Durham University

Durham E-Theses

Microprocessor-based on-load valve sequencing for a turbo-alternator

M. R. Ozgur

How to cite:

Ozgur, M. R. (1979) Microprocessor-based on-load valve sequencing for a turbo-alternator. Masters thesis, Durham University.

Use policy

The full-text may be used and/or reproduced, and given to third parties in any format or medium, without prior permission or charge, for personal research or study, educational, or not-for-profit purposes provided that:

- a full bibliographic reference is made to the original source
- a https://etheses.durham.ac.uk/id/eprint/9131/ is made to the metadata record in Durham E-Theses
- the full-text is not changed in any way

The full-text must not be sold in any format or medium without the formal permission of the copyright holders.

Please consult the full Durham E-Theses policy for further details.

Academic Support Office, The Palatine Centre, Durham University, Stockton Road, Durham, DH1 3LE
e-mail: e-theses.admin@durham.ac.uk Tel: +44 0191 334 6107
https://etheses.durham.ac.uk

MICROPROCESSOR-BASED ON-LOAD VALVE SEQUENCING FOR A TURBO-ALTERNATOR

by

M. R. OZGUR, B.Sc.

Thesis submitted for the Degree of Master of Science in the Faculty of Science, University of Durham.

The copyright of this thesis rests with the author. No quotation from it should be published without his prior written consent and information derived from it should be acknowledged.

April, 1979.

University
SCIENCE
1979
SECTION
Library

ABSTRACT

An analog on-load valve sequencing system that is used to test the performances of the valves on a turbo-alternator can be replaced by a microcomputer testing system. On the way to the full-scale computerising of turbo-generator control systems, this type of testing system may be used under the control of a full-size supervisory computer. This microprocessor-based testing system provides test sequencing of the valves of an Electro-Hydraulic Governor. The hardware for the interfacing and scaling, and the firmware for the microprocessor are developed for one valve. The extension of the technique to the task of testing multiple valves in sequence is also discussed.

ACKNOWLEDGEMENTS

The author would like to express his sincere gratitude to the following:

Dr. C. Preece, my supervisor, for his overall supervision on this project and his encouragement throughout this work.

The Electrical Laboratory technicians, and in particular, Mr. T. Nancarrow, Electronics Technician, for his advice and help in building the circuits.

Special appreciation is given to Madelaine Paterson for her assistance in typing this thesis.

C. A. Parsons & Co. Ltd., Newcastle Upon Tyne, is acknowledged for providing the author with technical knowledge of their governor model.

SUMMARY

Steam inlet valves on the turbine of a turbo-alternator control the mechanical input power of the alternator. In turn, this input power effects the electrical output power of a generator. Fast valving control of the output power improves the system stability during the transient disturbances. For this reason, the inlet valves must operate efficiently and reliably.

In time, steam inlet valves get worn or overlapped because of the mechanical and chemical effects and their efficiency decreases. The valves are tested remotely from the central control rooms and the permanent records of the valves performance provided by an X-Y plotter are collected, and the new records are checked against the previous ones. Comparison of the test traces with previous records highlights any change in valve permormance. The records also indicate failure of valve operating systems, majority-voting circuits and valve tripping systems.

A typical Valve Governor System with its specifications was obtained from a manufacturer and reduced to be established on an analog computer. Then the mathematical model of the Governor on the analog computer was connected to a microprocessor over a designed hardware interfacing system.

The software was developed to do the same test steps as done on actual valve. The test results are given. Some discussion is included on how the microprocessor-based on-Load valve testing technique can be extended to achieve the task of testing a number of valves.

CONTENTS

LIST OF FIGURES

CHAPTER I

TURBO-GENERATORS

This chapter briefly describes some of the basic characteristics of synchronous machines and also provides the reader with an understanding of the methods of control of turbo-generators.

Attempts have been made to outline the essential points of a brief specification. The emphasis is on steam turbine driven generators and an outline analysis is given of such a machine.

1.1 Description and Torque Concept

In the synchronous machines field windings are supplied with direct current and alternating currents are obtained from or impressed on armature windings. Depending on the rotor shapes, synchronous machines are divided into two groups; salient pole machines and round-rotor machines.

Usually the armature windings are placed on the stators and the field windings on the rotors. Synchronous machines run at constant speed. The flux wave created by the armature current rotates in the air-gap at the same speed as the rotor rotates. So it looks stable as it is viewed from the rotor. The interaction between the magnetic field created by the field current and the magnetic field created by the armature current creates electro-magnetic torque. This torque is a function of the angle between the magnetic axis of the stator and the rotor or alternatively between the stator and rotor magnetomotive force waves (mmf) and opposes rotation in the generators. The electro-magnetic torque has a negative sign

Durham SCIENCE - 7 ... SECTION

for generator action where the negative sign indicates that the electro-magnetic torque acts in the direction to decrease the displacement angle between the field of stator and rotor, and it has a positive sign for motor action where the positive sign of the torque indicates that the electro-magnetic torque acts in the direction to bring these fields into alignment. A steady torque is produced if the displacement angle is kept constant which means that the stator and rotor magnetic fields travel at the same speed, as the magnetic fields of stator and rotor have constant amplitudes.[8]. Figure 1.1.

1.2 Voltage - Current Relations

The equations giving the four voltage-current relations in a three phase synchronous machine are

$$v_a = r i_a + \frac{d\psi_a}{dt} \quad (1\text{-a})$$

$$v_b = r i_b + \frac{d\psi_b}{dt} \quad (1\text{-b})$$

$$v_c = r i_c + \frac{d\psi_c}{dt} \quad (1\text{-c})$$

$$v_f = r_f i_f + \frac{d\psi_f}{dt} \quad (1\text{-d})$$

where vs are the terminal voltages of the windings; i_a, i_b, i_c are the armature currents in the associated windings, i_f is the field current. r is the resistance of the armature winding and r_f is the resistance of the field winding. ψ is the flux linkage of each winding and can be written as functions of currents, and the self and mutual inductances of each winding so that one can obtain four voltage equations as functions of these quantities. Since this gives a very complicated set of equations, a set of imagined voltages, currents and flux linkages are defined as functions of the actual variables. These new electrical variables called "Direct-axis and Quadrature-axis Variables" can be solved as functions of time and this solution can help to find the actual variables as functions of time. For a three phase synchronous machine, at the instant of Figure 1.2, the relation between the fictitious voltages and the actual voltages are as follows

$$\begin{bmatrix} v_d \\ v_q \\ v_0 \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos\theta & \cos(\theta-120^\circ) & \cos(\theta+120^\circ) \\ -\sin\theta & -\sin(\theta-120^\circ) & -\sin(\theta+120^\circ) \\ 1/2 & 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \qquad (2)$$

$$\begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 1 \\ \cos(\theta-120^\circ) & -\sin(\theta-120^\circ) & 1 \\ \cos(\theta+120^\circ) & -\sin(\theta+120^\circ) & 1 \end{bmatrix} \begin{bmatrix} v_d \\ v_q \\ v_0 \end{bmatrix} \qquad (3)$$

These equations can also be rewritten for the flux linkages and currents, by replacing v with ψ and i respectively. In the equations above, v_d is the direct-axis voltage, v_q is the quadrature-axis voltage and v_0 is the zero-sequence voltage.

The self and mutual-inductances of a synchronous machine is the function of the displacement angle θ between the direct axis and the phase axis, except the self-inductance of the field winding because the stator has a cylindrical shape.

Flux linkages as functions of currents can be obtained

$$\psi_d = \left(L_s + M_s + \frac{3}{2} L_m\right) i_d + M_f i_f \qquad (4\text{-a})$$

$$\psi_q = \left(L_s + M_s - \frac{3}{2} L_m\right) i_q \qquad (4\text{-b})$$

$$\psi_0 = \left(L_s - 2M_s\right) i_0 \qquad (4\text{-c})$$

$$\psi_f = \frac{3}{2} M_f i_d + L_{ff} i_f \qquad (4\text{-d})$$

where L_s is the component of the self-inductance of one phase of the armature which is independent of the angular displacement θ , M_s is the component of the mutual inductance between two armature phases which is independent of the angle, M_f is the amplitude of the mutual inductance between the field winding and any armature phase, L_{ff} is the self-inductance of the field winding, L_m is the amplitude of the component of the stator self-inductance which is dependent on the angular displacement. Eqs.4 may be written

$$\psi_d = L_d i_d + M_f i_f \tag{5-a}$$

$$\psi_q = L_q i_q \tag{5-b}$$

$$\psi_o = L_o i_o \tag{5-c}$$

$$\psi_f = \frac{3}{2} M_f i_d + L_{ff} i_f \tag{5-d}$$

where

$$L_d = L_s + M_s + \frac{3}{2} L_m , \tag{6-a}$$

$$L_q = L_s + M_s - \frac{3}{2} L_m \tag{6-b}$$

and

$$L_o = L_s - 2M_s \tag{6-c}$$

L_d , L_q and L_0 are constant quantities and called direct-axis synchronous inductance, quadrature-axis synchronous inductance and zero-sequence inductance, respectively. These machine constants can be determined from tests.

From the equations given above, the following voltage-current relations are obtained

$$v_d = r i_d + L_d \frac{di_d}{dt} - \omega L_q i_q + M_f \frac{di_f}{dt} \tag{7-a}$$

$$v_q = r i_q + L_q \frac{di_q}{dt} + \omega (L_d i_d + M_f i_f) \tag{7-b}$$

$$v_0 = r i_0 + L_0 \frac{di_0}{dt} \tag{7-c}$$

$$v_f = r_f i_f + L_{ff} \frac{di_f}{dt} + \frac{3}{2} M_f \frac{di_d}{dt} \tag{7-d}$$

These equations are for motor action. For generator action the currents i_d and i_q change the signs so that the equations become

$$v_d = -r i_d + \omega L_q i_q - L_d \frac{di_d}{dt} + M_f \frac{di_f}{dt} \tag{8-a}$$

$$v_q = -r i_q - L_q \frac{di_q}{dt} + \omega M_f i_f - \omega L_d i_d \tag{8-b}$$

$$v_f = r_f i_f + L_{ff}\frac{di_f}{dt} - \frac{3}{2}M_f\frac{di_d}{dt} \qquad (8\text{-}c)$$

[8,11] .

1.3 Power Equations and The Concept of Stability

In the steady state, the power output of a salient-pole generator is given as

$$P = E_q V \frac{r^2+X_q^2}{r^2+X_d X_q}\sin(\delta+\alpha) - V^2\frac{r}{r^2+X_d X_q} + V^2\frac{X_d - X_q}{2(r^2+X_d X_q)}\sin 2\delta \qquad (9)$$

where E_q is the steady-state internal voltage of the generator, V is the terminal voltage, δ is the phase angle between the internal voltage E_q and the terminal voltage V , r is the resistance of the armature winding, $X_d = \omega L_d$ is called the direct-axis synchronous reactance and $X_q = \omega L_q$ is called the quadrature -axis synchronous reactance. (ω is the stator angular frequency and $\theta = \omega t$ where θ is the displacement angle). Here

$$\sin\alpha = \frac{r}{\sqrt{r^2+X_q^2}} \qquad (10)$$

and

$$\cos\alpha = \frac{X_q}{\sqrt{r^2+X_q^2}} \qquad (11)$$

If the machine is connected to the infinite bus V is the bus voltage, if it is connected to another machine V is the internal voltage of the second machine. If the connection was made through an internal impedance, r , x_d and x_q would be added to the external impedance when the equivalent circuit replacement of these impedances was being done. If the resistance is negligible, this power output equation of the generator simplifies to

$$P = \frac{E_q V}{X_d} \sin \delta + V^2 \frac{X_d - X_q}{2 X_d X_q} \sin 2\delta \qquad (12)$$

This equation was shown graphically in Figure 1.3.

If the machine had round rotor, x_d would be equal to x_q and therefore the reluctance power due to saliency represented by the second harmonic term would cease to exist, so that the power equation for the round rotor machine becomes

$$P = \frac{E_q V}{X_d} \sin \delta \qquad (13)$$

In the case of the constant E_q and V , the maximum power occurs at $\delta = 90^\circ$ and this power limit is known as the Steady State Stability Limit.

The values of the voltages and the currents are rms values.

In the transient state, the power-angle expression becomes

$$P = \frac{E_q' V}{X_d'} \sin \delta + V^2 \frac{X_d' - X_q}{2 X_d' X_q} \sin 2\delta \qquad (14)$$

The voltage E_q' is the quadrature-axis voltage behind transient reactance, and x_d' is the direct-axis transient reactance. The second harmonic is negative since usually $x_q > x_q'$. The amplitude of the transient power-angle curve is greater than that of the steady state. So, if the duration of the suddenly applied overload is short, synchronism will not be lost. Refer to Figure 1.4.

For a synchronous machine the accelerating power is given as

$$P_a = P_i - P_o = I \omega \frac{d^2 \delta}{dt^2} \qquad (15)$$

where P_i = Input power of the machine

P_o = Output power of the machine

I = Moment of inertia

w = Angular velocity

and

δ = Angular displacement of the machine with respect to another machine or to the infinite bus to which it is connected.

At steady state the accelerating power equals zero and the power input equals the power output. Negative P_a represents retarding power and positive P_a represents accelerating power. The P_a versus δ curve shows that the system is unstable if the angle δ increases without limit. For generator action P_i is the mechanical input power and P_e is the electrical output power. As derived earlier, this electrical output power is represented by the sinusoid on the power-angle curve.

The equal-area criterion is used to determine the Transient Stability Limit. As the assumptions of constant input power, constant voltage behind transient reactance and no damping are made, the generator and the infinite bus are in balance at point a (Figure 1.5) on the pre-fault power curve. P_i is constant and therefore is parallel to the horizontal axis. The power-angle curve during the fault is lower than the pre-fault curve. When a fault occurs electrical power output decreases and the operating point a drops to the point b on the fault output curve. At this point mechanical input power exceeds electrical output power. This accelerates the unit causing the angle δ increase. The acceleration causes the electrical output power to increase to the point c. Since at point c the accelerating power decreases but the speed of the machine is still high the angle increase to δ_m. This results in retarding power on the generator. Therefore the speed decreases as does the angle, and the operating point moves back towards c. The generator will not pull out of step during the transient disturbance if the area A_1 equals or is less than the area A_2 on the diagram. This is the definition of the transient stability limit. If the area A_1 is greater than the area A_2, the generator stays in synchronism with the system.

1.4 <u>The Voltage Regulator</u>

One of the ways of improving the stability margin is by increasing the internal voltages of the synchronous machine.

Since the power output of a synchronous machine is proportional to its excitation voltage this output power is controlled by varying the field current. This variation may be achieved by use of an Automatic Voltage Regulator (AVR) or by manual control. By increasing the excitation voltage the amplitude of the transmitted power equation increases. When the steady-state limiting angle exceeds 90° a fast acting AVR increases the applied voltage to the field circuit of the machine; in other words the demagnetising effect of the armature current is being opposed by maintaining the decaying field flux linkage by means of AVR.

1.5 <u>Fast Turbine Valve Control</u>

The technique of fast turbine valve control has been introduced elsewhere by some authors[4]. The principle idea of this technique is to enhance generating-unit stability for transient disturbances by increasing the critical fault clearing time.

Soon after a fault recognition the controller is designed to close the turbine valves for a short time period related to the time-integral of the difference between input power and output power, and then re-opened slowly. This action of the valves results in a decrease in the mechanical input power. By this means, the difference between the reduced generator electrical output power and mechanical input power is reduced quickly.

On referring to the power-angle curve in Figure 1.6, it can be seen that if a fault occurs while the turbine generator is in balance with the power system at point P, the electrical output power of the generator will drop to the point P1. After the fault clearance, the system is stable at newly established working point P2 on the post-fault power-angle curve. In the diagram of Figure 1.6, the mechanical input power is shown as a function of δ which arises from the fast valve operation. It is indicated by the dotted line. Fast valve action starts as the fault occurs.

Since the transient stability limit is given with $A_1 \geq A_2$, as the fast valve control concept is applied it is apparent that a significantly increased critical fault clearing time arises. [1].

1.6 Steam Turbine Governor Operation

The decreased inertia-to-torque ratio on large turbo-generator units increases the control complexity of turbines so that electro-hydraulic control systems essentially replaced the conventional mechanical hydraulic governing systems.

In basic form, an electro-hydraulic governor is illustrated in Figure 1.7 [9]. Three main feedback loops are shown dotted. To this figure, a main stop valve just ahead of the main governing valves and a reheater stop valve just to the reheater output may be added [4].

There are two different modes of governor operation; "Full-Arc Admission" and "Partial-Arc Admission". As a brief explanation; in "Full-Arc Admission" operation, the control of steam flow to turbine is accomplished by throttling the main steam through main stop valve while the governor valves are fully open. In the later mode, however, the main steam is throttled through control valves while the stop valve is wide open. Full-Arc Admission operation is done during wide range speed control and Partial-Arc Admission operation is done during normal operation.

In conventional mechanical-hydraulic governor systems the speed and the load reference signals were both applied before the regulating element as shown in Figure 1.8. The block 'Valve Operators of Steam Volumes' represents all valve controlling loops in one generator control system.

An attempt to overcome the difficulty that arises during the transfer of the operation modes, from one mode to the other, has been implemented in electro-hydraulic control systems by making the load reference signal independent of regulation [4]. Refer to Figure 1.9.

Another proposed approach uses a conditional loop so that the selected higher loop error assumes the control [9]. Figure 1.10.

For the purpose of generator protection against overspeed there are usually two independent emergency governors provided. The one of them is Normal Speed Governor that operates Control Valves and Interceptor Valves, and the other is an emergency Trip system that shuts the Main Stop Valve and the Reheater Stop Valve on overspeed conditions.

Since the input power of a turbo-generator is controlled by means of steam inlet valves, stability can be markedly altered by their action. They must operate efficiently and reliably.

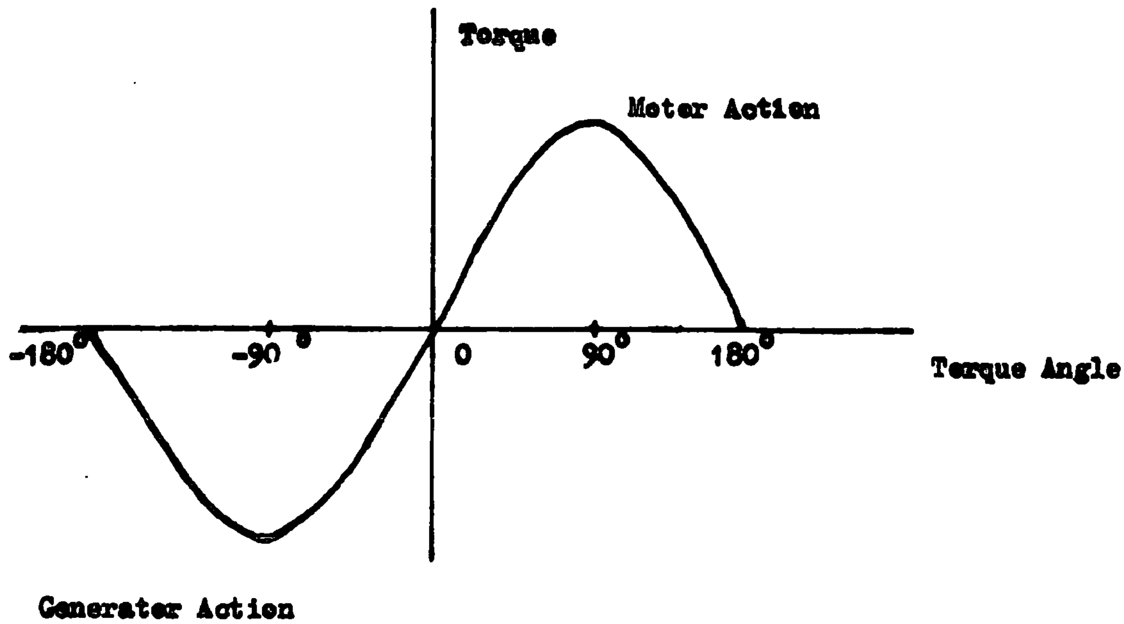


Figure 1.1 Torque-Angle Curve of a Synchronous Machine

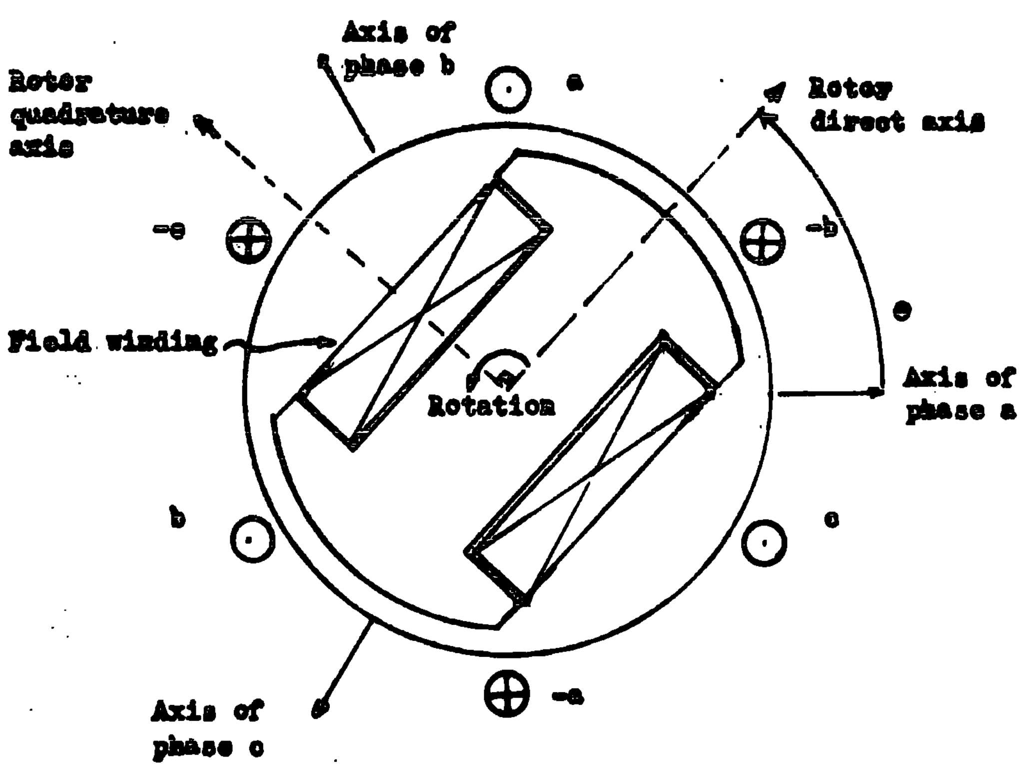


Figure 1.2 Elementary salient-pole 3-phase synchronous machine

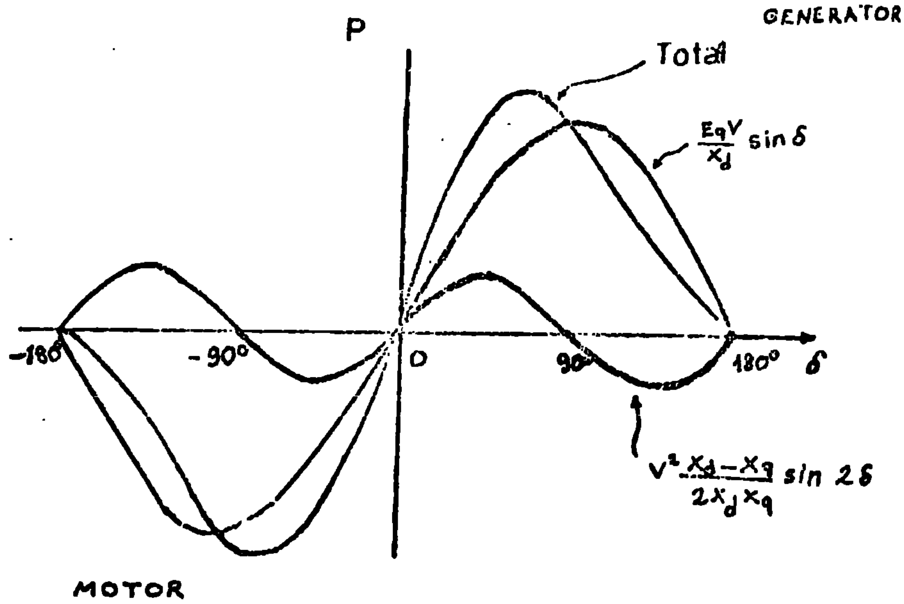


Figure 1.3 Steady-state power-angle curve of salient-pole synchronous machine.

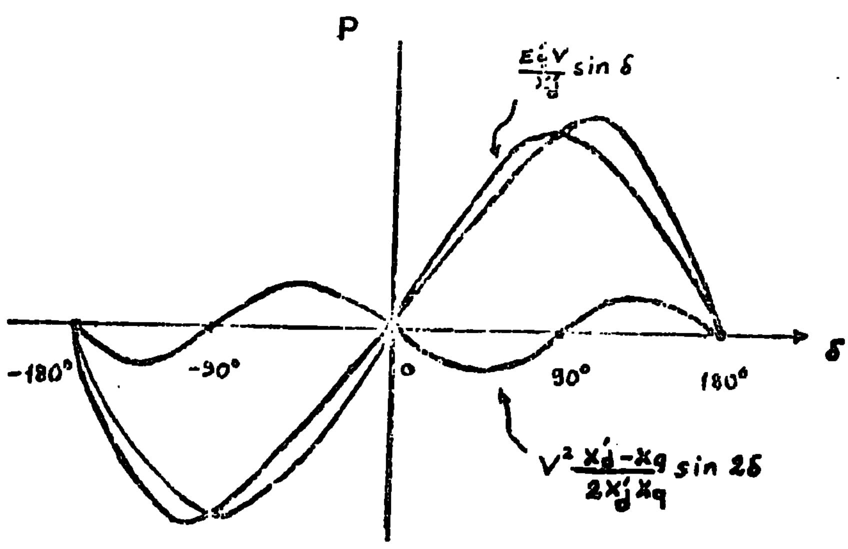


Figure 1.4 Transient power-angle curve of salient-pole synchronous machine.

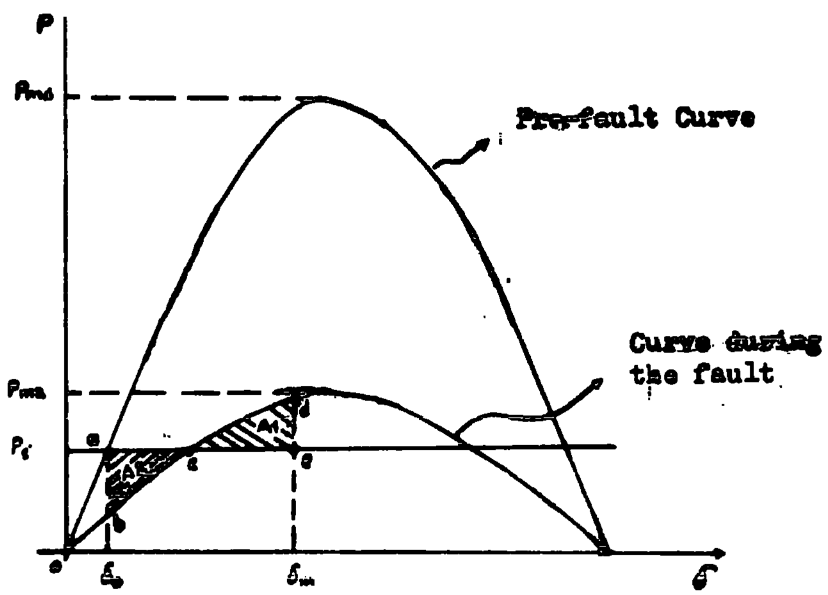


Figure 1.5 Determination of stability limit
by the equal-area criterion.

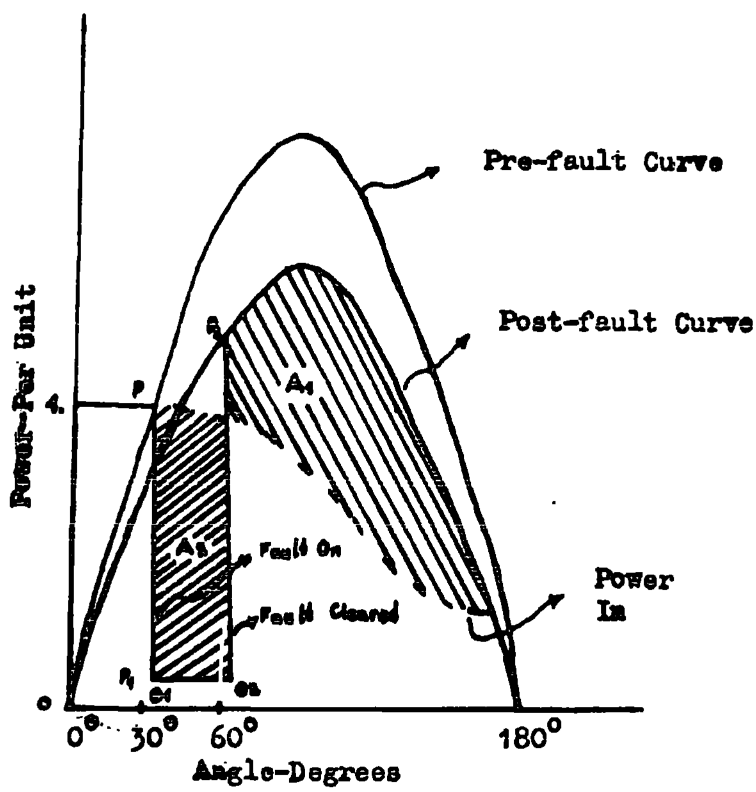


Figure 1.6 Transient-stability improvement by fast valve action.

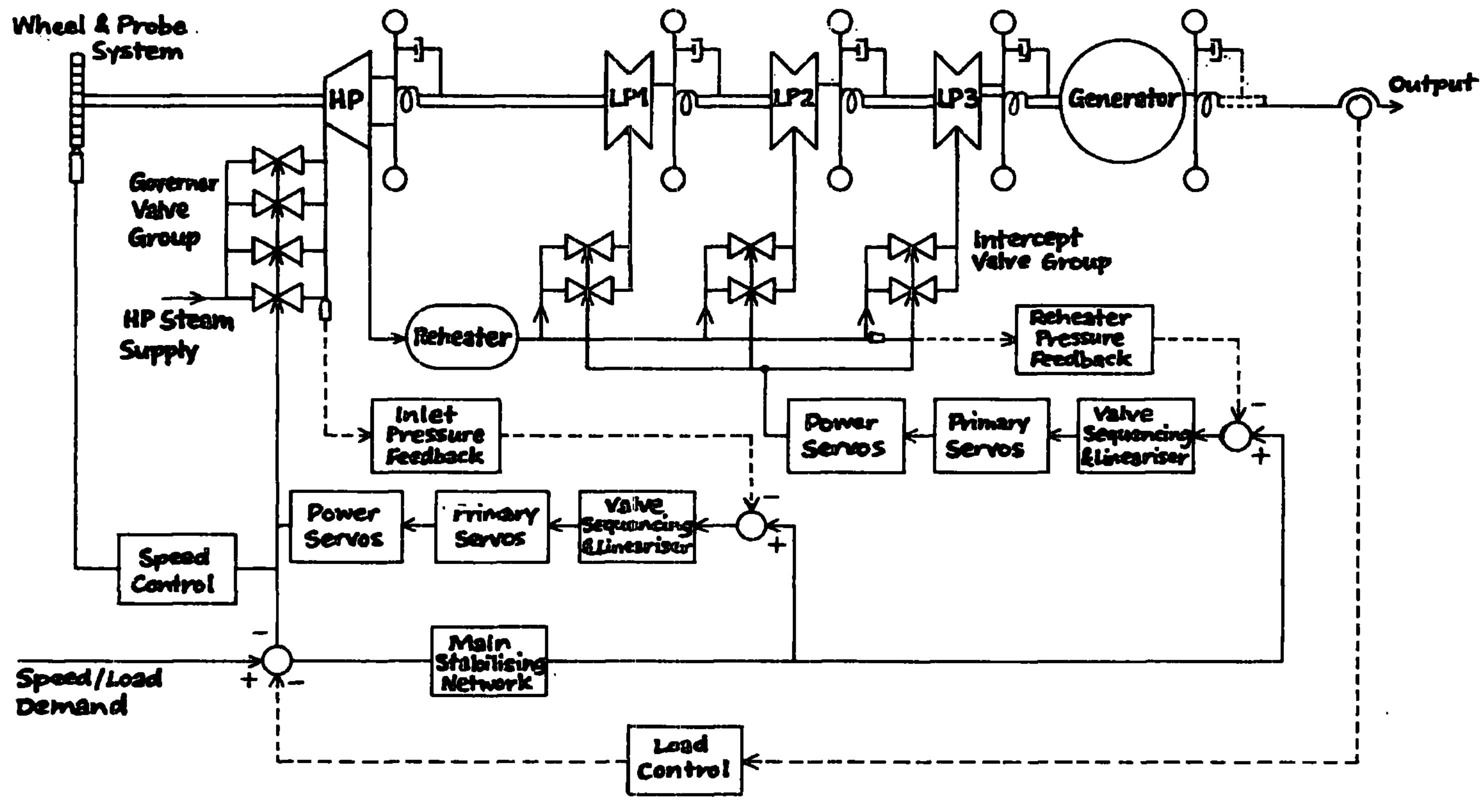


Figure 1.7 A typical Electro-Hydraulic Governor system block diagram of a Turbo-Generator

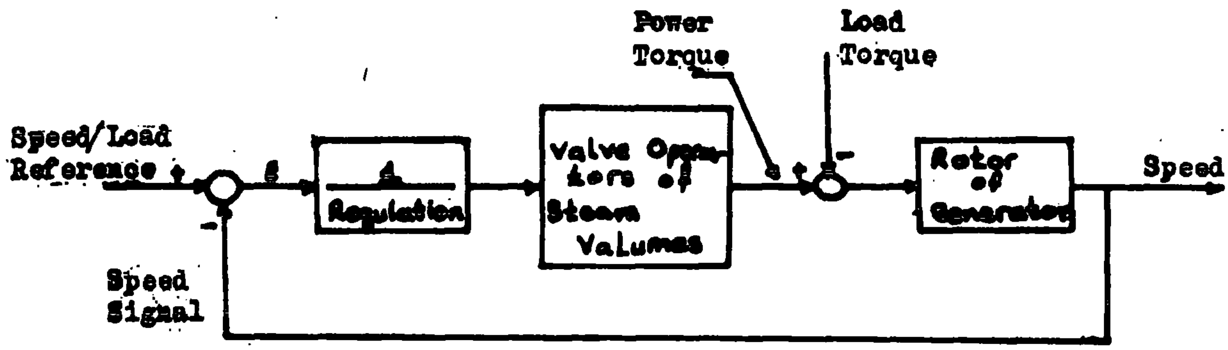


Figure 1.8 Block diagram of speed control loop for conventional turbine control system.

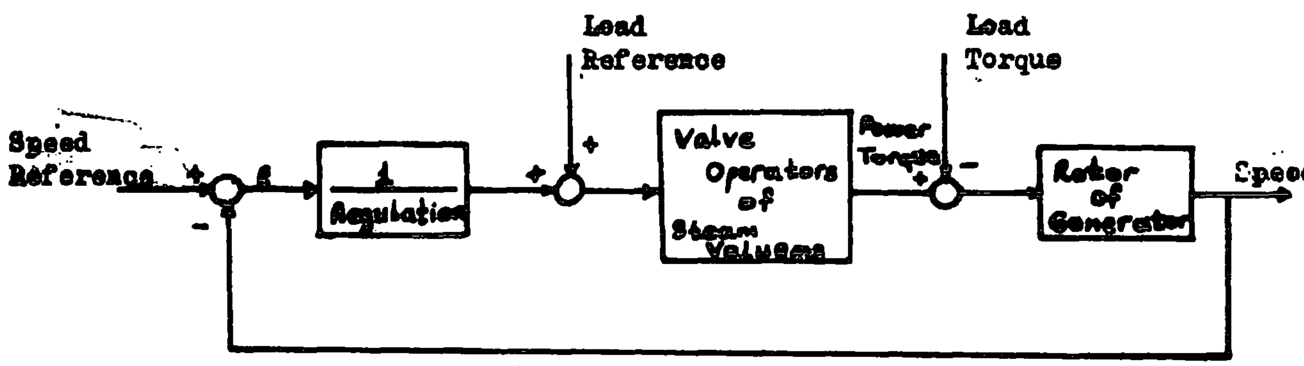


Figure 1.9 Block diagram of "Separate Speed and Load" controls for an electro-hydraulic control system.

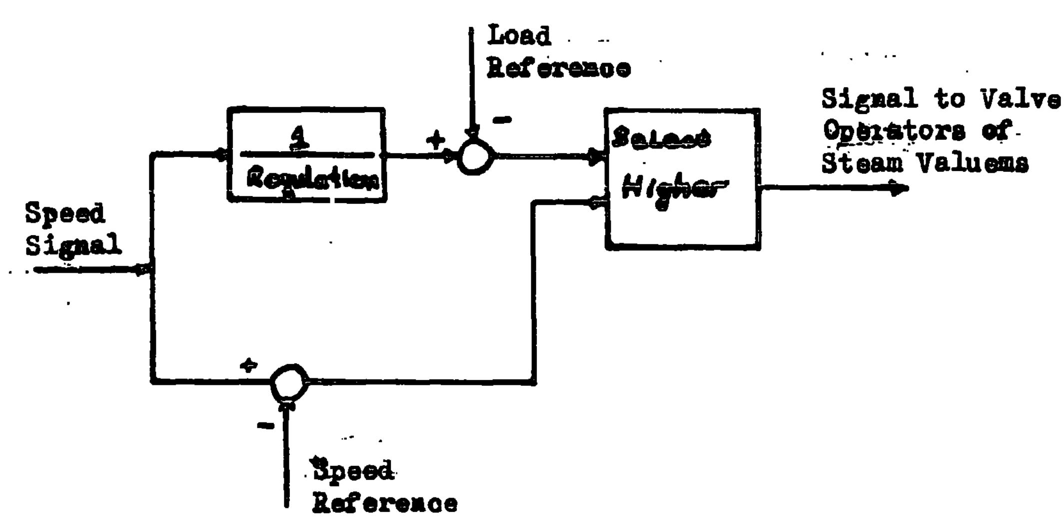


Figure 1.10 Block diagram of "Speed and Load Control by Conditional Loop" system for an electro-hydraulic control system.

CHAPTER II

GENERAL VIEW OF MICROPROCESSOR-BASED CONTROL SYSTEMS

The technical and economic features of microprocessor-based control systems are investigated in this chapter. This investigation has been extended into the use of some technical hardware arrangements. Some of the important hardware configurations for the reliability aspects of on-line computer control systems are also discussed.

2.1 The Convenience of the Installation of Microprocessors in Control Systems

Control equations can be formulated mathematically in a way such that digital computers can perform the same tasks as analog controllers. However, the choice of on-line computer control systems is limited since for single loop control problems they may provide more complex solutions than an analog electrical control system. As the complexity of control problems increases, both analog electrical and mechanical control systems soon become more and more complex whereas this is not necessarily the case for digital controllers. In fact, complexity of the controller directly effects the technical and commercial reasons for the choice of the kind of control system to be used, and both of these reasons are as important in the justification of on-line computer control systems as they are for the justification of the other type controllers.

Microprocessor-based control systems may replace analog electronic or other forms of hardwired digital electronic, mechanical and computer control systems. [20,3,22,10,12]. Since they are small, fast and have low cost computing power, and they can perform many functions minicomputers do, these systems can improve control and do it for a reduced cost. They offer advantages in performance, flexibility, maintenance, cost and size. [22,12]. They allow the control systems to perform logical operations and give intelligence to control systems, [18], and they are also easy to program. Indirect measurements which are difficult to measure with analog controllers can be obtained in microprocessor-based control systems by calculation, [7]. They can also be used as replacements for special/complex equipment which would otherwise be required to implement the control. They enhance accuracy [22]; especially with digital transducers [13], rather than with nonlinear analog transducers. Digital signals can be easily transmitted without distortion whereas analog signals are prone to error because of their hard-wired couplings and temperature variations. Since microprocessors are very reliable electronic devices and give exact calculations, they provide increased accuracy in control systems. As word length gets longer, accuracy of the machines increases. In the market, 4-bit, 8-bit, 12 bit and 16 bit machines exist; such as 4040 (4-bit), M6800 (8-bit) and CP1600 (16-bit).

Microprocessor-based controllers can be reprogrammed and their hardware reconfigured with minimal physical modification. The same hardware can therefore be used to obtain an entirely different controller for a variety of applications. This sort of modification increases the system flexibility. Since microprocessors are much simpler than full size computers and can replace a number of analog controllers, as a central element in a control system [25] , it is easier to locate a failure and replace it in microprocessor-based process control areas.

Microprocessors are inexpensive electronic devices. They can be used to provide a single control for a number of machines. Because microcomputers replace a number of analog control loops, the total cost of the replaced equipment helps to meet a considerable part of the cost of these computers. Because of their high performance and accuracy, in some systems they reduce system power consumption and thus provide an economic advantage, [26]. They also increase the bulk of product since they are so fast [22 ,12]. All these make them an economic solution for many applications.

In large control systems microprocessors share the control loops and achieve the control under the supervision of a master computer, [18]. In the control hierarchy microprocessors cannot perform the management functions because of their size. [22,18] . Coordinating entire complex systems, generating up-to-date reports, doing cost accounting, etc., require extensive storage and a single microcomputer may not meet these demands.

2.2 <u>Some Hardware Arrangement Techniques Used In Conjunction With Microprocessors</u>

Some authors have made proposals to keep the number of A/D and D/A converters used with one microprocessor down.[2,5,20,21] This approach results in a drop in the amount of the input and output port addresses necessary. To utilise only one A/D and only one D/A converter for more than one input and one output of a microcomputer, analog input and analog output multiplexors are proposed. These multiplexors get control commands to select the desired channels, Figure 2.1. As either of these two multiplexors receives a channel select signal, the converter connected to that multiplexor will be switched over to that chosen channel. After the selection of the channel, another command starts A/D or D/A conversion, depending on which is required. Thus one input address to the A/D and one output address to the D/A are dedicated. Track-hold circuits are proposed to keep the actuator at its latest position until the next data is available. [21, 5, 20].

In a microprocessor-based position controller of machine tools a register which is essentially an electronic device consisting of flip-flops was utilised at the digital side of each digital to analog converter. The first output data waits in one of these registers until the other output data are housed in the remaining registers. Then these data are sent to the D/A's. This gives an opportunity to apply all output data almost simultaneously, refer to Figure 2.2. The latch registers duplicate their data towards the D/A's as they receive individual control signals. [21].

A thyristor cycloconverter converts an alternating voltage of one frequency to an alternating voltage of another frequency by opening and closing the switches within the converter in appropriate manner. A microprocessor is used to calculate these triggering instants and to achieve the control of a three pulse-cycloconverter. The equation that gives the SCR triggering instants needs sine and arcsine calculations. To increase the speed, a table look-up technique is utilised. This reduces the software overhead time. Again for the speed increment, the author used the Advance Micro Devices (multiplication hardware) for 16 bit multiplication in a short time of 3 μsec, [3]. Table look-up techniques and multiplication hardware have been used also by some other authors to meet the demands of fast acting control systems where the process time is prime concern. Refer to [10] and [16].

2.3 Reliability

Considering technical and economic reasons it can be seen that there are many advantages in employing on-line computer control systems. However, since these digital computer controllers are required to perform a multitude of functions, attention must be paid to reliability. In conventional analog control systems, each functional task is performed by a dedicated hardware element and the failure of that particular element ends with the loss of that function. But, if a computer failure occurs in an on-line computer control system, we lose all control. This loss of control cannot be tolerated in most cases. In order to be able to overcome this problem, a number of different ways have been considered. Complete redundancy is one of these approaches. But this is not economic since twice as much money must be spent to obtain a reliable computerised process control.

The fact that the cost of microprocessor has become so low made it possible to employ more than one microprocessor in one control scheme in order that each microprocessor is to perform only a small number of functions. To do so, one microprocessor failure causes only the loss of the functions performed by this dedicated microprocessor. Thus, the control still exists since the other functions remained as they were. But this way of solution may not be satisfactory, since one microprocessor is obliged to perform only four or eight functions although it is capable of performing many more functions alone.

More than one computer in parallel was used as an alternative method of achieving reliability. If one of them fails the other one assumes control.

A multiple processor method is used where a switchover mode is acceptable, Figure 2.3. Here are, for example, two processors connected to the same inputs in parallel, and the same outputs can be switched over from one processor to the other. [2] .

For the reliability and survivability aspects an aircraft fly-by-wire system uses four computer and voter units. When a failure occurs on the computer that has been doing the control tasks, the control tasks start to be achieved by one of the other three computers. [6].

A backup system assumes the tasks to control the system as the computer fails, if manual control is acceptable as the backup mode. After the computer failure has occurred, the operator controls the process manually. In such a control system, only a couple of most important functions may be performed by the operator in order to keep the most important and small part of the system running while the failure is being located and replaced,

Figure 2.4. [2].

Where safety is of main concern all controlling signals should be checked before they have reached the process. In a gunpowder manufacturing plant, two computers are connected in series to perform the control tasks of the hazardous process. One of these computers is a complex one and works at supervisory level. The other one is a simple computer and has the tasks at direct digital control (DDC) level. Mainly the process control system has three distinct control systems; Process Controller or Computer (Supervisory Computer), Digital Control System which may be a microcomputer and Analog Control System. Refer to Figure 2.6. The supervisory computer sends start-up commands to the programmable controller and provides set-point and digital control signal to the analog control system. Since the programmable controller consists of the process shut-down logic and some control logic, it can check the sent commands, for example a command to switch a machine on or off, against its logic which has the status of the process elements. So there is a double check of the commands before any action is taken in the process area. The analog control system, in effect, is a self contained system and can monitor all process variables. In the event of a computer failure, the control of the process is achieved under the operator commands sent from the Manual Interface Panel (semi-automatic control). In fact, a computer failure is indicated by analog values going into alarm and a shutdown of the process computer. [18].

Some vendors introduced another method of backup systems to handle computer failure modes. For this backup system, firstly, manual control is to be acceptable and each output channel must have individual D/A converters. In addition, these output channels can be manipulated by an external source. Under these conditions, the discrete signals inhibit the processor from changing the output. On referring to Figure 2.5, it can be seen that two indicators are used to indicate the input and the output. For a given input the operator can change the output value, raise or lower, manually by using mode switching. In the figure there are four independent channels. Each channel has its own output card that consists of an D/A converter and hold station, so called. [2].

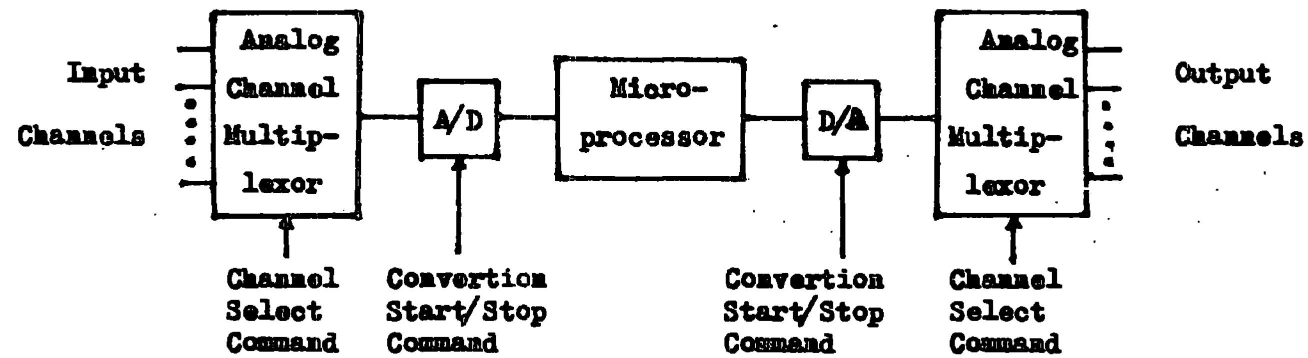


Figure 2.1 'Multiplexing Inputs and Outputs' system

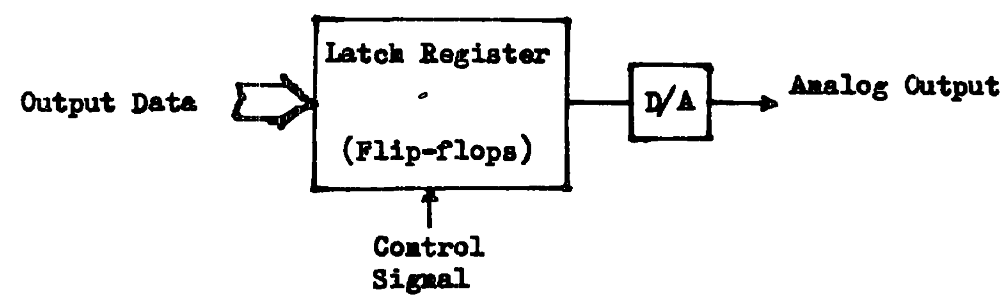


Figure 2.2 'Latching Parallel Data' system

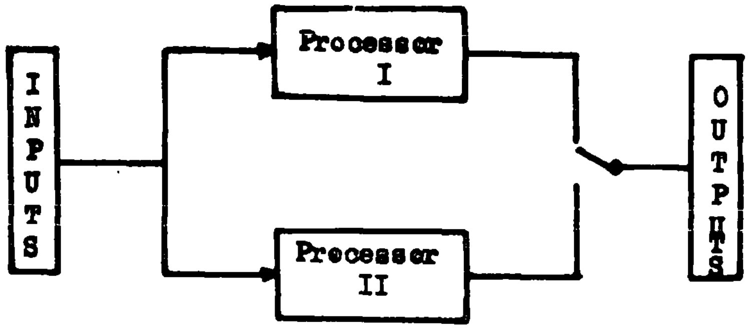


Figure 2.3 'Multiple processors' method

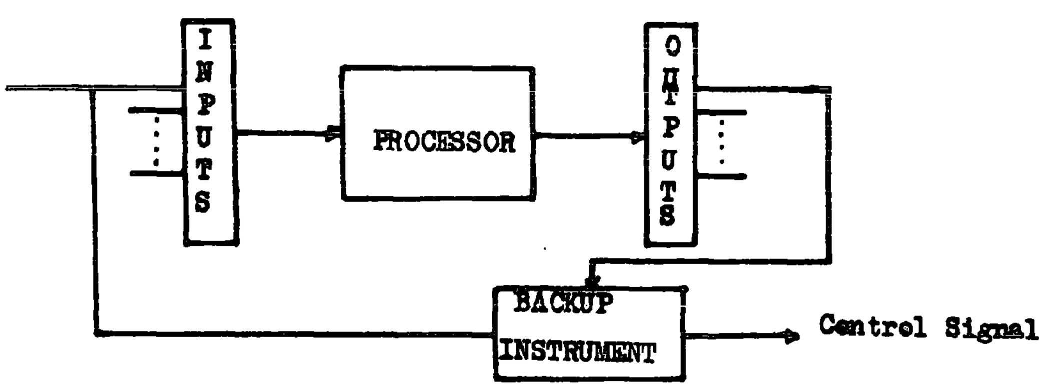


Figure 2.4 Manually backed-up system

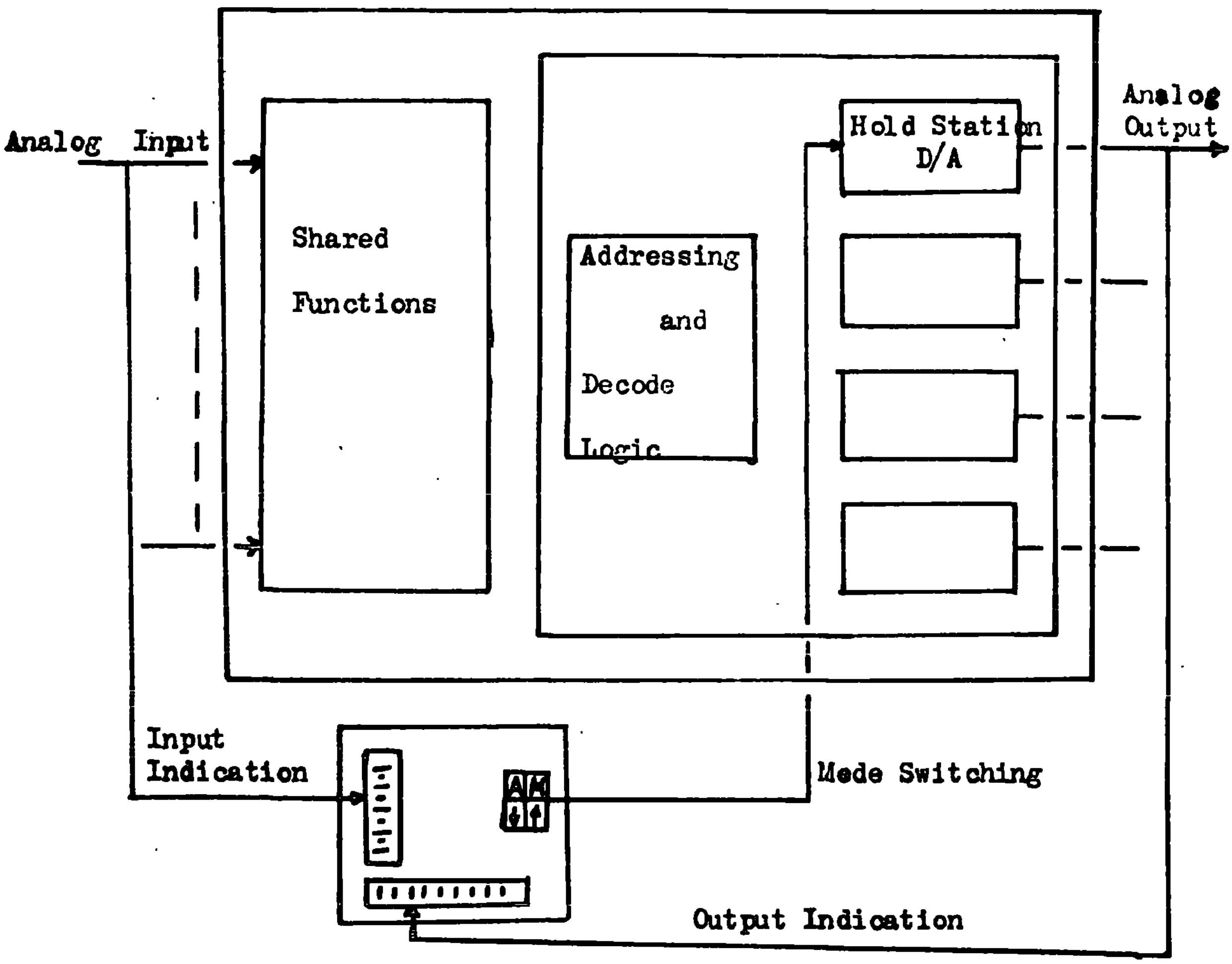


Figure 2.5 Display back-up system

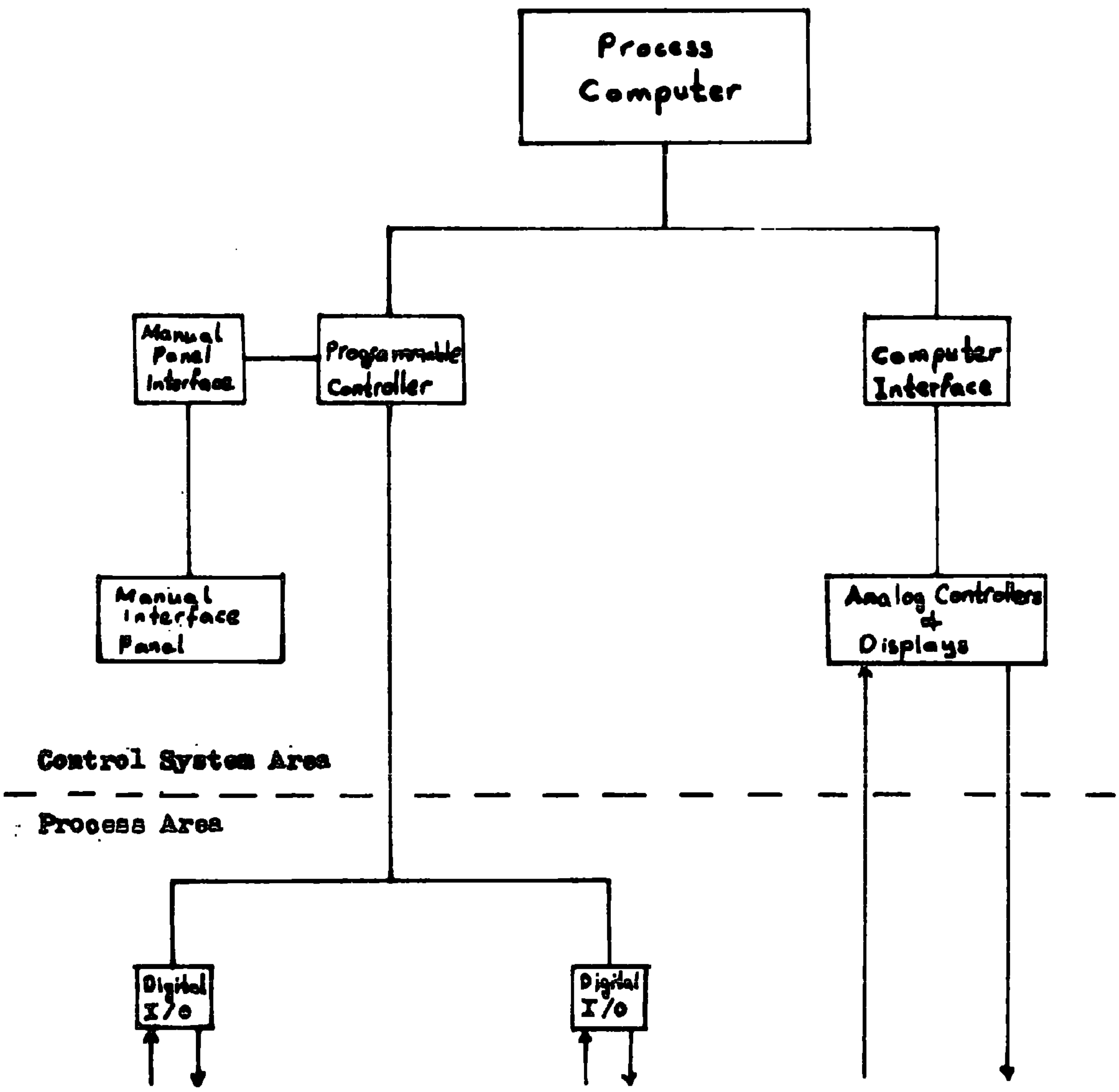


Figure 2.6 An on-line computer control system with two computers connected in series.

CHAPTER III

SWTPC 6800 COMPUTER SYSTEM

3.1 General View of the System

Since this system runs on 16 bit address buses, one MPU can address up to 64 K; in other words memory capacity is 64 K. Information flow between the systems elements and also between the system and the outside world is done on 8-bit bi-directional data buses. The word length is 8 bit. [24,14].

It has 72 basic instructions. It is loaded with the machine codes of its assembler instructions. To obtain the machine codes of a program written in assembler code there is a need of a host computer that has the compiler to translate the instructions into machine codes. There are eight addressing modes in the M6800; Dual Addressing, Accumulator Addressing, Inherent Addressing, Immediate Addressing, Relative Addressing, Indexed Addressing, Direct Addressing and Extended Addressing.

A SWTPC 6800 computer system consists of a Microprocessing Unit (MPU), a Read Only Memory (ROM), some Random Access Memory (RAM) s, some Peripheral Interface Adapter (PIA) s and some Asynchronous Communications Interface Adapter (ACIA) s. PIA provides parallel interfacing and ACIA gives serial interfacing. But, in fact, the MIKBUG hardware program in 1 K MC6830 ROM enables a MC6820 PIA package to be used as a serial interfacing adapter. For our project, a MC6820 PIA is used to make serial TTY interfacing to the MPU.

Each of the M6800 family elements in this computer system operates on a single five-volts power supply.

A clock is essential for the processor and the interface circuits. The system may be expanded up to eight interfaces and the various baud rate (one baud is so many bits' flow per second) of each interface circuit can be selected by a bit rate generator such as MC14411.

Figure 3.1 can give some idea of the connections of the system's elements in blocks. Figure 3.1 also consists of a crystal controlled oscillator used as a clock source. The control bus in the figure is shown bi-directional. In fact, some control lines are only inputs to the MPU and some of them are only outputs from the MPU. In addition, all control lines connected to memories have the signal flows from the MPU to the memories only.

3.2 MC6800 Microprocessing Unit (MPU)

This is a Large Scale Integration (LSI) device and consists of two 8-bit accumulators, one 16-bit Index Register (X), one 16-bit Program Counter Register (PC), one 16-bit Stack Pointer Register (SP) and one 8-bit Condition Code Register (CC). The number of the program instructions used to activate the logic and arithmetic functions of the MPU is 72. It contains instruction decoding logic, Arithmetic and Logic Unit (A.L.U.), and program sequence control. It is provided with an 8-bit bi-directional data bus, a 16-bit address bus and some control lines; Read/Write (R/W), Valid Memory Address (VMA), Data Bus Enable (DBE), Interrupt Request (IRQ), Restart (RES), Non-Maskable Interrupt (NMI), Go/Halt (G/H), Bus Available (BA) and Three State Control (TSC). It is also provided with a two-phase clock whose operating rate is up to 1 MHz. The MPU uses the clock as a timing reference to execute instructions. For example, the MPU places an address on the address bus during one phase of the clock and the data bus will be active during the other phase of the clock. Refer to Figure 3.2.

The MPU transfers information between the memory units and the outside world so that if it is required the MPU will fetch an item of data from a memory address and then store it to the desired output. The sequential fetching of instructions in the program memories is done through the PC. After loading a program and giving the starting address to the PC register, the Go (G) command is applied to start the execution of the program. The processor then loads the address in the PC on to the address bus and the Read pulse (high state of the R/W signal) strobes the data at the given address into the MPU. The instruction decoding logic of the MPU will enable the MPU to interpret the strobed data as an instruction or just a number. The execution of instructions is done in the A.L.U. After fetching a data the value of the PC will automatically increase by one. The new value in the PC is the next address for the MPU to get the next data to execute. However, if the program requires the result of an execution to be stored in a memory location, then the MPU places the address, where the resulting data is to be stored, on the address bus. The output signal of an AND whose inputs are the VMA and $\phi 2$ clock signals will inform the external devices of the MPU that there is a valid address on the address bus. In fact, this enable signal is applied to the

Enable lines (E) of the memories and I/O devices. The DBE which is normally the ϕ2 clock signal puts the data on the data bus from the MPU.

The IRQ line of the MPU gets signal from PIA/ACIA. An interrupt signal will be sent to this line if there is an available data on a peripheral while a program is running. Next, the present contents of the PC, X, accumulators and CC register will be stored in seven bytes of RAM starting with the memory address that the Stack Pointer (SP) contains, and proceeding in descending order of memory addresses downwards in sequential manner, if the interrupt mask bit in the processor condition code register is not set. Then the MPU sets the interrupt mask bit to ensure that it will not response to any further interrupt before the completion of the serving to the present one. The MPU starts running a program that serves the IRQ. The starting address of the subroutine is loaded into the PC from two memory locations. The instruction RTI (Return from Interrupt) causes the MPU status loaded with their contents before the interrupt occured. Upon completion of the service program the execution of the suspended program is resumed. In fact, this way of data transfer from the outside world may save processing time if it is compared with the way of data transferring where a program would periodically examine every single port in turn for an available data.

The RES signal recovers the MPU from a power failure and it may also be used for an initial start-up of the processor. To be able to communicate with the MPU through a teletype while there is a program continually this signal is to be applied. That results in stopping the program being executed. The RES signal is also connected to the RES line of PIAs. This signal has the effect of setting all PIAs' registers to logical zero.

The Go signal (high state of the G/H signal) starts the execution of program at the address in the PC. If this signal is at low state program execution will be halted.

In the high state the BA signal indicates that the MPU has stopped execution and the address bus is available.

The NMI signal has no effect of the interrupt mask bit in the CC register. At the presence of this signal the MPU status are stored away through the SP. The interrupt mask bit is set. The MPU branches to a routine that serves this interrupt. The address of the routine is stored in two memory locations and the CC is loaded with this address automatically. Upon completion of the routine, the previous program is resumed by the MPU.

3.3 MC6830 Read Only Memory (ROM)

This LSI device has static operation. It is a 1024 x Byte = 1K byte package. It has the MIKBUG hardware program in it and MIKBUG is unalterable. The MIKBUG monitor program does not only provide the user with the subroutines to use in his programs but also enables him to examine a memory location and change it at will (except the memory locations of ROMs), and to start execution of his program, and to load an object tape, and to print a block of memory via a keyboard.

MIKBUG also uses one Random Access Memory package as a temporary data storage.

3.4 MC6810 Random Access Memory (RAM)

This is used to store software programs. It is an alterable Read/Write memory. Each RAM is organised as 128 Bytes.

3.5 MC6820 Peripheral Interface Adapter (PIA)

The MC6820 Peripheral Interface Adapter is a parallel type interface adapter circuit. Its MPU side has 18 lines provided. (Refer to Figure 3.3). Eight of these lines form the bi-directional data bus, and the others are used for addressing and controlling the PIA and the internal registers of the PIA. The peripheral side has 20 lines and 16 of them form two 8 bit bi-directional data buses. Four out of these 20 lines are utilised as control lines.

The address of PIA is defined by hardware selection logic. The Chip Select (CS0, CS1, CS2) lines on the MPU side are used to select a PIA. The PIA is programmable. Each PIA has six 8-bit registers in two sets called A side registers and B side registers. Each side has one Data Direction Register (DDR), one Output Register and one Control Register. Which set will be utilised as input or output depends on the bit positions of the DDR.

The MPU treats the DDRs and the Output Registers on each side as a single memory location and the MPU treats one PIA as four memory locations.

In conjunction with the Register Select (RS0 and RS1) lines one bit of the control register directs the MPU to the DDR or Output Register.

On the MPU side of the PIA there are two IRQ lines; one of them is used for the interrupt request of one side and the other one for the interrupt request of the other side.

Two control lines CA1 and CB1 on the peripheral side are only input. The CA2 and CB2 may, however, be programmed to act as the peripheral outputs or the interrupt inputs.

Since the PIA is a parallel I/O device, to connect an analog signal to a PIA an Analog to Digital (A/D) converter circuit is to be placed between the signal and the PIA's data lines. The digital outputs of an 8 bit A/D converter are connected with these 8 bit data lines. Since a convenient bit pattern of the CRA (CRB) may set the interrupt flag of the same control register during a high to low transition on the CA1 (CB1), the CA1 (CB1) control line of the PIA may be connected to the status signal (the signal that informs that the A/D conversion has been completed) of the A/D converter circuit. Since setting one of the control register bits may make CA2 (CB2) go high, the CA2 (CB2) may be connected to the "Start Conversion" line of the A/D in order to make the A/D circuit start the conversion. Firstly a software program loads the control register with a bit position set so that the CA2 (CB2) goes high and consequently the conversion starts. Upon completion of the conversion the status signal of the A/D goes low, so does the CA1 (CB1) and the program may read the data into the MPU after the recognition of the IRQ flag that was set by high to low transition on the CA1 (CB1).

3.6 MC6850 Asynchronous Communications Interface Adapter (ACIA)

This M6800 family element enables the user to make serial data communications with the MPU. The user's TTY control terminal, Keyboard and Cassette Recorder may be interfaced to the MPU through ACIAs. The Motorola's MIKBUG hardware program in 1 K

MC6830 ROM enables PIA to be utilised for the same purpose. Since we are using a PIA in the project to make serial communications, the references [24], [14] and [15] should be referred to obtain the detailed information about ACIA.

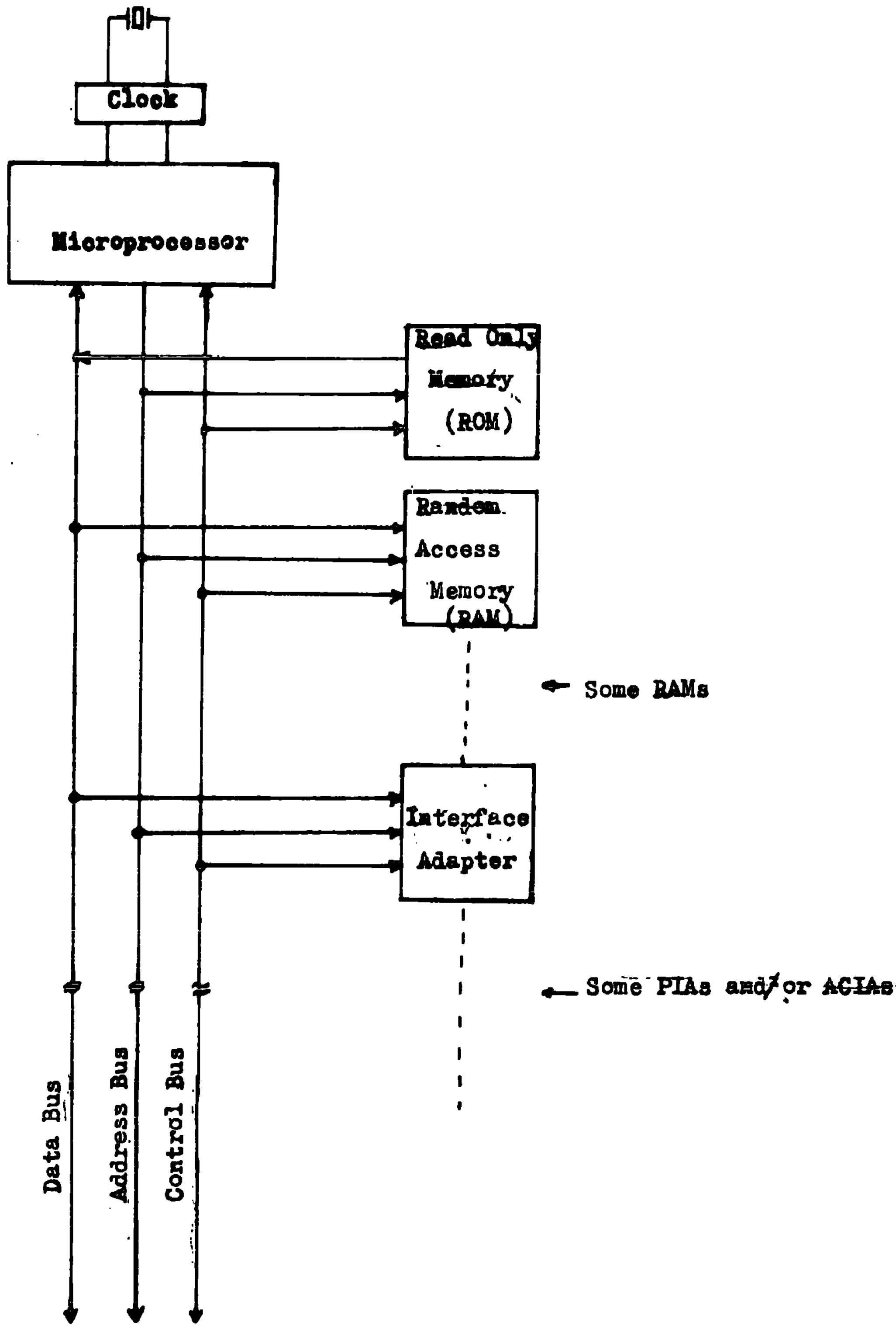


Figure 3.1 M6800 Microcomputer family block diagram

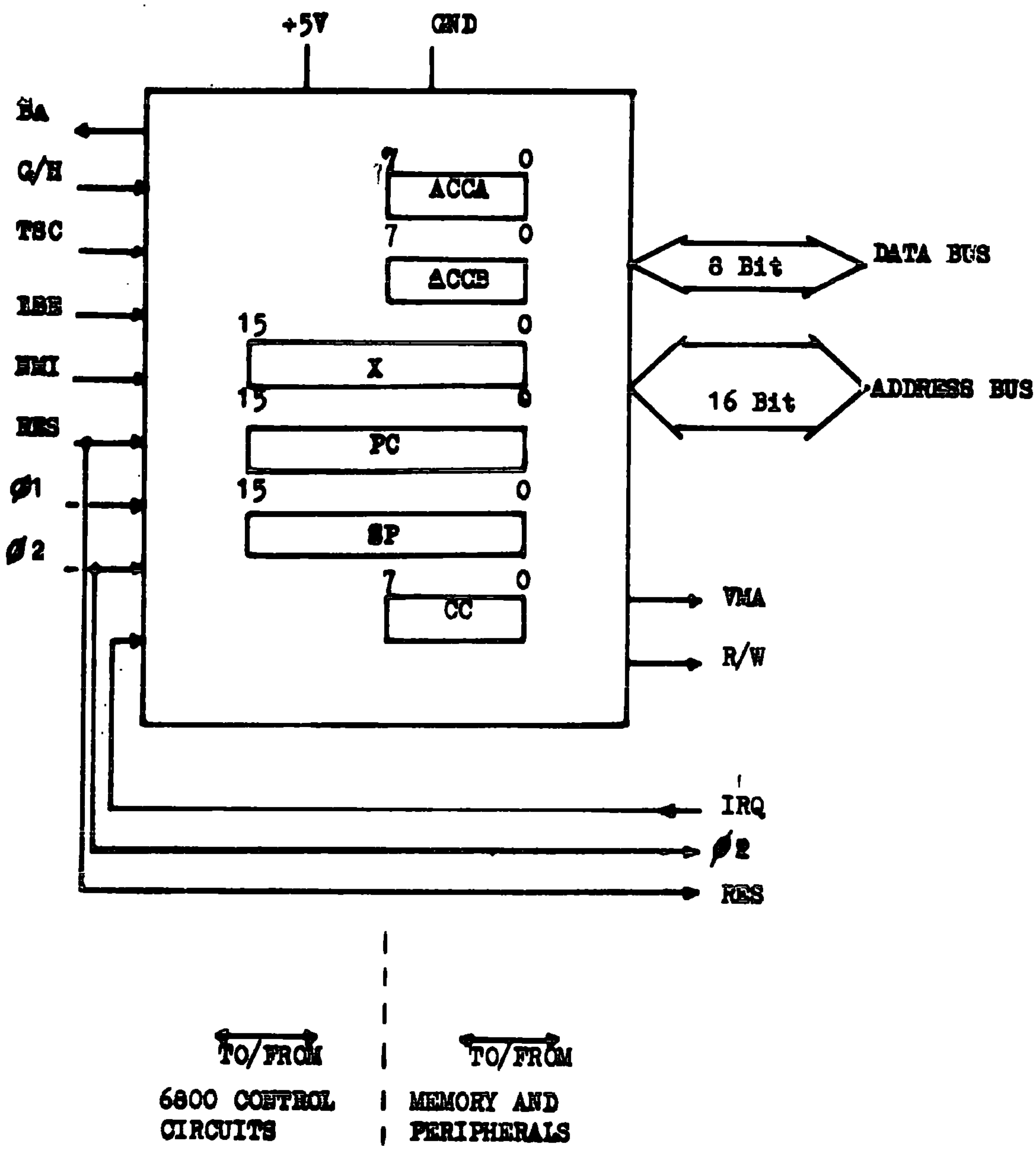


Figure 3.2 MC6800 Microprocessing Unit

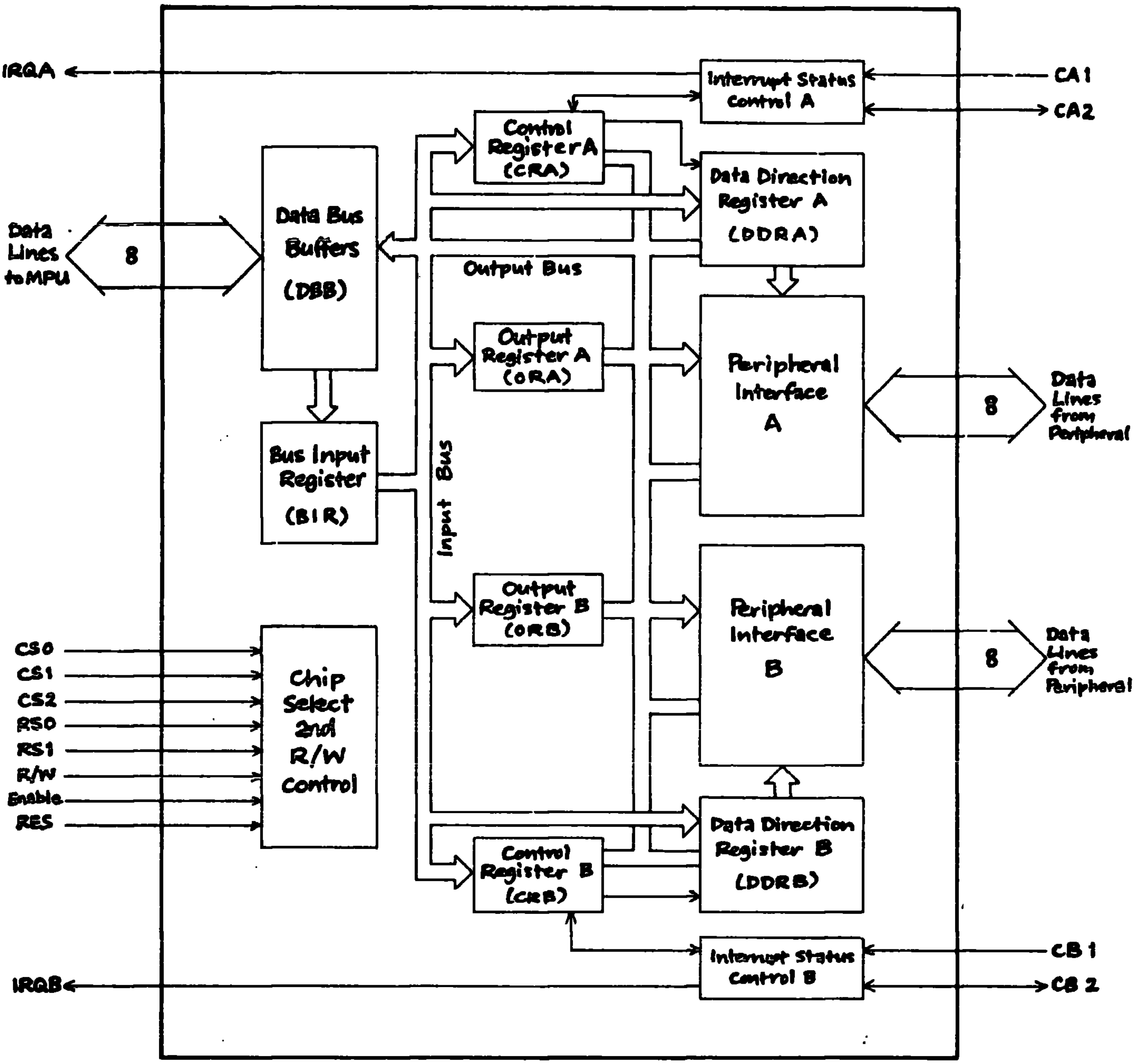


Figure 3.3 The expanded block diagram of MC 6820 Peripheral Interface Adapter

CHAPTER IV

EXISTING GOVERNOR SYSTEM AND ITS ANALOG COMPUTER MODELLING

A valve controlling system block diagram supplied with the step-by-step on-load valve testing procedure was obtained from a manufacturer. This chapter mentions the brief description of this electro-hydraulic valve control system operation and the valve testing steps. The reduced governor model with some changes in the block gains and time constants, and the mathematical model of this system established on the analog computer are also given in this chapter. The block diagram calculations for the analog computer simulation, however, are in Appendix A.

4.1 The Governor Specifications

The block diagram of the given Electro-Hydraulic Governor system is in Figure 4.1, and the gains and the time constants given by the manufacturer are listed in Table 1.

The Power Piston response in the opening direction and the Actuator response in the closing direction for a 6 volt step input also given by the manufacturer is illustrated in Figure 4.2.

This Governor achieves the control task for both Control and Intercept valves.

GAINs		UNITY	TIME CONSTANTs in seconds	
G1	5.38	milli-amps/volt	T1	.04
G2	2.666	cubic inches flow / milli-amp	T2	.013
G3	1.25	1/Area squ.inches	T3	.16
G4	20.	10Volt/Stroke inches	T4	.0028
G5	1.		T5	.1
G6	.0755	1/Area squ.inches	T6	.1
G7	1.666	10Volt/Stroke inches	T7	.1
			T8	1.

Table 1.

GAINs		UNITY	TIME CONSTANTs in seconds	
G2	2.66	cubic inches flow / milli-amp	T5	.1
G3	1.25	1/area squ.inches	T6	.1
G'4	2.	10Volt/Stroke inches	T7	0.0
G5	1.		T'8	.01
G6	.0755	1/Area squ.inches		
G'7	55.	10Volt/Stroke inches		

Table 2.

4.2 Brief Description of The System Operation

The Phase Rectifier with the gain G4 is a Linear Variable Differential Transformer and the position transducer with the gain G7 is a rotary type transducer. Since the Primary Actuator and the Power Piston are pure integration devices, to obtain a constant output position of these devices for a constant input signal and to enhance their performance, the feedback paths are established around these devices.

The Servo-Valve Amplifier is an electronic device and amplifies the error signal. The current supplied by the amplifier operates the Servo-Valve. The oil flow from the Servo-Valve to the Primary Actuator results in the piston movement of the Actuator. Since the Pilot Valve piston is mechanically in connection with the Primary Actuator piston, the Pilot Valve operates and a high pressure oil flows into the Power Piston. The valve operation is achieved over a lever that links the valve to the Power Piston.

The gains of the Servo-Valve, Primary Actuator, the rectifier gain G4, Pilot Valve and Power Piston may vary between contracts where different pressure requirements exist. The gains G1, G7 and also G6 in Figure 4.1 vary between Governor and Intercept valves.

4.3 The Actual On-Load Valve Test Sequencing

4.3.1 Purpose of the Test

The on-load valve testing is achieved by doing the following in sequence: selecting the valve to be tested, adding a negative going ramp signal to the operating Governor input signal and consequently closing the valve, tripping the valve, applying a positive going ramp signal to the Governor input, re-application of the negative going ramp, releasing trip and bringing the valve back to its pre-test position.

During the test a permanent record of the valve performance is provided by an X-Y plotter. The X input of the plotter receives the valve position output and the Y input is in connection with the Governor input. The graph plotted during the test is the input versus output curve. The curve characterises the behaviour of the Governor by indicating if the trip operation has failed and if the valve being tested gives unexpected response because of the erosion on the valve. Getting no vertical trace is the result of the fact that there is failure of the trip operation, because the plotter's X input is expected to be zero and the Y movement still exists during the ramp operation following the tripping of the valve. Any friction or faulty response of the valve will cause a kink in the traces being obtained during the valve closure and re-opening.

The newly recorded response for each valve is checked against its previous test record and the differences are observed to gain ideas on the present conditions of the trip operation system and the valve itself. Then the necessary precautions may be taken to avoid risk.

4.3.2 Basic Specifications of the Used Pushbuttons

The on-load tests on the valves are carried out remotely from a control room.

There are a certain number of pushbuttons on the control panel dedicated to the same number of the valves on the turbine. These buttons are called "Select Pushbuttons". One valve is tested and one plotter record for this valve is obtained at a time. When pressed the button initiates the test procedure on the appropriate valve. The Select buttons are latch-down type of buttons.

Also there are two non-latching pushbuttons that have to be pressed and held down in sequence during the tests. These buttons are common to all valves. One of these non-latching ones called "Test" initiates a command to interrupt the present Governor signal when pressed and causes the restoration of the Governor signal when released by means of the ramp signal. The other button called "Trip" is utilised to trip the valve under control.

The Select and Trip buttons also operate a set of switches so that only one valve can be selected at a time and the Trip button cannot be depressed before the Test button.

4.3.3 Detailed Test Procedure

The on-load valve test is done step by step as follows:

Select Valve

The valve subject to the test is selected by pressing the button dedicated to that particular valve. The signal obtained by pressing the button results in switching the X input of a plotter from its normal valve position input to another X input which in turn gives the X movement during the encircling of the valve number on the plotter. The Y input of the plotter is also supplied by the same module after the SELECT command has been reached. After the valve identification, the relay is released and the X input on this relay is switched to the valve position input. Another relay called "Fault Reset Relay" is also energised and this provides a signal to the Governor control panel to indicate "on-load test" in progress.

Press Test Pushbutton

When this test button is pressed down a ramp signal is added to the present Governor input signal and the valve closes.

During the valve closure, the system automatically checks the signal selection (voting) circuit in the Servo-Valve Amplifier.

Press Trip Test Pushbutton

The Trip Test Relay is energised, this results in tripping the valve. To avoid accidental rapid valve closure that would result by depressing the TRIP button before the TEST button, here some switches are used as interlocks.

Release Test Pushbutton

The close Governor signal is cancelled. The Primary Actuator operates but no valve movement occurs. A limit switch closes to guard against accidental rapid operation of the valve.

Press Test Pushbutton

The close Governor signal is applied and the Actuator closes down. The valve still stays at its close position.

Release Trip Test Pushbutton

The Trip signal is cancelled.

Release Test Pushbutton

The close Governor signal is interrupted and the valve re-opens.

Release Selector Pushbutton

The fault reset relay is de-energised and the "on-load test" lamp on the Governor control panel is extinguished.

4.3 Simulation of the Actual Governor

For simplicity, the given Governor model was reduced to the system whose block diagram is illustrated in Figure 4.3. The gains and time constants of the blocks used in this reduced system have been listed in Table 2. The block diagram calculation of the system for the analog computer simulation is given in Appendix A.

When the calculated blocks were connected in their order the mathematical model for the analog computer became as illustrated in Figure 4.4.

For better observation of the system response the amplitude of the step input was taken to be as big as -26.25 volts. The negative signal is used to obtain the positive outputs of the amplifiers 2A2 and B4. The Power Piston response in the opening direction and the Actuator response in the closing direction for -26.25 volts have been plotted and seen to be as in Figure 4.5. As it can be seen in the figure, the calibration has been done and the Position Output has been multiplied by ten. To compare the actual system responses of the Actuator and the Power Piston in Figure 4.2 with their model responses, a further scaling has been done by switching the capacitor value of the amplifier B4 from 0.1 µF to 0.01 µF and feeding the output signal of the pot P7 to the summing amplifier 2C5 over 1MΩ but not 0.1 MΩ. The Actuator and the Power Piston responses obtained after the scaling are illustrated in Figure 4.6. The actual system responses can be easily compared with the model responses in Figure 4.7. This figure is made up of Figure 4.2 and Figure 4.6.

The model response is ten times slower than that of the actual Governor.

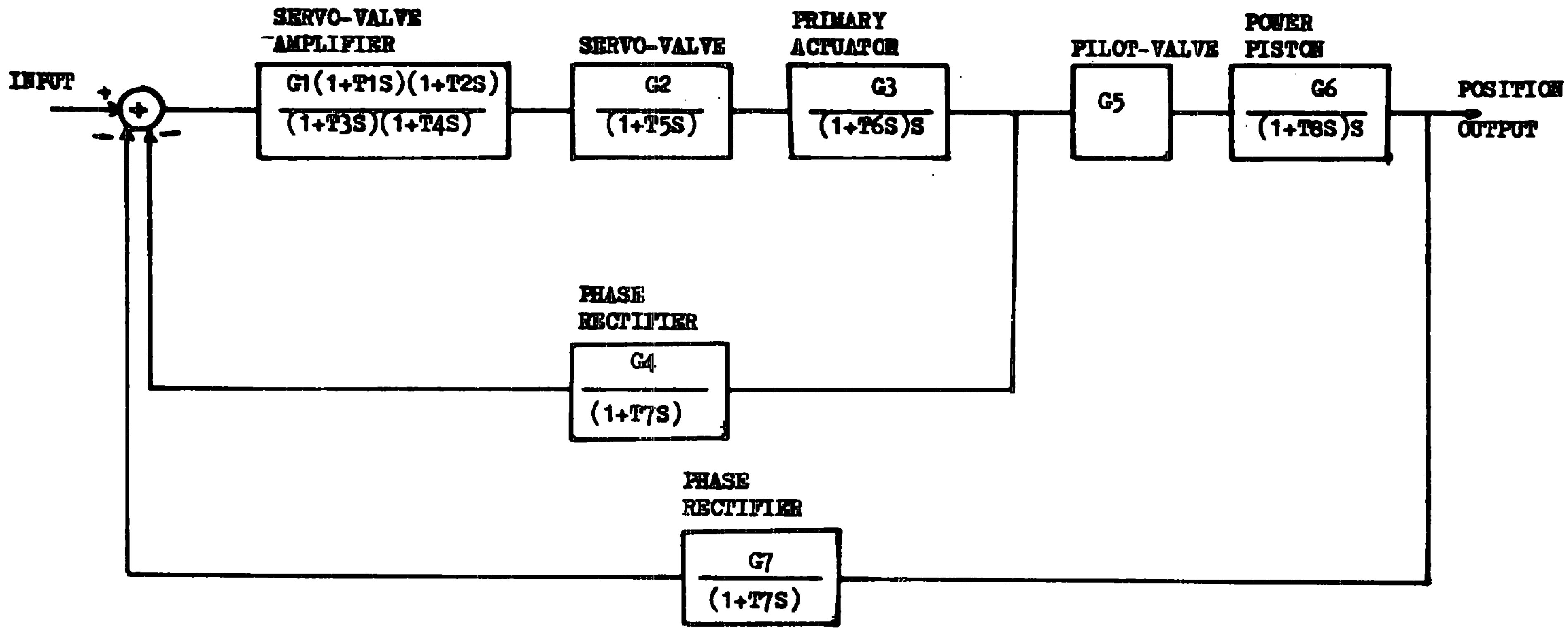


Figure 4.1. Basic block diagram of the present Governor.

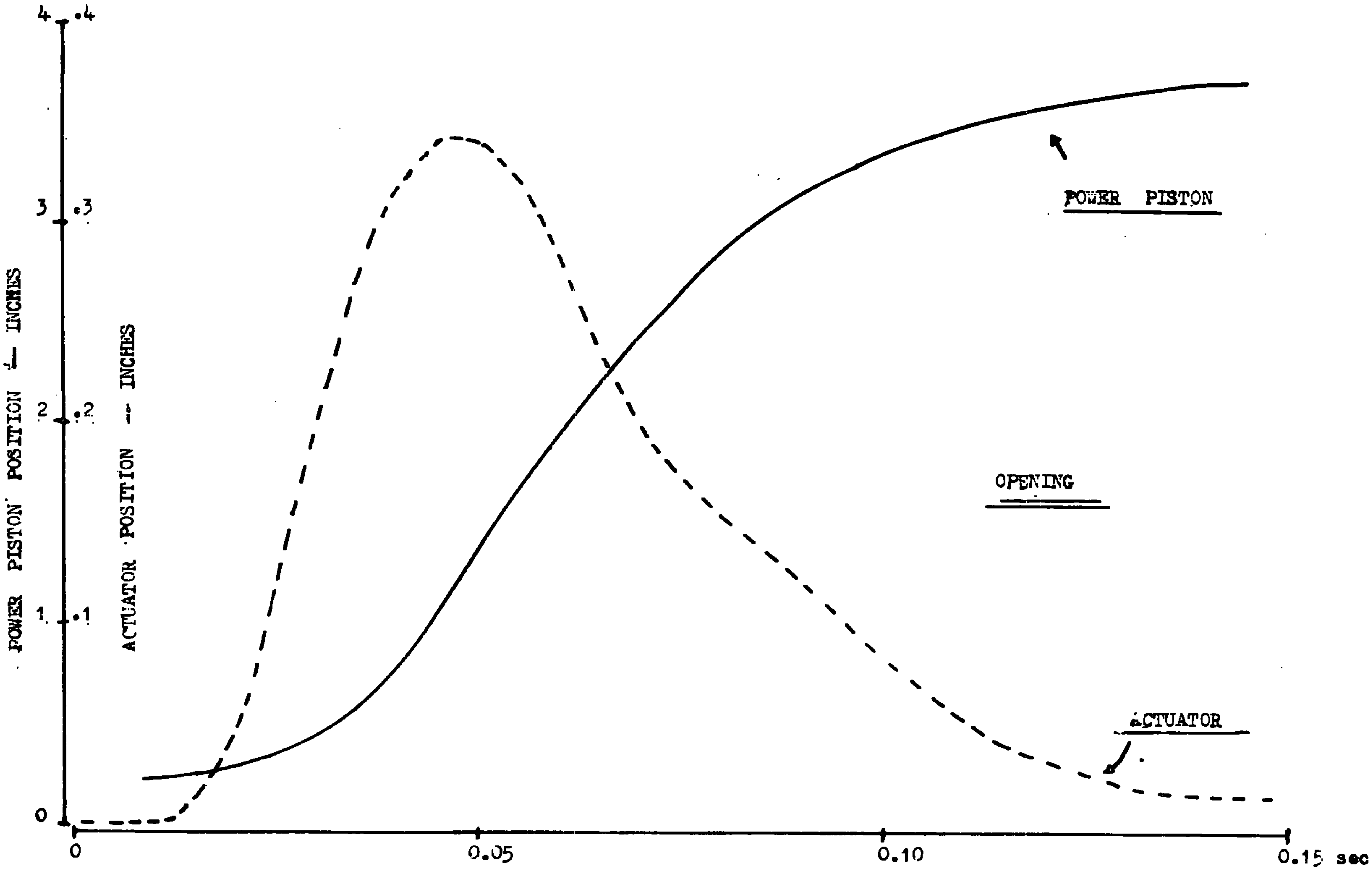


Figure 4.2 Power Piston response in opening direction and Actuator response in closing direction to 6 Volt step input.

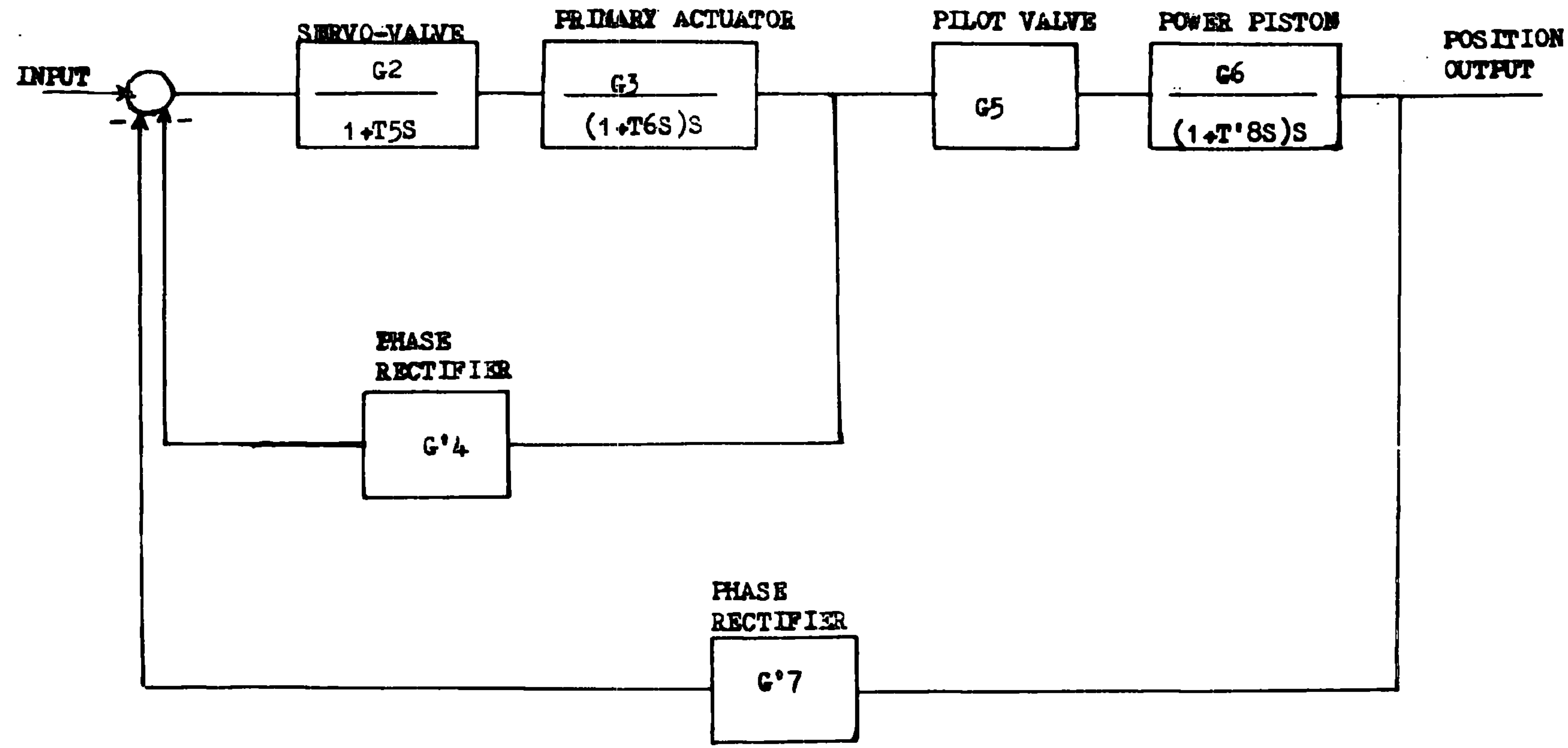


Figure 4.3 Reduced governor model

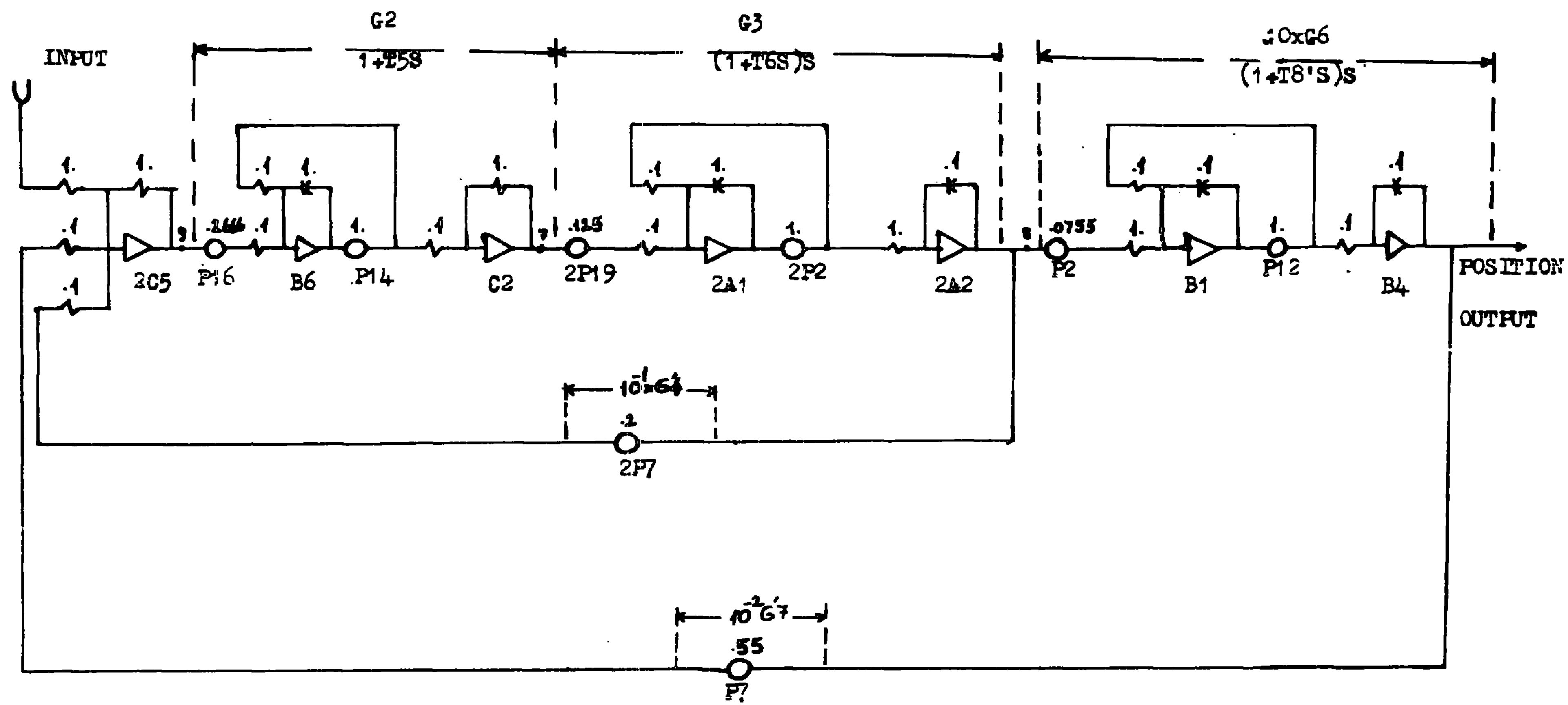


Figure 4.4 The mathematical model established on the analog computer

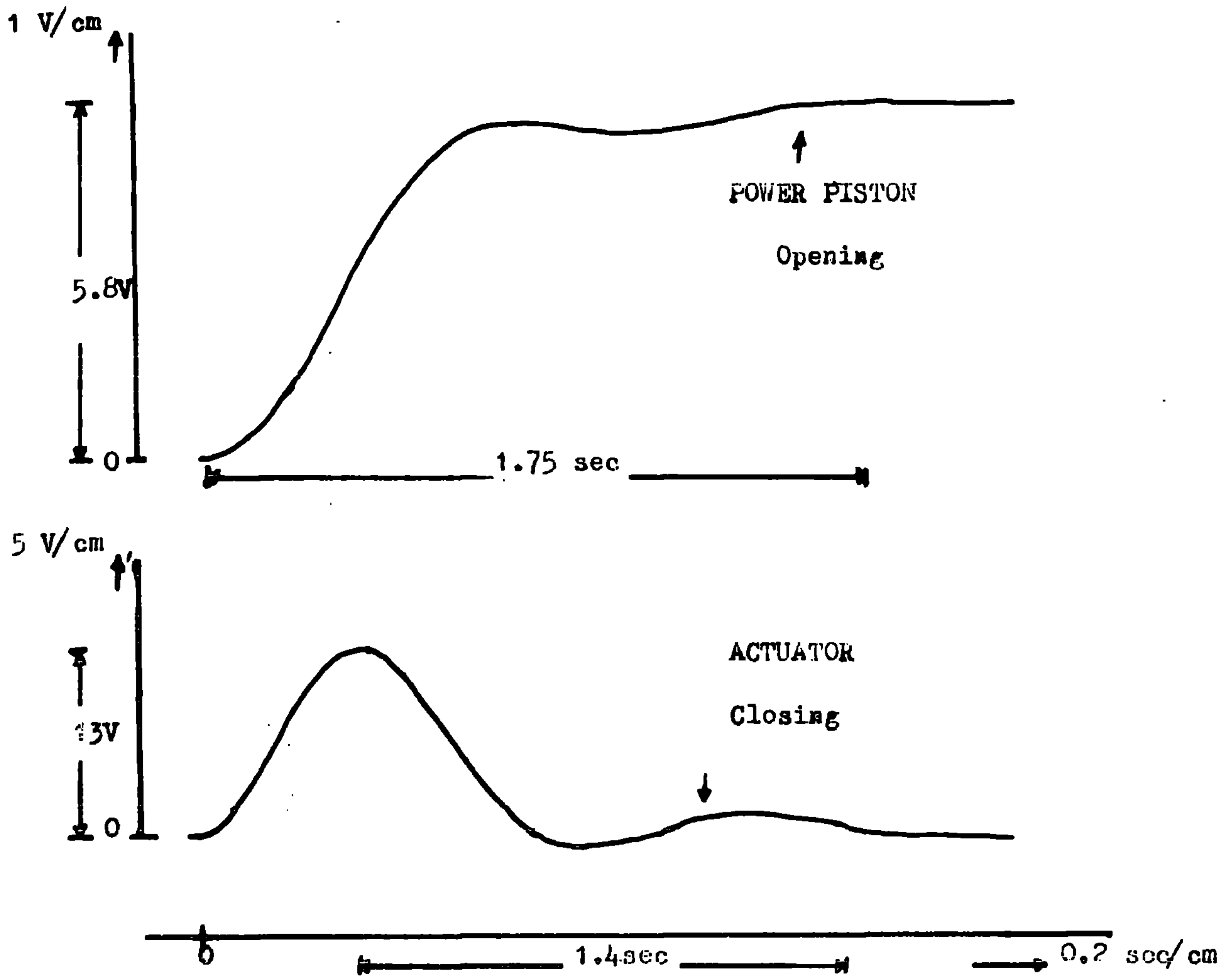


Figure 4.5 Power Piston response in opening direction and Actuator response in closing direction to -26.25 volt step input

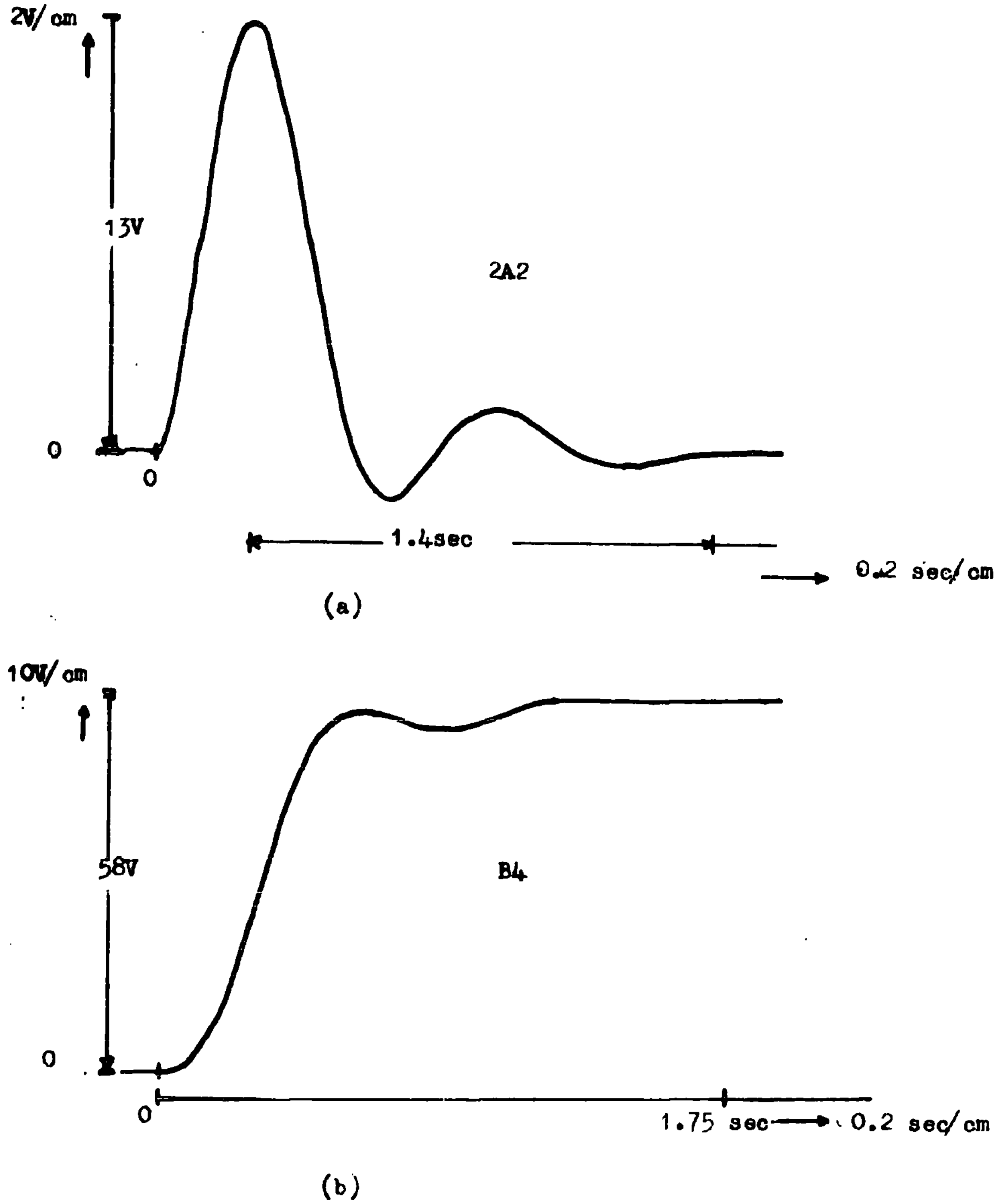


Figure 4.6 Response of the mathematical governer model to

-26.25^{V} step input

(a) Scaled response of the Actuator model in closing direction.

(b) Scaled response of the Power Piston model in opening direction.

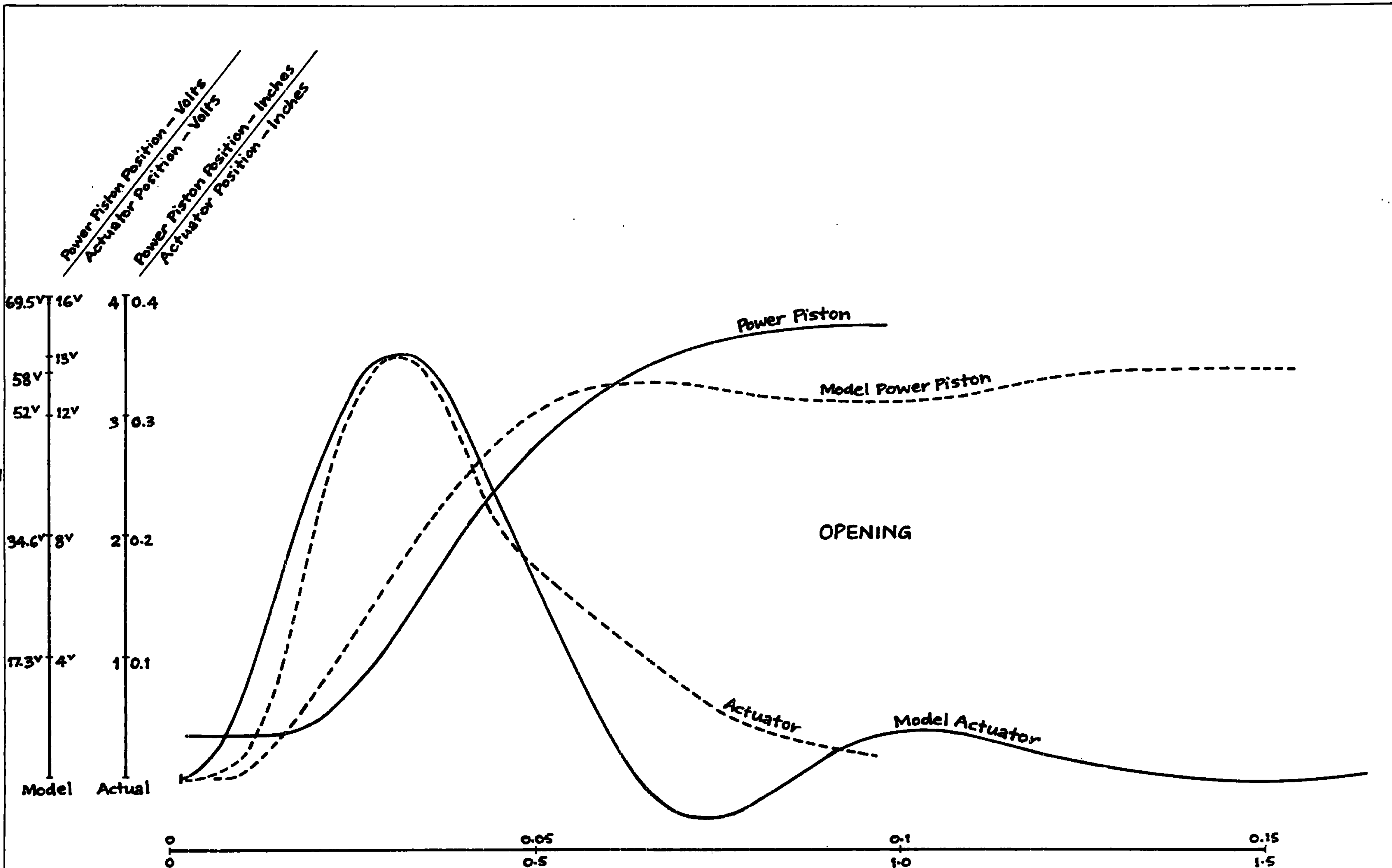


Figure 4.7 Comparable responses of the actual system and its reduced mathematical model

CHAPTER V

MICROPROCESSOR-BASED ON-LOAD VALVE SEQUENCING

Microprocessor-based On-Load Valve Sequencing is developed in this chapter. The chapter describes both the manner in which the MC6800 microprocessor is interfaced to the external systems and the manner in which the microprocessor program is developed. The arrangement of the plotter paper used to obtain a permanent record of valve performance and the time calculation of the ramp signal created via the system software are given in this chapter. [24,14,15].

5.1 System Hardware Organisation

The experimental system contains the MC6800 microprocessor, 4096 words of the 128 x 8 RAM (McM6810)s, a 1024 x 8 ROM (MC6830) and three MC6820 PIAs.

The present peripherals include: one A/D and two D/A converters to produce digital input to and analog outputs from the computer, a Teletype for operator communication, a Connection Box to obtain flexibility in connections during the system setup, the Control Panel to place the necessary equipment (pushbuttons, amplifiers and relays), an X-Y plotter and the Analog Computer. A fairly basic block diagram is shown in Figure 5.1.

There are two PIAs used for this project apart from the PIA-3 used for interfacing the TTY.

Because it is possible to arrange the control lines CA1, CB1, CA2 and CB2 of a PIA on the peripheral side as inputs or outputs and because the MPU may control the conditions of these lines on request, the "status" line of the A/D converter was connected to the control line CB3 and the "Convert Command" line was connected to the control line CB4. The necessary software was prepared to control and complete the conversion. The second side of the PIA-1 has been established as the output side and the 8-parallel data lines on the peripheral side of the PIA-1 were connected to the inputs of the converter D/A-1. The control lines CA1 and CA2 have been established by the software as input and output, respectively.

One side of the second PIA is also connected to another D/A converter named D/A-2. The control lines of this side CA3 and CA4 were established again by the software as input and output, respectively. Only one control line CB2 of the second side of the PIA-2 was established as input.

The commands SELECT, TEST, TRIP and PEN (on - off) were connected to CA1, CB2, CA3 and CA4, respectively. CA2 was used in connection with the relay R2 and R3 in order to obtain the computer control of the relays' operations.

The A/D and D/As consist of ZN425 8-bit dual mode single chip converters. The clock frequency of the A/D is 259.9 KHz. For full-scale reading the maximum conversion time is

$$\frac{2^8}{259.9 \times 10^3} = 0.985 \text{ millisecond}$$

The full scales of the converters have been set at 3.8 volts.

5.1.1 Connection Box

From both PIA-1 and PIA-2 six out of eight control lines CA1, CA2, CB1, CB2, CA3 and CA4 have been connected to 4 mm sockets on the box. Two D/A output lines and one A/D input have also been connected to three 4 mm sockets. In addition, the grounds of the converters have been placed on this box to obtain a reference for the input and outputs. Figure 5.2 illustrates the top view of the connection box.

This box has been prepared for quick and easy alteration of the connections during the system setup.

5.1.2 Control Panel

A control panel with three pushbuttons was built. The buttons latch down and glow when pressed. They have to be pressed a second time to release them. Each button energises a circuit made up with two invertors connected in series and the output of each circuit is connected to a 4 mm socket. When the button is on there is 5 volts available at the associated socket. These pushbutton circuits provide adequate interfacing of the commands SELECT, TEST and TRIP to the PIAs' control bits. In addition, the TRIP button has been directly connected to a double pole form C D.I.L. Reed Relay (R1) to operate two relays in one package, simultaneously. Figure 5.3.

A Zener Diode Circuit with the gain 0.5 was built to interface the analog model output with the maximum 3.1 volts to the computer. The Zener breakdown voltage is 6.2 volts. Refer to Figure 5.4.

To operate two relays R2 and R3 by means of one control line (CA2) of the PIA, the circuit in Figure 5.5 was established. The point 5 is a 4 mm socket.

The box also houses three non-inverting OP Amps used to secure no-current-flow to the plotter and analog computer. One of the purposes of the OP Amps is to effect the calibrations. Refer to Figure 5.6.

5.1.3 Hardware System Setup

The overall diagram is shown in Figure 5.8.

The test system was connected to the mathematical model of the Governor. Once the Analog Computer is switched to the "compute" mode, all the test sequencing is achieved from the Control Panel, TTY, and the Microcomputer.

A sheet of A4 chart paper is used to obtain a permanent record of the valve performance. How such a paper has been arranged is explained in the System Software part of the thesis.

When the software is run, the SELECT button has to be pressed to start the step-by-step testing sequence. When this button is on, the plotter identifies the required valve by underlining one of the pre-printed code numbers on the plotter paper. Then a constant input is fed to the model. The TEST button is pressed to close the valve by decreasing the Governor input to zero. Then the TRIP button is pressed to trip the Power Piston. Releasing the TEST initiates a command to increase the Governor input signal from nought to its pre-test value. Then the TEST button is pressed to decrease the Governor input to zero. At the bottom stop of the plotter pen the TRIP and TEST buttons are released in sequence. When the valve regains its pre-test position the SELECT button is released.

Driver

The Governor signal obtained from the computer is sent to the converter D/A-2 and the analog signal at the output of the D/A-2 is fed to the OP Amp-1. The gain of the amplifier is 2.26. The purpose of the amplifier is not only to prevent the converter from the possible current-sink effect of the plotter and the Analog Computer, but also to amplify the plotter's Y input signal during the valve identification procedure. The amplifier output is applied to the Y input of the plotter and to the pot 2P13 whose value is 0.306 when the relay R2 is short-circuited. This pot is on the Analog Computer. A summing amplifier (2A3) of the Analog Computer is utilised to obtain the gain of 10 and sign inversion. The sign inversion was necessary since the negative Governor input gives a positive Governor output.

The values of the OP Amp-1, the pot 2P13 and the amplifier 2A3 were arranged so that the amplitude of the Governor input would be 26.25 volts for the maximum D/A-1 output 3.8 volts. The obtainable maximum gain of the summing amplifier is not more than 10.

Sensor

The Zener Diode Circuit receives the Position Output, multiplies it by 0.5 and feeds to the OP Amp-3. The purpose of the circuit is that the diode regulates the A/D input against variations in the model output and also against variations in the A/D input current. The Zener breakdown voltage is 3.6 volts. Since the maximum value of the Position Output is less than 7.2 volts, the gain of the circuit was chosen to be 0.5. The OP Amp-3 with the gain of 1 was placed to prevent the interaction between the Zener Diode Circuit and the A/D converter. The output of the OP Amp-3 supplies the signal to the A/D and, when the relay R3 positioned to the X input of the plotter over the OP Amp-2 whose gain is 3.35. The OP Amp-2 prevents the current draw of the plotter from the converter D/A-1. The gain was chosen higher than one in order to obtain more distinct record of the valve performance and large movement on the X axis during the test sequencing.

Other Connections

The outputs of the SELECT, TEST and TRIP circuits were connected to CA1, CB2 and CA3 sockets on the Connection Box over the specially prepared leads shown in Figure 5.7. The reason why the leads were arranged was the fact that the PIA control lines CA1, CB2 and CA3 were floating when their inputs (SELECT, TEST and

TRIP) were zero. With this arrangement of connections it was ensured that when their inputs are zero these control lines are earthed and do not float.

The relays R11 and R12 in the relay package R1 were arranged as in Figure 5.8. When TRIP is on the input of the pot P2 is earthed and the input of the amplifier B4 over 0.1 MΩ is connected to its output. This R11 operation produces the same effect as the Trip Test Relay does on the actual valve. Although the pot P2 input was earthed the amplifier B4 would still integrate and produce non-zero output which is partly the result of the fact that the gain of B4 is as big as 100. For this reason the relay R12 was felt to be necessary. In fact, even if the B4 input over 0.1 MΩ was earthed when TRIP was on, the amplifier would still integrate. That was why the B4 output would be short-circuited with this B4 input when TRIP was on.

The CA4 socket on the Connection Box was connected to the PEN command input of the plotter over a piece of wire. The CA4 is set to +5 V or to 0 V through the software. When the PEN command input is supplied with +5 V the pen is on, otherwise it is released.

The socket called CA2 is connected, over a wire, to the point 5 socket on the Control Panel that positions the relays R2 and R3. The socket CA2 is energised and de-energised under the software control. This enables the computer to control the operations of the relays R2 and R3 when the socket 5 is connected to CA2. The D/A-1 output is used to give the X movement during valve identification on the plotter. This computer output is not utilised before or after the valve identification.

5.2 Software Development

The software has been prepared as program packages for ease of applications-oriented non-programmers. The system software includes 10 subroutines whose purposes in general are: creating the negative and positive going ramp signals at one of the microcomputer's output ports, reading the present input signal at the input port, storing the data used for the valve identification in the dedicated memory locations, and enabling the TTY to print the commands that inform the operator of what to do next. [15].

5.2.1 Plotter Paper Arrangement

There are 20 different numbers on the actual plotter paper sheet in two columns each consisting of 10 numbers. Each number belongs to one valve on the system.

The type of paper arranged to be used in the microprocessor-based control system is illustrated in Figure 59. The distance between two subsequent numbers on the same column has been assumed to be 1.5 cm and the distance between two numbers on the same row has been assumed to be 2 cm. Also it was assumed that the pen of the plotter draws 1 cm length of line under the number during the valve identification of the plotter. For these reasons the scaling on both axes has been taken to be 500 mV/cm.

The gain on the X-axis is 3.35 and on the Y-axis is 2.26.

To under-line one of the numbers in the first column the X-axis should start with the decimal 225 and finish with the decimal number 235 and the Y-axis should be adjusted to one of the following decimal numbers according to the valve numbers: 24, 46, 68, 90, 112, 134, 156, 178, 200, 222.

The decimal values of Y above are still valid for the following second column, but the starting and ending values on the X-axis should be decimal 245 and 255, for the numbers on the second column.

Table 3 summarises the values on both X and Y-axis required to underline each valve number. (See the example in APPENDIX C)

5.2.2 Preparation of The Ramp Signal

During on-load valve test sequencing, the actual steam valve movement from the fully open to the fully closed position takes approximately 3 seconds. Since the step input response of our Governer model is ten times slower than the actual system response, this ramp speed has also been assumed to be, 30 sec, ten times slower than the actual 3 sec duration.

Since the 8-bit converters are used in the project, it was assumed that the maximum opening of the valve will occur when the full scale reading FF of the converter in the driver circuit exists.

The hexadecimal number FF equals to 255 decimal. To enable the computer to produce a positive going ramp signal, nought to maximum, we may write a program where an accumulator is increased by one and each time the new value of the accumulator is stored at the output. Since this will be a loop repeated

Decimal Number to be put on		VALVE NUMBERS ON A PLOTTER PAPER																			
		First Column										Second Column									
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
X-axis	Starting	225										245									
	Finishing	235										255									
Y-axis		222	200	178	156	134	112	90	68	46	24	222	200	178	156	134	112	90	68	46	24

Table 3.

255 times because before the start the accumulator had been cleared, and it is required that the loop should take 30 sec, the time spent between the output two subsequent accumulator contents must be: $\frac{30\ \text{sec}}{255} \cong 0.117647058$ sec.

The first four figures after the comma may give the adequate approximation. But the value is being taken as it is to give some idea of how the command NOP (NO Operation) may be used for accuracy.

Normally the following piece of program is enough to store the content of the accumulator B after identifying the output address and clearing B:

```
POSGO   STAB   dddd
        INCB
        CMPB   mmmm
        BLS    POSGO
```

In this program dddd is the output address and mmmm is the address of the memory location whose content is hexadecimal FE (decimal 254). This program takes 5+2+4+4=15 cycles and, because the MPU used in this work executes one cycle in 2 μsec, 15x2=30 μsec

$$117.647058 \text{ msec} > 0.030 \text{msec}$$

Now a delay loop should de added. The delay loop is:

```
        LDX   kkkk
X1      DEX
        BNE   X1
```

Here kkkk is the content of the index register. LDX kkkk takes 3 machine cycles or 3x2=6 μsec of execution time.

$$30\mu\text{sec} + 6\ \mu\text{sec} = 36\mu\text{sec} = 0.036 \text{ msec}$$

Then, we find that

$$117.647058\text{msec} - 0.036\ \text{msec} = 117.611058\ \text{msec}.$$

DEX and BNE X1 together take 4+4 =8 cycles or 16 μsec = 0.016msec

The content of the index register then should be

$$\frac{117.611058\text{msec}}{0.016\ \text{msec}} \cong 7350.691125$$

The decimal 7350 equals to the hexadecimal 1CB6.

Since

$$7350 \times 0.016\ \text{msec} = 117.6\ \text{msec},$$

there is still need for a delay time of 11.058 μsec

because

$$117.611058\ \text{msec} - 117.6\ \text{msec} = 0.011058\ \text{msec} = 11.058\,\mu\text{sec}$$

Since one NOP takes 2 cycles or 4 μsec, two or three NOP should be added. Two NOPs take 2x4 μsec = 8 μsec and three NOPs take 3x4 μsec = 12 μsec.

Since 12 μsec is nearer to 11.058 μsec than 8 μsec is, it has been thought that adding 3 NOPs would be adequate.

This way of thinking made the loop POSGO to be:

```
PGOING  STAB  dddd
        LDX   1CB6
X1      DEX
        BNE   X1
        NOP
        NOP
        NOP
        INCB
        CMPB  mmmm
        BLS   PGOING
```

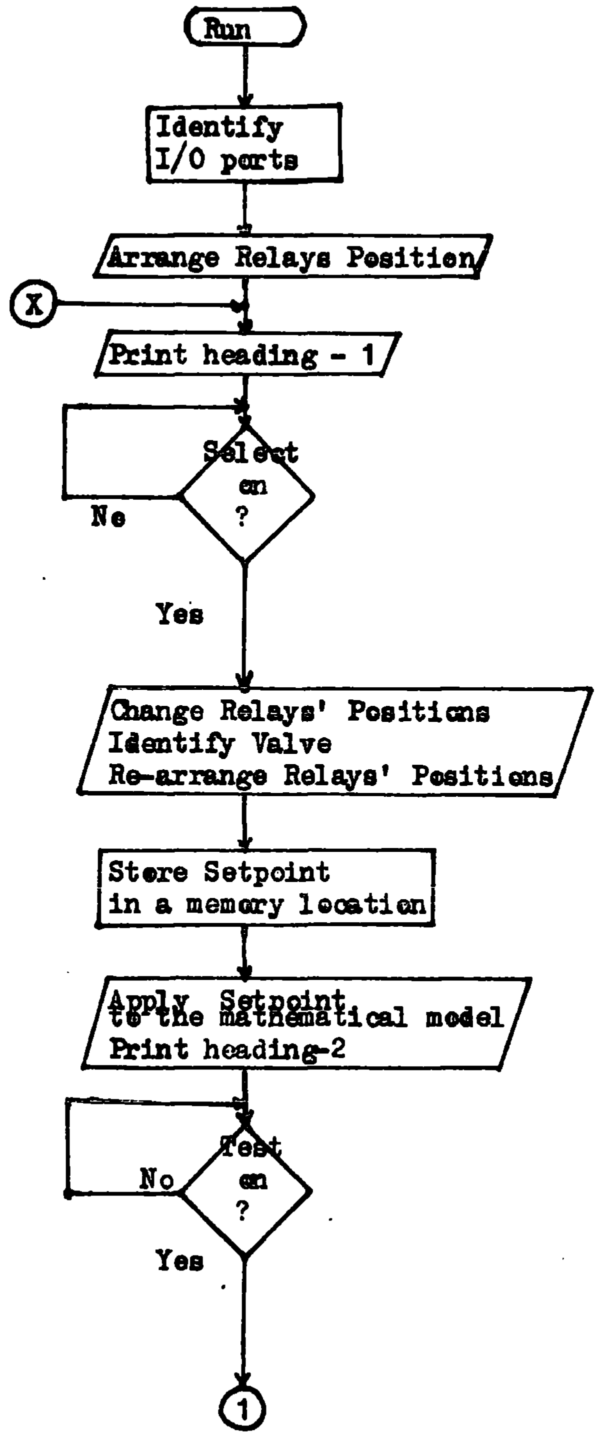
The same idea is used for the sub-program that would give the negative going ramp signal. The sub-program is:

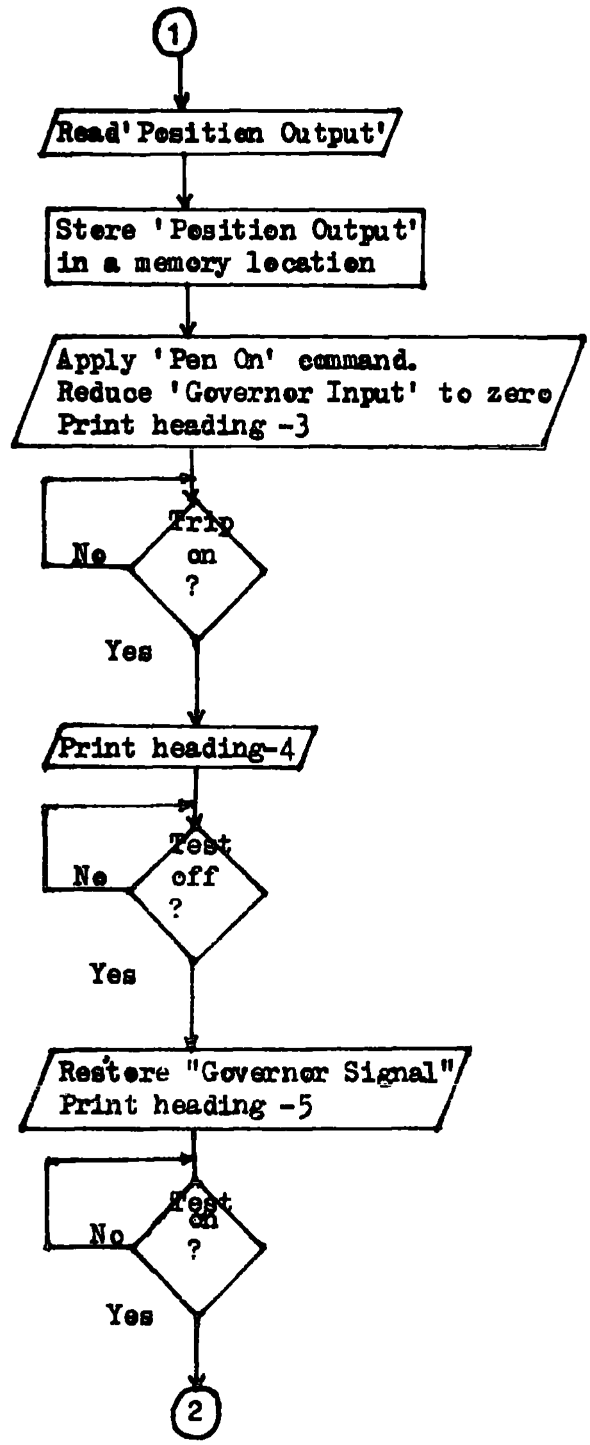
```
NGOING  STAA   dddd
L9      LDX    1CB6
        BNE    L9
        NOP
        NOP
        NOP
        NOP
        NOP


        DECA
        BNE    NGOING
```

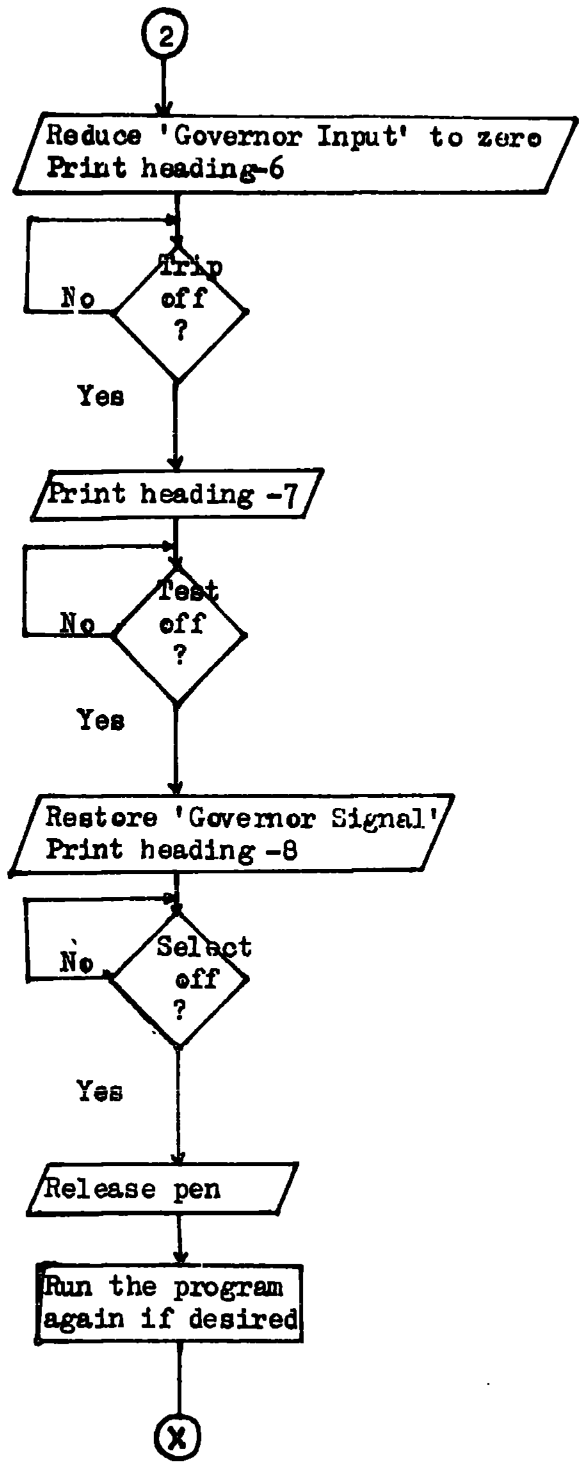
Here two more NOPs have been added to comprise the same length of time that CMPB mmmm in the previous sub-program spends.

5.2.3 Flowchart

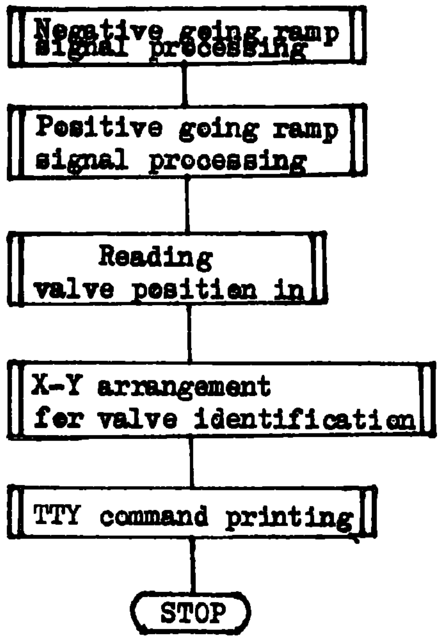
The following flowchart was prepared to organize the problem.




1
Read 'Position Output'
Store 'Position Output' in a memory location
Apply 'Pen On' command.
Reduce 'Governor Input' to zero
Print heading -3
Trip on ?
No
Yes
Print heading-4
Test off ?
No
Yes
Restore "Governor Signal"
Print heading -5
Test on ?
No
Yes
2


2
Reduce 'Governor Input' to zero
Print heading-6
Trip off ?
No
Yes
Print heading -7
Test off ?
No
Yes
Restore 'Governor Signal'
Print heading -8
Select off ?
No
Yes
Release pen
Run the program again if desired
X

The system subroutines;



5.3 On-Load Valve Testing Procedure

When the program starts running the relay R2 is short-circuited, the relay R3 has its points B and 2 short-circuited. Refer to Figure 5.8. The pen of the plotter becomes off.

The TTY prints:

(1) PRESS SELECT PUSHBUTTON

(

The MPU checks in a loop if the SELECT is on.

Select Valve

As the SELECT command is reached, the relays R2 and R3 change their positions so that the model on the analog computer gets no input signal and the X axis of the plotter gets switched over from its valve position input to the output of the converter D/A - 1. The computer generates the X and Y movements over the D/A -1 and D/A -2 respectively. The PEN ON command is also created by the software and fed to the pen circuit of the plotter. The Y input value is kept constant while the X input is being increased. After underlining the valve number on the plotter, the pen is released and it goes to its set-point since the inputs to both plotter axes are zeroed by the program. Then the relay R2 is short-circuited and the X axes is switched back to the valve position input. After this is done, a constant input is fed to the model and the Y input of the plotter over the OP Amp -1. Obviously the X input gets a position output as a response to the input signal fed to the mathematical model. The value applied as the input to the model is also stored in a memory location.

The TTY prints: 2) PRESS TEST PUSHBUTTON

(

Then the MPU starts checking in a loop if the TEST ON command has been applied.

Press Test Pushbutton

When TEST is on, the computer reads the valve position and stores it in a memory location. Then the PEN ON command is applied and the stored value on the D/A -2 is decreased to zero. This results in zero output of the valve position. The plotter draws the output of the model on the analog computer, which is the valve position, against the reducing input, continuosly. This is the line $\overline{ABO}$ shown in Figure 5.11.

After the pen has reached at its bottom stop, the TTY prints: 3) PRESS TRIP PUSHBUTTON

(

The MPU starts checking in a loop if TRIP is ON.

Press Trip Test Pushbutton

When the TRIP button is switched on the relays R11 and R12 change position so that the input of the pot P2 is earthed and the input of the amplifier B4 is switched from the output of pot P12 to its own output. (Refer to Figure 5.8). The TTY prints: 4) RELEASE TEST PUSHBUTTON

(

Then the MPU checks in a loop if the TEST button has been released.

Release Test Pushbutton

This initiates the execution of a subroutine which increases the input voltage to the model on the analog computer. This input value is increased from zero to the value existing before the first TEST ON command was applied. Since the TRIP ON command has switched off the Power Piston input, the X input of the plotter stays at zero and the Y input moves from zero to the point C. Figure 5.11 shows this by drawing a line between the points 0 and C. Then the TTY prints: 5) PRESS TEST PUSHBUTTON

(

The MPU checks in a loop if the TEST button has been pressed. The pen stays at the point C in Figure 5.11 until the TEST button is pressed down.

Press Test Pushbutton

This starts the execution of a subroutine that would give the decrement on the input value of the model until this input signal reaches at zero. During the decrement of the model input the pen moves from the point C to 0 indicating that the TRIP is still on and the Governor is getting the closure signal. When the pen reaches at its bottom step, the TTY prints:

6) RELEASE TRIP PUSHBUTTON

(

The MPU starts checking in a loop if the TRIP button has been released.

Release Trip Test Pushbutton

Releasing the TRIP button results in the position changes of both relays R11 and R12 simultaneously so that the input of the pot P2 is switched back to the output of the amplifier 2A2 and the input of the amplifier B4 is switched back to the output of the pot P12. (Refer to Figure 5.8). Then the TTY prints:

7) RELEASE TEST PUSHBUTTON

(

The MPU checks in a loop if the TEST button has been released.

Release Test Pushbutton

The Governor gets its operating value, before the test sequencing had been started by pressing the SELECT button back. The pen moves from its bottom-stop to the point A over the line $\overline{ODA}$ in Figure 5.8. The the TTY prints:

8) RELEASE SELECT PUSHBUTTON

(

The MPU checks in a loop if the SELECT button has been released.

Release Selector Pushbutton

This initiates a command to release the pen.

5.4 Test Results

For a typical test, the program was run after having adjusted the specified setpoint of the plotter. At the end of the test, the teletype commands shown in Figure 5.10 and the plotter responses illustrated in Figure 5.11 were obtained.

As it is seen in Figure 5.11, there is a phase difference between the valve closure line $\overline{ABO}$ and the valve opening line $\overline{ODA}$ which occurs because of the overall time constant in the analog Governor. If the delay did not occur during the closing and re-opening of the valve, the pen movement between A and O would be a straight line. Added to this there is a further delay which results from the processing time of the computer and the D/A - 2's conversion time. Since storing the contents of an accumulator in extended addressing mode takes five machine cycles and one machine cycle takes 2 μsec of processing time, loading data on a D/A converter takes 10 μsec. The conversion time of the D/A for full scale reading is 1 millisecond.

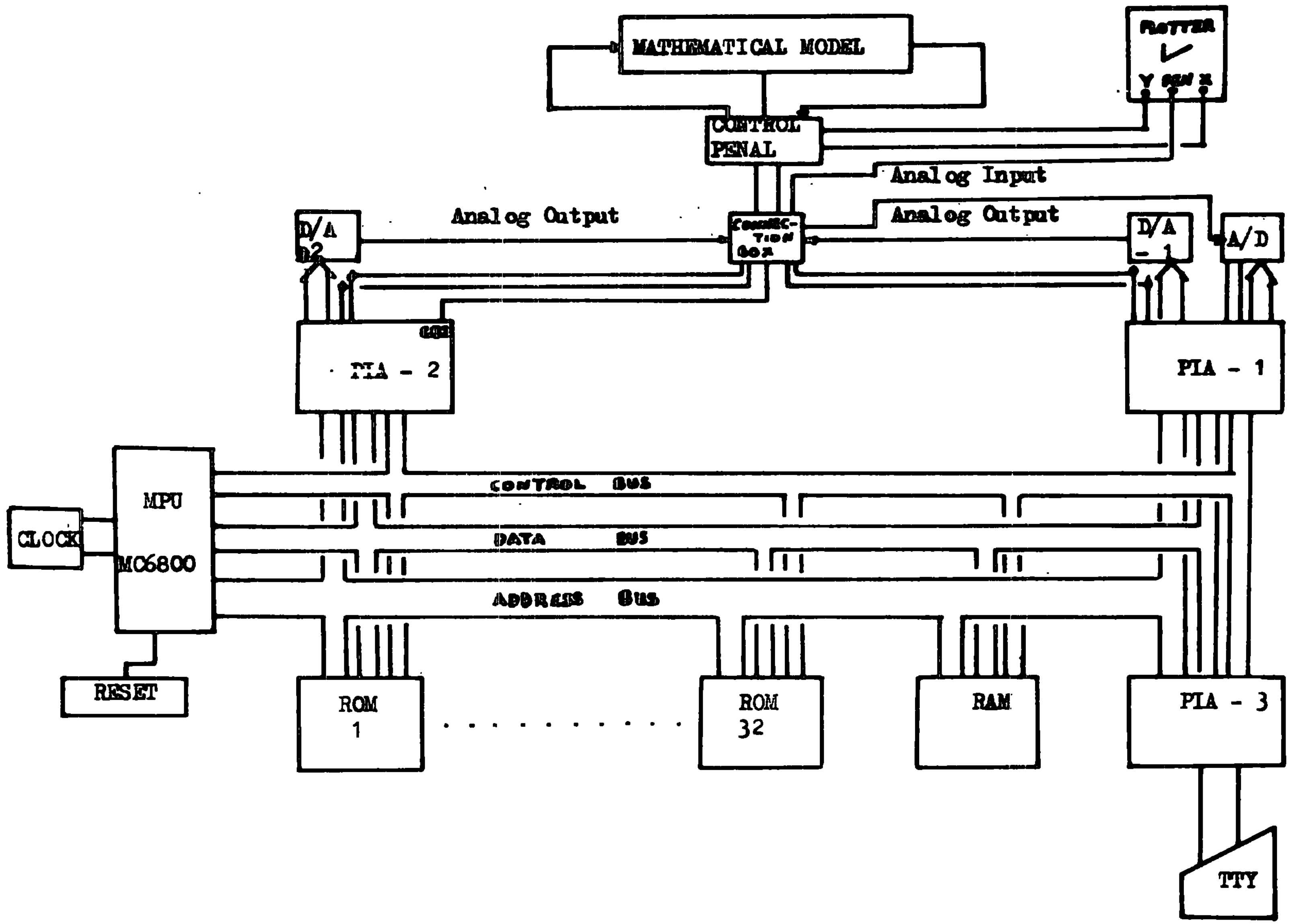


Figure 5.1 Overall system block diagram

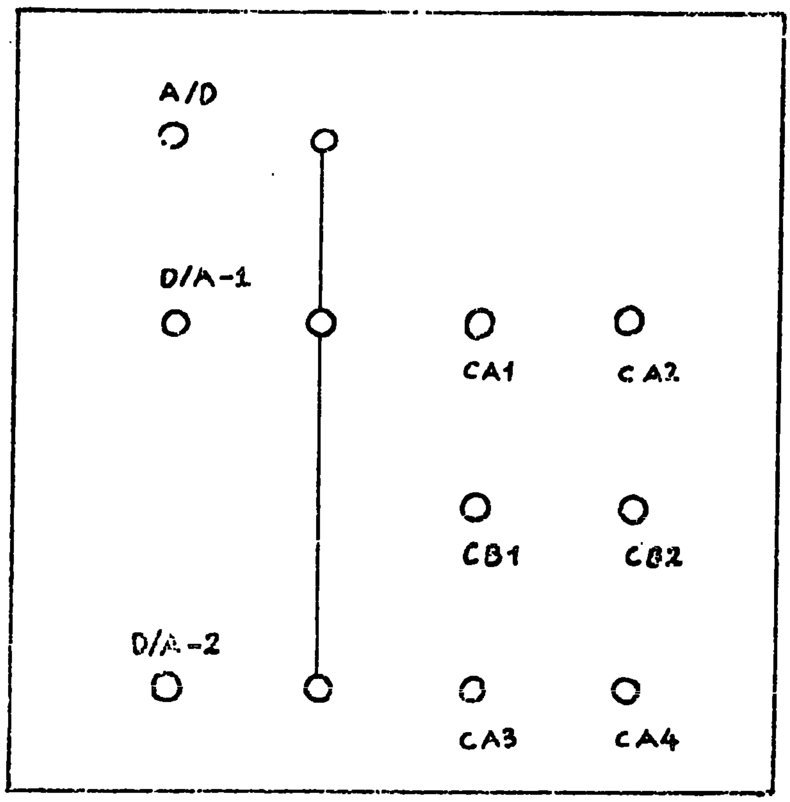


Figure 5.2 The top view of the connection box.

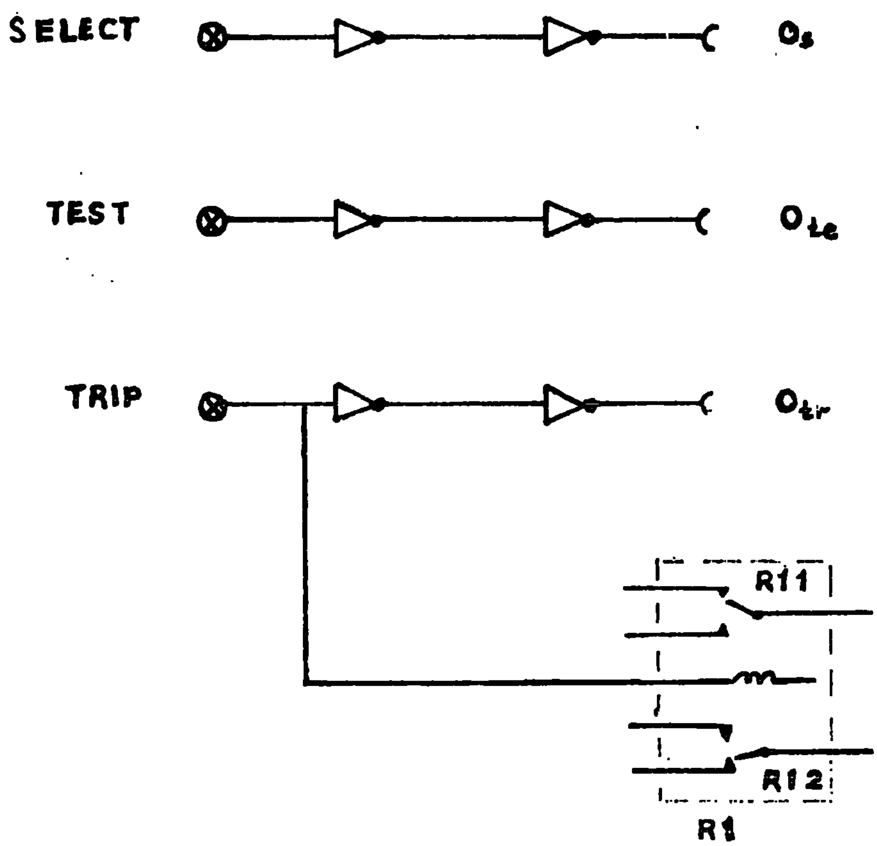


Figure 5.3 Pushbutten cennections.

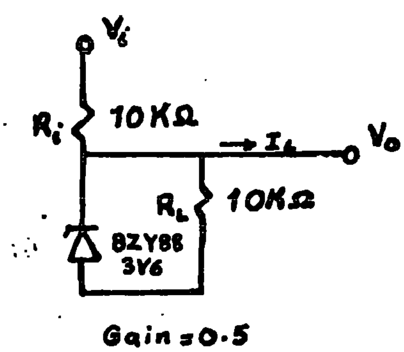


Figure 5.4 Zener diode circuit

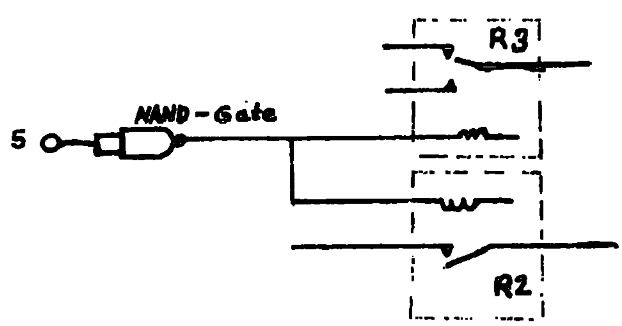


Figure 5.5 The operating input cennections of the relays R2 and R3.

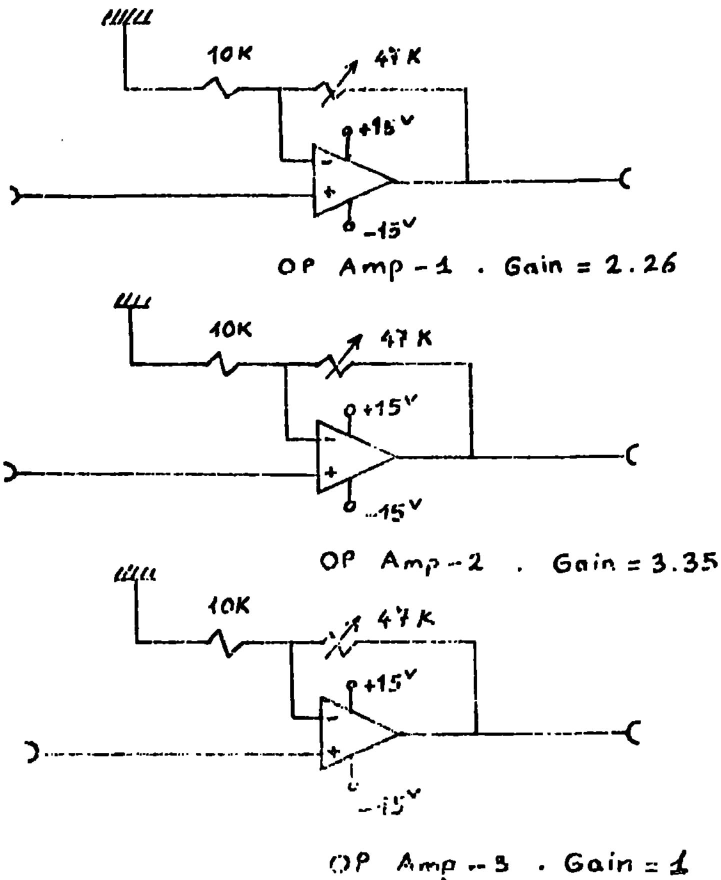


Figure 5.6 Existing OP Amp connections in the control panel.

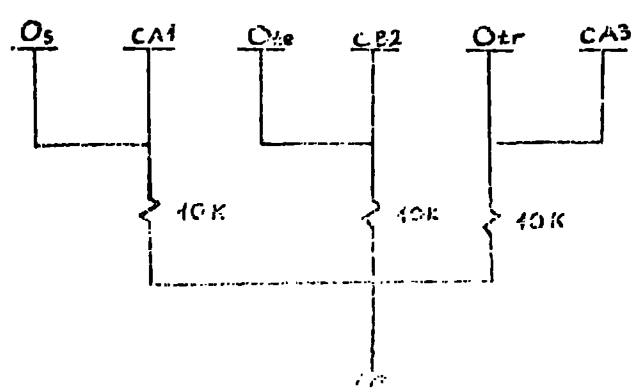


Figure 5.7 Special purpose leads

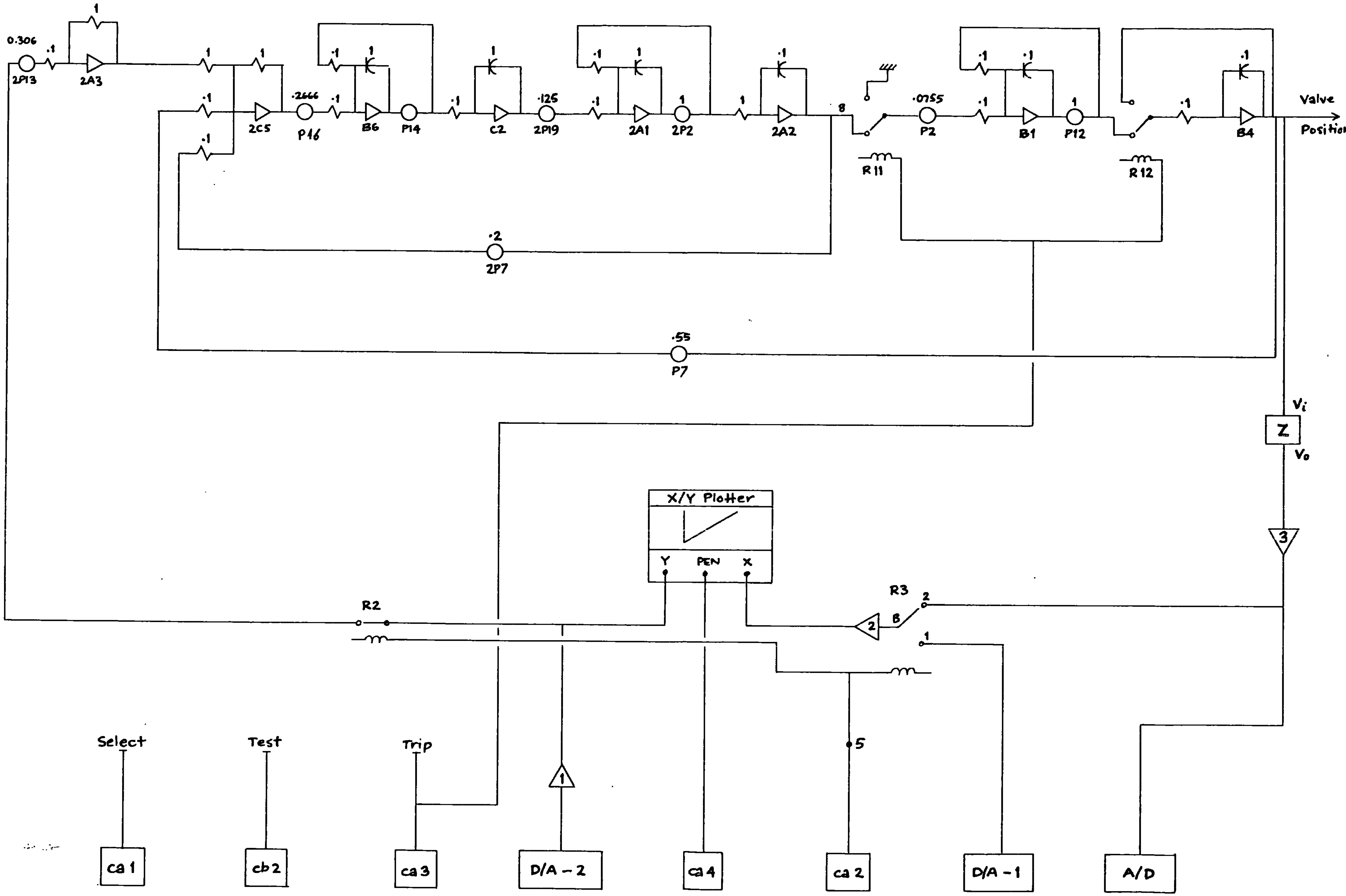


Figure 5.8 Valve testing system diagram

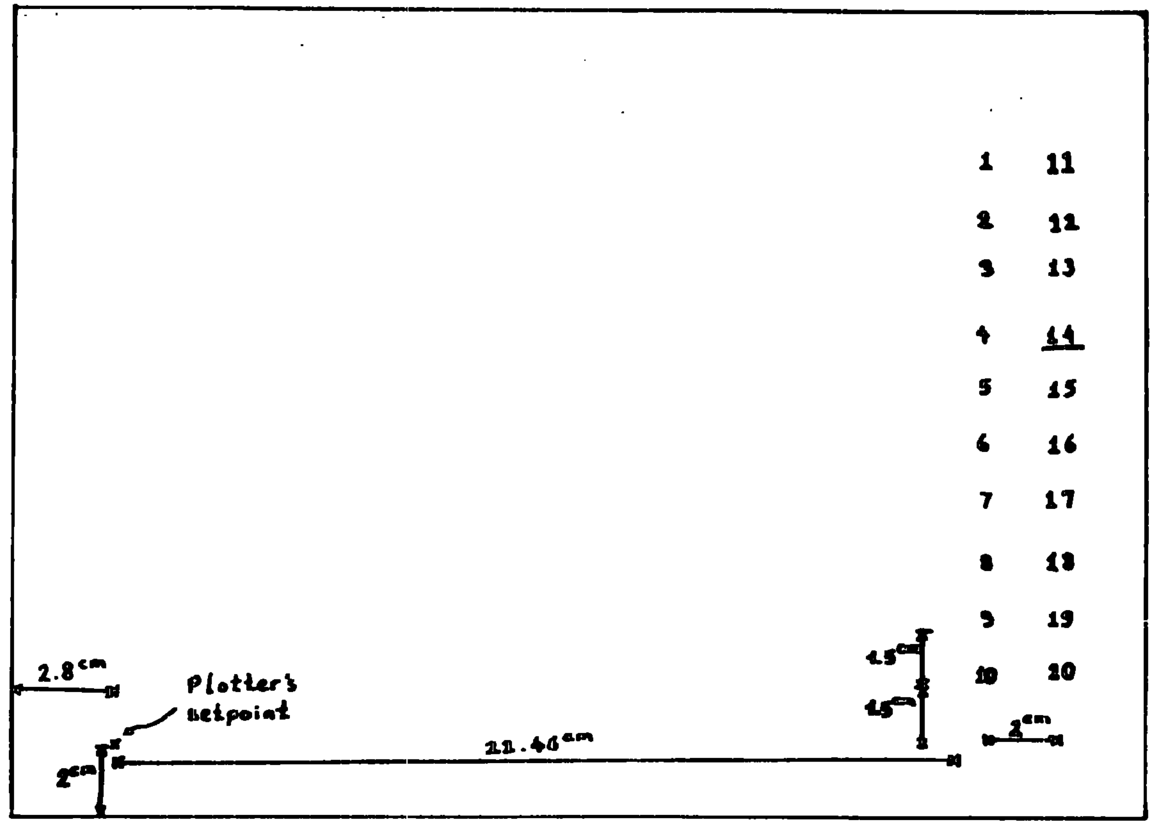


Figure 5.9 Dimentions on A4 plotter paper used for microprocessor-based on-load valve testing.

(1) PRESS SELECT PUSHBUTTON

(2) PRESS TEST PUSHBUTTON

(3) PRESS TRIP PUSHBUTTON

(4) RELEASE TEST PUSHBUTTON

(5) PRESS TEST PUSHBUTTON

(6) RELEASE TRIP PUSHBUTTON

(7) RELEASE TEST PUSHBUTTON

(8) RELEASE SELECT PUSHBUTTON

(

Figure 5.10 Teletype commands of the microprocessor -based on-load valve testing obtained at the end of the test procedure.

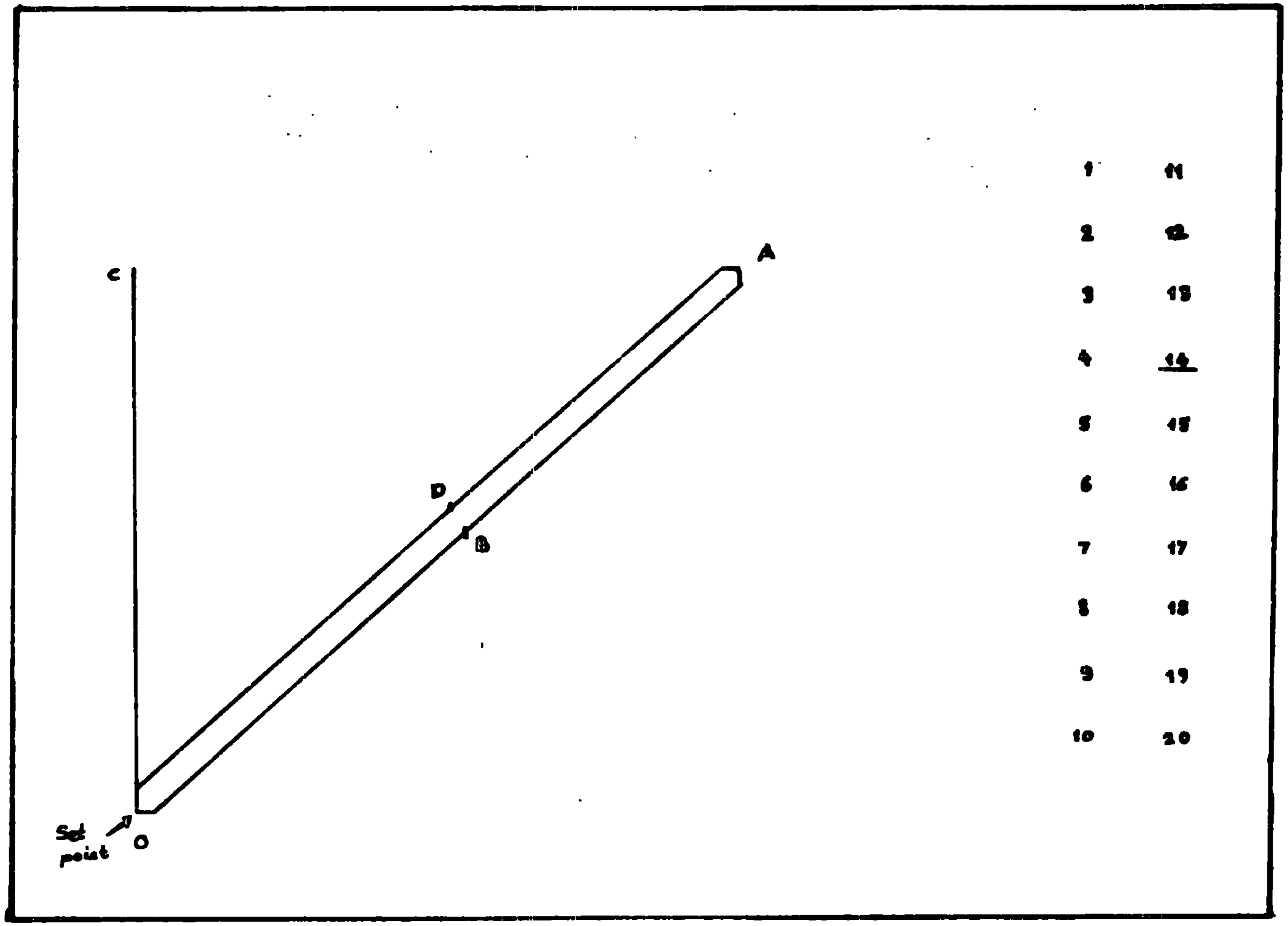


Figure 5.11 The plotter record of the microprocessor-based on Lead Valve testing system obtained at the end of the test procedure.

CHAPTER VI

MICROCOMPUTER IMPLEMENTATION OF THE EXISTING TESTING SYSTEM

This chapter deals with the differences between the existing analog on-load valve sequencing and its microcomputer implementation. It also discusses the different ways of how the implementation can be adapted on the actual system. Figure 5.8 should be referred to along with this chapter.

Some discussions about the system software have been found to be necessary and they are placed in this chapter. Since the on-load valve testing of one valve has been achieved in the project, a basic idea of how the microprocessor-based testing system may be utilized to test a number of valves of this type is also given.

6.1 Differences of Two on-Load Valve Sequencings

When the Select is on, the actual system identifies the valve by encircling the valve number on the left-hand side of the plotter paper, but the model draws a line under the number on the right-hand side of the paper.

Since the pushbuttons used in this project glow when pressed there is no need of any other signal indicating "On-Load Test" in progress.

The procedure of checking the majority-voting circuit in the Servo-Value Amplifier during the ramp operation may be achieved with additional hardware circuits and extra software for this purpose. The software then may be placed with the delay loops in the subroutines NGOING and PGOING of our program. In fact the execution of the software for this purpose is time-limited. This limit is 117.617 msec for our program where the ramp takes 30 sec to make the valve move from the fully open to the fully closed position. See CHAPTER V, 5.2.2.

If the relay package R1 is disconnected from the Trip pushbutton and operated by a computer controlled separate signal , like the relays R2 and R3 , there will be no need to use extra switches as interlocks, to prevent the accidental valve movement resulting from inadvertently depressing the Trip button. When the Trip is on, however, the computer controlled relay package R1 will receive a signal over an extra output channel from the microcomputer, and consequently the Power Piston will become tripped. This tripping position of the relay will be kept until the Trip is switched off at the end of the test. When the Trip is checked and found to be off by the computer, the program will re-arrange the position of the relays R11 and R12 in the R1 package.

The Governor and the Test signals in our case have been assumed to be produced by the computer and fed in the same input. In fact, on the actual system, the test signal which is a ramp signal is added to the Governor signal and subtracted from the Governor signal during the valve test. A method by which this sort of operation can be simulated is now discussed.

6.2 The Procedure of The Test Signal (Ramp) Addition To Governor Signal

On the actual system, the ramp signal is added to and substracted from the operating Governor signal. In this project, however, it was assumed that even the Governor signal was being obtained from the computer. Because of this assumption, after the valve selection had been completed a constant voltage created through the software was fed into the Governor Model, and it was zeroed to close the valve when the Test button was on and it was increased back to its operating value when the Test button was off. This can be arranged so that the ramp signal may only be added to a constant signal supposed to be the Governor signal and then fed to the Governor. Then the hardware and software should be rearranged. The changes in the hardware configuration in Figure 5.8 are shown in Figure 6.1. Here, the summing amplifier 01, the single inverter 02 and the potentiometer 12PK are on the Analog computer. The re-arranged hardware also includes a double-pole relay R4. The purpose of the pot 12PK is the calibration. When the Select pushbutton is on, the computer changes the position of the relays R2,R3 and R4 in Figure 6.1. These positions' change ensures the fact that during the valve identification the Governor is still on-load, and the X and Y movements of the plotter generated from the microcomputer are not effecting the operating Governor signal. After the identification procedure the relays mentioned above re-gain their positions in Figure 6.1. At this stage of the software already written there is no need of a constant input from the microcomputer to the OP Amp-1 in Figure 5.8. to make the valve on-load since there is a positive input called "INPUT" to the summing amplifier 01 in Figure 6.1. To provide the close Governor signal the amplitude of the computer generated ramp signal at the output of the amplifier 2A3 should be increased to be equal to the constant input signal. This can be done either by reading the constant input in and letting the microcomputer know how high the ramp amplitude is going to be or by reading the error and checking the error against the ramp signal during the time the ramp is being generated. When the error becomes zero the ramp amplitude must not be further increased. Alternatively, the restoration of the operating Governor signal may be achieved by decreasing the generated ramp to zero.

In the system software, the positive and negative ramp generating subroutines, PGOING and NGOING respectively, would be exchanged.

6.3 Discussion Based On The System Software

In the supervisory part of the software the Interrupt Request (IRQ) flags of the PIAs Control Registers were cleared before the control modes were set, and the Data Direction Registers of the output ports were cleared before the identification of these ports as the outputs. For this reason only twenty-six memory locations were needed. In fact, pressing the RESET button on the computer before running a program sets all PIA registers to zero so that these memory locations might be saved. But the flag clearing in this section of the firmware guarantees the fact that no mulfunction will occur because of the software even if a flag had been set for any reason before the program was run.

For the teletype (TYY) commands each word to be printed has been written in subroutine-form and an adequate set of these subroutines are called each time the operator's attention is to be drawn. This way of printing the commands calls for less than four hundred memory locations (356). However, if each piece of the program of the commands to be printed on the TTY was placed at the beginning of each system test step, 1140 memory locations would be needed. By preparing the software portion for this purpose in subroutine-forms more than 700 memory locations have been saved.

6.4 Adaptation of the Microprocessor-Based Single Valve Testing System To Multi-Valve Testing System.

Only one Governor System controlling the position of one valve has been simulated and valve testing of one unit has been outlined in this project.

To adapt the valve testing system in Figure 5.8 to a multi-valve testing system some hardware and software reconfigurations must be done. In the single valve testing system it was assumed that the operating Governer signal was processed by the computer, and to close the valve this signal was zeroed and then increased to open the valve. If all the valves on a steam turbine system are receiving their operating signals from the computer output channels, then closing or opening each valve requires the signal on each dedicated analog output to be zeroed or increased by means of the system software. If one select pushbutton is dedicated to each valve, all the select signals may be input to the computer. The software prepared for this purpose checks each select input in turn and when one of the select buttons is on then the program loads the necessary data in the dedicated memory locations to be referred during valve identification. The computer may also switch an X-Y plotter's X input from one valve position output to a computer output channel and then provide both X and Y movement. After a valve identification the computer switches the X and Y inputs of the plotter to the position output and the Governor input channel of the valve to be tested by arranging a set of relays. Since the computer will know which valve is to be tested after checking the select inputs, it should not be difficult for the computer to arrange the relay positions so that the plotter is switched to the valve under test. In the software, after establishing the I/O ports and the TTY has printed the "PRESS SELECT" command, the computer starts checking all select inputs in a loop. When one particular select button is on, then the program may jump to a subroutine and load the data dedicated to that particular valve in the dedicated memory locations to be referred to during the valve identification. In our program, the subroutine VALNUM stands for this purpose. If there are, say, twenty valves and select input channels, then there should be twenty subroutines each of which loads different maximum X, minimum X and Y values in the dedicated memory locations. When the Test is on, the position output of the desired valve may be switched over to one input channel of the computer. So there is a need of only one input channel to read the desired valve position. One test input will be enough to check if the test button has been pressed. When the valve closure or openning is desired, the software may call a subroutine to identify which computer output channel should receive the closure or openning signal. The common negative or positive going ramp processing subroutine may be used for any valve.

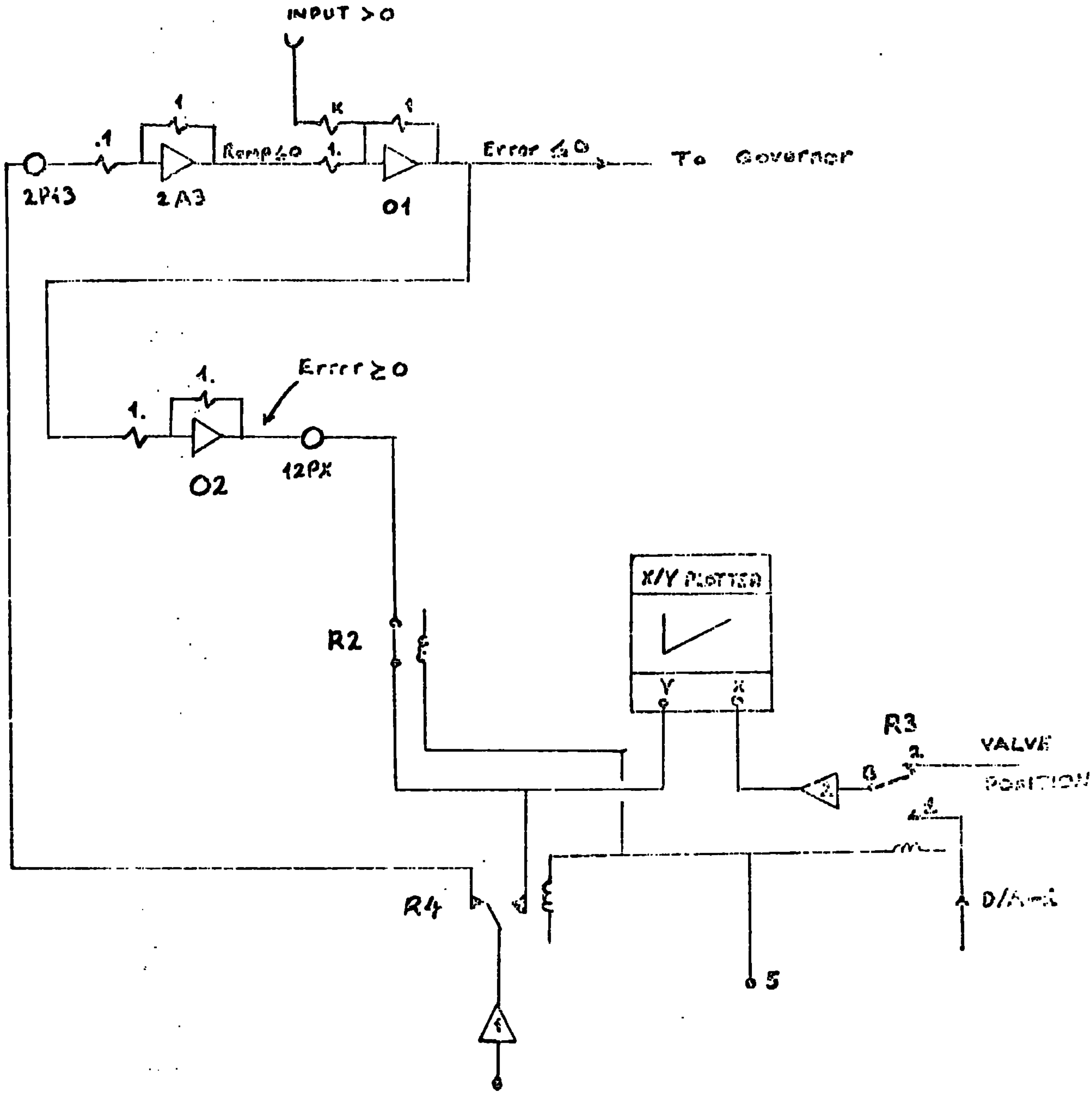


Figure 6.1 Hardware configuration for the simulation of the ramp signal addition procedure.

REFERENCES

1. Aanstad, O. J., and Lokay, H. E., "Fast Valve Control Can Improve Turbine-Generator Response to Transient Disturbances", Westinghouse Engineer, July 1970, pp 114-119.
2. Bur, P. W., "Design Considerations for Achieving Reliable Control Within Shared Microprocessor-Based Digital Controllers", Advances in Instrumentation, ISA-76 Annual Conference, Vol.31, Part 4, 1976.
3. Chen, H. H., "A Microprocessor Control of a Three-Pulse Cycloconverter", IEEE Transactions on Industrial Elec. and Control Instr., Vol. IECI-24, No.3, pp. 226-230, August 1977.
4. Eggenberger, M. A., "An Electrohydraulic Control System For Reheat Turbines", Proceedings of the American Power Conference, 1965, Vol.27, pp. 351-361.
5. Eggers, C. W., "Direct Digital Control Comes of Age", Westinghouse Engineer, July 1969, pp. 98-103.
6. Enfinger, J., and Flannigan, J., "Fly by Wire Technology", AIAA paper 72-882 AIAA Guidance + Control Conf.
7. Fadum, O. K., "A Computer-Coordinated Control System For Batch Digesters", Advances in Instrumentation ISA-76 Annual Conference, 1976, Vol.31, Part 4, Paper No.527.
8. Fitzgerald, A. E., Kingsley, C., and Kuso, A., "Electric Machinery", McGraw-Hill, Tokyo.
9. Ham, P. A. L., "Control Requirements for Large Turbines", Inst.M.C. Symposium: Turbine and Compressor Control, April 1972, pp. 28-35.
10. Higuchi, T., Saito, T., and Kanamata, A., "A Microprocessor - Based Digital Filter Programmed in a Block Diagram Language", IEEE Transactions on Industrial Electronics and Control Instrumentation, Vol. IECI-24, No.3, Aug. 1977, pp. 231-234

11. Kimbark, E.W., "Power System Stability", Vol.I andIII, pub. by John Wiley and Sons 1948.

12. Lewis, J.W., and Davis, J.E., "Development of a Microprocessor-Based Clinical Analyzer Controller", Advances in Instrumentation ISA-76 Annual Conference, 1976, Vol.31, Part 4, Paper No. 510.

13. Lin, A.K., and Koepsel, W.W., "A Microprocessor Speed Control System", IEEE Transactions on Industrial Electronics and Control Instrumentation, Vol.IECI-24, No.3, Aug.1977, pp. 241-247.

14. M6800 Microprocessor Application Mannual, Motorola Inc., 1975.

15. M6800 Microprocessor Programming Manual, Motorola Inc., 1975.

16. McKay, C.W., and Gross, C., "A Microprocessor Based Flow Monitoring System", Advances in Instrumentation ISA-76 Annual Conference, 1976, Vol.31, Paper No.821.

17. Noyes, E.G., and Birnbaum, M., "Electro-Hydraulic Control Improves Operation and Availability of Large Steam Turbines", Westinghouse Engineer, March 1966, pp.49-54.

18. Pfeiffer, J.C., "Process Safety and Reliability Through Process Control Computer/Programmable Controller Interconnection", Advances in Instrumentation ISA-76 Annual Conference, Vol.31, Part 4, 1976.

19. Pracht, C.P., "A Distributed Microprocessor-Based Control System Programmable by the Process Engineer", Advances in Instrumentation ISA-76 Annual Conference, 1976, Vol.31, Paper No.515.

20. Reed,M., and Mergler,H.W., "A Microprocessor-Based Control System", IEEE Transactions on Industrial Elec. and Control Intrumentation, Vol.IECI-24, No.3, August 1977, pp.253-257.

21. Sauer,W.J., and Frank,W.P., "Position Control of Machine Tools Using A Microcomputer", Advances in Instrumentation ISA-76 Annual Conference, Vol.31, Paper No.554, 1976.

22. Schlutz,R., "A Microprocessor-Based Solar Cell Test System", IEEE Transactions on Instrumentation and Measurement, Vol.IM-26, No.4, December 1977.

23. Smith,G.W., and Wood,R.C., "Principles of Analog Computation". Pub. by McGraw-Hill, 1959.

24. SWTPC 6800 Computer System Documentation Notebook, Motorola Inc.

25. Ugetani,A., "Multiloop Process Controller Based on Microprocessor", Advances in Instrumentation ISA-76 Annual Conference, Vol.31, Part 4, 1976.

26. Zimmer,H., "Chiller Control Using On-Line Allocation For Energy Conservation", Advances in Instrumentation ISA-76 Annual Conference, Vol.31, Part 4, 1976, Paper No. 522.

APPENDIX A

BLOCK CALCULATIONS FOR ANALOG COMPUTER MODELLING

The block calculations of the reduced system have been done step by step as the following : [23] (Ei is block input signal and Eo is block output signal).

BLOCK I - Servo-Valve

$$\frac{G2}{1 + T5S}$$

$$\frac{Eo}{Ei} = \frac{G2}{1 + T5S}$$

$$Eo + T5SEo = G2Ei$$

$$T5SEo = G2Ei - Eo$$

$$T5Eo = \int (G2Ei - Eo)\, dt$$

For $G2 = 2.666$ and $T5 = 0.1$,

it can be written $0.1Eo = \int (0.2666 \times 10 \times Ei - Eo) dt$

Figure 7.(a).

BLOCK II - Primary Actuator

$$\frac{G3}{(1 + T6S)S}$$

$$\frac{Eo}{Ei} = \frac{G3}{(1 + T6S)S}$$

$$SEo + T6S^2Eo = G3Ei$$

$$T6SEo = \frac{1}{S} \times G3Ei - Eo$$

We may also write this equation in the following form;

$$T6SEo = \frac{1}{S} \times G3 \times Ei - \frac{SEo}{S} = \frac{1}{S}(G3 \times Ei - SEo)$$

In the time-domain, this becomes

$$T6\frac{dEo}{dt} = \int (G3 \times Ei - \frac{dEo}{dt})dt$$

For G3 = 1.25 and T6 = 0.1, the equation may be written as

$$0.1\frac{dEo}{dt} = \int (0.125 \times 10 \times Ei - \frac{dEo}{dt})dt$$

Figure 7.(b).

BLOCK VI - Power Piston

$$\frac{G6}{(1 + T'8S)S}$$

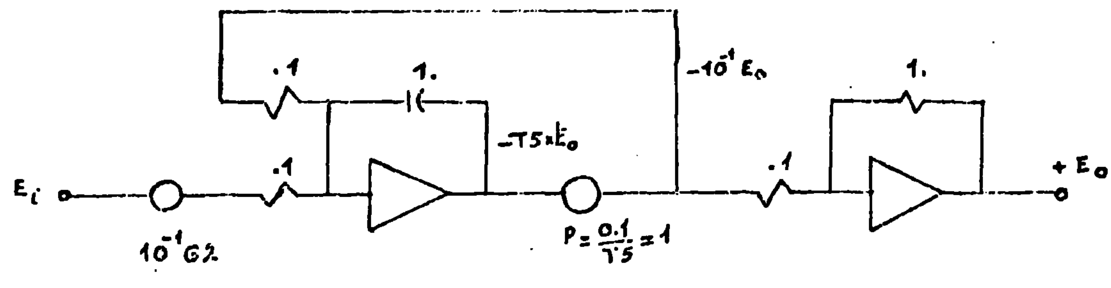
$$\frac{Eo}{Ei} = \frac{G6}{(1 + T'8S)S}$$

$$T'8 \frac{dEo}{dt} = \int (G6Ei - \frac{dEo}{dt})dt$$

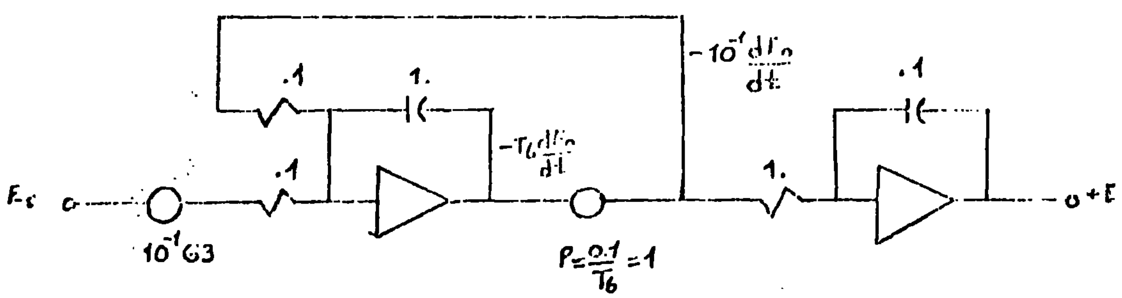
For $G6 = 0.0755$ and $T'8 = 0.01$, it becomes

$$0.01 \frac{dEo}{dt} = \int (0.0755Ei - \frac{dEo}{dt})dt$$

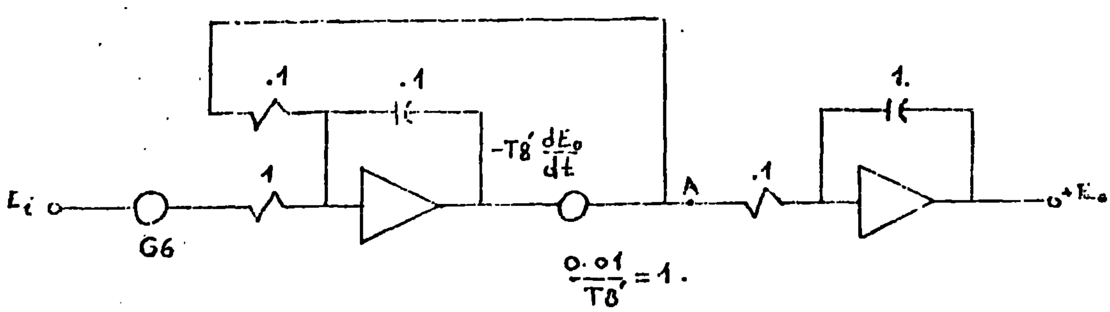
At the point A in the diagram , we have a signal corresponding to $10^{-1} \frac{dEo}{dt}$. See Figure 7. (c) overleaf.



(a)



(b)



(c)

Figure 7. Mathematical model of

(a) Servo-Valve

(b) Primary Actuator

(c) Power Piston

APPENDIX B

FIRMWARE DESCRIPTION AND PROGRAM LISTING

It was possible to punch the program onto computer cards and store the program in a memory file of the Newcastle IBM computer. Then this host computer could be instructed to compile the assembler coded program into machine code. An object tape of the machine code could then be obtained by using the departmental minicomputer Varian V73 because the Varian can easily be linked to the Newcastle computer.

But, our program was loaded via a keyboard and then the object tape of the program was obtained from the M6800. Since the software was developed step-by-step, entering the program parts via the keyboard was less time consuming than the procedure of obtaining the machine code by using the IBM computer and V73 microcomputer.

The software written for the project is given in the following pages. The program commands with their addresses in the successive memory locations are in a table form. The subroutines are arranged in two groups. GROUP-I includes the subroutines used to produce the output signals, read the input signal and store the data in the given memory locations. But the subroutines in GROUP-II are utilized to enable the TTY to print the commands to the operator.[15].

Durham University Science - 7 ... 1979 Section Library

Test Program Listing

```
Memory     Object Code   Source Statements
Location

0586  0F                 SEI
0587  B6 800A            LDAA   800A       DUMMY CLEAR
058A  86 30              LDAA   $£30
058C  B7 800B            STAA   800B
058F  4F                 CLRA
0590  B7 800A            STAA   800A       IDENTIFY THE INPUT PORT
                *
0593  01                 NOP               20 "No Operation" instructions
  .   .                  .
  .   .                  .
05A6  01                 NOP
                *
05A7  B6 8008            LDAA   8008       DUMMY CLEAR
05AA  B6 800C            LDAA   800C       DUMMY CLEAR
05AD  86 32              LDAA   $£32
05AF  B7 8009            STAA   8009       SELECT DDR1.CA2 TO LOW
05B2  B7 800D            STAA   800D       SELECT DDR2.CA4 TO LOW
05B5  4F                 CLRA
05B6  B7 8008            STAA   8008
05B9  B7 800C            STAA   800C
05BC  73 8008            COM    8008       ESTABLISH D/A-1 AS OUTPUT
05BF  73 800C            COM    800C       ESTABLISH D/A-2 AS OUTPUT
                *
05C2  86 0D      LOOP    LDAA   $£CD       CARRIAGE RETURN
05C4  BD E1D1            JSR    OUTEE
05C7  86 0A              LDAA   $£0A       LINE FEED
05C9  BD E1D1            JSR    OUTEE
05CC  86 28              LDAA   $£28       PRINT:(1) PRESS
05CE  BD E1D1            JSR    OUTEE      SELECT
```

```
05D1  86 31          LDAA  $£31      PUSHBUTTON
05D3  BD E1D1        JSR   OUTEE     (
05D6  BD 0314        JSR   PRESS
05D9  BD 03E3        JSR   SELECT
05DC  BD 035A        JSR   PUSBUT
05DF  B6 8008        LDAA  8008      DUMMY CLEAR
05E2  B6 800C        LDAA  800C      DUMMY CLEAR
                *
                *(1)SELECT VALVE
05E5  86 36     LP1  LDAA  $£36
05E7  B7 800D        STAA  800D      SELECT OR;SET MODE CONTROL
05EA  B7 8009        STAA  8009
05ED  B6 8009        LDAA  8009
05F0  2A F3          BPL   LP1       IS SELECT ON?
                *
05F2  01             NOP             40 "No Operation" instructions
  .   .              .
  .   .              .
0619  01             NOP

                *
061A  BD 08E6        JSR   VALNUM    IDENTIFICATION NUMBER
061D  F6 0000        LDAB  0000
0620  F7 8008        STAB  8008      APPLY MINIMUM X
                *
0623  CE DFFF        LDX   $£DFFF
0626  09        K1   DEX             DELAY LOOP
0627  26 FD          BNE   K1
                *
0629  B6 0001        LDAA  0001
062C  B7 800C        STAA  800C      APPLY Y
                *
062F  CE DFFF   K2   LDX   $£DFFF
0632  09             DEX             DELAY LOOP
0633  26 FD          BNE   K2
                *
0635  86 3E          LDAA  $£3E
```

```
0637  B7 800D        STAA  800D      APPLY "PEN ON" COMMAND
                *
063A  F7 8008 LP2    STAB  8008      IDENTIFY VALVE NUMBER
063D  CE 0FFF        LDX   $£0FFF
0640  09      K3     DEX
0641  26 FD          BNE   K3
0643  5C             DECB
0644  F1 0002        CMPB  0002
0647  23 F1          BLS   LP2
                *
0649  01             NOP
064A  01             NOP
                *
064B  CE 04FF        LDX   $£04FF
064E  09      K4     DEX             DELAY LOOP
064F  26 FD          BNE   K4
                *
0651  86 36          LDAA  $£36
0653  B7 800D        STAA  800D      RELEASE PEN
                *
0656  F7 8008 LP3    STAB  8008      SET X TO ZERO
0659  CE 00FF        LDX   $£00FF
065C  09      K5     DEX
065D  26 FD          BNE   K5
065F  5A             DECB
0060  C1 00          CMPB  $£00
0662  26 F2          BNE   LP3
                *
0664  F6 0001        LDAB  0001
0667  01             NOP
0668  01             NOP
0669  F7 800C LP4    STAB  800C      SET Y TO ZERO
066C  CE 00FF        LDX   $£00FF
066F  09      K6     DEX
0670  26 FD          BNE   K6
0672  5A             DECB
0673  C1 00          CMPB  $£00
```

```
0675  26 F2          BNE   LP4
               *
0677  01             NOP             7 "No Operation" instructions
  .    .              .
  .    .              .
067D  01             NOP
               *
067E  B7 8009        STAA  8009      SET CA2 TO HIGH
               *
0681  01             NOP             5 "No Operation"instructions
  .    .              .
  .    .              .
0685  01             NOP
               *
0686  86 C0          LDAA  $£C0      STORE SETPOINT
0688  B7 0F00        STAA  0F00      IN MEM-LOC 0F00
068B  B7 800C        STAA  800C      APPLY IT TO MODEL
068E  CE FFFF        LDX   $£FFFF
0691  09       K7    DEX
0692  26 FD          BNE   K7
0694  CE FFFF        LDX   FFFF
0697  09       K8    DEX
0698  26 FD          BNE   K8
               *
               *(2)PRESS TEST PUSHBUTTON
069A  86 32          LDAA  $£32
069C  BD E1D1        JSR   OUTEE
069F  BD 0314        JSR   PRESS
06A2  BD 03CE        JSR   TEST
06A5  BD 035A        JSR   PUSBUT
06A8  B6 800E        LDAA  800E      DUMMY CLEAR NECESSARY
06AB  86 16          LDAA  $£16      TO DO TEST AGAIN
06AD  B7 800F        STAA  800F      SET MODE CONTROL
06B0  B6 800F  LP5   LDAA  800F
06B3  49             ROLA
06B4  49             ROLA
```

```
06B5  24 F9          BCC   LP5          IS TEST ON?
                *
06B7  01             NOP                11 "No Operation" instructions
  .    .              .
  .    .              .
06C1  01             NOP
                *
06C2  BD 08C5        JSR   VALPOS       READ POSITION OUTPUT
                *                       STORE IN MEM-LOC 0503
06C5  01             NOP                58 "No Operation" instructions
  .    .              .
  .    .              .
06FE  01             NOP
                *
06FF  86 3E          LDAA $£3E
0701  B7 800D        STAA 800D          APPLY "PEN ON" COMMAND
0704  B6 0F00        LDAA 0F00          NEGATIVE GOING
0707  BD 088F        JSR   NGOING       RAMP SIGNAL
                *
070A  01             NOP                19 "No Operation" instructions
       .              .
       .              .
071C  01             NOP
                *
                *(3)PRESS TRIP PUSHBUTTON
071D  86 33          LDAA $£33          PRINT: 3) PRESS TRIP
071F  BD E1D1        JSR   OUTEE        PUSHBUTTON
0722  BD 0314        JSR   PRESS        (
0725  BD 033D        JSR   TRIP
0728  BD 035A        JSR   PUSBUT
072B  01             NOP
072C  01             NOP
072D  01             NOP
072E  B6 800C        LDAA 800C          DUMMY CLEAR
0731  86 3E          LDAA $£3E
0733  B7 800D        STAA 800D          SET MODE CONTROL
```

```
0736  B6 800D  LP6   LDAA  800D
0739  2A FB          BPL   LP6       IS TRIP ON?
                *
073B  01             NOP             21 "No Operation" instructions
  .    .              .
  .    .              .
074F  01             NOP
                *
                *(4)RELEASE TEST PUSHBUTTON
0750  86 34          LDAA  $£34      PRINT: 4)RELEASE TEST
0752  BD E1D1        JSR   OUTEE     PUSHBUTTON
0755  BD 0402        JSR   RELEAS    (
0758  BD 03CE        JSR   TEST
075B  BD 035A        JSR   PUSBUT
075E  01             NOP
075F  01             NOP
0760  01             NOP
0761  B6 800E        LDAA  800E      DUMMY CLEAR
0764  86 04          LDAA  $£04
0766  B7 800F        STAA  800F      SET MODE CONTROL
0769  B6 800F  LP7   LDAA  800F
076C  49             ROLA
076D  49             ROLA
076E  24 F9          BCC   LP7       IS TEST OFF?
                *
0770  01             NOP             36 "No Operation" instructions
  .    .              .
  .    .              .
0793  01             NOP
                *
0794  BD 08A6        JSR   PGOING    CREATE POSITIVE GOING RAMP
                *
0797  01             NOP             28 "No Operation" instructions
  .    .              .
  .    .              .
07B2  01             NOP
                *
                *(5) PRESS TEST PUSHBUTTON
```

```
07B3  86 35          LDAA  $£35     PRINT: 5)PRESS TEST
07B5  BD E1D1        JSR   OUTEE    PUSHBUTTON
07B8  BD 0314        JSR   PRESS    (
07BB  BD 03CE        JSR   TEST
07BE  BD 035A        JSR   PUSBUT
07C1  01             NOP
07C2  B6 800E        LDAA  800E     DUMMY CLEAR
07C5  86 16          LDAA  $£16
07C7  B7 800F        STAA  800F     SET MODE CONTROL
07CA  B6 800F  LP8   LDAA  800F
07CD  49             ROLA
07CE  49             ROLA
07CF  24 F9          BCC   LP8      IS TEST ON?
07D1  01             NOP
07D2  01             NOP
07D3  17             TBA
07D4  BD 088F        JSR   NGOING   CREATE NEGATIVE GOING RAMP
               *
07D7  01             NOP            21 "No Operation" instructions
  .    .             .
  .    .             .
07EB  01             NOP
               *
               *(6)RELEASE TRIP PUSHBUTTON
07EC  86 36          LDAA  $£36     PRINT: 6) RELEASE TRIP
07EE  BD E1D1        JSR   OUTEE    PUSHBUTTON
07F1  BD 0402        JSR   RELEAS   (
07F4  BD 033D        JSR   TRIP
07F7  BD 035A        JSR   PUSBUT
07FA  B6 800C        LDAA  800C     DUMMY CLEAR
07FD  86 3C          LDAA  $£3C
07FF  B7 800D        STAA  800D     SET MODE CONTROL
0802  B6 800D  LP9   LDAA  800D
0805  2A FB          BPL   LP9      IS TRIP OFF?
               *
0807  01             NOP            5 "No Operation" instructions
  .    .
  .    .
080B  01             NOP
```

```
                    *
                    *(7)RELEASE TEST PUSHBUTTON
080C  86 37              LDAA  $£37      PRINT:7)RELEASE TEST
080E  BD E1D1            JSR   OUTEE     PUSHBUTTON
0811  BD 0402            JSR   RELEAS    (
0814  BD 03CE            JSR   TEST
0817  BD 035A            JSR   PUSBUT
081A  B6 800E            LDAA  800E      DUMMY CLEAR
081D  86 04              LDAA  $£04
081F  B7 800F            STAA  800F      SET MODE CONTROL
0822  B6 800F      LPA   LDAA  800F
0825  49                 ROLA
0826  49                 ROLA
0827  24 F9              BCC   LPA       IS TEST OFF?
                    *
0829  01                 NOP             14 "No Operation" instructions
  .    .                  .
  .    .                  .
0836  01                 NOP
                    *
0837  B6 0503            LDAA  0503
083A  5F                 CLRB
083B  BD 08A6            JSR   PGOING    CREATE POSITIVE GOING RAMP
                    *
083E  01                 NOP             23 "No Operation" instructions
  .    .                  .
  .    .                  .
0854  01                 NOP
                    *
                    *(8) RELEASE SELECT PUSHBUTTON
0855  86 38              LDAA  $£38      PRINT: 8)RELEASE SELECT
0857  BD E1D1            JSR   OUTEE     PUSHBUTTON
085A  BD 0402            JSR   RELEAS    (
085D  BD 03E3            JSR   SELECT
0860  BD 035A            JSR   PUSBUT
0863  01                 NOP             7 "No Operation" instructions
  .    .                  .
  .    .                  .
0869  01                 NOP
```

```
086A  B6 8008        LDAA  8008       DUMMY CLEAR
086D  86 34          LDAA  $£34
086F  B7 8009        STAA  8009       SET MODE CONTROL
0872  B6 8009  LPB   LDAA  8009
0875  2A FB          BPL   LPB        IS SELECT OFF?
               *
0877  01             NOP              7 "No Operation" instructions
  .   .              .
  .   .              .
087D  01             NOP
087E  86 36          LDAA  $£36
0880  B7 800D        STAA  800D       RELEASE PEN
0883  01             NOP              9 "No Operation" instructions
  .   .              .
  .   .              .
088B  01             NOP
088C  7E 05C2        JMP   LOOP       RUN PROG AGAIN
               *
               *NEGATIVE GOING RAMP
088F  B7 800C  NGOINGSTAA  800C       NEGATIVE GOING ON D/A-1
0892  CE 1CB6        LDX   $£1CB6
0895  09       SD1   DEX
0896  26 FD          BNE   SD1
0898  01             NOP
0899  01             NOP
089A  01             NOP
089B  01             NOP
089C  01             NOP
089D  4A             DECA
089E  26 EF          BNE   NGOING
08A0  39             RTS
               *
08A1  01             NOP              5 "No Operation" instructions
      .              .
      .              .
08A5  01             NOP
               *
               *POSITIVE GOING RAMP
08A6  F7 800C  PGOINGSTAB  800C       POSITIVE GOING ON D/A-1
```

```
08A9  CE 1CB6         LDX   $£1CB6
08AC  09        SD2   DEX
08AD  26 FD           BNE   SD2
08AF  01              NOP
08B0  01              NOP
08B1  01              NOP
08B2  5C              INCB
08B3  F1 0F00         CMPB  0F00
08B6  23 EE           BLS   FGOING
08B8  39              RTS
                *
08B9  01              NOP                    12 "No Operation" instructions
      .               .
      .               .
08C4  01              NOP
                *
                * READ VALVE POSITION
08C5  86 34     VALPOSLDAA  $£34
08C7  B7 800B         STAA  800B
08CA  01              NOP
08CB  86 3C           LDAA  $£3C             CONVERT COMMAND TO HIGH
08CD  B7 800B         STAA  8008             A/D CONVERSION RUNS
08D0  B6 800B   SD3   LDAA  800B
08D3  2A FB           BPL   SD3
08D5  B6 800A         LDAA  800A             READ DATA IN
08D8  B7 0503         STAA  0503             STORE DATA AT 0503
08DB  39              RTS
                *
08DC  01              NOP                    10 "No Operation" instruction
  .   .               .
  .   .               .
08E5  01              NOP
                *
                * X-Y VALUES FOR VALVE IDENTIFICATION
```

```
08E6  86 3E    VALNUM  LDAA  $£3E     CA2 TO HIGH
08E8  B7 8009          STAA  8009
08EB  86 F5            LDAA  $£F5
08ED  B7 0000          STAA  0000     MINIMUM X
08F0  86 9C            LDAA  $£9C
08F2  B7 0001          STAA  0001     Y
08F5  86 FE            LDAA  $£FE
08F7  B7 0002          STAA  0002     MAXIMUM X
08FA  39               RTS
```

Machine Code Listing of the Test Program

```
*P
S11305860FB6800A8630B7800B4FB7800A01010187
S11305960101010101010101010101010101010141
S11305A601B68008B6800C8632B78009B7800D4F35
S11305B6B78008B7800C73800873800C860DBDE184
S11305C6D1860ABDE1D18628BDE1D18631BDE1D10E
S11305D6BD0314BD03E3BD035AB68008B6800C867A
S11305E636B7800DB78009B680092AF301010101E7
S11305F60101010101010101010101010101010101E1
S11306060101010101010101010101010101010101D0
S113061601010101BD08E6F60000F78008CEDFFF00
S11306260926FDB60001B7800CCEDFFF0926FD863C
S11306363EB7800DF78008CE0FFF0926FD5CF1005A
S11306460223F10101CE04FF0926FD8636B7800D8B
S1130656F78008CE00FF0926FD5AC10025F2F600EF
S1130666010101F7800CCE00FF0926FD5AC10026C0
S1130676F201010101010101B78009010101010132
S113068686C0370F00B7800CCEFFFF0926FDCEFF4C
S1130696FF0926FD8632BDE1D1BD0314BD03CEBDDF
S11306A6035AB6800E8616B7800FB6800F494924C2
S11306B6F90101010101010101010101BD08C501A1
S11306C6010101010101010101010101010101010110
S11306D6010101010101010101010101010101010100
S11306E60101010101010101010101010101010101F0
S11306F6010101010101010101863EB7800DB60F1A
S113070600BD088F0101010101010101010101017F
S113071601010101010101863 3BDE1D1BD0314BD0F
S1130726033DBD035A010101B6800C863EB7800D18
S1130736B6800D2AFB01010101010101010101013C
S113074601010101010101010101 8634BDE1D1BDAF
S11307560402BD03CEBD035A010101B6800E860410
S1130766B7800FB6800F494924F90101010101013F
S1130776010101010101010101010101010101015F
S11307860101010101010101010101010101BD088C
S11307966A01010101010101010101010101010119A
S11307A601010101010101010101010101018635BDBA
S11307B6E1D1BD0314BD03CEBD035A01B6800E8636
S11307C616B7800FB6800F494924F9010117BD08F1
S11307D68F01010101010101010101010101010171
S11307E60101010101018636BDE1D1BD0402BD034B
S11307F63DBD035AB6800C863CB7800DB6800D2AE3
S1130806FB01010101018637BDE1D1BD0402BD032F
S1130816CEBD035AB6800E8604B7800FB6800F4944
S11308264924F901010101010101010101010101014B
S113083601B605035FBD08A6010101010101010111D
S113084601010101010101010101010101010101 8609
S113085638BDE1D1BD0402BD03E3BD035A01010164
S113086601010101B6800886 34B78009B680092AD9
S1130876FB010101010101018636B7800D01010169
S1130886010101010101 7E05C2B7800CCE1CB60927
S113089626FD01010101014A26EF390101010101 89
S11308A6F7800CCE1CB60926FD0101015CF10F0090
S11308B623EE39010101010101010101010101018652
S11308C634B7800B01863CB7800BB6800B2AFBB687
S11308D6800AB7050339010101010101010101010182
S11308E6863EB7800986F5B70000869CB700018668
S10808F6FEB700023909
*
```

TTY Printing Program Listing

```
Memory    Object Code  Source Statements
Location

                       * PRINT : ) PRESS
  0314    86 29        PRESS   LDAA  $£29
  0316    BD E1D1              JSR   OUTEE
  0319    86 20                LDAA  $£20        SPACE
  031B    BD E1D1              JSR   OUTEE
  031E    86 50                LDAA  $£50
  0320    BD E1D1              JSR   OUTEE
  0323    86 52                LDAA  $£52
  0325    BD E1D1              JSR   OUTEE
  0328    86 45                LDAA  $£45
  032A    BD E1D1              JSR   OUTEE
  032D    86 53                LDAA  $£53
  032F    BD E1D1              JSR   OUTEE
  0332    86 53                LDAA  $£53
  0334    BD E1D1              JSR   OUTEE
  0337    86 20                LDAA  $£20
  0339    BD E1D1              JSR   OUTEE        SPACE
  033C    39                   RTS
                        *
                        * PRINT : TRIP
  033D    86 54         TRIP   LDAA  $£54
  033F    BD E1D1              JSR   OUTEE
  0342    86 52                LDAA  $£52
  0344    BD E1D1              JSR   OUTEE
  0347    86 49                LDAA  $£49
  0349    BD E1D1              JSR   OUTEE
  034C    86 50                LDAA  $£50
  034E    BD E1D1              JSR   OUTEE
  0351    39                   RTS
                        *
```

```
0352  01                 NOP                    8 "No Operation" instructions
  .    .                  .
  .    .                  .
0359  01                 NOP
                  *
                  * PRINT : PUSHBUTTON
035A  86 20       PUSBUT LDAA  $£20             SPACE
035C  BD E1D1            JSR   OUTEE
035F  86 50              LDAA  $£50
0361  BD E1D1            JSR   OUTEE
0364  86 55              LDAA  $£55
0366  BD E1D1            JSR   OUTEE
0369  86 53              LDAA  $£53
036B  BD E1D1            JSR   OUTEE
036E  86 48              LDAA  $£48
0370  BD E1D1            JSR   OUTEE
0373  86 42              LDAA  $£42
0375  BD E1D1            JSR   OUTEE
0378  86 55              LDAA  $£55
037A  BD E1D1            JSR   OUTEE
037D  86 54              LDAA  $£54
037F  BD E1D1            JSR   OUTEE
0382  86 54              LDAA  $£54
0384  BD E1D1            JSR   OUTEE
0387  86 4F              LDAA  $£4F
0389  BD E1D1            JSR   OUTEE
038C  86 4E              LDAA  $£4E
038E  BD E1D1            JSR   OUTEE
0391  86 0D              LDAA  $£0D             CARRIAGE RETURN
0393  BD E1D1            JSR   OUTEE
0396  86 0A              LDAA  $£0A             LINE FEED
0398  BD E1D1            JSR   OUTEE
039B  86 28              LDAA  $£28
039D  BD E1D1            JSR   OUTEE
03A0  39                 RTS
                  *
```

```
03A1   01                   NOP             45 "No Operation" instructions
  .    .                    .
  .    .                    .
03CD   01                   NOP
                   *
                   * PRINT : TEST
03CE   86 54       TEST     LDAA $£54
03D0   BD E1D1              JSR  OUTEE
03D3   86 45                LDAA $£45
03D5   BD E1D1              JSR  OUTEE
03D8   86 53                LDAA $£53
03DA   BD E1D1              JSR  OUTEE
03DD   86 54                LDAA $£54
03DF   BD E1D1              JSR  OUTEE
03E2   39                   RTS
                   *
                   * PRINT : SELECT
03E3   86 53       SELECT   LDAA $£53
03E5   BD E1D1              JSR  OUTEE
03E8   86 45                LDAA $£45
03EA   BD E1D1              JSR  OUTEE
03ED   86 4C                LDAA $£4C
03EF   BD E1D1              JSR  OUTEE
03F2   86 45                LDAA $£45
03F4   BD E1D1              JSR  OUTEE
03F7   86 43                LDAA $£43
03F9   BD E1D1              JSR  OUTEE
03FC   86 54                LDAA $£54
03FE   BD E1D1              JSR  OUTEE
0401   39                   RTS
                   *
                   * PRINT : RELEASE
0402   86 29       RELEAS   LDAA $£29
0404   BD E1D1              JSR  OUTEE
0407   86 20                LDAA $£20       SPACE
0409   BD E1D1              JSR  OUTEE
```

```
040C   86 52      LDAA  $£52
040E   BD E1D1    JSR   OUTEE
0411   86 45      LDAA  $£45
0413   BD E1D1    JSR   OUTEE
0416   86 4C      LDAA  $£4C
0418   BD E1D1    JSR   OUTEE
041B   86 45      LDAA  $£45
041D   BD E1D1    JSR   OUTEE
0420   86 41      LDAA  $£41
0422   BD E1D1    JSR   OUTEE
0425   86 53      LDAA  $£53
0427   BD E1D1    JSR   OUTEE
042A   86 45      LDAA  $£45
042C   BD E1D1    JSR   OUTEE
042F   86 20      LDAA  $£20        SPACE
0431   BD E1D1    JSR   OUTEE
0434   39         RTS
```

Machine Code Listing of the TTY Printing Program

«1. ?

```
S11303148629BDE1D18620BDE1D18650BDE1D186D7
S113032452BDE1D18645BDE1D18653BDE1D18653A9
S1130334BDE1D18620BDE1D1295554BDE1D18652D7
S1130344BDE1D18649BDE1D18650BDE1D139010178
S1130354010101010101862ΟBDE1D18650BDE1D135
S11303648655BDE1D18653BDE1D18648BDE1D18630
S113037442BDE1D18655BDE1D18654BDE1D1865457
S1130384BDE1D1864FBDE1D1864EBDE1D1860DBD1F
S1130394E1D1860ABDE1D18628BDE1D1390101014B
S11303A40101010101010101010101010101010135
S11303B40101010101010101010101010101010125
S11303C401010101010101010101018654BDE1D1864C
S11303D445BDE1D18653BDE1D18654BDE1D1398611
S11303E453BDE1D18645BDE1D1864CBDE1D18645FD
S11303F4BDE1D18643BDE1D18654BDE1D13986291D
S1130404BDE1D18620BDE1D18652BDE1D18645BD91
S1130414E1D1864CBDE1D18645BDE1D18641BDE142
S1130424D18653BDE1D18645BDE1D18620BDE1D15C
S10504343900 89
```

Memory Locations From	To	The Purposes of the Program Instructions
0586	-	Inhibiting the microprocessor from servicing an interrupt from a peripheral device. This instruction disables the MIKBUG interrupt routine and enables the use of the interrupt request flags of the PIA control registers as test bits without causing the program to jump into a service interrupt routine.
0587	05C1	Supervising the whole program in general terms. But the instructions in the locations 05AD to 05B4 of this section identify CA2 and CA4 as outputs, and make CA2 and CA4 staying at low.
05C2	05DE	Enabling the teletype to print: (1) PRESS SELECT PUSHBUTTON (
05E5	05F1	Checking if the SELECT is on or not.
061A	0634	Supplying the defined values to the X input and the Y input of the plotter. These values are hexa-decimal F5 for the X axis and 9C for the Y axis. There are two delay loops, one of which is after the X input is given (locations 0623 to 0628) and the other is after the Y input is given (locations 062F to 0634). These delay loops are necessary to obtain the smooth jumping of the pen of the plotter to the valve number on the plotter

paper. The Y axis value, the minimum and the maximum X axis values are stored in the memory locations 0001, 0000 and 0002 respectively via the VALNUM subroutine. As the SELECT command reaches the MPU loads these values in their locations and then the starting values of the both axises are placed on the corresponding plotter axises.

0635 0639 Applying the PEN ON command to the plotter . This is achieved by setting the PIA's control line CA4 to the logic 1 that corresponds to 5 volts.

063A 0648 Drawing an identifying line under the valve number. Since the number has already been reached by the plotter pen and the PEN ON command has been applied, increasing the value on the X axis to a constant value makes the pen draw a line parallel to the X axis of the plotter. For this particular program, after the PEN ON has been applied, the computer output to the plotter's X input increases from the hexadecimal F5 to FF.

064B 0650 The delay loop. This delay loop ensures the smooth return to the set-point on the plotter.

0651 0655 Releasing the pen. The pen is released when CA4 is de-energised.

0656	0676	Via these instructions the pen moves to its set point. This is done by decreasing the values on the X and Y axis to zero in sequence. The delay loops identified by L4 and L5 are used to prevent the pen from sudden jumps.
067E	0680	Short-circuiting the relay R2 and switching the point B of the relay R3 to the point 2 by energising the PIA control line CA2.
0686	068D	Storing an input value to the mathematical model in the memory location 0F00 and also giving the same value to the model itself. The value in the program is hexadecimal C0.
068E	0699	Two delay loops. Since the model itself takes 1.75 sec to response to the step input, these two loops are necessary to give enough time to the model to response adequately. Here two delay loops takes 2.1 seconds of execution time. The reason why 1.75 sec was not used was the fact that the plotter response was taking longer than that of the model.
069A	06A7	Enabling the TTY to print: 2) PRESS TEST PUSHBUTTON (

06A8	06B6	Checking if the TEST is on or not.
06C2	06C4	Reading the valve position via the VALPOS subroutine.
06FF	0703	Application of the PEN ON command by setting the PIA control line CA4 to the logic 1.
0704	0709	Decreasing the model input to zero via the subroutine NGOING.
071D	072A	Enabling the TTY to print: 3) PRESS TRIP PUSHBUTTON (
072E	073A	Checking if the TRIP is on.
0750	075D	Enabling the TTY to print: 4) RELEASE TEST PUSHBUTTON (
0761	0793	Checking if the TEST button has been released.
0794	0796	Enabling the computer to give a positive going signal to the model via the subroutine PGOING.

07B3	07C0	Enabling the TTY to print: 5) PRESS TEST PUSHBUTTON (
07C2	07D0	Checking if the TEST button has been pressed down.
07D3	07D6	Reducing the model input to zero through the subroutine NGOING.
07EC	07F9	Enabling the TTY to print: 6) RELEASE TRIP PUSHBUTTON (
07FA	0806	Checking if the TRIP button has been released
080C	0819	Enabling the TTY to print: 7) RELEASE TEST PUSHBUTTON (
081A	0828	Testing if the TEST button has been released.
083A	083D	Supplying back the operating input signal of the valve control device before the test sequencing had started.
0855	0862	Enabling the TTY to print: 8) RELEASE SELECT PUSHBUTTON (

086A	0876	Testing if the SELECT button has been released.
087E	0882	Releasing the PEN ON command.
088C	088E	Enabling the operator to do the test sequencing again.

GROUP I

Memory Locations		Subroutines	
From	To	Name	Purposes
088F	08A0	NGOING	Storing the contents of the accumulator A at the output address 800C, decreasing A by one and continuing the storing until the value in A becomes zero. This procedure is done so that 3 seconds of processing time are spent until the contents of A drops to zero from the hexadecimal FF.
08A6	08B9	PGOING	Storing the contents of the accumulator B at the output address 800C, incrementing B by one and comparing the new value with the contents of the memory location 0F00. Storing continues until the value in B becomes equal to the value compared with. In this subroutine, increasing the contents of B from 00 to hexadecimal FF takes 3 sec of execution time.
08C5	08DB	VALPOS	Reading the model output at the input port address 800A and storing it in the memory location 0503.

08E6	08FA	VALNUM	Setting CA2 to high resulting in the fact that the relay R2 becomes open-circuit and the points B and 1 of the relay R3 becomes short-circuit. Storing the hexadecimal data F5, 9C and FE in the memory locations 0000, 0001 and 0002 respectively.

GROUP II

Memory Locations		Subroutines	
From	To	Name	Purposes
0314	033C	PRESS	Enabling the TTY to print a close-bracket and leave a space. Then the TTY prints the letters P, R, E, S and S. These printed letters are followed by a second space.
033D	0351	TRIP	Enabling the TTY to print T, R, I and P.
035A	03A0	PUSBUT	Leaving a space and printing the letters P, U, S, H, B, U, T, T, O and N. This operation is followed by a Carriage Return and a Line Feed command. Then the TTY prints an open-bracket character at the beginning of the next line.
03CE	03E2	TEST	The TTY prints the letters T, E. S and T.
03E3	0401	SELECT	The letters S, E. L, E, C and T are printed.
0402	0435	RELEAS	Printing a close-parenthesis character, leaving a space and then printing the letters R, E, L, E, A, S and E. A

second space is obtained after these letters have been printed.

APPENDIX C

SELECTION OF A VALVE NUMBER ON THE PLOTTER PAPER USED WITH THE MICROPROCESSOR-BASED VALVE TESTING SYSTEM.

If it is desired to underline the number 14 on the plotter paper, the 'Load Accumulator A' instructions are to be loaded with 245, 254 and 156 decimal numbers in the memory locations 08EB, 08F5, and 08F0 respectively. Then the specified setpoint is arranged. When the program runs, the plotter draws the line under 14 if the command SELECT is on.

Checking back can be done as the following:

Firstly, because of the specification of the conditional jump in the memory location 0647, during underliningthe valve number 14 the maximum number on the Output Register that is connected to the converter D/A -1 on the X input path of the plotter is decimal 255 but not 254. This full scale reading gives 3.8 volts. The decimal 245 and 156 correspond to 3.65V 2.32V respectively. Since the gain of the amplifier on the X-axis is 3.35, the signals 3.65V and 3.8V should be multiplied by 3.35. The gain of the amplifier on the Y-axis is 2.26. Then the signal 2.32V should be multiplied by 2.26.

$$3.65V \times 3.35 = 12.23V$$

$$3.8V \times 3.35 = 12.73V$$

$$2.32V \times 2.26 = 5.24\ V$$

Since the scaling on both axes is the same 500mV/cm, the amplifiers' outputs should be divided by 0.5V/cm. Then the minimum

value on the X-axis is $\dfrac{12.23V}{0.5V/cm} = 24.46cm$

the maximum value on the X-axis is $\dfrac{12.73V}{0.5V/cm} = 25.46$ cm

the value on the Y-axis is $\dfrac{5.24V}{0.5V/cm} = 10.5$cm

These are the distances from the setpoint of the plotter. As it will be seen from Figure 58, these are the correct distances for the number 14.

ham Un
Librar