,1023);                              0000
FILE OUT PRINT 1(2,15);                                                    0010
INTEGER A,I,J,K,L,M,NVALVS,OX1,DUM,SEALI,MONEL,EXTCOL,RADFIN,              0015
 SLURRY,IND,CAPCTY,COMPAT,DP,TOP,MEON,J1,K1,STEEL,PLAIN,                   0015
 TANK,ONOFF,CONFIG,NEXTV,CR,SAUNDERS,PNEU,SPECIAL,STOPX,                   0015
 INITL,IFINAL,STATIC;                                                      0015
REAL DELTPI,P1MAX,P4MIN,TF,P15,P30,P40,P60,CV1,CVMIN,CVMAX,FMIN,FMAX,Z,    0015
 SUML,RO,D1,D3,H1,H3,ODM,P1,P0,H2,G,U,E,P13,L1,K20,P7,P8,X1M,X2M,          0015
 CVCN1,CVCN3,CV3,RP1,F2,F4,PB,PU,FLOUT,AB,R,RR,DS,DH,TRR,TRD,DV3,          0015
 TNKRATNG,NMRTR,ESS,P00,PBMAX,TSETR,DV1,COST,N1,N2,O6,TOTAL,COMPONENT;     0015
INTEGER ARRAY NBI[0:4],NTX[0:10],B[0:10],RATNGI[0:4,0:2],NTI[0:10],        0015
 MODEL,VPORT,OO,VPFC,VPRC,BODY,EXT,RAD,SEAL,LINVFC,LINVRC,NUMCVS[0:100],   0025
 ACTCOST[0:100,0:4,0:2],NB[0:100,0:3],RATNB,BCOST[0:100,0:3,0:4],          0030
 NBON,BONCOST[0:100,0:10],NT,TMCOST[0:100,0:6],AVAIL[0:100],ONIT[0:25],    0037
 MCOST[0:100,0:4],NBX,RATNGX,COSTX[0:4,0:2],NFIT[0:22],DISPL[0:9],         0046
 DISCOST[0:9,0:4],POSCOST[0:100,0:2],COL[0:100],CHEAP,INDX1,INDX2,         0055
 TMCOL,MAN,AVAILX[0:25];                                                   0063
REAL ARRAY CV[0:100,0:4],DELTP,SPRMIN,SPRMAX,AIR,ACTNO[0:100,0:4,0:2],     0065
 SIZE,TYPE,VPLINK,OOLINK,TRAVL,VPTIME,OOTIME,TMIN,TMAX[0:100],LT[0:5],     0071
 LO[0:22],LE[0:5],ORIGX[0:4],X[0:4],POS[0:100,0:2],NAME[0:2,1:10],         0077
 TOTCOST[0:25];                                                            0089
FORMAT FL1000("INVESTIGATE POSSIBILITY OF LINED SAUNDERS VALVES"),         0091
                                                    START OF SEGMENT ********** 0003
       FL1001("NO BONNET AVAILABLE"),                                      0091
       FL1003("DELTP BEYOND RANGE"),                                       0091
       FL1004("NO CV AVAILABLE"),                                          0091
       FL1005("NO BODY MATERIAL AVAILABLE"),                               0091
       FL1006("NO FLANGE SIZE AVAILABLE"),                                 0091
       FL1007("NO VALVE W/ALLOWABLE DELTP AVAILABLE"),                     0091
```

```
          FL1008("NO BONNET AVAILABLE"),                                        0091
          FL1009("NO TRIM AVAILABLE"),                                          0091
          FL1010("NO MANUAL OPERATOR AVAILABLE"),                               0091
          FL2000("PROGRAM TIME =",F10.2," SECONDS"),                            0091
          FL2010("FLUID NOT COMPATIBLE W/MONEL OR 316 STEEL"),                  0091
          FL2011("FLUID NOT COMPATIBLE W/316 STEEL"),                           0091
          FL2012("FIRST RP1 = ",E15.5),                                         0091
          FL2013("RUNGKUTTA RP1 = ",E15.5),                                     0091
          FL2014("RUNGKUTTA RP3 = ",E15.5),                                     0091
          FL3000("NO ELECTRIC VALVES AVAILABLE"),                               0091
          FL20("CVCNI = ",F8.2),                                                0091
          FL2(I4,F7.2,A5,I5,I4,X2,A6,F8.2,X2,A6,F8.2,2F7.2,F8.4/),              0091
          FL3(F6.2,2F6.1,F8.1,X6,F5.1,X11,A6,I9/),                              0091
          FL4(I5,X4,2I7/),                                                      0091
          FL5(I3,X1,2I7,I8/),                                                   0091
          FL6(I6,X6,I6/),                                                       0091
          FL7(I5,X5,I5/),                                                       0091
          FL8("TOTAL COST = ",I6/),                                             0091
          FL9(3F6.2,6F7.2,F8.2,F7.2),                                           0091
          FL10(20I4),                                                           0091
          FL11(4F7.2,F6.2,6I2,2E9.2,F5.3,F5.2),                                 0091
          FL12(21I3),                                                           0091
          FL13(3F8.2,F6.1,I4,F6.1,F7.1,F6.3,F6.2,I2),                           0091
          FL1(2I2,I4,I2,F6.1,I2,7F7.2);                                         0091
                                                    0003 IS 0247 LONG, NEXT SEG 0002
      PROCEDURE TAKOUT;                                                         0091
      BEGIN                                                                     0091
      INTEGER I,K,TEMP;                                                         0091
                                                    START OF SEGMENT ********** 0004
      LABEL L1,L2;                                                              0000
      I+1;                                                                      0000
      TEMP+COMPAT;                                                              0000
```

```
L1: IF UNIT[I]#1 THEN BEGIN                                   0001
 TEMP+TEMP-1;                                                 0003
 IF I>COMPAT-1 THEN GO TO L2;                                 0004
 FOR K+I STEP 1 UNTIL COMPAT-1 DO BEGIN                       0006
  AVAIL[K]+AVAIL[K+1]; UNIT[K]+UNIT[K+1];                     0013
 COL[K]+COL[K+1]; MAN[K]+MAN[K+1];                            0017
 END;                                                         0021
 AVAIL[TEMP+1]+0; UNIT[TEMP+1]+0;                             0021
 COL[TEMP+1]+0; MAN[TEMP+1]+0;                                0025
 GO TO L1;                                                    0028
END;                                                          0029
I+I+1;                                                        0029
IF I≤COMPAT THEN GO TO L1;                                    0030
L2: COMPAT+TEMP;                                              0031
END TAKOUT;                                                   0032
                                        0004 IS 0035 LONG, NEXT SEG 0002
PROCEDURE NTRIM;                                              0091
BEGIN                                                         0091
INTEGER J;                                                    0091
                                        START OF SEGMENT ********** 0005
LABEL L1,L2;                                                  0000
FOR J+1 STEP 1 UNTIL 10 DO BEGIN                              0000
 IF M#ABS(NTX[J]) THEN GO TO L1;                              0001
 IND+1;                                                       0002
 M+NTX[J];                                                    0003
 GO TO L2;                                                    0004
 L1: IND+0;                                                   0005
END;                                                          0005
L2: END NTRIM;                                                0008
                                        0005 IS 0010 LONG, NEXT SEG 0002
PROCEDURE CVS(LIMIT);                                         0091
VALUE LIMIT; INTEGER LIMIT;                                   0091
```

```
BEGIN                                                                          0091
INTEGER K;                                                                     0091
                                                       START OF SEGMENT ********** 0006
FOR K←J+1 STEP 1 UNTIL J+LIMIT DO                                              0000
 READ(TAPE1,*,CV[I,K],FOR DX←1 STEP 1 UNTIL 2 DO[DELTP[I,K,DX],                0007
 SPHMIN[I,K,DX],SPHMAX[I,K,DX],AIR[I,K,DX],ACTNO[I,K,DX],                      0018
 ACTCOST[I,K,DX]]);                                                            0030
END CVS;                                                                       0039
                                                       0006 IS 0041 LONG, NEXT SEG 0002
PROCEDURE PRINTZ;                                                              0091
BEGIN                                                                          0091
FORMAT FL8(X21,"BODY MATERIAL ",2A6),                                          0091
                                                       START OF SEGMENT ********** 0007
                                                       START OF SEGMENT ********** 0008
       FL9(X22,"(SCREWED END)"),                                               0000
       FL10(X22,"(FLANGED END)"),                                              0000
       FL11(X22,"(PROCEED WITH CAUTION)"),                                     0000
       FL12(X21,"BODY RATING",I4," LBS.",X7,"$",F7.2),                         0000
       FL13(X21,"PLAIN BONNET"),                                               0000
       FL14(X21,"BELLOWS SEAL = ",2A6," $",F6.2),                              0000
       FL15(X21,"EXTENSION COLUMN",X12,"$",F6.2),                              0000
       FL16(X21,"RADIATING FIN",X15,"$",F6.2),                                 0000
       FL17(X21,"TRIM MATERIAL ",2A6," $",F6.2),                               0000
       FL22(X21,"MAN. OPERATOR (SIDE)",X8,"$",F6.2),                           0000
       FL23(X21,"MAN. OPERATOR (TOP)",X9,"$",F6.2),                            0000
       FL24(X21,"POSITIONER (SIDE)  ",A6,X3,"$",F6.2),                         0000
       FL25(X21,"POSITIONER (TOP)  ",A6,X4,"$",F6.2),                          0000
       FL20(X36,"TOTAL COST $",F8.2);                                          0000
                                                       0008 IS 0139 LONG, NEXT SEG 0007
LABEL L3,L4,L5,L6,L7,L8,L9,L10;                                                0000
FOR I←1 STEP 1 UNTIL COMPAT DO IF J=AVAIL[I] THEN BEGIN                       0000
 J←INVAL[I]; K←INUM2[I]; END;                                                  0002
```

```
COST←BCOST[J,J1,K1]; TOTAL←TOTAL+COST;                                 0006
I←NB[J,J1];                                                            0010
WRITE(PRINT,FL8,NAME[1,I],NAME[2,I]);                                  0012
FOR I←1 STEP 1 UNTIL 10 DO IF NB[J,J1]=ABS(NB1[I]) THEN GO TO L3;      0025
L3:  IF NB1[I]<0 THEN WRITE(PRINT,FL11);                               0031
WRITE(PRINT,FL12,ABS(RATNG[J,J1,K1]),COST);                            0036
IF RATNG[J,J1,K1]<0 THEN WRITE(PRINT,FL9) ELSE WRITE(PRINT,FL10);      0048
IF PLAIN=1 THEN BEGIN WRITE(PRINT,FL13); GO TO L4; END;                0058
IF SEAL1=0 THEN GO TO L5;                                              0064
IF STEEL=0 THEN BEGIN FOR I←4,5,6 DO IF NB[J,J1]=NBON[J,I] THEN M←I;   0065
 END ELSE BEGIN FOR I←7,8,9 DO IF NB[J,J1]=NBON[J,I] THEN M←I; END;    0077
COST←BONCOST[J,M]; TOTAL←TOTAL+COST;                                   0089
WRITE(PRINT,FL14,IF STEEL=0 THEN NAME[1,5] ELSE NAME[1,4],             0092
 IF STEEL=0 THEN NAME[2,5] ELSE NAME[2,4],COST);                       0103
GO TO L4;                                                              0114
L5:  FOR I←1 STEP 1 UNTIL 3 DO IF NB[J,J1]=NBON[J,I] THEN              0114
 COST←BONCOST[J,I]; TOTAL←TOTAL+COST;                                  0118
IF EXTCOL=1 THEN WRITE(PRINT,FL15,COST);                               0124
IF RADFIN=1 THEN WRITE(PRINT,FL16,COST);                               0134
L4:  FOR I←1 STEP 1 UNTIL COMPAT DO IF J=AVAIL[I] THEN M←THCOL[I];     0143
I←NT[J,M];                                                             0149
COST←THCOST[J,M]; TOTAL←TOTAL+COST;                                    0151
WRITE(PRINT,FL17,NAME[1,I],NAME[2,I],COST);                            0154
FOR I←1 STEP 1 UNTIL 10 DO IF NT[J,M]=ABS(NT1[I]) THEN GO TO L6;       0168
L6:  IF NT1[I]<0 THEN WRITE(PRINT,FL11);                               0174
IF UP=1 THEN GO TO L7;                                                 0179
FOR I←1 STEP 1 UNTIL COMPAT DO IF AVAIL[I]=J THEN M←MAN[I];            0180
COST←MCOST[J,M]; TOTAL←TOTAL+COST;                                     0185
IF TOP=1 THEN WRITE(PRINT,FL22,COST) ELSE WRITE(PRINT,FL23,COST);      0188
L7: IF ONOFF=1 THEN GO TO L10;                                         0207
IF POSCOST[J,2]≤0 THEN GO TO L8;                                       0209
COST←POSCOST[J,2];                                                     0211
```

```
DX1+2;                                                                  0213
GO TO L9;                                                               0214
L8: IF POSCOST[J,1]=0 THEN GO TO L10;                                   0214
COST+POSCOST[J,1];                                                      0217
DX1+1;                                                                  0219
L9: TOTAL+TOTAL+COST;                                                   0219
IF DX1=2 THEN WRITE(PRINT,FL24,POS[J,DX1],COST) ELSE                    0221
 WRITE(PRINT,FL25,POS[J,DX1],COST);                                     0233
L10: WRITE(PRINT,FL20,TOTAL);                                           0245
END PRINTZ;                                                             0254
                                                   0007 IS 0257 LONG, NEXT SEG 0002
PROCEDURE RUNGKUTTA(DIAP1,DIAP3,DIAV1,DIAV3,QOTIN,RESTART);             0091
VALUE DIAP1,DIAP3,DIAV1,DIAV3,QOTIN;                                    0091
REAL DIAP1,DIAP3,DIAV1,DIAV3,QOTIN;                                     0091
INTEGER RESTART;                                                        0091
BEGIN                                                                   0091
INTEGER I,K,V1,VLAST,N,CHANGE,OSOK,SETTLNG;                             0091
                                                   START OF SEGMENT ********** 0009
REAL T,DT,T1,FAC0,FAC1,FAC10,FAC2,FAC20,FAC3,FAC30,FAC300,FAC4,FAC40,   0000
 FAC5,FAC50,FAC6,FAC33,TLAST,H1X2,RP3,RV3,P,RP1,RV1,XT,T3,T4,TSET0,     0000
 DELT2,RE1,H,DIFF;                                                      0000
ARRAY TR[0:2],CX[0:2],XLAST[0:4],D[0:4,0:4],TSET[0:2];                  0000
FORMAT FMT(6E20.5);                                                     0010
                                                   START OF SEGMENT ********** 0010
                                                   0010 IS 0004 LONG, NEXT SEG 0009
FORMAT FL1("RISE TIME O.K."),                                           0010
                                                   START OF SEGMENT ********** 0011
       FL2("NEW CV = ",E15.5),                                          0010
       FL3("CANNOT MEET RISE TIME REQUIREMENTS"),                       0010
       FL4("OVERSHOOT O.K."),                                           0010
       FL5("CANNOT MEET OVERSHOOT REQUIREMENTS"),                       0010
       FL6("CANNOT MEET SETTLING TIME REQUIREMENTS"),                   0010
```

```
          FL7("SETTLING TIME O.K."),                                     0010
          FL8("STEADY STATE ERROR O.K."),                                0010
          FL9("CANNOT MEET OVERSHOOT REQUIREMENTS(RR<0.25)"),            0010
          FL10("CANNOT MEET SETTLING TIME REQUIREMENTS(RR<0.25)"),       0010
          FL12("PROPORTIONAL+RESET"),                                    0010
          FL13("LOW-COST SYSTEM NOT AVAILABLE"),                         0010
          FL14(/X14,"T",X17,"X[1]",X14,"X[2]",X16,"X[3]",X17,"XT"),      0010
          FL15(/X14,"T",X17,"X[1]",X14,"X[2]",X16,"X[3]",X17,"X[4]"),    0010
          FL11("NO PROPORTIONAL+RESET SYSTEM AVAILABLE");                0010
                                              0011 IS 0133 LONG, NEXT SEG 0009
     LABEL L0,L10,L31,L4,L5,L6,L7,BOB,START,L2,MAX,L11,L12,L13,L14,      0010
      L8,L9,L15,L16,L17,L18,L19,L20,L21,L22;                             0010
     DIFF+H1-ORIGX[2];                                                   0010
     N+TRR×4;                                                            0011
     IF RESTART≠0 THEN PB+0.30;                                          0013
     CX[1]+0.1; CX[2]+0.V0;                                              0015
     IF RR≠0.0 THEN WRITE(PRINT,FL14) ELSE WRITE(PRINT,FL15);            0017
     START: V1+1;                                                        0029
     VLAST+0;                                                            0030
     T3+0.0;                                                             0031
     OSOK+0;                                                             0032
     SETTLNG+0;                                                          0033
     T+0.0;                                                              0033
     DT+0.1;                                                             0034
     FOR I+1,2 DO BEGIN TR[I]+0.0; TSET[I]+0.0; END;                     0035
     IF RR≠0.0 THEN BEGIN                                                0044
      ORIGX[3]+X[3]+X[4];                                                0045
      IF CONFIG=1 THEN N+2 ELSE N+3;                                     0047
      GO TO L2;                                                          0051
     END;                                                                0051
     IF CONFIG=1 THEN N+3 ELSE N+4;                                      0051
     L2: FACO+PH×CR;                                                     0054
```

```
FAC1←303×DIAP1×DIAP1/(LT[2]+LT[3]);                                        0056
FAC2←(LE[2]+LE[3])/DIAP1↑5;                                                 0059
FAC3←3.3@-3×G/(DIAP1×U);                                                    0063
IF DIAV1≠0 THEN FAC33←3.3@-3×G/(DIAV1×U);                                   0065
FAC4←1.94@4×DIAP1×U/G;                                                      0069
FAC5←2@4×E/DIAP1;                                                           0071
FAC6←1/A8;                                                                  0073
IF CONFIG=2 THEN BEGIN                                                      0074
 FAC10←303×DIAP3×DIAP3/(LT[4]+LT[5]);                                       0075
 FAC20←(LE[4]+LE[5])/DIAP3↑5;                                               0078
 FAC30←3.3@-3×G/(DIAP3×U);                                                  0082
 FAC300←3.3@-3×G/(DIAV3×U);                                                 0084
 FAC40←1.94@4×DIAP3×U/G;                                                    0087
 FAC50←2@4×E/DIAP3;                                                         0089
END;                                                                        0091
L0: FOR I←1 STEP 1 UNTIL 4 DO XLAST[I]←X[I];                                0091
TLAST←T;                                                                    0098
CHANGE←SIGN(D[2,4]);                                                        0099
FOR K←1 STEP 1 UNTIL 4 DO BEGIN                                             0103
RP1←FAC3×X[1];                                                              0104
RV1←FAC33×X[1];                                                             0105
IF RP1<200 THEN BEGIN WRITE(PRINT,FL2013,RP1); STOPX←1; GO TO L6; END;      0107
IF RP1≤2000 THEN F2←FAC4/X[1] ELSE F2←5.5@-3×(1+(FAC5+1@6/RP1)↑0.33333);    0117
IF X[2]>H3 THEN H2←X[2]-H3 ELSE H2←0.0;                                     0126
IF TANK=3 THEN P0←P00×L1/(L1-X[2]);                                         0134
IF DIAV1≠0.0 THEN RE1←RV1 ELSE RE1←RP1;                                     0138
IF RE1≤1000 THEN CV1←CVCN1×(0.707+0.11×(LN(RE1)-5.3))                       0141
 ELSE IF RE1≤5000 THEN CV1←CVCN1×(0.884+0.0398×(LN(RE1)-6.91))              0145
 ELSE IF RE1≤100000 THEN CV1←CVCN1×(0.948+0.01734×(LN(RE1)-8.52))           0154
 ELSE CV1←CVCN1;                                                            0165
H1X2←H1-X[2];                                                               0173
IF ONOFF=1 THEN BEGIN                                                       0175
```

```
IF H1X2>=DH AND H1X2<DH THEN BEGIN                                          0176
 IF VLAST=0 THEN BEGIN XT+0.0; GO TO BOB; END;                              0179
 IF K=2 OR K=3 THEN T1+T1+DT/2;                                             0181
 IF T1≤QDTIM THEN XT+T1/QDTIM ELSE XT+1.0;                                  0185
 GO TO BOB;                                                                 0189
END;                                                                        0191
IF H1X2≥DH THEN BEGIN                                                       0191
 IF VLAST=0 THEN T1+0.0 ELSE BEGIN IF K=2 OR K=3 THEN T1+T1+DT/2; END;      0192
 VLAST+V1;                                                                  0198
 IF T1≤QDTIM THEN XT+T1/QDTIM ELSE XT+1.0;                                  0199
 GO TO BOB;                                                                 0203
END;                                                                        0205
VLAST+0;                                                                    0205
XT+0.0;                                                                     0205
GO TO BOB;                                                                  0206
END;                                                                        0207
IF RR=0.0 THEN BEGIN                                                        0207
 XT+H1X2/FAC0+0.82;                                                         0208
 IF XT<0.0 THEN XT+0.0;                                                     0210
 IF XT>1.0 THEN XT+1.0;                                                     0212
GO TO BOB;                                                                  0214
END;                                                                        0217
IF X[3]<0.0 THEN X[3]+0.0;                                                  0217
IF X[3]>1.0 THEN X[3]+1.0;                                                  0219
D[1,K]+(FAC1×((P1-P0)/G-H2-(27.6/(R*(2×(X[3]-1))×CV1+2)+2.10-3×F2          0222
 ×FAC2)×X[1]×X[1]))×DT;                                                     0231
IF N=3 THEN D[2,K]+FAC6×(X[1]-FLOUT)×DT ELSE D[2,K]+FAC6×(X[1]-X[4])×DT;    0235
D[3,K]+(RR×H1X2-D[2,K])/FAC0×DT;                                            0247
GO TO HAK;                                                                  0253
BOB: IF ONOFF=1 THEN IF XT=0.0 THEN X[1]+ORIGX[1];                          0253
     D[1,K]+(FAC1×((P1-P0)/G-H2-(27.6/(R*(2×(XT-1))×CV1×CV1)+2.10-3×F2      0258
 ×FAC2)×X[1]×X[1]))×DT;                                                     0266
```

```
IF N=2 THEN D[2,K]←FAC6×(X[1]−FLOUT)×DT                                0270
 ELSE D[2,K]←FAC6×(X[1]−X[3])×DT;                                      0274
BAX: IF CONFIG=2 THEN BEGIN                                            0282
 RP3←FAC30×X[N];                                                       0283
 RV3←FAC300×X[N];                                                      0284
 IF RP3<200 THEN BEGIN WRITE(PRINT,FL2014,RP3); GO TO L6; END;         0286
 IF RP3≤2000 THEN F4←FAC40/X[N]                                        0295
  ELSE F4←5.50−3×(1+(FAC50+1@6/RP3)*0.33333);                          0297
 IF RV3≤1000 THEN CV3←CVCN3×(0.707+0.11×(LN(RV3)−5.3))                 0304
  ELSE IF RV3≤5000 THEN CV3←CVCN3×(0.884+0.0398×(LN(RV3)−6.91))        0308
  ELSE IF RV3≤10000 THEN CV3←CVCN3×(0.948+0.01734×(LN(RV3)−8.52))      0320
  ELSE CV3←CVCN3;                                                      0331
 D[N,K]←(FAC10×((P0−P7)/G+X[2]−(27.6/(K20×CV3)*2+2.10−3×F4×FAC20)×X[N]  0339
  ×X[N]))×DT;                                                          0346
END;                                                                   0349
IF K=4 THEN GO TO L3; ELSE IF K=2 THEN GO TO L10;                      0349
T←T+0.5×DT;                                                            0352
L10: P←0.5;                                                            0353
IF K=3 THEN P←1.0;                                                     0354
FOR I←1 STEP 1 UNTIL N DO X[I]←X[I]+D[I,K]×P;                          0356
IF X[1]<0.0 THEN GO TO L11;                                            0367
IF ONOFF=1 THEN IF X[1]≤150×ORIGX[1] THEN GO TO L7;                    0369
IF X[1]<3.0×ORIGX[1] THEN GO TO L7;                                    0372
L11: DT←DT/10;                                                         0375
 T←0.0;                                                                0376
FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                               0377
 GO TO L0;                                                             0382
L7: IF K>1 THEN FOR I←1 STEP 1 UNTIL N DO X[I]←X[I]−0.5×D[I,K−1];      0383
IF X[1]<0.0 THEN GO TO L12;                                            0392
IF ONOFF=1 THEN IF X[1]≤150×ORIGX[1] THEN GO TO L3;                    0393
IF X[1]<3.0×ORIGX[1] THEN GO TO L3;                                    0397
L12: DT←DT/10;                                                         0399
```

```
T+0.0;                                                              0401
FOR I+1 STEP 1 UNTIL 4 DO X[I]+ORIGX[I];                            0402
 GO TO L0;                                                          0408
L31: END;                                                           0409
FOR I+1 STEP 1 UNTIL N DO                                           0412
 X[I]+X[I]+(D[I,1]+2×D[I,2]+2×D[I,3]+D[I,4])/6.0×D[I,3];           0414
IF X[1]<0.0 THEN GO TO L13;                                         0426
IF ONOFF=1 THEN IF X[1]≤150×ORIGX[1] THEN GO TO L14;                0428
IF X[1]<3.0×ORIGX[1] THEN GO TO L14;                                0431
L13: DT+DT/10;                                                      0434
 T+0.0;                                                             0435
FOR I+1 STEP 1 UNTIL 4 DO X[I]+ORIGX[I];                            0436
 GO TO L0;                                                          0442
L14: IF TR[2]≠0 THEN GO TO L8;                                      0443
IF TR[1]≠0 THEN I+2 ELSE I+1;                                       0445
IF X[2]=(CX[I]×DIFF+ORIGX[2]) THEN BEGIN                            0449
 TR[I]+T;                                                           0452
 GO TO L4;                                                          0453
END;                                                                0453
IF X[2]>(CX[I]×DIFF+ORIGX[2]) THEN IF XLAST[2]<(CX[I]×DIFF+ORIGX[2]) 0453
 THEN GO TO L5;                                                     0458
GO TO L4;                                                           0459
L5: TR[I]+TLAST+DT/2;                                               0460
IF TR[2]≤0 THEN GO TO L4;                                           0463
TRD+TR[2]-TR[1];                                                    0464
IF TRUSTRN THEN BEGIN WRITE(PRINT,FL1); GO TO L6; END;              0466
IF RESTART=0 THEN BEGIN RESTART+1; GO TO L6; END;                   0471
IF PB=0.30 THEN GO TO L22;                                          0474
CVCN1+CVCN1×1.25;                                                   0475
IF CVCN1>500.0 THEN BEGIN WRITE(PRINT,FL3); STOPX+1; GO TO L6; END; 0477
WRITE(PRINT,FL2,CVCN1);                                             0486
FOR I+1 STEP 1 UNTIL 4 DO X[I]+ORIGX[I];                            0494
```

```
L18: IF T3<DELTT2 THEN GO TO L4;                                              0581
L17: IF T>TSETR THEN BEGIN                                                   0583
IF PB=0.30 THEN GO TO L22;                                                    0585
IF RESTART=0 THEN BEGIN RESTART←1; GO TO L6; END;                             0586
IF RR>0.0 THEN BEGIN                                                          0591
 RR←RR*0.25;                                                                  0592
 IF RR<0.25 THEN BEGIN WRITE(PRINT,FL10); STOPX←1; GO TO L6; END;             0593
 SETTLNG←0;                                                                   0601
 FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                                     0601
 GO TO START;                                                                 0606
END;                                                                          0607
 PB←PB+0.1;                                                                   0607
 IF PB>PBMAX THEN BEGIN WRITE(PRINT,FL6); STOPX←1; GO TO L6; END;             0608
 SETTLNG←0;                                                                   0616
 FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                                     0616
 GO TO START;                                                                 0621
END;                                                                          0622
SETTLNG←1;                                                                    0622
WRITE(PRINT,FL7);                                                             0623
L20: IF T≤TSETR THEN GO TO L4;                                                0626
IF ABS(H1-X[2])>ESS THEN BEGIN T4←0.0; GO TO L4; END;                         0628
T4←T4+DT;                                                                     0631
IF T4<DELTT2 THEN GO TO L4;                                                   0633
WRITE(PRINT,FL8);                                                             0634
GO TO L6;                                                                     0637
L4: IF RR≠0.0 THEN WRITE(PRINT,FMT,T,X[1],X[2],X[3],X[4]);                    0638
IF T≤M THEN GO TO L0;                                                         0654
IF RESTART=0 THEN BEGIN RESTART←1; GO TO L6; END;                             0655
L22: IF PB=0.30 THEN BEGIN                                                    0658
 PB←0.05; PBMAX←0.25;                                                         0660
 FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                                     0661
 WRITE(PRINT,FL13); GO TO START; END;                                         0669
```

```
GO TO START;                                                        0498
L8: IF USOK=1 THEN GO TO L9;                                        0499
IF SIGN(D[2,4])≠CHANGE THEN GO TO L15;                              0501
IF D[2,4]>.0001 THEN GO TO L4;                                      0505
L15: IF ABS(H1-X[2])<05 THEN BEGIN USOK←1;                          0507
WRITE(PRINT,FL4); GO TO L4; END;                                    0511
IF RESTART=0 THEN BEGIN RESTART←1; GO TO L6; END;                   0516
IF PB=0.30 THEN GO TO L22;                                          0518
IF RR>0.0 THEN BEGIN                                                0520
 RR←RR-0.25;                                                        0521
 IF RR<0.25 THEN BEGIN WRITE(PRINT,FL9); STOPX←1; GO TO L6; END;    0522
FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                            0531
 GO TO START;                                                       0535
END;                                                                0536
PB←PB+0.1;                                                          0536
IF PB>PBMAX THEN BEGIN                                              0537
 WRITE(PRINT,FL5); STOPX←1; GO TO L6; END;                          0538
FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                            0545
GO TO START;                                                        0549
L9: IF SETTLNG=1 THEN GO TO L20;                                    0550
IF ABS(H1-X[2])≤0.05×DIFF THEN T3←T3+DT ELSE BEGIN                  0552
 T3←0.0; GO TO L4; END;                                             0558
IF SIGN(H1-XLAST[2])≠SIGN(H1-X[2]) THEN GO TO L16;                  0559
IF (H1-X[2])=0.0 THEN GO TO L16;                                    0565
L19: IF T3<TRR/2 THEN GO TO L4;                                     0567
DELTT2←TRR/2;                                                       0569
GO TO L17;                                                          0571
L16: IF TSET[2]≠0 THEN GO TO L18;                                   0571
IF TSET[1]≠0 THEN I←2 ELSE I←1;                                     0573
TSET[I]←T;                                                          0577
IF I=1 THEN GO TO L19;                                              0578
DELTT2←TSET[2]-TSET[1];                                             0579
```

```
IF RN≠0.0 THEN GO TO L21;                                              0673
 RR←0.50;                                                              0675
ORIGX[3]←0.82;                                                         0675
FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                               0677
WRITE(PRINT,FL12);                                                     0683
WRITE(PRINT,FL15);                                                     0687
GO TO START;                                                           0690
L21: RR←RN-0.25;                                                       0691
IF RR<0.25 THEN BEGIN WRITE(PRINT,FL11); STOPX←1; GO TO L6; END;       0693
FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                               0701
GO TO START;                                                           0705
L6: END RUNGKUTTA;                                                     0706
                                                   0009 IS 0724 LONG, NEXT SEG 0002
BEGIN                                                                  0091
LABEL L2,L50,L51,L100,LX;                                              0091
                                                   START OF SEGMENT ********** 0012
Z←TIME(1);                                                             0000
READ(CARD,FL1,CONFIG,TANK,MEDM,SLURRY,TNKRATNG,PNEU,P00,P1,P1MAX,P0MIN,   0001
 P7,P13,T7);                                                           0019
WRITE(PRINT,FL1,CONFIG,TANK,MEDM,SLURRY,TNKRATNG,PNEU,P00,P1,P1MAX,    0025
 P0MIN,P7,P13,T7);                                                     0041
READ(CARD,FL9,D1,D3,DV3,L1,H1,H3,M1,M2,D8,QOM,CVCN3);                  0069
WRITE(PRINT,FL9,D1,D3,DV3,L1,H1,H3,M1,M2,D8,QOM,CVCN3);                0071
READ(CARD,FL11,FOR DX1←2 STEP 1 UNTIL 5 DO LT[DX1],RD,NBI[1],NTI[1],   0093
 SEALI,CAPCTY,OP,TOP,E,U,G,K20);                                       0108
WRITE(PRINT,FL11,FOR DX1←2 STEP 1 UNTIL 5 DO LT[DX1],RD,NBI[1],NTI[1], 0121
 SEALI,CAPCTY,OP,TOP,E,U,G,K20);                                       0136
READ(CARD,FL13,FOR DX1←1,2,4 DO ORIGX[DX1],TRR,CR,DS,TSETR,ESS,DH,     0149
 STATIC);                                                              0171
WRITE(PRINT,FL13,FOR DX1←1,2,4 DO ORIGX[DX1],TRR,CR,DS,TSETR,ESS,DH,   0175
 STATIC);                                                              0197
READ(CARD,FL10,FOR DX1←1 STEP 1 UNTIL 20 DO LD[DX1]);                  0201
```

```
FOR I+1 STEP 1 UNTIL 4 DO X[I]+OR[GX[I]];
FLOUT+X[4];
FMAX+1.1;
FMIN+0.85;
R+50.0;
HR+0.0;
IF DB>0.0 THEN AB+3.14159×DB×DB/4 ELSE AB+W1×W2;
IF DH>0 THEN ONOFF+1 ELSE ONOFF+0;
FOR I+2 STEP 1 UNTIL 5 DO BEGIN
 IF LT[I]=0 THEN FOR J+1 STEP 1 UNTIL 21 DO NFIT[J]+0
 ELSE READ(CARD,FL12,FOR J+1 STEP 1 UNTIL 21 DO NFIT[J]);
 WRITE(PRINT,FL12,FOR J+1 STEP 1 UNTIL 21 DO NFIT[J]);
 SUML+0;
 FOR J+1 STEP 1 UNTIL 20 DO SUML+SUML+NFIT[J]×LD[J];
 IF NFIT[21]=0 THEN GO TO L100;
 SUML+SUML+NFIT[21]×
  (5.9996568@-01-5.9953110@+01×RD+2.5896379@+01×RD*2
  -7.2291399×RD*3+1.0614061×RD*4-9.353222@-02×RD*5
  +4.8506803@-03×RD*6-1.3621214@-04×RD*7+1.5958646@-06×RD*8);
L100: IF I>3 THEN LE[I]+LT[I]+D3×SUML ELSE LE[I]+LT[I]+D1×SUML;
END;
CLOSE(CARD,RELEASE);
IF ONOFF=1 THEN X1M+1.1×QUM ELSE X1M+X[1];
RP1+3.3@-3×X1M×G/(D1×U);
IF RP1<200 THEN BEGIN WRITE(PRINT,FL2012,RP1); GO TO LX; END;
IF RP1≤2000 THEN F2+1.94@4×D1×U/(G×X1M)
 ELSE F2+5.5@-3×(1+(2@4×E/D1+1@6/RP1)*0.33333);
IF H1>H3 THEN H2+H1-H3 ELSE H2+0.0;
P0+IF TANK=1 THEN 15.0 ELSE IF TANK=2 THEN P13 ELSE P00×L1/(L1-H1);
IF ONOFF=1 THEN NMRIR+5.275×X1M ELSE NMRIR+5.275×X1M×R*0.19;
CV1+NMRIR/SQRT((P1-P0)/G-H2-(2.1@-3×F2×(LE[2]+LE[3])×X1M×X1M/D1*5));
IF RP1≤1000 THEN CVCN1+CV1/(0.707+0.11×(LN(RP1)-5.3))
```

```
 ELSE IF RP1≤5000 THEN CVCN1+CV1/(0.864+0.0398×(LN(RP1)-6.91))          0386
 ELSE IF RP1≤100000 THEN CVCN1+CV1/(0.948+0.01734×(LN(RP1)-8.52))       0399
 ELSE CVCN1+CV1;                                                          0410
WRITE(PRINT,FL20,CVCN1);                                                  0418
NEXTV+1;                                                                  0427
DV1+0.0;                                                                  0428
RUNGKUTTA(D1,D3,DV1,DV3,IF ONOFF=1 THEN 60.0 ELSE 0.0,NEXTV);             0428
IF STOPX=1 THEN GO TO LX;                                                 0432
NVALVS+0;                                                                 0434
DELTPI+IF TANK=1 THEN P1MAX-15.0 ELSE IF TANK=2 THEN P1MAX-P13 ELSE       0434
 P1MAX-P00;                                                               0439
CVMIN+FMIN×CVCN1×0.26;                                                    0441
CVMAX+FMAX×CVCN1×0.26;                                                    0442
P8+P1MAX;                                                                 0444
DO READ(TAPE1,*,A,FOR DX1+1 STEP 1 UNTIL 10 DO B[DX1]) UNTIL A=NEOM;      0445
DO READ(TAPE1,*,A) UNTIL A=9999;                                          0464
READ(TAPE1,*,FOR I+1 STEP 1 UNTIL 9 DO[DISPL[I],FOR DX1+1 STEP 1 UNTIL    0474
 4 DO DISCOST[I,DX1]]);                                                   0482
IF NBI[1]≠0 THEN BEGIN                                                    0493
FOR I+2 STEP 1 UNTIL 10 DO NTX[I]+0;                                      0495
FOR I+2,3,4 DO NBI[I]+0;                                                  0499
GO TO L50; END;                                                           0508
J+0;                                                                      0509
FOR I+1 STEP 1 UNTIL 4 DO BEGIN                                           0509
 DUM+ABS(B[I]);                                                           0511
 RATNG1[I,1]+0;                                                           0512
 RATNG1[I,2]+0;                                                           0514
 IF DUM=1 OR DUM=2 OR DUM=3 OR DUM=4 THEN BEGIN                           0516
  J+J+1;                                                                  0520
  NBI[J]+B[I];                                                            0521
END END;                                                                  0523
FOR I+1 STEP 1 UNTIL 10 DO NTX[I]+B[I];                                   0525
```

```
L50: I←1;                                                                 0530
L51: READ(TAPE1,*,MODEL[I],SIZE[I],TYPE[I],VPORT[I],QU[I],VPFC[I],        0531
 VPRC[I],VPLINK[I],VPTIME[I],QULINK[I],QUTIME[I],TMIN[I],TMAX[I],         0545
 TRAVL[I],MODY[I])[L2];                                                   0556
J←0;                                                                      0563
CVS(VPFC[I]);                                                             0564
J←J+VPFC[I];                                                              0565
LIMVFC[I]←J;                                                              0567
CVS(VPRC[I]);                                                             0568
J←J+VPRC[I];                                                              0569
LIMVRC[I]←J;                                                              0570
CVS(QU[I]);                                                               0572
J←J+QU[I];                                                                0573
NUMCVS[I]←J;                                                              0574
READ(TAPE1,*,FOR J←1,2,3 DO[NB[I,J],FOR DX←1 STEP 1 UNTIL 4               0575
 DO[RATNG[I,J,DX],BCOST[I,J,DX]]]);                                       0590
FOR J←1,2,3 DO RATNG[I,J,4]←-RATNG[I,J,4];                                0602
READ(TAPE1,*,EXT[I],RAD[I],FOR J←1,2,3 DO[NBON[I,J],BONCOST[I,J]],        0615
 SEAL[I],FOR J←4 STEP 1 UNTIL 9 DO[NBON[I,J],BONCOST[I,J]]);              0635
READ(TAPE1,*,FOR J←1 STEP 1 UNTIL 6 DO[NT[I,J],TMCOST[I,J]],              0647
 FOR J←1 STEP 1 UNTIL 4 DO MCOST[I,J],FOR J←1,2 DO[POS[I,J],              0660
 POSCOST[I,J]]);                                                          0673
I←I+1;                                                                    0679
GO TO L51;                                                                0681
L2: NVALVS←I-1;                                                           0681
BEGIN                                                                     0683
LABEL       L4,L5,L10,L11,L7,L6,L8,L13,L9,L12,L14,L15,L16,L17,L18,L20,    0683
                                               START OF SEGMENT ********** 0013
 L21,L24,L25,L23,L22,L26,L27,L30,L35,L36,L34,L37,L32,L31,L33,             0000
 L46,L45,L47,L41,L40,L43,L44,L61,L62,L3,L60,L63,L64,L65,L66,L67,L68,      0000
 L42,L28;                                                                 0000
SWITCH SWGO1←L10,L11,L12,L12;                                             0000
```

```
FOR I←1 STEP 1 UNTIL 4 DO BEGIN
 IF NBI[I]=0 THEN GO TO L21;
 DUM←ABS(NBI[I]);
 GO TO SWGO[DUM];
 L10: IF P8>125 THEN GO TO L3;
 IF T7>353 THEN GO TO L4;
 RATNGI[1,1]←125;
 RATNGI[1,2]←-125;
 GO TO L20;
 L3: IF P8>175 THEN GO TO L5;
 IF T7>150 THEN GO TO L4;
 RATNGI[1,1]←125;
 RATNGI[1,2]←-125;
 GO TO L20;
 L4: IF T7≤406 THEN BEGIN
 RATNGI[1,1]← 250;
 RATNGI[1,2]←-250; END;
 GO TO L20;
 L5: IF P8≤250 THEN GO TO L4;
 IF P8>400 OR T7>150 THEN GO TO L20;
 RATNGI[1,1]← 250;
 RATNGI[1,2]←-250;
 GO TO L20;
 L11: IF P8>125 THEN GO TO L6;
 IF T7>400 THEN GO TO L7;
 RATNGI[2,2]←-125;
 GO TO L20;
 L7: IF T7>422 THEN GO TO L20;
 RATNGI[2,1]←300;
 GO TO L20;
 L6: IF P8>150 OR T7>400 THEN GO TO L8;
 RATNGI[2,1]←150;
```

```
GO TO L20;
L8: IF P8>175 THEN GO TO L9;
IF T7>150 THEN GO TO L13;
RATNGI[2,2]←-125;
GO TO L20;
L13: IF T7>406 THEN GO TO L7;
RATNGI[2,2]←-250;
GO TO L20;
L9: IF P8≤250 THEN GO TO L13; IF P8≤300 THEN GO TO L7;
IF P8>400 THEN GO TO L14;
L15: IF T7≤150 THEN RATNGI[2,2]←-250;
GO TO L20;
L14: IF P8>500 THEN GO TO L20;
IF T7≤150 THEN RATNGI[2,1]←300;
GO TO L20;
L12: IF DUM=3 THEN P15←3.0105262⏨2-1.6768865⏨-1×T7-7.0941098⏨-4×T7×T7
 +1.2550600⏨-6×T7↑3-6.5264316⏨-10×T7↑4 ELSE P15←2.5691054⏨2
 +4.6506157⏨-1×T7-4.4136753⏨-3×T7×T7+1.1069110⏨-5×T7↑3
 -1.3206283⏨-8×T7↑4+7.4534550⏨-12×T7↑5-1.5912509⏨-15×T7↑6;
IF P8>P15 THEN GO TO L16;
RATNGI[DUM,1]← 150;
RATNGI[DUM,2]←-150;
GO TO L20;
L16: IF DUM=3 THEN P30←7.6256457⏨2-7.1520411⏨-1×T7+2.4541191⏨-3×T7↑2
 -4.0495042⏨-6×T7↑3+1.6352804⏨-9×T7↑4 ELSE P30←8.3103557⏨2-2.402578×T7
 +2.0435072⏨-2×T7×T7-9.8796968⏨-5×T7↑3+2.8161753⏨-7×T7↑4
 -4.9388141⏨-10×T7↑5+5.1928453⏨-13×T7↑6-3.0120952⏨-16×T7↑7
 +7.2269829⏨-20×T7↑8+6.8980817⏨-24×T7↑9-4.6982276⏨-27×T7↑10;
IF P8>P30 THEN GO TO L17;
RATNGI[DUM,1]← 300;
RATNGI[DUM,2]←-300;
GO TO L20;
```

```
L17: IF DUM=3 THEN P40←1.0174975@3-8.4497081@-1×T7+3.0019446@-3×T7×T7
 -5.1700392@-6×T7*3+2.1133420@-9×T7*4 ELSE P40←1.1861529@3-4.0173019×T7
 +2.3388695@-2×T7×T7-6.4850059@-5×T7*3+8.8982563@-8×T7*4
 -6.3425416@-11×T7*5+2.2109879@-14×T7*6-2.8998542@-18×T7*7;
IF P8>P40 THEN GO TO L18;
RATNGI[DUM,1]← 400;
RATNGI[DUM,2]←-400;
GO TO L20;
L18: IF DUM=3 THEN P60←1.2966267@3+4.2001297×T7-4.2924664@-2×T7*2
 +1.9283055@-4×T7*3-4.5558303@-7×T7*4+5.7819949@-10×T7*5
 -3.7739105@-13×T7*6+9.9562479@-17×T7*7 ELSE P60←1.6750744@3
 -4.4849198×T7+3.0768629@-2×T7*2-1.1768237@-4×T7*3+2.7933649@-7×T7*4
 -4.4917889@-10×T7*5+4.6604068@-13×T7*6-2.7668428@-16×T7*7
 +6.9711866@-20×T7*8+5.6479928@-24×T7*9-4.4213702@-27×T7*10;
IF P8>P60 THEN GO TO L28;
RATNGI[DUM,1]← 600;
RATNGI[DUM,2]←-600;
L20: END;
L28: FOR I←1 STEP 1 UNTIL 4 DO FOR K←1,2 DO IF RATNGI[I,K]≠0 THEN
 GO TO L21;
WRITE(PRINT,FL1000);
GO TO LX;
L21: IF SEALI=0 THEN GO TO L22;
IF T7≤750 THEN GO TO L25;
L24: WRITE(PRINT,FL1001);
GO TO LX;
L25: IF P8>150 THEN GO TO L23;
FOR I←1 STEP 1 UNTIL 10 DO IF ABS(A[I])=5 THEN MONEL←1;
FOR I←1 STEP 1 UNTIL 10 DO IF ABS(A[I])=4 THEN STEEL←1;
IF MONEL=0 AND STEEL=0 THEN BEGIN
 WRITE(PRINT,FL2010); GO TO LX;
END;
```

```
GO TO L30;                                                        0335
L23: IF P8>400 THEN GO TO L24;                                    0336
MONEL←0;                                                          0337
FOR I←1 STEP 1 UNTIL 10 DO IF ABS(R[I])=4 THEN STEEL←1;           0338
IF STEEL=0 THEN BEGIN                                             0343
 WRITE(PRINT,FL2011); GO TO LX;                                   0345
END;                                                              0350
GO TO L30;                                                        0350
L22: IF T7≤=450 OR T7>32 THEN GO TO L26;                          0351
EXTCOL←1;                                                         0353
GO TO L30;                                                        0354
L26: IF T7≤32 OR T7>450 THEN GO TO L27;                           0355
GO TO L30;                                                        0357
L27: IF T7≤450 OR T7>1200 THEN GO TO L24;                         0358
RAOFIN←1;                                                         0360
L30: FOR I←1 STEP 1 UNTIL 10 DO NTI[I]←0;                         0361
IF SLURRY=1 THEN GO TO L31;                                       0367
IF DELTPI>50 THEN BEGIN                                           0368
 IF DELTPI>300 THEN GO TO L32 ELSE GO TO L35;                     0370
END;                                                              0371
M←2;                                                              0371
NTRIM;                                                            0372
IF IND=1 THEN NTI[1]←M ELSE NTI[1]←0;                             0373
L35: M←4;                                                         0377
NTRIM;                                                            0378
IF IND=1 THEN NTI[2]←M ELSE NTI[2]←0;                             0379
IF NTI[1]≠0 OR NTI[2]≠0 THEN GO TO L33;                           0383
L36: IF ONOFF≠1 THEN GO TO L34;                                   0386
NTI[1]←10;                                                        0388
GO TO L33;                                                        0389
L34: M←5;                                                         0390
NTRIM;                                                            0390
```

```
IF IND=1 THEN NTI[1]←M ELSE NTI[1]←0;                                0391
M←7;                                                                 0395
NTR[M];                                                              0396
IF IND=1 THEN NTI[2]←M ELSE NTI[2]←0;                                0396
M←8;                                                                 0401
NTR[M];                                                              0401
IF IND=1 THEN NTI[3]←M ELSE NTI[3]←0;                                0402
M←9;                                                                 0406
NTR[M];                                                              0407
IF IND=1 THEN NTI[4]←M ELSE NTI[4]←0;                                0407
IF NTI[1]≠0 OR NTI[2]≠0 OR NTI[3]≠0 OR NTI[4]≠0 THEN GO TO L33;      0412
L37: SAUNDERS←1;                                                     0417
GO TO L33;                                                           0418
L32: IF DELTPI≤600 THEN GO TO L36;                                   0419
WRITE(PRINT,FL1003);                                                 0421
GO TO LX;                                                            0424
L31: IF DELTPI>50 THEN GO TO L37;                                    0426
M←4;                                                                 0428
NTR[M];                                                              0429
IF IND=1 THEN NTI[1]←M ELSE NTI[1]←0;                                0429
IF NTI[1]≠0 THEN GO TO L37;                                          0433
L33: J←0;                                                            0435
END REQUIREMENT BLK;                                                 0436
                                          0013 IS 0441 LONG, NEXT SEG 0012
BEGIN                                                                0684
COMMENT ELIMINATION BLK;                                             0684
LABEL L45,L46,L47,L41,L40,L42,L43,L44,L61,L60,L62,L63,L30,           0684
                                          START OF SEGMENT ********** 0014
 L66,L67,L65,L64,L68,L1,L3,L75,L76,L80,L105,L101,L102,L103,L104,L74, 0000
 L77,L79,L78,L81,L82,L83,L84,L85,L86,L20,L21,L22,L90,L91,L92,L69;    0000
IF UNOFF≠1 THEN GO TO L45;                                           0000
FOR I←1 STEP 1 UNTIL NVALVS DO BEGIN                                 0001
```

```
 IF UO[I]=0 THEN GO TO L46;
 K←NUMCVS[I];
 IF CV[I,K]<CVMIN OR CV[I,K]>CVMAX THEN GO TO L46;
 J←J+1;
 AVAIL[J]←I;
L46: END;
 GO TO L47;
L45: FOR I←1 STEP 1 UNTIL NVALVS DO BEGIN
 IF CAPCTY=1 THEN BEGIN INITL←1; IFINAL←LINVFC[I]; END
 ELSE BEGIN INITL←LINVFC[I]+1; IFINAL←LINVRC[I]; END;
 FOR K←INITL STEP 1 UNTIL IFINAL DO BEGIN
  IF CV[I,K]<CVMIN OR CV[I,K]>CVMAX THEN GO TO L1;
  J←J+1;
  AVAIL[J]←I;
  COL[J]←K;
  GO TO L3;
 L1: END;
L3: END;
L47: IF J=0 THEN BEGIN
 WRITE(PRINT,FL1004);
 GO TO LX;
END;
COMPAT←J;
FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN
 N←AVAIL[I];
 OMIT[I]←0;
 FOR J←1 STEP 1 UNTIL 4 DO BEGIN
  IF NB[[J]=0 THEN GO TO L41;
  L←0;
  FOR K←1 STEP 1 UNTIL 3 DO IF ABS(NBI[J])=NB[N,K] THEN L←L+1;
  IF L>0 THEN GO TO L40;
 L41: END;
```

```
 OMIT[I]+1;                                                    0063
L40: END;                                                      0064
TAKOUT;                                                        0067
IF COMPAT=0 THEN BEGIN                                         0067
 WRITE(PRINT,FL1005); GO TO LX;                                0069
END;                                                           0074
FOR I+1 STEP 1 UNTIL COMPAT DO BEGIN                           0074
 M+AVAIL[I];                                                   0076
 OMIT[I]+0;                                                    0077
 L+0;                                                          0078
 FOR J+1 STEP 1 UNTIL 4 DO BEGIN                               0079
 DX1+0;                                                        0080
 IF NBI[J]=0 THEN GO TO L43;                                   0080
  FOR K+1 STEP 1 UNTIL 3 DO IF ABS(NBI[J])=NB[M,K] THEN DX1+K; 0082
 IF DX1=0 THEN GO TO L43;                                      0088
 IF RATNGI[J,1]=0 THEN GO TO L42;                              0090
 FOR K+1 STEP 1 UNTIL 3 DO IF RATNGI[J,1]≥RATNG[M,DX1,K] THEN  0092
 L+L+1;                                                        0096
 IF L>0 THEN GO TO L44;                                        0100
 L42: IF RATNGI[J,2]=0 THEN GO TO L43;                         0101
 IF RATNGI[J,2]≥RATNG[M,DX1,4] THEN L+L+1;                     0104
  IF L>0 THEN GO TO L44;                                       0109
 L43: END;                                                     0110
 OMIT[I]+1;                                                    0113
L44: END;                                                      0114
TAKOUT;                                                        0117
IF COMPAT=0 THEN BEGIN                                         0117
 WRITE(PRINT,FL1006); GO TO LX;                                0119
END;                                                           0124
IF DNUFF≠1 THEN GO TO L60;                                     0124
FOR I+1 STEP 1 UNTIL COMPAT DO BEGIN                           0125
 M+AVAIL[I];                                                   0127
```

```
 OMIT[L]←0;                                                       0128
 IF DELTP[N,NUMCVS[M],K]=0 THEN GO TO L61;                        0129
 L←0;                                                             0132
FOR K←1,2 DO                                                      0133
 IF DELTP≤DELTP[N,NUMCVS[M],K] THEN L←L+1;                        0138
 IF L>0 THEN GO TO L61;                                           0143
 OMIT[I]←1;                                                       0144
L61: END;                                                         0145
GO TO L62;                                                        0148
L60: FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN                         0148
 N←AVAIL[I];                                                      0150
 OMIT[I]←0;                                                       0151
 K←COL[I];                                                        0152
 IF DELTP[N,K,1]=0 THEN GO TO L30;                                0153
 L←0;                                                             0156
 FOR DX1←1,2 DO IF DELTP≤ DELTP[N,K,DX1] THEN L←L+1;              0157
 IF L>0 THEN GO TO L30;                                           0166
 OMIT[I]←1;                                                       0168
L30: END;                                                         0169
L62: TAKOUT;                                                      0172
IF COMPAT=0 THEN BEGIN                                            0173
 WRITE(PRINT,FL1007); GO TO LX;                                   0174
END;                                                              0180
FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN                              0180
 N←AVAIL[I];                                                      0181
 OMIT[I]←0;                                                       0182
 IF TYPE[N]="SAN" THEN BEGIN                                      0183
  IF EXTCUL≠1 AND RADFIN≠1 AND SEALI=0 THEN GO TO L64 ELSE BEGIN  0184
   OMIT[I]←1;                                                     0188
   GO TO L64;                                                     0189
  END;                                                            0191
 END;                                                             0191
```

```
IF SEAL[I]>0 THEN GO TO L63;
IF EXTCOL=1 THEN IF EXT[M]=1 THEN GO TO L64 ELSE GO TO L65;
IF RADFIN=1 THEN IF RAD[M]=1 THEN GO TO L64 ELSE GO TO L65;
GO TO L64;
L63: IF SEAL[M]=0 THEN GO TO L65;
L←0;
IF NUMEL≠1 THEN GO TO L66;
FOR K←4 STEP 1 UNTIL 9 DO FOR J←1 STEP 1 UNTIL 4 DO
IF ABS(NBI[J])=NBUN[M,K] THEN L←L+1;
GO TO L67;
L66: FOR K←7,8,9 DO FOR J←1 STEP 1 UNTIL 4 DO
 IF ABS(NBI[J])=NBUN[M,K] THEN L←L+1;
L67: IF L>0 THEN GO TO L64;
L65: OMIT[I]←1;
L64: END;
TAKOUT;
IF COMPAT=0 THEN BEGIN
 WRITE(PRINT,FL[1008]); GO TO LX;
END;
FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN
 M←AVAIL[I];
 OMIT[I]←0;
 IF TYPE[M]="SAN" THEN GO TO L68;
 L←0;
 FOR K←1 STEP 1 UNTIL 10 DO BEGIN
  IF NT[K]=0 THEN GO TO L69;
  FOR J←1 STEP 1 UNTIL 6 DO IF ABS(NT[K])=NT[M,J] THEN L←L+1;
  IF L>0 THEN GO TO L68;
 L69: END;
 OMIT[I]←1;
L68: END;
TAKOUT;
```

```
IF COMPAT=0 THEN BEGIN
 WRITE(PRINT,FL1009); GO TO LX;
END;
IF OP≠1 THEN GO TO L20;
FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN
 M←AVAIL[I];
 OMIT[I]←0;
 IF ONOFF=1 THEN K←NUMCVS[M] ELSE K←COL[I];
 IF DELTP[M,K,1]=0.0 THEN BEGIN DX1←1; GO TO L22; END;
 FOR DX1←1,2 DO IF DELTP≤DELTP[M,K,DX1] THEN GO TO L22;
 L22: IF ACTCOST[M,K,DX1]>0 THEN BEGIN
  IF TOP=1 THEN J←2 ELSE J←4; END ELSE
  BEGIN IF TOP=1 THEN J←1 ELSE J←3; END;
 MAN[I]←J;
 IF MCOST[M,J]>0 THEN GO TO L21;
 OMIT[I]←1;
L21: END;
TAKOUT;
IF COMPAT=0 THEN BEGIN WRITE(PRINT,FL1010); GO TO LX; END;
L20: IF ONOFF=1 THEN GO TO L105;
IF RR≠0 THEN GO TO L105;
IF PNEU=1 THEN GO TO L101;
FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN
 M←AVAIL[I];
IF (MODEL[M] MOD 100)=6 THEN GO TO L102;
 OMIT[I]←1;
L102: END;
TAKOUT;
IF COMPAT=0 THEN BEGIN WRITE(PRINT,FL3000); GO TO LX; END;
GO TO L105;
L101: FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN
 M←AVAIL[I];
```

```
IF (MODEL[N] MOD 100)≠6 THEN GO TO L103;                    0345
 OMIT[I]+1;                                                 0347
L103: END;                                                  0348
TAKOUT;                                                     0351
L105: FOR I+1 STEP 1 UNTIL COMPAT DO BEGIN                  0351
 TOTCOST[I]+0.0;                                            0353
 M+AVAIL[I];                                                0354
 IF ONOFF=1 THEN K+NUMCVS[N] ELSE K+COL[I];                 0355
 IF DELTP[M,K,1]=0.0 THEN BEGIN DX1+1; GO TO L104; END;     0359
 FOR DX1+1,2 DO IF DELTP≤DELTP[M,K,DX1] THEN GO TO L104;    0363
 L104: TOTCOST[I]+ACTCOST[M,K,DX1];                         0371
 FOR J+1 STEP 1 UNTIL 4 DO FOR K+1,2 DO BEGIN               0375
  NBX[J,K]+0; RATNGX[J,K]+0; COSTX[J,K]+0;                  0381
 END;                                                       0387
 FOR J+1 STEP 1 UNTIL 4 DO BEGIN                            0389
  IF NBI[J]=0 THEN GO TO L76;                               0391
  FOR K+1 STEP 1 UNTIL 3 DO                                 0392
   IF ABS(NBI[J])=NB[M,K] THEN BEGIN                        0394
    IF RATNGI[J,1]=0 THEN GO TO L75;                        0396
    FOR DX1+1 STEP 1 UNTIL 3 DO                             0399
     IF RATNGI[J,1]≤RATNG[M,K,DX1] THEN BEGIN               0400
      NBX[J,1]+NB[M,K];                                     0404
      RATNGX[J,1]+RATNG[M,K,DX1];                           0407
      COSTX[J,1]+BCOST[M,K,DX1];                            0410
      GO TO L75;                                            0414
     END;                                                   0415
     L75: IF RATNGI[J,2]=0 THEN GO TO L76;                  0417
     IF RATNGI[J,2]≥RATNG[M,K,4] THEN BEGIN                 0420
      NBX[J,2]+NB[M,K];                                     0424
      RATNGX[J,2]+RATNG[M,K,4];      COSTX[J,2]+BCOST[M,K,4];  0427
     END;                                                   0434
   L76: END END;                                            0434
```

```
FOR J←1 STEP 1 UNTIL 4 DO FOR K←1,2 DO
 IF NBK[J,K]≠0 THEN BEGIN
  J1←J; K1←K; GO TO L80;
 END;
L80: FOR J←1 STEP 1 UNTIL 4 DO FOR K←1,2 DO
IF COSTX[J,K]≠0 THEN
 IF COSTX[J,K]<COSTX[J1,K1] THEN BEGIN
  J1←J; K1←K;
 END;
TOTCOST[I]←TOTCOST[I]+COSTX[J1,K1];
FOR J←1 STEP 1 UNTIL 3 DO IF NBX[J1,K1]=NB[M,J] THEN BEGIN
INDX1[I]←J; FOR K←1 STEP 1 UNTIL 4 DO IF RATNGX[J1,K1]=RATNG[M,J,K]
THEN INDX2[I]←K; END;
IF SEALI≠0 THEN GO TO L74;
IF EXTCOL=1 OR RADFIN=1 THEN BEGIN
 FOR J←1 STEP 1 UNTIL 3 DO
  IF NBX[J1,K1]=NBUN[M,J] THEN K←J;
END ELSE PLAIN←1;
IF PLAIN≠1 THEN TOTCOST[I]←TOTCOST[I]+BONCOST[M,K];
GO TO L77;
L74: IF STEEL≠0 THEN BEGIN FOR J←4,5,6 DO
 IF NBX[J1,K1]=NBUN[M,J] THEN K←J; END
ELSE BEGIN FOR J←7,8,9 DO
 IF NBX[J1,K1]=NBUN[M,J] THEN K←J; END;
 TOTCOST[I]←TOTCOST[I]+BONCOST[M,K];
 L77: FOR J←1 STEP 1 UNTIL 10 DO FOR K←1 STEP 1 UNTIL 6 DO
 IF NTI[J]≠0 THEN
  IF ABS(NTI[J])=NT[M,K] THEN BEGIN
   K1←K; GO TO L79;
  END;
 L79: FOR J←1 STEP 1 UNTIL 10 DO BEGIN
  IF NTI[J]=0 THEN GO TO L78;
```

```
  FOR K←1 STEP 1 UNTIL 6 DO
   IF ABS(NT[J])≠N[M,K] THEN IF TMCOST[M,K]<TMCOST[M,K1] THEN K1←K;
 L78: END;
 TOTCOST[I]←TOTCOST[I]+TMCOST[M,K1];
 TMCOL[I]←K1;
 IF OP≠1 THEN GO TO L90;
 TOTCOST[I]←TOTCOST[I]+MCOST[M,MAN[I]];
 L90: IF POSCOST[M,2]≤0 THEN GO TO L91;
 TOTCOST[I]←TOTCOST[I]+POSCOST[M,2];
 GO TO L92;
 L91: IF POSCOST[M,1]≤0 THEN GO TO L92;
 TOTCOST[I]←TOTCOST[I]+POSCOST[M,1];
L92: END;
FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN
AVAILX[I]←AVAIL[I]; CHEAP[I]←0; END;
J←0;
IFINAL←COMPAT;
L81: J←J+1;
K←1;
FOR I←1 STEP 1 UNTIL IFINAL DO
 IF TOTCOST[I]<TOTCOST[K] THEN K←I;
CHEAP[J]←AVAILX[K];
FOR M←K STEP 1 UNTIL IFINAL-1 DO BEGIN
 TOTCOST[M]←TOTCOST[M+1]; AVAILX[M]←AVAILX[M+1];
END;
IFINAL←IFINAL-1;
IF IFINAL=0 THEN GO TO L82;
GO TO L81;
L82: J←1;
M←0;
L83: IF PNEU≠1 OR HK>0.0 THEN GO TO L84;
IF TANK≠1 THEN GO TO L84;
```

```
IF H157.2/G AND PB>0.25 AND PB≤0.35 THEN GO TO L85;                    0622
L84: IF ONOFF=1 THEN BEGIN K←NUMCVS[CHEAP[J]]; GO TO L86; END;          0626
FOR I←1 STEP 1 UNTIL COMPAT DO IF AVAIL[I]=CHEAP[J] THEN K←COL[I];     0633
L86: CVCN1←CV[CHEAP[J],K]/0.26;                                        0639
NEXTV←0;                                                               0641
FOR I←1 STEP 1 UNTIL 4 DO X[I]←ORIGX[I];                               0642
RUNGKUTTA(D1,D3,SIZE[CHEAP[J]],DV3,IF ONOFF=1 THEN QDT1/2[CHEAP[J]]    0647
 ELSE 0,NEXTV);                                                        0651
IF NEXTV=0 THEN BEGIN M←M+1; CHEAP[M]←CHEAP[J]; END;                   0652
IF M>L THEN GO TO L85;                                                 0656
J←J+1;                                                                 0658
IF J≤COMPAT THEN GO TO L83;                                            0659
L85: END ELIMINATION BLK;                                              0660
                                              0014 IS 0662 LONG, NEXT SEG 0012
BEGIN                                                                  0685
COMMENT PRINTING BLOCK;                                                0685
FORMAT FL4000("WARNING - CHECK DISPLACER COMPATIBILITY"),              0685
                                              START OF SEGMENT ********** 0015
                                              START OF SEGMENT ********** 0016
      FL4001(" (1)",X16,"ORDER SPECIAL DISPLACER"),                    0000
      FL4002("NO TRANSMITTER AVAILABLE"),                              0000
      FL4003("SYSTEM",X18,"COMPONENT",X18,"COST"),                     0000
      FL4004(" (1)",X16,"TYPE 72-25,SERIES 1",X10,"$",F6.2),           0000
      FL4005(" (2)",X16,"TYPE 72-25,SERIES 1",X10,"$",F6.2),           0000
      FL4006(" (2)",X16,"ORDER SPECIAL DISPLACER"),                    0000
      FL4007(" (1)",X24,A1,X20,"$",F6.2),                              0000
      FL4008(" (2)",X24,A1,X20,"$",F6.2),                              0000
      FL4009(" (1)",X16,"MODEL NO. 29501",X14,"$",F6.2),               0000
      FL4010(X20,"VERTICAL SCALE CONTROL",X7,"$",F6.2),                0000
      FL4011(" (2)",X16,"MODEL NO. 29501",X14,"$",F6.2),               0000
      FL4012(" (1)",X16,"ELECTRIK TEL-O-SET DELTP/I    ",F6.2),        0000
      FL4013(X20,"K7165A",X23,"$",F4.2),                               0000
```

```
FL4014(X20,"727070-001",X19,"$",F6.2),                          0000
FL4015(" (2)",X16,"ELECTRIK TEL-O-SET DELTP/I    $",F6.2),      0000
FLO("RR = ",F5.2),                                              0000
FL1(/"PB = ",F5.2),                                             0000
FL4016(" (1)",X16,"R7089 B",X23,"$",F5.2),                      0000
FL4017(X20,"PROBES",X24,"$",F5.2),                              0000
FL4018(" (2)",X16,"R7089 B",X23,"$",F5.2),                      0000
FL2(X20,"VALVE MODEL NO. ",I4/X21,"SIZE",F5.2," INCHES"),       0000
FL3(X21,"TYPE ",A6,"-SEATED"),                                  0000
FL4(X21,"TYPE ANGLE"),                                          0000
FL5(X21,"TYPE SAUNDERS"),                                       0000
FL6(X21,"CV",F7.2),                                             0000
FL7(X21,"ACTUATOR NO. ",A3,X12,"$",F6.2),                       0000
FL30(/"DELTA H = ",F5.2),                                       0000
                                              0016 IS 0256 LONG, NEXT SEG 0015
                                              START OF SEGMENT ********** 0017
FL21(X21,"LINKAGE NO.",A6/X21,"MOTOR NO.",A6,X13,"$",F6.2);     0000
                                              0017 IS 0015 LONG, NEXT SEG 0015
LABEL L12,L3,L4,L5,L6,L7,L8,L9,L10,L11,L20,L21,L22,L27,L28,     0000
 L30,L31,L32,L33,L34,L40,L41,L42,L80,L81,L52,L57,L60,           0000
 L70,L71,L72,L73;                                               0000
FILL NAME[1,*] WITH "CAST I","BRONZE","CARBON","316 S.","MONEL ",   0000
                                              START OF SEGMENT ********** 0018
 "DURIME","HASTEL","HASTEL","440C S","STELLI")                  0002
                                              0018 IS 0010 LONG, NEXT SEG 0015
FILL NAME[2,*] WITH "RON   ","      "," STEEL"," S.   ","      ",   0002
                                              START OF SEGMENT ********** 0019
 "T 20  ","LOY B ","LOY C ",". S.  ","TE    ")                   0004
                                              0019 IS 0010 LONG, NEXT SEG 0015
IF ONOFF=1 THEN GO TO L12;                                      0004
IF RR#0 THEN GO TO L3;                                          0005
IF PNEUM1 THEN GO TO L4;                                        0006
```

```
L3: IF TANK=1 THEN GO TO L5;                                          0007
L11: FOR I←1 STEP 1 UNTIL 10 DO BEGIN                                 0009
 IF ABS(B[I])=1 THEN GO TO L6;                                        0011
END;                                                                  0012
L10: FOR I←1 STEP 1 UNTIL 10 DO BEGIN                                 0015
 IF ABS(B[I])=3 THEN GO TO L7;                                        0016
END;                                                                  0017
WRITE(PRINT,FL4000);                                                  0020
L7: IF TNKRATNG≤300.0 THEN BEGIN K←3; GO TO L8; END;                  0023
IF TNKRATNG≤600.0 THEN BEGIN K←4; GO TO L8; END;                      0028
WRITE(PRINT,FL4001);                                                  0032
SPECIAL←1;                                                            0035
GO TO L9;                                                             0036
L8: FOR I←1 STEP 1 UNTIL 9 DO IF CR<DISPL[I] THEN GO TO L9;           0036
I←9;                                                                  0041
L9: WRITE(PRINT,FL1,PB);                                              0042
IF RR≠0.0 THEN WRITE(PRINT,FL0,RR);                                   0051
WRITE(PRINT,FL4003);                                                  0060
IF SPECIAL=1 THEN BEGIN TOTAL←0.0; GO TO L27; END;                    0063
COST←DISCOST[I,K];                                                    0066
IF RR≠0.0 THEN COST←COST+25.0; TOTAL←COST;                            0068
WRITE(PRINT,FL4004,COST);                                             0071
L27: J←CHEAP[1];                                                      0080
L30: FOR I←1 STEP 1 UNTIL COMPAT DO IF AVAIL[I]=J THEN M←COL[I];      0082
WRITE(PRINT,FL2,MODEL[J],SIZE[J]);                                    0087
IF TYPE[J]≠" DS" AND TYPE[J]≠" SS" THEN GO TO L20;                    0096
IF TYPE[J]=" DS" THEN TYPE[J]←"DOUBLE" ELSE TYPE[J]←"SINGLE";         0101
WRITE(PRINT,FL3,TYPE[J]);                                             0109
GO TO L21;                                                            0119
L20: IF TYPE[J]="ANG" THEN WRITE(PRINT,FL4) ELSE WRITE(PRINT,FL5);    0119
L21: WRITE(PRINT,FL6,CV[J,M]);                                        0130
IF DELTP[J,M,1]=0.0 THEN BEGIN DX1←1; GO TO L22; END;                 0140
```

```
     FOR DX1←1,2 DO IF DELTPISDELTP[J,M,DX1] THEN GO TO L22;          0144
L22: COST←ACTCOST[J,M,DX1]; TOTAL←TOTAL+COST;                         0153
WRITE(PRINT,FL7,ACTNO[J,M,DX1],COST);                                 0156
PRINTZ;                                                               0168
IF CHEAP[2]=0 OR J=CHEAP[2] THEN GO TO LX;                            0168
J←CHEAP[2];                                                           0173
WRITE(PRINT[DBL]);                                                    0174
IF SPECIAL=1 THEN BEGIN WRITE(PRINT,FL4006); TOTAL←0.0; GO TO L30; END;   0177
FOR I←1 STEP 1 UNTIL 9 DO IF CR<DISPL[I] THEN GO TO L28;              0183
I←9;                                                                  0188
L28: COST←DISCOST[I,K];                                               0189
IF RR≠0.0 THEN COST←COST+25.0;                                        0191
TOTAL←COST;                                                           0194
WRITE(PRINT,FL4005,COST);                                             0195
GO TO L30;                                                            0204
L6: IF TNKRATNG≤125.0 THEN BEGIN K←1; GO TO L8; END;                  0204
IF TNKRATNG≤250.0 THEN BEGIN K←2; GO TO L8; END;                      0209
GO TO L10;                                                            0213
L5: IF H1>7.2/G THEN GO TO L11;                                       0213
IF RR≠0.0 THEN GO TO L33;                                             0215
IF PB<0.25 OR PB>0.35 THEN GO TO L11;                                 0217
PB←0.30;                                                              0219
J←0;                                                                  0220
FOR I←1 STEP 1 UNTIL COMPAT DO BEGIN                                  0221
 M←CHEAP[I];                                                          0226
 IF MODEL[M]≠4705 AND MODEL[M]≠1405 AND MODEL[M]≠9105                 0227
 AND MODEL[M]≠9205 THEN GO TO L31;                                    0229
 NEXTV←0;                                                             0232
 FOR K←1 STEP 1 UNTIL 4 DO X[K]←OHIGX[K];                             0233
 RUNGKUTTA(01,D3,S1ZE[M],DV3,0,NEXTV);                                0241
 IF NEXTV=0 THEN BEGIN J←J+1; CHEAP[J]←CHEAP[I]; END;                 0244
 IF J>1 THEN GO TO L32;                                               0244
```

```
L31: END;                                                              0249
IF J=0 THEN GO TO L32;                                                 0252
CHEAP[2]←0;                                                            0253
L32: IF J=0 THEN COMPONENT←"B" ELSE COMPONENT←"A";                     0254
WRITE(PRINT,FL1,PB);                                                   0258
WRITE(PRINT,FL4003);                                                   0266
IF COMPONENT="A" THEN COST←118.0 ELSE COST←147.0;                      0269
TOTAL←COST;                                                            0274
WRITE(PRINT,FL4007,COMPONENT,COST);                                    0275
J←CHEAP[1];                                                            0286
L39: FOR I←1 STEP 1 UNTIL COMPAT DO IF AVAIL[I]=J THEN M←COL[I];       0287
WRITE(PRINT,FL2,MODEL[J],SIZE[J]);                                     0293
IF TYPE[J]≠" DS" AND TYPE[J]≠" SS" THEN GO TO L40;                     0304
IF TYPE[J]=" DS" THEN TYPE[J]←"DOUBLE" ELSE TYPE[J]←"SINGLE";          0307
WRITE(PRINT,FL3,TYPE[J]);                                              0315
GO TO L41;                                                             0325
L40: IF TYPE[J]="ANG" THEN WRITE(PRINT,FL4) ELSE WRITE(PRINT,FL5);     0325
L41: WRITE(PRINT,FL6,CV[J,M]);                                         0336
IF DELTP[J,M,1]=0.0 THEN BEGIN DX1←1; GO TO L42; END;                  0346
     FOR DX1←1,2 DO IF DELTP1≤DELTP[J,M,DX1] THEN GO TO L42;           0350
L42: COST←ACTCOST[J,M,DX1]; TOTAL←TOTAL+COST;                          0359
WRITE(PRINT,FL7,ACTNO[J,M,DX1],COST);                                  0362
PRINTZ;                                                                0374
IF CHEAP[2]=0 OR J=CHEAP[2] THEN GO TO LX;                             0378
J←CHEAP[2];                                                            0379
WRITE(PRINT[DBL]);                                                     0380
IF RR≠0 THEN BEGIN                                                     0383
 COST←85.0; TOTAL←COST;                                                0385
 WRITE(PRINT,FL4011,COST);                                             0386
 COST←310.0; TOTAL←TOTAL+COST;                                         0395
 WRITE(PRINT,FL4010,COST);                                             0397
 GO TO L39;                                                            0406
```

```
END;                                                        0406
COST←IF COMPONENT="A" THEN 118.0 ELSE 147.0;                0408
TOTAL←COST;                                                 0409
WRITE(PRINT,FL4008,COMPONENT,COST);                         0410
GO TO L34;                                                  0422
L33: IF STATIC=0 THEN GO TO L11;                            0422
WRITE(PRINT,FL1,PB);                                        0424
WRITE(PRINT,FL0,RR);                                        0432
WRITE(PRINT,FL4003);                                        0440
COST←65.0; TOTAL←COST;                                      0443
WRITE(PRINT,FL4009,COST);                                   0445
COST←310.0; TOTAL←TOTAL+COST;                               0454
WRITE(PRINT,FL4010,COST);                                   0456
J←CHEAP[1];                                                 0465
GO TO L34;                                                  0466
L4: IF H1514.9/G THEN GO TO L80;                            0466
L52: WRITE(PRINT,FL4002);                                   0468
GO TO LX;                                                   0472
L80: FOR I←1 STEP 1 UNTIL 10 DO                             0474
 IF ABS(B[I])=3 OR ABS(R[I])=4 THEN GO TO L81;              0477
GO TO L52;                                                  0482
L81: WRITE(PRINT,FL1,PB);                                   0483
WRITE(PRINT,FL4003);                                        0492
COST←345.0; TOTAL←COST;                                     0495
WRITE(PRINT,FL4012,COST);                                   0497
COST←125.0; TOTAL←TOTAL+COST;                               0506
WRITE(PRINT,FL4013,COST);                                   0508
COST←95.0; TOTAL←TOTAL+COST;                                0517
WRITE(PRINT,FL4014,COST);                                   0519
J←CHEAP[1];                                                 0528
L60: FOR I←1 STEP 1 UNTIL COMPAT DO IF AVAIL[I]=J THEN M←COL[I];   0529
WRITE(PRINT,FL7,MODEL[J],SIZE[J]);                          0535
```

```
IF TYPE[J]=" DS" THEN TYPE[J]←"DOUBLE" ELSE TYPE[J]←"SINGLE";          0546
WRITE(PRINT,FL3,TYPE[J]);                                                0553
WRITE(PRINT,FL6,CV[J,M]);                                                0561
IF DELTP[J,M,1]=0.0 THEN BEGIN DX1←1; GO TO L57; END;                    0572
FOR DX1←1,2 DO IF DELTP[≤DELTP[J,M,DX1] THEN GO TO L57;                  0576
L57: COST←ACTCOST[J,M,DX1]; TOTAL←TOTAL+COST;                            0585
WRITE(PRINT,FL21,VPLINK[J],ACTNO[J,M,DX1],COST);                         0588
PRIN[2];                                                                 0602
IF J=CHEAP[2] OR CHEAP[2]=0 THEN GO TO LX;                               0602
J←CHEAP[2];                                                              0607
WRITE(PRINT[DBL]);                                                       0608
COST←345.0; TOTAL←COST;                                                  0611
WRITE(PRINT,FL4015,COST);                                                0613
COST←125.0; TOTAL←TOTAL+COST;                                            0622
WRITE(PRINT,FL4013,COST);                                                0624
COST←95.0; TOTAL←TOTAL+COST;                                             0633
WRITE(PRINT,FL4014,COST);                                                0635
GO TO L60;                                                               0644
L12: WRITE(PRINT,FL30,DH);                                               0644
WRITE(PRINT,FL1003);                                                     0653
COST←30.05; TOTAL←COST;                                                  0656
WRITE(PRINT,FL4016,COST);                                                0658
COST←10.0; TOTAL←TOTAL+COST;                                             0667
WRITE(PRINT,FL4017,COST);                                                0669
J←CHEAP[1];                                                              0678
L70: FOR I←1 STEP 1 UNTIL COMPAT DO IF AVAIL[I]=J THEN M←COL[I];         0679
WRITE(PRINT,FL2,MODEL[J],SIZE[J]);                                       0685
IF TYPE[J]≠" DS" AND TYPE[J]≠" SS" THEN GO TO L71;                       0696
IF TYPE[J]=" DS" THEN TYPE[J]←"DOUBLE" ELSE TYPE[J]←"SINGLE";            0699
WRITE(PRINT,FL3,TYPE[J]);                                                0707
GO TO L72;                                                               0717
L71: IF TYPE[J]="ANG" THEN WRITE(PRINT,FL4) ELSE WRITE(PRINT,FL5);       0717
```

```
L72: WRITE(PRINT,FL6,CV[J,M]);                                      0728
IF DELTP[J,M,1]*0.0 THEN BEGIN DX1+1; GO TO L73; END;               0738
FOR DX1+1,2 DO IF DELTP1≤DELTP[J,M,DX1] THEN GO TO L73;             0742
L73: COST+ACTCOST[J,M,DX1]; TOTAL+TOTAL+COST;                       0751
IF (MODEL[J] MOD 100)*6 THEN WRITE(PRINT,FL2],QOLINK[J],ACTNO[J,M,DX1],   0754
 COST) ELSE WRITE(PRINT,FL7,ACTNO[J,M,DX1],COST);                   0766
PRINTZ;                                                             0782
IF CHEAP[2]*0 OR CHEAP[2]*J THEN GO TO LX;                          0782
J+CHEAP[2];                                                         0787
WRITE(PRINT[DBL]);                                                  0788
COST+30.05; TOTAL+COST;                                             0791
WRITE(PRINT,FL4016,COST);                                           0793
COST+10.0; TOTAL+TOTAL+COST;                                        0802
WRITE(PRINT,FL4017,COST);                                           0804
GO TO L70;                                                          0813
END PRINTING BLK;                                                   /0813
                                          0015 IS 0815 LONG, NEXT SEG 0012
LX: WRITE(PRINT[PAGE]);                                             0686
WRITE(PRINT,FL2000,(TIME(1)-Z)/60);                                 0689
END END.                                                            0700
                                          0012 IS 0704 LONG, NEXT SEG 0002
                                          0002 IS 0096 LONG, NEXT SEG 0001
EXP IS SEGMENT NUMBER 0020,PRT ADDRESS IS 0331
LN IS SEGMENT NUMBER 0021,PRT ADDRESS IS 0330
SQRT IS SEGMENT NUMBER 0022,PRT ADDRESS IS 0355
OUTPUT(W) IS SEGMENT NUMBER 0023,PRT ADDRESS IS 0311
OUTPUT(C) IS SEGMENT NUMBER 0024,PRT ADDRESS IS 0307
INPUT(W) IS SEGMENT NUMBER 0025,PRT ADDRESS IS 0304
INPUT(C) IS SEGMENT NUMBER 0026,PRT ADDRESS IS 0301
X TO THE I IS SEGMENT NUMBER 0027,PRT ADDRESS IS 0332
GO TO SOLVER  IS SEGMENT NUMBER 0028,PRT ADDRESS IS 0365
FILE CNTRL(W) IS SEGMENT NUMBER 0029,PRT ADDRESS IS 0014
```

```
     FILE CNTRL(C) IS SEGMENT NUMBER 0030,PRT ADDRESS IS 0015
     READ/WRITE IS SEGMENT NUMBER 0031,PRT ADDRESS IS 0016
NUMBER OF ERRORS DETECTED = 000.  COMPILATION TIME = 0131 SECONDS.
PRT SIZE=0316;TOTAL SEGMENT SIZE=04602 WORDS;DRUM STORAGE REQ.=05109 WORDS;NO. SEGS.=0031.
ESTIMATED CORE STORAGE REQUIREMENT  =  10574 WORDS.
```

LITERATURE CITED

1. L. F. Knappe, "A Computer-Oriented System," *Mechanical Engineering,* vol. 87, p. 35, May, 1965.

2. R. W. Mann, "The CAD Project," *Mechanical Engineering,* vol. 87, p. 41, May, 1965.

3. John Callan, "Analog and Digital Computing Aids in the Design of a numerically controlled Positioning Servo," *Transactions of the American Institute of Electrical Engineers,* Applications and Industry, Pt. 2, p. 33, March, 1961.

4. D. P. Campbell, *Process Dynamics,* p. 52, John Wiley and Sons, Inc., 1958.

5. P. S. Buckley, *Techniques of Process Control,* p. 162, John Wiley & Son, Inc., 1964.

6. D. P. Eckman, *Automatic Process Control,* pp. 241, 249, 257, 263, John Wiley & Sons, Inc., 1958.

7. E. R. Forman and D. J. Oriolo, "Selection of Control Valves for Continuous Chemical Processes," *Proceeding of the Instrument Society of America,* vol. 17, p. 34.2.62-1, 1962.

8. Forman and Oriolo, *op. cit.*

9. Minneapolis-Honeywell Regulator Co., *Process Control Valves,* Catalog C800-1b, 1962.

10. Eckman, *op. cit.,* p. 229.

11. Forman and Oriolo, *op. cit.*

12. Eckman, *op. cit.,* p. 225.

13. Eckman, *op. cit.,* p. 225.

14. Forman and Oriolo, *op. cit.*

15. D. G. Shepard, *Elements of Fluid Mechanics,* Harcourt, Brace & World Inc., 1965, pp. 133-145.

16. V. L. Streeter, *Fluid Mechanics,* p. 358, McGraw-Hill Book Co., Inc., 1958.

17. P. J. Potter, *Power Plant Theory and Design,* p. 60, Ronald Press, 1959.

18. Crane Co., *Flow of Fluids,* p. A-30, Crane Industrial Products Group, 1957.

19. Eckman, *op. cit.*

20. Buckley, *op. cit.*

21. Minneapolis Honeywell, *Pneumatic Control and Transmission Systems,* p. 6, Minneapolis Honeywell Regulator Co., 1954.

22. Buckley, *op. cit.,* p. 189.

23. B. A. Knight, *Vibration Analysis of Flexible Cam Follower Systems,* Unpublished thesis, Georgia Institute of Technology, 1965.

24. F. A. Willers, *Practical Analysis,* Dover Publications, 1948.