

AN ADAPTIVE CONTROL ALGORITHM FOR A

CNC MILLING MACHINE

AN ADAPTIVE CONTROL ALGORITHM FOR A
CNC MILLING MACHINE

By

Gajananda Nandakumar Mailvaganam
B.Sc (Eng.) (Hon.) (Ceylon)
C.Eng. M.I., Mech. E. (London)

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfilment of the Requirements
for the Degree
Master of Engineering

McMaster University

April, 1974

MASTER OF ENGINEERING (1974)
(Mechanical)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: An Adaptive Control Algorithm for a CNC
Milling Machine

AUTHOR: Gajananda Nandakumar Mailvaganam
B.Sc.(Eng.) (Hon.) (Ceylon)
C.Eng. M.I. Mech. E. (London)

SUPERVISOR: Professor J. Tlusty

NUMBER OF PAGES: 120

ABSTRACT:

The purpose of this project was to develop an Adaptive Control Algorithm for a CNC milling machine. The milling machine is controlled by a 2100A Hewlett Packard mini-computer. The Adaptive Control Software has to operate in unison with an already available Numerical Control Software. Both these programmes are stored in the computer and the computer operates on them with the aid of the interrupt pulses received from the Time Base Generator located in the Controller.

The Adaptive Control Software should be capable of optimising the milling process, that is enabling the milling machine to operate at the highest feed-rate without violating or overriding the maximum permissible values of the horizontal force and torque acting on the cutter. These maximum values of the force and torque are determined from the tool strength and capacities of the servo drives and spindle motor. Further, the machine should be able to arrive at the above feed-rate in the shortest possible time interval without causing cyclic variations in the feed-rate which could lead to an unstable system. The programme should be able to obtain ten samples of the parameters per revolution of the spindle. The feed-rate thus obtained (after comparing with the maximum and minimum feed-rates of the machine and making any corrections, if necessary) should be stored in a memory location accessible to the Numerical Control Programme. The instantaneous values of the force and torque are transmitted to the computer via the transducers attached on the spindle of the machine and the Analog-to-Digital Processor, therefore, the Adaptive Control Software will have to communicate with the Analog-to-Digital Processor in order to receive the values of the forces and torque. Thus the above mentioned requirements will have to be met by this piece of software. With this end in view, the following algorithm was developed.

The algorithm consists of two portions, namely, the Data Reading Routine and the Policy Routine. The former accepts the two horizontal forces (which are phase shifted by 90°) and the torque acting on the cutter by communicating with the Analog-to-Digital Processor. However, all these three parameters are received through the same channel from the Analog-to-Digital Processor as such a method of identifying the variables was necessary. For this purpose, the Data Reading Routine consists of software capable of communicating with the Analog-to-Digital Processor at time intervals of 10 m.sec. and receiving the data in a digital form, decoding the input and ascertaining which input parameter was received. The Policy Routine has two modes of operation viz., the constraint and optimizing modes. This routine ascertains the critical error and arrives at the new feed-rate depending on the Policy used. After checking the value of this feed-rate with the maximum and minimum feed-rates available on the machine (and corrections made if necessary), the suitable value of this feed-rate is stored in a memory location accessible to the Numerical Control programme. This gives the general structure of the Adaptive Control Algorithm developed in this project.

ACKNOWLEDGEMENTS

I would like to sincerely thank my supervisor, Professor J. Tlusty, for suggesting the research topic, his knowledgeable advice, critical appraisal, continuous enthusiasm and encouragement, through this research programme and for the rewarding experience of working under his guidance.

I extend my appreciation and sincere thanks to Dr. Y. Koren for his devoted guidance, encouragement and unrelenting insistence on adequate rigour and the valuable contributions he made for the fruitful completion of this work.

My thanks are extended to Dr. Frans Van Dijck and Mr. Peter McNeil for the valuable discussions during the course of this work.

I gratefully acknowledge the financial assistance from the National Research Council of Canada and McMaster University.

I am grateful for the patience and efficiency of Miss Deborah Tudin in typing from a difficult manuscript.

TABLE OF CONTENTS

	<u>Page</u>
Chapter 1	1
INTRODUCTION	
1.1 Numerical Control (NC)	1
1.2 Direct Numerical Control (DNC) and Computer Numerical Control (CNC)	6
1.3 Adaptive Control (AC)	9
1.4 Programmable Adaptive Control	11
Chapter 2	13
The General Description of the System	
2.1 The Milling Machine	14
2.2 The Controller	16
2.2.1 The Auxiliary Controller	16
2.2.2 The Servo Drive Unit (SDU)	17
2.2.3 Drives and Feedback Devices	18
2.3 The Hewlett Packard 2100A Computer	19
2.3.1 General Specifications	20
2.3.2 Interrupt System	21
2.4 System Operation	26
Chapter 3	27
The Sensed Variables	
Chapter 4	34
The Dynanometer Description	
Chapter 5	40
The Adaptive-Control Hardward	
Chapter 6	44
The CNC System	
6.1 System Description	44
6.2 The NC Programme	47
6.2.1 The Initiator Routine	47
6.2.2 The Continuator Routine	48
6.2.3 The Digital Differential Analyzer (DDA)	50
6.2.4 The Interpolator	51
6.3 Lost Motion	53
Chapter 7	55
The NC Data Programme (DP) and the Data Reading Programme (DRP)	
7.1 General	55
7.2 NC Data Programming	55
7.3 The Data Reading Programme (DRP)	60

TABLE OF CONTENTS Continued

	Page	
Chapter 8	The Adaptive Control Programme	62
	8.1 Data Reading Routine	62
	8.2 AC Policy Routine	64
	8.3 AC-NC System	69
Chapter 9	Conclusion	71
APPENDIX A	The HP 2100A Computer	105
APPENDIX B	The Adaptive Control Programme (AC)	111
APPENDIX C	The Data Reading Programme (DRP)	115
BIBLIOGRAPHY		120

LIST OF FIGURES

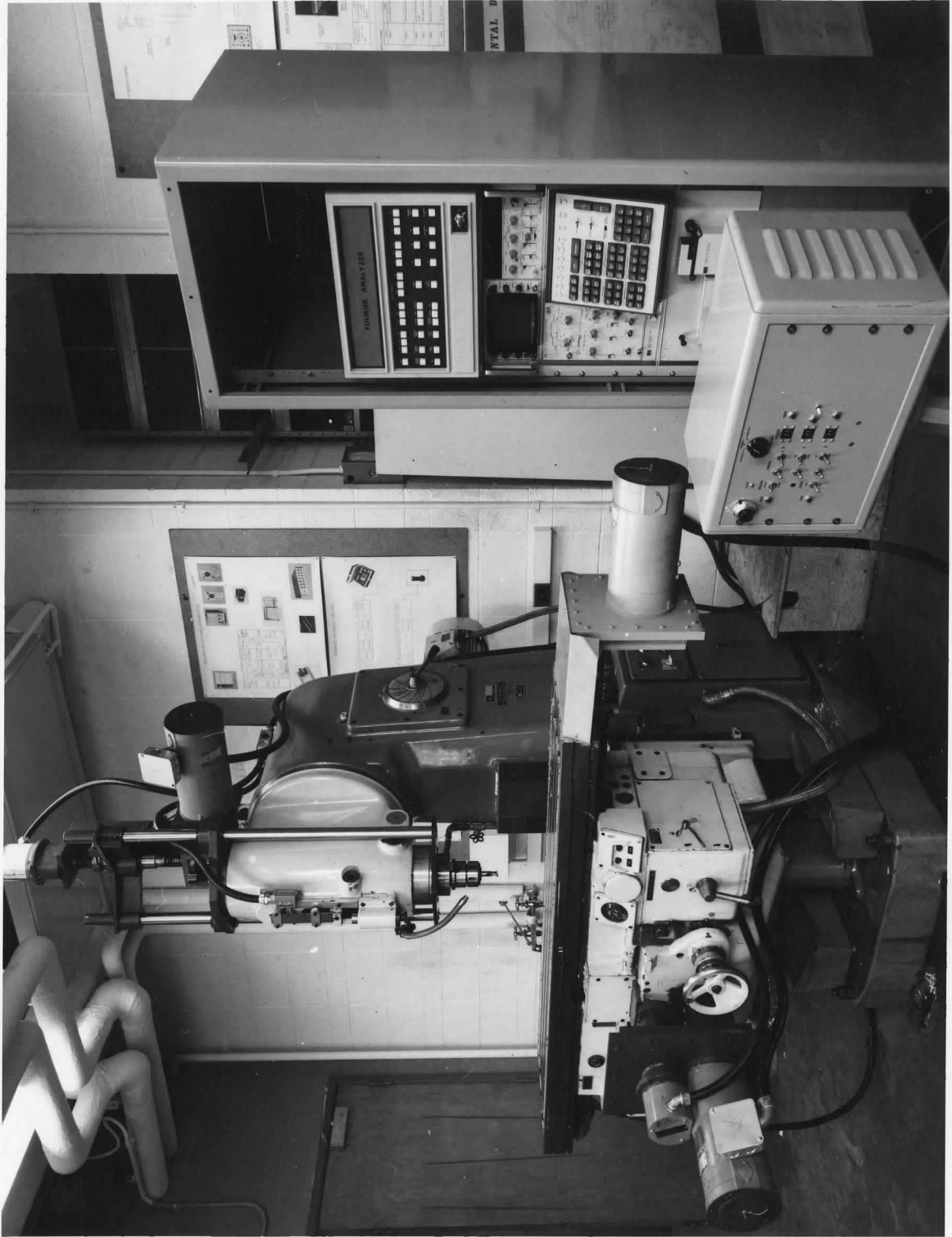
Figure No.	Illustration	Page
1.1	Block diagram for a numerically controlled machine tool, closed-loop system	75
1.2	Conventional Hardwired NC Control System	76
1.3	Hybrid Configuration	77
1.4	Software CNC Configuration	77
1.5	Adaptive Control System block diagram	78
2.1	Block diagram of CNC-AC System	79
2.2	Block diagram of the Milling Machine, Computer and Controller	80
2.3	Priority Linkage	81
2.4	Interrupt Sequences	82
3.1	Graph of Force Vs Length of Cutter	83
3.2	Graph of Torque Vs Diameter of Cutter	84
4.1	Dynamometer (Assembly)	85
4.2	Flexure Plate	86
4.3	Torque Ring	87
5.1	Electronic Bridge	88
5.2	Analog-to-Digital Processor (ADP)	89
5.3	ADC-10Z Timing Diagram	90
6.1	Initiator Routine	91
6.2	Continuator Routine	92
6.3	Digital Integrator	95
6.4a	Interpolator	96
6.4b	Circular Interpolator	96
6.5	Initial conditions of circuit	96

LIST OF FIGURES continued

Figure No.	Illustration	Page
6.6	Profile to check "Lost Motion"	97
7.1	Data Manuscript	94
7.2	Data Reading Programme (DRP)	98
8.1	Data Reading Routine	99
8.2	AC Policy	100
8.3	The AC Strategy	101
A1	Data Formats and Octal Notation	102

LIST OF TABLES

Table No.		Page
1	Preparatory Functions	93
2	Miscellaneous Functions	94
A2	Memory Pages	103
A1	Extended Arithmetic Execution Times	104



The CNC-AC System

CHAPTER 1

INTRODUCTION

1.1 Numerical Control (NC)

Numerical control may be termed as the most dynamic manufacturing process in use. The history of this process dates back to July of 1949, when at the Massachusetts Institute of Technology an initial research project was carried out under the sponsorship of Parsons Corporation of Traverse City, Michigan and the United States Air Force. Numerical Control has had a tremendous influence on manufacturing in the twentieth century. This discipline presents not only engineering problems, but also economic and sociological ones, whose impact in one way or another will eventually effect all of society.

Numerical control is defined by The Electronics Industries Association as: "A system in which actions are controlled by the direct insertion of numerical data at some point. The system must automatically interpret at least some portion of this data". The basic system may be represented as shown by the block diagram in Figure 1.1. The total system may be subdivided into eight distinct units. These units being the input medium, reader, store, comparing equipment, amplifying equipment, operational devices, machine drive, and feedback. The input

medium consists of physical material carrying the sequential instructions that the machine tool will need to perform its operation. The input to a numerically controlled machine tool could take a wide variety of forms, such as manual input, punched-paper-tape input, magnetic tape and punched cards. The most commonly used input medium is a 1-in.-wide punched paper or Mylar tape. The control unit of the system consists of the reader, store and the comparing equipment. The information present on the input medium is transferred by the reader to the store element of the control unit. The store unit holds and preserves this information received from the reader until it is required for the operation of the system and also generates position signals which is analogous to the desired position of the operational device. The operational device transfers the workpiece to a predetermined position or series of positions and it enables operations to be performed on the workpiece at the above position or positions. It is evident from this that the operational device has two functions. The machine drive provides the necessary power to the operational device. The feedback unit may be considered as a unit capable of generating a position signal which is analogous to the present position of the operational device. The feedback unit is coupled to the operational device. The comparing equipment compares the theoretical-position signal from the store and the present- or actual-position signal from the feedback mechanism and

generates an error signal, which, when amplified causes the machine tool to move the operational device to a location where the theoretical position signal and the feedback signal will be at electrical null. The purpose of the amplifying equipment is to amplify the signal from the comparing equipment, so that, this signal will cause the machine drive to move the operational device and consequently the feedback mechanism, to the point called for by the theoretical signal.

The control unit is an information-handling device and could be classified as the most important element in a numerically controlled machine tool. The basic function of the control unit is to accept information from the input medium, store it, interpret it, and then generate electrical command pulses which cause the movement of the operational device. The frequency of these command pulses determine the velocity, while the number of pulses determine the distance of this movement. As mentioned above, the control unit comprises the tape reader, buffer storage (temporary), the interpreter section and electronic circuitry necessary to produce the necessary and appropriate pulses to actuate the machine tool. The buffer storage functions as a temporary information handling section for the data received from the tape. The purpose for a buffer storage arises so as to nullify any stoppage of the machining process due to the delay of data transfer from tape to the control unit. Such stoppage could cause a strain on the whole system and defects on the surface finish of the work.

The interpolation section is typical for contouring systems, but the section is absent in positioning systems. The purpose of the interpolation section is to obtain the intermediary points (co-ordinate values) on a straight line from given start and end co-ordinate values or on a curve from its given mathematical expression. Machine Tool control systems may be classified into four categories:

(a) Positioning System: This is the simplest form of numerical control. This form of control is usually associated with equipment that perform operations at specific positions and does not effect the workpiece when moving from one point to the next. The exact path that the spindle takes in moving from point to point is immaterial, providing, of course, the time required is reasonable and the spindle does not collide with either the part or the holding fixture. The feed-rate used in point to point positioning is the maximum feed-rate permissible for the machine tool.

(b) Positioning/Straight-Cut System: This system has, in additon to positioning, the capability to perform milling along each machine tool axis. Thus it could perform milling and facing operations on any rectangular shape.

(c) Contouring System: This system is capable of all operations described under positioning systems and straight cut systems and in addition, it could execute straight lines at any

angle; circles, conics or any other mathematically definable configurations.

(d) Contouring/Positioning Combination System:

This system combines the advantages of each mode. If the machine control unit is in the contouring mode, it works exactly like a contouring system. However, if strictly positioning is necessary, the path control is eliminated which increases the operating speed.

The conventional NC systems discussed above, fall into two categories. In the first case, the control system could be programmed to receive the "gross" data, such as two points specifying a straight cut, and using this information, to perform the necessary interpolation to make the machine tool move in a line between these two points or given the specification of a circle or conic, the necessary interpolation is performed for the machine tool to move along the specified circle or conic within acceptable tolerances. In these control systems the interpolation and feed-rate calculation features are incorporated. The part programmer processes the part programme in a general-purpose computer and the punched tape obtained from the computer which contains the "gross" data is used as the input medium to the above control systems. It is evident from this that the above mentioned capabilities call for more sophisticated control systems. In the second category the interpolating and feed-rate calculation features are not incorporated in the control system of the machine tool. This function is performed in a special

purpose computer, usually called a director. The director, performing interpolation and feed-rate calculations, produces a control medium which contains all of the interpolated data necessary for the machine tool on a real time basis. In this case, the part programmer first processes the part programme in a general-purpose computer and obtains the punched tape with "gross" data. This tape is then fed into the director which performs the geometric and feed-rate interpolation from the "gross" data punched tape and produces a control tape containing the interpolated geometry and feed rate information. This tape is next fed into the control system of the machine tool which reads it and commands the servosystem of the machine tool. The first category is called the on-line director process while the latter is called an off-line director process.

1.2 Direct Numerical Control (DNC) and Computer Numerical Control (CNC)

In numerical control systems we have observed that there are systems

(a) which utilize the director to produce the interpolated control media from the "gross" data produced by the general purpose computer,

(b) systems which are capable of internal interpolation.

(c) Finally, we have systems which utilize the computer for directly controlling the machine tool, thereby completely eliminating any type of control medium.

The term DNC will denote a numerical control system wherein a multimachine manufacturing process is controlled by a remote powerful computer or a group of computers. In this system instructions are issued directly by a central computer wherein all the functions (primary and secondary) are executed. In DNC systems the most commonly used input devices are: teleprinter and cathode ray communications terminals with keyboards, magnetic tape, paper tape, and Hollerith card readers; while in hardwired numerical controls, the part programme data is fed in at the machine location by means of a built-in tape reader. The part programme data can be fed into the computer at a relatively high speed using any appropriate input device.

In a DNC system, the individual machines can be connected in the following ways:

(a) The machine tool in the DNC system has a stand-alone controller which could operate off-line from the DNC system, with a tape input as well as under the central computer control. The stand-alone controller can be a computer or a conventional NC controller (hardwired). [2]

(b) In this case, the NC control functions are shared between the central computer and a reduced special-purpose controller which is dedicated to the machine tool. Here the path generation function is separated into a coarse interpolation in the central computer and a fine linear interpolation in the special-purpose controller. This special-purpose controller may consist of either hardwired logic or a small dedicated computer. Since functions are shared with the central computer, stand-alone operation of the machine tool is not possible.^[2]

(c) Here all the NC functions for several machines, including path generation and loop closure are performed in a central computer.

In more sophisticated DNC systems more than one central computer is used and the various functions are shared by the individual computers.

A computer Numerical Control System (CNC) may be visualized as a system in which, a single machine tool is controlled by a dedicated mini-computer. The mini-computer serves as part of the hardware of the system. Figure 1.2 illustrates in detail a conventional hardwired NC control system. Data is read into the core unit either by tape or by manual input. The core unit provides buffer storage, decoding and sequencing logic, decimal-to-binary conversion, etc., while the auxiliary

function commands and tool change commands are directly sent to the machine tool via the appropriate interface relay logic at the machine tool. It should be noted here that virtually all conventional NC contouring control systems utilize pulse-oriented digital circuitry. In the CNC systems two alternatives to the conventional control systems have been adopted. They are:

(a) A hybrid combination consisting of a computer and hardwired logic, the general-purpose mini-computer replaces the functions of the conventional core unit and the conventional hardwired circuits are retained for the interpolation and position loop logic. This system is illustrated in Figure 1.3.[11]

(b) A mini-computer system capable of all control functions, including interpolation and position-loop closure. These are performed with the aid of the software present in the computer. Figure 1.4 depicts this system. [11]

1.3 Adaptive Control (AC)

Adaptive control may be understood as a control system which measures some process variable or variables such as torque, force, spindle deflection, cutting temperature, etc. and uses these measurements to adjust feed and/or speed in the operation. Figure 1.5 illustrates an NC system with adaptive control. The sensors mounted on the machine tool generate signals

to the adaptive control system. The signals which are received from the sensors are analyzed by the adaptive control system, and appropriate modifier signals are sent to the NC system. The NC system in turn provides the necessary slide velocities and spindle speed commands to the machine tool. The adaptive control system like a human operator senses tool deflection, vibration, tool temperature, etc. at the work zone and enables the numerical control system to make the necessary changes in the feed-rates and/or spindle speed in order to optimize the cutting process within the prevailing condition in the work zone. In the conventional adaptive control systems, the limiting values of the constraints are manually preset in the adaptive controller. In effect these settings define the range of permissible operation for the adaptive control system. Using these values as references, this system analyzes the prevailing conditions at the work zone (from the information it receives from the sensors) and informs the NC system for necessary action. The preset constraints could be

- (a) maximum spindle speed
- (b) minimum spindle speed
- (c) maximum torque
- (d) maximum chip load
- (e) maximum feed-rate
- (f) impact chip load
- (g) maximum vibration

(h) maximum force

(i) minimum chip load

In effect these would define the range of permissible operation of the machine tool.

1.4 Programmable Adaptive Control

With the development of stored programme control systems, the adaptive control function is achieved by software. In the conventional type of adaptive control systems, the adaptive control function was achieved via hardware components. In this new system the adaptive controller is a software programme which contains the limits of the constraints and the necessary software to handle the signals from the sensors on the machine and compute the optimum feeds and speeds for the particular machining process. Therefore, it is noticed that in an adaptive control system real-time measurements of the parameters affecting the workpiece are found and controlled in order to optimize the cutting process. If a mathematical model is constructed for a (say) milling operation, the spindle speed and feed are the independent-variables while the tool tip temperature, spindle torque, the resultant force on the cutter and workpiece vibration are the dependent variables. Now, if this model is accurate, then the optimum spindle speed and feed of the cutter could be preprogrammed for all the individual stages of the operation and, hence, the need for adaptive control

would not arise. However, at present it is not possible to construct such a mathematical model except in the most elementary or in ideal cases such as the turning of a bar of constant diameter, of homogenous material, clamped at both ends in a perfectly rigid machine (lathe) having a tool bit of unvarying sharpness. It should be mentioned here that the introduction of "noise" into a simplified system is so great that to make a good simulation is too laborious and impracticable. Noise is represented by cutting fluids that cool the tool tip in an unpredictable manner by a variety of work clamping methods, by cutter wear, and by non-homogeneous metallurgical properties of the material. The vibration characteristics are determined by the shape of the workpiece as such require a mathematical analysis for each workpiece shape. As such, mathematical models may become a reality as computation and simulation skills improve, however they may not be economically realistic at present.

CHAPTER 2

THE GENERAL DESCRIPTION OF THE SYSTEM

An adaptive control system (in general) consists of three stages:

(1) The Identification Stage, in which the parameters (which will be used to ascertain the optimal working point) are identified.

(2) The Decision Stage, in this stage the optimal working point is calculated according to a prescribed strategy. This stage uses as an input the identified parameters of the Identification Stage.

(3) The Implementation Stage, here the theoretical optimal point which was obtained from the decision stage is carried out.

In a CNC-AC System the identification process requires the following equipment:

(1) Transducers for sensing the process variables which will be used in the decision process.

(2) An amplification unit to amplify the sensed variables.

(3) An Analog-to-Digital Processor (ADP) which is necessary to convert the analog data to a digital form which is acceptable to a mini-computer.

The general structure of a CNC-AC system is given in Figure (2.1). The elements of the identification process

will be given in Chapters 4 and 5 of this thesis.

The decision process in a CNC-AC system consists of pure software and will be referred to henceforth as the Adaptive Control (AC) Programme. The latter will be described in Chapter 8 of this thesis.

The Implementation Stage carries out the decisions arrived at by the Decision Stage so as to optimize the working conditions. In a CNC-AC System, the CNC portion of the system acts as the implementation unit. This unit consists of two portions (1) a software portion, which will be denoted henceforth as the NC Programme,

(2) the hardware portion including the control loops termed as the Controller (shown in Figure 2.1), the Servomotors and the feedback elements.

A general description of the milling machine, its main control components, the controller and the mini-computer which was used in this project are given below.

2.1 THE MILLING MACHINE

A FA4V knee type TOS milling machine was retrofitted for this project. "The word retrofit is used to outline at least three machine tool applications. These may include , first, replacement of old numerically controlled electronic systems. Many current users of numerically controlled equipment originally purchased their machine tools during the early days of numerical control development and currently are finding considerable merit in removing the older and often less

reliable control system and replacing it with new solid-state electronic hardware. In these cases, the machine tool is less than ten years old and is far from being mechanically worn out, and the addition of the new control system adds substantially to the overall reliability and performance. The addition of numerical control hardware to machine tools originally designed for, but not delivered with numerical control capability, is the second category of retrofits. This type of machine tool is usually delivered equipped with ball-nut lead screws, anti-friction ways and some type of tracer or profiling capability. The third and perhaps the most important classification of retrofits is furnishing numerical control to machines originally designed for manual or tracer operation. "[10]

The TOS milling machine falls under the third category. This machine was fitted with two ball bearing lead screws, one for the table and the other for the saddle, having a pitch of 0.25 inches.

The total length of the saddle lead screw is 22 inches and its diameter is 1.0 inches. The lead screw for the table has a length of 46 inches and a diameter of 1.5 inches. Each lead screw has a bearing housing. On the drive end, two thrust bearings are preloaded against each other and a radial bearing is used to support the shaft. The bearing housing on the opposite end of the shaft carries only a supporting radial bearing. The drive for the vertical motion of the spindle

employs a conventional lead screw driven by a D.C. motor through a worm gear reduction.

The basic design is similar for both axes, although different sizes of bearings, oil seals and gears are used. In the saddle, the lead screw is fixed axially while the nut is attached to the saddle, moving the latter in response to the lead screw rotation. The table, however, is moved by axial motion of the lead screw through a stationary nut.

2.2 THE CONTROLLER

The control, which serves as an interface between the computer and the motors, consists of two units: the Auxiliary Controller and a Servo Drive Unit and is illustrated in Figure (2.2).

2.2.1 The Auxiliary Controller

The Auxiliary Controller contains a Time Base Generator (TBG), the x, y, z axes position control boards and a manual control. The Time Base Generator serves the purpose of providing

(a) interrupt pulses for the computer (via a frequency divider) to switch the computer control from the AC programme to the NC programme.

(b) pulses for the manual control of the milling machine

(c) reference signals for the resolvers through a divide by 1000 counter and two low-pass filters. The computer

sends pulses which are converted into command signals. The x,y,z control feedback signal from the resolvers and the resulting error is fed to the servomotors via the servo control unit. This causes the motor to rotate in a direction to reduce the error. In the manual mode pulses are obtained from the Internal Pulse Generator (instead from the computer) and these pulses are controlled by the external knob of the potentiometer on the controller panel.

2.2.2 The Servo Drive Unit (SDU)

This unit controls the DC motors by a switching technique known as pulse with modulation. This unit is basically a velocity controller. A velocity command analog signal is applied to the servo and the servo operates to force the actual velocity analog signal to match it, thus achieving velocity control. To do this, the servo controls the width of the pulses applied to the motor circuit from 0 to 100 percent "on" time, at a rate of 2000 per second (the exact frequency is not critical for correct servo operation). The rectangular pulses applied to the motor terminals are filtered by inductance (natural or added externally) of the armature circuit. The armature current produced is related to the average pulse width. Thus, varying the pulse width effectively varies the applied motor voltage over a continuous range and enables the use of transistor switches in their most efficient operating mode.

2.2.3 Drives and Feedback Devices

Servomotors: The servomotors used have a maximum speed of 1750 R.P.M., torque of 54 lb. in. and a horse power of 1.7 Type SV-7114 BW-1, Rating 20.8 V/1000 R.P.M., manufactured by Servo-Tek Products Co., Hawthorne, N. J.

The motor is connected in a separate exciting mode which means that the field is connected to a constant voltage of 90V and the armature is connected to the power amplifier output. The speed of the motor is linearly proportional to the voltage supplied by the power amplifier, where the latter could change between 0 to +90V for one direction of motion and from 0 to -90V for the other direction.

Resolvers: The position measuring device used in the control loop is a resolver. Type MFG 09301 manufactured by Singer-General Precision, Inc. It consists of two stator windings and one rotor winding. The two stator windings are fed by two reference sine-wave signals equal in amplitude, which are 90 degrees-phase shifted. The rotor output voltage, V_b :

$$V_b(t) = V_r \sin(\omega t + \phi) \quad (2-1)$$

where $\omega = 2\pi f$ and f is the reference voltage frequency, V_r = amplitude of the output voltage, $V_b(t)$ is the feedback signal. The phase-angle ϕ depends on the angular position of the rotor axis, which is geared to the lead screw of the machine. If the rotor is rotated through 90 mechanical degrees for example, the output voltage of the rotor winding is shifted by 90

electrical degrees from the reference.

In the case where the rotor is continuously rotated with an angular velocity ω_o , the feedback signal is:

$$V_b(t) = V_r[\sin(\omega + \omega_o)t + \phi_o] \quad (2-2)$$

where ϕ , is the cumulative error value from $t=0$ until reaching the steady state. From Equation 2-2 one could observe the feedback has the form of a phase modulation wave.

The resolver is geared to make one revolution for each 0.1 inch of linear motion*. The practical accuracy of a resolver is one part in a thousand.

D. C. Tachogenerator: The velocity feedback device of the control loop is a DC Tachogenerator with the rating of 20.8V/1000 R.P.M. and the internal DC resistance, $R_b = 850\Omega$.

2.3 The Hewlett Packard 2100A Computer

"The Hewlett-Packard 2100 A Computer is a compact data processor featuring a powerful extended instruction set, plug-in interfaces, and modular software. Standard features include memory parity generation and checking, memory and

* The lead screws have a pitch of 0.25, which means that with a transmission ratio of 2.5:1 the resolver rotates once per 0.1 inch.

I/O protect for executive systems, extended arithmetic capacity, and power fail interrupt with automatic restart. Optional features include two-channel direct memory access, multiplexed input/output, a controller panel, and the I/O interfaces. The controller panel, which provides a minimum of controls and indicators, is available for applications where the full complement of controls and indicators provided on the operator panel is not necessary.

2.3.1 General Specifications

Basic Characteristics

- (a) 16-Bit word length; 17th bit memory parity checking.
- (b) 980 nanosecond cycle time.
- (c) A memory size of 16384.
- (d) 1024 - word page size.

Processor

- (a) 80 basic instructions, including extended arithmetic.
- (b) Two accumulators, addressable as memory locations.
- (c) Six working registers, may be selected for display and instant modification (A, B, T, P, M, S)
- (d) Illuminated control pushbuttons allow simultaneous display and control of internal features.
- (e) All instructions fully executed in 1.96 microseconds except ISZ and extended arithmetic (2.94 to 16.7 microseconds).

(f) 980 nanoseconds added for each level of indirect addressing.

Input/Output System

(a) 14 internal I/O channels, externally expandable to 45.

(b) All channels buffered and bi-directional.

(c) Multilevel priority interrupt for device servicing.

(d) Peripherals interfaced simply with plug-in cards.

(e) General purpose interface cards available.

A further description of the main features of the computer is given in Appendix A. However, since an understanding of the interrupt system of the computer is essential to gain an insight into the CNC/AC system, it will be explained in this chapter.

2.3.2 Interrupt System

The computer interrupt system has 60 distinct interrupt levels. Each level has a unique priority assigned to it, and is associated with a numerically corresponding interrupt location in the core memory.

As an example of the simplicity of this system, a service request from I/O channel 13 will cause an interrupt to core location 00013. The request for service will be granted on a priority basis higher than channel 14 but lower

than channel 12. Thus a transfer in progress via channel 14 would be suspended to let channel 13 proceed, but a transfer via channel 12 could not be interrupted by channel 13.

Under programme control, any device may be selectively enabled or disabled, thus switching the device in or out of the interrupt structure. In addition the entire interrupt system may be enabled or disabled under programme control using a single instruction (except powerfail and parity error interrupts)

The interrupt locations (octal 00010 through 00077) are available to I/O devices. This represents a total of 56 decimal locations, one for each of 56 I/O channels. In typical input/output operations, the computer issues a programmed command (e.g., set control/clear flag instruction STC,C) to one or more external devices, causing these devices to begin their read or write operation. Each device will put data into (input) or take data from (output) the input/output buffer on each individual interface card. During this time, the computer may continue running a programme or may be programmed into a waiting loop to wait for a specific device. On completion of the read or write operation, each device returns an operation completed (flag) to the computer. The flags are passed through a priority network which allows only one device to be serviced regardless of the number of flags simultaneously present. The flag with the highest priority generates an interrupt signal at the end of the current machine cycle.

A set flag flip-flop inhibits all interrupt requests below it on the priority string (provided that the control flip-flop is also set). Once the flag flip-flop is cleared the next lower device can then interrupt. A service subroutine for any device can be interrupted only by a higher priority device; then, after the higher priority device is serviced, the interrupted subroutine may continue. In this way, it is possible for several service subroutines to be in a state of interruption at one time; each will be permitted to continue when the higher priority device is serviced. All service subroutines normally end with JMP indirect instruction to return the computer to the point of interrupt.

When a device is ready to be serviced, it causes its interface to request an interrupt so that the computer will interrupt the current programme and service the device. Since many device interfaces will be requesting service at random times, it is necessary to establish an orderly sequence for granting interrupts. Secondly, it is desirable that high speed devices should not have to wait for low speed device transfers.

Both of these requirements are met by a series-linked priority structure illustrated in a simplified form in Figure(2.3). The bold line, representing a priority enabling signal, is routed in series through each card which is capable of causing an interrupt. The card may not interrupt unless this enabling signal is present at its input.

Each device (or other interrupt function) can break the enabling line when it request an interrupt. If two devices simultaneously request an interrupt, obviously the device with the lowest select code number will be the first one which can interrupt, since it has broken the enable line for the higher select code. The other device cannot begin its service routine until the first device is finished, however a still higherpriority device (lower select code) may interrupt the service routine of the first device.

Figure 2.4 illustrates a hypothetical case in which several devices require servicing by interrupting a CPU programme. Both simulataneous and time-separated interrupt requests are considered.

Assume that the computer is running a CPU programme when an interrupt from I/O channel 12 occurs (at reference time t_1). A JSB instruction in the interrupt location for select code 12 causes a programme jump to the service routine for the channel 12 device. The JSB instruction automatically saves the return address (in a location which the programmer must reserve in his routine) for a later return to the CPU programme.

The routine for channel 12 is not completed when several other devices request service (set flag). First, channels 13 and 14 request simulataneously at t_2 , however neither has priority over channel 12, so their flags are ignored and channel 12 continues its transfer. But at t_3 ,

a higher priority device on channel 10 request service. This request interrupts the channel 12 transfer and causes the channel 10 transfer to begin. The JSB instruction saves the return address for return to the channel 12 routine.

During the channel 10 transfer, device 11 sets the channel 11 flag (t4). Since it has lower priority than channel 10, device 11 must wait until the end of the channel 10 routine. And since Channel 10, when it ends, contains a return address to the channel 12 routine, programme control temporarily returns to channel 12 (even though the waiting channel 11 has higher priority). The JSB, I instruction used for the return inhibits all interrupts until fully executed (plus one phase of the next instruction). At the end of this short interval, the channel 11 interrupt request is granted.

When channel 11 has finished its routine, it returns control to channel 12, which at last has sufficient priority to complete its routine. Since channel 12 has been saving a return address in the main CPU programme, it returns control to this point.

The two waiting interrupt requests from channels 13 and 14 are now enabled. Since channel 13 has higher priority, it goes first. At the end of its routine, it temporarily returns to the CPU programme. Then the lowest priority channel 14, interrupts and completes its transfer. Finally, control is returned to the CPU programme, which continues processing." [6]

2.4 System Operation

The NC, AC programmes and the DATA are stored in the relevant memory locations of the computer. The operation of the system is initiated by addressing the memory location at which the NC programme starts. The computer control resides on the NC programme until it completely executes the whole programme and then it acts on the AC programme. Meanwhile, interrupt pulses are received from the TBG at a fixed frequency (5kHz.). When the computer control is on the AC programme as soon as a interrupt pulse is received, the computer control jumps from the AC programme to the NC programme and after it executes the latter, the control is transferred to the location in the AC programme at which it was interrupted. In this manner with the aid of the interrupt pulses the computer is able to run on both programmes. During this process data required for the NC programme is taken from the data available in the relevant memory locations by the computer. The AC programme receives its information from the transducers through the Analog-to-Digital Processor (details Chapter 3 and 5) and calculates an optimal feed-rate according to a prescribed strategy (see Chapter 8). This new feed-rate is used by the NC Programme in order to drive the servomotors through the Auxiliary Controller as explained in Chapter 6.

CHAPTER 3

THE SENSED VARIABLES

The identification portion of the adaptive control system has been developed to receive three sensed variables.

The sensed variables are:

(a) the torque on the cutter

(b) the horizontal forces F_v and F_u acting on the cutter, the resultant of these forces being F .

However, the cutter deflection, or the armature current of the servometers could also be found if required. As outlined above, the force F is the resultant of the two forces F_v and F_u , which are 90 degrees phase shifted from each other. These two force vectors rotate with the spindle and are monitored by the dynamometer. The sensing elements in the dynamometer are strain gauges. The values of the forces F_v and F_u are sensed and passed into the computer via the Analog-to-Digital Processor. The adaptive control programme in the computer accepts these values and calculates the resultant force F acting on the cutter where

$$F = \sqrt{F_v^2 + F_u^2} \quad (3-1)$$

The maximum allowable value $F_{max.}$ of the resultant force for a particular milling operation depends on the strength of the cutter material, the geometry of the cutter, the forces, the table and saddle of the milling machine could accommodate

without overloading the servomotors. This value of $F_{max.}$ is used as the constraint limit for the force on the cutter. During the milling process the adaptive control system ensures that the prevailing value F in the work zone does not exceed the value of $F_{max.}$ In the case of the torque acting on the cutter, the maximum allowable torque $T_{max.}$ depends on the strength of the cutter in torsion and the torque the spindle could take at the particular r.p.m. The value of $T_{max.}$ is used as the constraint limit for the torque on the cutter, and the adaptive control system maintains the value of the torque on the cutter below $T_{max.}$ It should be mentioned here that the strategy used in the adaptive control software utilizes the smaller error of $(F_{max.}-F)$ and $(T_{max.}-T)$ to calculate the new feed rate. It is the duty of the part programmer to calculate the values of $F_{max.}$ and $T_{max.}$ for each cutter he intends using and the respective spindle speed.

These values are then stored in the appropriate memory location of the computer to be used by the adaptive control software. It is evident at this stage that these two variables, F and T , are functions of the material strength, length and diameter of the cutter. But on the other hand, the maximum value of F and T are limited by the maximum force and torque which are available from the servomotors and the spindle motor respectively. Therefore, in each case the maximum value of the variable has to be checked from these two points of view and the smaller value of the two

taken as the maximum allowable value for a particular case. The maximum force and torque which are available from the motors are not dependent on the cutter.

In the analysis to follow it is assumed that the failure criteria of the cutter is due to either pure bending or torsion and the cutter tends to fail across its cross-section. However, failure could originate due to pure shear (for smaller cutters) or the individual teeth of the cutter breaking. It is assumed that the resultant force acts at the end of the cutter and the cutter is considered as a circular cantilever built-in at its other end (spindle end). The following equation gives the stress in the fibres at a distance y from the centre of the cutter, maximum stress distribution occurring at the outer fibers.

$$\sigma = \frac{My}{I} \quad (3-2)$$

Where σ - tensile stress in the fibers, distance y from the center of the cutter, M - bending moment due to the force F at this cross-section, y - distance of the fibers from the center of the cross-section, i.e. $y = \frac{d}{2}$ for maximum stress concentration, I - the moment of inertia of the cross-section.

Eq. (3-2) may be written as:

$$\sigma = \frac{F \ell d}{2I} \quad (3-3)$$

Where ℓ - is the length of the cutter. $I = \frac{\pi d^4}{64}$

for a circular cross-section. For the safety of the cutter

$\sigma < \sigma_{\max.}$, where $\sigma_{\max.}$ is the permissible tensile stress of the material of the cutter. Hence,

$$F_{\max.} < \frac{\sigma_{\max.} \pi d^3}{32l} \quad (3-4)$$

We see that $F_{\max.}$ depends on the maximum permissible tensile strength of the cutter material, the diameter and length of the cutter. To analyze the stress due to the torque on the cutter, the cutter was assumed to be a solid round bar built-in at one end. The following equation gives the stress due to torsion:

$$\tau = \frac{Td}{2J} \quad (3-5)$$

Where τ - is the shear stress, T - torque, d - diameter of the cutter, J - the polar moment-of-inertia, which is $\frac{\pi d^4}{32}$ in our case. For the safety of the cutter in torsion we have,

$$T_{\max.} < \frac{\pi \tau_{\max.} d^3}{16} \quad (3-6)$$

Where $\tau_{\max.}$ is the permissible shear stress for the material of the cutter.

The values of F and T were calculated for cutters of diameter $1/4$, $3/8$, $1/2$, $5/8$, $3/4$, and 1 inches and lengths varying from 1 inch to 6 inches (in steps of one inch) respectively where the material of the cutter had $\sigma_{\max.} = 60$ T/sq." and $\tau_{\max.} = 40$ T/sq.". From the values thus obtained, Figure 3.1 and 3.2 were arrived at. Figure 3.1 indicates the variation of F vs. l (the length of cutter) for various diameters of the cutter, while Figure 3.2 illustrates the variation of T vs. D (the diameter of the cutter).

In determining the allowable forces on the table and saddle, it was assumed that the energy supplied by the servomotors was utilized in overcoming the frictional forces and the force on the table (saddle) due to the cutter work-piece interaction. The following formula was used for determining the force on the table (saddle).

$$T_m \times \omega \times n = F \times v + T_F \times \omega \times n \quad (3-7)$$

(Energy supplied by motor per second) (Energy consumed by F per second) (Energy consumed by frictional torque per/sec.)

Where, T_m - motor torque, ω - speed of motor shaft, n - efficiency of the gear box (transmission), F - linear force in the table (saddle), v - linear velocity of the table (saddle) T_F - frictional torque on the system (measured from the motor shaft). The torque required to overcome the static friction on each axis of the milling machine was determined experimentally. A simple lever arm was fastened to each leadscrew at the motor end and weights added to a cradle on the end of the lever until the leadscrew began to rotate. The torque was the product of the total weight on the lever and the length of the arm. The frictional torques for the x-axis and y-axis were found to be 5.66 lb. in. and 13.03 lb. in. respectively. n was assumed to be 90%. The linear motion of the table (x-axis) was found to be 0.106" per revolution of the x-axis servomotor and the linear motion of the saddle was 0.090" per revolution of the y-axis servomotor. Using this data in equation (3-7) the

permissible forces on the table and saddle was determined as 2570 lbf and 2560 lbf respectively. For simplicity, the smallest of the two values i.e. 2560 lbf was taken as the limiting value for F as far as the table and saddle was concerned. This value is represented in Figure 3.1 by a straight line running parallel to the horizontal axis (length of cutter) through the point 2560 lbf on the force axis.

For the determination of the allowable torque on the spindle, the following equation was used:

$$P = \frac{T\Omega}{33000} \quad (3-8)$$

Where P - the power of the spindle motor in hp, T - the torque on the spindle, Ω - r.p.m. of the spindle in rad./min. The value of this torque for different spindle speeds is represented in Figure 3.2 by straight lines running parallel to the cutter diameter axis.

For a particular milling operation the part programmer first decides on the dimensions of the cutter to be used and the spindle speed. Then, using Figure 3.1, checks the maximum force the cutter could withstand and compares this with the maximum table (saddle) force (which is also given in Figure 3.1) and takes the lower value for F_{\max} . Next, using Figure 3.2, he obtains the maximum torque the cutter could take without breakage and compares this value with allowable torque the spindle could take at the particular spindle speed and uses the lower value

of these two as T_{\max} . This ensures that the milling operation will be optimized by the adaptive control system without overloading the motors or breaking the cutter.

CHAPTER 4

THE DYNAMOMETER DESCRIPTION

Figures (4.1), (4.2) and (4.3) illustrate the dynamometer used in this system. This dynamometer has been designed by the Metal Cutting Research group. The horizontal forces F_u and F_v and the torque T acting on the cutter are sensed by the strain gauges located on the dynamometer. These values of the forces and torque are next transmitted to the computer via the adaptive control hardware for the optimization of the milling process.

The dynamometer consists of:

(a) Spindle plate, which secures the main body of the dynamometer on to the spindle.

(b) A flexure plate, which is fixed on to the spindle plate and the housing of the dynamometer. The purpose of this member is to ensure that the measuring zone of the dynamometer takes almost the full values of the forces and torque which are to be measured. This plate consists of four spokes with a thin slot for torsional flexibility. The thick middle portion provides radial stiffness. The plate has been almost cut through (except for the four small pieces of metal remaining) on two levels. This provides angular flexibility about the x and y axes.

(c) The housing of the dynamometer, which is bolted on to the spindle plate on the top and to the rim plate at the bottom. This serves as a support for the rim plate and the torque ring which is suspended by leaf springs from it.

(d) The collet holder, which is centrally located in the dynamometer, secures the collet to the dynamometer. The collet holder is bolted on to the flexure plate.

(e) The stone, which is located inside the collet enables the collet to be drawn into the collet holder.

(f) The pin, this secures the stone onto the collet.

(g) The M-elements, these consists of three legs, the the outer two are bolted on to the collet holder while the inner leg is bolted on to the rim plate. The strain gauges used to measure the radial forces are fixed on the outer legs of the M-elements.

(h) The leaf springs, are used to suspend the torque ring from the housing.

(i) The rim plate, is fixed onto the inner leg of the M-elements and the housing of the dynamometer.

(j) The torque ring, is connected onto the housing by the leaf springs and retains the strain gauges for measuring the torque.

(k) The torque plate, which is bolted on to the collet holder, transmits the torque on the cutter to the torque ring.

(l) The "L members " are located on the torque plate. (Their location on the torque plate could be varied) and transmits the torque on to the limbs of the torque ring. These elements have been designed with a very thin cross-section to ensure flexibility under radial forces.

The measurement of F_u and F_v by the dynamometer and the subsequent calculation of the resultant force F by the adaptive control programme enables the system to ascertain the resultant horizontal force acting on the cutter. Forces F_u and F_v act at some point along the length of the cutter. The point of application of these forces depend on various factors such as: depth of cut, geometry of the cutter, cutter deflection etc. However, to simplify the design it is assumed at this stage that these forces act at the end of the cutter. Due to the radial forces acting on the cutter, strain is induced on the outer legs of the M-elements. If tensile forces are induced on a particular M-element then compressive forces will be induced on the M-element which is located directly opposite to the latter. There are four M-elements located at 90° to each other as such two of them will monitor the effects due to F_u while the other two would monitor the effects due to F_v . The collet holder on to which the M-elements (outer legs) and the cutter are secured is fixed on to the flexure plate.

As mentioned earlier the flexure plate has been designed so that it is angularly flexible (i.e. flexible in bending) about the x and y axes. This characteristic ensures for all practical purposes that the collet holder is pin-jointed on to the spindle. Which means that it is under the effect of pure forces. Therefore, knowing the forces acting on the M-elements, the distance of the M-elements from the flexure plate and the distance of the end

of the cutter from the flexure plate, F_u and F_v could be calculated by taking moments about the axes parallel to the x and y axes through the flexure plate. It was mentioned in the first paragraph that the inner legs of the M-element are connected onto the rim plate. The rim plate is not directly in contact with the collet holder, as such when the collet holder deflects the strain induced in the M-elements will be due to the absolute value of F_v and F_u , this ensures that the absolute value of F_u and F_v could be determined by this system. This is one of the important features of the system. The torque ring has been designed in such a manner to take almost the full value of the torque acting on the cutter and further the M-elements have been designed with small slots to allow for torsional flexibility as such the effects due to the torque on the cutter are not monitored by the strain gauges on the M-elements which are intended only for measuring the effects due to the radial forces acting on the cutter. The strain gauges placed on the two limbs of the torque ring measure the torque.

Due to the torque acting on the cutter the collet holder experiences an angular deflection, which causes the torque plate to deflect horizontally about the centre of the cutter. This movement of the torque plate transmits a force on to the limbs of the torque ring via the "L members". This induces a strain in the limbs of the torque ring which is monitored by the strain gauges placed on them. Thus the value of the torque

acting on the cutter could be ascertained. There are two torque plates connected on to the collet holder transmitting two forces on the two limbs respectively of the torque ring. These forces are produced by the torque acting on the cutter. If these two forces do not constitute a couple, then a resultant force will be induced in the system and will be sensed by the strain gauges on the M-elements (which are supposed to monitor only the effects due to the radial forces). This will give erroneous values for the radial forces computed. To avoid this anomaly the leaf springs have been included in the design. When there is a difference between these two forces, the leaf spring would deflect (as they are radially flexible) until both the forces are equalized, thus eliminating a resultant - induced force on the system due to the torque on the cutter. The radial forces on the other hand have no effect on the torque measuring strain gauges as the above leaf springs which suspend the torque ring from the housing are flexible in the radial direction. Provision is made on the torque plate to vary the distance of the L members from the centre of the cutter to obtain reasonable sensitivity with various diameter cutters used.

It is evident from the above discussion that the dynamometer consists of two distinct zones. The first being the zone where the measurements of the forces and torque are effected. This zone is situated at the lower region of the dynamometer where the various strain gauges are positioned. This measuring zone has been designed so that the force measuring strain gauges

would measure only the effects due to the radial forces while the torque measuring strain gauges would measure only the effects due to the torque. While the other zone being where the flexure plate is situated, at the upper end of the dynamometer. This zone ensures that the total values of the forces and torque on the cutter are experienced by the measuring zone. This provides that the total value of F_u , F_v and T are measured. Further this zone contributes the necessary radial stiffness to withstand the radial forces on the tool to avoid chatter conditions and excessive tool deflection.

CHAPTER 5

THE ADAPTIVE-CONTROL HARDWARE

As was mentioned in Chapter 2, the identification process contains a piece of hardware which consists of amplification units and analog-to-digital converters. A detailed description of the hardware will be given in this chapter.

The hardware has been designed to sense three variables at the same instant. Normally, the sensed variables would be two horizontal cutting forces 90° phase-shifted from each other and the torque. However, the hardware is capable of accepting any other variables (three).

The two forces and the torque are measured through the strain gauges located on the dynamometer. The strain gauges are connected in the form of electronic bridges, one of which is shown in Figure (5.1). The output from the bridges is connected to an amplifier which has an output range 0 to $\pm 10V$. This is the voltage range acceptable by the A/D converters used in this system.

Figure (5.2) describes the entire block diagram of the hardware portion of the identification process.

The Analog-to-Digital Processor (ADP) contains three identical A/D converters, a digital scanner, a code generator and a status generator. The A/D converters ADC-10Z are produced by Analog Devices Inc. and have the following features:

The ADC-10Z is a 10 bit converter and has a maximum relative accuracy error of $\pm 1/2$ LSB. All of its logic inputs and outputs are fully TTL/DTL compatible.

The ADC-10Z is designed for flexibility and ease of use and contains an internal temperature-compensated precision voltage reference, eliminating the problem of supplying an external reference voltage. Any of four calibrated input ranges (two unipolar and two bipolar) can be selected with jumpers and connections to the module terminals. A direct input allows the input ranges to be set to any desired value by selecting the value of an external series resistor. An internal clock is provided in this system.

Binary output coding is used for unipolar operation, but the user selects either two's complement or offset binary coding when operating in the bipolar mode. The two codes differ only in that their MSB are in complementary states. The MSB output is used for offset and binary coding, (the MSB can also be used to give two's complement coding). STATUS, which indicates when the parallel output data is valid, and its complement, $\overline{\text{STATUS}}$ are both available.

There are four fixed input voltage ranges available, 0 to +5V, $\pm 5V$, +10V and $\pm 10V$. As shown in Figure (5.3) the timing diagram, the leading edge ("0" to "1" transition) of the convert command pulse sets the STATUS and MSB outputs to the "1" state, and the output bits 2-10 to "0".

The conversion begins on the trailing edge of the conversion command pulse with the starting of the internal clock. The conversion command pulse is generated by the data reading routine of the AC programme. During the conversion process bit decisions are made on successive "1" to "0" clock pulse transitions with the MSB decision occurring first. At the completion of the conversion, the STATUS output returns to zero, signalling that the parallel output data is valid. The STATUS output of each converter is fed into the status generator. When all the three status outputs of the three A/D converters are fed to the status generator, the latter produces a pulse which is used to set the flag of the ADC interface board of the computer (i.e. channel No. 12). This signals to the Data Reading Routine that the entire data conversion has been accomplished and the input data is available.

The input to the AC programme consists of 10-bit data plus 3-bit code where the latter is generated by the code generator and used to identify the A/D channel. Each of the three A/D converters has its own code. The Digital Scanner picks up only one of the A/D outputs and adds to it the approximate code and sends it to the computer. The same pulse which is used as a conversion command is used to change the code as well which means, that in every cycle only one of the A/D outputs is fed into the computer. The A/D outputs are selected successively by the Digital Scanner and their approximate code word is combined with the respective output. In the Data Reading Routine the data from each A/D converter is identified by means of its code which

means that the input is decoded and then stored in the memory for further use. The data reading is done for a period of 10 m. sec. and then the programme is switched into the Policy Routine. This implies that the latter will use the last data picked up from the converters.

The Analog-to-Digital Processor is connected to the computer through an interface board located in slot No. 12 (OCT.) The same board is used by the ADC of the Fourier Analyzer. The plug of the board has to be connected to the ADP when the adaptive control system is used.

CHAPTER 6

THE CNC SYSTEM

6.1 System Description

The block diagram of the entire system is shown in Figure 2.2. One could observe that the system consists of the following units:

(a) The milling machine with the servodrives, resolvers, tachogenerators and the transducers for the adaptive control system. The servodrive unit with the resolvers and the tachogenerators were purchased from General Electric as a package.

(b) The Controller which functions as an interface between the computer and the milling machine. This unit consists of three identical control boards (which apply the position control loop to each axis of motion), a time base generator (TBG) and a manual control. The TBG supplies interrupt pulses to the computer, two waves 90° phase shifted to the stators of the resolvers and provides clock pulses to the control boards in order to synchronize the entire system.

(c) The mini-computer which takes the place of the conventional NC controller. Details on the computer have been given in Chapter 2 and Appendix A.

(d) A tape reader which is used for reading the NC and AC programmes and the NC data tape.

The computer handles two programmes viz., the NC and the AC programmes simultaneously. Normally, the computer works on the AC programme and as an interrupt pulse is accepted from the controller, the computer control jumps from the AC programme to the NC programme and the NC programme is executed. Once the computer has completed working on the NC programme the control is returned to the AC programme to the point at which it was interrupted. The AC programme calculates the optimal feed rate at which the machine has to be operated. The feed rate is fed automatically into the NC programme and used as a command for driving the servodrives. The AC programme will be explained in Chapter 8, while the NC programme will be explained in Section 6-2 of the present chapter.

The NC programme supplies (through a interface board located in the computer) command pulses to the control boards. There are altogether six channels between the computer and the controller: Three channels for command pulses, one for each axis and three channels to establish the direction of motion of each axis. Each command pulse represents an instruction to advance the appropriate axis of motion by one

tenth of a thousand of an inch. Every control board contains a modulator and a phase comparator as shown in Figure (2.2). The modulator supplies a square wave command signal and converts each pulse to a phase-shift of $1/1000$ of a cycle. The phase-comparator (discriminator) compares the falling edge of the feed back signal from the resolver. The modulator includes the following blocks: Rate flip-flops, variable count counter and divide by ten counter. The phase-comparator output is fed through the Servodrive unit into the DC motor. The loop is connected in such a way that the motor rotates in a direction to reduce the phase-error. The phase-comparator holds the position error. When sliding motion is confronted with a higher mechanical resistance, the position error increases. This causes a higher voltage to be supplied to the motor and the motor will overcome this resistance. The normal command pulses still being supplied. Normally, every control loop operates with a small position error which is used to drive the motor. The command signal is supplied to the phase-comparator from an electronic modulator located on the control board. For every command pulse which is received from the computer, the modulator converts it to $1/1000$ th of a cycle. The lagging or leading of this phase shift depends on the direction of motion which is also fed from the computer through the sign channels. The sign flip-flop is used for storing the sign signal which is accepted from the computer. The Schmitt trigger converts the

sign wave signal which is received from the rotor of the resolver into a square wave signal which is compared with the square wave command signal from the modulator.

6.2 The NC programme

The NC programme consists of two routines: an Initiator Routine which is activated each time a new data block* is required, and a Continuator Routine which is performed for each interrupt pulse. The general flow charts of both routines are given in Figure (6.1) and Figure (6.2) and their main functions are summarized below.

6.2.1 The Initiator Routine

The main functions of the Initiator Routine are:

1. Loading a new data block into the memory locations which the Continuator Routine is using.
2. To chose two of the three axes to be controlled. This is done according to the g instruction in the data block: namely, g17 is for the X-Y plane, g18 is for the Z-X plane and g19 is for the Y-Z plane (see Table 1). The details of the plane chosen is stored until these are replaced by a new plane-selection instruction.
3. Settling the computer digital output lines (which will be used in the current segment) according to the selected plane.

* A data block contains the numerical data required to machine a single segment of the workpiece.

4. Loading the position counters by the required incremental distances.
5. Choosing the mode of circular interpolation (C1 or C2) according to the direction - of - motion and the direction of producing the arc (clockwise or counter-clockwise).
6. Halt the computer after the completion of the instructions in a block which contains the "miscellaneous function" M30 (see Table 2). In this case the initial address of the first data block is loaded before halting the computer. When the RUN push-button is pressed the computer starts to machine a new workpiece.

6.2.2 The Continuator Routine

The positional control is performed by the position loop and software counters, which are situated in the Continuator Routine. The counters are loaded with the required incremental distance at the beginning of the segment. Each axis-of-motion is provided with a counter. Each time a command pulse is sent by the computer, the contents of the appropriate counter are reduced by one unit.

The functions of the Continuator Routine are performed in the following order:

1. When the interrupt occurs, the contents of the arithmetic-unit registers are stored. This stored data is required for the continuation of the AC programme.

2. If point-to-point (PTP, i.e., a rapid traverse) operation is required, high frequency command pulses are sent out. A subtraction of the appropriate counter by one unit is carried out for each command pulse. An automatic deceleration is accomplished when a counter is almost zero. The PTP deceleration is carried out in three steps and it is a part of the PTP routine. As long as the deceleration is not actuated the PTP routine continues to supply the high frequency pulses. Once this deceleration is actuated, the programme jumps either to step 5 or to step 6 below.

3. If interpolation is needed a FEED routine is actuated and determines, according to the F-word in the data block, whether to start interpolation, or to jump to step 6 below. A programmed deceleration can be added by using the function g09 in a data block (see Table 1).

4. The type of interpolation (linear, or one of the two circular interpolations) is chosen and a single cycle of a DDA interpolator is simulated. More details about a DDA interpolator are given later. If as a result an overflow pulse in one axis or both of the axes is generated, the appropriate software counter is decremented by one unit and a command pulse is sent to the proper output line.

5. When all counters are in a zero position, the programme jumps to the Initiator Routine. When the performance of the latter is finished, the programme jumps back to this point.

6. The registered values of the AC programme that were stored are returned to the arithmetic unit of the computer. The AC programme is continued from the point at which it was interrupted.

6.2.3: The Digital Differential Analyzer (DDA)

The principle of the DDA is based on the digital integrator which takes $\Sigma Y\Delta x$ as a close approximation of $\int Y dx$ (where Δx is a small increment in x). The digital integrator uses two registers Y and R as shown in Figure 6.3.

With the completion of each cycle, the quantity $Y\Delta x$ is added to the contents of the R register and ΔY incremented as desired. As the capacity of R is finite, after a certain number of cycles, the R register is filled to capacity and an overflow results. Δz represents this overflow, which is an incremental output in the form of a discrete pulse. If Δx is made to equal ± 1 or 0 , there is a simplification of the process as follows: The operation takes the form of an addition (if, $\Delta x = +1$), and a subtraction (if, $\Delta x = -1$), and no operation (if, $\Delta x = 0$). If $\Delta x = 1$, the contents of the Y register are added to that of the R register. Then, the process may be considered as a gating operation. Wherein, the arrival of a pulse Δx results in the gate being opened and the entire contents of the Y register added to the R register. If the overflow Δz are stored in another register, these will represent the most significant portion of the integral $\int y dx$, with the less significant portion remaining in the R register [9].

6.2.4: The Interpolator

"The Interpolator which is shown in Figure 6.4a consists of a simulation of two DDA integrators and is capable of linear and circular interpolation, in accordance with characters from the data tape.

The integrators function separately in linear interpolation. The data tape feeds the Y register a number proportional to the feed required from the axis. The pulses from $+\Delta z$ are fed directly to the control system and serve as commands to the motor, with integrator 1 controlling the X axis and 2 the Y axis. For a linear cut with path lengths a and b along the X and Y axes, respectively, these numbers must be fed to the respective registers.

In circular interpolation, Figure 6.4b the following equation is satisfied:

$$(X-R)^2 + Y^2 = R^2 \quad (6-1)$$

Where,

$$X = R(1 - \cos t)$$

and

$$Y = R \sin t \quad (6-2)$$

The integrator outputs are

$$dx = R \sin t \, dt = -d(R \cos t) \quad (6-3a)$$

and

$$dy = R \cos t \, dt = d(R \sin t) \quad (6-3b)$$

On feeding the initial conditions to the y registers, care should be taken that the integrator emitting $R \sin t$ receives $R(1 - \cos t)$ and vice versa. The Y register of integrator 1 belongs to the X-axis section on the command panel, hence it receives the number i which is the initial condition for $R(1 - \cos t)$ Figure 6.5. At the integrator output we have the value $R \sin t$, which by Equation (6-2) should be connected to the Y axis. Accordingly, integrator 2 (which belongs to that axis) will receive the value j , the initial condition for $R \sin t$. Its output, $R(1 - \cos t)$, is connected to the X axis.

The output of integrator 2 is connected to $-\Delta y$ of integrator 1, so as to secure sign reversal in Equation (6-3a); similarly, that of integrator 1 is connected to $+\Delta y$ of integrator 2.

The feed rate is set by a clock connected to $+\Delta t$ in the integrators. Three clocks are provided for a given work-piece, and selected as required by the F word of the data tape.

As an illustration, assume that a circular quadrant with radius r is to be produced, with initial conditions $i = r$, $b = 0$. The number r is fed to the y register of integrator 1 and connected to its R register in every cycle, so that the integrator 1 initially emits pulses at a high frequency. The reading y is gradually reduced by pulses entering $-\Delta y$, and the output frequency drops until $y = 0$.

By contrast, a zero value is connected initially to the R register of integrator 2, but its y register is gradually filled up with pulses from integrator 1, and the output frequency increases accordingly.

As the output of integrator 1, is connected to the Y axis and vice versa, and as the motor speed is directly proportional to the output frequency of the interpolator, the Y axis motor starts at a high speed and decelerates ($dY/dt = R \cos t$), while that of the X axis starts at zero speed and accelerates ($dx/dt = R \sin t$). Since the position of the axes is the integral of the motor's speeds, the movement will be according to Equation (6-2), so that the required circuit is obtained"[7]

6.3 Lost Motion

In order to check the lost motion in the X and Y axes of the milling machine, a two-dimensional programme (in the X and Y plane) was developed and the profile (shown in Figure 6.6) was produced on pexi-glass using a sharp pointer fixed on to the spindle. The spindle was not rotated during the X and Y motion. The profile was produced at two different feed rates viz., 10 inches per minute and 1 inch per minute. Three runs were made at the feed rate of 10 inches per minute to check repeatability. The following results were obtained by the above tests using a Tool maker's microscope:

Axis	Lost Motion at	Feed rate 10"/min, 3 runs	Feed rate 10"/min, 1 run	Feed rate 1"/min, 1 run
Y	B	.0040"	.0050"	.0050"
X	C	Not visible	Not visible	Not visible
Y	D	.0040"	.0040"	.0040"
X	E	Not visible	Not visible	Not visible
Y	F	.0040"	.0040"	.0040"

Overshoots: Of the order of .0015"

CHAPTER 7

THE NC DATA PROGRAMME (DP) AND DATA READING PROGRAMME (DRP)

7.1 General

One of the main differences between a conventional NC System and the CNC System is that in the latter case, the NC data tape is read only once for a particular manufacturing series. The data in the NC data tape is stored in known locations of the computer memory. The NC programme which is also stored in the computer memory has access to these memory locations and uses the required data to control the machine tool.

The NC data is arranged in accordance with the EIA standards, which will be explained later in this chapter. This data is punched on a tape called the data tape. A special programme denoted as the data reading programme (DRP) has been developed which enables the data tape acceptable to the computer and inserts the data into relevant memory locations. The flow chart of this programme will be explained in detail in Section 7.3 of this chapter.

7.2 NC Data Programming

Part programming and the preparation of the manuscript consists of a list of instructions which describe the detailed and precise step-by-step operation of a machine tool under

numerical control. A manuscript is illustrated in Figure (7.1). To prepare a manuscript for a particular part, the part programmer has to collect all data necessary to produce the part and arrange this in a standard format. This chart or list of instructions is termed the manuscript. Each horizontal line in the manuscript consists of a "block" of information and represents one complete instruction for a particular machine movement including the relevant machining instructions during this movement. This "block" of information is called a data block. A character is a collection of holes positioned on one line across the tape and represents either a number, letter or symbol on the manuscript. A group of characters is referred to as a word and initiates a specific action of the machine tool. Each data block ends with the end of block (EB) character. The EB character is not printed, but only punched, and is represented by the "Line Feed" code (which is punched immediately after the "Carriage return" code). The manuscript is typed with a teletype during which the tape is prepared simultaneously.

The data block contains the following words:

- n - sequential number of the block
- g - preparatory function, consisting of 2 digits
- x,y,z,i,j,k - dimension words
- f - feed rate code
- m - miscellaneous function

It should be mentioned here that the EIA standards use small letters to address the above words, while on the teletype, only capital letters are available. Figure (7.1) shows the order in which a data block must be arranged. At least one "0" will have to be typed in the position when a word is not needed in a particular block. The plus sign and the leading zero in the dimension words may be ignored. At least one "space" character must be inserted between every two words in a block. The last word in a block can be followed by a space character (s). The ninth word in a block is the feed rate code (f). Since this is necessary in every block (as the NC programme does not store the previous feed rate code), a data block contains at least 9 words, except in a point-to-point operation when rapid traverse is required. a data block can contain even three words.

A detailed description of the various words used in the manuscript will be given below:

1. Sequence Number, n.

Each block of the tape has a sequential number, the sequence number word is the first in the block. The sequence number is displayed in the display register of the computer (in a binary form) during the period the particular block instructions are performed. This enables the operator to check the performance of the machine with the programmed instructions.

2. Preparatory Function, g

The preparatory function prepares the NC programme to be ready to perform a specific mode of operation. Table 1 gives a brief explanation of the various preparatory functions used in this system. The g00 indicates a point-to-point operation, as such the g function must be programmed in every single block. A point-to-point operation could be performed only in the X-Y plane.

3. Dimension Words

The following dimension words are used in this system:

- (a) x,y,z - Distance dimension words for motion in the X-Y-Z axis respectively.
- (b) i,j,k - Circular dimension words. These determine the distance of the starting point of the arc from the center of the arc in directions parallel to the X,Y,Z axes respectively.

A maximum of two distance dimension words and two circular dimension words can be programmed in one block.

The dimension words are programmed in incremental form and should adhere to the following specifications:

- (i) Decimal points should not be used.
- (ii) Dimensions should be expressed in units of 1/10 thousands of an inch.
- (iii) Only negative signs must be programmed.
- (iv) The i,j,k words are programmed without algebraic signs.
- (v) Leading zeros need not be programmed.

- (vi) The maximum length of the path that can be programmed in one block is limited to 3.276 inches.

In the distance dimension words the sign indicates the direction of the axis of motion. The X-, Y-, and Z-axes of motion denote a right-hand co-ordinate system. Which means that if one stands in front of the machine then:

for a positive x command the workpiece will move to the right.

for a positive y command the workpiece will move IN, i.e. towards the machine.

for a positive z command the tool will move up.

4. Feed Function, f

The f word is used for the selection of a particular feed rate in contouring operations. In point-to-point operation (g00) a constant maximum feed rate of 158 ipm/axis is selected automatically. Feed rates are independent of spindle speed and are expressed in inches/min.

The feed rate word will range from 1 to 3000, and is equivalent to the required feed rate in ipm multiplied by 100. The feed rate has to be punched on the tape even though there may be no change from the previously programmed feed rate code. The maximum feed rate allowed in the Z-direction is only 15 ipm.

5. Miscellaneous Function, m

The miscellaneous function pertains to auxiliary information which does not relate to dimensional movement of the machine such as spindle command and other functions as explained in Table 2.

6. Circular Interpolation

The preparatory function is used to select the plane on which the circular interpolation has to be performed (i.e. g17, g18, g19) and to determine sense of motion (i.e. g02, g03) of the tool during circular interpolation. In circular interpolation, four dimension words are required per block. The first two dimension words (two of x, y, z) denotes the distance in the X and Y co-ordinates respectively of the final point of the arc from the initial point. The sign of these two numbers depend on the direction of motion in the two co-ordinate direction. The next two dimension words (two of i,j,k) denote the distance of the initial point of the arc from the center of the arc in the direction of the two selected co-ordinate axes. These dimension words are taken as positive always.

7.3 The Data Reading Programme (DRP)

The purpose of this programme is two fold:

- (a) To transfer the data in the data tape to relevant memory locations of the computer and store them at these locations.
- (b) To calculate the PATH or RADIUS (depending whether linear or circular interpolation respectively) for the particular block of data and store this value at the appropriate memory location. First the binary version of this programme is read into the computer then the data tape is placed at the photo reader and the origin of the DRP is addressed, then the computer starts

to read the data tape and the functions mentioned (a) and (b) are performed. The flow chart of the DRP is given at Figure (7.2) and the programme in Appendix C. It is evident from the flow chart that when LF (line feed) is encountered (i.e. the end of block character) the PATH or RADIUS is computed and stored. Further, when a slash is noticed, the computer understands that the end of the data tape has been reached and it stops reading the tape and halts displaying 75(OCT) in the display register. The computer reads a character at a time, but a number may consist of one or more characters. Therefore, before storing the number in the appropriate memory location, the number must be formed correctly with its digits in the correct position with respect to each other. For this purpose the logic in the block "Calculate Number N" in Figure (7.2) has been developed. Also whenever the computer encounters a negative number (which it would realize as soon as it sees the character for - in front of the number) it makes a note in one of its accumulators("B" register) and after it formulates the number (i.e. the absolute value of the number) using the logic in block "calculate number N", it adds the -ve sign for the number and stores it in the relevant memory location. This ensures that a distinction is made between positive and negative numbers when storing them in the memory locations. The calculation of PATH or RADIUS is necessary to determine the feed rates in the X and Y directions by the NC programme.

CHAPTER 8

THE ADAPTIC CONTROL PROGRAMME

The Adaptive Control Programme (AC) consists of two main routines: Data Reading Routine and the Policy Routine. The function of these routines are clear from their nomenclature. The Data Reading Routine receives the input (i.e. the data and the code) from the Analog-to-Digital Processor (see Chapter 5) and stores this in the appropriate memory locations. Approximately every 10 m.secs. the programme jumps from the Data Reading Routine to the AC Policy Routine. In the latter the calculations for the new feed rate are carried out, the purpose of this calculation is to modify the prevalent feed rate towards achieving an optimum feed rate consistent with the cutting conditions. A detailed explanation and relevant flow charts of the AC programme itself is given in Appendix B.

8.1 Data Reading Routine

Figure 8.1 illustrates the flow chart of the Data Reading Routine. Since the data is received through the same channel from three different sources, the first task of the Data Reading Routine is to decode the input immediately after reading it. This is achieved by removing the 3-bit code from the 13 - bit input word. The 3 - bit code is used to identify the source from which the

data was received (i.e. whether the data is T , F_u or F_v). The Data Reading Routine runs in a closed loop until the contents of a software counter, denoted as the Time Counter (TC) exceeds the value of 5000, an event which occurs approximately every 10 m. secs. Once the value of 5000 is exceeded the computer control is transferred to the AC Policy Routine. The time of 10 m. sec. was arrived at by assuming that the maximum spindle speed which would be used is 600 R.P.M. (for die-sinking purposes) and at least ten samples of measurements would be required per revolution of the spindle.

The details of the Data Reading Routine are given below: The execution of this routine begins by clearing the counter TC and the flag in the interface board of select code 12, the Digital-to-Analog Processor. Then the control bit is set in the interface of the select code 12 and the computer enters into a waiting loop until the flag is set by a pulse received from the Analog-to-Digital Processor, informing that the data conversion has been completed. Subsequently, the input is received and the control bit in the interface board 12 is cleared. The input thus received is decoded and the data is stored in the appropriate memory location after ascertaining the source from which the data has been received. At this stage an output pulse is sent and simultaneously the flag in the interface of select code 12 is cleared. This output pulse (which is between the STC and CLC commands) is utilized to :

(a) activate the three Analog-to-Digital converters in the Analog-to-Digital Processor in order to perform the conversion of the analog signals received from the Dynanometer.

(b) to advance the Digital Scanner by one step via the Code Generator.

It may be pointed at this stage that it takes $20 \mu \text{ sec.}$ for the data to be ready after conversion, this means that the data will be ready eleven instructions (since $1.96 \times 11 > 20 \mu \text{ sec.}$) after the conversion Command. Next the software counter is checked whether $TC > 5000$? If $TC < 5000$, control is transferred to L0 to receive a new input. The above logic has been developed and arranged so that sampling is done approximately every 5000 instructions. The counter TC counts infact the number of computer instructions executed. Since a single instruction is fully executed in $1.96 \mu \text{ sec.}$ it takes approximately 10 m. sec. to execute 5000 instructions. The TC Counter is present in both the NC and AC programmes. As and when a certain group of instructions is executed, the number of instructions in the particular group is added to the counter.

8.2 AC Policy Routine

On completion of the Data Reading Routine, computer control is transferred to the AC Policy Routine which is illustrated in Figure 8.2. This routine is divided into two main parts viz.,

an Error Calculator and a Feed Rate Calculator. In the Error Calculator Section of the AC Policy Routine, two errors (of the resultant force (F) and torque (T)) are evaluated and compared between themselves. The smaller of the two errors is taken as the critical error (E), for the continuation of the feed rate calculation. The errors are defined as a percentage of a desired value of the variables (F and T). The desired values are obtained using the methods enumerated in Chapter 3 (SensedVariables).

Once the smaller error has been determined, the strategy of the algorithm begins. Two separate strategies are used, depending on whether the error (E) is greater or smaller than zero. When E is smaller than zero, which means that the desired value of the variables (F or T) has been already exceeded and an urgent corrective action on the feed-rate is required (immediate reduction). While in the case when E is greater than zero, the feed rate will have to be increased in accordance with a suitable policy. The policy calculates the derivative of the feed rate with respect to time, which is then multiplied by Δt (Δt being the calculated time increment 10 m. sec.) and the result is added to the original feed rate i.e.

$$f_{t+\Delta t} = f_t + \frac{df}{dt} \Delta t \quad (8-1)$$

The new feed rate thus calculated is checked to ascertain whether it is greater or smaller than the maximum or minimum values of the feed rate respectively, allowable on the machine (the minimum limit may be exceeded for $E < 0$ strategy), and

a suitable correction made if necessary. Once the above limits have been checked and a new feed rate arrived at, the value of this feed rate is stored in the relevant memory location and is used by the NC programme.

The details of the AC Policy Routine is as follows: As mentioned earlier this routine consists of two parts viz., Error Calculator and the Feed Rate Calculator. The routine begins with the Error Calculator. Using the two force signals (F_u and F_v which are 90 degrees phase-shifted) obtained from the Dynanometer, first the resultant force (F) is calculated where F is given by:

$$F = \sqrt{F_u^2 + F_v^2} \quad (8-2)$$

Next, the fractional errors E_f and E_T of the force F and the torque T respectively, are calculated. Where E_f and E_T are given by:

$$E_f = \frac{F_m - F}{F_m} \quad (8-3)$$

$$E_T = \frac{T_m - T}{T_m} \quad (8-4)$$

Where, F_m - maximum permissible value of the force
 T_m - maximum permissible value of the torque.

As mentioned previously, the maximal permissible value of force F_m and torque T_m (obtained as explained in Chapter 3) are used for the above calculation. Of the two errors thus obtained, the smaller one is taken as the basis for the feed rate calculation.

The error E is checked whether it is greater or smaller than zero. Initially, when the AC process starts to operate the feed rate has to be increased. When the feed rate is not close to its optimal value, the rate of increase of the feed rate should be large, while only small feed rate changes are required when the value of the feed rate is close to its optimal value. Various strategies which fulfill this requirement could be applied. The one used here is a very simple strategy, a parabolic one given by the following equation:

$$\dot{f} = \dot{f}_{\max.} \times E^2 \quad (8-5)$$

Where a superior dot denotes a time derivative.

Hence, the new feed rate is given by:

$$f_{t+\Delta t} = f_t + \dot{f}_{\max.} \times E^2 \times TC \times T \quad (8-6)$$

$TC = 5000$ instructions

$T =$ Time to execute one instruction ($\approx 2 \mu$ sec.)

and $TC \times T = \Delta t$ (≈ 10 m.sec.)

In the case where E is smaller than zero, a more drastic correction of the feed rate has to be carried out. The strategy chosen in this instant was given by equation:

$$\dot{f} = \dot{f}_{\max.} \times (10E)^2 \quad (8-7)$$

Which is again a parabolic curve and a equation similar to Eq. (8-6) is used to calculate the new feed rate. The strategy curve (for both cases $E > 0$ and $E < 0$) is given in Figure (8.3). It is worth-while to comment here that in the parabolic curves pertaining to Eqs. 1 and 3 the acceleration is equal to zero for the values of E in the immediate neighbourhood of the point $E=0$

(i.e. at the immediate vicinity of the saddle point of the strategy curve). This is due to the fact that the computer functions with discrete values rather than with continuously varying functions. This phenomena creates a natural dead zone and consequently stabilizes the system. In order to arrive at more sophisticated strategies than the one presented by Eqs. (8-5) and (8-7) and illustrated in Figure 8.3 more experimental work will have to be carried out as a theoretical solution for this problem is not simple or even impossible as the relationship between the force and the feed rate are not known.

Next the new feed rate arrived at is checked whether it exceeds the maximum permissible feed rate ($F_{max.}$) of the machine, if so, $F_{max.}$ is used as the new feed rate. If the calculated feed rate is less than $F_{max.}$ it is checked whether it is smaller than the minimum feed rate (F_{min}) of the machine, if so, F_{min} is used as the new feed rate. If the calculated feed rate lies between $F_{max.}$ and $F_{min.}$, then this value is used as the new feed rate. The new feed rate thus obtained is stored in the relevant memory location to be used by the NC programme. Finally, the software counter is cleared and the computer control is transferred to the Data Reading Routine (at LO) to obtain a new input.

8.3 AC - NC System

The AC and the NC programmes are developed in such a manner as to operate as one system. The three factors which enable this particular mode of operation are:

- (1) The interrupt system of the computer.
- (2) The common software counter TC.
- (3) A common memory location for storing the feed rate which is accessible to both programmes.

The computer runs on the AC programme and whenever an interrupt pulse is received the computer control jumps (by a JSB instruction) to the NC programme and begins to execute the latter. Once the NC programme is completed the computer control is returned to the AC programme to the point where it was interrupted. The AC programme continues to be executed until the next interrupt occurs.

As was already mentioned above, the counter TC is common to both the NC and AC programmes. This counter is used for indirect measurement of real time, by counting the computer instructions which are executed. Each time a group of instructions are performed, the counter is advanced by the appropriate number. These groups of instructions can be either in the NC programme or in the AC programme, and they influence the counting in the same manner.

The current feed rate word is stored in a memory location which is common to both the NC and AC programmes. Each time that a feed rate word is required for the calculations, this

memory location is indirectly addressed. The memory location in our case was chosen to be OCT 20004, a memory location which is neither in the stored data area nor in the programme pages. By this method any modification of the feed rate by the AC programme is automatically inserted in the above location and subsequently used by the NC programme.

CHAPTER 9

CONCLUSION

The purpose of an adaptive control system is to increase the productivity of the milling process by maintaining the highest possible metal-removal rate consistent with milling acceptable parts and maintaining an economical tool life. This means that the desired operation is an optimal combination of maximum stock removal rates and minimum costs. Further, an adaptive control system operates in a time varying environment which cannot be predicted in advance.

In practice, operating the machine at the maximum feed value subject to the force or torque constraints (determined with respect to the strength of the cutter) does not guarantee that the process is operating to maximize the metal removal rate as is often reported.

To maximize the metal removal rate, an optimum balance must be achieved between the rate at which material is being removed from the workpiece and at the rate at which the tool is worn out.

Material removal from the workpiece is directly proportional to the value of the feed. The rate of tool wear is also directly related to feed. A faster rate of tool wear means that less work material can be machined with a given tool. Therefore, increasing the feed-rate above a certain level will

actually decrease the metal removal rate due to increased process down-time. The inference which could be drawn from the above analysis, is that metal removal rate can be maximized by properly balancing volume rate of metal removal against rate of tool wear.

This balance is not necessarily obtained by operating at the highest feed that is consistent with some force or torque constraint. It is evident from the above analysis that the identification function of the adaptive control system should be concerned, or should monitor as many process variables that would effect the metal-removal rate and tool life.

Therefore, for the system to operate successfully, process variables such as :Force and torque on the cutter, horsepower of the drive motors, spindle deflection, air gaps between the workpiece and cutter, temperature and vibration will have to be monitored and an index of performance evaluated and the control variable (feed in our case) should be adjusted to maximize this index without the violation of any **constraint limits**.

The index of performance is determined with respect to the optimal combination of the metal-removal rate and tool life. Once the identification function has determined the performance of the process next the decision function will have to decide how best the performance index could be increased. This depends on the strategy used in the system. The decision function should be capable of selecting a line of attack which will optimize the performance index at the shortest period of time without the violation of any constraint limits.

In our case, the strategy used is a parabolic one and there is no guarantee at this stage that the optimal performance index will be reached in the shortest period of time by using this strategy. Further research will have to be done to decide on the most suitable for the above system.

It is difficult (if not impossible) to measure on-line certain process variables such as temperature and tool wear. This requires certain off-line experimental data collection before the on-line operation of the system. The purpose of this experimental work is to establish the max. (or min,) values of the allowable constraint limits beyond which the operation of the process is forbidden. Any initial off-line experimentation that is required before the adaptive control system can be put into operation makes the system less attractive from the economic point of view. This is one of the problem areas of adaptive control machining.

In the absence of a sensing device which will measure a particular process variable, an indirect measuring scheme may be developed based on direct measurement of other process variables which are related to the particular process variable. If the above additional functions are incorporated in the system, the system could be considered as satisfactory for practical purposes.

It is noticed in the present system we have two feed back loops. The first is the conventional NC control loop. The purpose of this loop is to compare the commands with the feed back signals and eventually control the position and velocity of the slides and quill of the machine. Next, is the adaptive control loop which is used to optimize the milling process.. Finally, we could add a third loop. The purpose of this loop is to inspect the output from the machine (the finished workpiece) and evaluate the actual performance index and compare this with the performance index arrived at by the adaptive control system and depending on the outcome of the comparison, the adaptive control system is informed of the most suitable strategy to be used to arrive at the optimum. This inspection can be performed automatically on one workpiece while the next part is being made or it can be performed on one portion of the workpiece while the remainder is being finished. The signals received from the inspection system is used to evaluate the actual performance index. The inclusion of the above function in this system would increase the efficiency of the system tremendously.

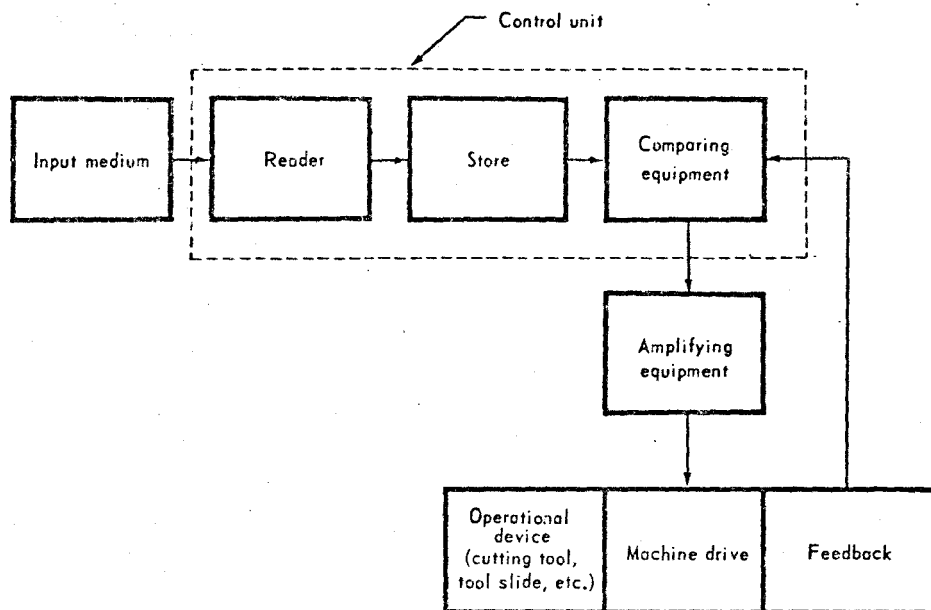


FIGURE 1.1: Block diagram for a numerically controlled machine tool, closed-loop system Ref(4)

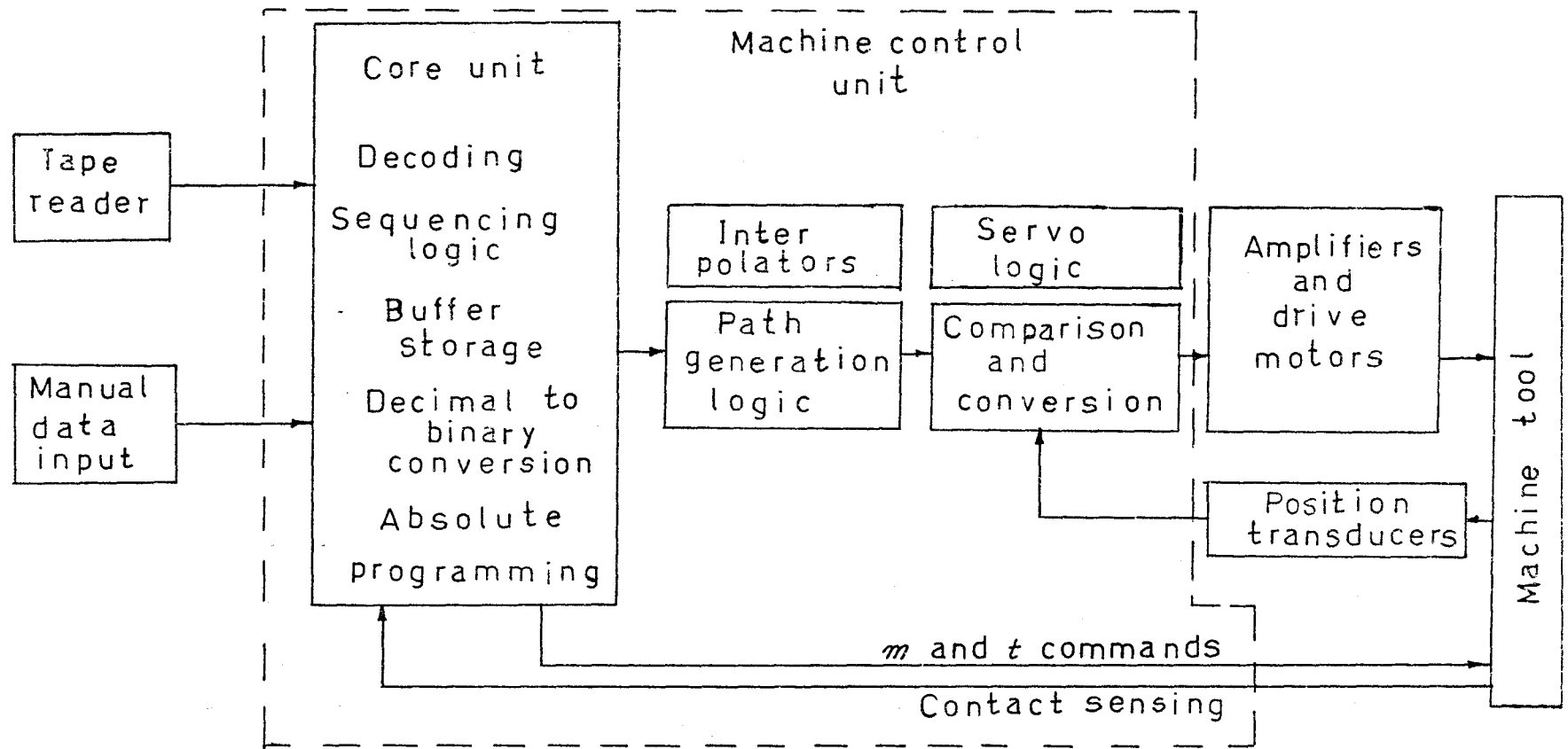


FIGURE 1.2: Conventional Hardwired NC Control System Ref(12)

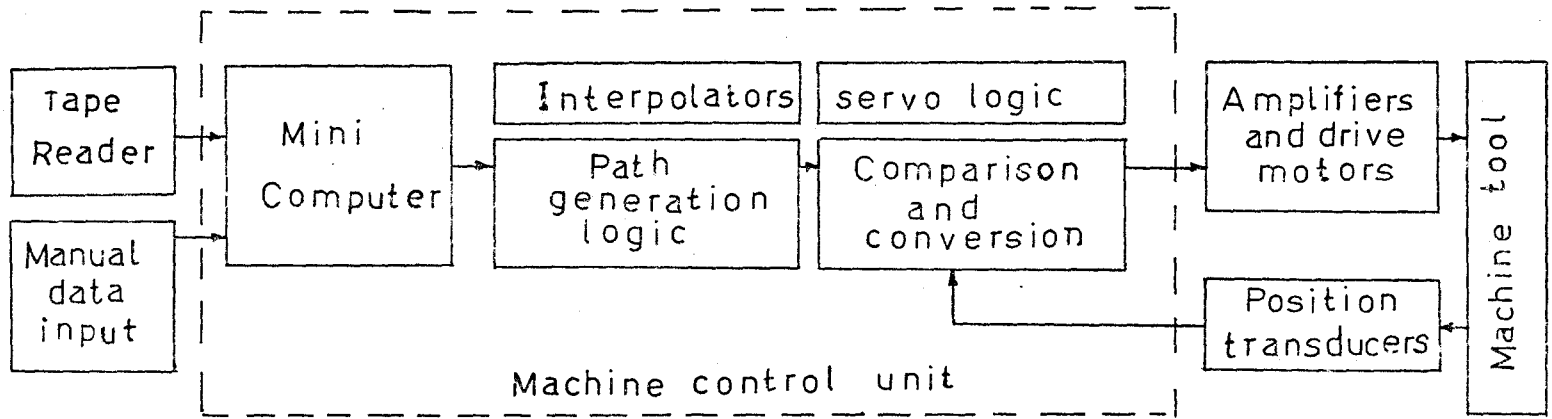


FIGURE 1.3: Hybrid Configuration Ref(12)

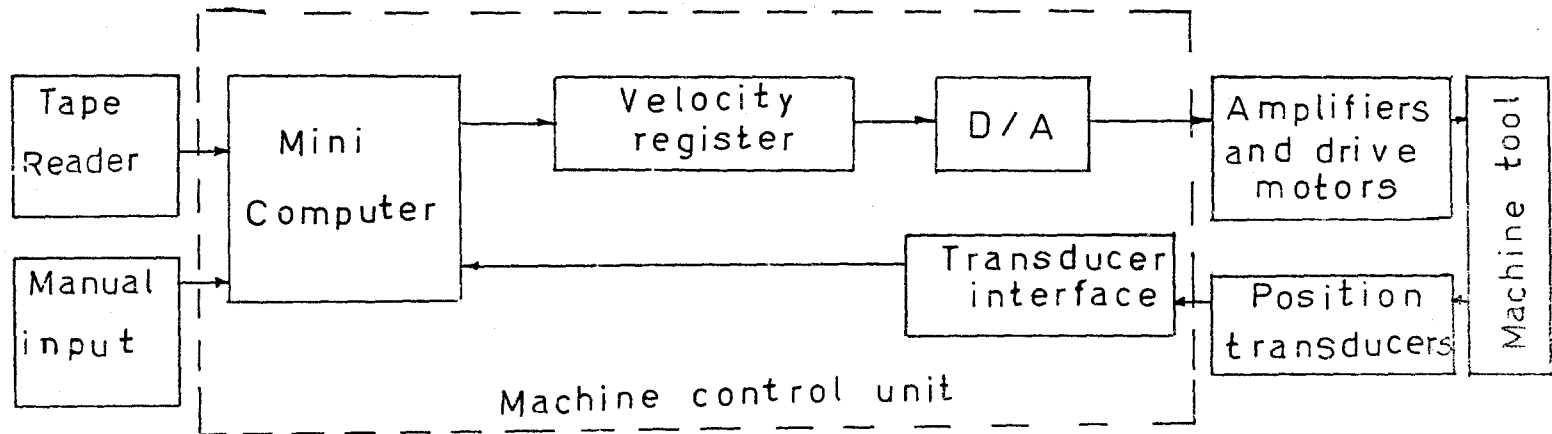


FIGURE 1.4: Software CNC Configuration Ref(12)

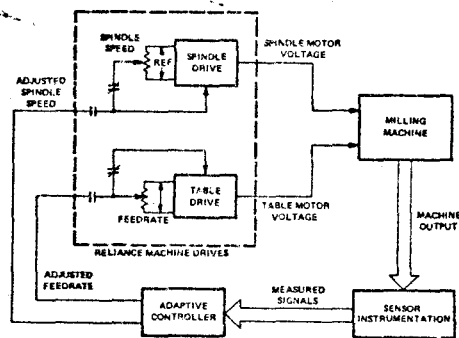
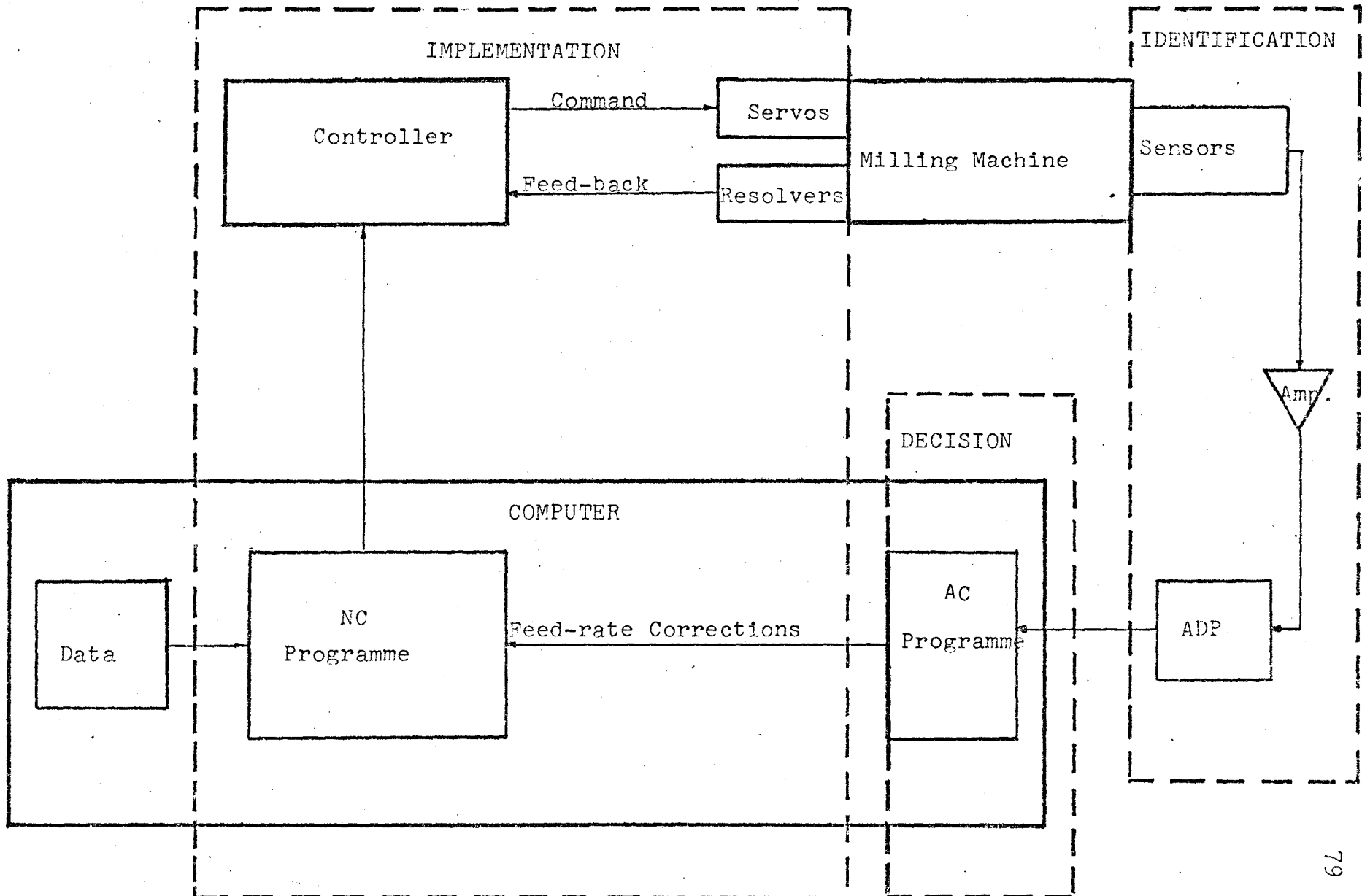


FIGURE 1.5: Adaptive control system block diagram
Ref(5)



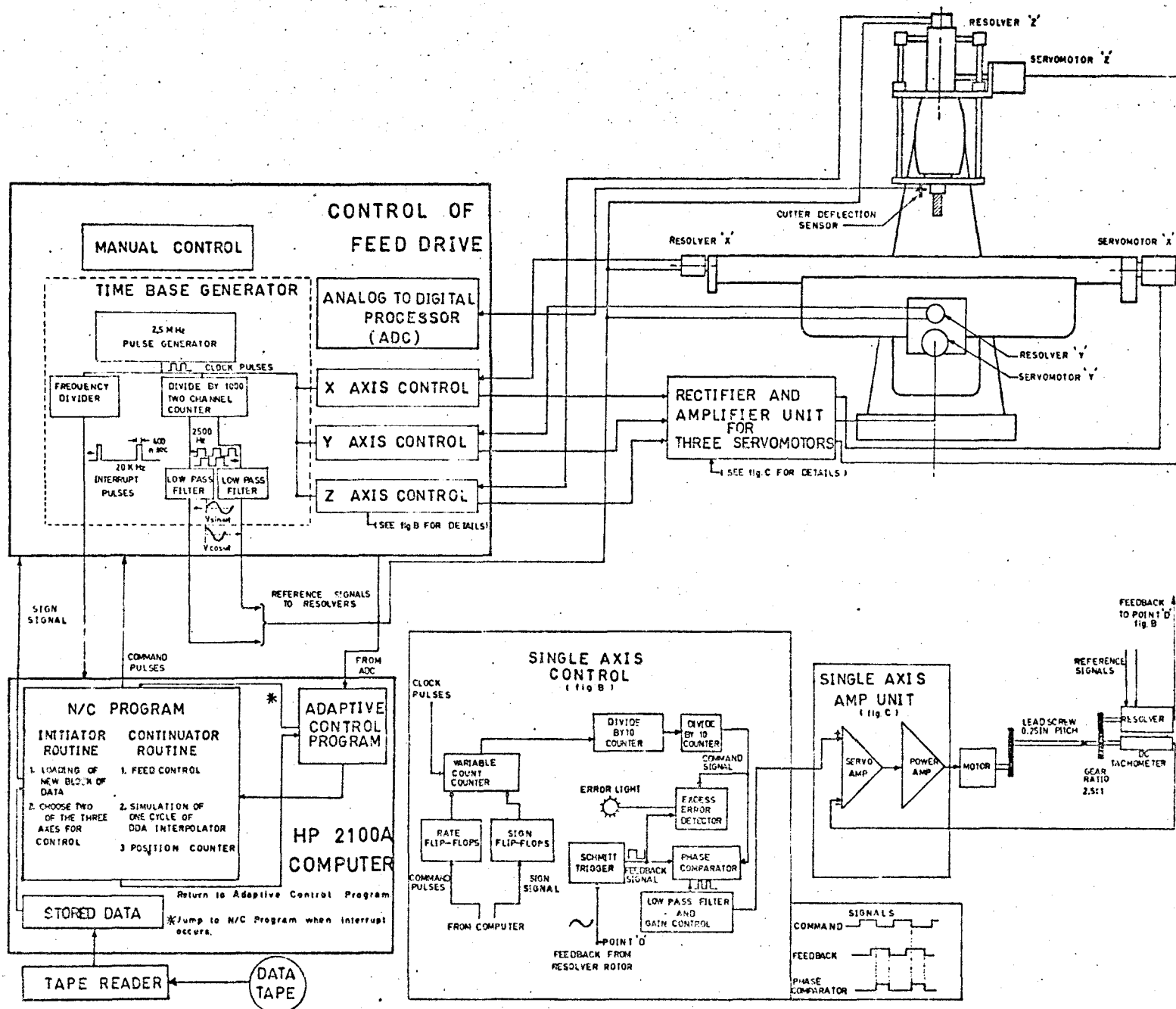


FIGURE 2.2: Block Diagram of the Milling Machine, Computer and Controller

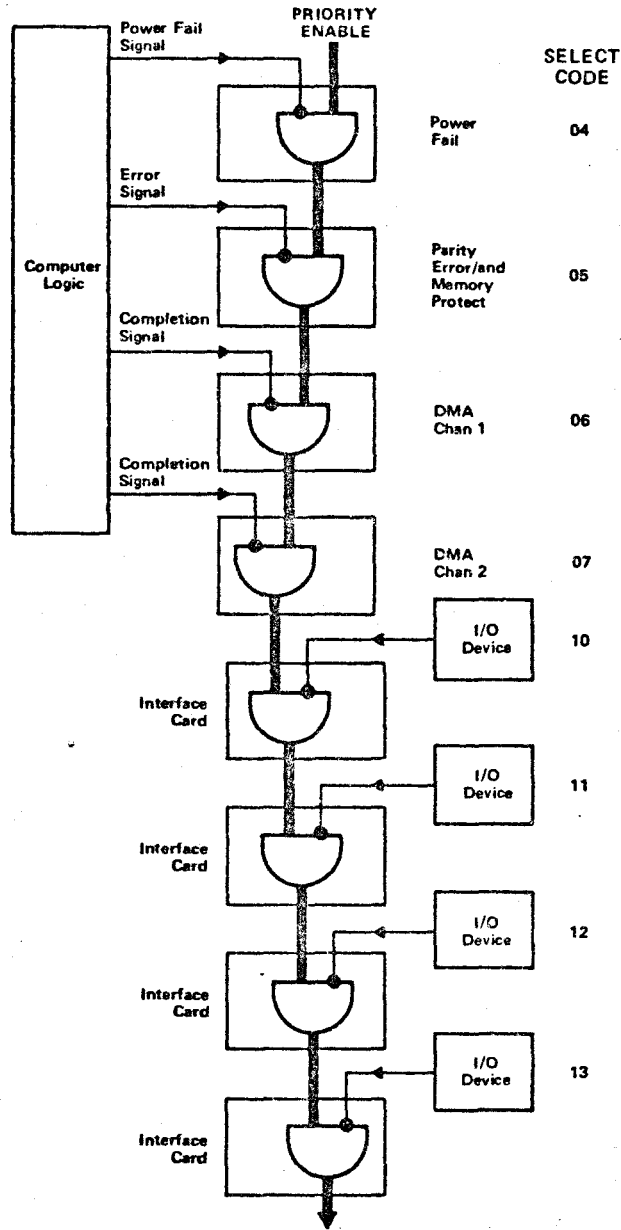


FIGURE 2.3: Priority Linkage Ref(6)

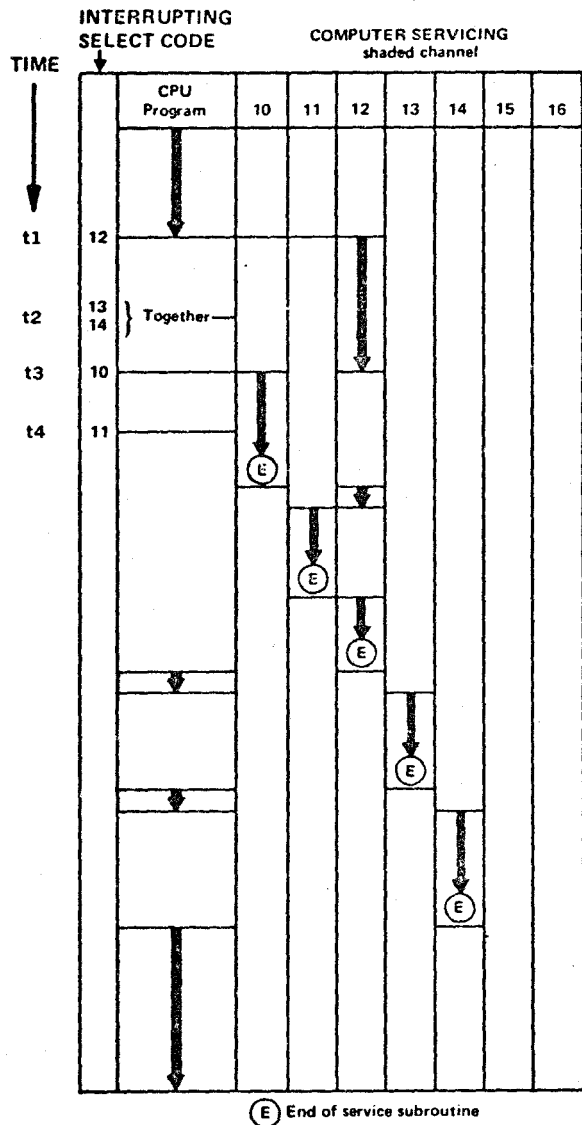


FIGURE 2.4: Interrupt Sequences Ref(6)

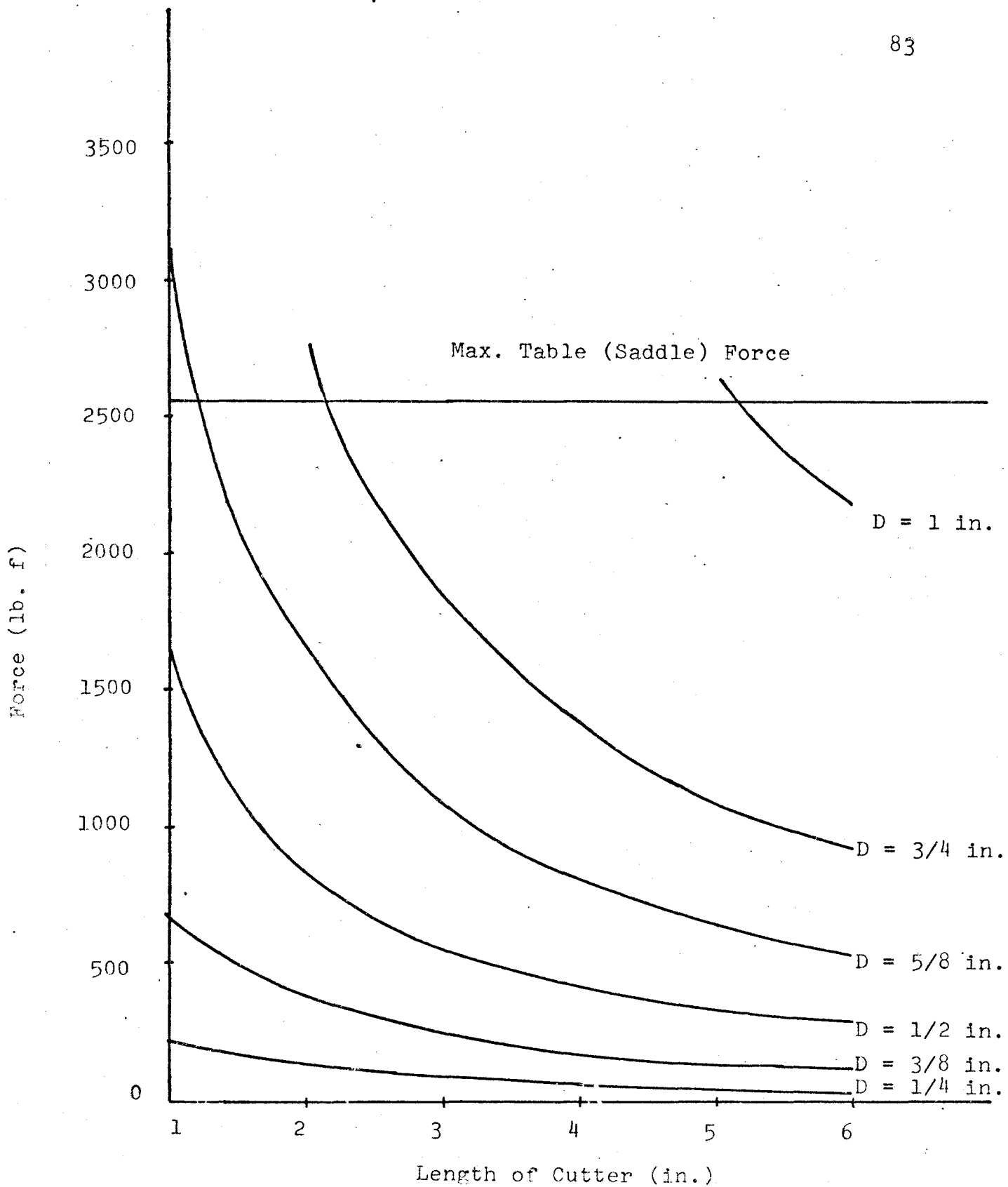


FIGURE 3.1: Graph of Force vs. Length of Cutter

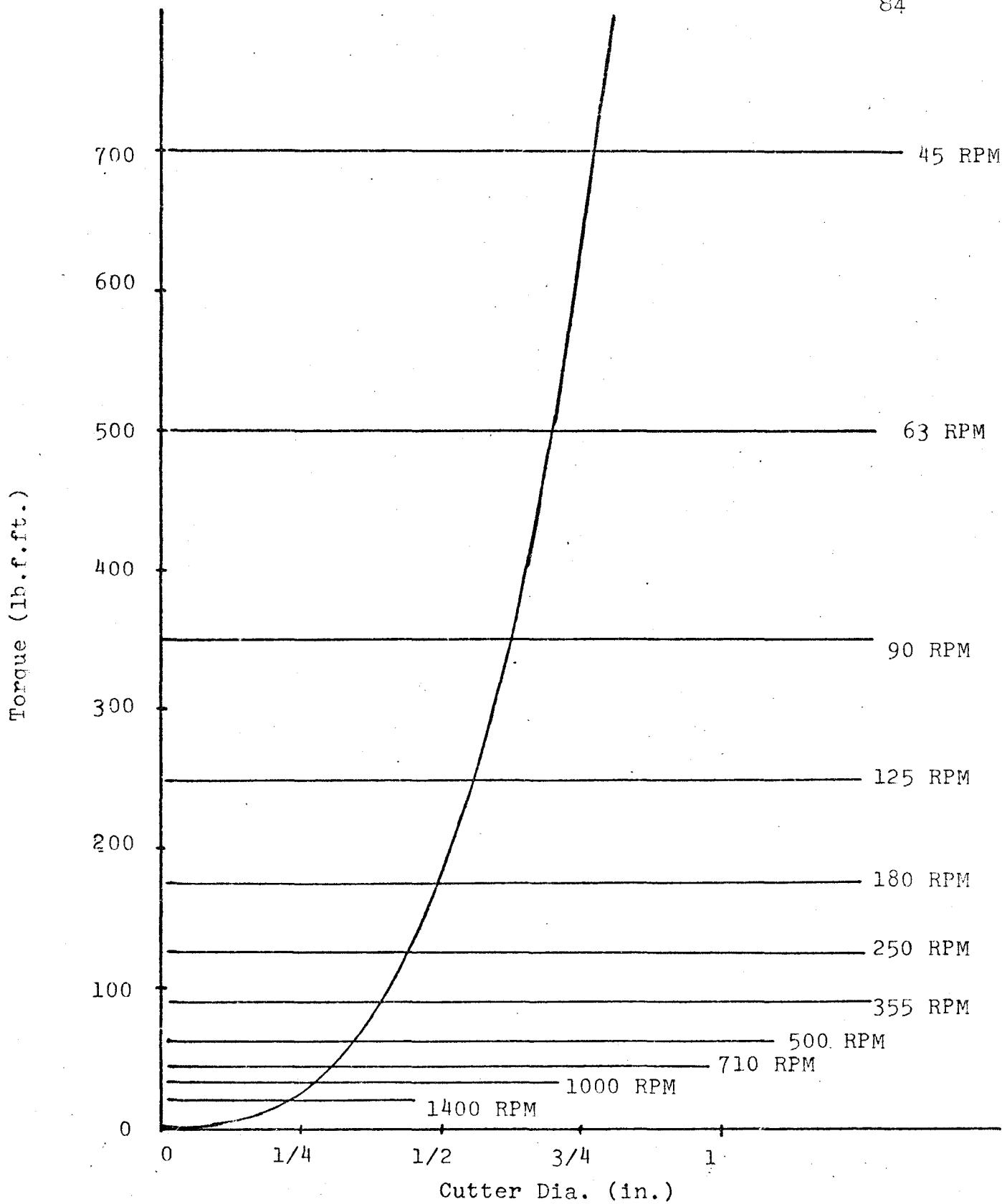


FIGURE 3.2: Graph of Torque vs. Diameter of Cutter

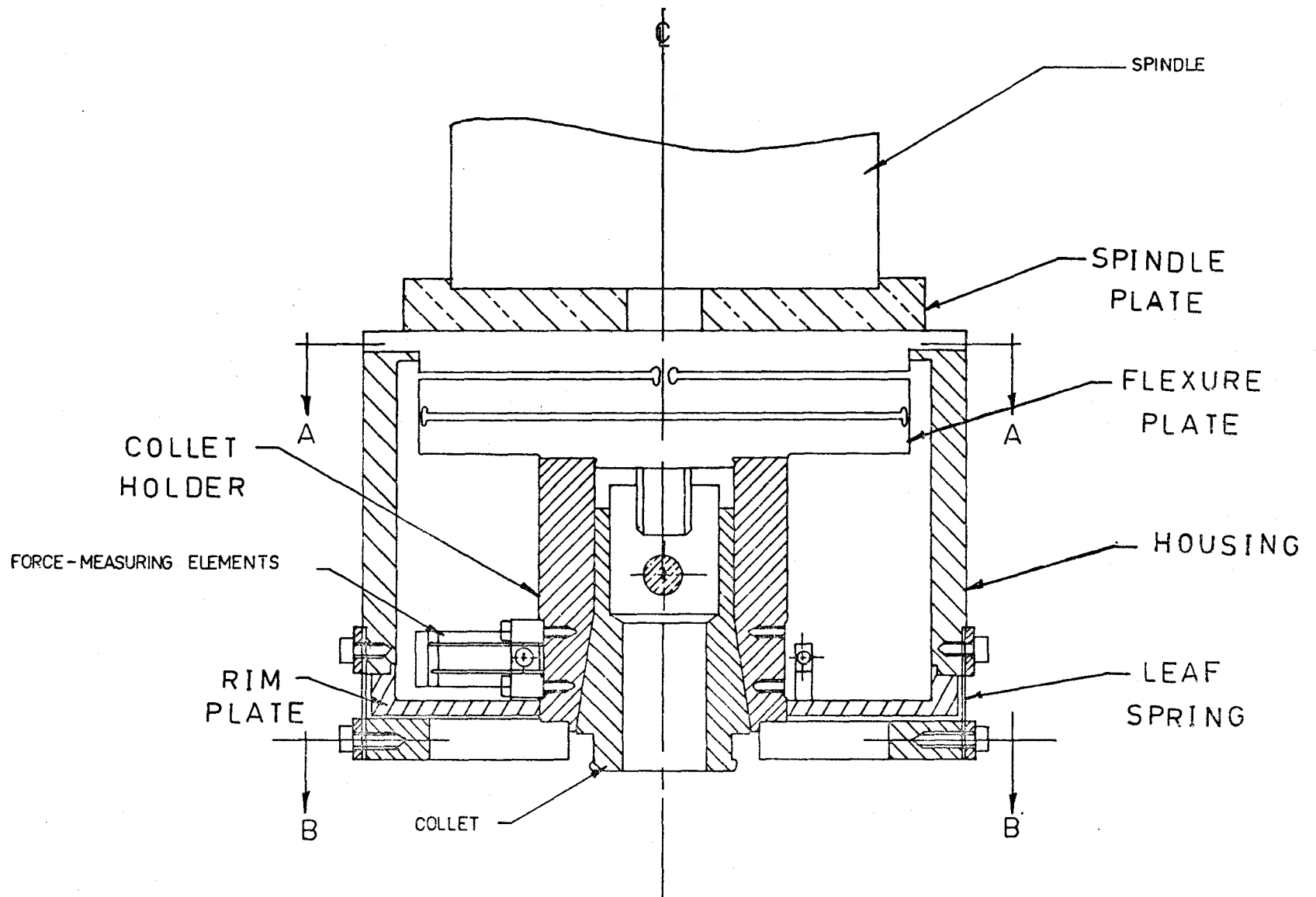


FIGURE 4.1: Dynamometer (Assembly)

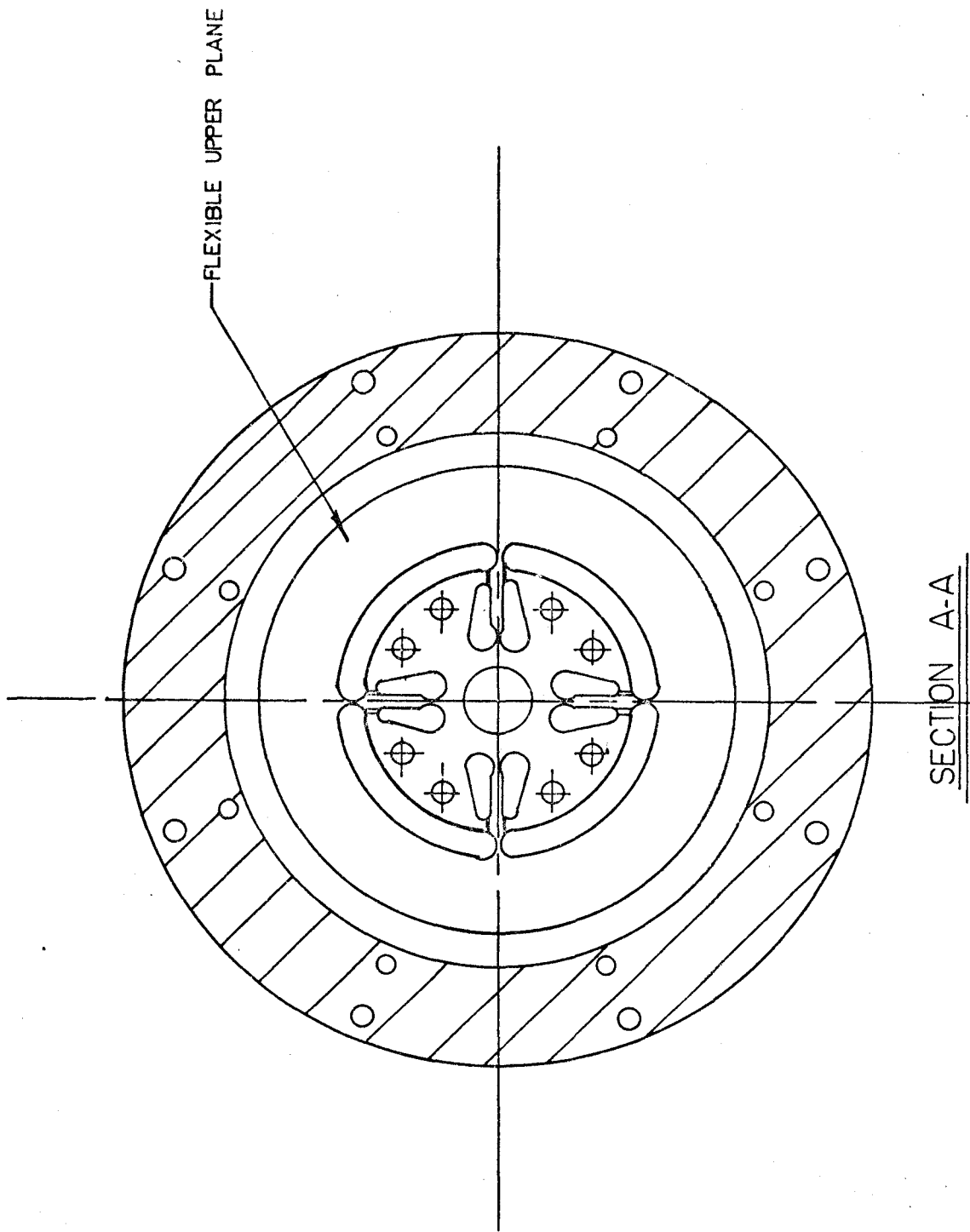


FIGURE 4.2: Flexure plate

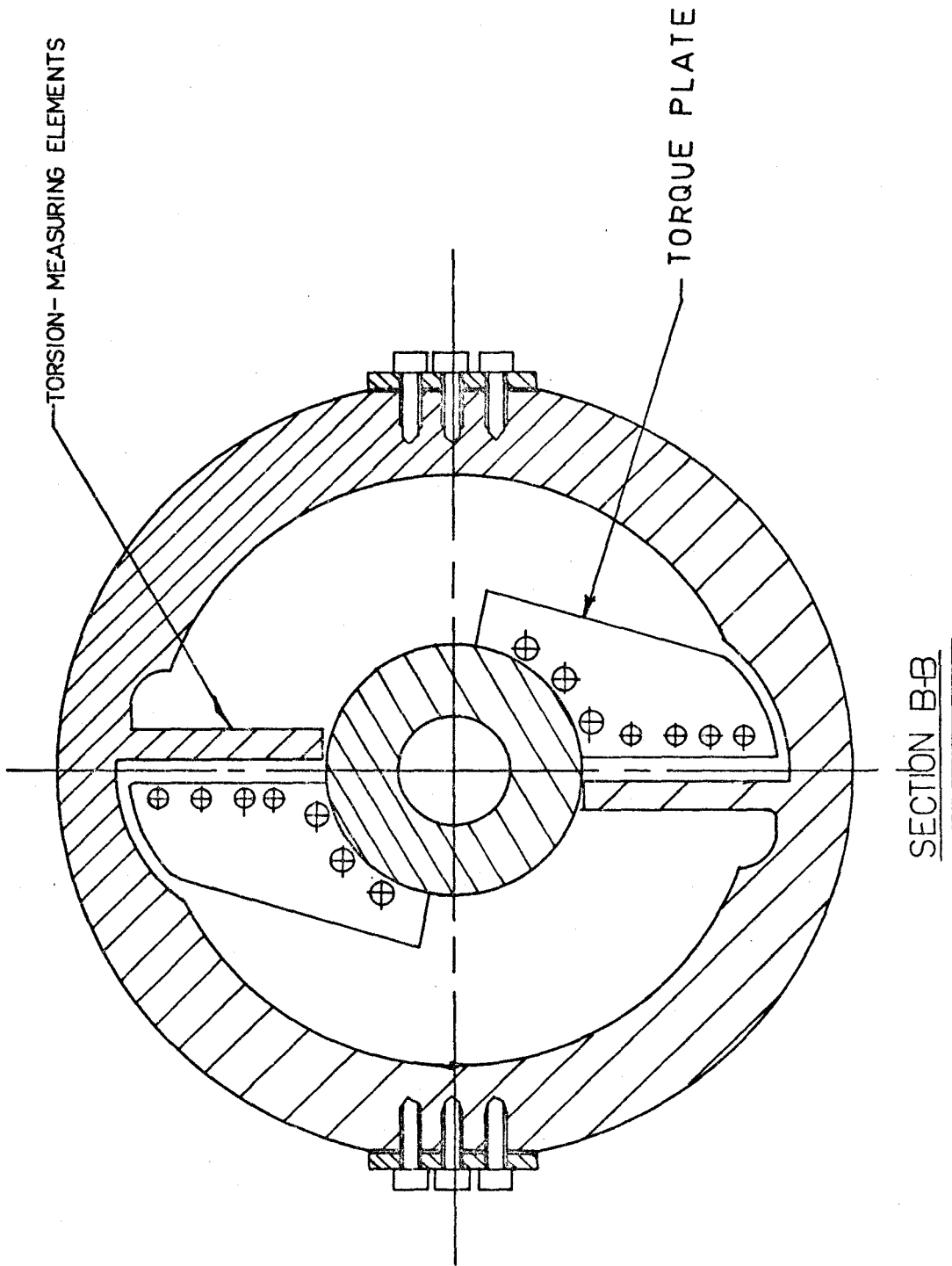


FIGURE 4.3: Torque Ring

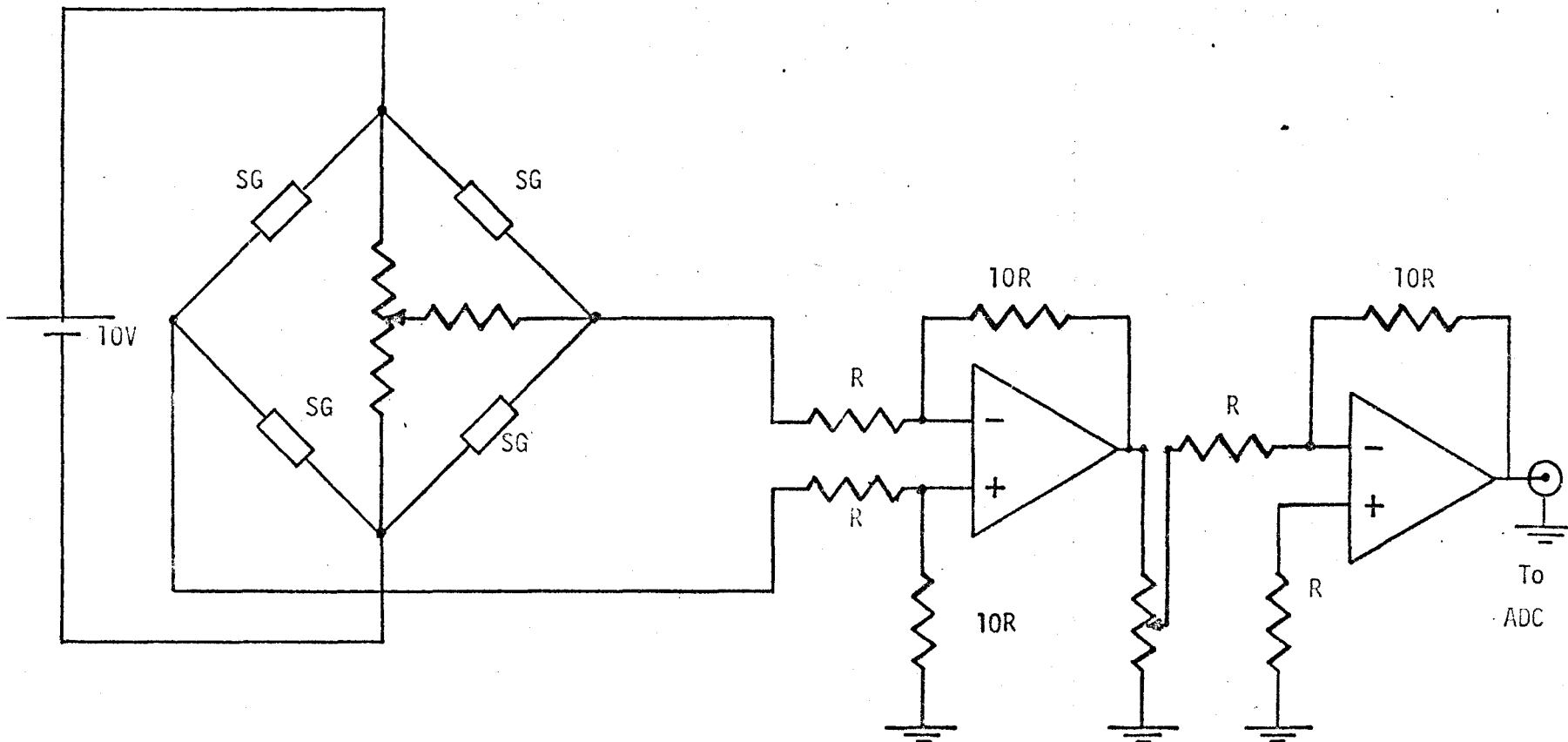


FIGURE 5.1: Electronic Bridge

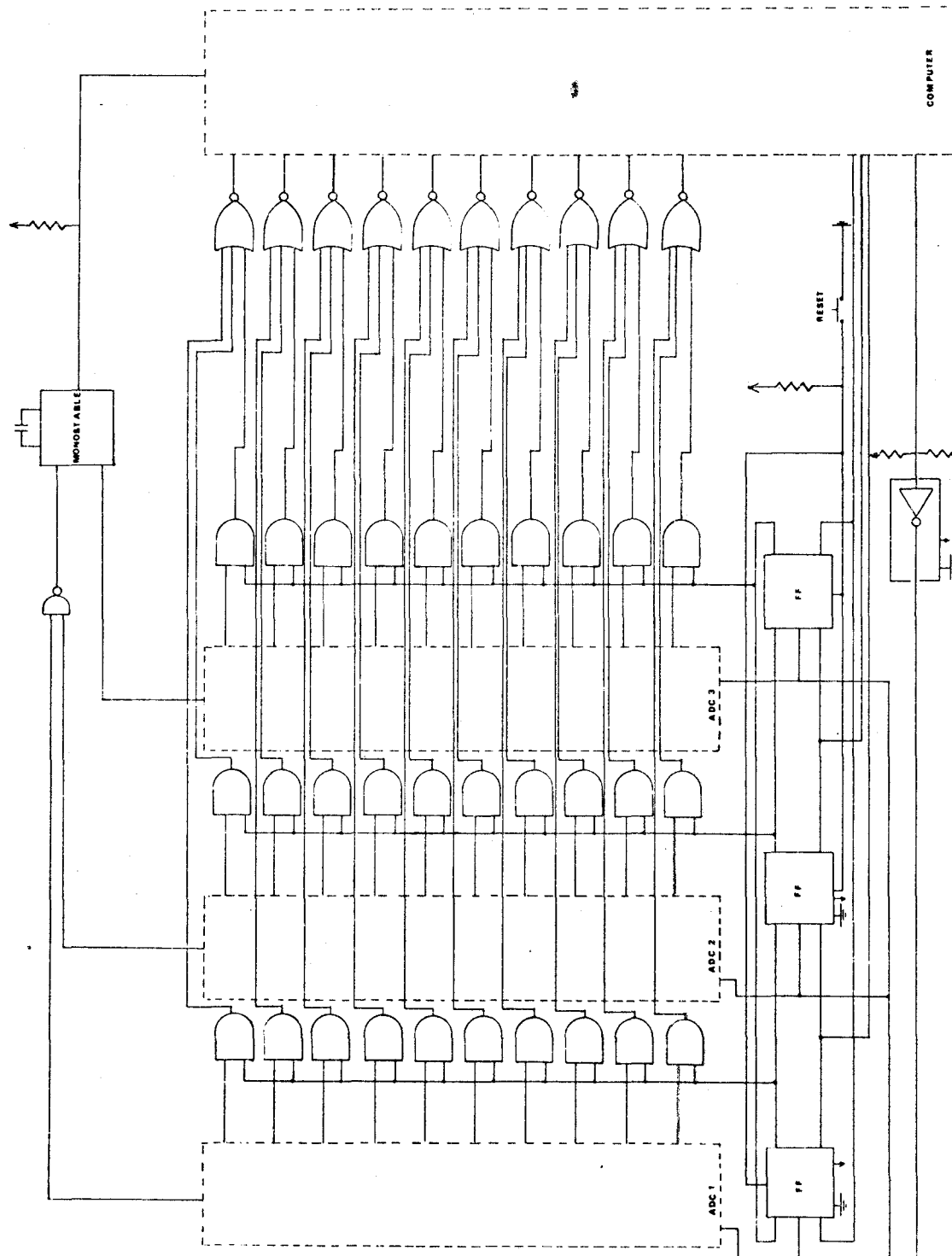


FIGURE 5.2: Analog-to-Digital Processor (ADP)

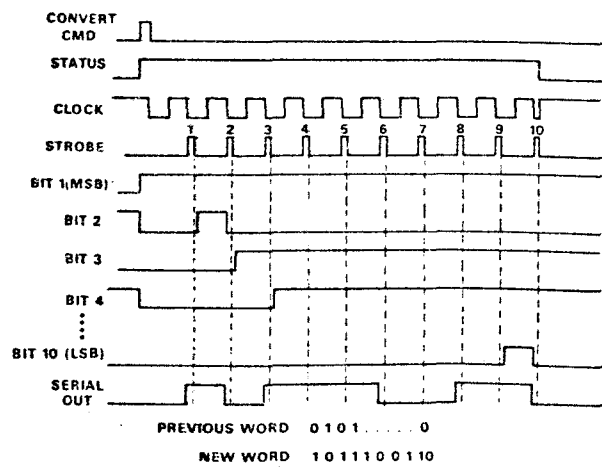


FIGURE 5.3: ADC-10Z Timing Diagram

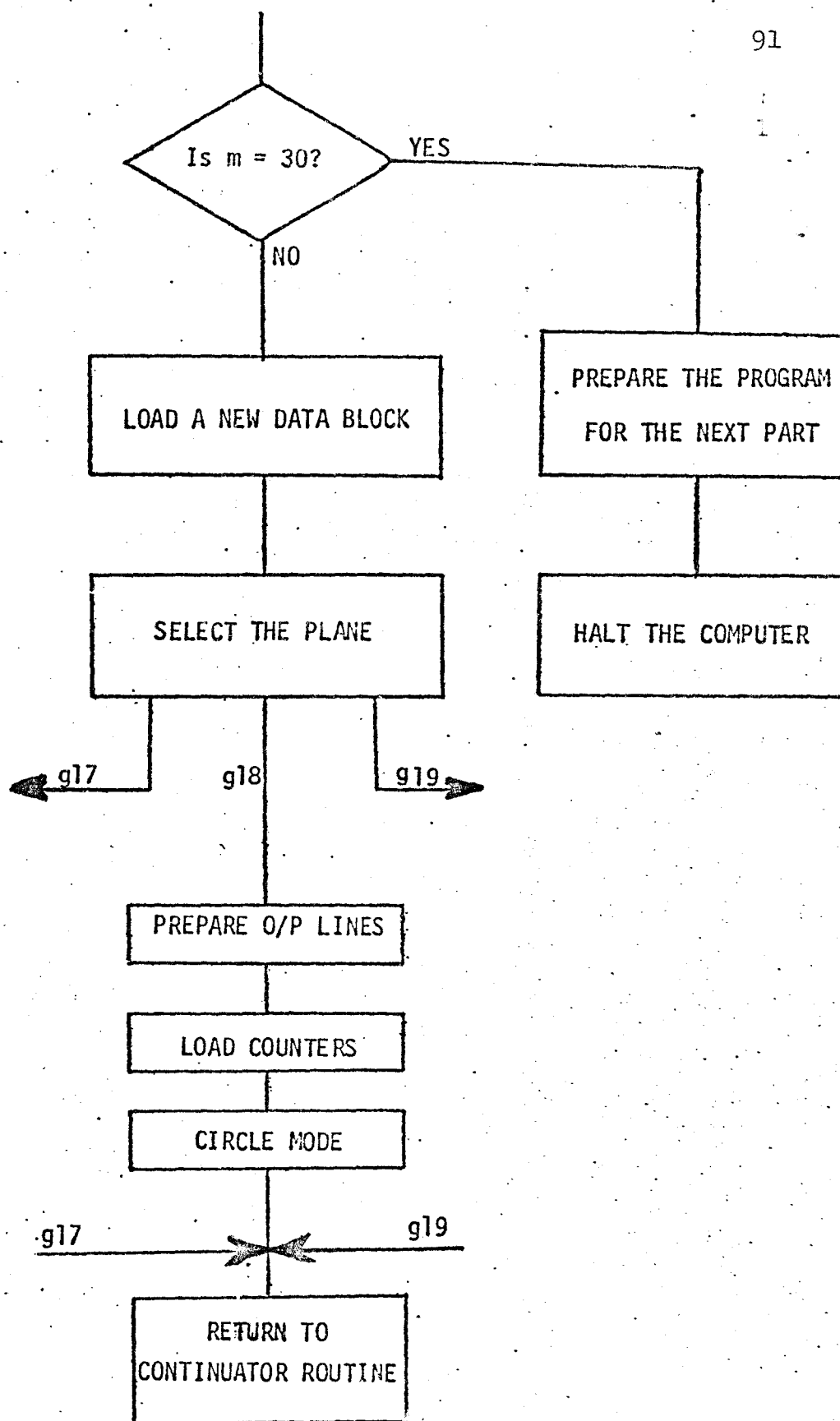


FIGURE 6.1: Initiator Routine

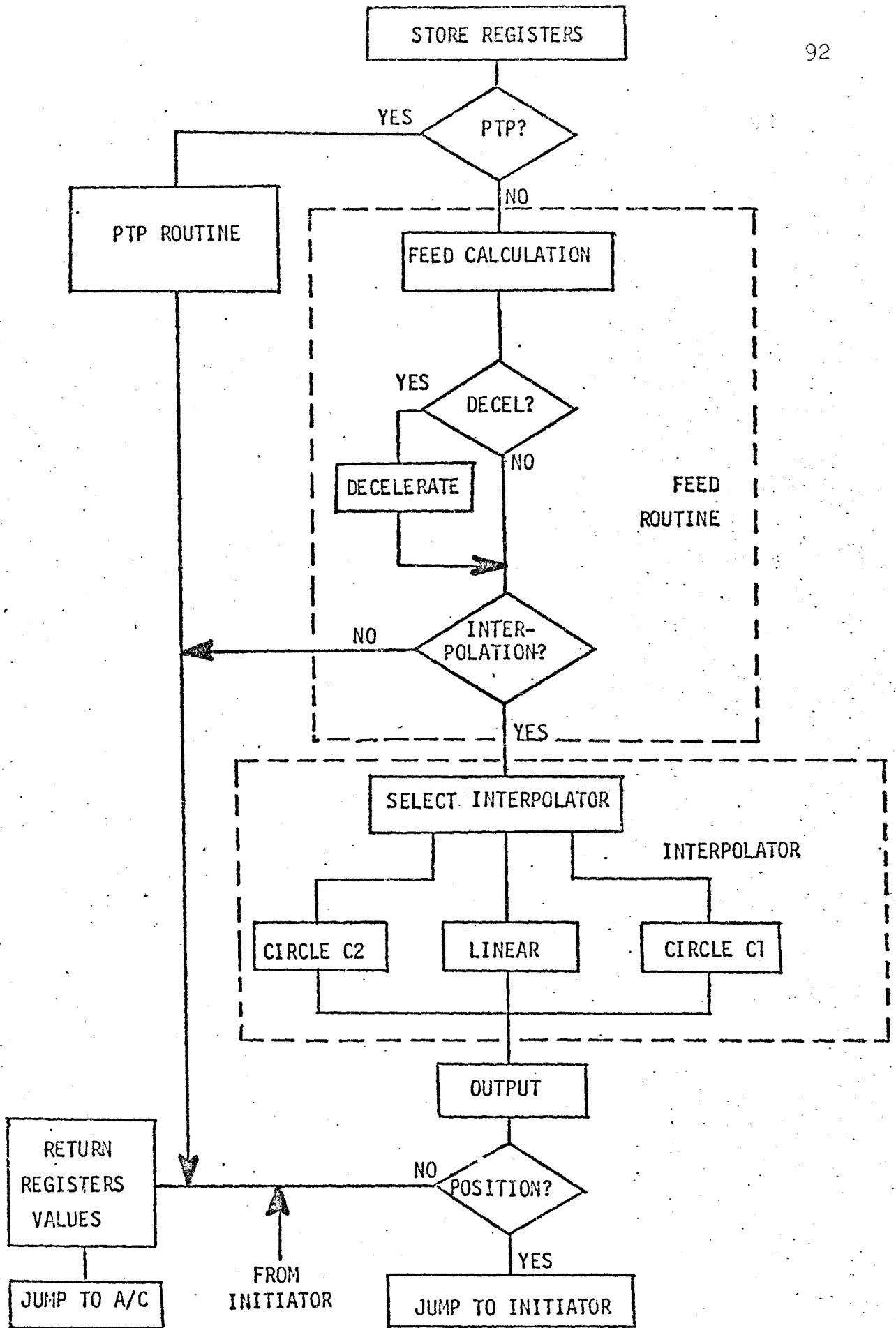


FIGURE 6.2 Continuator Routine

Code	Function	Explanation
g 00	Point-to-Point, Positioning	Used with combination point-to-point/ contouring systems for indicating positioning operation.
g 01	Linear Interpolation (Normal Dimensions)	A mode of contouring control used for generating a slope or straight cut, where the incremental dimensions are normal, i.e.: input resolution is as specified.
g 02	Circular Interpolation Arc CW (Normal Dimensions)	A mode of contouring control which produces an arc of a circle by the coordinated motion of two axes. The curvature of the path (clockwise=g02, or counter clockwise=g03) is determined when viewing the plane of motion in the negative direction of the perpendicular axis. The distances to the arc center (i,j,k) are "normal dimensions".
g 03	Circular Interpolation Arc CCW (Normal Dimensions)	
g 04	Dwell	A programmed (or established) time delay, during which there is no machine motion. Its duration is adjusted elsewhere, usually by the F word. In this case dimension words should be set at zero.
g 09	Deceleration	The feed rate decreases smoothly (usually exponentially) to a fixed percent (20% in our case) of the programmed feedrate in the deceleration block.
g 17 g 18 g 19	XY Plane Selection XZ Plane Selection YZ Plane Selection	Used to identify the plane for Interpolation. The Plane Code is stored until it is replaced by another. In a block which contains the Plane Selection Code, a linear interpolation is performed.

TABLE 1: Preparatory Functions

Code	Function	Explanation
m 03	Spindle CW	Start spindle rotation in a clockwise direction.
m 05	Spindle OFF	Stop spindle.
m 06	Tool Change	Execute the change of a tool.
m 30	End of Tape	Indicates completion of the workpiece. It stops spindle and feed after completion of all instructions in the block. It includes rewinding of tape to the end of record character, thus ready for next workpiece.

TABLE 2: Miscellaneous Functions

PART NAME:

DATE:

N	G	X	Y	Z	I	J	K	F	M
1	18	2000	0000	4000	0000	0000	0000	300	03 (EB)
2	17	-3000	-3000	0	0	0	0	700	(EB)
3	03	-5000	-5000	0	0	5000	0	700	(EB)
4	03	5000	-5000	0	5000	0	0	700	(EB)
5	01	8500	0	0	0	0	0	700	(EB)

FIGURE 7.1: Data Manuscript

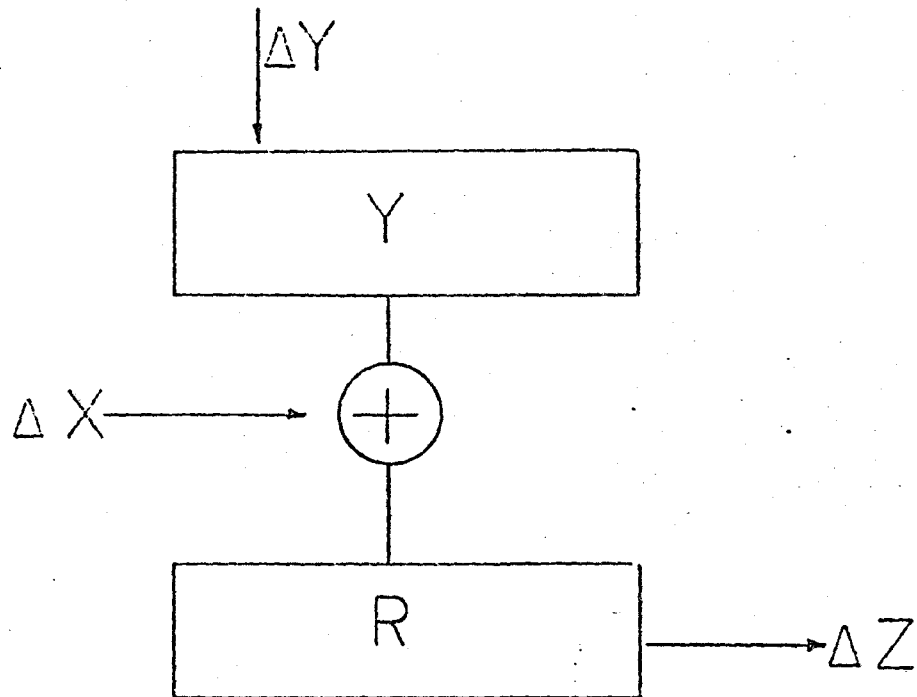


FIGURE 6.3: Digital Intergrator Ref(10)

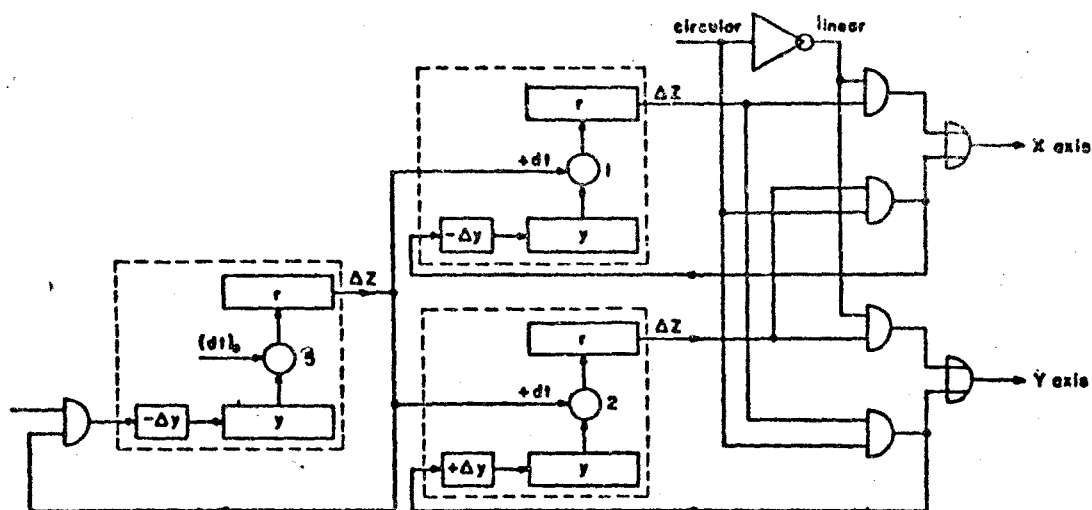


FIGURE 6.4a: Interpolator

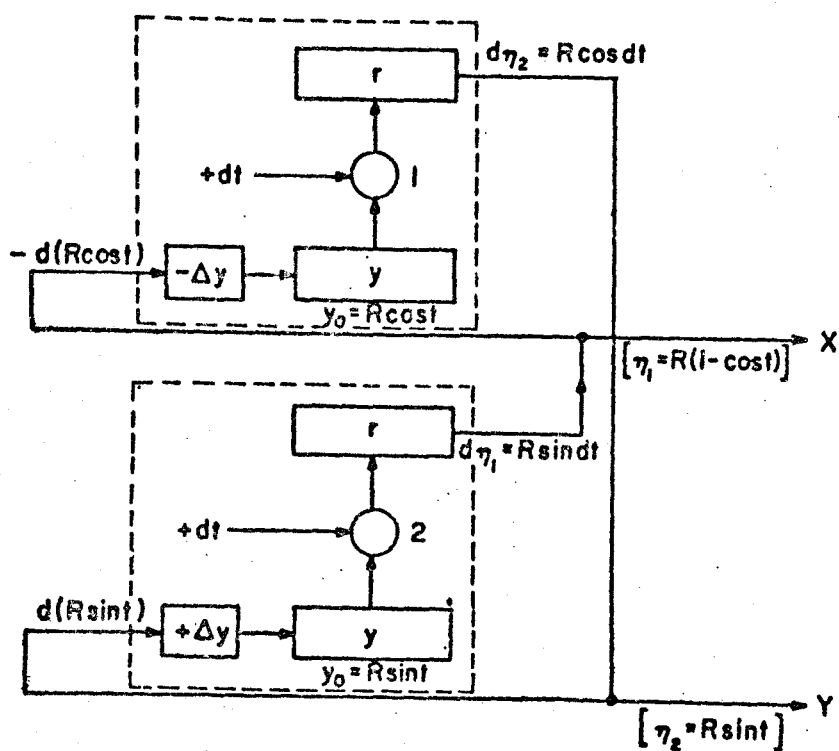


FIGURE 6.4b: Circular Interpolator

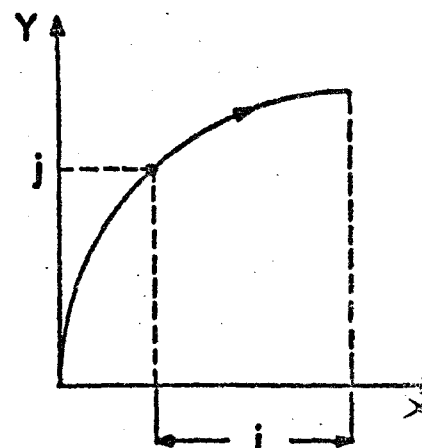
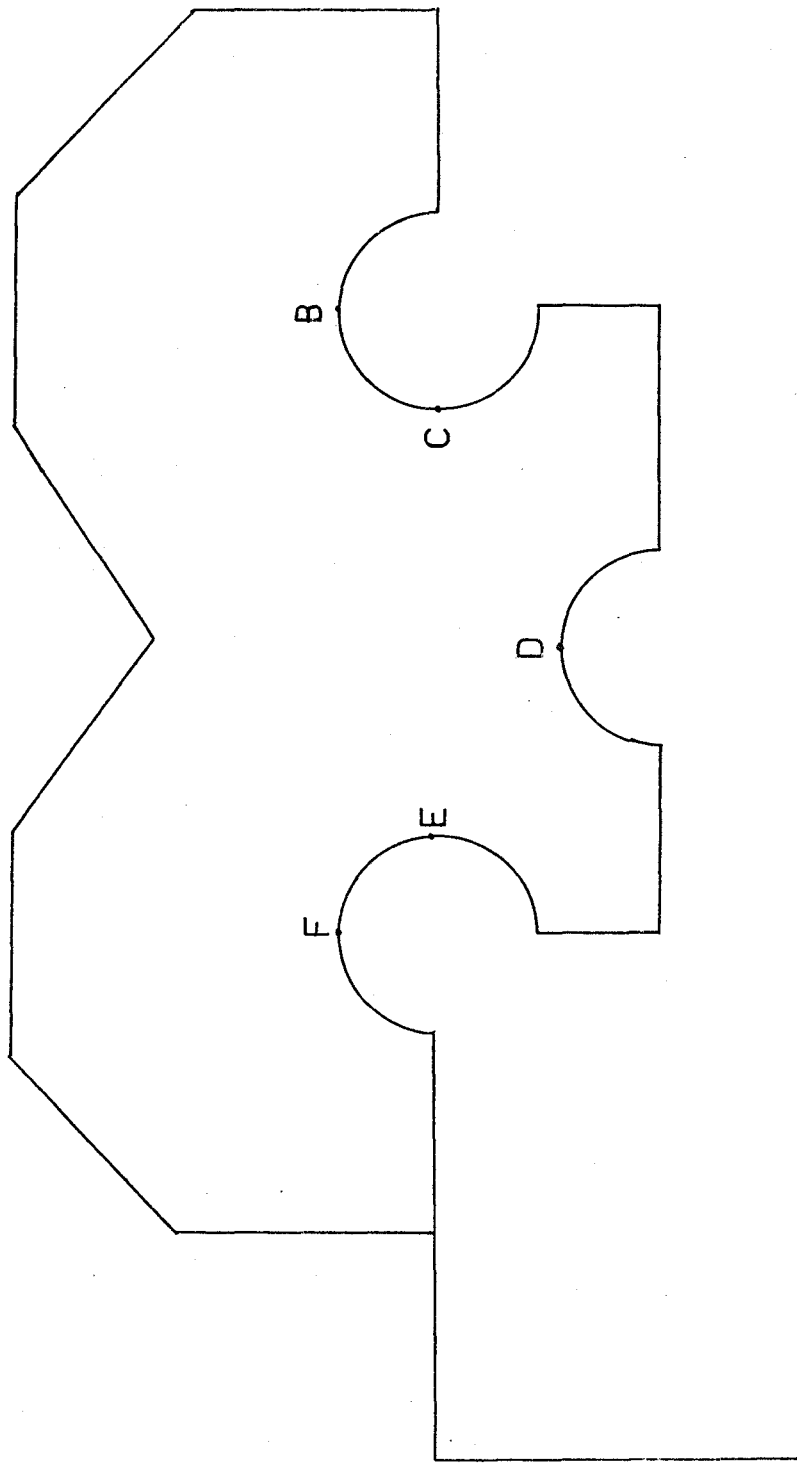


FIGURE 6.5: Initial conditions of circuit

FIGURE 6.6: Profile to check "Lost Motion"



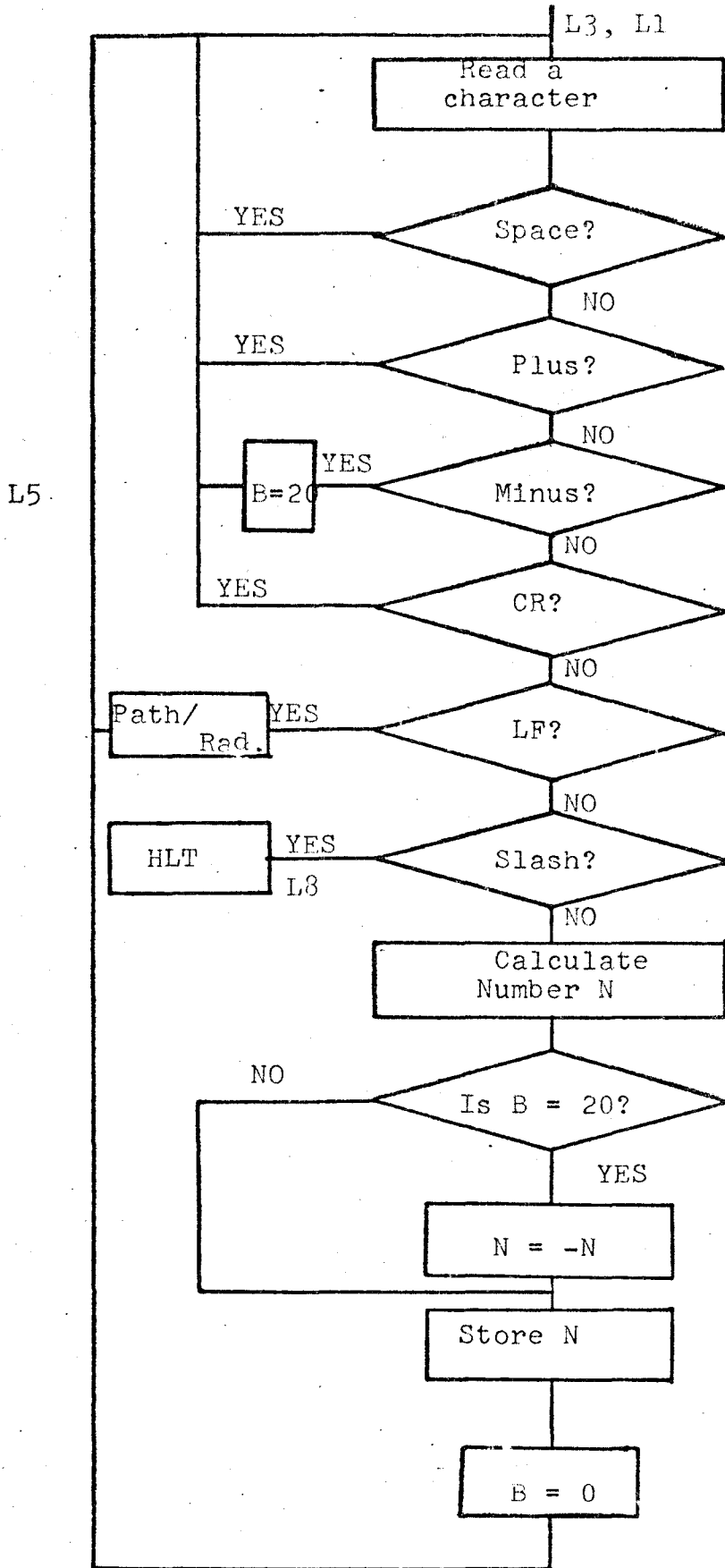


FIGURE 7.2: Data Reading Programme (DRP)

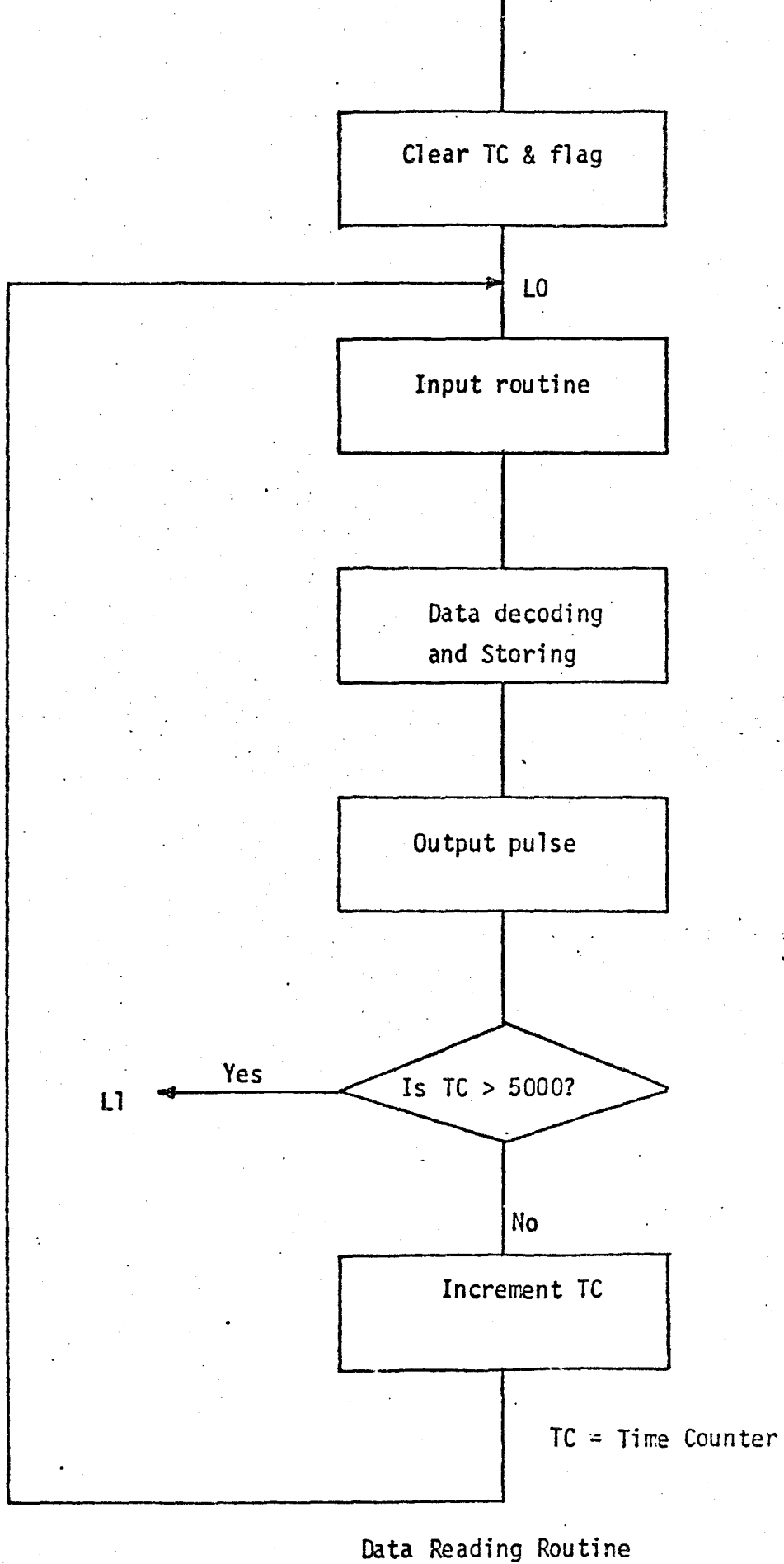
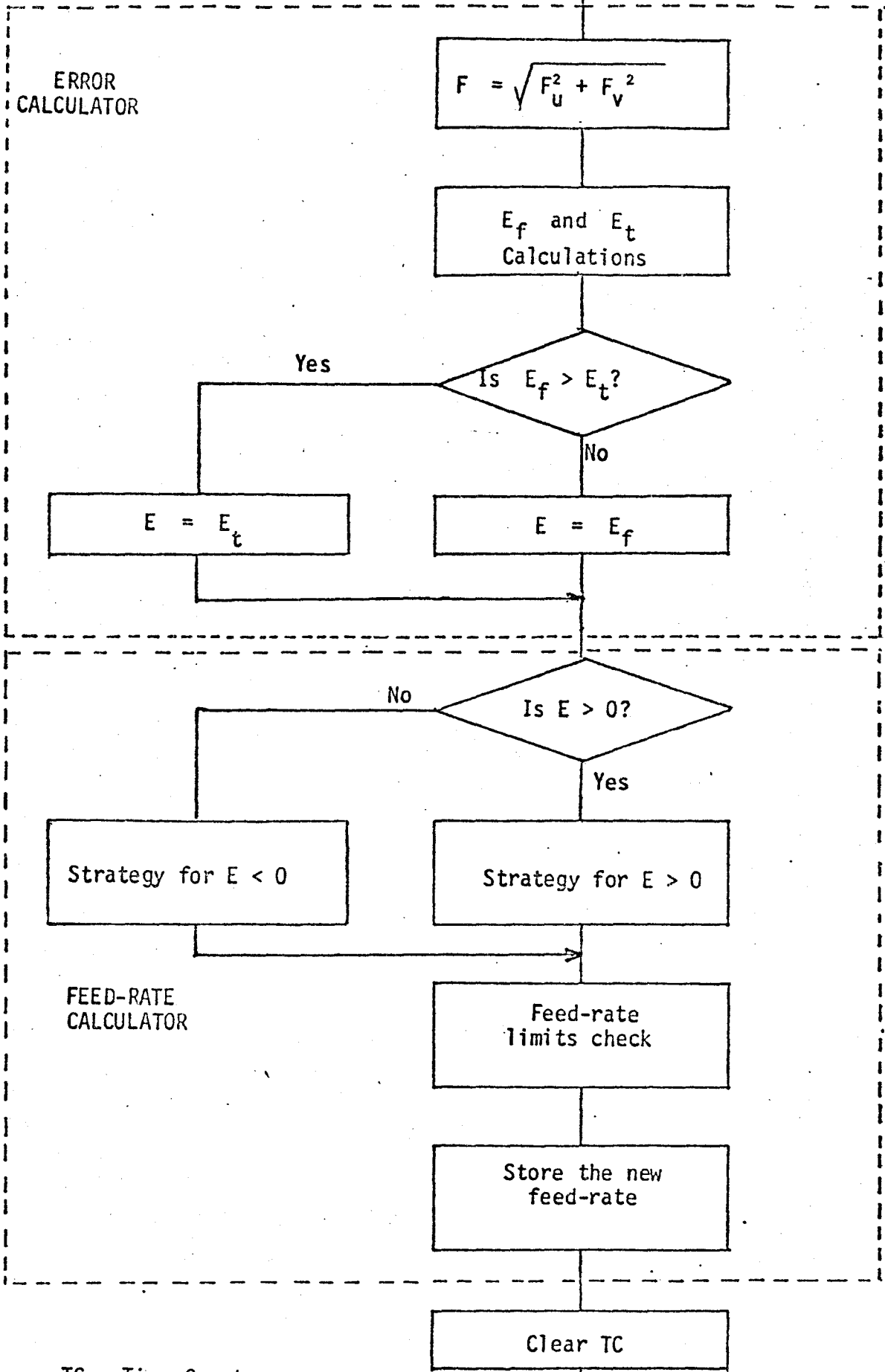


FIGURE 8.1



TC - Time Counter
 E_f - Force Error
 E_t - Torque Error

FIGURE 8.2

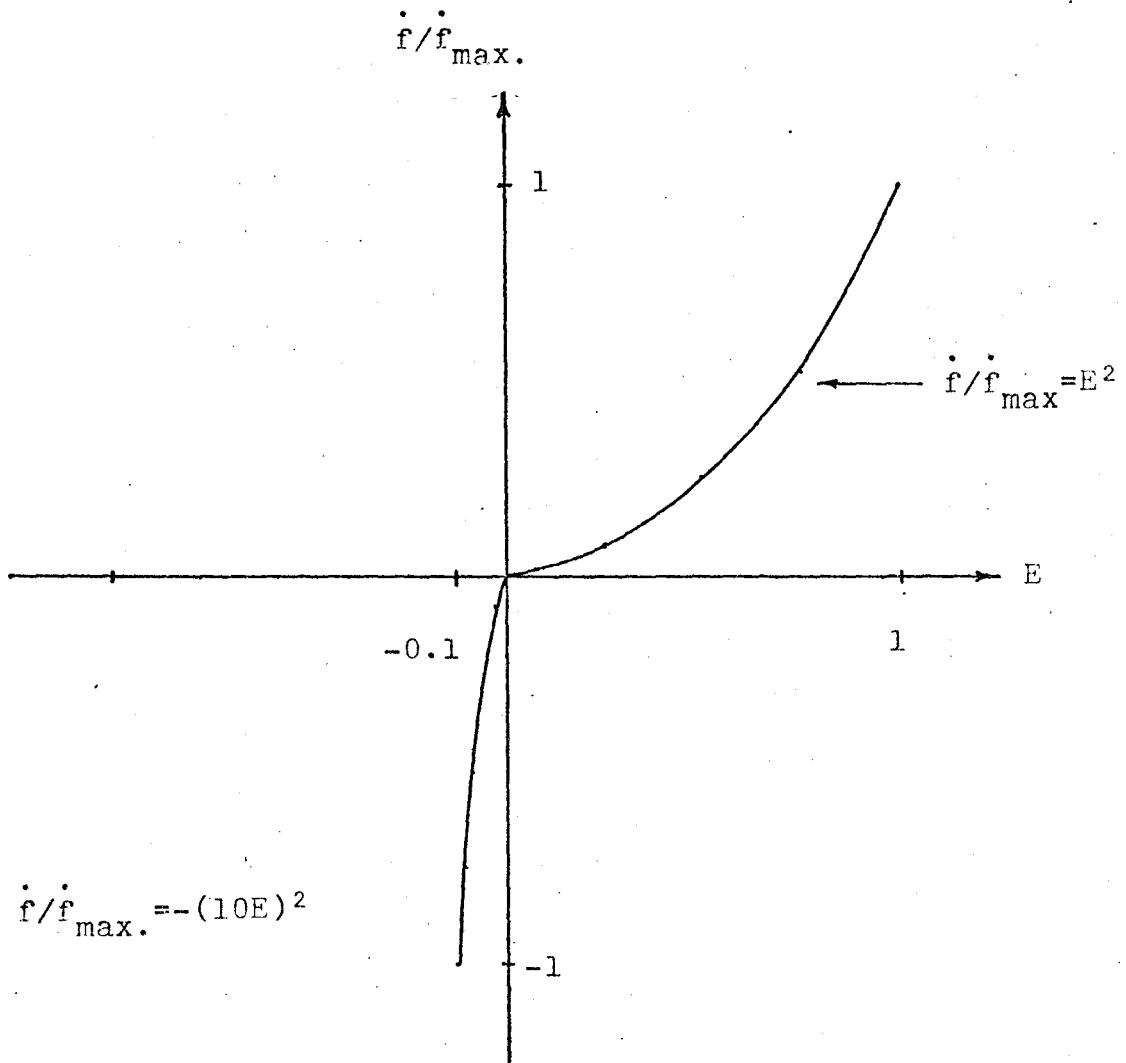
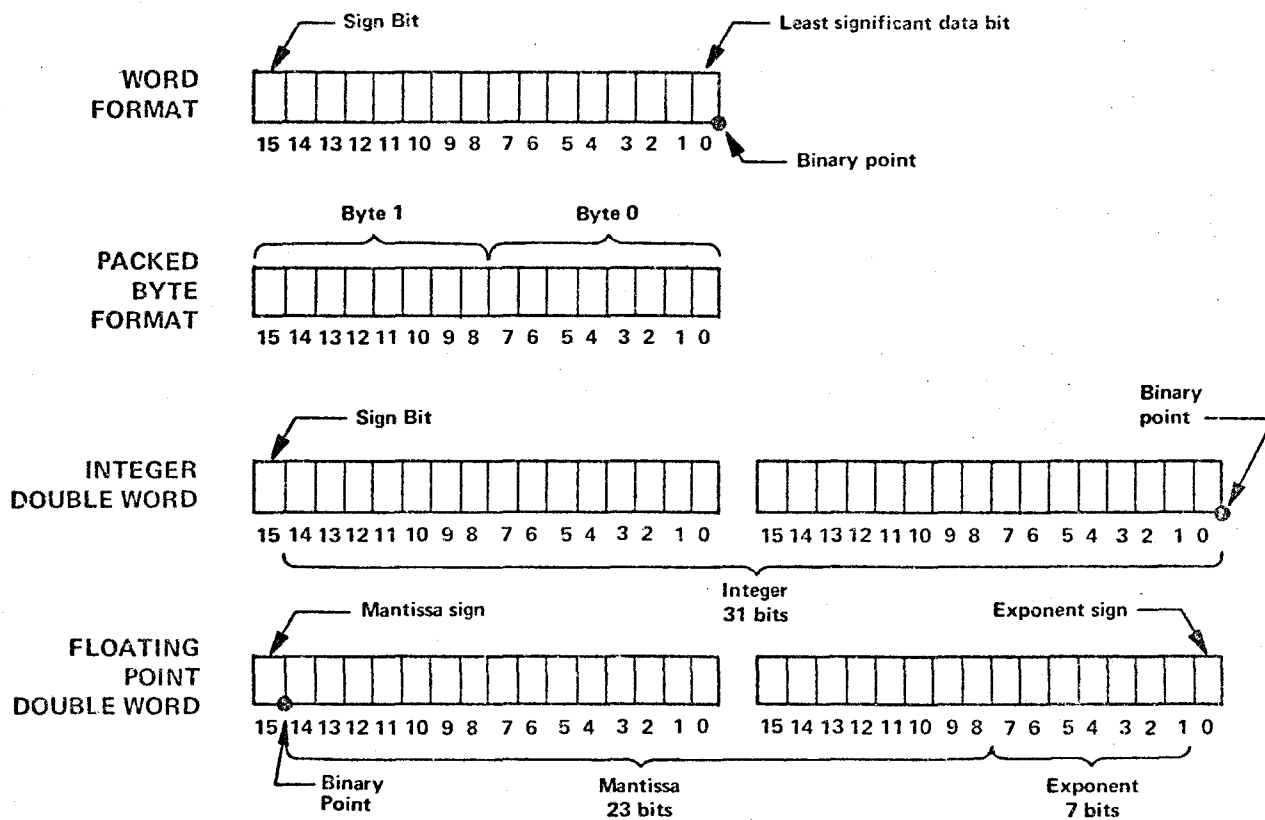


FIGURE 8.3: The AC strategy



OCTAL NOTATION

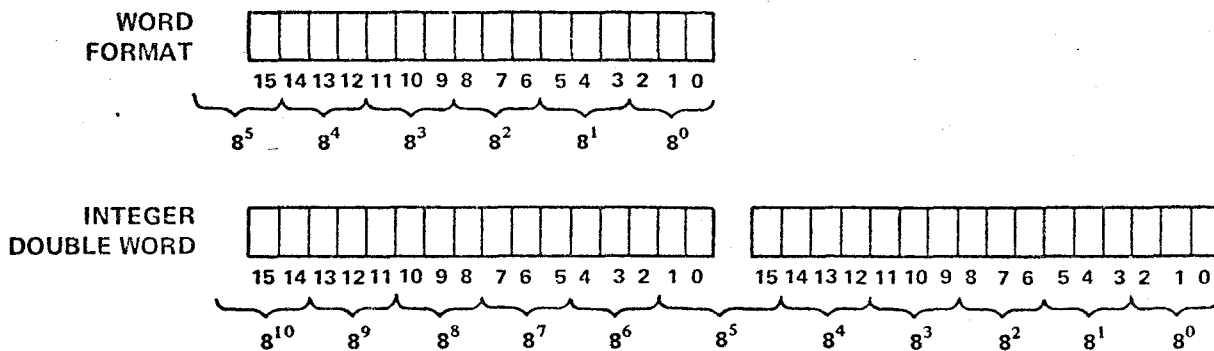


FIGURE A1: Data Formats and Octal Notation Ref(6)

MEMORY SIZE	PAGE	OCTAL ADDRESSES
4K ↓	0	00000 to 01777
	1	02000 to 03777
	2	04000 to 05777
	3	06000 to 07777
8K ↓	4	10000 to 11777
	5	12000 to 13777
	6	14000 to 15777
	7	16000 to 17777
12K ↓	8	20000 to 21777
	9	22000 to 23777
	10	24000 to 25777
	11	26000 to 27777
16K ↓	12	30000 to 31777
	13	32000 to 33777
	14	34000 to 35777
	15	36000 to 37777
24K ↓	16	40000 to 41777
	17	42000 to 43777
	18	44000 to 45777
	19	46000 to 47777
	20	50000 to 51777
	21	52000 to 53777
	22	54000 to 55777
	23	56000 to 57777
32K ↓	24	60000 to 61777
	25	62000 to 63777
	26	64000 to 65777
	27	66000 to 67777
	28	70000 to 71777
	29	72000 to 73777
	30	74000 to 75777
	31	76000 to 77777

TABLE A2: Memory Pages Ref(6)

INSTRUCTION		TIME (μ sec)
MPY (Multiply)		10.78
DIV (Divide)	Max	16.66
DLD (Double Load)		5.88
DST (Double Store)		5.88
	Number of Shifts	
ASR (Arithmetic Shift Right)	1, 2, 3 4, 5, 6, 7, 8 9, 10, 11, 12, 13 14, 15, 16	2.94 3.92 4.90 5.88
ASL (Arithmetic Shift Left)	1, 2, 3, 4, 5 6, 7, 8, 9, 10 11, 12, 13, 14, 15 16	4.90 5.88 6.86 7.84
LSR, RRR (Logical Shift Right, Rotate Right)	1, 2 3, 4, 5, 6, 7 8, 9, 10, 11, 12 13, 14, 15, 16	2.94 3.92 4.90 5.88
LSL, RRL (Logical Shift Left, Rotate Left)	1, 2, 3, 4 5, 6, 7, 8, 9 10, 11, 12, 13, 14 15, 16	4.90 5.88 6.86 7.84

TABLE A1: Extended Arithmetic Execution Times
Ref(6)

APPENDIX A

Data Formats

"The basic data format for the 2100 A computer is a 16-bit word. Bit positions are numbered from 0 through 15, in order of increasing significance. Bit position 15 of the data format is used for the sign bit; a "0" in this position indicates a positive number and a "1" indicates a negative number. The data is assumed to be a whole number, thus the binary point is assumed to be to the right of the number.

The basic word, as shown in Figure A1 can also be divided into two 8-bit bytes or combined to form a 32-bit double word. The byte format is used for character-oriented input/output devices. Packing of the bytes into one word is accomplished by the software drivers. In I/O operations the higher order byte (Byte 1) is the first to be transferred.

The integer doubleword format is used for extended precision arithmetic instructions. Bit 15 of the most significant word is the sign bit, and the binary point is assumed to be to the right of the least significant word location; the next higher location contains the most significant word. When loaded into the accumulators, the B-register contains the most significant word and the A-register contains the least significant word.

The floating point doubleword format is used with floating point software. Bit 15 of the most significant word is the exponent sign bit. Bits 1 through 17 are used to express

the mantissa. The mantissa is assumed to be a fractional value, thus the binary point appears to the left of the mantissa. Software converts decimal number to this binary form and normalizes the quantity expressed (sign and leading mantissa bit differ). If either the mantissa or the exponent is negative, that part is stored in two's complement form. The number must be in the approximate range of 10^{-38} through 10^{+38} .

Figure A1 also illustrates the octal notation of data for both single-length and double-length words. Each group of three bits, beginning at the right, is combined to form an octal digit. Each digit to the left increases in significance. A single-length 16-bit word can therefore be fully expressed by six octal digits and a double-length 32-bit word can be fully expressed by 11 octal digits. Octal notation is not shown for byte or floating point formats, since bytes normally represent characters and floating point numbers are given in decimal form.

For single-word data, the range of representable numbers is +32,767 to -32,768 (decimal), or +77,777 to -100,000 (octal). For doubleword integer data, the range is +2,147,483,647 to -2,147,483,468 (decimal) or +17,777,777,777 to -20,000,000 (octal).

Memory Addressing

The computer memory is logically divided into pages of 1024 words each. A page is defined as the largest block of memory which can be directly addressed by the memory address

bits of a memory reference instruction (single length). These memory reference instructions have 10-bits to specify a memory address, and thus the page size is 1024 locations (2000 in octal notation). Octal addresses for each page, up to the maximum memory size, are given in Table A2.

Provision is made to address directly one of two pages: page zero (the page in which the instruction itself is located). Memory reference instructions include a bit (bit 10) reserved to specify one or the other of these two pages. To address locations in any other page, indirect addressing is used. Page references are specified by bit 10 as follows:

Logic 0 = Page Zero (z)

Logic 1 = Current Page (c)

All memory reference instructions reserve a bit to specify direct or indirect addressing. For single-length memory reference instructions, bit 15 of the instruction word is used; for extended arithmetic memory reference instructions, bit 15 of the address word is used. Indirect addressing uses the address part of the instruction to access another word in memory, which is taken as a new memory reference for the same instruction. This new address word is full 16 bits long, 15 bits of address plus another direct/indirect bit. The 15-bit length of the address permits access to any location in memory. If bit 15 again specified indirect addressing, still another address is obtained; this multiple-step indirect addressing may be done to any number of levels. The first address

obtained in the indirect phase which does not specify another indirect level becomes the effective address for the instruction. Direct or indirect addressing is specified by bit 15 as follows:

Logic 0 = Direct

Logic 1 = Indirect

Interrupt System

Of the 60 interrupt levels, the two highest priority levels are reserved for hardware faults (power fail and parity error), the next two are reserved for DMA completion interrupts, and the remaining 56 are available for the I/O device channels. Interrupt requests received while the computer is in half mode will be processed, in order of priority, when the computer is put into run mode or is stepped single cycle.

The flag with the highest priority generates an interrupt signal at the end of the current machine cycle except in order any of the following circumstances

(a) Interrupt system disabled or device interrupt disabled.

(b) JMP indirect or JSB indirect not sufficiently executed. These instructions inhibit all interrupts, except memory protect, until the instruction (plus one phase of the succeeding instruction) is completed, or until at least three indirect references have occurred. The memory protect interrupt for a jump violation will occur on completion of the execute phase, but the jump itself will be inhibited.

(c) Instruction in an interrupt location not sufficiently executed, even if of lower priority. Any interrupt inhibits the entire interrupt system at least two phases have been completed. (JMP indirect and JSB indirect will be fully executed.)

Instruction Timing

All instructions except ISZ and the extended arithmetic instructions are fully executed in 1.96 microseconds. ISZ is executed in 2.94 microseconds, and the extended arithmetic instructions executed in the times shown in Table A1. The Divide instruction executes faster than shown if the divider is positive (15.68 microseconds) or if overflow occurs (11.76 microseconds). If indirect addressing is used with any of the single-word memory reference instructions, 0.98 microseconds is added for each level of indirect addressing used; 1.96 microseconds are added for each level of indirect addressing with extended arithmetic memory reference instructions.

Instructions are executed in two or more phases. The first phase is the fetch phase, which obtains an instruction from memory and transfers it onto the central processor's instruction register. Next, there can be one or more indirect phases. The indirect phase, which applies only to single-length memory reference instructions, obtains a new operand address for the same (current) instruction. Lastly, there is an execute phase, which accomplishes actual execution of the instruction. For extended arithmetic memory reference instructions indirect addressing is also accomplished in the execute phase.

Although the duration of a phase varies considerably (588 nanoseconds to an indeterminate time in the case of extended arithmetic indirect addressing), synchronization with memory or input/output operations results in overall execution times as specified in the preceding paragraph."^[6]

COMMENTS

0001		ASMB, A, B, L	
0002	02024		ORG 2024B
0003	02024 002400		CLA
0004	02025 172263		STA K, I
0005	02026 102112		STF 12B
0006	02027 002400	L0	CLA
0007	02030 003000		CMA
0008	02031 102612		OTA 12B
0009	02032 102712		STC 12B
0010	02033 102312		SFS 12B
0011	02034 026033		JMP *-1
0012	02035 102512		LIA 12B
0013	02036 106712		CLC 12B
0014	02037 072316		STA TS
0015	02040 012273		AND COD
0016	02041 072317		STA TSS
0017	02042 066317		LDB TSS
0018	02043 062316		LDA TS
0019	02044 012272		AND CHH
0020	02045 052267		CPA CH1
0021	02046 076313		STB M1
0022	02047 052270		CPA CH2
0023	02050 076314		STB M2
0024	02051 052271		CPA CH3
0025	02052 076315		STB M3
0026	02053 002400		CLA
0027	02054 102612		OTA 12B
0028	02055 103712		STC 12B, C
0029	02056 106712		CLC 12B
0030	02057 162263		LDA K, I
0031	02060 042302		ADA FT
0032	02061 002021		SSA, RSS
0033	02062 026067		JMP L1
0034	02063 162263		LDA K, I
0035	02064 042274		ADA CN1
0036	02065 172263		STA K, I
0037	02066 026027		JMP L0
0038	02067 062313	L1	LDA M1
0039	02070 003004		CHA, INA
0040	02071 042314		ADA M2
0041	02072 002021		SSA, RSS
0042	02073 026101		JMP L2
0043	02074 062313		LDA M1
0044	02075 072323		STA P
0045	02076 062314		LDA M2
0046	02077 072324		STA Q
0047	02100 026105		JMP L3
0048	02101 062313	L2	LDA M1
0049	02102 072324		STA Q
0050	02103 062314		LDA M2
0051	02104 072323		STA P
0052	02105 062324	L3	LDA Q
0053	02106 100200		MPY Q
0054	02110 104300		DST R
0055	02111 002381		STB I
0056	02112 100200		MPY Q

Input Routine

Data Decoding and Storing

Output Pulse

Is TC > 5,000?

Is $F_u > F_v$

Storing F_u, F_v Appropriately

Calculation of $F (= \sqrt{F_u^2 + F_v^2})$

```

02113 002323
0056 02114 001100      ARS
0057 02115 072325      STA Y
0058 02116 100200      MPY Y
02117 002325
0059 02120 100400      DIV P
02121 002323
0060 02122 001100      ARS
0061 02123 072326      STA Z
0062 02124 100200      MPY Y
02125 002325
0063 02126 100400      DIV P
02127 002323
0064 02130 042323      ADA P
0065 02131 042325      ADA Y
0066 02132 072320      STA RR
0067 02133 062326      LDA Z
0068 02134 003004      CMA, INA
0069 02135 042320      ADA RR
0070 02136 072332      STA SQR
0071 02137 003004      CMA, INA
0072 02140 142264      ADA EN1, I
0073 02141 100200      MPY TY
02142 002301
0074 02143 100400      DIV EN1, I
02144 102264
0075 02145 072333      STA E1
0076 02146 062315      LDA M3
0077 02147 003004      CMA, INA
0078 02150 142265      ADA EN3, I
0079 02151 100200      MPY TY
02152 002301
0080 02153 100400      DIV EN3, I
02154 102265
0081 02155 072336      STA E3
0082 02156 003004      CMA, INA
0083 02157 042333      ADA E1
0084 02160 002000      SSA
0085 02161 026165      JMP L4
0086 02162 032335      LDA E3
0087 02163 072335      STA E
0088 02164 026167      JMP L5
0089 02165 062333      L4 LDA E1
0090 02166 072335      STA E
0091 02167 002000      L5 SSA
0092 02170 026216      JMP L6
0093 02171 100200      MPY E
02172 002335
0094 02173 072331      STA EE
0095 02174 002307      ADA EE
0096 02175 002021      SSA, RSS
0097 02176 026213      JMP LL1
0098 02177 162263      LL2 LDA K, I
0099 02201 100400      DIV HIF
02202 002300
0101 02203 100400      MPY HIF

```

Calculation of E_f

Calculation of E_t

Is $E_f \leq E_t$

Store E_t in E

Store E_f in E

Is $E \leq 0$

Calculation of E^2

Strategy of $E > 0$

	02204	002331			
0102	02205	100200		MPY FIF	
	02205	002337			
0103	02207	100400		DIV TND	
	02210	002340			
0104	02211	072330		STA DF	
0105	02212	026237		JMP L7	
0106	02213	062310	LL1	LDA PB	Calculation of E ²
0107	02214	072331		STA EE	
0108	02215	026177		JMP LL2	
0109	02216	100200	L6	MPY E	Calculation of E ²
	02217	002335			
0110	02220	072331		STA EE	
0111	02221	042307		ADA MB	
0112	02222	002021		SSA, RSS	
0113	02223	026253		JMP LL3	
0114	02224	162263	LL4	LDA K, F	Strategy of E < 0
0115	02225	042275		ADA CN2	
0116	02226	100400		DIV HNT	
	02227	002312			
0117	02230	100200		MPY EE	
	02231	002331			
0118	02232	100200		MPY FIF	
	02233	002337			
0119	02234	100400		DIV HTEN	
	02235	002311			
0120	02236	072330		STA DF	
0121	02237	062330	L7	LDA DF	Is new feed-rate
0122	02240	142266		ADA F, I	$\begin{matrix} < \\ > \end{matrix} F_{min}$
0123	02241	172266		STA F, I	
0124	02242	042306		ADA FMIN	
0125	02243	002020		SSA	
0126	02244	026256		JMP L8	
0127	02245	042355		ADA FMAX	Is new feed-rate
0128	02246	002020		SSA	$\begin{matrix} < \\ > \end{matrix} F_{max}$
0129	02247	026260		JMP L9	Store F
0130	02250	062303		LDA FX	Calculation of E ²
0131	02251	172266		STA F, I	
0132	02252	026260		JMP L9	
0133	02253	062310	LL3	LDA PB	
0134	02254	072331		STA EE	
0135	02255	026224		JMP LL4	
0136	02256	062304	L8	LDA FL	F = F _{min}
0137	02257	172266		STA F, I	Store F
0138	02260	002400	L9	CLA	Clear TC
0139	02261	172266		STA K, I	
0140	02262	026027		JMP L0	
0141	02263	020000	K	OCT 20000	
0142	02264	020001	EN1	OCT 20001	
0143	02265	020003	EN3	OCT 20003	
0144	02266	020004	F	OCT 20004	
0145	02267	010000	CH1	OCT 10000	
0146	02270	020000	CH2	OCT 20000	
0147	02271	020000	CH3	OCT 20000	
0148	02272	020000	CH4	OCT 20000	
0149	02273	001777	COB	OCT 01777	
0150	02274	002000	COI	OCT 00200	

0151	02275	000260	CN2	DEC	130
0152	02276	000001	ONE	DEC	1
0153	02277	000002	TWO	DEC	2
0154	02300	000144	HUN	DEC	100
0155	02301	000040	TY	DEC	32
0156	02302	166170	FT	DEC	-5000
0157	02303	005670	FX	DEC	3000
0158	02304	000036	FL	DEC	30
0159	02305	172146	FMAX	DEC	-2970
0160	02306	177742	FNIN	DEC	-30
0161	02307	176600	MB	DEC	-640
0162	02310	001200	PB	DEC	+640
0163	02311	177766	HTEN	DEC	-10
0164	02312	000146	HNT	DEC	102
0165	02313	000000	M1	NOP	
0166	02314	000000	M2	NOP	
0167	02315	000000	M3	NOP	
0168	02316	000000	TS	NOP	
0169	02317	000000	TSS	NOP	
0170	02320	000000	RR	NOP	
0171	02321	000000	R	NOP	
0172	02322	000000	R2	NOP	
0173	02323	000000	P	NOP	
0174	02324	000000	Q	NOP	
0175	02325	000000	Y	NOP	
0176	02326	000000	Z	NOP	
0177	02327	000000	AX	NOP	
0178	02330	000000	DF	NOP	
0179	02331	000000	EE	NOP	
0180	02332	000000	SQR	NOP	
0181	02333	000000	E1	NOP	
0182	02334	000000	E2	NOP	
0183	02335	000000	E	NOP	
0184	02336	000000	E3	NOP	
0185	02337	000017	FIF	DEC	15
0186	02340	002000	TND	DEC	1024
0187				END	

** NO ERRORS*

	ASMB, A, B, L	COMMENTS
0001		
0002	01400	ORG 1400B
0003	01400 006400	CLB
0004	01401 002400	START CLA
0005	01402 171727	STA SA, I
0006	01403 035727	ISZ SA
0007	01404 061727	LDA SA
0008	01405 051730	CPA FA
0009	01406 025410	JMP L1
0010	01407 025401	JMP START
0011	01410 103710	L1 STC 10B, C Read a character
0012	01411 102310	SFS 10B
0013	01412 025411	JMP *-1
0014	01413 102510	LIA 10B
0015	01414 051740	CPA SPACE Space?
0016	01415 025450	JMP L3
0017	01416 051743	CPA PLUS Plus?
0018	01417 025450	JMP L3
0019	01420 051744	CPA PLSC Plus?
0020	01421 025450	JMP L3
0021	01422 051732	CPA SLASH Slash?
0022	01423 025513	JMP L8
0023	01424 051733	CPA CR CR?
0024	01425 025450	JMP L3
0025	01426 051734	CPA LF LF?
0026	01427 025514	JMP L7
0027	01430 051735	CPA LFF
0028	01431 025514	JMP L7
0029	01432 051741	L4 CPA MINUS Minus?
0030	01433 025504	JMP L5
0031	01434 051742	CPA MINC Minus?
0032	01435 025504	JMP L5
0033	01436 011746	L2 AND M Calculate Number N
0034	01437 041737	ADA SP
0035	01440 171736	STA LP, I
0036	01441 001000	ALS
0037	01442 071737	STA SP
0038	01443 001000	ALS
0039	01444 001000	ALS
0040	01445 041737	ADA SP
0041	01446 071737	STA SP
0042	01447 025410	JMP L1
0043	01450 055745	L3 CPB TW Is N < 0
0044	01451 025506	JMP LN
0045	01452 103710	LM STC 10B, C Read a character
0046	01453 102310	SFS 10B
0047	01454 025453	JMP *-1
0048	01455 102510	LIA 10B
0049	01456 051740	CPA SPACE Space?
0050	01457 025450	JMP L3
0051	01460 051743	CPA PLUS Plus?
0052	01461 025450	JMP L3
0053	01462 051744	CPA PLSC
0054	01463 025450	JMP L3
0055	01464 051732	CPA SLASH Slash?
0056	01465 025513	JMP L8
0057	01466 051733	CPA CR CR?

0058	01467	025452		JMP LM	
0059	01470	006400		CLB	116
0060	01471	075737		STB SP	
0061	01472	051734		CPA LF	LF?
0062	01473	025514		JMP L7	
0063	01474	051735		CPA LFF	
0064	01475	025514		JMP L7	
0065	01476	065736		LDB LP	Increment of memory location
0066	01477	045726		ADB HUN	
0067	01500	075736		STB LP	
0068	01501	006400		CLB	
0069	01502	075737		STB SP	
0070	01503	025432		JMP L4	
0071	01504	065745	L5	LDB TW	B=20
0072	01505	025410		JMP L1	
0073	01506	161736	LN	LDA LP, I	N=-N
0074	01507	003004		CMA, INA	
0075	01510	171736		STA LP, I	Store N
0076	01511	006400		CLB	B=0
0077	01512	025452		JMP LM	
0078	01513	102075	L8	HLT 75B	Halt
0079	01514	065731	L7	LDB RP	Calculating Path/ Rad. and Store
0080	01515	045725		ADB WH	
0081	01516	075736		STB LP	
0082	01517	006400		CLB	
0083	01520	075737		STB SP	
0084	01521	061762		LDA TWO	
0085	01522	151747		CPA RG, I	
0086	01523	025530		JMP CIRCL	
0087	01524	061776		LDA THREE	
0088	01525	151747		CPA RG, I	
0089	01526	025530		JMP CIRCL	
0090	01527	025534		JMP LINE	
0091	01530	061747	CIRCL	LDA RG	
0092	01531	041757		ADA FHUN	
0093	01532	071747		STA RG	
0094	01533	025537		JMP PATH	
0095	01534	061747	LINE	LDA RG	
0096	01535	041726		ADA HUN	
0097	01536	071747		STA RG	
0098	01537	161747	PATH	LDA RG, I	
0099	01540	002020		SSA	
0100	01541	003004		CMA, INA	
0101	01542	071753		STA ABX	
0102	01543	100200		MPY ABX	
		01544		001753	
0103	01545	104400		DST SOX	
		01546		001751	
0104	01547	161747		LDA RG, I	
0105	01550	071772		STA X	
0106	01551	061753		LDA ABX	
0107	01552	100200		MPY TWO	
		01553		001762	
0108	01554	041750		ADA ONE	
0109	01555	104400		DST TX	
		01556		001764	
0110	01557	061747		LDA RG	

0111	01560	041726	ADA HUN	117
0112	01561	071747	STA RG	
0113	01562	161747	LDA RG, I	
0114	01563	002020	SSA	
0115	01564	003004	CMA, INA	
0116	01565	071754	STA ABY	
0117	01566	100200	MPY ABY	
	01567	001754		
0118	01570	104400	DST SQY	
	01571	001755		
0119	01572	161747	LDA RG, I	
0120	01573	071773	STA Y	
0121	01574	061754	LDA ABY	
0122	01575	100200	MPY TWO	
	01576	001762		
0123	01577	041750	ADA ONE	
0124	01600	104400	DST TY	
	01601	001766		
0125	01602	061747	LDA RG	
0126	01603	041726	ADA HUN	
0127	01604	071747	STA RG	
0128	01605	161747	LDA RG, I	
0129	01606	002020	SSA	
0130	01607	003004	CMA, INA	
0131	01610	071777	STA ABZ	
0132	01611	100200	MPY ABZ	
	01612	001777		
0133	01613	104400	DST SQZ	
	01614	002000		
0134	01615	161747	LDA RG, I	
0135	01616	071774	STA Z	
0136	01617	061777	LDA ABZ	
0137	01620	100200	MPY TWO	
	01621	001762		
0138	01622	041750	ADA ONE	
0139	01623	104400	DST TZ	
	01624	001770		
0140	01625	061754	LDA ABY	
0141	01626	003004	CMA, INA	
0142	01627	041753	ADA ABX	
0143	01630	002020	SSA	
0144	01631	025654	JMP P	
0145	01632	061777	LDA ABZ	
0146	01633	003004	CMA, INA	
0147	01634	041753	ADA ABX	
0148	01635	002020	SSA	
0149	01636	025662	JMP R	
0150	01637	104200	DLD SQY	
	01640	001755		
0151	01641	100400	DI V TX	
	01642	001764		
0152	01643	171736	STA LP, I	
0153	01644	104200	DLD SQZ	
	01645	002000		
0154	01646	100400	DI V TX	
	01647	001764		
0155	01650	141736	ADA LP, I	

0156	01651	041753		ADA ABX	
0157	01652	171736		STA LP, I	118
0158	01653	025716		JMP S	
0159	01654	061777	P	LDA ABZ	
0160	01655	003004		CMA, INA	
0161	01656	041754		ADA ABY	
0162	01657	002020		SSA	
0163	01660	025662		JMP R	
0164	01661	025700		JMP Q	
0165	01662	000000	R	NOP	
0166	01663	104200		DL D SQX	
	01664	001751			
0167	01665	100400		DI V TZ	
	01666	001770			
0168	01667	171736		STA LP, I	
0169	01670	104200		DL D SQY	
	01671	001755			
0170	01672	100400		DI V TZ	
	01673	001770			
0171	01674	141736		ADA LP, I	
0172	01675	041777		ADA ABZ	
0173	01676	171736		STA LP, I	
0174	01677	025716		JMP S	
0175	01700	000000	Q	NOP	
0176	01701	104200		DL D SQX	
	01702	001751			
0177	01703	100400		DI V TY	
	01704	001766			
0178	01705	171736		STA LP, I	
0179	01706	104200		DL D SQZ	
	01707	002000			
0180	01710	100400		DI V TY	
	01711	001766			
0181	01712	141736		ADA LP, I	
0182	01713	041754		ADA ABY	
0183	01714	171736		STA LP, I	
0184	01715	025716		JMP S	
0185	01716	061731	S	LDA RP	
0186	01717	002004		INA	
0187	01720	071731		STA RP	
0188	01721	071736		STA LP	
0189	01722	041726		ADA HUN	
0190	01723	071747		STA RG	
0191	01724	025410		JMP L1	
0192	01725	005000	WH	OCT 5000	
0193	01726	000400	HUN	OCT 400	
0194	01727	030000	SA	OCT 30000	
0195	01730	036000	FA	OCT 36000	
0196	01731	030000	RP	OCT 30000	
0197	01732	000257	SLASH	OCT 257	
0198	01733	000215	CR	OCT 215	
0199	01734	000012	LF	OCT 12	
0200	01735	000212	LFF	OCT 212	
0201	01736	030000	LP	OCT 30000	
0202	01737	000000	SP	NOP	
0203	01740	000240	SPACE	OCT 240	
0204	01741	000055	MINUS	OCT 55	

0205	01742	000255	MINC	OCT	255
0206	01743	000053	PLUS	OCT	53
0207	01744	000253	PLSC	OCT	253
0208	01745	000024	TW	DEC	20
0209	01746	000017	M	OCT	17
0210	01747	030400	RG	OCT	30400
0211	01750	000001	ONE	DEC	1
0212	01751	000000	SQX	NOP	
0213	01752	000000	AD1	NOP	
0214	01753	000000	ABX	NOP	
0215	01754	000000	ABY	NOP	
0216	01755	000000	SQY	NOP	
0217	01756	000000	AD2	NOP	
0218	01757	002000	FHUN	OCT	2000
0219	01760	000000	JSQ	NOP	
0220	01761	000000	KSQ	NOP	
0221	01762	000002	TWO	DEC	2
0222	01763	000000	I	NOP	
0223	01764	000000	TX	NOP	
0224	01765	000000	AD3	NOP	
0225	01766	000000	TY	NOP	
0226	01767	000000	AD4	NOP	
0227	01770	000000	TZ	NOP	
0228	01771	000000	AD5	NOP	
0229	01772	000000	X	NOP	
0230	01773	000000	Y	NOP	
0231	01774	000000	Z	NOP	
0232	01775	000000	TI	NOP	
0233	01776	000003	THREE	DEC	3
0234	01777	000000	ABZ	NOP	
0235	02000	000000	SQZ	NOP	
0236	02001	000000	AD6	NOP	
0237				END	

119

** NO ERRORS*

BIBLIOGRAPHY

1. Olesten, N. O., Numerical Control, John Wiley and Sons, Inc., 1970.
2. The Expanding World of NC, Proceedings of the Ninth Annual Meeting and Technical Conference Numerical Control Society, April 17-19, 1972, Chicago, Illinois.
3. Childs, James, J., Principles of Numerical Control, Industrial Press Inc., New York, N. Y., 1965.
4. Thornhill, Robert, B., Engineering Graphics and Numerical Control, McGraw-Hill, 1967.
5. Numerical Control for Tomorrow, Frontiers in Manufacturing Technology, Vol. IV., Institute of Science and Technology, The University of Michigan, Ann Arbor, Michigan, 1969.
6. Hewlett Packard 2100A Computer Reference Manual, 1971.
7. Koren, Y., Shani, A., Ben-Uri., Numerical Control of a Lathe, I.E.E.E., Transactions on Industry and General Applications, Volume 1GA-6, Number 2, March/April, 1970.
8. Mueller, P. A., Trainable Adaptive Control for Automated Machining, Manufacturing/Numerical Control Systems Technical Paper, Society of Manufacturing Engineers, 1972.
9. Seth, M. K., Direct Digital Control of Machine Tools (Ph.D Thesis), Wisconsin, Department of Mechanical Engineering, University of Wisconsin, 1972.
10. Hardy, James, M., Retrofitting Machine Tools, Ann Arbor, Michigan: Industrial Development Division of the Institute of Science and Technology, University of Michigan, 1967.
11. Evans, L., Computerized Numerical Control Revised, Control Engineering, August, 1972.