

627.0285

54/86

CAPE/KAAPSE TECHNIKON
LIBRARY ITEM: 86000054



54/86

KAAPSE TECHNIKON
CAPE
LIBRARY / BIBLIOTEEK

CAPE PENINSULA UNIVERSITY OF TECHNOLOGY
LIBRARY AND INFORMATION SERVICES
BELLVILLE CAMPUS

TEL: (021) 959-6210

FAX: (021) 959-6109

Renewals may be made telephonically.

This book must be returned on/before the last date shown.

Please note that fines are levied on overdue books

627.0285
54/86

KUIPERS, J.J.J.

Automation of
coastal engineering
modelling facilities

1983

Donation

AUTOMATION OF COASTAL ENGINEERING MODELLING FACILITIES

by

Jan Jacob Kuipers

Thesis submitted in partial compliance with the requirements
for the Diploma in Technology in the
Department of Electrical Engineering (light current)
at the Cape Town Technikon

ELECTRONICS AND INSTRUMENTATION GROUP
NATIONAL RESEARCH INSTITUTE FOR OCEANOLOGY
COUNCIL FOR SCIENTIFIC AND INDUSTRIAL RESEARCH

Stellenbosch, South Africa
September 1983

ACKNOWLEDGEMENTS

Without the help and assistance of the following people, this thesis may never have materialised.

(1) The CSIR, and in particular, the Directorate of NRIO, Mr F P Anderson and Mr C C Stavropoulos, for giving me the opportunity to do this research project.

(2) My divisional head, Mr S G Holroyd, for his guidance, encouragement and support.

(3) For the members of the Electronic Systems Division for their help, and in particular, Pat Moultray for her excellent drawings.

(4) Monica Strauss for transforming my illegible handwriting into fine typing.

(5) Felix Lancaster for his editing.

(6) Mr Lubbe from Altronics for his assistance.

(7) My wife for her support and encouragement.

CONTENTS

Page

ACKNOWLEDGEMENTS

LIST OF FIGURES AND DRAWINGS

1.	INTRODUCTION	1
2.	THE MODEL HALL	3
3.	THE MODEL HALL DATA ACQUISITION SYSTEM (MHDAS)	5
3.1	The MHDAS in Use	5
3.2	MHDAS Operating System, STANOP	7
3.3	Interfacing Instruments to the MHDAS	9
3.4	MHDAS Command and Data Transfer	10
4.	PROBE CALIBRATOR	17
4.1	Introduction	17
4.2	General Description	17
4.3	Requirements to Interface to the MHDAS	18
4.4	Theory of Operation	19
5.	MEASUREMENT PLATFORM CONTROLLER	26
5.1	Introduction	26
5.2	General Description	26
5.3	Requirements to Interface to the MHDAS	28
5.4	Theory of Operation	30
5.5	Position-Fixing Module	44
6.	TROLLEY CONTROLLER	50
6.1	Introduction	50
6.2	General Description	50
6.3	Requirements to Interface to the MHDAS	51
6.4	Theory of Operation	51
6.5	MK2 Speed Controller	54
7.	CONSIDERATIONS IN THE DESIGN OF NEW SYSTEMS	59
7.1	Guide to the Writing of a Test-control Program	59
7.2	Programming Remote Instruments	61
8.	TOWARDS AUTOMATIC MODEL TESTING	65
8.1	The Instruments in Use	65
8.2	User Response	66

CONTENTS (continued)

	<u>Page</u>
8.3 Future Developments	67
8.4 Costs and Justification	72
8.5 Conclusion	74
REFERENCES	75
APPENDIX A: MPC/TROLLEY CONTROLLER PC BOARDS	
APPENDIX B: THE NRIO MICROPROCESSOR DEVELOPMENT FACILITIES	
APPENDIX C: DATA SHEETS	
APPENDIX D: PROBE CALIBRATOR PROGRAM LISTING	
APPENDIX E: MPC/TROLLEY CONTROLLER PROGRAM LISTING	
APPENDIX F: POSITION-FIXING MODULE PROGRAM LISTING AND FLOW DIAGRAMS	
APPENDIX G: MK2 SPEED CONTROLLER PROGRAM LISTING AND FLOW DIAGRAMS	

LIST OF FIGURES AND DRAWINGS

	<u>Page</u>
Figure 2.1 Layout of Coastal Engineering Facilities	4
3.1 MHDAS Data Routing Structure	6
3.2 Typical STANOP Control Loop	14
4.1 Probe Calibrator Control Unit	15
4.2 Probe Calibrator Jig	16
4.3 Calibrator Prompts	23
5.1 Measurement Platform Controller	25
5.2 MPC Prompts	40
6.1 Trolley Controller and Associated Instrumentation	49
6.2 Trolley Prompts	52
8.1 Wave Generator Linearity Test	69
8.2 Wave Height Test	70
Drawing No HB81001 MPC block diagram	A7
HB81002 MPC X and Y position finder	A8
HB81003 MPC interface 1	A9
HB81004 MPC interface 2	A10
HB81005 MK1 motor controller	A11
HB81006 MPC motor power supply	A12
HB81007 MPC position plate and sensor	A13
HB81008 MK2 motor controller timing diagram	A14
HB81009 MPC wiring diagram	A15
MA79001 CPU board	A16
MA79004 PIA board	A17
MA79007 PTM board	A18
MA79009 RS232, TTY interface	A19
MA79013 STANOP PIC board	A20
MA79015 Function board	A21
MA79016 System ID board	A22
MA79019 Hexadecimal keyboard module	A23
MA79020 12 digit led display module	A24
MA79022 STANOP 16 k EPROM/RAM board	A25

1. INTRODUCTION

The Coastal Engineering and Hydraulics Division of the National Research Institute for Oceanology (NRIO) undertakes research projects so as to be able to supply specialised expertise and consultancy services in coastal engineering with special reference to the needs in South Africa.

Physical models of harbours, beaches, etc. are built at NRIO and are used mainly for sediment and breakwater studies. To collect the vast amount of data in the tests done on these models a Model Hall Data Acquisition System, MHDAS, was developed (Holroyd, 1980) to provide a data-logging facility and also to provide the control signals required to co-ordinate model operation and thus permit automatic model testing.

A model hall automation plan was launched by the Electronic Systems Division at NRIO to speed up the time-consuming task of collecting data from the model test instrumentation. In order to achieve the aim of automating model control and testing, it was necessary to interface instrumentation and control systems to the MHDAS. The most logical way of doing this was to regard these instruments and control systems as extensions of the MHDAS as described by Holroyd (1980), and therefore to have these units based on the standard hardware and software of the MHDAS microcomputer units. This thesis will form part of this overall model hall automation plan. The main objectives covered by this thesis will be:

- (a) To update and convert existing first-generation microprocessor-controlled instrumentation, namely, the probe calibrator, in order to interface it to the MHDAS.
- (b) To interface the experimental measurement platform controller (MPC) to the MHDAS.
- (c) To add from the design stage all the necessary software and hardware to automate newly developed instrumentation, namely, the trolley controller.

In this thesis a brief overview of the data-acquisition system will be given with special emphasis on how control signals are handled and routed within the system between controllers and instrumentation. The theory of operation of the probe calibrator will not be repeated, and only an explanation of the hardware and software changes made will be given. Since no technical report exists, a complete description of the theory of operation of the measurement platform controller (MPC) will be given. Using the MPC operating system as a starting point attempts will be made to adapt and to modify the MPC control program so as to also control movement of the trolley. An explanation of the adaptive changes made to the MPC to enable it to control the trolley will be given.

2. THE MODEL HALL

Modeling facilities for coastal engineering are housed in two large model halls which are divided into a number of smaller model basins and flumes (Figure 2.1). The instrumentation developments in this thesis will be tested and used in the Granger Bay basin and the 1 m tilting flume. The Granger Bay model will shortly be replaced by a model of Richards Bay and will be equipped with the newly acquired irregular wave generators. The aim is to make this the first fully automated model basin.

The Granger Bay model is equipped with two mobile instrumentation platforms which are suspended from an overhead mobile gantry. These platforms can travel in both the X and Y horizontal axes at about 0,5 m above the water surface. The platforms are usually moved to measuring points on a pre-determined grid of the model where probes are placed into the water to measure wave heights or any other required model variables.

In the tilting flume wave height or current measurements are done from a mobile platform which runs on rails along the top edges of the flume. Tests are usually done from the trolley when it moves at a pre-determined constant speed or when it is stationary anywhere along the length of the flume.

Each model or flume is usually managed and controlled from a conveniently placed local control room.

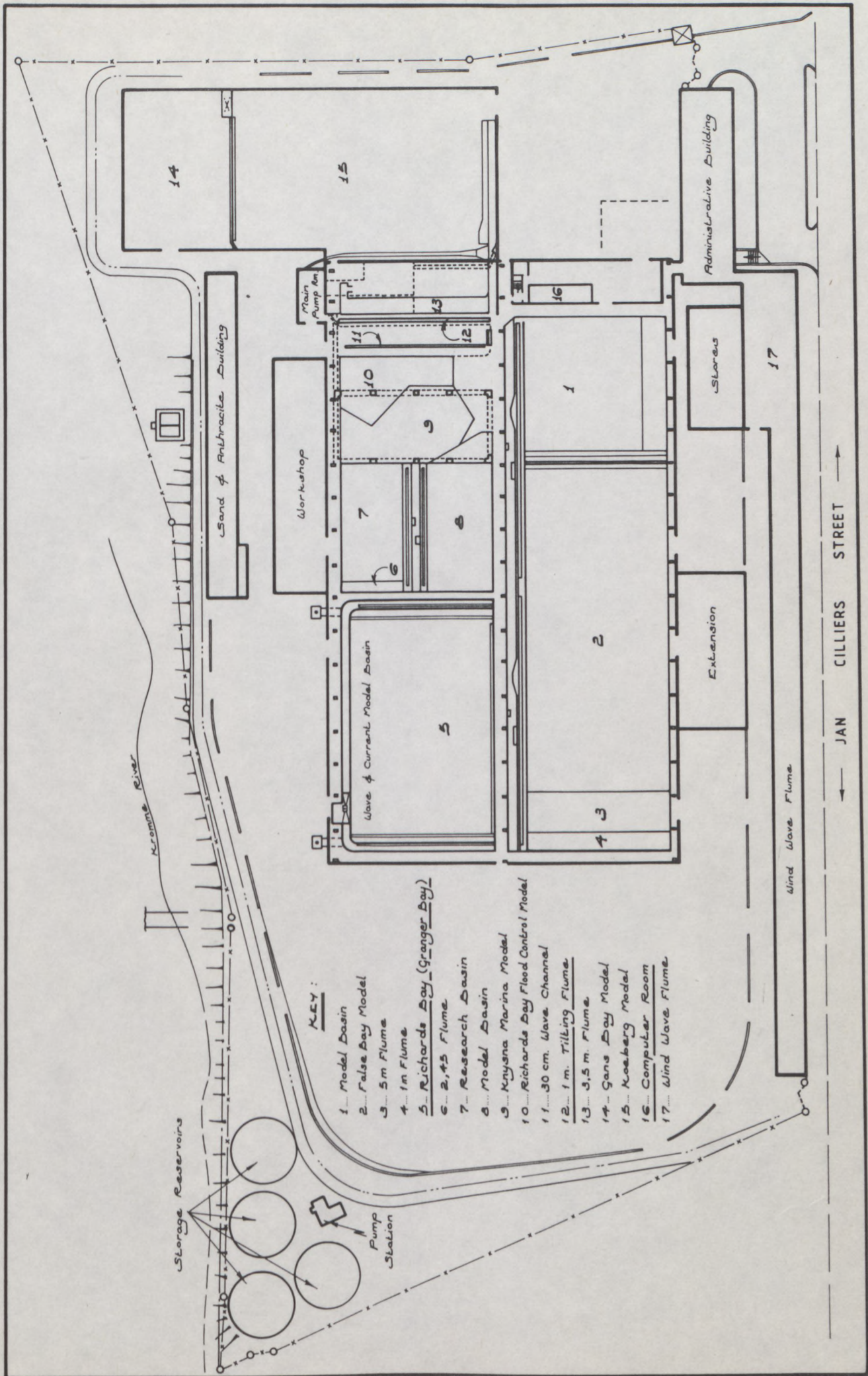


FIGURE 2.1 LAYOUT OF COASTAL ENGINEERING FACILITY EA 80 001

3. THE MODEL HALL DATA ACQUISITION SYSTEM (MHDAS)

3.1 The MHDAS in Use

The coastal engineering model hall data-acquisition system (MHDAS) was developed by S G Holroyd, head of the Electronic Systems Division at NRIO (see Holroyd, 1980). The main aim of this system is to provide a data-collection system which links the instrumentation mounted on the measurement platforms, via model control consoles to the minicomputer (see Figure 3.1). Data can either be stored locally on a model or can be passed onto another machine further up the hierarchical chain. The various machines of the MHDAS are connected in a tree-type configuration and software routines were written to sort out the data - transfer protocol between machines and any contentions that may arise. A major requirement of this data highway is that all the units from instruments up to the minicomputer interface should have the same operating system and hardware configuration. This ensures a high degree of flexibility and expandability. As the facility now exists, commands and data files can be transferred between machines of the MHDAS (see Figure 3.1) and they may also be used to transfer control signals to automate and control the instrumentation used in model testing.

Although the main control and data path that will be used is that between the model control console, slave unit and test instrumentation, control signals may also originate from the minicomputer or from a program running in any unit along this control link. The model control console is housed in the local model control room, and usually consists of a standard DAS microcomputer unit with a keyboard, VDU, printer, cassette data logger (CDL) and cassette reader. From this console the operator will be able to control model components such as the irregular wave generators, move the measurement platforms, control the test instrumentation, run tests, collect data and store them on any selected storage device. Commands to the model instrumentation are given via a slave unit which is mounted with

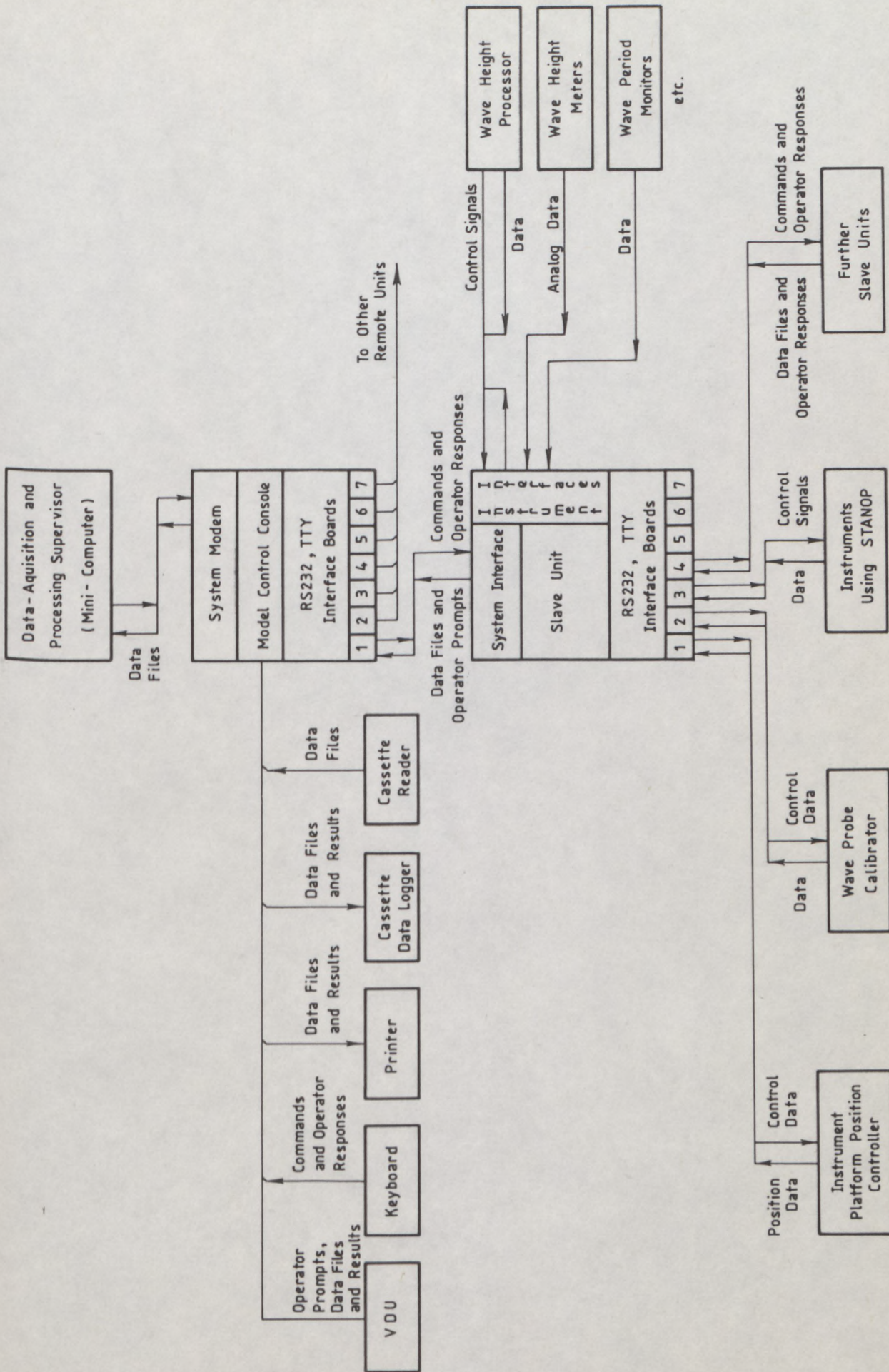


FIG. 3.1

MHDAS DATA ROUTING STRUCTURE

the model test and control instrumentation on the measurement platforms. This slave unit has its own buffer memory which enables it to perform certain tasks independently and thus relieve the control console of some of its controlling tasks.

One of the NRIO microcomputer system cards, the RS232 TTY interface card, is used for data transfer between any two units. This card is based on the Motorola MC6850 ACIA and the baud rate is switch-selectable between 100 and 9600. As only the transmit data and receive data lines of the RS232 interface are used, the data link need be only a screened twin wire where the screen acts as a common.

3.2 MHDAS Operating System, STANOP

The common operating system used by all the machines in the MHDAS is called STANOP. The main task of this operating system is to provide the structure for the collection of data and the handling of data files. This includes the supervision of buffer memory and the communication with other units. The main STANOP control loop, SYSRUN, is shown in Figure 3.2. STANOP contains the following routines:

(a) Control routines to perform operations such as:

- (i) Display system prompts.
- (ii) Get commands from the operator.
- (iii) Check for inputs from other units.
- (iv) Handle communication protocol.
- (v) Interpret commands and execute them.
- (vi) Run STANOP (main control loop, SYSRUN).

(b) Data-handling routines to:

- (i) Create new data files.
- (ii) Index and store the captured data.
- (iii) Output the data files.

(c) Input/output routines to drive the following standard devices:

- (i) Keyboard.
- (ii) VDU.
- (iii) Printer.
- (iv) Cassette data logger.
- (v) Cassette reader.
- (vi) MODEMS and ACIAS.
- (vii) Analogue-to-digital converter.
- (viii) Digital-to-analogue converter.

(d) General routines which include:

- (i) Code conversion routines (BCD/BIN/ASCII).
- (ii) Character editing routines.
- (iii) Delay routines.
- (iv) Display routines for standard prompts (e.g. output TRY AGAIN).
- (v) Index register storage routines, etc.

(e) Set-up routines to set up the standard devices used by STANOP (system PIA, PTM, etc.). Apart from setting up the standard devices, these routines also determine system parameters or variables such as buffer memory size and the machine hierarchy.

(f) Testing and debugging routines to permit *in situ* testing of interfaces, data transmission, small additional routines, etc. The instruction set of this routine is similar to that for EXBUG (refer to Exorciser Manual).

(g) STANOP makes use of explicit prompts and messages in order to simplify the use of MHDAS and to keep the operator informed, as fully as possible, as to the status of all the units at his command. The operator/machine dialogue has been kept as simple as possible and user friendly to reduce the amount of training required to operate the system.

3.3 Interfacing Instruments to the MHDAS

A standard system under STANOP control has the following cards: ID card, STANOP memory, STANOP PTM, STANOP PIC and System PIA. When remote instrumentation is being automated the cards used are based on the standard MHDAS with extra routines added to STANOP to perform the special requirements of that instrument. Such extra routines are referred to as the external program. Apart from running its normal setup routines, STANOP will run setup routines of any instruments under its control, execute any valid STANOP commands, test for any data-transfer requests and will run any external instrument program under its control. The presence of an external program is indicated by the ID board (bytes 11 and 12) which points to the start address of the non-standard external program. The main advantage of having a machine under STANOP control is that all routines used in STANOP (communication, code conversion, testing and debugging routines, etc.) are at the disposal of the external program. The testing and debugging routines in STANOP can prove to be very useful for *in situ* evaluation of an external program.

Any external program added to a machine running under STANOP must begin with the following instructions at the start of that program:

```
JMP XX (address of external program command interpreter)
JMP YY (address of external program set-up routine)
LDX ZZ (address of external program command listing to output on
        VDU)
JSR VDUOUT
RTS
JMP UU (address of external program main control routine or
        loop).
```

These instructions, together with the address set on the ID card will permit STANOP to include and perform these external routines.

3.4 MHDAS Command and Data Transfer

The MHDAS responds to a set of standard commands (run the A to D converter, output a data file, etc.) which define the various functions of the operating system (refer to page 39 of Holroyd, 1980). STANOP can also handle and route the standard STANOP commands or any additional, external, non-STANOP commands if they suit the specified format. The STANOP and non-STANOP command set plus the format used to transfer these commands to a remote unit is displayed on the VDU by entering on the keyboard the STANOP list (L) command.

The remote intelligent unit will be autonomous with its own control program (external program) and, having received a command from the model control console, it will be able to perform the requested function independently until stopped by the local control program or by the model control console.

Commands entered at the model control console are stacked into an 11-byte command buffer. If byte 5 is not zero the buffer will be transferred via the RS232 interface pointed to by the contents of byte 5. The command buffer has the following format:

- Byte 0 - defining the requested operation
- Byte 1 - data file number
- Byte 2 - data-handling options or command options
- Byte 3 - data-handling options or command options
- Byte 4 - data-handling options or command options
- Byte 5 - remote unit number, that is, the local RS232 interface number that will transfer the command buffer
- Byte 6 - Nested remote unit number if the command buffer be passed to a still lower order machine
- Byte 7 - Nested remote unit number if the command buffer be passed to a still lower order machine
- Byte 8 - Nested remote unit number if the command buffer be passed to a still lower order machine
- Byte 9 - Forced remote unit destination
- Byte 10 - General purpose flag.

To transfer data and commands from one unit to another the machine requesting the data transfer will output a single character to the selected RS232 interface. This character will define the type of data transfer being requested. If the second unit is able to handle the data transfer, it will respond with a 'Y' (yes) after which the data will be transferred according to the predetermined procedure. Valid transfer request characters are C, P, I, V, S, R, K, T, Z and EOT. These characters are defined on page 44 of Holroyd (1980), but those most often used are

- C - request to transfer command buffer.
- P - request to transfer prompt buffer.
- S - request to transfer a character string to be displayed on VDU.
- K - request to check if a character has been entered on the keyboard.

To explain how a command is routed from the model control console via a slave unit to a remote machine the following description is given, assuming all the units in the data link have been set up and are controlled by the main STANOP control loop, SYSRUN. A condensed flow diagram of SYSRUN to support the description is given in Figure 3.2.

(a) Operation inside the model control console

- (i) Do subroutine COMDIN. This subroutine will be performed only when the particular MHDAS unit has a VDU present and will check if a command is being entered on the keyboard (subroutine KBDCHK checks if a keyboard entry has been made and if so gets the character). The command characters are entered and stacked into the 11-byte command buffer until it is full or the return key (CR) is depressed.
- (ii) Do subroutine COMINT. This subroutine will execute STANOP or non-STANOP (if bytes 11 and 12 of the ID card

contain an external program address) commands by jumping to the relevant subroutine. If remote units are present in the command, the command is output to the remote unit via a RS232 interface indicated by the contents of byte 5 of the command buffer. Before the command is transferred, byte 5 is deleted and the remaining nested remote unit numbers moved one up.

- (iii) Do subroutine CHECKO. The subroutine is performed to sort out communication protocol when a command must be transferred to a remote unit. Before it outputs the data transfer request-character it checks and services any data transfer request at the input of that RS232 interface. It will then output the character (C - to transfer the command buffer) and attempt to make contact by waiting for a 'Y' response. If after 6,5 s no contact has been made, the operator will be asked whether he wishes to continue to try to make contact. When contact is made, the command buffer is transferred via the RS232 interface and then cleared.

(b) Operation in the slave unit

- (i) Do subroutine CMODEM. This is a subroutine called by SYSRUN to poll the RS232 interfaces and check if there have been any data transfer requests from other units. If a command buffer data transfer request character 'C' has been detected, subroutine CHECKC will transfer the command buffer from the next higher unit in the hierarchical chain to the slave's command buffer (subroutine CHECKC services the system RS232 interface while CHECKI services all other RS232 interfaces. Commands can be entered into a remote unit only via the system RS232 interface).
- (ii) Do subroutine COMINT. The events that occur and follow this routine are explained in steps (ii) and (iii) of the model control console.

(c) Operation in an instrument

- (i) Do subroutine CMODEM as explained in step (i) of the above slave unit.
- (ii) Do subroutine COMINT. As byte 5 of the command buffer is clear this routine will execute any valid STANOP or remote unit command. Bytes 11 and 12 of the ID card will contain the starting address of the external program. The external program starts with the six specified STANOP instructions as discussed in Section 3.3. The first instruction causes program control to jump to the external program command interpreter.

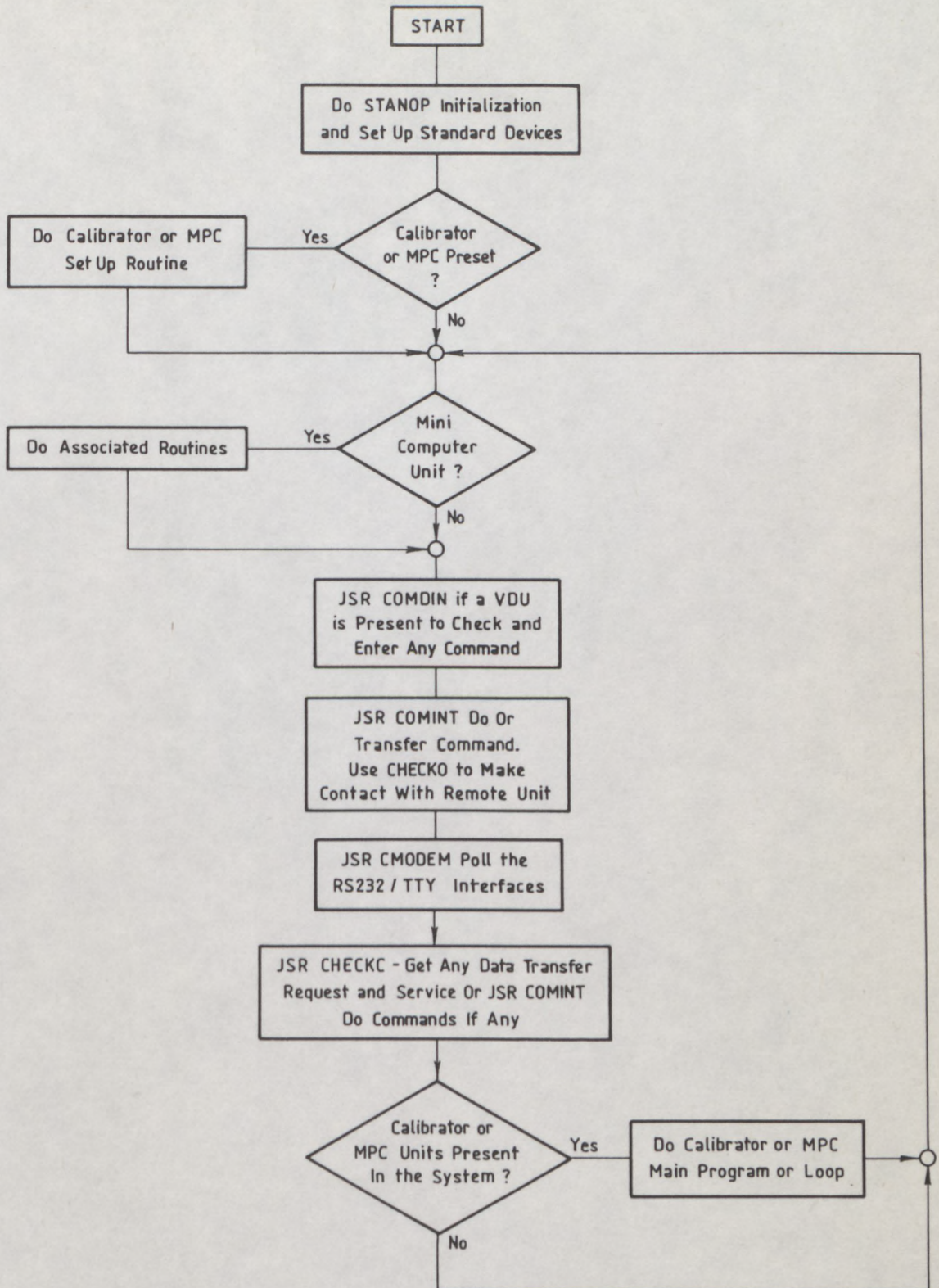
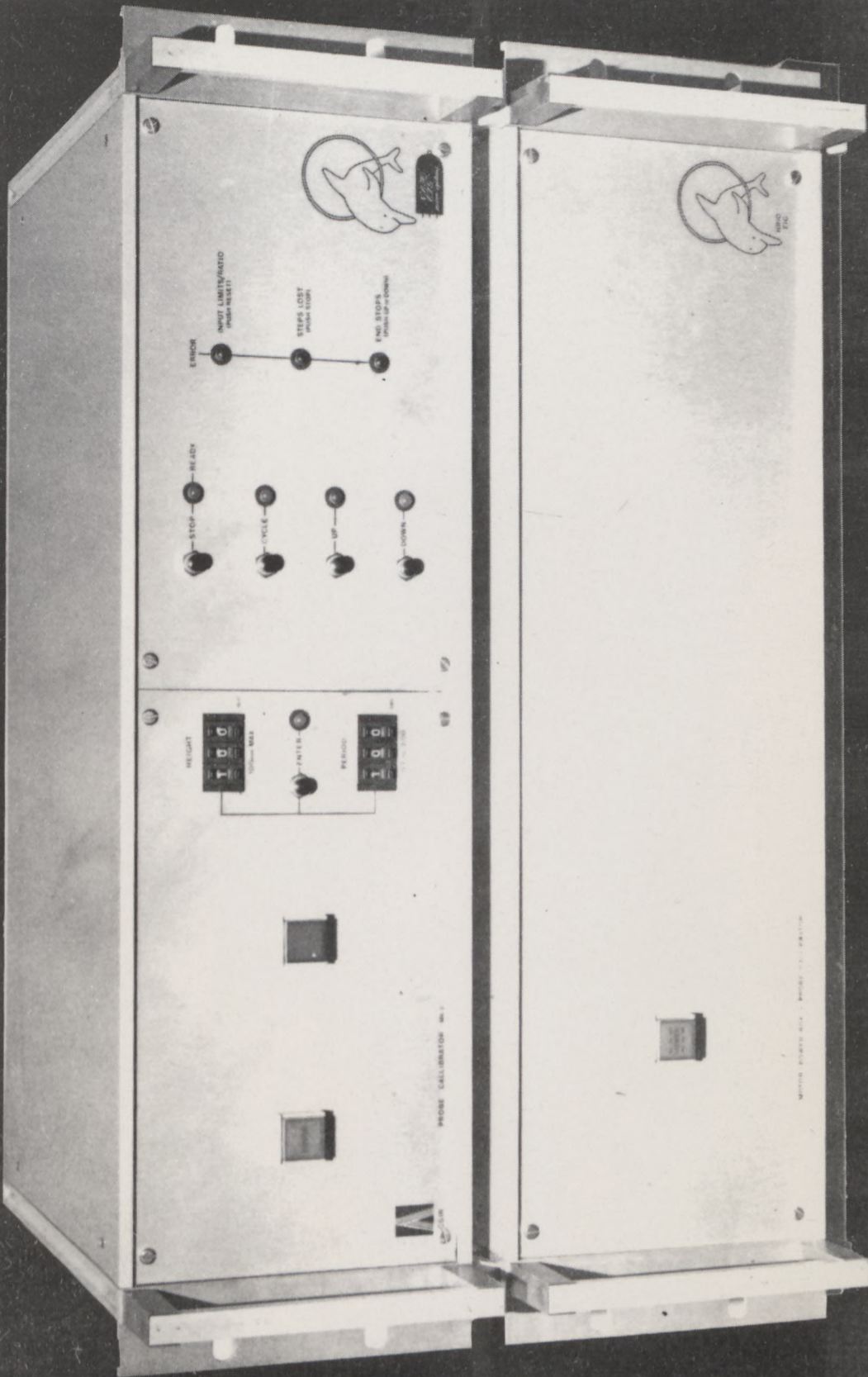
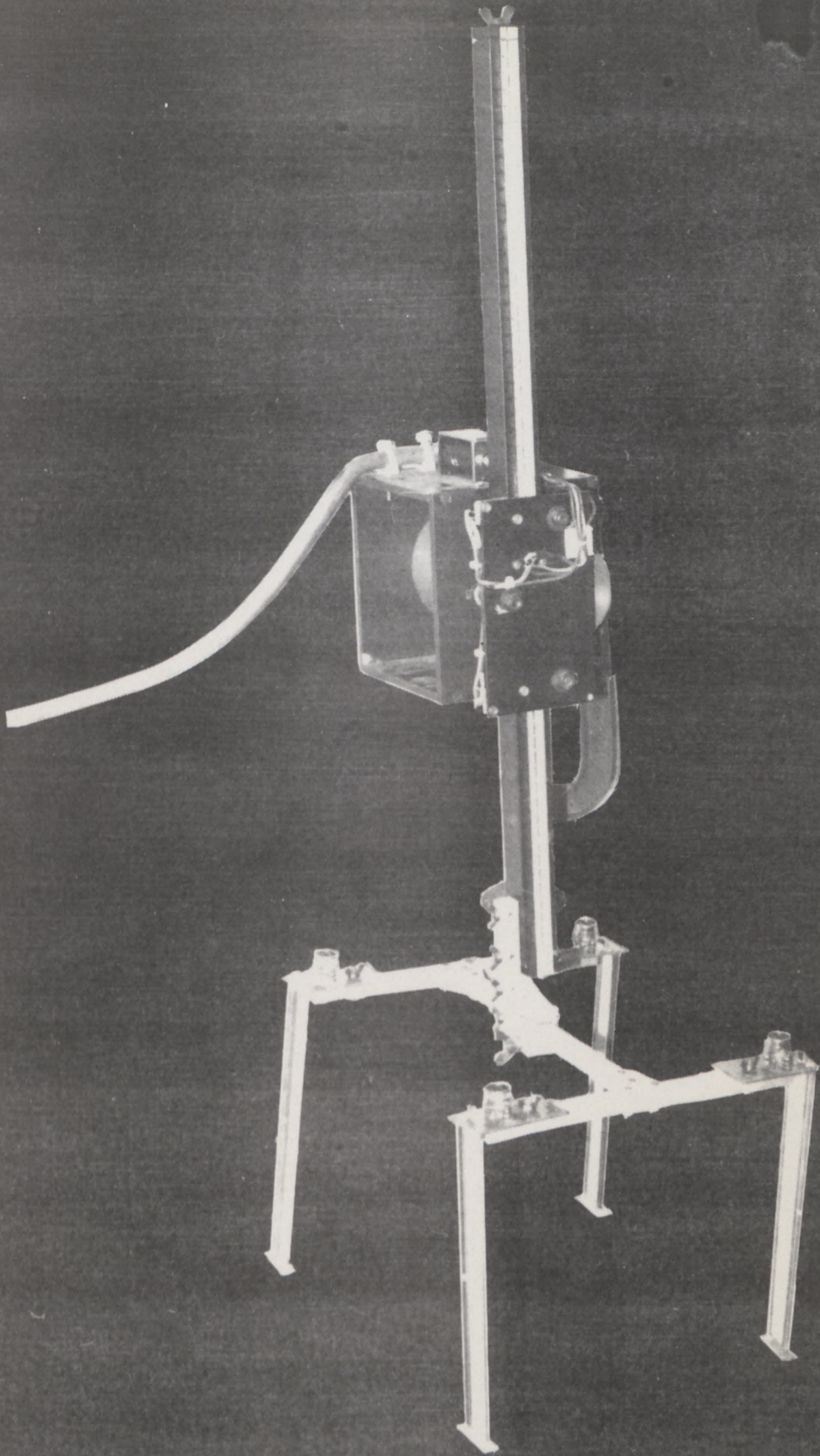


FIG. 3-2

STANOP CONTROL LOOP



Probe Calibrator Control Unit



Probe Calibrator Jig

4. PROBE CALIBRATOR

4.1 Introduction

The probe calibrator (Kuipers, 1982) is based on the microcomputer system (Holroyd, 1982) and was developed to facilitate the calibration of wave-height probes. This unit is used together with a wave height processor, WHP (Holroyd, 1982) which makes use of four parallel resistance probes to measure waves in coastal engineering models. The gain of each of the four channels of the WHP can be adjusted to compensate for the differences in probe responses.

The adjustments are made while the probes are moved sinusoidally up and down in still water to simulate waves of a given period and amplitude. It is the task of the probe calibrator to generate these sinusoidal motions to selectable period and amplitude requirements. The unit can function either as a stand-alone unit or as a remotely controlled (automated) instrument which can tie into the model hall DAS.

4.2 General Description

The probe calibrator consists of a calibrator control unit, motor supply unit and a motor/probe jig assembly, see Figures 4.1 and 4.2.

The calibrator control unit has front panel pushbuttons with which to drive the measuring probes either up, down or sinusoidally up and down for probe calibration. The wave period and amplitude of the simulated wave are set on front panel-mounted thumb-wheel-switches and selected values are then entered to calculate a cycle table via the ENTER push-button. Front panel light emitting diodes (leds) are used to indicate the various operating modes of the unit.

The motor supply section is housed in a separate unit because much heat and electrical noise are generated by the switching

circuitry supplying the DC stepping motor. The jig shaft onto which the probes are fastened is moved by a tensioned cable wound around a pulley on the motor shaft. Warning leds will indicate if slip occurs between the pulley and cable or when the jig has reached either its bottom or top limit.

4.3 Requirements to Interface to the MHDAS

When the probe calibrator was initially developed, an extra PIA card was included to facilitate remote or automated control. Subsequently the model hall data-acquisition system (MHDAS) was developed which has data transfer routines that made communications between machines using the same operating system easier. To standardise all equipment, rather than use the PIA method, it was decided to incorporate STANOP into the probe calibrator, thereby permitting it to be interfaced to the MHDAS. Using the STANOP transfer routines the probe calibrator operation could then be automated. This required that the standard operating system, STANOP, with its associated set of cards, be installed into the existing unit.

The STANOP cards used by this unit are:

- (i) RS232/TTY interface to communicate with units in the MHDAS.
- (ii) RAM/EPROM board to house the STANOP program.
- (iii) ID card. Bytes 11 and 12 hold the start address of the probe calibrator external program.
- (iv) STANOP shares the existing PTM.
- (v) The STANOP-PIC card is omitted due to lack of space, with the result that the debugging facilities in STANOP cannot be fully utilised. The interrupt vectors also have to be programmed into the top STANOP EPROM as the 128 bytes of RAM at the top of the memory map are not present.

The main software changes and additions to the existing probe calibrator program as discussed in Section 3.3 and in the MHDAS (Holroyd, 1980) are:

- (i) The addition of a command interpreter routine COMINT so that commands generated at the model control console can be understood by the probe calibrator.
- (ii) Changed the existing setup routine which was pointed to by the restart vector to be a separate subroutine called SETUP.
- (iii) VDU prompts were added to communicate with the operator.
- (iv) The main probe calibrator control program changed to be a subroutine (ENT) and serviced by STANOP as an auto command.
- (v) The addition of a list of STANOP system equates to provide the addresses of STANOP routines used by the probe calibrator. (VDUOUT to output a character string to the VDU; KBDCHCK to check if a keyboard entry has been made, etc.)

4.4 Theory of Operation

With the required changes made to the probe calibrator control unit the theory of operation is as follows:

(a) Stand-alone mode

To the operator this instrument still functions as it did before the STANOP operating system was added, that is, the front panel pushbuttons will still evoke the same response. Internally, the probe calibrator control program has become a subroutine of the main operating system control loop SYSRUN. As bytes 11 and 12 of the ID board are set, SYSRUN will perform the probe calibrator setup routine SETUP (see Appendix D), and also the main

stand alone control loop ENT (see Appendix D). Routine ENT checks the front panel push-buttons and executes the command, if any, by jumping to the relevant subroutine. A flag (FLA) is set in this routine to determine whether the command has been generated locally or at the model control console. This flag is used to disable all VDU outputs and keyboard input routines since the same calibrator control routines are used for both internal and external (MHDAS) control. FLA is always cleared before returning to STANOP.

(b) MHDAS controlled mode (automatic model testing)

After the addition of this new mode the operator is able to control the probe calibrator from a remote model control console from where commands are entered via a keyboard. These commands are transferred to the remote probe calibrator under STANOP control with the aid of the STANOP transfer routines as explained in Section 3.4. Since bytes 11 and 12 on the ID board of the remote unit are set, STANOP is aware of the external program of the probe calibrator and will jump to the command interpreter routine COMINT (see Appendix D), where the commands are interpreted and reacted upon as if these commands had come from the front panel push-buttons.

The operation field of the command buffer, that is, byte 0 of the 11 byte command buffer is used as a recognition flag and must always contain a 'K' so that the probe calibrator can distinguish if the command is directed to it or not. It was decided to do this to allow various instruments to have the same command characters in the data-handling options field.

The data-handling options field of the command buffer (bytes 2, 3 and 4) contain the actual command. With the probe calibrator only byte 2 is used and may contain either characters U (up), D (down), C (cycle probes) or R (display or enter new period and amplitude values). Any other character will be ignored. Figure 4.3(a) indicates a typical command listing when a probe calibrator is present in the system. Whenever an entered command is

ignored a prompt TRY AGAIN is displayed to inform the operator that he has given an incorrect command or that the probe calibrator is absent from the system.

After start-up, whenever a valid command is given to the probe calibrator and if no wave period and amplitude has yet been entered (this may have been previously entered in the manual mode), the unit will respond with a prompt instructing the operator to do so (see latter part of Figure 4.3(b)). After the requested values have been successfully entered by the operator a BUSY prompt will be displayed while the cycle table is being computed. A DONE prompt will signify the end of this operation after which the unit may be fully utilised. New wave height and amplitude values may be entered using the 'R' option where all previously entered values are repeated on the VDU with the prompt 'DO YOU WISH TO CHANGE THESE VALUES' giving the operator the opportunity to change or correct them (see Figure 4.3(b)).

Whenever a calibration (cycle probes) command is given, the model control console is left free to perform any other task, that is, it does not stay tied to the probe calibrator. However, when the UP or DOWN command is given the two units stay linked together via STANOP routine KBDCHK since to stop the up or down movements any key may be depressed. The VDU prompts displayed when the probes are moved is shown in Figure 4.3(c) and (d). Other tasks performed by the model control console while the probes are cycling is to run the A to D gathering information from the probes.

The calibrator jig is not always visible to the operator so that great care must be taken not to submerge the probes into the water. However, error messages are displayed on the VDU when the jig is slipping on the pulley or when the jig has reached its travel limits (see Figure 4.3(e) and (f)).

The program listing of the additional and modified software routines enabling the probe calibrator to be interfaced to the MHDAS is given in Appendix D. A modified unit has been

installed in the instrumentation rack at the 1 m tilting flume as illustrated in Figure 6.1. The interfacing of the probe calibrator to the MHDAS enabled the author to get acquainted with the MHDAS operating system. The knowledge obtained by this project will act as a basis to interface the measurement platform controller (MPC) (Section 5) and the trolley controller (Section 6) to the MHDAS.

RETURN ENDS A COMMAND. RUN=REMOTE UNIT NO., 0 BY DEFAULT
 DATA FORMAT OPTIONS A=ASCII, B=BINARY, O=AUTO DATA OUTPUT.
 DATA DESTINATION OPTIONS V=VDU, P=PRINT, D=DAC, C=CDL, M=MODEM
 DISPLAY FILE INDEX X:(RUN)
 OUTPUT A DATA FILE O,(FILE NO.);OPTIONS:(RUN)
 DELETE A DATA FILE D,(FILE NO.): (RUN)
 READ A CDL FILE T;(R=REWIND, S=SEARCH)
 SETUP ADC S:(RUN)
 RUN ADC A:(RUN)
 PROBE CALIBRATOR K;OPTIONS:(RUN)

CALIBRATOR OPTIONS ; U=PROBES MOVED UP, D=PROBES MOVED DOWN,
 C=CYCLE PROBES, R=DISPLAY OR ENTER NEW PARAMETERS

FIG 4.3 a COMMAND LISTING

Operator Responses Highlighted

K;R:11
 REMOTE UNIT NO. 1

HEIGHT IN mm = 80
 PERIOD IN msec = 1200
 DO YOU WISH TO CHANGE THESE VALUES?.....Y OR N
 Y
 ENTER THE CALIBRATION HEIGHT
 BETWEEN LIMITS 10mm TO 100mm
 100
 ENTER THE CALIBRATION PERIOD
 BETWEEN LIMITS 700 TO 2000 msec
 1000
 BUSY

REMOTE UNIT NO. 1

DONE
 STANOP VER.821115 AT YOUR SERVICE.
 ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
 MODEL CONTROL CONSOLE NO. 1

FIG 4.3 b RE-ENTER PARAMETERS

K;C:11
REMOTE UNIT NO. 1

HEIGHT IN mm = 100
PERIOD IN msec = 1000

TO STOP THE PROBES ENTER : ANY KEY:(RUN)

K:11
STANOP VER.821115 AT YOUR SERVICE.
ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
MODEL CONTROL CONSOLE NO. 1

FIG 4.3 c CYCLE PROBES

K;U:11
REMOTE UNIT NO. 1
STRIKE ANY KEY TO STOP PROBES

K
STANOP VER.821115 AT YOUR SERVICE.
ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
MODEL CONTROL CONSOLE NO. 1

FIG 4.3 d MOVE PROBES UP

K;D:11
REMOTE UNIT NO. 1
STRIKE ANY KEY TO STOP PROBES
JIG EXCEEDS ITS BOTTOM LIMIT
PUSH K;U:(RUN)

STANOP VER.821115 AT YOUR SERVICE.
ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
MODEL CONTROL CONSOLE NO. 1

FIG 4.3 e JIG LIMIT ERROR

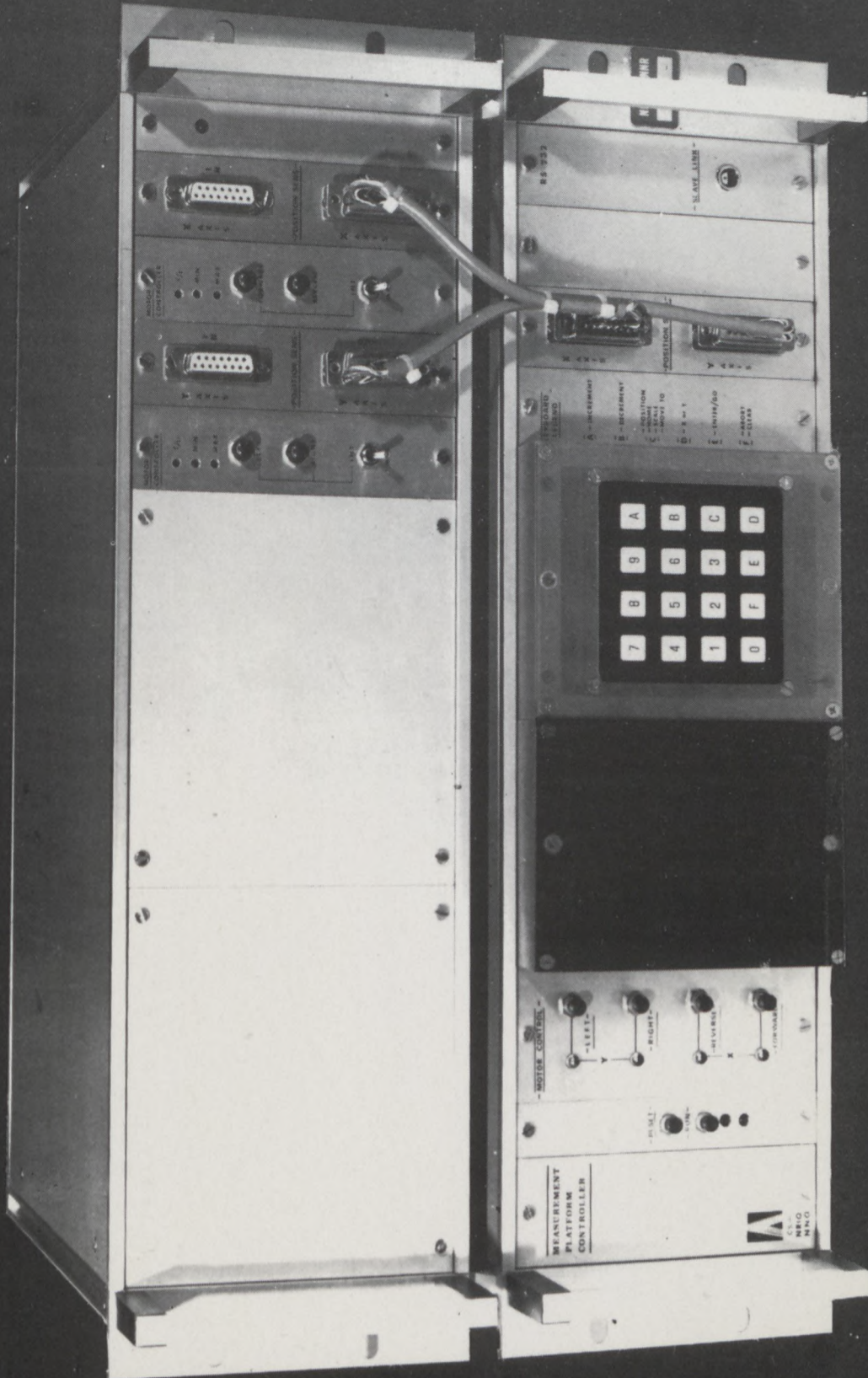
K;C:11
REMOTE UNIT NO. 1

HEIGHT IN mm = 100
PERIOD IN msec = 1000

TO STOP THE PROBES ENTER : ANY KEY:(RUN)

REMOTE UNIT NO. 1
CALIBRATOR MOTOR STEPS LOST

FIG 4.3 f STEPS LOST ERROR



Measurement Platform Controller

5. MEASUREMENT PLATFORM CONTROLLER

5.1 Introduction

The measurement platform controller (MPC) was developed at NRIO to control the DC motors which move the measurement platform from one position to another. These mobile measurement platforms are used to support the instrumentation that is used in gathering data from tests done in physical model studies. The platform is able to move in both the X and Y axes at about 0,5 m above the water level. The MPC can move a platform smoothly and accurately to any position (± 2 cm) inside a model basin within the movement limits of the measurement platform.

5.2 General Description

The MPC consists of an MPC control unit and a motor supply and control unit.

(a) MPC control unit

The MPC control unit is based on the NRIO microcomputer system which was developed at NRIO (Holroyd, 1982). This unit allows the operator to move the bridge in either or both the X and Y axes by means of front panel control buttons. There are four of these motor control pushbuttons (override control) that can move the platform either left, right, forward or reverse whenever the final position of the platform need not be very accurate. A front panel display enables the operator to follow the exact position of the platform along each axis (five digits per axis). A block diagram of the MPC is given in Appendix A, drawing No HB81001.

The MPC has four different operating modes and these may be selected via the front panel mounted keyboard while leds at the display indicate these operating modes.

(i) Position mode

With the MPC in this mode the display will monitor and follow the position of the measurement platform while any of the override motor control pushbuttons are depressed.

(ii) Move to mode

When the MPC is in this mode the operator may enter on the keyboard the position in either of or both axes to which he wishes to travel. While this position is being entered it will be echoed on the display. The platform will move to its destination after the keyboard E key has been depressed (ENTER/GO).

(iii) Home mode

With the MPC in this mode the measurement platform will move in both axes to a preset or home destination. This destination is set on the system ID card but may be changed via the keyboard. The keyboard E key will initiate the move to this destination.

(iv) Scale mode

Normally data are collected on a model at predetermined points on a grid. This grid will have an X and a Y scale and these scale factors may be entered into the MPC via the keyboard. When the MPC is in the MOVE-TO mode the X and Y values displayed can be incremented (key A) or decremented (key B) by the value entered in the SCALE mode. The platform can move to the new position by depressing the E key (ENTER/GO).

The F key (ABORT/CLEAR) may be used to erase any incorrect value or values and to replace them with zero's.

(b) Motor supply and control unit

The motor supply unit contains the DC motor speed controller, motor supplies and the motor direction relays.

The shunt DC motor speed is varied by changing the armature voltage. Voltage feedback from the armature is compared with a reference voltage (desired speed) which controls the power pack supplying power to the armature. The reference voltage is obtained from a frequency-to-voltage converter from which the frequency originates at the MPC control unit. An on-board oscillator permits the motor supply unit to operate without the presence of the MPC control unit. To protect the motor against damage due to overcurrent, the armature current is monitored and the speed (voltage) is decreased whenever the set limit is exceeded. Provision is made to set different current limits for the different types of motors used. This controller is referred to as the MK1 speed controller (MK2 is used on the trolley, refer to Section 6.4). A drawing and circuit description of the MK1 speed controller is given in Appendix A.

5.3 Requirements to Interface to the MHDAS

From the design stage it was decided to incorporate this unit into the model hall data-acquisition system in order to permit remote control of the platform. This is necessary in order to automate model testing. Slots were made available in the MPC control unit for the full set of STANOP cards.

Normally after a command has been transferred to the MPC it will respond by asking for data to be entered e.g. entering the MOVE TO position. For this the MPC uses the STANOP CHARIN routine (CHARIN gets a character from a keyboard via the system RS232 interface and exits with the character in the A acc.). In this unit provision is made whereby data may also be transferred via the 11-byte prompt buffer (p data-transfer-request). This allows commands and data to originate from a test-control program stored in a unit anywhere along the data highway (slave, model control console or minicomputer).

The STANOP cards used by this unit are:

- (i) RS232/TTY interface to communicate with the MHDAS.

- (ii) RAM/EPROM card to house the STANOP operating system.
- (iii) ID card byte 11 and 12 are set to \$1000 to indicate the start address of the MPC program.
- (iv) STANOP-PTM card used for delays generated in STANOP.
- (v) STANOP PIC card which holds the highest EPROM of STANOP. The restart vector of STANOP is switch selectable on this card (\$C000).

The standard set of six instructions required by STANOP to control an external program was added to the beginning of the program. They are:

```

JMP COMINT (routine to interpret the commands generated at the
            model control console and destined for the MPC)
JMP SETUP  (routine to setup the devices used by the MPC)
JMP PTEST  (select between the address of MPC or trolley
            introduction prompt and return to ABC)
ABC JSR VDUOUT (output MPC/TROLLEY introduction prompt on the
               VDU)
RTS          (return to STANOP)
JMP MPC      (MPC control loop used in stand alone mode).
```

A deviation to the standard set of instructions is the JMP PTEST (jump to) statement. The program jumps to a routine to select whether an MPC or a trolley controller introduction prompt must be displayed at the model control console VDU and exists from this routine by jumping back to ABC to display the required prompt.

After the unit is running under STANOP control it must preferably be able to switch between internal (stand alone mode) or external (automated mode either from model control console or from a model test-control program) control without prior setting of any external switches, etc.

5.4 Theory of Operation

After all the requirements for interfacing the MPC to the MHDAS have been met, the theory of operation is as follows:

(a) Stand-alone mode

In this mode the MPC commands are entered via the front panel keyboard and push-buttons. At start-up STANOP will run the MPC setup routine to initialise all the devices and flags used in the MPC. The main stand alone control routine is called MPC. This routine sets the internal/external control flag, FLA (\$FF internal; \$00 external), reads and displays the position from the position-fixing module and checks if any of the motor control pushbuttons have been depressed. The int/ext flag is always cleared at the end of this routine.

To display the X and Y positions information the MPC makes use of the NRIO microcomputer system developed 12-digit led display module, drawing MA79020. The display is divided into two rows of six digits each, the first five digits of each row being used to display the position information, the top row for the X-axis and the bottom for the Y-axis. The bottom row's sixth digit is omitted while the top row's sixth digit is used to indicate when an error occurs. The first three digits in each row are used to display the coarse position in metres while the next two digits (4 and 5) are used to display the fine position in centimetres.

The X and Y positions which are in binary are read and converted to their equivalent five-digit decimal value by the program controlling the position-fixing module (Section 5.5). It was decided to do this conversion at the position-fixing module rather than at the MPC control program because whenever the platform is in motion the MPC control program must perform numerous tasks such as changing the motor speed, checking for movement limits, and reading and displaying the X and Y axis positions. Transferring the converted positions to the MPC control

program saves time and thus improves the efficiency of the program controlling the movement.

The speed is varied by changing the input voltage to the speed controller between 0V to 5V (0V = min speed; 5V max speed). To obtain this varying voltage a frequency is generated by a PTM which is converted to a voltage with the aid of a 9400 frequency to voltage converter (PTM timer #1 for the x-axis and PTM timer #3 for the y-axis). A table of frequencies corresponding to 16 different speeds is stored in memory and is stepped through at a constant rate to give an even increase or decrease in the speed (PTM timer #2 determines the rate). Four set speeds are normally used; slow (1 step), medium slow (6 steps), medium (11 steps) and fast (16 steps).

At start up the display will be filled with A's to indicate that the MPC does not know its present position. The operator may determine the platform position by depressing any of the motor control push-buttons. The platform will move at slow speed until the first coarse position sensor plate has been reached (see Section 5.5) where the motor will automatically stop. The position obtained from the position-fixing module (see Section 5.5) is displayed and the operator is given the opportunity to enter the position (ENTER/GO = E key) if it is correct or to repeat the position finding sequence if incorrect (ABORT/CLEAR = F key).

When any of the motor control push-buttons are depressed the program will jump to sub-routine OMO2 (override movement control), a brief description of which is given below (see also flow diagram, Figure E5):

- (i) Remember the requested direction(s) of movement in temporary memory T5 (to later check for any change in movement).
- (ii) Determine the axis and direction of travel then load flags MODE with the X direction and MODE1 with the Y

direction. (\$11 = forward, \$22 = reverse, \$44 = left, \$88 = right, \$00 = no movement.)

- (iii) Does the platform know its present position in the requested axis of movement?
- (iv) If not, preset the incremented steps counter to do only one step of the speed table (move the platform at slow speed in the requested direction until the first position plate is reached).
- (v) If yes, preset the incremented steps counter to 16 (fast speed).
- (vi) Set the speed table increment/decrement flag (\$FF = increment speed; \$00 = decrement speed and stop movement).
- (vii) Do subroutine DIRSET (flow diagram, Figure E10) to activate the motor direction led(s) and relay(s); also set the direction flag (DIR) to inform the position-fixing module of the direction of movement. (\$00 = no movement, \$11 = forward, \$22 = reverse, \$44 = left, \$88 = right, \$14 = forward/left, \$18 = forward/right, \$24 = reverse/left, \$28 = reverse/right).
- (viii) Do subroutine SPM. Transfer the direction flag (DIR) to the position-fixing module (Section 5.5) to inform it of the direction(s) of movement.
- (ix) Load and start the speed control oscillators (PTM: timer #1 = X axis; timer #3 = Y axis). Timer #2 determines the increment/decrement rate.
- (x) Do subroutine GETPOS (flow diagram, Figure E2) - read the position from the position-fixing module. GETPOS also performs subroutine TEST to check when the first coarse position has been obtained. Subroutine ER4

stops the motor, displays the position and an 'E' on the error digit giving the operator the opportunity to either reject or accept the position.

- (xi) Do subroutine DISPLA (flow diagram, Figure E4) - display the position.
- (xii) Do subroutine ESXY (flow diagram, Figure E8) - test if the platform has overridden the limits of movement as set on the ID board (bytes 0, 1 = X lower limit, bytes 2, 3 = X higher limit, bytes 4, 5 = Y lower limit, bytes 6, 7 = Y higher limit).
- (xiii) Test timer #2 for time out (T0), if yes, continue from (xvii).
- (xiv) Test for any movement change (test MODE, MODE1 and T5).
- (xv) If there is no movement (MODE and MODE1 = zero) preset the timers, reset the direction leds and relays and return to STANOP.
- (xvi) If there has been a change (T5 $\neq$ present state of the motor control push-buttons) jump back to (i).
- (xvii) Increment:
 - (a) Has the preset speed been reached?
 - (b) If yes, then return to (x).
 - (c) If not yet, store the next higher value (freq.) from the speed table in the timer and return to (x).
- (xviii) Decrement:
 - (a) Has the preset speed been reached?
 - (b) If yes, and at zero speed, clear MODE and/or MODE1, JSR DIRSET to set leds and relays, etc. and JSR SPM to inform position fixing module. Return to (x). If yes, and not at zero speed, return to (x).

- (c) If no, load the next lower speed frequency into the timers. Then return to (x).

To prevent damage caused by reversing of the motor at high speeds its direction can be changed only after the motor has stopped. Subroutine DIRSET activates the direction relays and thus the motor rotation according to the contents of MODE and MODE1. These flags are set at the start of the move routine and are only cleared when the motor has reached zero speed. By monitoring these two flags and prohibiting them to change before they are cleared prevents the motor from being damaged.

X and Y axis limits may be set on the ID card (bytes 0 to 7) to confine the area of movement and thus prevent the platform from ramming the wave generators or other obstacles on the boundaries of a model basin. For a distance of one metre from and to these set limits the platform will move at a slow speed until it reaches the limit where the platform will automatically stop. A digit on the display will indicate when the platform is in this area (C on the error digit).

A HEX keyboard is used (Drawing MA79019) to select and then enter values for the different operating modes. Leds indicate the selected mode of operation.

1 - 9 = numerical input

A = INCREMENT - [the selected X or Y value displayed in the
B = DECREMENT - [MOVE-TO mode by the value entered in the
scale mode

C = MODE SELECT (display position, home, scale and move to)

D = X or Y axis select in the home, scale and move to modes

E = ENTER/GO to position

F = ABORT/CLEAR.

Values are entered or changed using the same method in all the operating modes (except the position mode where the display monitors the position only). First, the mode is selected by

depressing the C key; second, the desired axis is selected by depressing the D key and only then may the desired information (numericals) be entered. Any incorrect entry is deleted (replaced by 0) with the aid of the F key. The entered values are echoed on the display form left to right and deleted in the opposite direction. The enter routine in any mode is terminated by the E (ENTER/GO) or C (MODE SELECT) key.

To initiate the controller to move to the destination as indicated on the display in the HOME or MOVE-TO mode the E key (ENTER/GO) must be depressed and is referred to as the non-override movement control. The following sequence of events occurs in these two modes after the E key has been depressed (refer to the flow diagrams EG and MOVE Figures E9 and E9(f)).

- (i) The destination and the present positions are converted to conform to the mathpack format (Liadsky, 1978). Calculate the minuend and subrahend and subtract to find the distance of travel (subroutines CHEX, MCON, MINX). MODE and MODE1 are set in subroutine MINX as indicated by step (ii) of the override movement control.
- (ii) Do subroutine ESX (flow diagram, Figure E9(d)) to check that the destination does not exceed the limits and then displays an error if they do.
- (iii) Do subroutine XSPEED (flow diagram, Figure E9(e)) to select the speed (> 2 m from destination = fast speed; < 2 m from destination = med spd; < 1 m from dest = med/slow spd; < 0,5 m from des = slow spd).
- (iv) Repeat (i) to (iii) for the Y axis in the HOME mode or if selected in the MOVE-TO mode ((i) to (iv) forms part of the interrupt routine EG, while the following steps form part of subroutine MOVE).
- (v) Do subroutine DIRSET to activate the direction relays and leds, also set the direction flag (DIR) used to inform the position-fixing module.

- (vi) Load and start the timers (speed oscillators).
- (vii) Inform the position-fixing module of the direction of movement, subroutine SPM (transfer DIR).
- (viii) Determine if there is to be movement in the X or/and Y axes. The subsequent sequence of events will be the same for both axes (MOVE routine).
- (ix) Test the keyboard for an abort (in both these modes of operation any key will initiate the abort routine to stop the platform movement (see flow diagram Figure E19(e)).
- (x) Do subroutines GETPOS and DISPLAY, read and display the position.
- (xi) Test the increment/decrement rate timer for timer time out (TO), that is, end of rate delay cycle.
- (xii) If yes, increment or decrement speed (same as for the override mode, steps xvii and xviii).
- (xiii) If no, convert the latest position to mathpack format (JSR MIN1).
- (xiv) Subtract the destination from present position using mathpack.
- (xv) If the exponent has changed (distance from destination) select new speed (XSPEED).
- (xvi) Check if the platform has reached its destination (MODE or MODE1 = zero, set in (xii)), if not go to (ix).
- (xvii) Do HALT sequence:
 - (a) Stop and preset timers.
 - (b) Inform the position-fixing module (subroutine SPM).

- (c) Test and display the number of incorrect positions fixed, if any (see Section 5.5 on the position-fixing module).

The increment and decrement keys (A and B) can be used only in the MOVE-TO mode when either the X or the Y axis has been selected. The X or Y MOVE-TO position displayed will then either be incremented or decremented by the value previously entered in the SCALE mode. The scale may be any value between 0,01 and 9,99 and is set at switch-on to be 1,00 by default.

A digit to the right of the x axis display is used to indicate when an error has occurred. The keyboard (ABORT/CLEAR) key is used to reset the error and/or the error/digit.

The errors are:

- (1) When the 'move-to' position is outside the set limits of movement (non-override mode) a 'C' will be displayed.
- (2) A 'C' will be displayed when the platform is closer than 1 m from the set limits in the override mode (platform will slow to slow speed and stop at the limit).
- (3) When changing or entering values when both axes are selected. Values can be entered into one axis at a time only, while movement can be in both axes when selected
- (4) It will display the recorded X and/or Y incorrect readings from the position sensors after the movement has stopped.
- (5) A 'E' will be displayed when the first position is read. The operator must enter an E if the position is good or an F if not.

(b) MHDAS controlled mode

In this mode of operation, commands are either generated at the model control console keyboard or obtained from a model-test control program running in a unit as an external program anywhere along the data link. In both cases the command is transferred to the MPC via the 11-byte command buffer, byte 4 being reserved to indicate the origin of the command (0 - from model control console; 1 - from external program). When the MPC is controlled from the model-control console keyboard the MPC will send a prompt to the VDU, instructing the operator to enter scale, home and other data on the keyboard. The MPC uses subroutine CHARIN, which is present in STANOP, to collect the input data from the keyboard. The keyboard transfer request (R data transfer) is passed up via the RS232 interface to the model-control console to get the keyboard entry and return it to the requesting unit. On exit from CHARIN the character will be in the A acc. When the MPC is controlled via a model-test control program running in another unit, scale data, etc. are transferred to the MPC using the 11-byte prompt buffer. If byte 4 in the command buffer is set, the MPC will respond with a P transfer request via the system modem (request to transfer the prompt buffer). The unit running the model-test control program (also under STANOP control) will respond to this request by transmitting the prompt buffer (also refer to Section 8).

Valid MPC commands are transferred via bytes 2 and 3 of the command buffer and are:

L - left	G - move to (enter value)
R - right	GH - move to HOME
B - reverse	GT - move to TARGET
F - forward	A - increment
P - set parameters	B - decrement.

The only difference between the stand alone and MHDAS controlled mode is the addition of a pre-setable, using the set parameters P command, MOVE TO position called the TARGET. The TARGET can

be used by the operator to enter and remember a position inside a model basin that is often visited, e.g. the first test point on the measurement grid, or as a second HOME position as described in Section 5.2. A typical command listing displayed on the VDU when the MPC is present in the MHDAS is given in Figure 5.2(a).

Byte 0 of the command buffer is used by the MPC as a recognition flag and must always contain an M or the command will be ignored. A typical command entered on the model control keyboard has the following standard format:

M; GH : (RUN)

where M = recognition flag

GH = instruction

RUN = remote or nested remote unit number(s).

If a command is incorrectly entered or not understood by the MPC an alarm will sound and a TRY AGAIN prompt will be displayed on the VDU.

The move routines used in the MHDAS controlled mode are the same as described in the stand alone mode. Commands F - forward, B - reverse, L - left and R - right will access the override move routine while the commands G - move to position, GH - move home and GT - move to target position will access the non-override routine. To stop movement when the override commands are being used the keyboard space bar must be depressed. When the non-override commands are being used the measurement platform may be stopped before the platform has reached its destination by setting the receive data register on the RS232 interface card at the MPC. This can be done by transferring any character to the MPC (ANY KEY : (RUN)). (See Figures 5.2(c)-(e) and (g) for the VDU prompts displayed when the platform is requested to move.) Whenever a non-override move command is given, the MPC can perform it independently and the model-control console is left free to do any other task.

RETURN ENDS A COMMAND. RUN=REMOTE UNIT NO., 0 BY DEFAULT
 DATA FORMAT OPTIONS A=ASCII, B=BINARY, O=AUTO DATA OUTPUT.
 DATA DESTINATION OPTIONS V=VDU, P=PRINT, D=DAC, C=CDL, M=MODEM
 DISPLAY FILE INDEX X:(RUN)
 OUTPUT A DATA FILE O,(FILE NO.):OPTIONS:(RUN)
 DELETE A DATA FILE D,(FILE NO.): (RUN)
 READ A CDL FILE T;(R=REWIND,S=SEARCH)
 SETUP ADC S:(RUN)
 RUN ADC A:(RUN)
 MOVE MEAST PLATFORM M;OPTIONS:(RUN)

MEASUREMENT PLATFORM OPTIONS ; F=FORWARD, B=REVERSE, L=LEFT,
 R=RIGHT, G=MOVE TO POSITION, IX=IY=INCREMENT GRID POSITION,
 DX=DY=DECREMENT GRID POSITION, P=DISPLAY\CHANGE PARAMETERS,
 GH=MOVE 'HOME', GT=MOVE TO TARGET POSITION

FIG 5.2 a COMMAND LISTING

Operator Responses Highlighted

PLATFORM X POSITION IS = 8, 0
 PLATFORM X AXIS LIMITS ARE 2,50
 AND 14,50
 X AXIS 'HOME' POSITION IS = 2, 0
 MY X TARGET POSITION IS = 10, 0
 X AXIS GRID SPACING IS = 1, 0
 PLATFORM Y POSITION IS = 39,98
 PLATFORM Y AXIS LIMITS ARE 8, 0
 AND 55,20
 Y AXIS 'HOME' POSITION IS = 55, 0
 MY Y TARGET POSITION IS = 20, 0
 Y AXIS GRID SPACING IS = 1, 0
 DO YOU WISH TO CHANGE THIS?...Y OR N

Y
 X AXIS 'HOME' POSITION IS = 2, 0
 DO YOU WISH TO CHANGE THIS?...Y OR N
 N
 MY X TARGET POSITION IS = 10, 0
 DO YOU WISH TO CHANGE THIS?...Y OR N
 Y
 MY X TARGET POSITION IS = 14
 X AXIS GRID SPACING IS = 1, 0
 DO YOU WISH TO CHANGE THIS?...Y OR N
 N
 Y AXIS 'HOME' POSITION IS = 55, 0
 DO YOU WISH TO CHANGE THIS?...Y OR N

STANOP VER.821115 AT YOUR SERVICE.
 ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
 MODEL CONTROL CONSOLE NO. 1

FIG 5.2 b PARAMETER LISTING


```

M;GH:1
REMOTE UNIT NO. 1
PLATFORM X POSITION IS =      8, 0
X AXIS 'HOME' POSITION IS =    2, 0
PLATFORM Y POSITION IS =    39,99
Y AXIS 'HOME' POSITION IS =    55, 0

```

TO ABORT ENTER ANY KEY:(RUN)

N:1

REMOTE UNIT NO. 1

STANOP VER.821115 AT YOUR SERVICE.
 ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
 MODEL CONTROL CONSOLE NO. 1

FIG 5.2 c MOVE HOME

```

M;GT:1
REMOTE UNIT NO. 1
MY X AXIS DESTINATION IS =    10, 0
MY Y AXIS DESTINATION IS =    20, 0

```

TO ABORT ENTER ANY KEY:(RUN)

W:1

REMOTE UNIT NO. 1

STANOP VER.821115 AT YOUR SERVICE.
 ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
 MODEL CONTROL CONSOLE NO. 1

FIG 5.2 d MOVE TO TARGET

```

M;G:1
REMOTE UNIT NO. 1
PLATFORM X POSITION IS =      8, 0
ENTER..Q FOR NO X AXIS MOVEMENT
OR.....X PLUS NEW DESTINATION
X12
PLATFORM Y POSITION IS =    39,99
ENTER..Q FOR NO Y AXIS MOVEMENT
OR.....Y PLUS NEW DESTINATION
Y15

```

TO ABORT ENTER ANY KEY:(RUN)

FIG 5.2 e MOVE TO POSITION

```

M;IX:1
REMOTE UNIT NO. 1
PLATFORM X POSITION IS =      8, 0
MY X AXIS DESTINATION IS =    13, 0
ENTER :  G...MOVE TO DESTINATION
          D...DECREMENT DESTINATION
          I...INCREMENT DESTINATION
          Q...QUIT

```

FIG 5.2 f INCREMENT

M;F:1

REMOTE UNIT NO. 1
STRIKE THE SPACE BAR TO STOP

STANOP VER.821115 AT YOUR SERVICE.
ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
MODEL CONTROL CONSOLE NO. 1

FIG 5.2 g OVERRIDE FORWARD

M;F:1

REMOTE UNIT NO. 1

ENTER F(FORWARD) OR B(REVERSE) FOR 1st X AXIS POSITION FIX

F

BUSY

PLATFORM X POSITION IS = 8, 0

IS THIS POSITION CORRECT?.....Y OR N

Y

ENTER L(LEFT) OR R(RIGHT) FOR 1st Y AXIS POSITION FIX

L

BUSY

PLATFORM Y POSITION IS = 40, 0

IS THIS POSITION CORRECT?.....Y OR N

FIG 5.2 h START-UP

After start-up, if the MPC does not yet know its present position (this may have been previously found in the manual mode) and if a valid MPC command is entered at the model-control console, the MPC will respond by determining its present position (see Figure 5.2(h) for the VDU response). The platform will move in the requested direction until the first position plate (drawing HB81007) has been reached where the movement will stop and the operator asked if the position obtained is correct. Only after the position has been determined in both the X and Y axes will the MPC correctly respond to any valid command.

A listing of the MPC parameters (both axes) will be displayed on the VDU by giving the P command. A prompt DO YOU WISH TO CHANGE THIS is displayed giving the operator the opportunity to make any changes to the given listing, this sequence is displayed in Figure 5.1(b).

To terminate this command an N (no) character is entered on the keyboard.

When testing in a grid pattern the position of the platform can be incremented or decremented by the scale value using one of the following commands:

- IX - increment the x move-to position
- IY - increment the y move-to position
- DX - decrement the x move-to position
- DY - decrement the y move-to position

The VDU will display the format as shown in Figure 5.1(f). The operator is given the options to move to the new destination (G), to increment or decrement again or to quit the operation (Q).

The stand alone and MHDAS modes can easily be used simultaneously enabling the operator to change from one mode to the other without any prior setting.

5.5 Position-Fixing Module

The MPC obtains its position with the aid of a function card, the binary-coded position plates, a sensor to read the position plates and a slotted disc read by a slotted opto Schmitt switch. The program listing and flow diagrams of this module are given in Appendix F.

The position is computed by a function card which is a one-board microcomputer (Drawing MA79015). This card is one of a family of cards designed by the NRIO Electronics and Instrumentation Group (EIG) for the NRIO Microcomputer System (Holroyd, 1982). Communication between the main microcomputer bus and the function card is done via a communal block of RAM which can be accessed by either MPU. The function card's MPU has priority over this RAM and sets a busy flag to prevent any RAM accessing contentions. The common RAM area for this position-fixing unit is set for address \$A200 to \$A21F.

Coarse position fixing is done using binary coded plates placed 1 m apart along both axes of movement. These plates are read by an eight-bit light fork (bits 0 to 7) attached to the MPC. The light fork is a U-shaped device with eight transmitter infra-red leds on the one prong and eight receiver light-activated-switches (LAS) on the other (see drawing No HB81007). The output from these LASSs will always be at a logic high and go low when the light beam is interrupted. The X axis coarse position information is monitored on bits 0 to 5 of PIA port A and the Y axis information on bits 0 to 7 of PIA part B. The MSB for the Y axis and bit 5 for the X axis are 'read' trigger pulses. The pulses originate from a pin situated on the end of a position plate, see drawing No HB81007. Both axes 'read' trigger pulses are fed to the function card interrupt input CB1 to initiate an MPU IRQ while bits 6 and 7 of PIA port A are used to indicate to the function card control program on which axis a coded plate has been read (a monostable triggered by the 'read' trigger pulse momentary changes the voltage level on bit 6 for the y axis and on bit 7 for the x axis).

Fine position fixing is accomplished by counting pulses from a slotted disc and a slotted opto Schmitt switch mounted on each motor shaft. The pulses are fed to hardware counters which can be preset to give one output pulse per centimetre travel (see drawing No HB81002). Up-down counters are used to cater for both the directions of travel. These centimetre pulses are fed to the function card PIA, CA1 and CA2 interrupt inputs for the X and Y axes, respectively.

Position updating is initiated via the MPU IRQ input originating from the PIA interrupt inputs CA1, CA2 and CB1. The IRQ routine establishes the origin of the interrupt, jumps off to update the respective position (coarse or fine) and then rechecks the IRQ flags to ensure that no interrupts are missed. A control loop in the position-fixing program updates the communal RAM and continuously monitors the direction flag (see flow diagram Figure F1). The IRQ mask is normally set and cleared only when the main MPC program requests a move. At the end of the IRQ routine there is a deviation from the normal return from interrupt RTI. Here the stack is re-initialised and the program jumps to the start of the main control loop. This is done to immediately transfer the latest position to the communal memory. The primary task of the coarse position-fixing routines is to read the position which is in binary from the position sensors and convert these to their equivalent three-digit decimal values as required by the MPC. The primary task of the fine position routines is to either increment or decrement, depending on the direction, the contents of the memory registers which act as fine-position counters (software counters) and also to convert these to their equivalent decimal values (see flow diagrams Figures F3 and F4).

At start-up the contents of all the position memory registers in the position-fixing module are filled with A's and transferred to the communal memory. This shows that the position module program is running and also acts as a flag to itself and the main MPC program that there has not yet been a coarse position fix.

Initially this unit gave numerous spurious coarse and fine position fixes. This was due mainly to noise generated by the motor speed controller and the noisy electrical environment of the model hall. Extensive decoupling of the supply and transmission lines at the sensor and at the MPC X and Y position finder card, drawing No HB81002, largely solved the problem. Further corrective steps were incorporated to ignore any coarse input if all the sensor bits were high which occurs when the sensor is between plates, and to use the PTM of the function card to disable the coarse setting routines for about 2,8 sec after the course position has been updated. The latter measure also eliminated any read-trigger pulse bounce.

If an incorrect reading is received from the position sensors while the platform is in motion and is being controlled from a model control console or model-test control program, the platform might stop in the middle of the model because the incorrect reading can fall outside the platform limits as set on the ID card, which would cause the platform to stop. To prevent such an occurrence the next coarse position is continuously computed and stored into the position memory registers at the next read pulse. This requires that the position at the start of movement must be correct. (As part of the MPC initialisation the operator may enter or reject the first X and Y coarse position fix found by the position-fixing module). The number of incorrect readings is recorded and displayed as soon as the platform has reached its destination. When errors are recorded in both axes of movement they will alternately be displayed on the error digit. For convenience the maximum count is limited to nine. If a coarse trigger pulse has been missed, the relevant position will still be updated because a fine position overflow will adjust the coarse position. If the coarse position read trigger does not occur within 20 cm after the fine position has overflowed an incorrect reading is recorded and the next position computed. As it is mechanically difficult to set and keep the sensor plates exactly 1 m apart and also because an update from a fine position overflow may occur, the danger exists that an incorrect value may be stored in the position

memory if the movement starts close to a coarse position sensor plate. To explain the course of events when the movement starts in this area the following description is given (also refer to Figure 5.5). Although only the reverse direction is described the arguments are equally valid for the forward direction of movement and the same corrective steps are taken.

Situation 1 (Figure 5.5a)

Normally if the movement starts at 5,00 or at any value above 5,00 the next computed coarse position will be 5,00 (see flow diagram PUTX, Figure F2). This computed value is correct if the movement starts from position A. If the movement starts from position B the fine position counter will overflow (4,99 will be displayed) and the 20 cm distance for a coarse position read trigger will be allowed. As the read trigger does not occur an incorrect reading is recorded. To prevent an incorrect reading from being recorded a free-play area is allowed when the movement starts within 5,05 to 5,00 (5 cm) of the recorded position. If a coarse read trigger pulse is not obtained before 4,95 the next coarse position will be computed without an incorrect reading being recorded.

Situation 2 (Figure 5.5b)

Normally if the movement starts at a value above 4,00 the next computed coarse position will be 4,00. This computed value is correct if the movement starts from position D. If the movement starts from position C the wrong value (4,00 instead of 5,00) will be displayed and an incorrect reading will be recorded. To prevent this, a free-play area is allowed (4,99 to 4,95). When the movement starts in this area the present coarse position is incremented and referred to as the computed value. If no read trigger pulse is obtained before 4,95 the computed value is revised and set to be 4,00.

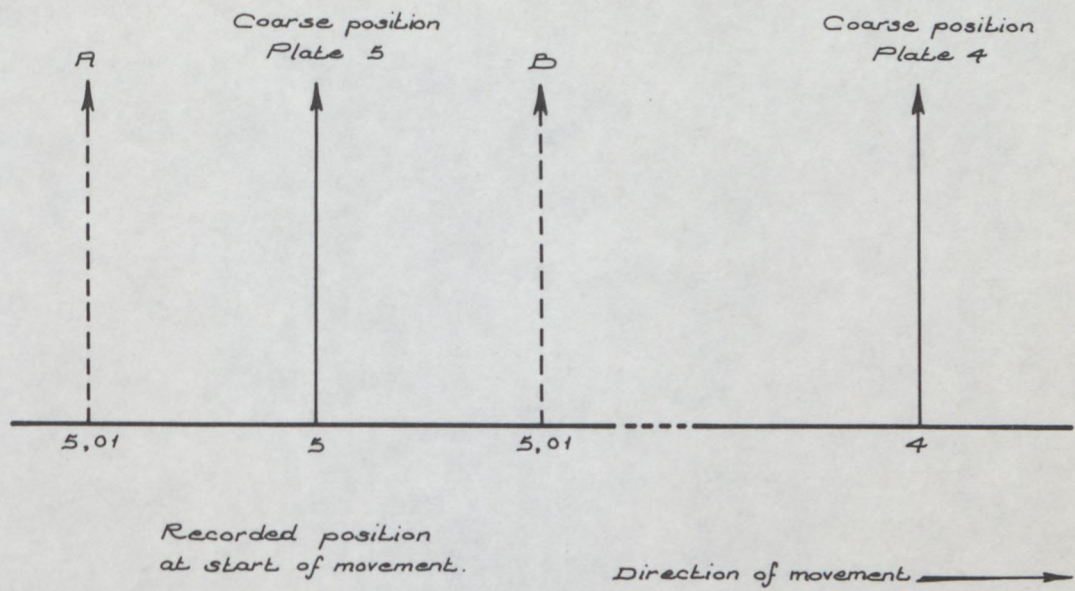


Fig. 5-5 a

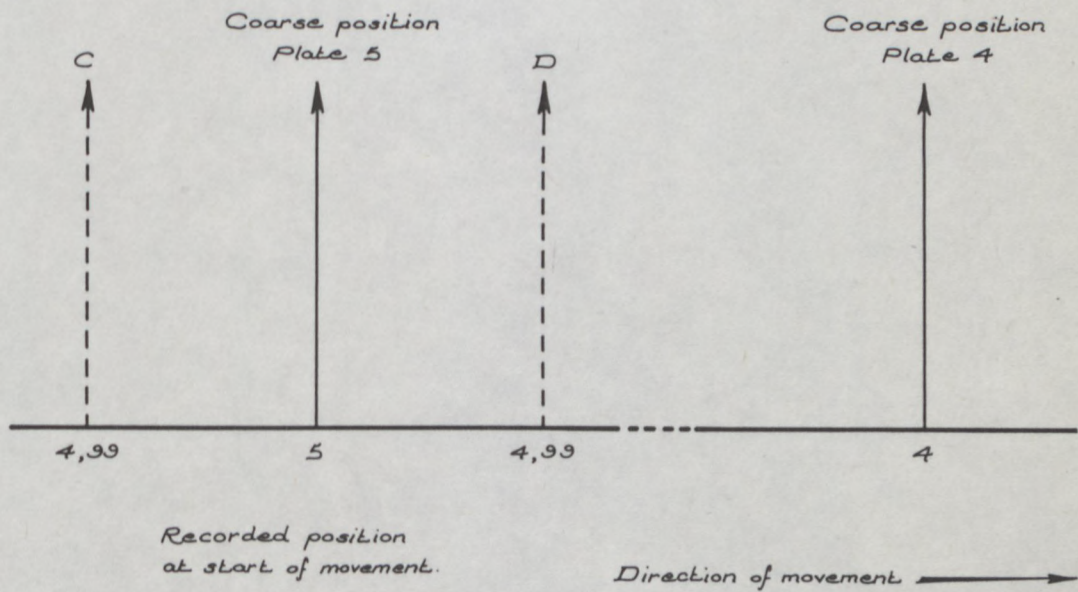
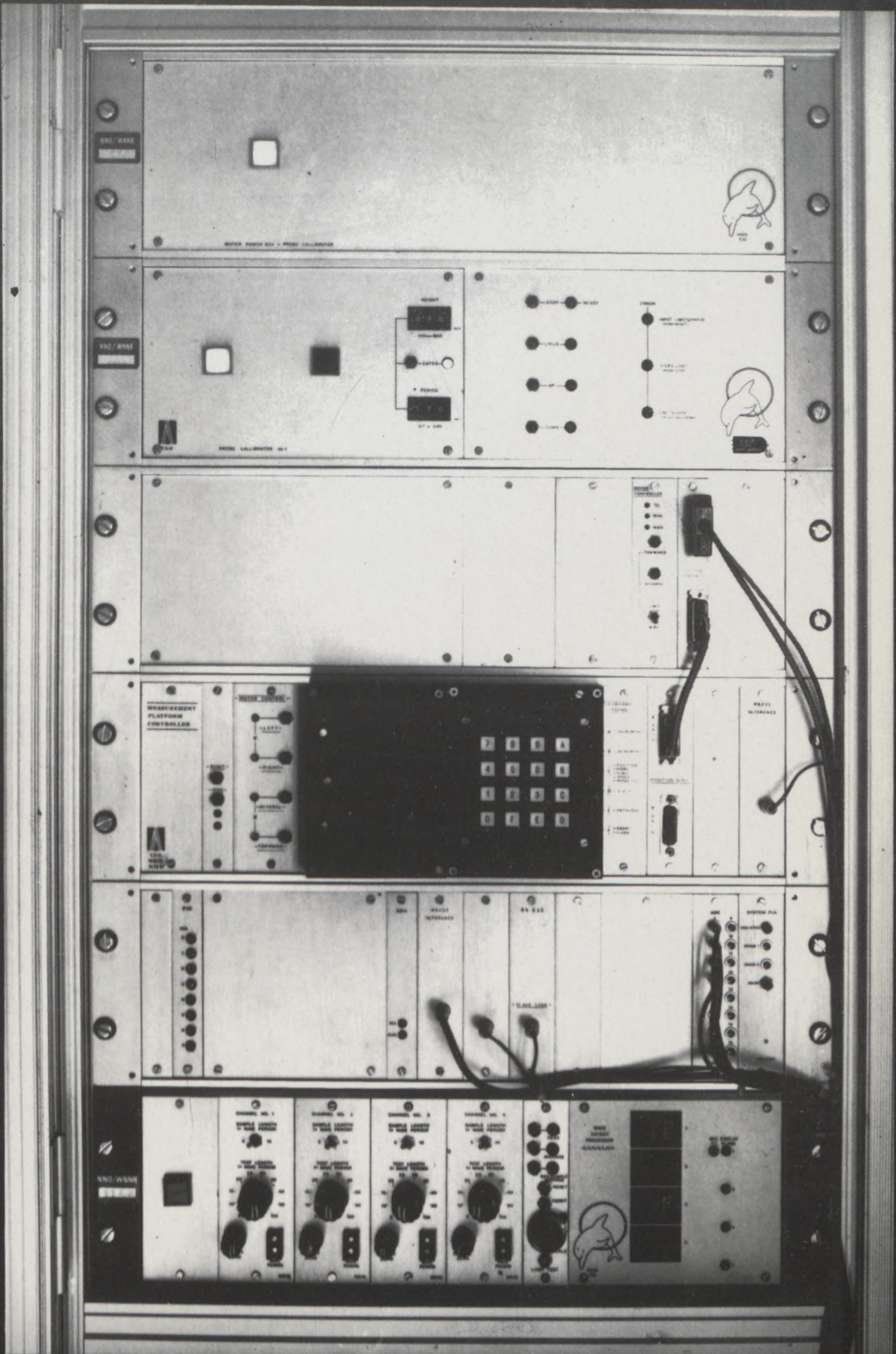


Fig. 5-5 b



**Trolley Controller and
Associated Instrumentation**

6. TROLLEY CONTROLLER

6.1 Introduction

The trolley controller was developed to control the speed and position of the measurement platform that runs on rails on top of a flume. Measurement instrumentation is mounted in an instrument rack on the platform while probes are lowered into the flume to make measurements such as of wave heights and current speeds. The type of studies done in these flumes (e.g. breakwater studies) requires the platform either to be stationary or to move at a known constant speed between any two points along the length of the flume. With the aid of this controller the operator is able to position the platform anywhere along the flume or move it at a selectable speed (between 1 + 10 cm/sec).

6.2 General Description

As both the MPC and the trolley controller basically control the movement of measurement platforms it was decided that rather than have two separate control programs it would be preferable to have one program that could adapt to both systems. The main differences between the two units are:

- (1) In place of the MK1 speed controller a new speed controller was designed using a single board microcomputer function card (Section 6.5).
- (2) The addition of an operating mode whereby the motor speed can be entered via the front panel keyboard in the stand alone mode or via the model control console keyboard under MHDAS control. Because a flume requires movement along one axis only, it was decided to use an MPC to control the trolley movement but with the Y axis disabled. Position-fixing in this single axis (X axis) of movement is done using the same method as described in Section 5.5 for the MPC. For a block diagram of the trolley controller, see drawing No HB81001.

6.3 Requirements to Interface to the MHDAS

The interfacing requirements for the trolley controller are the same as has been discussed in Section 5.3 for the measurement platform controller (MPC). Note that the MK2 speed controller is interfaced to the trolley control program and not to MHDAS.

6.4 Theory of Operation

The operation of the trolley controller is the same as for the MPC. Therefore, only the MK2 speed control and related operations will be discussed.

At program start-up, as part of the setup routine, the unit will determine whether the MK1 or MK2 speed controller is present. This is done by reading a target word from the communal memory of the function card. If it corresponds to a set value a flag (XSPMK2) is set to indicate that the MK2 speed controller is present, otherwise it is assumed that the MK1 speed controller is present.

Whenever the MK2 speed controller is present, the Y axis of movement will be disabled and the routines to enter different motor speeds will be enabled. Motor speeds of up to 10 cm/s in 1 cm/s steps may be entered via the front panel keyboard in the stand alone mode. Note that only the last two X axis display digits will respond showing the decimal value. By default the speed will always be set to 10 cm/s. When the speed is entered via the model control console two digits must always be entered, that is, 01 for 1 cm/s. The platform will move at the selected speed in the MOVE-TO and the TARGET mode but will move at the fastest speed (10 cm/s) in the HOME mode of operation.

To control the speed, the period between slots of a rotating disc on the motor shaft is monitored and compared with a calculated period (reference period) that is proportional to the selected speed. Using the available gear ratio we have 2,94

RETURN ENDS A COMMAND. RUN=REMOTE UNIT NO., 0 BY DEFAULT
 DATA FORMAT OPTIONS A=ASCII, B=BINARY, O=AUTO DATA OUTPUT.
 DATA DESTINATION OPTIONS V=VDU, P=PRINT, D=DAC, C=CDL, M=MODEM
 DISPLAY FILE INDEX X:(RUN)
 OUTPUT A DATA FILE O,(FILE NO.);OPTIONS:(RUN)
 DELETE A DATA FILE D,(FILE NO.):(RUN)
 READ A CDL FILE T;(R=REWIND,S=SEARCH)
 SETUP ADC S:(RUN)
 RUN ADC A:(RUN)
 MOVE TROLLEY M;OPTIONS:(RUN)

 TROLLEY CONTROLLER OPTIONS ; F=FORWARD , R=REVERSE
 IX=INCREMENT GRID POSITION, ID=DECREMENT GRID POSITION
 G=MOVE TO ..., GH=MOVE 'HOME', GT=MOVE TO 'TARGET' POS.
 P=DISPLAY\CHANGE TROLLEY PARAMETERS

FIG 6.2 a COMMAND LISTING

Operator Responses Highlighted

M;P:12

REMOTE UNIT NO. 1

PLATFORM X POSITION IS = 8,99
 PLATFORM X AXIS LIMITS ARE 1,50
 AND 25, 0
 X AXIS 'HOME' POSITION IS = 2, 0
 MY X TARGET POSITION IS = 0, 0
 X AXIS GRID SPACING IS = 1, 0
 PLATFORM X AXIS SPEED IS = 10 CM PER SEC.

DO YOU WISH TO CHANGE THIS?...Y OR N

PLATFORM X AXIS SPEED IS = 10 CM PER SEC.
 DO YOU WISH TO CHANGE THIS?...Y OR N
 YPLATFORM X AXIS SPEED IS = 02 CM PER SEC.

PLATFORM X POSITION IS = 8,99
 PLATFORM X AXIS LIMITS ARE 1,50
 AND 25, 0
 X AXIS 'HOME' POSITION IS = 2, 0
 MY X TARGET POSITION IS = 0, 0
 X AXIS GRID SPACING IS = 1, 0
 PLATFORM X AXIS SPEED IS = 02 CM PER SEC.

DO YOU WISH TO CHANGE THIS?...Y OR N

FIG 6.2 b PARAMETER LISTING

M;GH:12
 REMOTE UNIT NO. 1
 PLATFORM X POSITION IS = 7,89
 X AXIS 'HOME' POSITION IS = 6, 0

TO ABORT ENTER ANY KEY:(RUN)

Q:12
 REMOTE UNIT NO. 1

STANOP VER.821115 AT YOUR SERVICE.
 ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
 MODEL CONTROL CONSOLE NO. 1

FIG 6.2 c MOVE HOME

M;GT:12
 REMOTE UNIT NO. 1
 MY X AXIS DESTINATION IS = 5, 0
 PLATFORM X AXIS SPEED IS = 02 CM PER SEC.

TO ABORT ENTER ANY KEY:(RUN)

M:12
 REMOTE UNIT NO. 1

STANOP VER.821115 AT YOUR SERVICE.
 ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
 MODEL CONTROL CONSOLE NO. 1

FIG 6.2 d MOVE TO TARGET

M;G:12
 REMOTE UNIT NO. 1
 PLATFORM X POSITION IS = 8,68
 ENTER..Q FOR NO X AXIS MOVEMENT
 OR.....X PLUS NEW DESTINATION

X8
 PLATFORM X AXIS SPEED IS = 02 CM PER SEC.

TO ABORT ENTER ANY KEY:(RUN)

FIG 6.2 e MOVE TO POSITION

M;P:12
 REMOTE UNIT NO. 1

ENTER F(FORWARD) OR B(REVERSE) FOR 1st X AXIS POSITION FIX
 F

PLATFORM X POSITION IS = 8, 2
 IS THIS POSITION CORRECT?.....Y OR N
 Y

STANOP VER.821115 AT YOUR SERVICE.
 ENTER L COMMAND TO SEE THE COMMAND SET OF ANY UNIT.
 MODEL CONTROL CONSOLE NO. 1

FIG 6.2 f START-UP

motor shaft turns per 1 cm travel. The PTM on the function card uses the 1MHz clock to decrement its timers. To calculate the reference period the following formula is used:

$$P = \frac{t \cdot x}{C}$$

where C = PTM clock rate (1MHz)

t = pulses per second (disc has 100 slots, therefore
t = 294)

x = required speed in cm/s.

These calculations are done by using routine SPCON (flow diagram Figure E12) and the mathpack (Laidsky, 1978). The direction of movement is transferred to the speed controller where it:

- (a) Acts as a flag to start the speed controller program, and
- (b) is used to select the direction relay (MK1 speed controller uses a separate PIA to set the relays while the MK2 uses the PIA on the function card.

Section 6.5 will give a detailed description of the MK2 speed controller. Although the MK2 speed controller when used together with the trolley controller has only 10 different speeds, it can be changed to have a much larger speed range by using a different gear ratio or using it without a gearbox.

The output prompts to the model control console VDU are much the same as those for the MPC with the exception that only the x-axis prompts are displayed and that the speed prompts have been added. Figure 6.2 shows these prompts which can be compared with the MPC prompts in Figure 5.2.

6.5 MK2 Speed Controller

DC motors are used to move the measuring platforms, see Appendix C for motor data. The speed is varied by changing the armature voltage while the field excitation remains constant.

The armature voltage is obtained from a power rectifier bridge with two of its diodes in two arms replaced with SCRs. By varying the delay of the SCR gate trigger, the conducting angle of the 100 Hz half wave output from the bridge, and thus the armature voltage, can be controlled. The control of the output to the SCRs and the processing of the input data from the light fork is done by a function board which is a one-board micro-computer (drawing No MA79015). The program listing and flow diagrams of the MK2 speed controller is given in Appendix G.

The desired speed is entered via the front panel keyboard and the main MPC program then processes this value to obtain a period which is proportional to the required speed. This period will then act as a reference value for the required speed.

Data transfer between a control program and the function module program is performed via the communal RAM on the function board (communication between the MPC and a function card is explained in Section 5.5). The common RAM area for the MK2 speed controller is set for address \$A220-A23F.

When the main MPC program requires a 'move', the direction of movement (DIR) and the reference period are transferred to the common memory. On detection of the direction flag being set at the speed control module, the reference period and direction will be transferred to the speed controller RAM. The speed controller will now initialise itself by starting the period timer (PTM timer #1), clearing the interrupt mask, setting the pointer to the start of the period look-up table, etc (see Figure G1 for flow diagram). The table consists of times which will cause PTM timer #2 output to vary in period from 10 000 μ s to 18 000 μ s in steps of 10 μ s. The negative half of the period can thus be controlled between 5 000 to 9 000 μ s. By triggering the SCRs as soon as timer #2 output goes high the 100 Hz half wave ($t = 10\,000\,\mu$ s) can be controlled from almost zero (0,9 of full half wave) to half of the half wave. In order to ensure that misfiring of the SCR is eliminated, a 10 kHz signal from timer #3 is fed to the SCR gates (refer to timing diagram, drawing

No HB81008). Synchronisation pulses are obtained from a zero voltage switch which outputs a positive pulse at every zero crossing of the AC supply. These pulses are used to trigger the processor NMI routine (flow diagram, Figure G3) which reloads the SCR gate control timer #2. Amongst other tasks the NMI routine performs tests to check if the slotted opto schmitt switch is working and sets the look-up table pointer to the next timer value. The pulse obtained from the slotted opto schmitt switch is used to trigger the MPU IRQ routine where the opto switch test flag (FORK) is cleared. In the NMI routine this flag (FORK) is incremented, and if it exceeds a set value this will indicate that the opto switch is out of order. A watchdog timer used to prevent program runaway is also triggered in this routine (see Figure G2(a) for flow diagram).

Feedback for accurate speed control is obtained from the slotted disc and a slotted opto schmitt switch mounted on the motor shaft. The slotted disc which is used for fine position sensing now has the additional task of also supplying the required feedback. The period between the slots of the rotating disc is obtained by triggering the IRQ routine (flow diagram Figure G4) at the start of each slot. The count in timer #1 will indicate the period. Some average value is computed to dampen the effect of large changes in the period readings from the slotted disc. The method of obtaining this average period is summarised by the following steps:

- (a) $\text{Period} + \text{Period} + \text{Period} + \text{Period} = \text{SUM}$
- (b) $\text{SUM} \div 4 = \text{Average}$
- (c) $\text{SUM} - \text{Average} = \text{SUM of } 3$
- (d) $\text{SUM of } 3 + \text{New period} = \text{New SUM}$
- (e) $\text{New SUM} \div 4 = \text{New Average}$
- (f) Go back to (c).

To begin with four periods are added to obtain the first usable average. Subsequent average values are obtained by subtracting the average value from the SUM of four to obtain the SUM of three. The new period is added to obtain the new SUM of four from which the new average is calculated.

Within the NMI routine the average period is compared with the reference period with the end result that the speed is either incremented or decremented.

To reduce motor hunting or ringing at the required speed to a minimum, two corrective measures were investigated. First, the look-up table was made as large as possible resulting in small voltage changes between steps. Second, to give the motor time to respond to a voltage level, the same value from the period look-up table is reloaded 16 times as soon as the required speed is obtained. Both these measures gave positive results in the laboratory using the motor without a load. With the motor under load the latter measure did not improve but aggravated the situation. The following table gives the results of a speed test done over a 2 m distance. The motor had a 1 m start-up to stabilise the speed. Three tests were done where a period in the period look-up table was reloaded 16 times or 8 times or one time. All tests were done with a 60:1 ratio from motor to drive pulley.

Times taken for the platform to travel 2 m:

Speed cm/s	Desired time/2 m s/m	Resultant time taken for a 16:1 ratio	Resultant time taken for a 8:1 ratio	Resultant time taken for a 1:1 ratio
1	200	204	204	203
2	100	101	100	100
3	66,6	68	67	67
4	50	51	51	50
5	40	41	41	40
6	33,3	35,5	34	33
7	28,4	29	29	29
8	25	27	25	25
9	22,2	27	24	22
10	20	25	21	20

It was concluded from this result to use the 1:1 reload ratio. Use of the 60:1 gear ratio damped the effect of any ringing that still remained.

Two 2708 EPROMS are used in the speed controller where one holds the control program and the other the look-up table.

7. CONSIDERATIONS IN THE DESIGN OF NEW SYSTEMS

7.1 Guide to the Writing of a Test-control Program

This section acts as a guide to the use of the MPC or the trolley controller in conjunction with a test-control program. When the MPC is controlled from a test-control program under STANOP control the command buffer is used to transfer commands while the prompt buffer is used to transfer data. This control program may run in any unit in the data highway but will usually reside in the slave unit. The test control program could have the following format:

```

PSHA
PSHB (remember acc's)
JSR PBUFF (load data into prompt buffer)
JSR CBUFF (load command into the command buffer)
JSR COMINT (transfer the command buffer)
CHK JSR CHECKI (waits for a 'P' request and transmit prompt
buffer)
CMPA #$04 (if EOT character then done)
BNE CHK
CLR NAMBUF
PULA
PULB
RTS.
```

Notes

- (1) The command buffer is always cleared after transmission. (A standard procedure done by STANOP.)
- (2) The P request is transmitted up the data highway until it finds a unit with NAMBUF not cleared (byte 0 of the 11-byte prompt buffer).
- (3) The values entered into the prompt buffer are in HEX.

- (4) Only the MOVE-TO, SCALE, X and/or Y axis and the speed if a MK2 speed controller is used are transferred via the prompt buffer.
- (5) Only the G (move to), I (increment) and D (decrement) commands may be used.
- (6) Byte 4 of the command buffer must be set to indicate that the command originated at a test control program.
- (7) The remote destination must be reprogrammed if the control program is transferred to another unit (that is, bytes 5, 6 and 7 of the command buffer).

Prompt buffer format

Byte 0 - x coarse move to position
 1 - x fine move to position
 2 - y coarse move to position
 3 - y fine move to position
 4 - x coarse scale
 5 - x fine scale
 6 - y coarse scale
 7 - x fine scale
 8 - x speed
 9 - x and/or y
 10 - N/U
 11 - N/U.

Typical CBUFF subroutine to give a move to command (M; G:12).

```

CBUFF  LDAA #'M (recognition flag)
        STAA COMBUF
        LDAA #'G (instruction)
        STAA COMBUF + 2
        STAA COMBUF + 4 (test-control program/console flag)
        LDAA #01 (remote unit)
        STAA COMBUF + 5
  
```



```
LDAA #02 (nested remote unit)
STAA COMBUF + 6
RTS.
```

7.2 Programming Remote Instruments

This section acts as a summary of guidelines for adding STANOP and/or an external program to a remote instrument. It is suggested that the reader should be familiar with Section 3.3.

When an instrument requires to be incorporated in the MHDAS it is advisable to add all the cards referred to in Section 3.3 to that instrument. Doing this will allow the user to benefit fully from the versatility of the STANOP routines limited only by the amount of RAM available. However, if the situation requires, some boards may be omitted from, or shared with, the remote instrument (probe calibrator).

If the following tasks are not performed by the remote unit the system PIA board may be omitted:

- (i) Using the A-to-D convertor. A sense input on this card (sense 1) when high will initiate the sampling sequence while an abort button, on the front panel of this card, when depressed, will terminate the A-to-D routine.
- (ii) Using STANBG debugging routines. When using the debugging routines the abort button may be used to return the program to the debugging control routine (debugging can be done without the abort button).
- (iii) Using sense 2 as a special function input. Sense 2 may be used to inhibit or enable external functions, for example, a high level on sense 2 will automatically output gathered data to the VDU.
- (iv) Using the alarm. A high level on PAO will trigger the alarm.

If need for the debugging routines in STANOP is small (simple or pre-debugged program) the STANOP-PIC board may be omitted. The STANOP-PIC is used to stop the program at the requested break-point addresses. (Remember that the program must reside in RAM before it can be debugged; a routine in STANOP can be used to transfer programs from EPROM to RAM.) STANOP uses memory area \$C000 to \$FFFF and is mapped as follows:

\$C000 to \$C3FF	-	STANOP RAM
\$C400 to \$F7FF	-	STANOP program in EPROM/RAM board
\$F800 to \$FF7F	-	Highest EPROM in STANOP-PIC board
\$FF80 to \$FFFF	-	STANOP-PIC board RAM
\$FFCF	-	Stack pointer.

Whenever the STANOP-PIC board is omitted the highest STANOP EPROM is transferred to the EPROM/RAM board. Because the top RAM is no longer available, the interrupt vectors must be programmed into the highest STANOP EPROM, and the stack pointer, which is usually initialised at \$FFCF, must be initialised at some convenient available memory, say at \$C3FF.

When a remote instrument has incorporated in its system a PTM to perform its own independent tasks it may be shared with STANOP. STANOP uses the PTM to produce random delays to prevent communication contentions. Whenever a PTM is shared in this way care must be taken that it is set up correctly when returning from the external program to the STANOP control program or when routines using the PTM in STANOP are used by the external program.

When writing an external program it must be remembered that these external routines become subroutines of STANOP by the set of instructions as discussed in Section 3.3 and that an RTS (return from subroutine) will return the program to SYSRUN, the STANOP control routine. The address of the first of these instructions must be set on the ID board (Bytes 11 and 12). If any of these instructions are not used by the remote instrument, for example, if there are no external program set-up routines,

then each instruction cycle must be replaced by an NOP (no operation) so that the instructions following the external program set-up routines still have the same address in the program listing.

When STANOP routines are used by the external program these routines and their RAM scratchpad, if any, must be defined in the external program equate table.

The external program can be placed anywhere in memory between \$0000 to \$BFFF. However, it is advisable to position the external program at the top of this memory so that a block of continuous RAM can be installed from \$0000 up. STANOP checks for available RAM by searching for a continuous block of RAM from position \$0000.

STANOP cards switch positions.

STANOP-PIC card

- (i) Reset vector set to \$C400 (switches ON = low)
- (ii) 2708/2716 EPROM select switch set to suit
- (iii) RAM select switch switched ON.

ID card. Set-up as indicated on page 1 of the STANOP program listing, refer to Holroyd (1980).

System PIA

- (i) Address switches set to 1 (\$A076 - \$A07F)
- (ii) PIAs IRQA line switched to the bus IRQ line
(a switched to I).

PTM card

- (i) Address switches set to 0 (\$A090)
- (ii) PTMs IRQ line switched to the bus IRQ
- (iii) PTM #2 external clock (C2) set to divide by 100.

EPROM/RAM card

- (i) All programming switches in the off position (down)
- (ii) The address switches (A14, A15) set to 1 (\$C000)
- (iii) The RAM (STANOP RAM) switched ON
- (iv) EPROM select switches switched to suit.

8. TOWARDS AUTOMATIC MODEL TESTING

8.1 The Instruments in Use

The trolley controller and the probe calibrator have been installed and used in the 1 m flume and it can be said that they have been well accepted by those who have had the opportunity to use them. The operator saves time as he can control these instruments from the model-control console. The automation of the probe calibrator will make the results from the probes more reliable as recalibration of the probes has become a simple task, and this will inspire the operator to calibrate the probes more often while performing tests. The automation of the trolley has been a major step forward because previously the direction of the motor (3 ph) was changed by engaging a limit switch on the trolley which in turn activated the direction change-over contactors. Limit blocks were fastened to the rails to trigger the limit switches. To change the speed of the 3 ph motor the gearbox ratio had to be changed, which required a stop-watch (time) and a tape measure (distance) before the speed could be calculated. The measurement platform controller installed at the Granger Bay/Richards Bay basin has not been used for other than demonstration and testing purposes; therefore, a true user response cannot be given. However, it is expected that

- (i) the controller will shorten the time taken to move the measurement platform from one position to another,
- (ii) it will accurately position the platform anywhere in the model basin, and
- (iii) it will be controlled from the model control console.

Modifications, improvements and the incorporation of new ideas to the MPC control program are inevitable until the new Richards Bay model is fully operational.

8.2 User Response

One of the major objectives in developing and automating the instrumentation referred to in this thesis was to make them simple to use. To achieve this it was necessary to incorporate the valid suggestions of the prospective users into the relevant operating systems. After the particular instrument had been completed it became important to investigate the constructive criticism from the users and to modify the systems where possible.

Some of the long and untidy prompts originally used by the probe calibrator evoked criticism. These shortcomings were discussed with the prospective operators and then corrected to suit their needs. Among the positive suggestions received from the users of the probe calibrator some were:

- (a) To use the unit as a surveying tool in model basins where sediment transport tests are done (False Bay model) or
- (b) to vehicle any sensor or probe used in model tests.

The probe calibrator has been used to drive probes, on rails, across the width of the 5 m flume to test the linearity of the newly acquired irregular wave generators.

The TARGET mode was added to the MPC and the trolley controller as a result of user response.

When the user is familiar with the MHDAS he should not require much training in order to control these instruments from the model-control console since the VDU output prompts are short, instructive and user-friendly, while the commands are entered using the standard STANOP format. It is suggested that the user should first be able to operate the instruments in the stand alone mode before he attempts to control them from the model-control console. It has been proved at the NRIO open day that an outsider can perform simple tasks like moving the probes out

of the water with relatively little training as long as he adheres to the instructive prompts displayed on the VDU.

8.3 Future Developments

As stated previously it is hoped to make the Richards Bay model the first fully automated model. Towards achieving this it has been decided to have all the model hardware completed by the end of March 1984, this being:

- (i) The model control cabin will be equipped with a model control console.
- (ii) Both platforms belonging to this model basin will be equipped with position sensors, motors and a measurement platform controller. (The existing equipment on the prototype measurement platform will be transferred to one of these platforms.
- (iii) Each platform will be equipped with a slave unit. The slave unit will house an A-to-D converter and enough RAM to gather data from the wave probes. Test-control programs will later be installed into these slave units.
- (iv) Each platform will be equipped with at least one modified (under STANOP control) probe calibrator including a set of four wave height probes.
- (v) A screened cable will be installed to communicate between the slave unit and the model control console. This cable will run along overhead rails in a looped fashion to take up the slack (matching the existing system).

To fully utilise the MPC and probe calibrator and to obtain the ultimate goal of automatic model testing, it is required to start working towards the development of model-test control programs. Two different test control programs will be investigated and installed into the slave units.

A. Wave generator linearity-test

Before measurements are done inside a model basin (e.g. wave-heights, sediment transport, etc.) the linearity of the wave generators is tested to verify that the required waves are generated. This test is done by measuring the wave output at an equal spacing, 2 to 3 m in front of the wave generators, along the entire length of the model.

To automate this test the program the flow diagram of which is shown in Figure 8.1 will be investigated. Before the program is started the operator will be requested to set up the A-to-D and to enter the expected RMS value of the generated wave. The program will start by moving the platform to the TARGET position (first test point) in front of the wave generators. (Note that in the automated mode the platform will always start and end in the HOME position.) The probes are then lowered into the water and a time delay is given to permit the wave height measuring electronics to stabilise. At this stage the first test is run by jumping to a separate program (module) to run the A-to-D converter and to obtain the RMS value of the test. This A-to-D and RMS value module will be an adapted version of the already written REFLEC program (Holroyd, 1980). If the RMS value obtained at this test position is within limits the y-axis value is incremented and the platform is moved to this new position. If the RMS value obtained is outside the limits the operator will be informed via the VDU. At this point the wave generators may be re-adjusted and the test run again. After the last position has been tested and when the probes are lifted out of the water the platform will return to the HOME position.

B. Wave height tests by grid

To automate this type of test the following program will be investigated (refer to the flow diagram in Figure 8.2). This program will make use of pre-programmed modules:

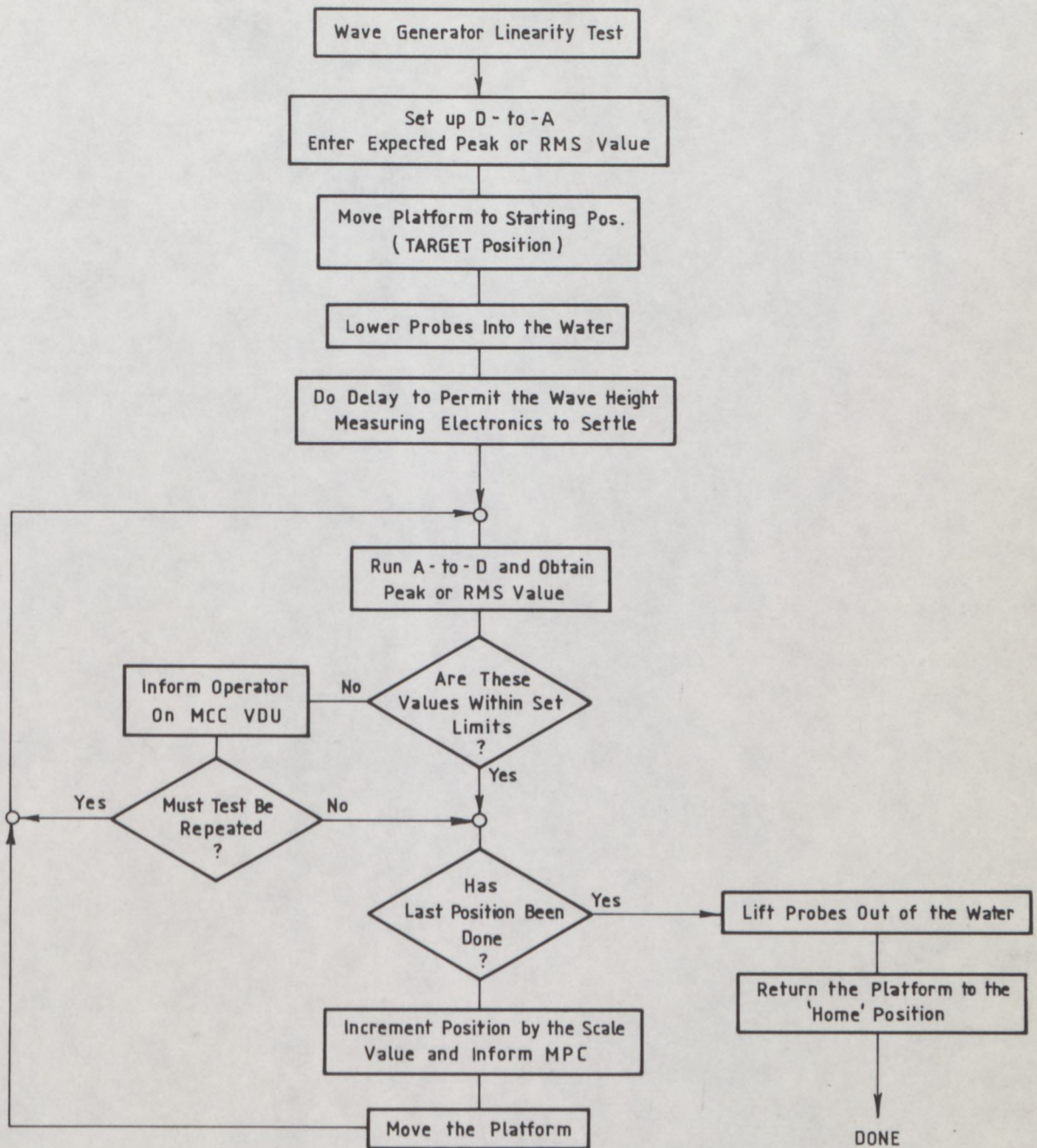


FIG. 8-1

WAVE GENERATOR LINEARITY TEST

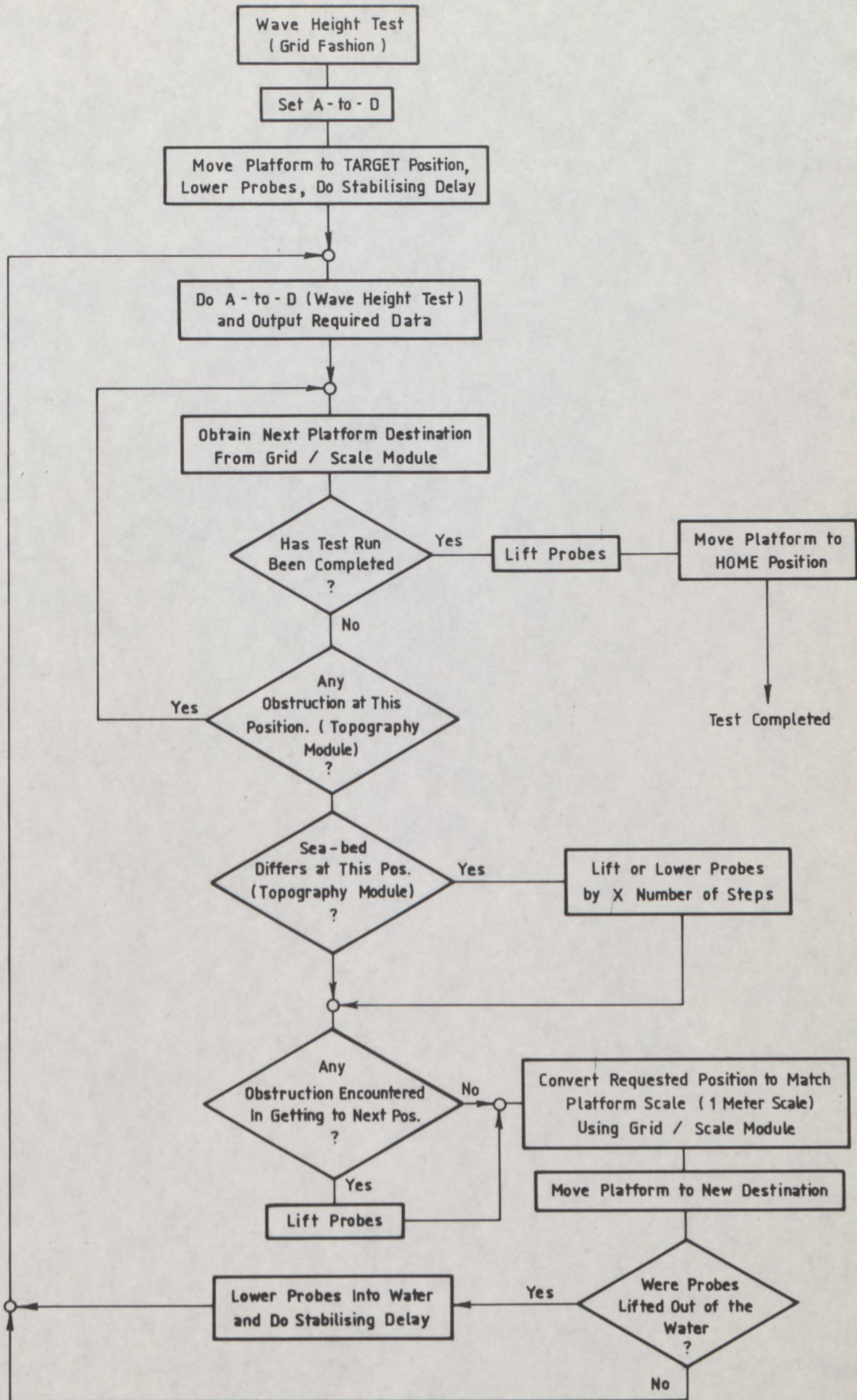


FIG. 8.2

WAVE HEIGHT TEST (GRID FASHION)

- (i) A grid/scale module which will contain the current test grid and the conversion factors to convert these grid positions to be actual positions on the platform scale (remember that the platform has a 1 m scale).
- (ii) A topography module which will contain the positions, with reference to the grid in (i) of harbour walls, other obstacles and the depth of the sea bed.
- (iii) An A-to-D module which can either output a complete data file to a higher unit in the hierarchical chain or to compute statistical data and RMS values of the test and output these values to a higher unit (this is the same module as used in the wave generator linearity test).

The pre-programmed modules will be self-contained programs which by changing a few parameters (model constants) can be used in any model basin. These parameters must either be entered via the model control console or programmed into EPROM prior to testing the model.

Again, before the program starts the operator will be requested to set up the A-to-D. The automated program will start by moving the platform from its HOME position to the TARGET position (first test point on the measuring grid). At this position the probes will be lowered into the water and a time delay will be given for the wave height measuring electronics to stabilise. The first test will be conducted and the required results will be transferred to storage. The next test position is obtained from the grid/scale module. The topography will be checked and after the corrective steps are taken, if any, the position is converted to the platform scale. The platform will move to its new destination where the next wave height test is run. After the last position on the grid has been tested the probes are lifted out of the water and the platform returns to its HOME position.

It is hoped that both these programs will be operational in the second half of 1984.

On the basis of the experience gained by this project it is intended:

- (a) to extend this facility at the other measurement platforms in the model hall,
- (b) to interface further instrumentation (more probe calibrators) to the MHDAS,
- (c) to apply the probe calibrator principles to support other types of sensors (e.g. current sensors),
- (d) to apply the ideas (experience) gained to develop new types of instrumentation used for model testing, and
- (e) to incorporate other machines (tide controllers, irregular wave generators) into the MHDAS.

8.4 Costs and Justification

By the end of 1984 the costs incurred in automating the Richards Bay model will be as follows:

1 x Model control console	R6 000
2 x MPC & motor supply unit	6 000
2 x Modified probe calibrators	3 000
2 x Slave units	4 000
Workshop materials	5 000
Other hardware (motors, wire, etc.)	<u>2 000</u>
	<u>R26 000</u>
Workshop time	R10 000
Electronics System Div. time (3 years)	<u>130 000</u>
	<u>R140 000</u>

Total cost R26 000 and R140 000 = R166 000.

The costing figures show that the Electronics Systems Division's time contributes to most of the costs involved with this project. About 90 per cent of this time was used for development while the cost of extending this facility to another model basin would be only about R12 000 (3 months to complete). Once this facility has been installed at a model basin it becomes part of that basin with the only remaining electronics expense being maintenance.

The probe calibrator has already proved its merits and the next natural step was to interface it to the MHDAS. It has already been stated that this would inspire the operator to check the probe response more often, so as to give more reliable results.

Building a new Richards Bay model would cost about R60 000 while its running costs, with three persons employed on the model, will be about R6 000 per month. At present the collection of data is a time-consuming task due to the limitations in the speed and accuracy with which the platforms are moved from one position to another. Because of the monotony of the tests as done at present, the accuracy of the data is likely to suffer towards the end of the test. Using the MPC will speed up and position the platform more accurately giving more reliable results and a saving in model running time. By introducing a test-control program into the system it is possible to employ one less person on a model as the positioning of the platform and measurements are done by a pre-written program. If, say, one less person is employed on a model and if the running time is cut down from 4 months to 3,5 months the running costs would be R14 000 instead of R24 000, giving a saving of R10 000. The Institute will also save by a shorter run of a model as water and electricity costs are booked against overheads. Because of the positional accuracy by which a particular test can be re-run reliable comparisons of the results can be made, especially when small changes are made to the model topography (e.g. harbour wall positioning).

On completion of this project, the measuring instrumentation on the platform will for the first time be linked to the model control console which in turn can be linked to the minicomputer via the mini-interface. This will enable on-line, real-time, analyses of tests to be done, which will increase the efficiency of model testing.

In development, one important aspect, the value of which cannot be estimated, is the amount of experience that is gained which will be used to develop and automate new and existing instrumentation. Hardware and software routines can be copied and modified to be used in other systems, as for example, the MK2 speed controller.

8.4 Conclusion

The probe calibrator, trolley controller and measurement platform controller have successfully been interfaced to the MHDAS so that they are therefore able to operate as stand alone or MHDAS-controlled instruments, and tests run so far show that they fulfill their design requirement. Their reliability has been relatively good with only a few minor hardware faults.

The advances made have contributed significantly to automated model testing.

REFERENCES

ALTRONICS (1980). Thyristor convertors.

HOLROYD, S G (1980). Coastal engineering model hall data-acquisition system. CSIR Research Report 383.

HOLROYD, S G (1981). Wave height processor. NRIO Memorandum 8109.

HOLROYD, S G (1982). NRIO microcomputer system. CSIR Technical Report T/SEA 8134.

KUIPERS, J J (1982). Probe calibrator. NRIO Memorandum 8204.

LIADSKY, J (1978). 24-bit floating point routines for addition, subtraction, multiplication, division, square root, sine, cosine and tangent. M6800 User Group Library, 89: 1-10.

MOTOROLA (1982). Microprocessors data manual.

APPENDIX AMPC/TROLLEY CONTROLLER PC BOARDS

	<u>Page</u>
Description of MPC PC boards	A2
<u>Drawings</u>	
Drawing No HB81001 MPC block diagram	A7
Drawing No HB81002 MPC X and Y position finder	A8
Drawing No HB81003 MPC interface 1	A9
Drawing No HB81004 MPC interface 2	A10
Drawing No HB81005 MK1 motor controller	A11
Drawing No HB81006 MPC motor power supply	A12
Drawing No HB81007 MPC position plate and sensor	A13
Drawing No HB81008 MK2 motor controller timing diagram	A14
Drawing No HB81009 MPC wiring diagram	A15
Drawing No MA79001 CPU board	A16
Drawing No MA79004 PIA board	A17
Drawing No MA79007 PTM board	A18
Drawing No MA79009 RS232, TTY interface	A19
Drawing No MA79013 STANOP-PIC board	A20
Drawing No MA79015 Function board	A21
Drawing No MA79016 System ID board	A22
Drawing No MA79019 Hexadecimal keyboard module	A23
Drawing No MA79020 12-digit led display module	A24
Drawing No MA79022 STANOP 16k EPROM/RAM board	A25

DESCRIPTION OF MPC PC BOARDS

For convenience the circuit diagrams of the microcomputer cards with switches set as used by the measurement platform controller are included (drawing No MA79001 to MA79022). For a detailed circuit description of these cards consult Holroyd (1982).

The cards developed for the MPC conform to the standards laid down for the microcomputer system cards, card size being 185 mm × 100 mm and using the DIN 41612 C64 edge connector and mounted in a 19'Elma card frame.

(1) MPC X and Y position finder card, drawing No HB81002

This card is used to count the pulses obtained from the slotted disc on the motor shaft giving one output pulse for every centimetre of platform travel. Provision is made to accommodate the wide range of drive wheel sizes and gearbox ratios encountered on different platforms so that one pulse per centimetre is always obtained. Although this card houses two similar X and Y counting circuits only one is shown in the drawing.

Two pre-settable up/down counters (74LS193) are used giving a maximum count of 256 per centimetre. In the forward direction the counters count up until a set value is reached when a pulse is generated (IC5a 74LS123). This pulse resets the counters to zero and triggers the position-fixing module (fine position update trigger pulse). In the reverse direction, however, the counters count down until zero is reached at which point a pulse is generated (IC5b). This pulse presets the counters to a set value and also triggers the position-fixing module. A component header on this card is provided where these reset and preset values may be set in HEX by using suitable links. The up/down mode of the counter is controlled from PIA4 while ICs 1 and 2 are used to shape the input pulses from the slotted disc. This card also accommodates the coarse position line receivers (ICs 13, 14, 15) and circuitry to trigger the position-fixing module for a coarse position update. Since both axes generate a pulse

at pin 6 of IC4 (output to CB1 of position-fixing module) a re-triggerable monostable (IC 15) is used to generate a pulse so that the position-fixing module can determine whether it is an X or Y axis update.

(2) MPC interface 1, drawing No HB81002

This card is made up of seven separate circuits which are:

- (a) Direction OR. Each motor control pushbutton (override control) is wired to a two input OR gate (IC1, 74LS32) the second input being for a possible additional controlling input. The outputs from IC1 are fed to a four input NOR gate (74LS260) to generate a strobe pulse whenever any of the pushbuttons is depressed. The strobe pulse is fed to CB2 of PIA port 2 while the push-button input is decoded via bits 4 to 7 of PIA port 1.
- (b) Led driver. To control the mode indicating leds.
- (c) Speed frequency/ G^1 ; G^2 interface. (i) Speed frequency, to isolate the PTM generating the speed frequency from the MK1 speed controller in the motor control unit. (ii) G^1 ; G^2 interface, to isolate MK2 speed controller, which is situated in the MPC mainframe, from the motor control unit.
- (d) Watchdog pulse generator. This circuit is used to reset the MK2 speed controller when the control program has lost control over the motor. If a timer trigger pulse is missed (generated by the MK2 motor control program) the 555 timer which is wired to act as a monostable will generate a reset pulse that will re-initialise the speed controller.
- (e) Direction led/relay drivers. These control the direction leds and isolate the relay control pulse from the relay situated in the motor control unit.

- (f) X or Y position interface. This circuit is used to isolate the position-sensing circuitry in the motor control unit from the position-sensing circuitry in the measurement platform controller.
 - (g) Zero voltage pulse generator. This circuit is used to generate the output pulse required by the MK2 speed controller as a sync pulse every time the AC input goes through zero volts.
- (3) MPC interface 2, drawing No HB81004

This card is housed inside the motor control and consists mainly of line receivers/drivers (74LS241) used for pulse shaping on the position sensor and motor control lines between the motor control unit and the MPC. The other circuitry on this card is used only in conjunction with the MK2 speed controller and consists of:

- (a) A pulse transformer to isolate the motor power SCRs from the SCR gate pulse generating circuitry.
- (b) A Hall effect IC (RS 304267) used for motor over-current limiting. The power line to the DC motor is coiled around this IC with the result that one output increases linearly in voltage while the other decreases linearly for a linear increase in magnetic flux density. The output is then amplified (LM747) and compared (LM311) to an adjustable limit that is proportional to the maximum motor armature current.

- (4) MK1 motor controller, drawing No HB81005

Motor speed is controlled by changing the motor armature supply voltage. This is done with the aid of an SCR-type power bridge (see Drawing HB 81006) whose angle of conduction can be varied.

The voltage amplifier IC 14(a) computes the error between a reference voltage which is proportional to the desired speed and the feedback voltage which is proportional to the actual speed. The reference voltage is obtained from the 9400 frequency-to-voltage converter IC 13 while the feedback voltage is obtained from the armature via the feedback amplifier IC 14(b). The feedback amplifier is a non-inverting amplifier with a gain adjustment whereby the maximum speed can be adjusted. The error output from the voltage amplifier is fed to a comparator IC 15(a).

Negative armature current feedback, derived from a shunt resistor in the armature circuit, is applied to the current amplifier IC 14(c) via terminal IFB. Should the armature current exceed the set reference (current control), the amplifier output will go positive which will reduce the output from the voltage amplifier IC 14(a).

IC 15(b) is a linear ramp generator (negative going from 5V) which is reset to 5 V for every half-cycle of mains frequency. At every zero crossing of the AC supply, T1 and T2 produce a positive going pulse. T2 drives T3 which resets the ramp while T1 produces a pulse to switch the comparator IC 15(a) off.

The output from the ramp generator is fed to the voltage comparator IC 15(a) where it is compared with the error voltage obtained from the voltage amplifier IC 14(a). When the error voltage becomes more positive it will equal the ramp voltage earlier, causing the output of the comparator to switch positive and thus initiate the output oscillator sooner. The output from the oscillator is fed to the power SCRs to ensure that they are switched on.

The required speed reference voltage is obtained via a 9400 frequency-to-voltage converter IC 13 where the control frequency may be generated either at a PTM (external control) or by a local oscillator IC 4 (internal control). The power supply zero volt (common) of the speed controller is live with respect to

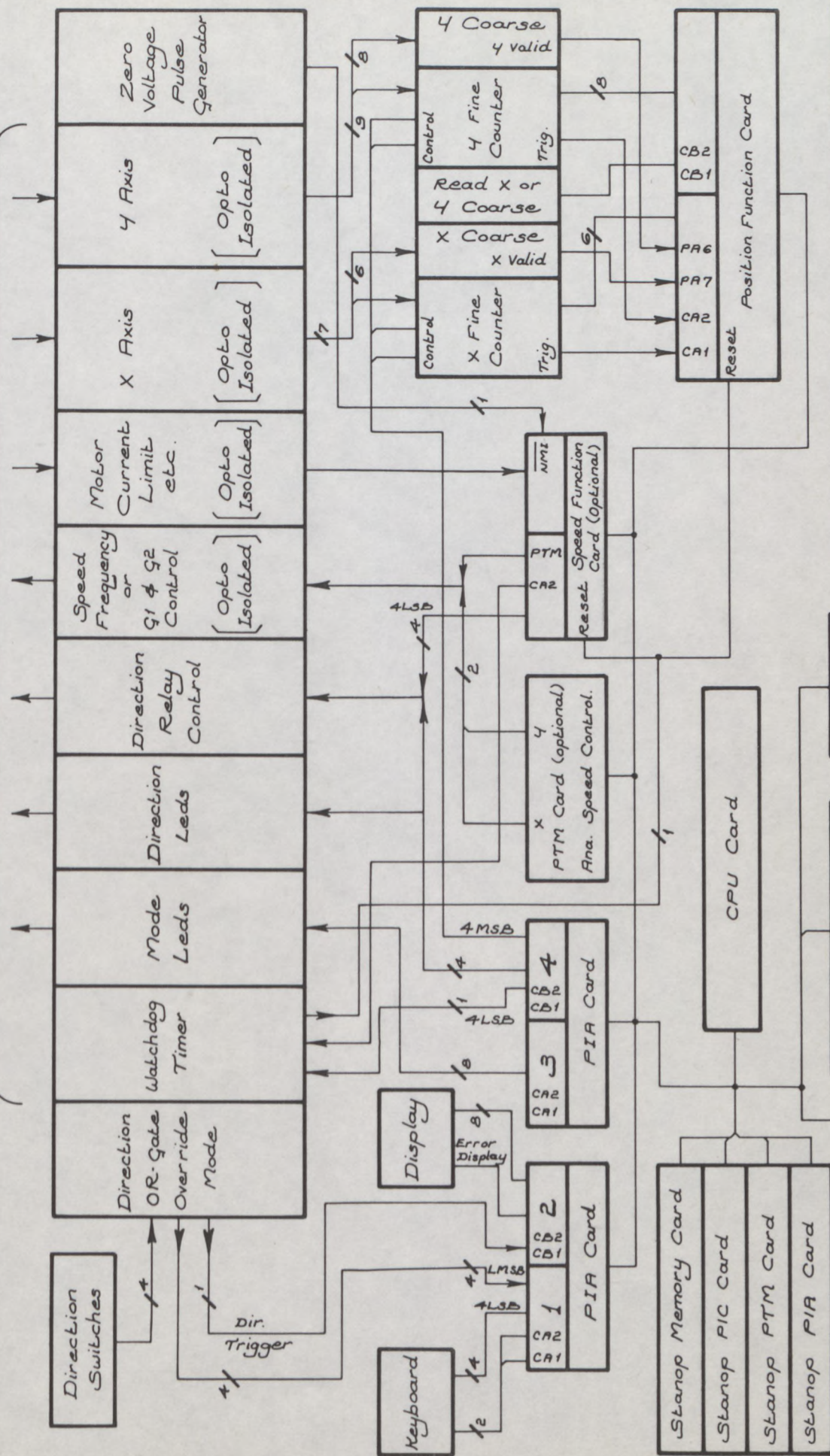
mains earth because it is connected to the motor power SCR cathodes. The oscillatory circuit is isolated from the zero volt supply by an MCC 671 opto-isolator IC 9.

Whenever the forward or reverse button (internal control) is depressed or released ICs 3 and 5 will produce a slow positive or negative going ramp. This slow-changing voltage is fed to a VCO IC 4 to produce a frequency that will smoothly increase or decrease the motor speed. A comparator and flip-flop (ICs 10 and 11) prevent the motor from changing direction before it has stopped.

Darlington transistors (MPSA12) are used to switch the direction relays and speed-controller inhibit circuit. The contacts of R11 force the input of amplifier IC 14(a) negative thus inhibiting the trigger circuits. The circuit, ICs 7 and 8, associated with the relay is to prevent both direction relays being activated simultaneously.

(5) MPC motor power supply, drawing No HB81006

The power supplies for the motor supply unit are shown in this diagram. There are three regulated supplies, +5 V, +15 V and -15 V. Note that the +15 V and -15 V supplies are isolated from mains ground. A bridge rectifier is used to supply the DC motor field voltage while an SCR power bridge is used to control the armature voltage. Also shown are the current feedback resistor (MK1 speed controller current limit), current coil (MK2 speed controller current limit) and the wiring of the direction relays.



MPC BLOCK DIAGRAM

NRIO

designed: J.J.K.	date
------------------	------

drawing number

HB 81 001

MPC 'X' and 'Y' POSITION FINDER

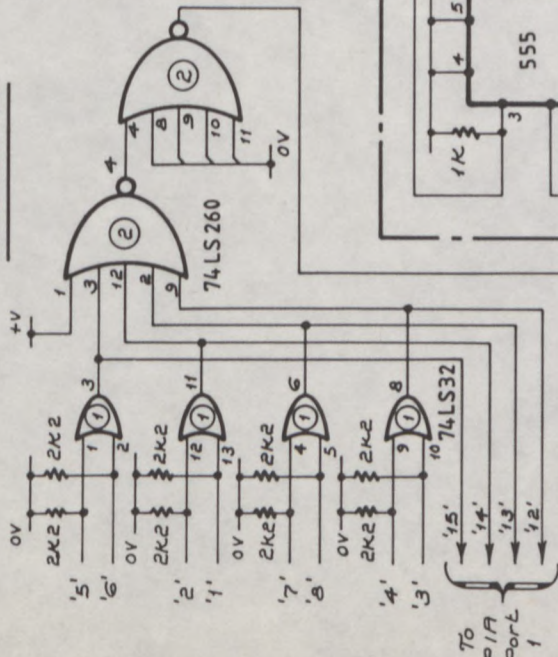
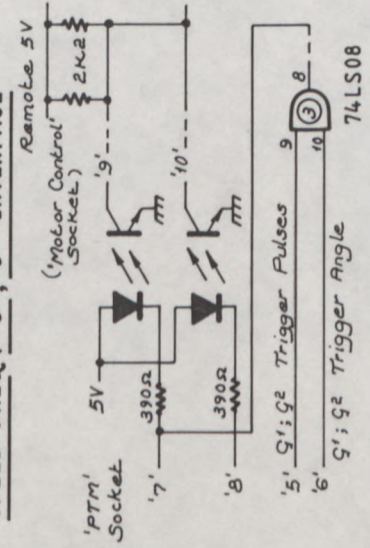
NRIO

designed: J.J.K.
drawn: Pat.

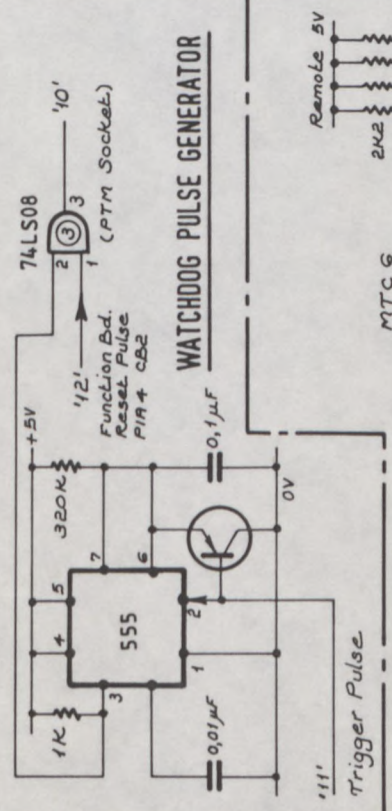
date
February, 1983

drawing number
HB 81 002

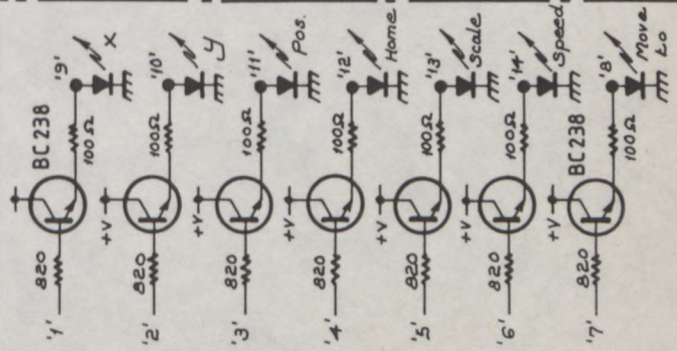
DIRECTION OR.

SPEED FREQ. / G^1 ; G^2 INTERFACE

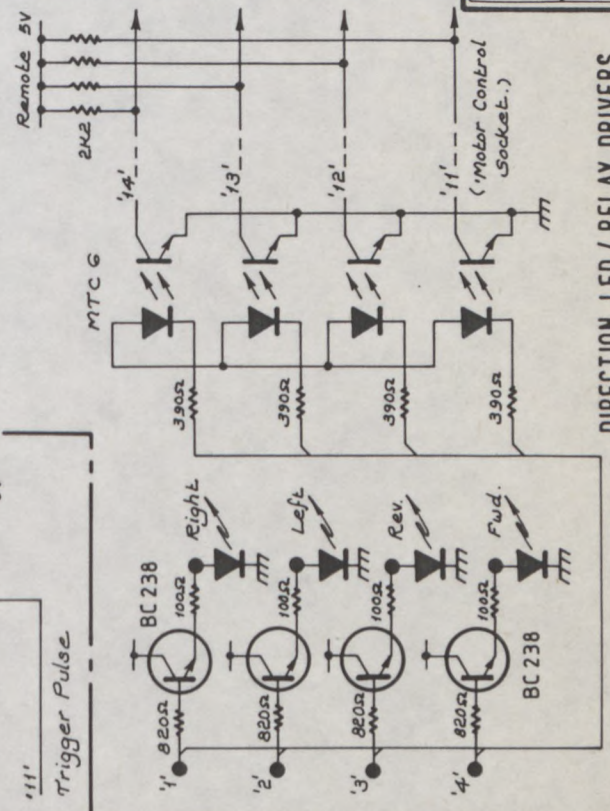
WATCHDOG PULSE GENERATOR



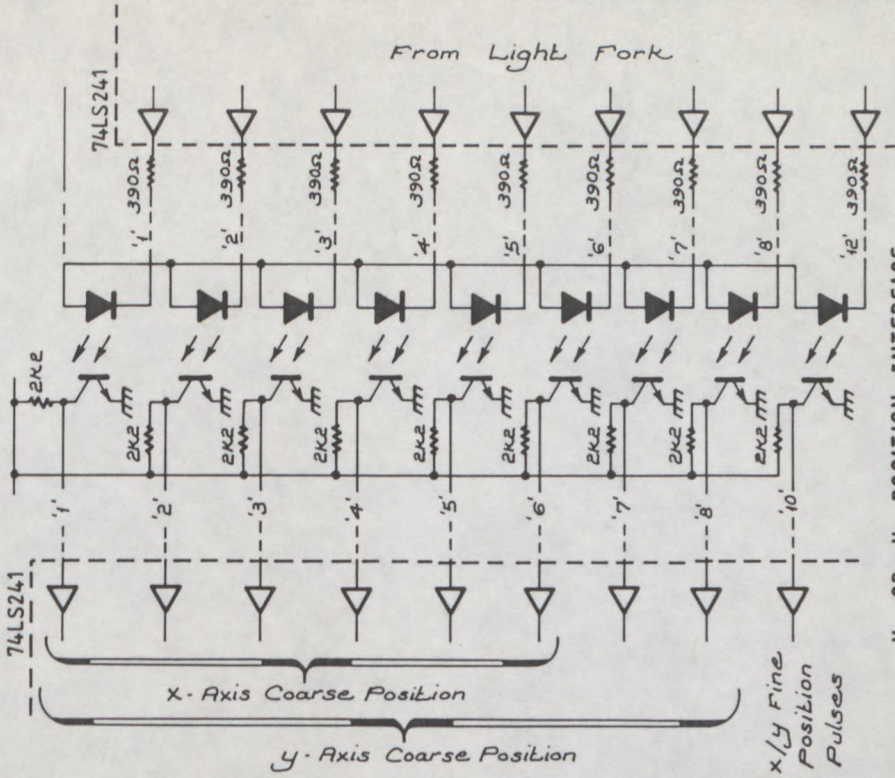
LED DRIVERS



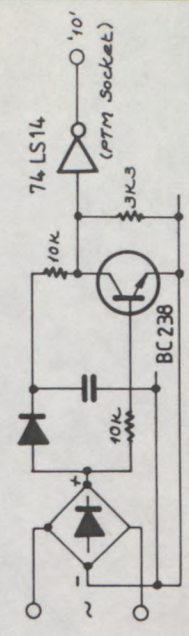
DIRECTION LED / RELAY DRIVERS



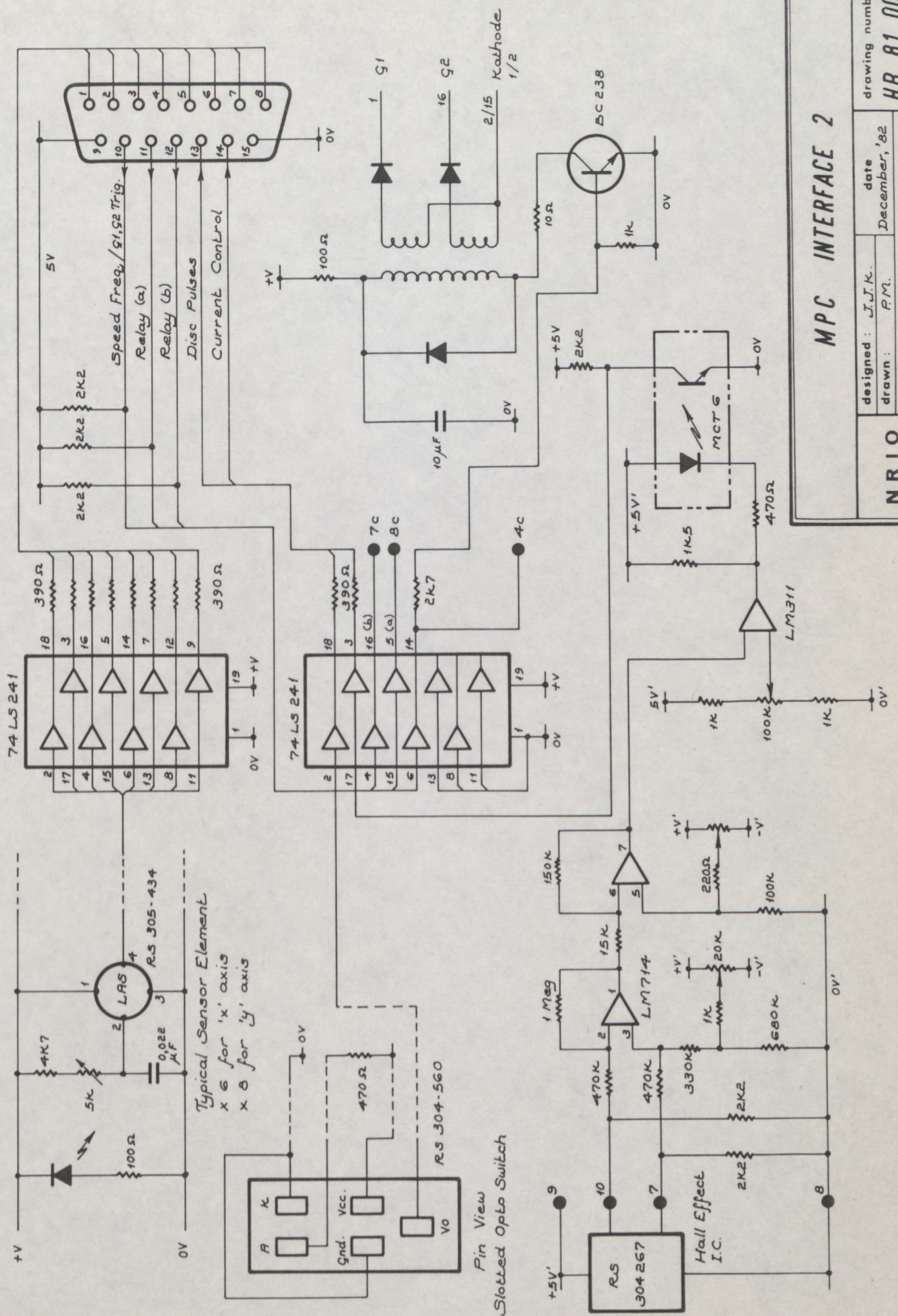
X OR Y POSITION INTERFACE



ZERO VOLTAGE PULSE GENERATOR



N R I O	designed : J. J. K.		drawing number HB 81 003
	drawn : Pat.	date March, 1983	
MPC INTERFACE 1			
Electronics & Instrumentation Group			

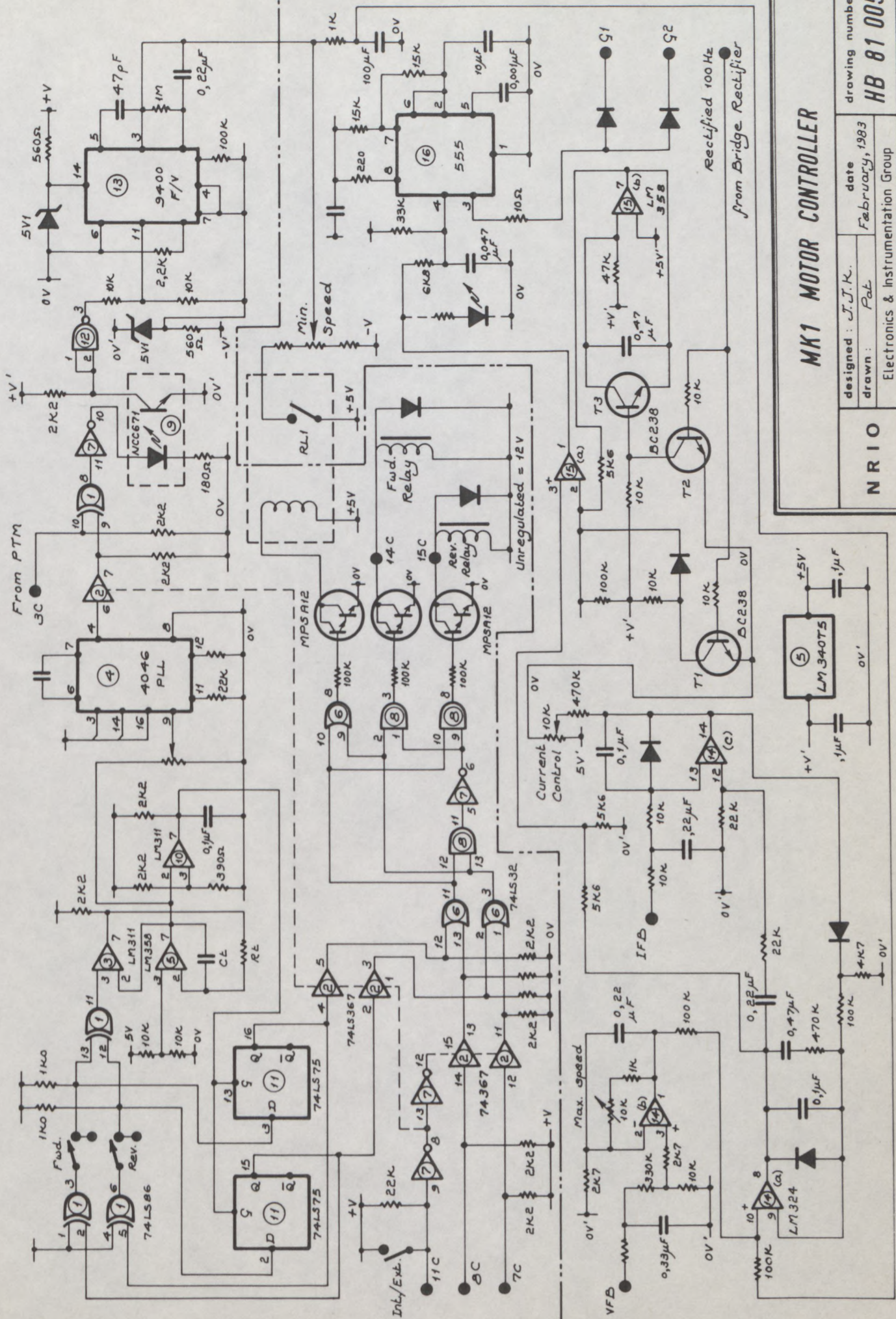


MPC INTERFACE 2

NRIO

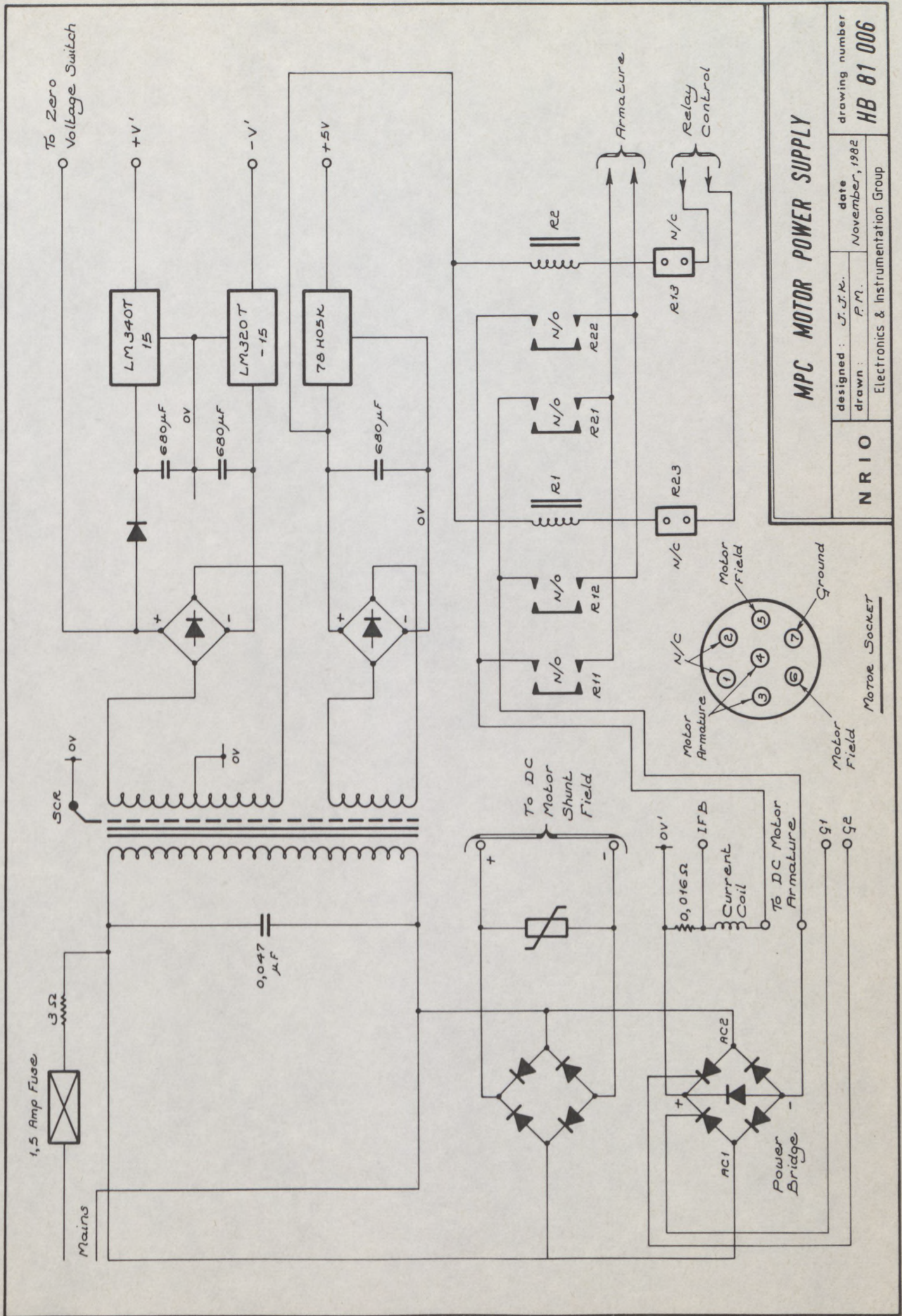
designed: J.J.K.	date
drawn: P.M.	December, '82
Electronics & Instrumentation Group	

drawing number
HB 81 004



MK1 MOTOR CONTROLLER

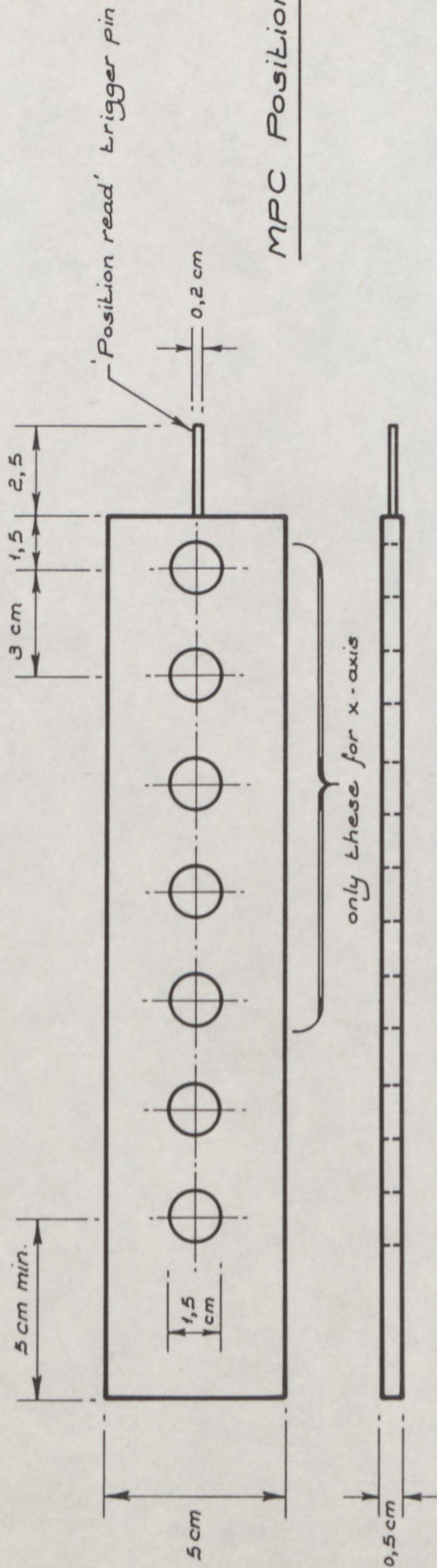
NR10
 designed: J.J.K.
 drawn: Pat
 date: February, 1983
 drawing number: HB 81 005
 Electronics & Instrumentation Group



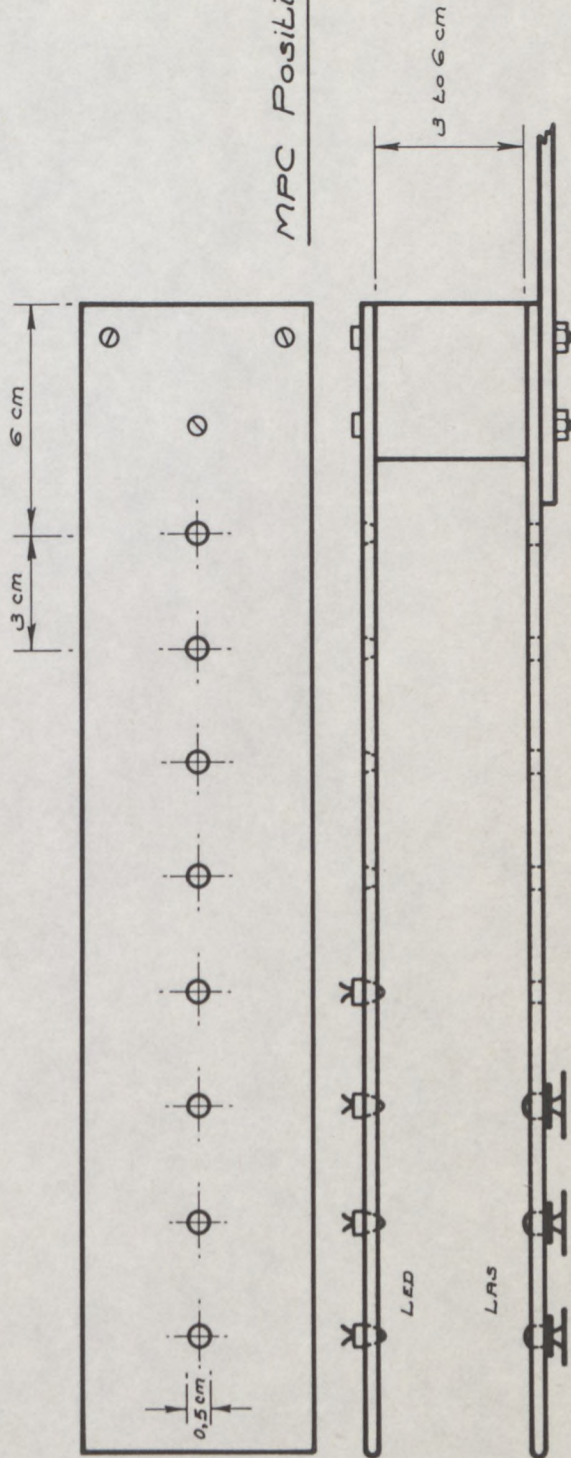
MPC MOTOR POWER SUPPLY

drawing number	HB 81 006
date	November, 1982
designed by	J. J. K.
drawn by	P. M.
Electronics & Instrumentation Group	

Motor Socket



MPC Position Plate

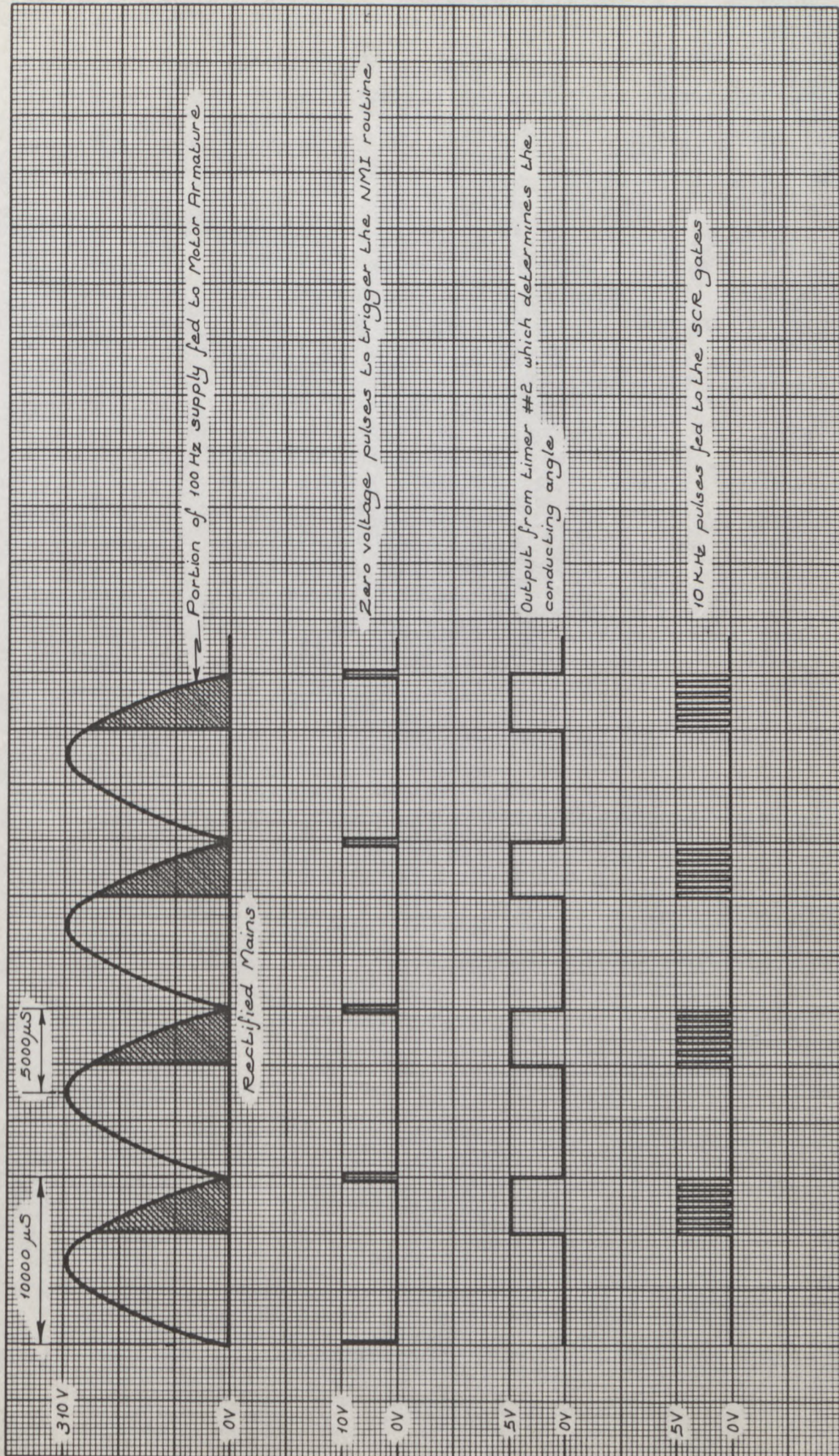


MPC Position Sensor

y - Sensor only

MPC POSITION PLATE AND SENSOR

NR 10	designed	J.J.K.	date	drawing number
	drawn	Pat		
Electronics & Instrumentation Group				



MK2 MOTOR CONTROLLER TIMING DIAGRAM

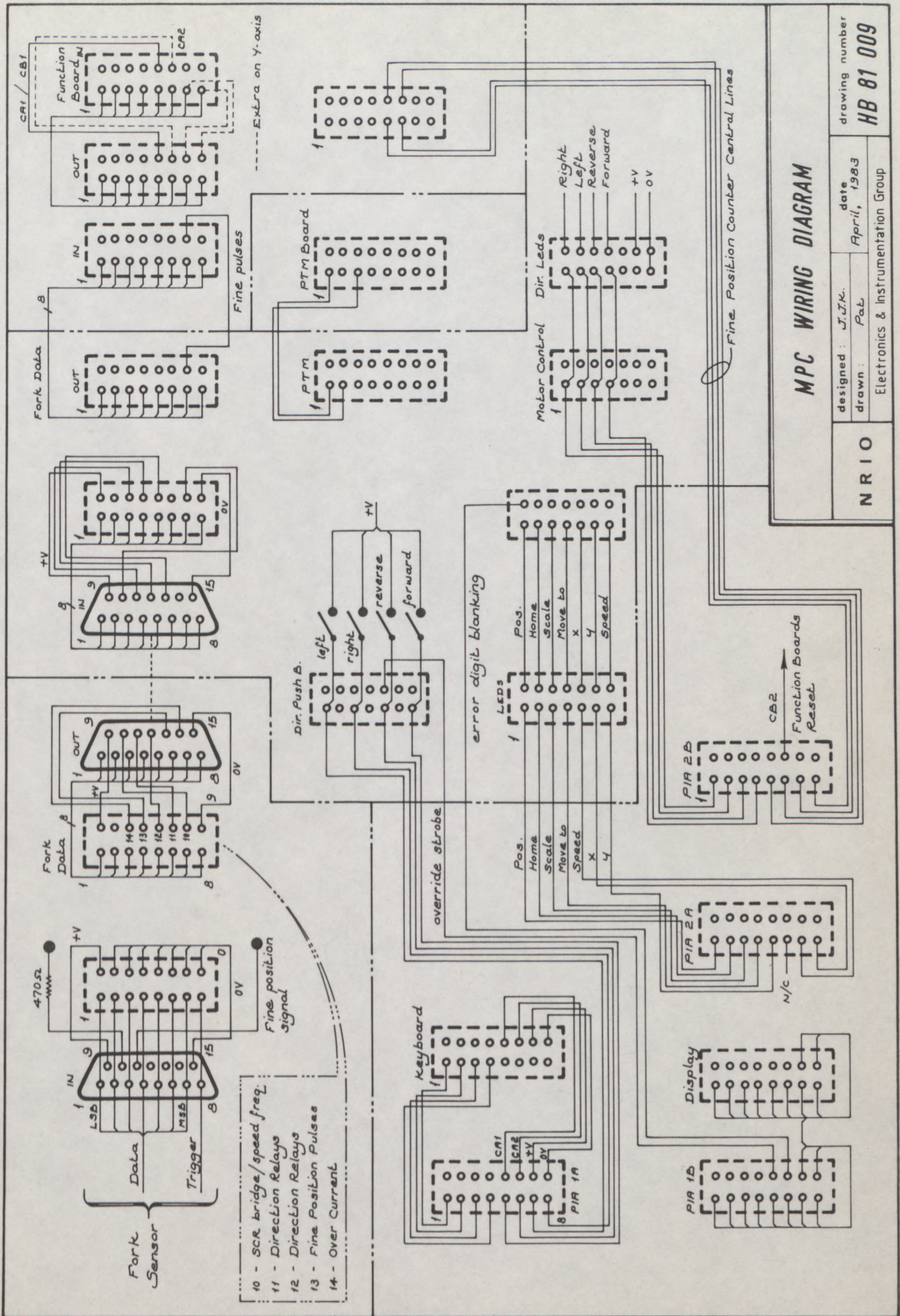
NR10

designed J.J.K.
drawn Pat

date March, 1983

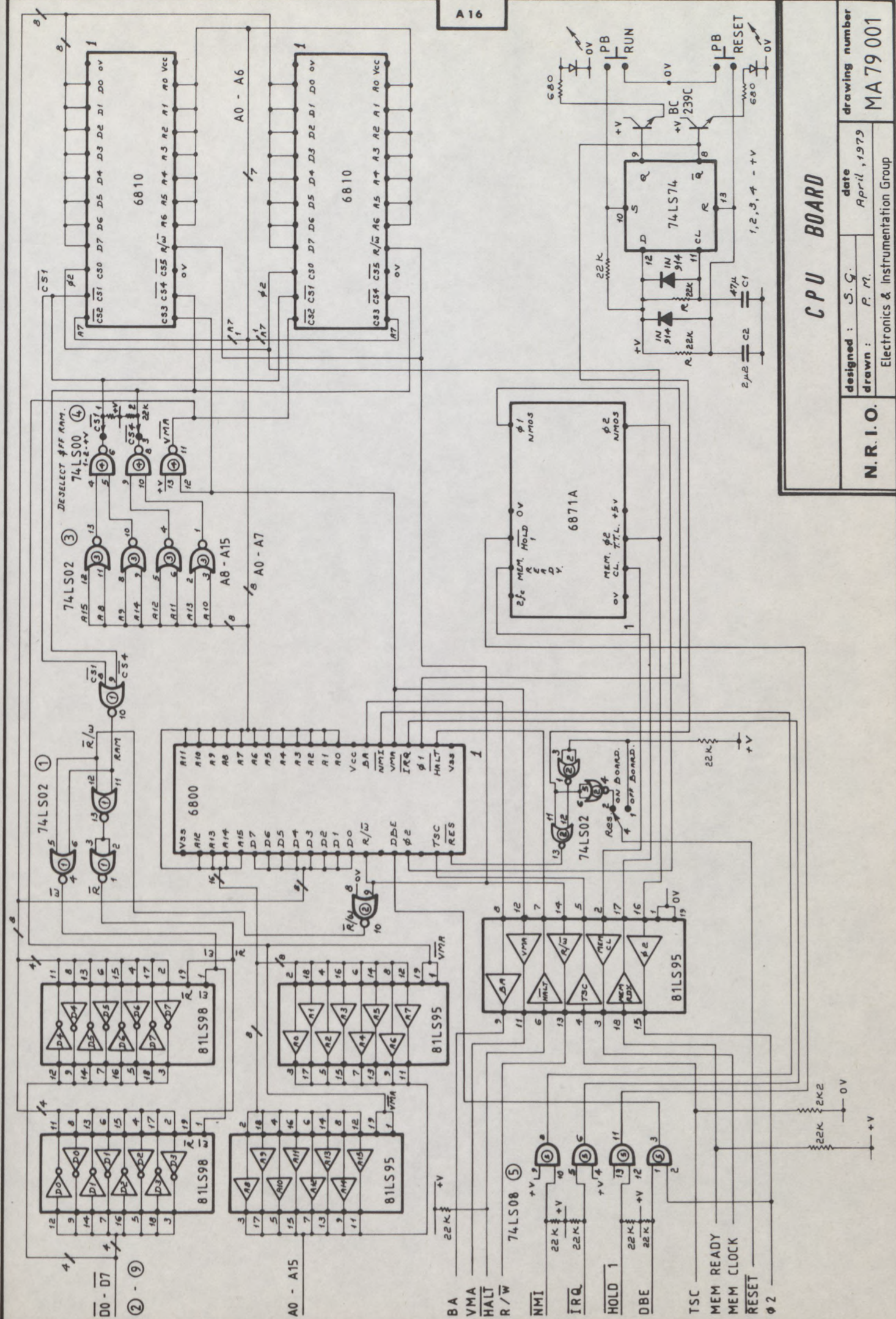
drawing number
HB 81 008

Electronics & Instrumentation Group



MPC WIRING DIAGRAM

NR10	designed : J.K.K.	date April, 1983	drawing number HB 81 009
	drawn : P.L.		
	Electronics & Instrumentation Group		

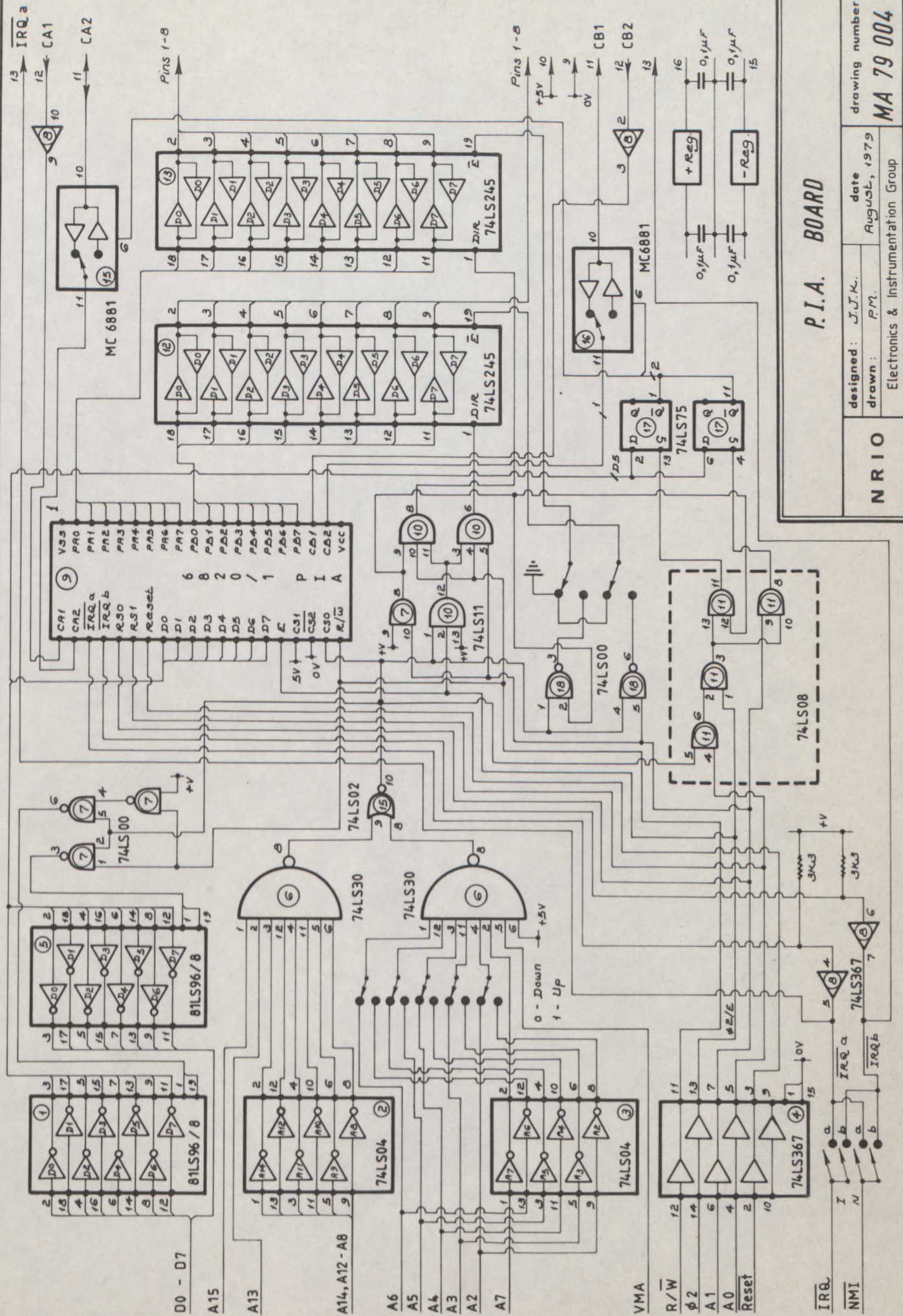


A 16

CPU BOARD

N.R.I.O.	designed :	S. G.	date April , 1979	drawing number MA 79 001
	drawn :	P. M.		
	Electronics & Instrumentation Group			

MA 79 001



P.I.A. BOARD

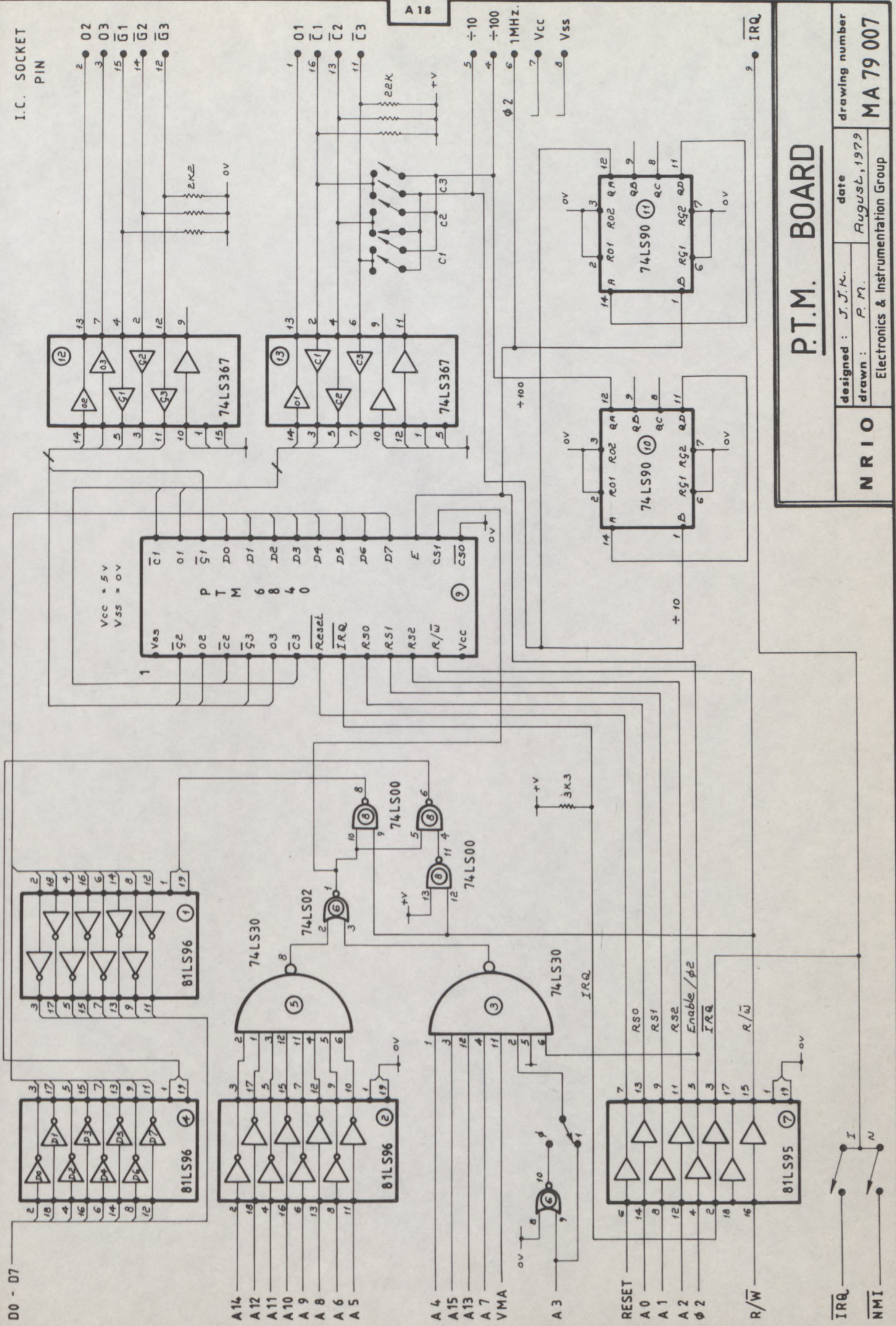
NRIO

designed: J.J.K.
drawn: P.M.
Electronics & Inst

date	station	group
August, 1979		

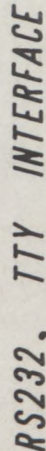
drawing number
MA 79 004

I.C. SOCKET
PIN



P.T.M. BOARD

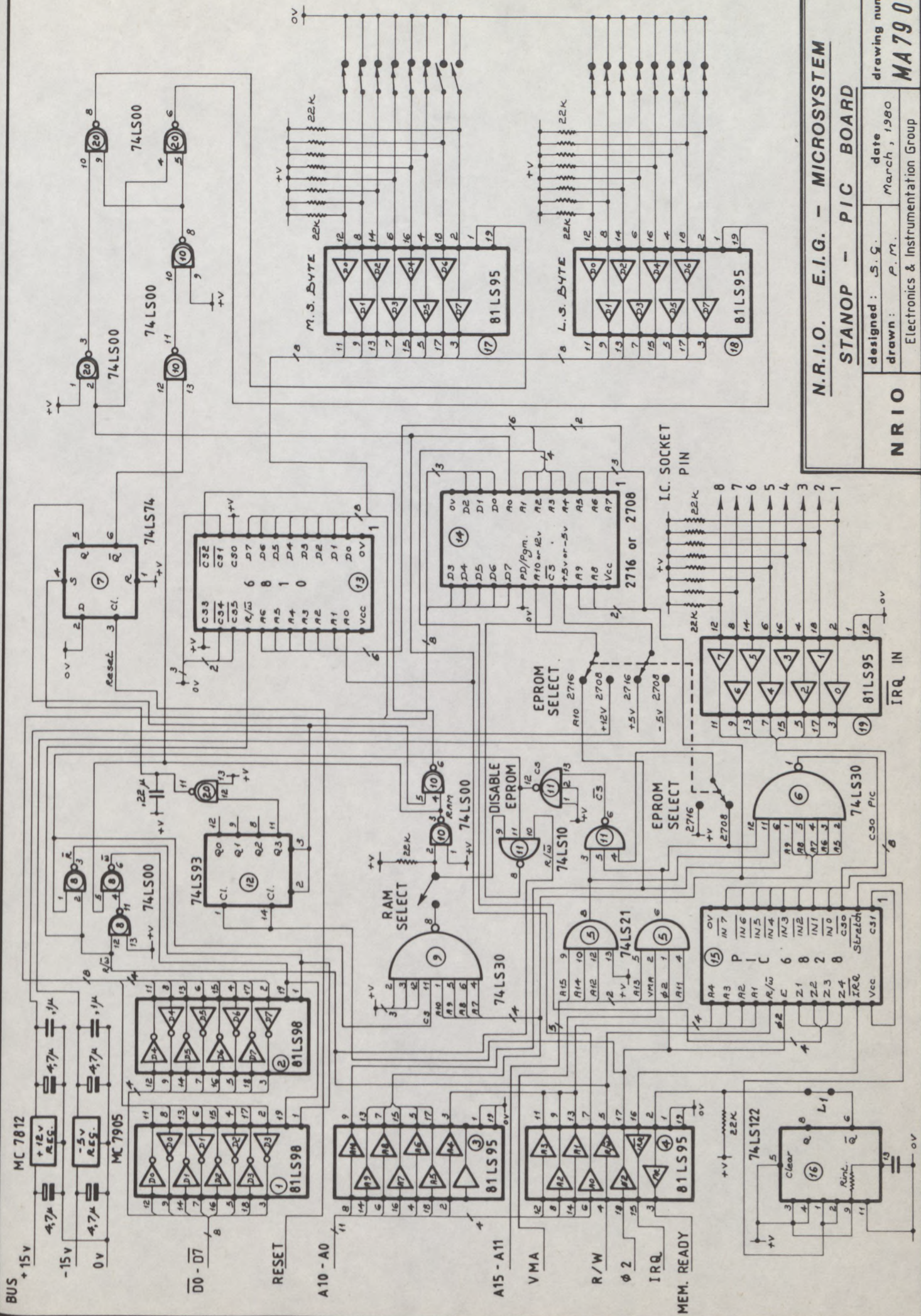
NRIO	designed : J. J. K.	date	drawing number
	drawn : P. M.	August, 1979	MA 79 007
Electronics & Instrumentation Group			



drawing number MA 79 009	date <i>August, 1979</i>	
	designed : <i>S.G.</i>	
	drawn : <i>P.M.</i>	

Electronics & Instrumentation Group

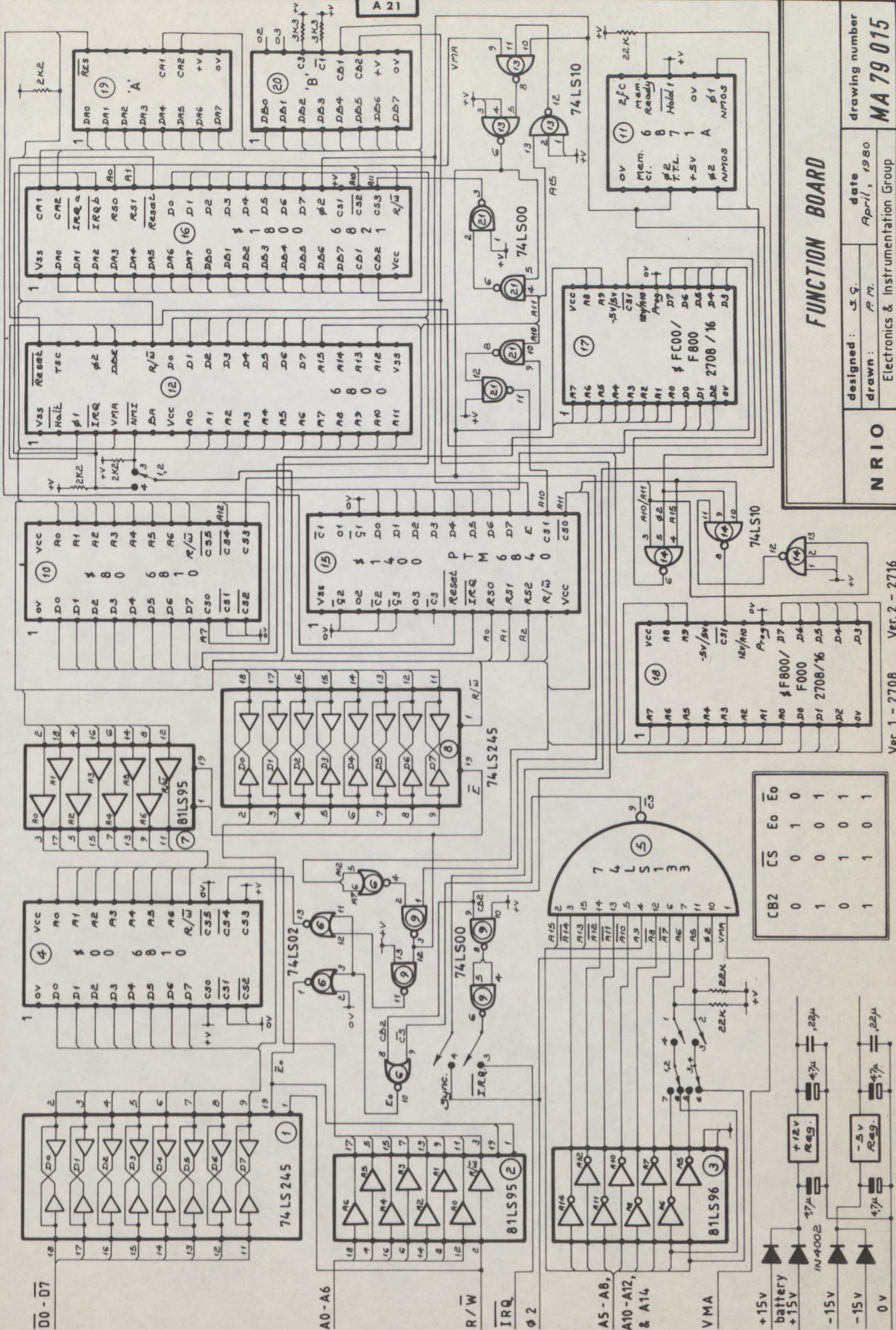
N R I O



N.R.I.O. E.I.G. - MICROSYSTEM

STANOP - PIC BOARD

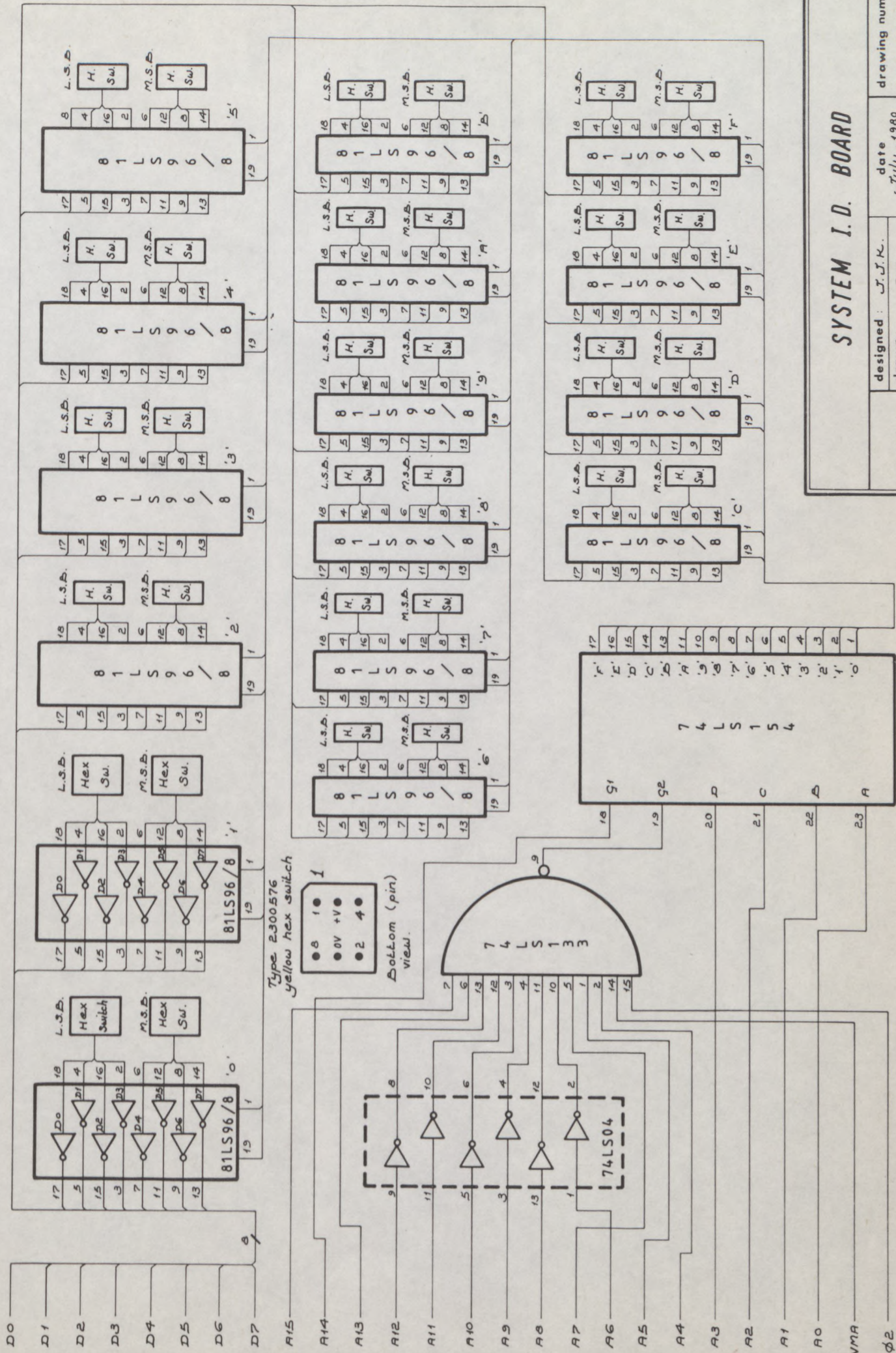
NRIO	designed: S. G.		date March, 1980	drawing number MA 79 013
	drawn: P. M.			
	Electronics & Instrumentation Group			



FUNCTION BOARD

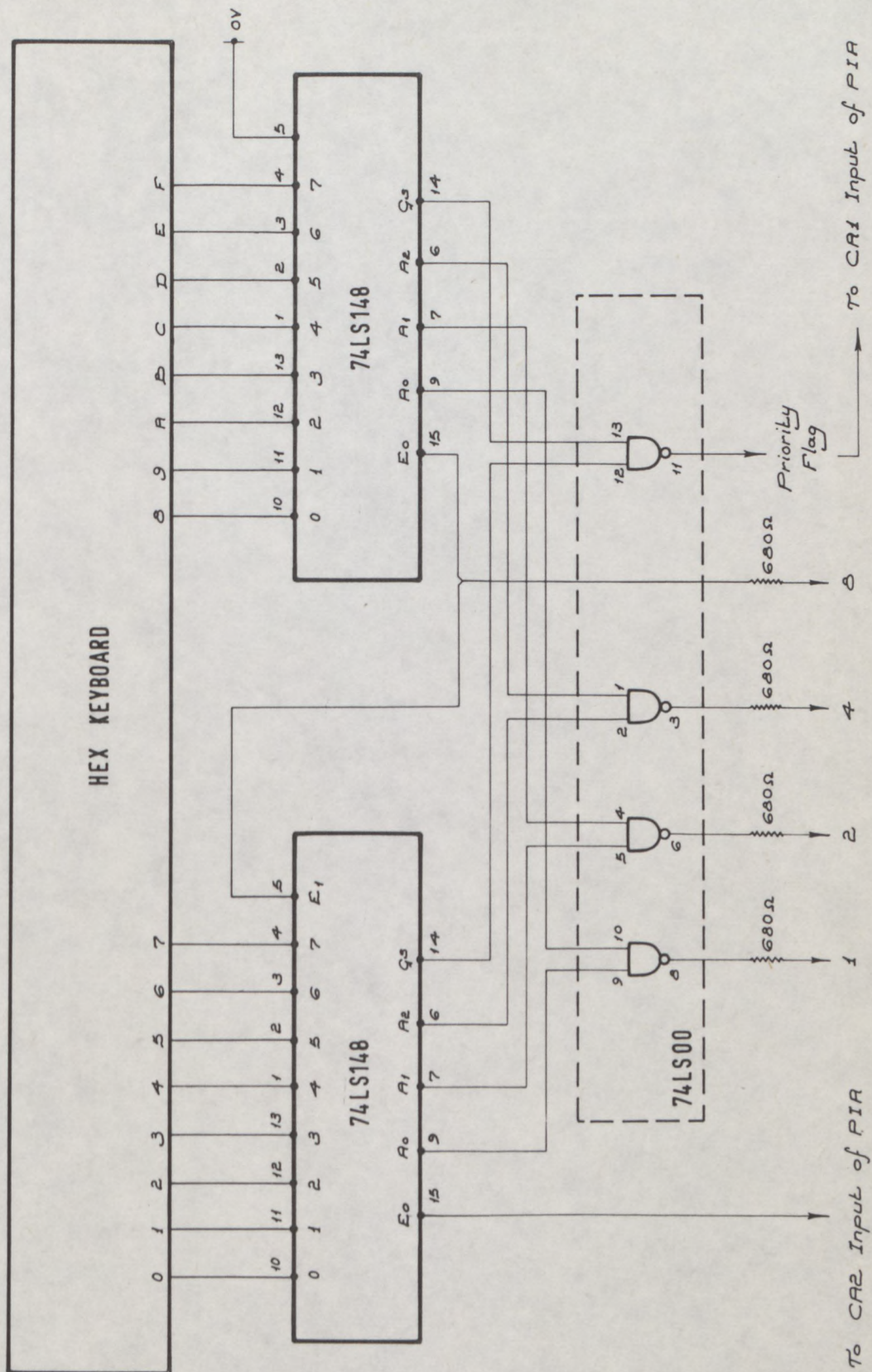
NRIO	designed by	J. S.	date	April, 1980	drawing number	MA 79 015
	drawn by	P. M.			Electronics & Instrumentation Group	

	C82	CS	E0	E0
0	0	0	1	0
1	0	0	0	1
0	1	0	0	1
1	1	1	0	1



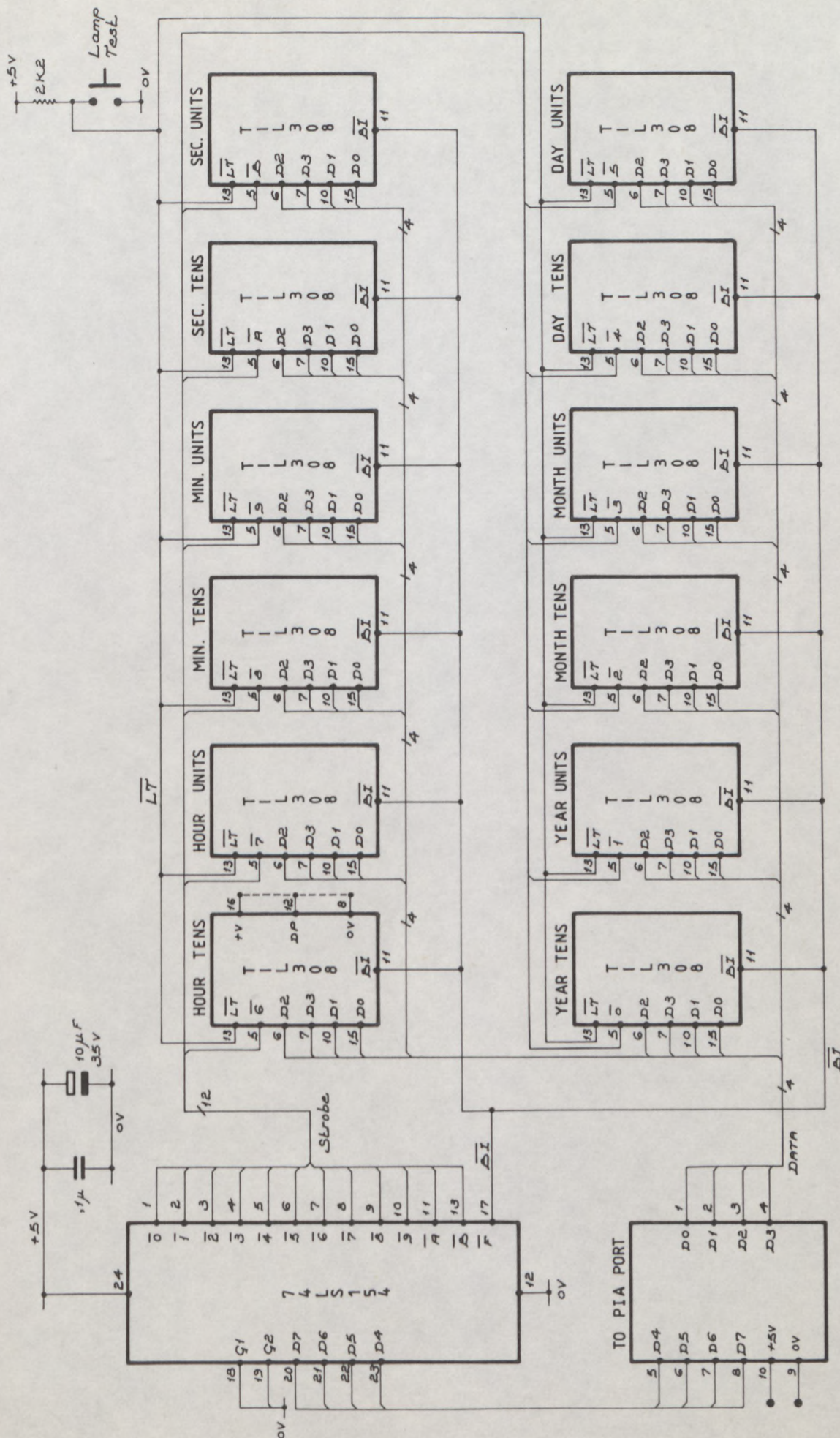
SYSTEM I.D. BOARD

NRIO	designed : J.J.K.	date	drawing number
	drawn : R.M.	July, 1980	MA 79 016
Electronics & Instrumentation Group			



HEXIDECIMAL KEYBOARD MODULE

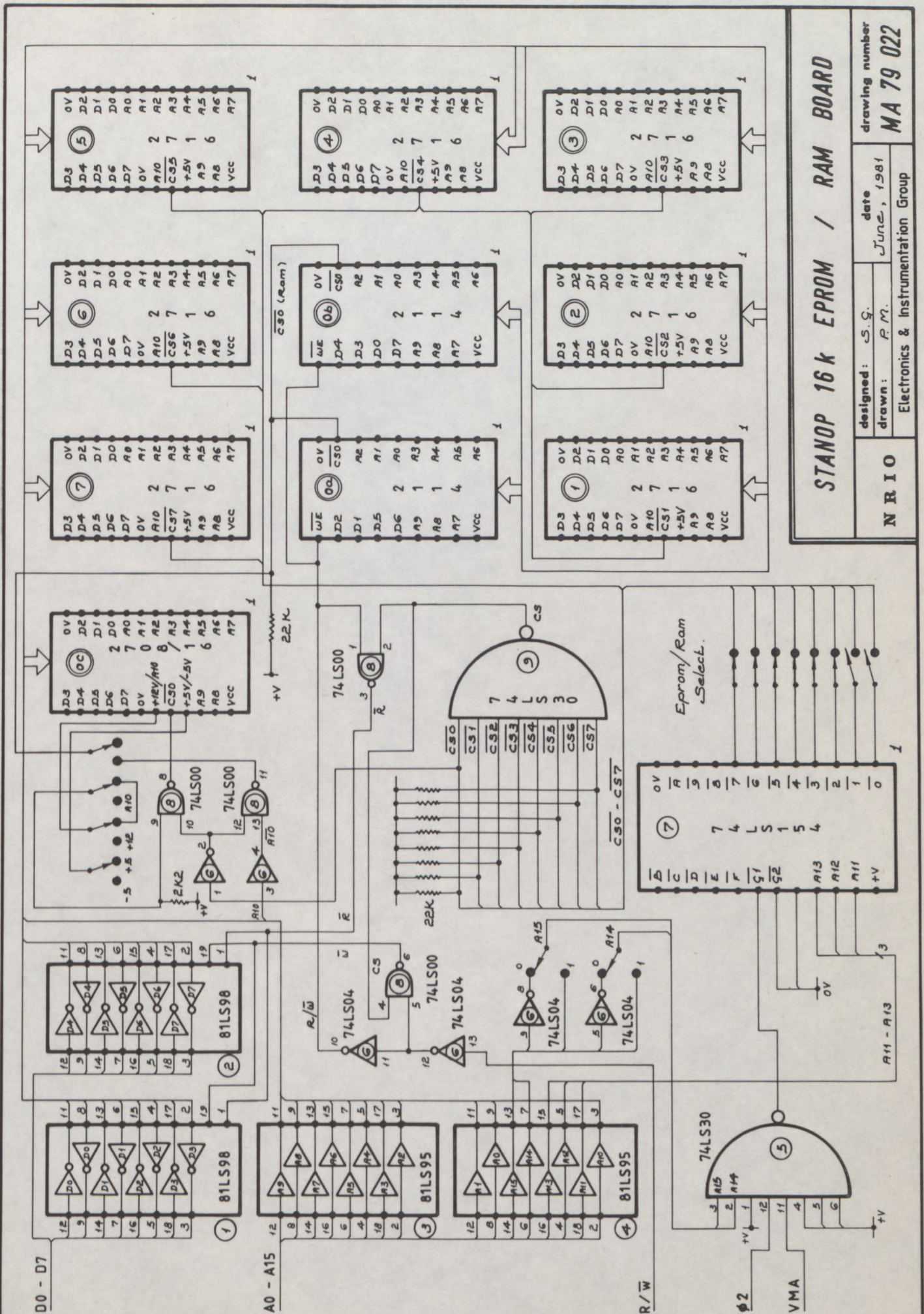
NR10	designed : J.J.K.	date	drawing number
	drawn : P.M.	October, 1980	MA 79 019
Electronics & Instrumentation Group			



Left decimal point on any display.
 Decimal point on - link 12-16
 Decimal point off - link 12-8

12-DIGIT LED DISPLAY MODULE

NRIO	designed: M.J.S.	date	drawing number
	drawn: P.M.	November, 1980	MA 79 020
Electronics & Instrumentation Group			



APPENDIX BTHE NRIO MICROPROCESSOR DEVELOPMENT FACILITIES

The probe calibrator, measurement platform controller and trolley controller software and hardware were designed and tested with the following facilities:

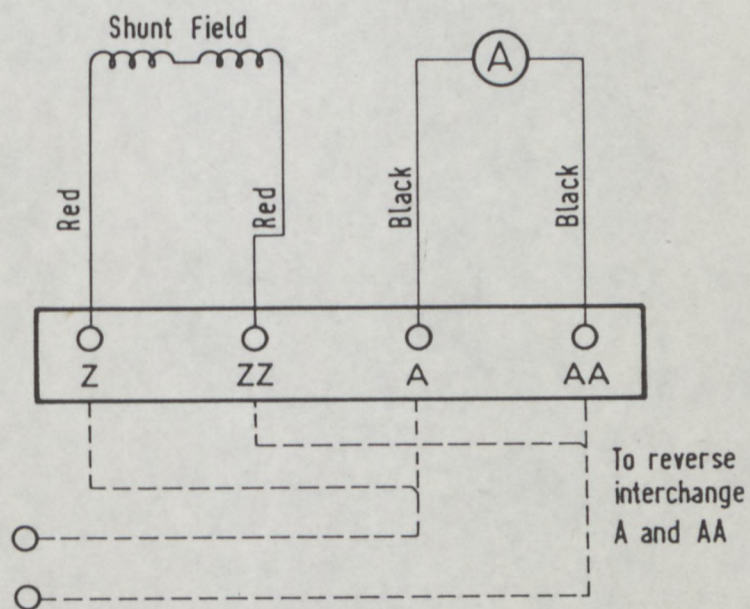
- (a) A Motorola Exorciser equipped with
 - (i) The standard MPU and Exbug card
 - (ii) 40 k of memory (the facility exists to expand it to 56 k)
 - (iii) A 1 k EPROM programmer module modified to also accommodate 2 k EPROMs
 - (iv) A dual floppy disc controller.
- (b) A Motorola Exordisc, dual floppy disc.
- (c) MDOS version 3.00 including
 - (i) MDOS assembler
 - (ii) MDOS editor.
- (d) A Soroc terminal and Axiom IMP4 line printer.
- (e) A Paratronics model 240 logic analyser.
- (f) A model control console consisting of the standard STANOP operating system, keyboard and VDU.

APPENDIX CDATA SHEETS

	<u>Page</u>
DC Motors	C2
Infra-red LED	C3
Light-activated Switch (LAS)	C5
Slotted Opto Switch	C5
Hall Effect IC	C6
Power Module	C7
V to F Converter (9400)	C8

NCCO ELECTRIC MOTORS

		<u>Motor (x)</u>	<u>Motor (y)</u>
Rated output	(KW)	0,37	0,75
Frame size		D 905 AS	905 BL
Type		DC	DC
Base speed	(rpm)	2000	2000
Winding		Shunt	Shunt
Max. Armature Current	(Amp)	3,3	5,6
Armature Voltage	(V)	200	200
Field Voltage	(V)	180	180

SHUNT 4 LEADS OUT

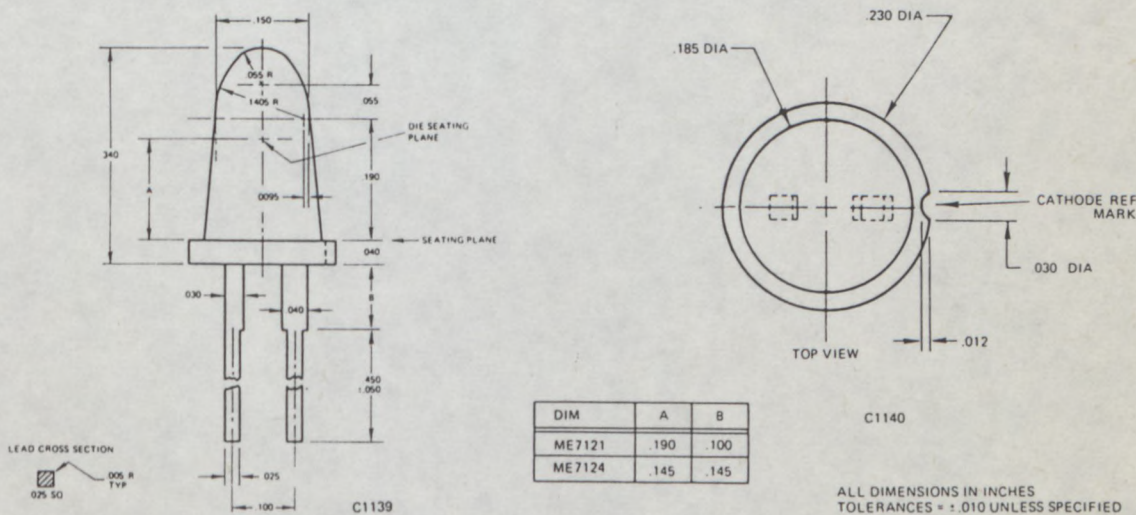
GENERAL INSTRUMENT
Optoelectronics

ME7121
ME7124
HIGH POWER INFRARED EMITTERS

PRODUCT DESCRIPTION

This family of high power liquid phase epitaxial IR Emitters is designed to accommodate all needs of the emitter detector relationship. Products range from a wide angle power spread for non-critical detector location to sharp-angle concentration of power for detectors located a significant distance from the emitter. The devices can be mounted with a plastic pop-in, furnished upon request.

PACKAGE DIMENSIONS



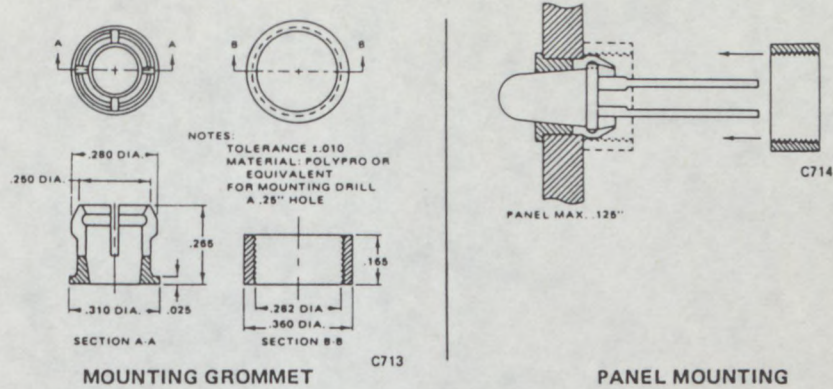
ABSOLUTE MAXIMUM RATINGS

Power dissipation @ 25°C ambient	150 mW
Derate linearly from 50°C	2.8 mW/°C
Storage & operating temperature	-55° to 100°C
Lead solder time @ 230°C (Note 3)	5 sec
Continuous forward current	100 mA
Reverse voltage	3.0 V
Peak forward current (PW = 1.0 μsec, Duty Cycle = 0.3%)	1.0 A

ELECTRO-OPTICAL CHARACTERISTICS

	TYPICAL HALF ANGLE (DEGREES)		TYPICAL ON AXIS INTENSITY (MW/STR.) @ 50 mA		{ into cone @ 1/2 power points @ I _F = 50 mA ROP = 3 mW
	MIN.	TYP.	MAX.	UNITS	
ME7121		17°	10.8		
ME7124		6°	243.6		
Total External Output Power (Note 2)	1.0	3.0		mW	I _F = 50 mA
Peak Emission Wavelength		940		nm	I _F = 50 mA
Spectral Line Half Width		50		nm	I _F = 50 mA
Forward Voltage		1.4	1.8	V	I _F = 50 mA
Light Turn On & Turn Off Time		500		nsec	50 Ω Load
Reverse Current		10		μA	V _R = 3.0 V

PANEL MOUNTING TECHNIQUES



TYPICAL ELECTRO-OPTICAL CHARACTERISTIC CURVES (25°C Free air temperature unless otherwise specified.)

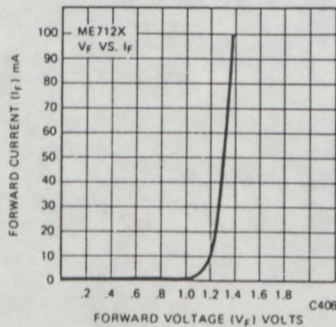
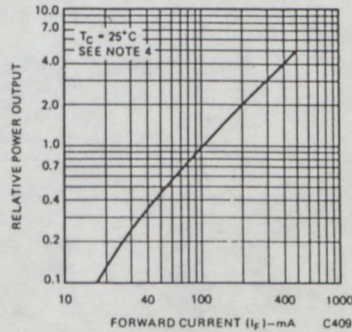
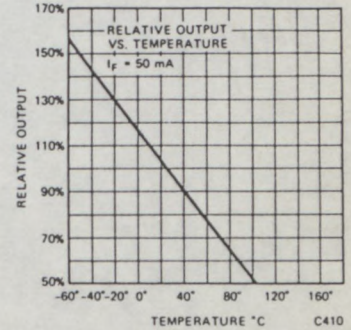
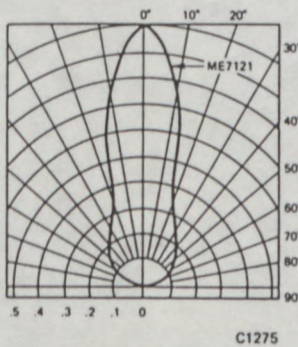
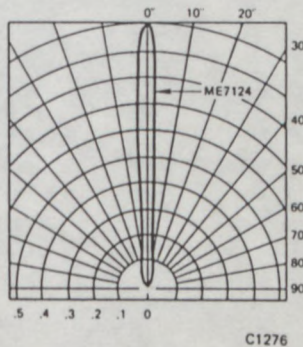
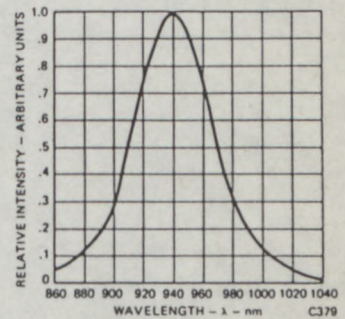
Fig. 1. I_F vs. V_F Fig. 2. ROP vs. I_F PeakFig. 3. ROP vs. Temperature
(Note 1)Fig. 4. Spatial Distribution
(ME7121)Fig. 5. Spatial Distribution
(ME7124)

Fig. 6. Spectral Distribution

NOTES

1. The curves in figure 3 are normalized to the power output at 25°C to indicate the relative efficiency over the operating temperature range.
2. The total external radiated power output measurements are made with a Centralab 110C solar cell terminated into a 100Ω impedance.
3. The leads of the ME7121 and ME7124 were immersed in molten solder, heated to 230°C, to a point 1/16 inch from the body of the device, per MIL-S-750.
4. This parameter is measured using pulse techniques $p_w = 40\text{ }\mu\text{sec}$ duty cycle $\leq 10\%$.

light-activated switches

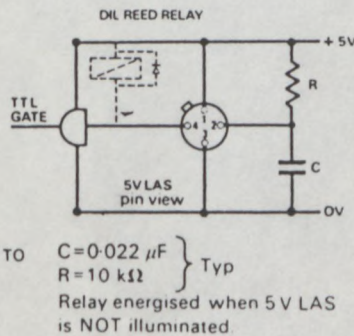
Combined photodiode and monolithic integrated circuit. One external resistor and capacitor sets the light switching threshold level. When the normally incident light is brighter than the threshold level, device switches from logic 0 to 1. TO18-style case with end window. These devices have a response peaking in the infra-red region (850 nm). Suitable infra-red light sources include sunlight and tungsten filament lamps. Visible red L.E.D.s can also be used, provided the separation between source and detector is small.

Data sheet R/2107 November 81 available on request.

5 V



PIN 3
CONNECTED TO
CASE

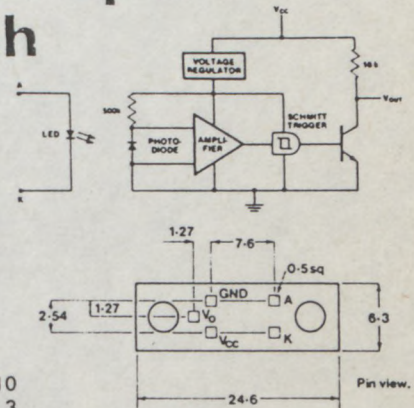


Will drive T.T.L. circuits or the D.I.L. Reed Relay form A direct using a 5 V d.c. supply as shown.

slotted opto-switch with logic



L. 25 W. 6 H. 10
F.C. 19 Dia. 3



A slotted opto-switch comprising an infra-red source and integrated photodetector. The i.c. photodetector consists of a photodiode, amplifier, voltage regulator, schmitt trigger and output stage. An important feature is its wide supply voltage range, +4.5 V to 16 V, which allows the output to interface directly to T.T.L., L.S./T.T.L. and C-MOS. The detector can sink 10 T.T.L. loads when the infra-red beam is interrupted. The output rise and fall times are independent of object speed and the integrated voltage regulator ensures high noise immunity. Applications include position detection, paper sensing, counting, optical encoding and level sensing. Operating temperature range: -40°C to $+100^{\circ}\text{C}$.

diode

V_F max (at $I_F = 20 \text{ mA}$) 1.5 V

V_R max. 3 V

I_F max. (cont.) 50 mA

photodetector

Operating supply voltage range +4.5 to +16 V d.c.

Max. dissipation 250 mW at 25°C .

Output sink current 50 mA max.

Operating current 15 mA max.

coupled

Propagation delay times

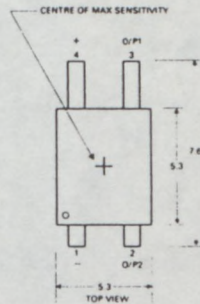
(high to low, or low to high) 5 μs

Output rise time 180 ns max. 150 ns typ.

Output fall time 50 ns max. 23 ns typ.

Max. operating frequency 100 kHz.

linear hall effect i.c.



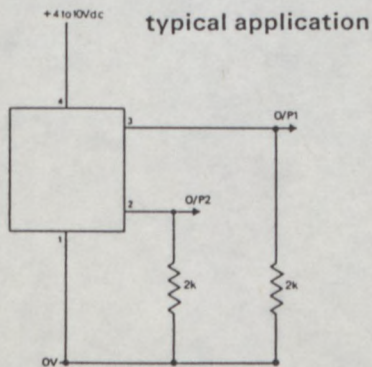
H. 7-5

A miniature linear output Hall effect sensor in a moulded 4-pin plastic D.I.L. package. This device features a differential output stage. One output increases linearly in voltage whilst the other decreases for a linear increase in magnetic flux density over a ± 40 mT range.

Typical applications for this versatile i.c. include magnetic field investigation in the vicinity of transformers and cables, current sensors with high isolation, linear feedback elements in analogue control systems, etc.

The sensor is immune from damage by high values of flux density.

Equivalent to 634SS2.



Supply voltage +4 to +10 V d.c.

Supply current 3.5 mA (typ.)

Output current 10 mA

Output voltage (at zero mag. flux) 1.75 to 2.25 V at 5 V supply

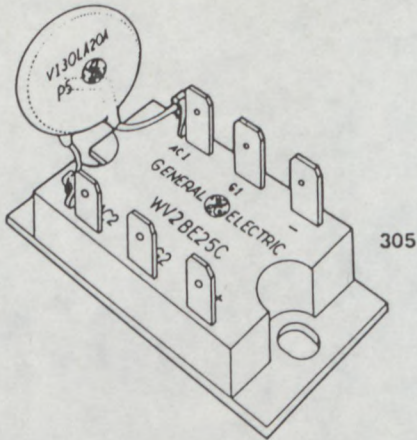
Sensitivity 7.5 to 10.6 mV/mT

Frequency of operation 0 to 100 kHz

Operating temperature range 0 °C to +70 °C



POWER MODULES



New General Electric power modules are miniaturized, self-contained, Epoxy encapsulated modules capable of performing basic AC to DC conversion functions. Typical applications include — DC power supplies, DC motor controls, battery chargers, magnetic clutches and brakes.

All General Electric power modules incorporate Power Glas™ passivated semiconductors with the latest pellet mountdown and interconnect techniques, thereby assuring the utmost in reliability.

COMMON CHARACTERISTICS @ 25°C

Isolation Breakdown	2,500 V _{PEAK}
Surge, Peak One Cycle	300 A
Fusing, I ² t @ 8.3 msec	370 A ² SEC
Gate Current to Trigger (Max.)	40 mA
Gate Voltage to Trigger (Max.)	2.5 V
On-State Current Rate of Rise (di/dt).	100 A/μSEC
Off-State Voltage Rate of Rise (dv/dt)	20 V/μSEC
Operating Temperature	-40 to 125°C

			GE TYPES			
BASIC CIRCUIT SCHEMATIC	I _O AVERAGE @ 85° (A)	V _{IN} (V)	BASIC CIRCUIT	WITHOUT FREE WHEELING DIODE	WITHOUT GE-MOV® VARISTOR	WITHOUT EITHER DIODE OR VARISTOR
	25	120	WV2BE25C	WV2BC25C	W2BE25C	W2BC25C
		240	WV2BE25E	WV2BC25E	W2BE25E	W2BC25E
	25	120	WV2BJ25C	WV2BK25C	W2BJ25C	W2BK25C
		240	WV2BJ25E	WV2BK25E	W2BJ25E	W2BK25E
	25	120	WV2BA25C	—	W2BA25C	—
		240	WV2BA25E	—	W2BA25E	—

VOLTAGE-TO-FREQUENCY CONVERTER

DESCRIPTION

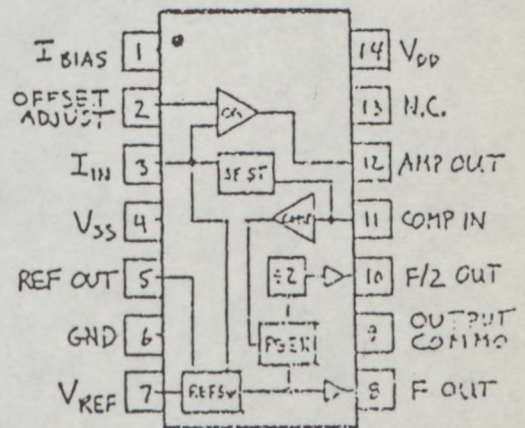
The Teledyne 9400 is a low cost voltage-to-frequency converter combining Bipolar and CMOS technology on a single chip. The converter accepts a variable analog input signal and generates an output pulse train whose frequency is linearly proportional to the input voltage. A complete V to F system requires addition of only 2 capacitors, 3 resistors, and 2 supply voltages.

FEATURES

- . 10Hz to 100kHz operation
- . $\pm 0.01\%$ typical Linearity to 10kHz
- . $\pm 25\text{PPM}/^\circ\text{C}$ typ gain temperature stability
- . Open collector output
- . Output can drive 5TTL loads as well as CMOS
- . Pulse and square wave outputs
- . Programmable scale factor
- . Low power dissipation-27mW typ

APPLICATIONS

- . Precision V/F Converters
- . Precision F/V Converters
- . 13 bit A/D Converters
- . μP data acquisition
- . Ultra long time interval integrator
- . Digital scales
- . Thermostats
- . Digital panel meters
- . Phase locked loops
- . Remote control
- . FSK data transmission
- . Analog data transmission & recording
- . VCO
- . Communications scrambler
- . Sound in Video Games



J Package

14-Pin Plastic DIL

ABSOLUTE MAXIMUM RATINGS

V_{DD} to V_{SS}	18Volts
I_{IN}	$\pm 10\text{ma}$
I_{REF}	$\pm 10\text{ma}$
$V_{O\text{ max}} - V_{O\text{ COM}}$	18Volts
$ V_{REF} - V_{SS} $	1.5Volts
Operating temp.	0°C to 70°C

UD = 0V, $V_{SS} = -5V$, $V_{REF} = -0.2V$, $R_{BIAS} = 100k\Omega$, $T_A = 25^\circ C$

CONVERSION ACCURACY

Linearity (10kHz) $\pm 0.01\%$ $(100\text{kHz}) \pm 0.1\%$

Full Scale Temperature Stability ± 25 PPM/ $^{\circ}$ C

INPUT CIRCUIT

1. 1000

$$V_{off} \text{ (offset)} \leq \pm 10 \text{ mV}$$
$$\nu_{\text{iso}} \text{ (drift)} < \pm 5 \text{ PPM}/^{\circ}\text{C}$$

SUPPLY REQUIREMENTS

l_{pp} $2ma$

$$I_{SS} = -1.5 \text{ mA}$$

OUTPUTS

 $V_{OL} \quad 0.4V$

CONDITIONS

$V_{in} = 0$ to 10 Volts

$V_s = 0$ to 10 Volts

$$0^{\circ}\text{C} < T_2 < 70^{\circ}\text{C}$$
 $V_{cc} = 10V, R_{cc} = 1M$ $0^{\circ}\text{C} < T_1 < 70^{\circ}\text{C}$
$$0^{\circ}\text{C} < T_A < 70^{\circ}\text{C}$$

EQUATIONS

$$f_{OUT} = \frac{V_{in}}{(R_{in})} \times \frac{1}{(V_{REF})(C_{REF})}$$

$$R_{in} = \frac{V_{in} \text{ (MAX)}}{10\mu A}$$

$$82K \leq R_{BIAS} \leq 120K$$

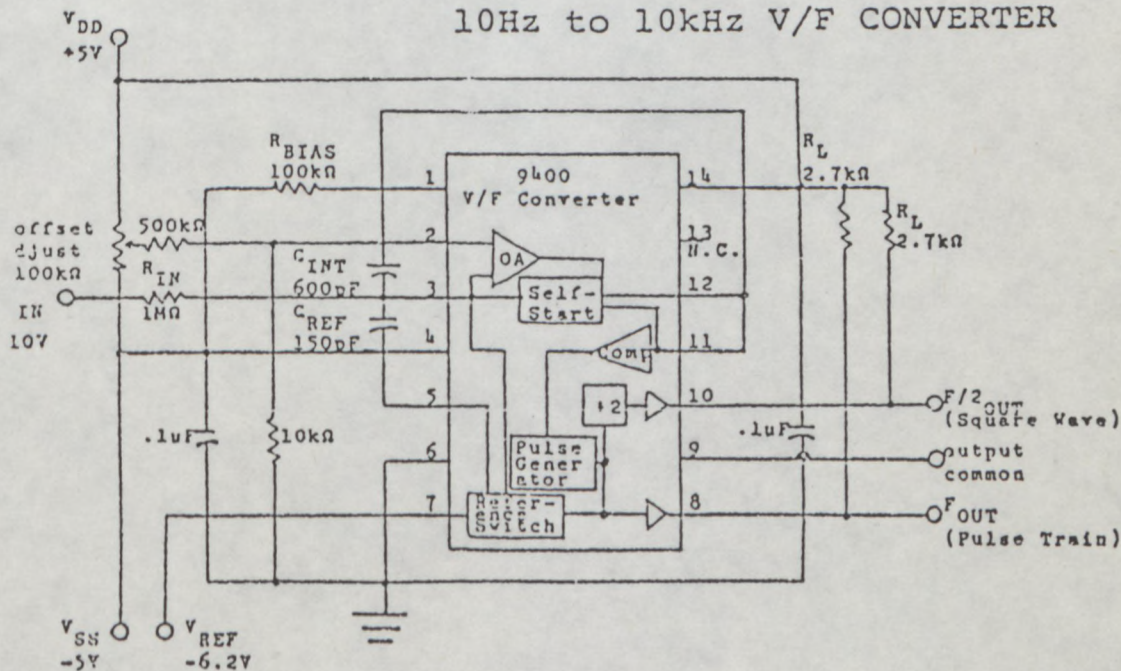
$$3C_{REF} < C_{INT} \leq 5C_{REF}$$

For optimum stability:

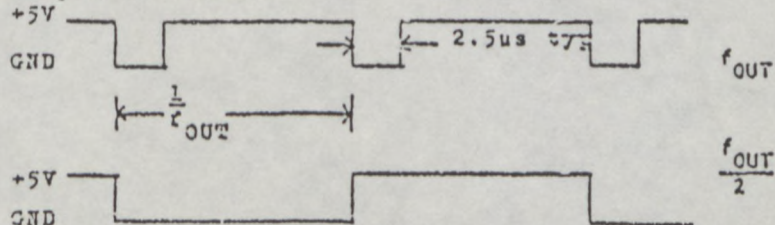
$$C_{INT} \approx 4 \times C_{REF}$$

TYPICAL APPLICATION

10Hz to 10kHz V/F CONVERTER

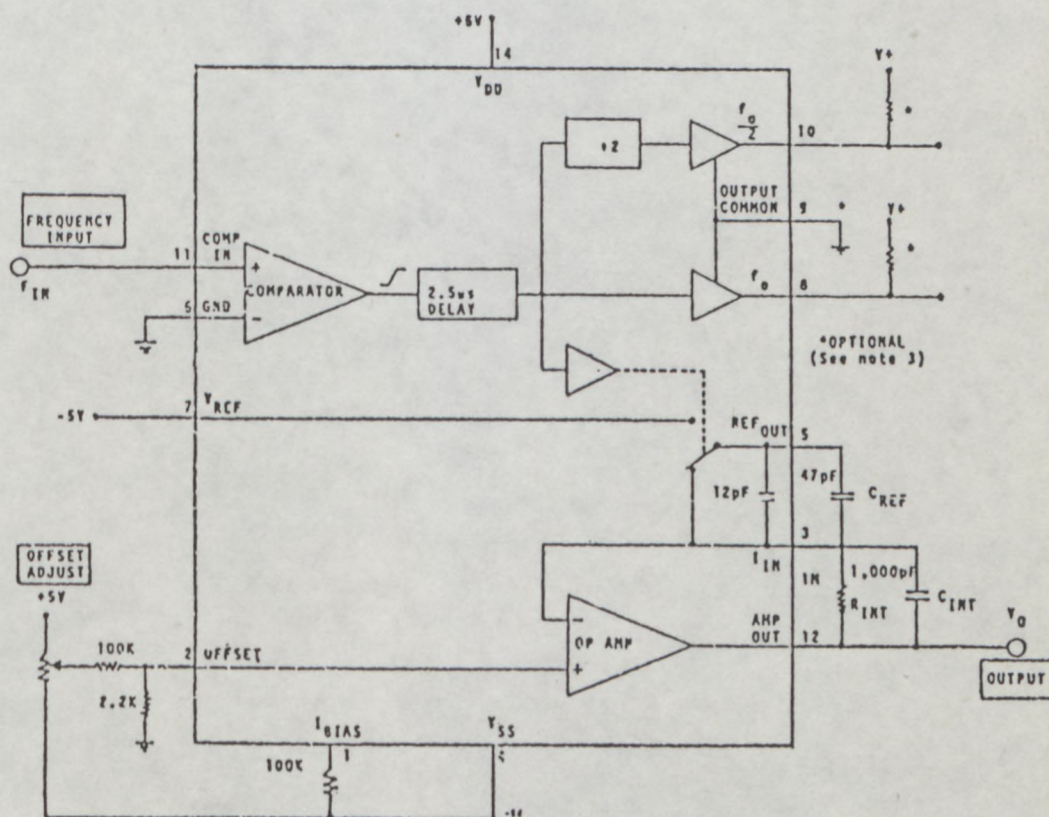


1. To adjust $f_{\min}$, set $V_{in} = 10\text{mV}$ and adjust the 100K offset for 10Hz out.
2. To adjust $f_{\max}$ set $V_{in} = 10\text{V}$ and adjust R_{in} or V_{REF} for 10kHz out.
3. Output waveforms:



4. To increase $f_{OUT\ MAX}$ to 100kHz change C_{REF} to 20pF and C_{INT} to 80pF.
5. For high performance applications use high stability components for R_{in} , C_{REF} , and V_{REF} . (metal film resistors and glass film capacitors) Also separate the output ground (Pin 9) from the input ground (Pin 6).

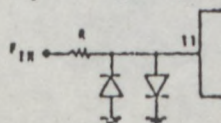
ACCURACY Better than 0.1% FS



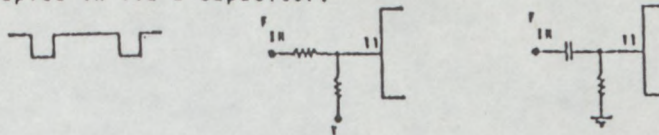
NOTES

1. The input signal must cross through zero in order to trip the comparator. In order to overcome the hysteresis the amplitude must be greater than +200mV. If the comparator input voltage exceeds -2.5V then the Op Amp output will go to its maximum positive output voltage for the duration of the overvoltage.

If the input voltage has a wide amplitude variation then a pair of back to back diodes may be used to limit the voltage to $\pm 0.7V$.



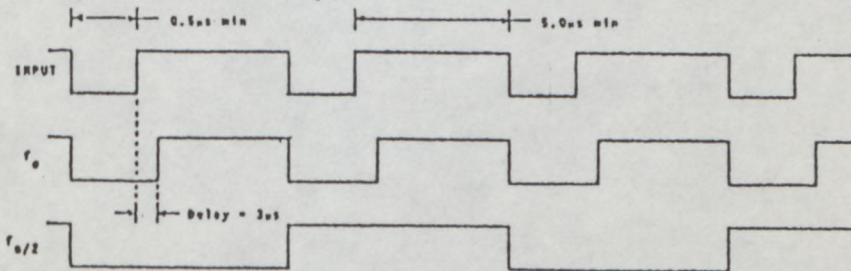
If only a unipolar input signal (f_{IN}) is available it is recommended that either an offset circuit using R_{INT} resistor be used or that the signal be coupled in via a capacitor.



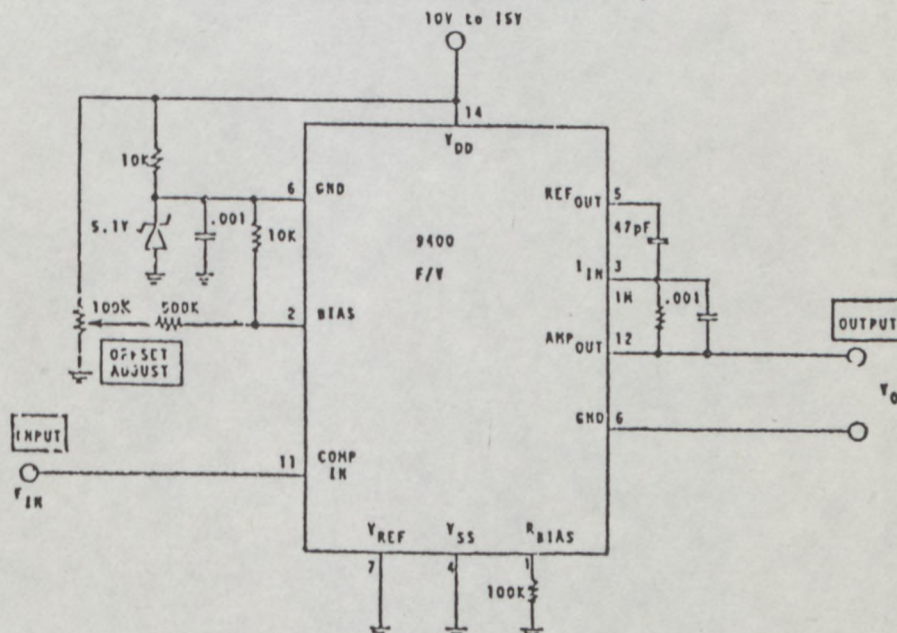
The output voltage of the Op Amp is referenced to Pin 6 (GND). So if Pin 6 is used to determine the comparator threshold the Op Amp output reference will also be shifted.

- For 100KHz maximum input R_{INT} should be decreased to 100K Ω .
- f_o and $f_o/2$ are not used in the F/V mode. However, these outputs may be useful for some applications, such as a buffer to feed additional circuitry. f_o will then follow the input frequency waveform; except that f_o will go high 3 μ s after F_{IN} goes high. $f_o/2$ will be square wave with a frequency of one half f_o .

If these outputs are not used then Pins 8, 9, and 10 may be left floating or connected to ground.



SINGLE SUPPLY F/V



NOTE:

- The input is now referenced to 5.1V (Pin 6). The input signal must therefore be restricted to be greater than 3 volts (Pin 6 -2V) and less than 10 to 15V (V_{DD}).

If the signal is AC coupled then a resistor (100K to 10M Ω) must be placed between the input (Pin 11) and Pin 6.

- The output will now be referenced to Pin 6, which is at 5.1V (V_7). For frequency meter applications a 1mA meter with a series scaling resistor can be placed across Pins 6 and 12.

APPENDIX DPROBE CALIBRATOR PROGRAM LISTING

A trimmed version of the assembler listing of the modified probe calibrator program (interfaced to the MHDAS) is given in this appendix.


```

00001
00002
00003
00004
00005
00006      A000      PIAD1      EQU      $A000      4msb--PUSHB. 4lsb--THUMB.S/W
00007      A001      PIAC1      EQU      $A001
00008      A002      PIAD2      EQU      $A002      4msb--INPUT CONT/LEDS 4lsb--M
00009      A003      PIAC2      EQU      $A003
00010
00011      A091      CR2        EQU      $A091      WRITE CONTROL REGISTER #2 (PT
00012      A090      CR13       EQU      $A090      WRITE CONTROL REGISTER #1,#3
00013      A092      MSB1       EQU      $A092      WRITE MSB BUFFER REGISTER
00014
00015      A0A0      MODEMC      EQU      $A0A0      ACIA CONTROL REG
00016      A0A1      MODEMD      EQU      $A0A1      ACIA DATA REG
00017
00018      * STANOP EQUATES
00019      * ROUTINES IN STANOP USED BY PROBE CALIBRATOR
00020
00021      D961      BINBCD      EQU      $D961
00022      D996      VDUASC      EQU      $D996
00023      C52F      VDURET      EQU      $C52F
00024      C53C      VDUOTA      EQU      $C53C
00025      C545      VDUOUT      EQU      $C545
00026      EA7C      BUSY       EQU      $EA7C
00027      EA74      DONE        EQU      $EA74
00028      C0A1      COMBUF      EQU      $C0A1
00029      DA42      INDIG4      EQU      $DA42
00030      C072      FF          EQU      $C072
00031      C06C      XX          EQU      $C06C
00032      C06E      U           EQU      $C06E
00033      C5F9      KBDCHK      EQU      $C5F9
00034      D551      WRONG      EQU      $D551
00035
00036      * SYSTEM RAM POSITIONS(SCRATCH PAD)
00037
00038      0000      00C8      TABLE      RMB      200
00039      00C8      0005      OP          RMB      5      2PIEF...CONSTANT
00040      00CD      0005      OPRAND      RMB      5
00041      00D2      0001      XREG        RMB      1
00042      00D3      0002      XXREG       RMB      2
00043      00D5      0002      CCOUNT      RMB      2
00044      00D7      0001      TEMP        RMB      1      STORE HEIGHT
00045      00D8      0001      TEMP1       RMB      1      STORE PERIOD
00046      00D9      0001      TEMP2       RMB      1
00047      00DA      0001      TEMP3       RMB      1
00048      00DB      0001      TEMP4       RMB      1
00049      00DC      0001      TEMP5       RMB      1
00050      00DD      0001      TEMP6       RMB      1
00051      00DE      0001      TEMP7       RMB      1
00052      00DF      0001      TEMP8       RMB      1
00053      00E0      0013      MATH        RMB      19      RESERVED FOR MATH SUBROUTINES
00054      00F3      0002      STEP1       RMB      2
00055      00F5      0001      DIR         RMB      1
00056      00F6      0001      FLA         RMB      1      INT/EXT FLAG
00057      00F7      0002      HE          RMB      2      REMEMBER HEIGHT TO VDU
00058      00F9      0002      PE          RMB      2      REMEMBER PERIOD TO VDU

```



```

00059 00FB 0002    STACK   RMB      2
00060              *
00061              * MATH PACK RAM POSITIONS
00062              *
00063          00E7    FLAG    EQU     $E7
00064          00E8    T1      EQU     $E8
00065          00EA    FLAG1   EQU     $EA
00066          00EB    ANS1    EQU     $EB
00067          00EF    ANS2    EQU     $EF
00068              *
00069 B000          ORG      $B000
00070              *
00071              * CONSTANTS.....EPROM
00072              *
00073 B000 C9        CONS    FCB      $C9,$0F,$DA,$03,$00    2PIE
00074 B005 F4        CON     FCB      $F4,$01,$FA,$00,$7D    SUBTRACTION TABLE
00075 B00A 3F                FCB      $3F,$1F,$10,$08,$04    (500,250,125,63,E
00076 B00F 80        SRATE    FCB      $80,$00,$00,$F5,$00    SAMPLE RATE
00077 B014 0A        STEP     FCB      $0A,$08,$09,$01    MOTOR STEPS
00078 B018 05                FCB      $05,$04,$06
00079 B01B 02        STEPA    FCB      $02                STEP 8
00080 B01C 2F        UPDOWN   FCB      $2F,$07    UP/DOWN STEP RATE
00081              *
00082 B020          ORG      $B020
00083              *
00084 B020 7E B168          JMP      COMINT    COMMAND INTERPRETER
00085 B023 7E B030          JMP      SETUP     CAL. SETUP ROUTINE
00086 B026 CE BDF8          LDX      #COM      OUTPUT PROMPT
00087 B029 BD C545          JSR      VDUOUT
00088 B02C 39                RTS
00089 B02D 7E B05E          JMP      ENT       AUTO COMMANDS(STAND ALONE)
00090              *
00091              * PROG STARTUP ROUTINE
00092              *
00093 B030 86 00        SETUP    LDA  A    #$00          *LOAD CONTROL REG. SO
00094 B032 B7 A001          STA  A    PIAC1          *THAT DDR CAN BE ADDRESSED
00095 B035 B7 A003          STA  A    PIAC2
00096 B038 86 FF          LDA  A    #$FF          *SET ALL EIGHT BITS TO
00097 B03A B7 A002          STA  A    PIAD2          *ACT AS OUTPUT PORTS
00098 B03D 86 04          LDA  A    #$04          *SELECT OUTPUT REG.
00099 B03F B7 A003          STA  A    PIAC2
00100 B042 86 00          LDA  A    #$00          *SET UP TO ACT AS
00101 B044 B7 A000          STA  A    PIAD1          *INPUT PORTS
00102 B047 86 05          LDA  A    #$05
00103 B049 B7 A001          STA  A    PIAC1
00104 B04C 86 F2          LDA  A    #$F2
00105 B04E B7 A002          STA  A    PIAD2          *LOAD STOP
00106 B051 0F                SEI
00107 B052 CE B01B          LDX      #STEPSA
00108 B055 DF F3            STX      STEP1
00109 B057 7F 00F6          CLR      FLA          INT/EXT FLAG($FF=INT;$00=EXT)
00110 B05A 7F 00D7          CLR      TEMP
00111 B05D 39                RTS
00112              *
00113              *
00114              *ENTER LOOP FROM FRONT PANEL PUSHBUTTONS
00115              *
00116              *

```


00117	B05E	9F	FB	ENT	STS	STACK	
00118	B060	7D	00D7		TST	TEMP	
00119	B063	27	03		BEQ	ENT1	IF VALUES HAS BEEN ENTERED
00120	B065	7E	B4E4		JMP	CONTR0	JUMP TO CONTROL LOOP
00121	B068	86	00	ENT1	LDA A	#\$00	ENTER LED
00122	B06A	DE	F3		LDX	STEP1	ADD TO PREVENT JERK
00123	B06C	E6	00		LDA B	0,X	
00124	B06E	1B			ABA		
00125	B06F	B7	A002		STA A	PIAD2	*DISPLAY ENTER LIGHT
00126	B072	86	FF		LDA A	#\$FF	
00127	B074	97	F6		STA A	FLA	SET INT/EXT FLAG
00128	B076	B6	A000		LDA A	PIAD1	
00129	B079	BD	B642		JSR	ASL	
00130	B07C	81	07		CMP A	#\$07	UP
00131	B07E	26	03		BNE	TS8	
00132	B080	BD	B584		JSR	UP	
00133	B083	81	0E	TS8	CMP A	#\$0E	DOWN
00134	B085	26	03		BNE	TS7	
00135	B087	BD	B5C6		JSR	DOWN	
00136	B08A	81	0F	TS7	CMP A	#\$0F	ENTER
00137	B08C	27	04		BEQ	IN	
00138	B08E	7F	00F6		CLR	FLA	CLEAR INT/EXT FLAG
00139	B091	39			RTS		
00140	B092	BD	B5D0	IN	JSR	INPUT	6DIGITS FROM THUMBWHEEL S/W'S
00141				*			
00142	B095	96	D9		LDA A	TEMP2	HEIGHT TEST/CONVERT
00143	B097	81	01		CMP A	#01	WITHIN LIMITS ?
00144	B099	2F	03		BLE	OK.HUN	
00145	B09B	BD	B6A0		JSR	ERROR	
00146	B09E	81	00	OK.HUN	CMP A	#0	
00147	B0A0	27	0F		BEQ	ST.HT	
00148	B0A2	96	DA		LDA A	TEMP3	
00149	B0A4	81	00		CMP A	#0	
00150	B0A6	2E	06		BGT	ERR	
00151	B0A8	96	DB		LDA A	TEMP4	
00152	B0AA	81	00		CMP A	#\$00	
00153	B0AC	27	03		BEQ	ST.HT	STORE HEIGHT
00154	B0AE	BD	B6A0	ERR	JSR	ERROR	
00155	B0B1	7F	00D7	ST.HT	CLR	TEMP	*CONVERT HEIGHT TO HEX
00156	B0B4	96	DB		LDA A	TEMP4	
00157	B0B6	97	D7		STA A	TEMP	
00158	B0B8	96	D9		LDA A	TEMP2	
00159	B0BA	81	01		CMP A	#1	
00160	B0BC	26	06		BNE	TENA	
00161	B0BE	86	64		LDA A	#100	
00162	B0C0	9B	D7		ADD A	TEMP	ADD HUNDREDS
00163	B0C2	97	D7		STA A	TEMP	
00164	B0C4	96	DA	TENA	LDA A	TEMP3	
00165	B0C6	81	00		CMP A	#0	
00166	B0C8	27	0C		BEQ	SHE	
00167	B0CA	4A		TENB	DEC A		
00168	B0CB	5F			CLR B		
00169	B0CC	C6	0A		LDA B	#10	
00170	B0CE	DB	D7		ADD B	TEMP	ADD TENS
00171	B0D0	D7	D7		STA B	TEMP	
00172	B0D2	81	00		CMP A	#0	
00173	B0D4	2E	F4		BGT	TENB	
00174	B0D6	D6	D7	SHE	LDA B	TEMP	REMEMBER HEIGHT FOR

00175	B0D8	D7	F8		STA B	HE+1	VDU DISPLAY
00176				*		HEIGHT STORED	IN TEMP.....D7
00177				*			
00178	B0DA	96	DC	NULL	LDA A	TEMP5	PERIOD TEST/CONVERT
00179	B0DC	81	02		CMP A	#02	WITHIN LIMITS ?
00180	B0DE	2E	0E		BGT	FAULT	
00181	B0E0	2D	0F		BLT	OK.PH	
00182	B0E2	96	DD		LDA A	TEMP6	
00183	B0E4	81	00		CMP A	#00	
00184	B0E6	2E	06		BGT	FAULT	
00185	B0E8	96	DE		LDA A	TEMP7	
00186	B0EA	81	00		CMP A	#00	
00187	B0EC	27	0F		BEQ	ST.PER	
00188	B0EE	BD	B6A0	FAULT	JSR	ERROR	
00189	B0F1	96	DC	OK.PH	LDA A	TEMP5	
00190	B0F3	81	01		CMP A	#01	
00191	B0F5	27	06		BEQ	ST.PER	
00192	B0F7	96	DD		LDA A	TEMP6	
00193	B0F9	81	07		CMP A	#07	
00194	B0FB	2D	F1		BLT	FAULT	
00195	B0FD	7F	00D8	ST.PER	CLR	TEMP1	CONVERT PERIOD TO HEX
00196	B100	96	DE		LDA A	TEMP7	
00197	B102	97	D8		STA A	TEMP1	
00198	B104	96	DC		LDA A	TEMP5	
00199	B106	81	02		CMP A	#2	
00200	B108	27	08		BEQ	HUND2	
00201	B10A	81	01		CMP A	#1	
00202	B10C	27	0C		BEQ	HUND1	
00203	B10E	81	00		CMP A	#0	
00204	B110	27	0E		BEQ	TENC	
00205	B112	86	C8	HUND2	LDA A	#200	
00206	B114	9B	D8		ADD A	TEMP1	
00207	B116	97	D8		STA A	TEMP1	
00208	B118	20	06		BRA	TENC	
00209	B11A	86	64	HUND1	LDA A	#100	
00210	B11C	9B	D8		ADD A	TEMP1	ADD HUNDRED
00211	B11E	97	D8		STA A	TEMP1	
00212	B120	96	DD	TENC	LDA A	TEMP6	
00213	B122	81	00		CMP A	#0	
00214	B124	27	0C		BEQ	NUL1	
00215	B126	4A		TENN	DEC A		
00216	B127	5F			CLR B		
00217	B128	C6	0A		LDA B	#10	
00218	B12A	DB	D8		ADD B	TEMP1	ADD TENS
00219	B12C	D7	D8		STA B	TEMP1	
00220	B12E	81	00		CMP A	#0	
00221	B130	2E	F4		BGT	TENN	
00222	B132	D6	D8	NUL1	LDA B	TEMP1	CONVERT PER. FOR VDU O/P.
00223	B134	D6	D8		LDA B	TEMP1	
00224	B136	D7	E3		STA B	\$E3	CONVERT ROUTINE TO DISPLAY ON
00225	B138	7F	00E4		CLR	\$E4	
00226	B13B	7F	00E5		CLR	\$E5	
00227	B13E	86	08		LDA A	#08	
00228	B140	97	E6		STA A	\$E6	EXPONENT
00229	B142	7F	00E7		CLR	FLAG	
00230	B145	BD	B850		JSR	NORM	NORMALISE VALUE FOR MATH
00231	B148	BD	B629		JSR	SET10	SET OPERAND TO MULTIPLY BY 10
00232	B14B	BD	B907		JSR	MULT	MULTIPLY BY TEN


```

00233 B14E 96 E6      LDA A   $E6
00234 B150 81 10      EXS     CMP A   #$10      (DE-NORMALISE)
00235 B152 27 09      BEQ     EXF
00236 B154 74 00E3     LSR     $E3
00237 B157 76 00E4     ROR     $E4
00238 B15A 4C          INC A
00239 B15B 20 F3      BRA     EXS
00240 B15D 96 E3      EXF     LDA A   $E3
00241 B15F 97 F9      STA A   PE      STORE FOR VDU DISPLAY
00242 B161 96 E4      LDA A   $E4
00243 B163 97 FA      STA A   PE+1
00244 B165 7E B24C     JMP     NUL
00245                *      PERIOD STORED IN TEMP1.....D8
00246                *
00247                * CALIBRATOR CONTROLLED EXTERNALLY
00248                *
00249 B168 36          COMINT PSH A
00250 B169 81 4B      CMP A   #'K      HAS CALIB. BEEN CALLED FOR
00251 B16B 0C          CLC              K=RECOGNITION CHARACTER
00252 B16C 26 04      BNE     NOC      NO
00253 B16E BD B174     JSR     COMIN1   YES
00254 B171 0D          SEC
00255 B172 32          NOC     PUL A
00256 B173 39          RTS              RETURN TO STANOP
00257                *
00258                *
00259 B174 9F FB      COMIN1 STS     STACK
00260 B176 7D 00D7     TST     TEMP      HAS PER AND HI BEEN ENTERED
00261 B179 27 65      BEQ     CALL      NO
00262 B17B B6 C0A3     LDA A   COMBUF+2
00263 B17E 81 55      CMP A   #'U      UP
00264 B180 26 09      BNE     INT1
00265 B182 CE BEB2     LDX     #KEY
00266 B185 BD C545     JSR     VDUOUT   OUTPUT STOP PROMPT
00267 B188 7E B584     JMP     UP      MOVE PROBES UP
00268 B18B 81 44      INT1    CMP A   #'D      DOWN
00269 B18D 26 09      BNE     INT2
00270 B18F CE BEB2     LDX     #KEY
00271 B192 BD C545     JSR     VDUOUT   OUTPUT STOP PROMPT
00272 B195 7E B5C6     JMP     DOWN   MOVE PROBES DOWN
00273 B198 81 43      INT2    CMP A   #'C      CYCLE PROBES
00274 B19A 26 19      BNE     INT3
00275 B19C 86 A0      LDA A   #$A0      SET PUSHB. INPUT TO CYCLE
00276 B19E DE F3      LDX     STEP1
00277 B1A0 E6 00      LDA B   0,X
00278 B1A2 1B          ABA
00279 B1A3 B7 A002     STA A   PIAD2
00280 B1A6 BD B667     JSR     HEPE      DISPLAY PERIOD AND HEIGHT
00281 B1A9 BD C52F     JSR     VDURET
00282 B1AC CE BED2     LDX     #AKEY
00283 B1AF BD C545     JSR     VDUOUT   (STOP INSTRUCTION)
00284 B1B2 7E B507     JMP     CYCLE   CYCLE PROBES
00285 B1B5 81 52      INT3    CMP A   #'R      REENTER NEW VALUES
00286 B1B7 26 24      BNE     INT4
00287 B1B9 BD B667     JSR     HEPE      DISPLAY PERIOD & HEIGHT
00288 B1BC CE BF00     MIS     LDX     #QUES
00289 B1BF BD C545     JSR     VDUOUT   (CHANGE THESE VALUES?)
00290 B1C2 0C          CLC

```


00291	B1C3	BD	C5F9	CC	JSR	KBDCHK	KEYBOARD ENTRY ?
00292	B1C6	24	FB		BCC	CC	
00293	B1C8	81	59		CMP A	#'Y	YES
00294	B1CA	26	08		BNE	NN	
00295	B1CC	7F	00D7		CLR	TEMP	
00296	B1CF	7F	00DB		CLR	TEMP1	
00297	B1D2	20	0C		BRA	CALL	CHANGE PARAMETERS
00298	B1D4	81	4E	NN	CMP A	#'N	
00299	B1D6	27	05		BEQ	INT4	
00300	B1D8	BD	D551		JSR	WRONG	
00301	B1DB	20	DF		BRA	MIS	DO AGAIN
00302	B1DD	7E	B659	INT4	JMP	STOP	
00303				*			
00304	B1E0	CE	BCE0	CALL	LDX	#ENTHI	ENTER HEIGHT
00305	B1E3	BD	C53C		JSR	VDUOTA	
00306	B1E6	CE	C400		LDX	#\$C400	(ASCII STROBE)
00307	B1E9	BD	DA42		JSR	INDIG4	INPUT THE HEIGHT
00308	B1EC	25	F2		BCS	CALL	TO LARGE NO.
00309	B1EE	B6	C071		LDA A	U+3	
00310	B1F1	81	64		CMP A	#\$64	
00311	B1F3	22	EB		BHI	CALL	OUTSIDE LIMITS
00312	B1F5	81	0A		CMP A	#\$0A	
00313	B1F7	2B	E7		BMI	CALL	
00314	B1F9	97	D7		STA A	TEMP	STORE HEIGHT
00315	B1FB	97	F8		STA A	HE+1	REMEMBER VDU OUTPUT
00316	B1FD	CE	BD1D	CALL1	LDX	#ENTP	
00317	B200	BD	C53C		JSR	VDUOTA	OUTPUT PERIOD PROMPT
00318	B203	CE	C400		LDX	#\$C400	
00319	B206	BD	DA42		JSR	INDIG4	INPUT PERIODE
00320	B209	25	F2		BCS	CALL1	
00321	B20B	FE	C070		LDX	U+2	ENTERED VALUE
00322	B20E	DF	F9		STX	PE	VDU DISPLAY
00323	B210	DF	E3		STX	\$E3	MATH
00324	B212	86	10		LDA A	#\$10	EXPONENT
00325	B214	97	E6		STA A	\$E6	
00326	B216	7F	00E5		CLR	\$E5	
00327	B219	7F	00E7		CLR	FLAG	SIGN FLAG
00328	B21C	BD	B850		JSR	NORM	ADJUST EXPONENT ETC.
00329	B21F	BD	B629		JSR	SET10	SET TO DIVIDE BY 10
00330	B222	BD	B97B		JSR	DIV	DIVIDE BY 10
00331	B225	D6	E3		LDA B	\$E3	TEST FOR INTEGER IN \$E3
00332	B227	96	E6	AEX	LDA A	\$E6	EXPONENT
00333	B229	81	08		CMP A	#\$08	
00334	B22B	22	D0		BHI	CALL1	
00335	B22D	27	06		BEQ	GOOD	
00336	B22F	54			LSR B		
00337	B230	7C	00E6		INC	\$E6	
00338	B233	20	F2		BRA	AEX	
00339	B235	C1	C8	GOOD	CMP B	#\$C8	200 WITHIN LIMITS
00340	B237	24	C4		BCC	CALL1	TO BIG
00341	B239	C1	46		CMP B	#\$46	
00342	B23B	25	C0		BCS	CALL1	TO SMALL
00343	B23D	D7	D8		STA B	TEMP1	
00344	B23F	86	A0		LDA A	#\$A0	ALL LED'S OFF
00345	B241	DE	F3		LDX	STEP1	
00346	B243	E6	00		LDA B	0,X	
00347	B245	1B			ABA		
00348	B246	B7	A002		STA A	PIAD2	


```

00349 B249 BD EA7C      JSR      BUSY      COMPUTE CYCLE TABLE
00350
00666
00667
00668
00669
00670
00671 B4D6 09      SKIP2   DEX
00672 B4D7 DF D5      STX      CCOUNT
00673 B4D9 7D 00F6     TST      FLA      INT/EXT FLAG
00674 B4DC 26 03      BNE      SKIP3
00675 B4DE BD EA74     JSR      DONE      DISPLAY DONE
00676 B4E1 BD B659     SKIP3   JSR      STOP      DISPLAY/SET STOP LED
00677
00678
00679
00680 B4E4 86 FF      CONTRO  LDA  A  #$FF
00681 B4E6 97 F6      STA  A  FLA      $FF=INT; $00=EXT
00682 B4E8 B6 A000     LDA  A  PIAD1
00683 B4EB BD B642     JSR      ASL
00684 B4EE 81 0A      CMP  A  #$0A      *CYCLE*
00685 B4F0 26 03      BNE      TS1
00686 B4F2 BD B507     JSR      CYCLE
00687 B4F5 81 07      TS1     CMP  A  #$07      *UP*
00688 B4F7 26 03      BNE      TS2
00689 B4F9 BD B584     JSR      UP
00690 B4FC 81 0E      TS2     CMP  A  #$0E      *DOWN*
00691 B4FE 26 03      BNE      TS3
00692 B500 BD B5C6     JSR      DOWN
00693 B503 7F 00F6     TS3     CLR      FLA      CLEAR INT/EXT FLAG
00694 B506 39      RTS      RETURN TO STANOP
00695
00696
00697
00698 B507 7F 00DE     CYCLE    CLR      TEMP7      SET FLAGS
00699 B50A 96 D7      LDA  A  TEMP
00700 B50C 97 DC      STA  A  TEMP5
00701 B50E 97 DD      STA  A  TEMP6
00702 B510 97 D8      STA  A  TEMP1      DEC. TO GO UP
00703 B512 86 FF      LDA  A  #$FF
00704 B514 97 EA      STA  A  FLAG1      VDU ST LOST FLAG
00705 B516 97 DA      STA  A  TEMP3
00706 B518 97 DB      STA  A  TEMP4
00707 B51A 97 F5      STA  A  DIR      MOVE UP FIRST
00708 B51C 97 E7      STA  A  FLAG      IRQ MASK
00709 B51E 7F 00D9     CLR      TEMP2      (FIRST IRQ)
00710 B521 86 10      LDA  A  #$10
00711 B523 97 D2      STA  A  XREG
00712 B525 FE B01C     LDX      UPDOWN      START TIMERS
00713 B528 FF A092     STX      MSB1
00714 B52B 86 01      LDA  A  #$01      SELECT TH1
00715 B52D B7 A091     STA  A  CR2
00716 B530 86 D2      LDA  A  #$D2      TIMER MODE
00717 B532 B7 A090     STA  A  CR13      TIMER GO
00718 B535 DE D5      LDX      CCOUNT
00719 B537 DF CD      STX      OPRAND
00720 B539 B6 A0A1     LDA  A  MODEMD      CLEAR RDRF
00721 B53C B6 A000     CY1     LDA  A  PIAD1      TEST FOR STOP PUSHBUTTON

```


00722	B53F	BD	B642		JSR	ASL	
00723	B542	81	02		CMP	A	#02
00724	B544	27	25		BEQ		CY2
00725	B546	7D	00F6		TST		FLA
00726	B549	26	14		BNE		CY3
00727	B54B	F6	A0A0		LDA	B	MODEMC
00728	B54E	57			ASR	B	
00729	B54F	25	1A		BCS		CY2
00730	B551	7D	00EA		TST		FLAG1
00731	B554	26	09		BNE		CY3
00732	B556	CE	BD7C		LDX		#STEPL
00733	B559	BD	C545		JSR		VDUOUT
00734	B55C	73	00EA		COM		FLAG1
00735	B55F	96	E7	CY3	LDA	A	FLAG
00736	B561	81	FF		CMP	A	#FF
00737	B563	26	03		BNE		CY1A
00738	B565	0F			SEI		
00739	B566	20	D4		BRA		CY1
00740	B568	0E		CY1A	CLI		
00741	B569	20	D1		BRA		CY1
00742	B56B	96	D8	CY2	LDA	A	TEMP1
00743	B56D	81	00		CMP	A	#00
00744	B56F	27	02		BEQ		STP
00745	B571	20	F8		BRA		CY2
00746	B573	86	01	STP	LDA	A	#01
00747	B575	B7	A091		STA	A	CR2
00748	B578	86	02		LDA	A	#02
00749	B57A	B7	A090		STA	A	CR13
00750	B57D	DE	F3		LDX		STEP1
00751	B57F	0F			SEI		
00752	B580	BD	B659		JSR		STOP
00753	B583	39			RTS		
00754				*			
00755				*UP			
00756				*			
00757	B584	0F		UP	SEI		
00758	B585	86	20		LDA	A	#20
00759	B587	97	D2		STA	A	XREG
00760	B589	86	FF		LDA	A	#FF
00761	B58B	97	F5	DOWN1	STA	A	DIR
00762	B58D	86	00		LDA	A	#00
00763	B58F	97	DA		STA	A	TEMP3
00764	B591	FE	B01C		LDX		UPDOWN
00765	B594	FF	A092		STX		MSB1
00766	B597	86	01		LDA	A	#01
00767	B599	B7	A091		STA	A	CR2
00768	B59C	86	D2		LDA	A	#D2
00769	B59E	B7	A090		STA	A	CR13
00770	B5A1	7D	00F6		TST		FLA
00771	B5A4	27	0C		BEQ		UP3
00772	B5A6	B6	A000	UP1	LDA	A	PIAD1
00773	B5A9	BD	B642		JSR		ASL
00774	B5AC	81	02		CMP	A	#02
00775	B5AE	27	08		BEQ		UP2
00776	B5B0	20	F4		BRA		UP1
00777	B5B2	0C		UP3	CLC		
00778	B5B3	BD	C5F9	KC	JSR		KBDCHK
00779	B5B6	24	FB		BCC		KC

INT/EXT FLAG

TEST RDRF

TEST STOP FROM EXTERNAL CONTR

OUTPUT STEPS LOST TO VDU

SET IRQ FLAG(STEPS LOST TEST)

CLEAR IRQ FLAG(STEPS LOST TES

HALT STEPPER MOTOR

SELECT T#1

MASK IRQ

AS SETUP BY STANOP

SET MASK

DIRECTION

CYCLE OR UP/DOWN FLAG

START TIMERS

SELECT T#1

MODE

INT/EXT FLAG

CHECK FOR STOP(FRONT PANEL)

TEST EXTERNAL STOP

00780	B5B8	86	01	UP2	LDA A	#\$01	HALT TIMERS
00781	B5BA	B7	A091		STA A	CR2	SELECT TH1
00782	B5BD	86	02		LDA A	#\$02	
00783	B5BF	B7	A090		STA A	CR13	RESET TO AS SETUP BY STANOP
00784	B5C2	BD	B659		JSR	STOP	
00785	B5C5	39			RTS		
00786				*			
00787				*DOWN			
00788				*			
00789	B5C6	0F		DOWN	SEI		SET MASK
00790	B5C7	86	30		LDA A	#\$30	
00791	B5C9	97	D2		STA A	XREG	
00792	B5CB	86	00		LDA A	#\$00	
00793	B5CD	7E	B58B		JMP	DOWN1	
00794				*			
00869					OPT	L	
00870				*			
00871	B659	DE	F3	STOP	LDX	STEP1	
00872	B65B	A6	00		LDA A	0,X	LAST STEP TO MOTOR
00873	B65D	C6	F0		LDA B	#\$F0	
00874	B65F	1B			ABA		
00875	B660	B7	A002		STA A	PIAD2	LIGHT STOP LED
00876	B663	7F	00F6		CLR	FLA	CLEAR INT/EXT FLAG
00877	B666	39			RTS		RETURN TO STANOP
00878				*			
00879	B667	CE	BE90	HEPE	LDX	#HEI	DISPLAY HEIGHT ON VDU
00880	B66A	BD	C53C		JSR	VDUOTA	
00881	B66D	7F	00F7		CLR	HE	
00882	B670	CE	00F7		LDX	#HE	
00883	B673	BD	D961		JSR	BINBCD	BINARY TO BCD
00884	B676	7F	C072		CLR	FF	
00885	B679	BD	D996		JSR	VDUASC	OUTPUT TO VDU
00886	B67C	BD	C52F		JSR	VDURET	
00887	B67F	CE	BEA0		LDX	#PER	DISPLAY PERIODE ON VDU
00888	B682	BD	C545		JSR	VDUOUT	
00889	B685	CE	00F9		LDX	#PE	
00890	B688	BD	D961		JSR	BINBCD	CONVERT TO BCD
00891	B68B	7F	C072		CLR	FF	
00892	B68E	BD	D996		JSR	VDUASC	OUTPUT TO VDU
00893	B691	BD	C52F		JSR	VDURET	
00894	B694	39			RTS		
00895				*			
00896	B695	86	B0	ALM	LDA A	#\$B0	SOUND ALARM
00897	B697	DE	F3		LDX	STEP1	
00898	B699	E6	00		LDA B	0,X	
00899	B69B	1B			ABA		
00900	B69C	B7	A002		STA A	PIAD2	
00901	B69F	39			RTS		
00902				*			
00903				* ERROR MESSAGES			
00904				*			
00905				* RATIO/LIMITS FAULT			
00906	B6A0	BD	B695	ERROR	JSR	ALM	ALARM
00907	B6A3	7D	00F6		TST	FLA	INT/EXT FLAG
00908	B6A6	27	0C		BEQ	E1EX	
00909	B6A8	86	E0	ERRORR	LDA A	#\$E0	RATIO/LIMITS LIGHT
00910	B6AA	DE	F3		LDX	STEP1	
00911	B6AC	E6	00		LDA B	0,X	

00912	B6AE	1B		ABA		
00913	B6AF	B7	A002	STA A	PIAD2	
00914	B6B2	20	F4	BRA	ERRORR	
00915	B6B4	CE	BD5D	LDX	#LIMRAT	
00916	B6B7	BD	C545	JSR	VDUOUT	PROMPT TO VDU
00917	B6BA	31		INS		FOR RTS
00918	B6BB	31		INS		
00919	B6BC	7F	00D7	CLR	TEMP	
00920	B6BF	7F	00D8	CLR	TEMP1	
00921	B6C2	7E	B1E0	JMP	CALL	REENTER VALUES FROM KEYBOARD
00922			*			
00923			*	STEPS	LOST	FAULT
00924	B6C5	BD	B695	JSR	ALM	
00925	B6C8	7D	00F6	TST	FLA	INT/EXT FLAG
00926	B6CB	27	0E	BEQ	E2EX	
00927	B6CD	86	D0	LDA A	#D0	STEPS LOST LIGHT
00928	B6CF	97	D2	STA A	XREG	
00929	B6D1	73	00D9	COM	TEMP2	
00930	B6D4	96	D7	LDA A	TEMP	
00931	B6D6	97	DC	STA A	TEMP5	
00932	B6D8	97	DD	STA A	TEMP6	
00933	B6DA	39		RTS		
00934	B6DB	7F	00EA	CLR	FLAG1	SET VDU OUTPUT FLAG
00935	B6DE	20	ED	BRA	E2	
00936			*			
00937			*	LIMITS	FAULT	
00938	B6E0	86	01	ESTOPS	LDA A	#01
00939	B6E2	B7	A091	STA A	CR2	SELECT T#1
00940	B6E5	86	02	LDA A	#02	RESET AS SET BY STANOP
00941	B6E7	B7	A090	STA A	CR13	PTM STOP
00942	B6EA	BD	B695	JSR	ALM	
00943	B6ED	86	C0	LDA A	#C0	E/S LIGHT
00944	B6EF	DE	F3	LDX	STEP1	
00945	B6F1	E6	00	LDA B	0,X	
00946	B6F3	1B		ABA		
00947	B6F4	B7	A002	STA A	PIAD2	
00948	B6F7	73	00F5	COM	DIR	
00949	B6FA	96	F5	LDA A	DIR	
00950	B6FC	81	00	CMP A	#00	
00951	B6FE	27	19	BEQ	DOWN2	
00952	B700	7D	00F6	TST	FLA	INT/EXT FLAG
00953	B703	27	0C	BEQ	E3U	
00954	B705	B6	A000	LDA A	PIAD1	PRESS UP TO RESET
00955	B708	BD	B642	JSR	ASL	
00956	B70B	81	07	CMP A	#07	
00957	B70D	27	21	BEQ	PROG	RETURN TO PROGRAM
00958	B70F	20	F4	BRA	UP4A	
00959	B711	CE	BD9B	LDX	#ULIM	
00960	B714	BD	C545	JSR	VDUOUT	
00961	B717	20	17	BRA	PROG	
00962	B719	7D	00F6	TST	FLA	INT/EXT FLAG
00963	B71C	27	0C	BEQ	E3D	
00964	B71E	B6	A000	LDA A	PIAD1	PRESS DOWN TO RESET
00965	B721	BD	B642	JSR	ASL	
00966	B724	81	0E	CMP A	#0E	
00967	B726	27	08	BEQ	PROG	
00968	B728	20	F4	BRA	D02	
00969	B72A	CE	BDCB	LDX	#DLIM	


```

00970 B72D BD C545      JSR      VDUOUT
00971 B730 9E FB      PROG    LDS      STACK      RESTORE STACK
00972 B732 7D 00F6      TST      FLA          INT/EXT
00973 B735 27 03      BEQ      PROGEX
00974 B737 7E B05E      JMP      ENT
00975 B73A 7E B659      PROGEX  JMP      STOP
00976                      *
00977 BC00                      ORG      $BC00
00978                      *
00979                      *INTERRUPTS
00980                      *
00981 BC00 B6 A000      NMI      LDA      A      PIAD1      TEST END/STOPS
00982 BC03 BD B642      JSR      ASL
00983 BC06 84 02      AND      A      #$02
00984 BC08 81 00      CMP      A      #$00
00985 BC0A 26 03      BNE      NMI1
00986 BC0C 7E B6E0      JMP      ESTOPS      LIMITS ERROR
00987 BC0F 96 DA      NMI1     LDA      A      TEMP3      (NMI)
00988 BC11 81 FF      CMP      A      #$FF      CYCLE
00989 BC13 27 09      BEQ      CYC
00990 BC15 FE B01C      LDX      UPDOWN
00991 BC18 FF A092      STX      MSB1      START TIMERS
00992 BC1B 7E BC89      JMP      DIRRR      UP OR DOWN ONLY
00993 BC1E DE CD      CYC      LDX      OPRAND      CYCLE PROBES
00994 BC20 9F D3      STS      XXREG
00995 BC22 AE 00      LDS      0,X
00996 BC24 BF A092      STS      MSB1
00997 BC27 9E D3      LDS      XXREG
00998 BC29 96 DB      LDA      A      TEMP4      INC OR DEC
00999 BC2B 81 00      CMP      A      #$00      DEC=FF, INC=00
01000 BC2D 27 2D      BEQ      INC1
01001 BC2F 7A 00D8      DEC      TEMP1
01002 BC32 26 08      BNE      DEX1      AT BOTTOM OF CYCLE ?
01003 BC34 73 00DB      COM      TEMP4      YES, SET TO INCREMENT
01004 BC37 7C 00DE      INC      TEMP7      1/4 CYCLE COUNTER
01005 BC3A 20 38      BRA      DIRR
01006 BC3C 09      DEX1     DEX      DEC. CYCLE TABLE
01007 BC3D 09      DEX
01008 BC3E DF CD      STX      OPRAND
01009 BC40 96 DE      LDA      A      TEMP7
01010 BC42 81 00      CMP      A      #$00      FIRST 1/4 CYCLE ?
01011 BC44 26 2E      BNE      DIRR
01012 BC46 96 DC      LDA      A      TEMP5      WITHIN IRQ LIMITS
01013 BC48 D6 D8      LDA      B      TEMP1      (STEPS LOST TEST)
01014 BC4A 10      SBA
01015 BC4B 2D 09      BLT      SMASK
01016 BC4D 91 DD      CMP      A      TEMP6
01017 BC4F 2C 05      BGE      SMASK
01018 BC51 7F 00E7      CLR      FLAG      CLI
01019 BC54 20 33      BRA      DIRRR
01020 BC56 86 FF      SMASK    LDA      A      #$FF
01021 BC58 97 E7      STA      A      FLAG      SEI
01022 BC5A 20 2D      BRA      DIRRR
01023 BC5C 7C 00D8      INC1     INC      TEMP1      INCREMENT
01024 BC5F 96 D7      LDA      A      TEMP
01025 BC61 91 D8      CMP      A      TEMP1
01026 BC63 26 0B      BNE      INX1      NOT AT TOP OF CYCLE YET
01027 BC65 7C 00DE      INC      TEMP7      1/4 CYC COUNT

```


01028	BC68	73	00F5	COM	DIR	CHANGE MOTOR DIRECTION
01029	BC6B	73	00DB	COM	TEMP4	INC/DEC TABLE FLAG
01030	BC6E	20	04	BRA	DIRR	
01031	BC70	08		INX1	INX	INCREMENT CYCLE TABLE
01032	BC71	08			INX	
01033	BC72	DF	CD		STX	OPRAND
01034	BC74	96	DE	DIRR	LDA A	TEMP7
01035	BC76	81	04		CMP A	#\$04
01036	BC78	26	0F		BNE	DIRRR
01037	BC7A	86	FF		LDA A	#\$FF
01038	BC7C	91	DF		CMP A	TEMP8
01039	BC7E	27	03		BEQ	CLR2
01040	BC80	BD	B6C5		JSR	ERRO
01041	BC83	7F	00DF	CLR2	CLR	TEMP8
01042	BC86	7F	00DE		CLR	TEMP7
01043	BC89	96	F5	DIRRR	LDA A	DIR
01044	BC8B	81	00		CMP A	#\$00
01045	BC8D	27	18		BEQ	CYDOWN
01046	BC8F	DE	F3		LDX	STEP1
01047	BC91	A6	00		LDA A	0,X
01048	BC93	D6	D2		LDA B	XREG
01049	BC95	1B			ABA	
01050	BC96	B7	A002		STA A	PIAD2
01051	BC99	A6	00		LDA A	0,X
01052	BC9B	81	02		CMP A	#\$02
01053	BC9D	26	05		BNE	INST
01054	BC9F	CE	B014		LDX	#STEP
01055	BCA2	20	19		BRA	STX1
01056	BCA4	08		INST	INX	SET NEXT STEP
01057	BCA5	20	16		BRA	STX1
01058	BCA7	DE	F3	CYDOWN	LDX	STEP1
01059	BCA9	A6	00		LDA A	0,X
01060	BCAB	D6	D2		LDA B	XREG
01061	BCAD	1B			ABA	
01062	BCAE	B7	A002		STA A	PIAD2
01063	BCB1	A6	00		LDA A	0,X
01064	BCB3	81	0A		CMP A	#\$0A
01065	BCB5	26	05		BNE	DECST
01066	BCB7	CE	B01B		LDX	#STEPS
01067	BCBA	20	01		BRA	STX1
01068	BCBC	09		DECST	DEX	
01069	BCBD	DF	F3	STX1	STX	STEP1
01070	BCBF	3B			RTI	
01071				*		
01072				* IRQ		
01073				*		
01074	BCC0	B6	A000		LDA A	PIAD1
01075	BCC3	96	D9		LDA A	TEMP2
01076	BCC5	81	FF		CMP A	#\$FF
01077	BCC7	27	12		BEQ	IRQ1
01078	BCC9	C6	05		LDA B	#\$05
01079	BCCB	DB	D8		ADD B	TEMP1
01080	BCCD	96	D7		LDA A	TEMP
01081	BCCF	10			SBA	
01082	BCD0	2D	0D		BLT	RET
01083	BCD2	D7	DC		STA B	TEMP5
01084	BCD4	86	0A		LDA A	#\$0A
01085	BCD6	97	DD		STA A	TEMP6


```

01086 BCDB 73 00D9      COM      TEMP2
01087 BCDB 86 FF      IRQ1      LDA A  #$FF
01088 BCDD 97 DF      STA A  TEMP8
01089 BCDF 3B      RET      RTI
01090      *
01091      *
01092      * PROMPTS
01093      *
01094      *
01095 BCE0 45      ENTHI      FCC      /ENTER THE CALIBRATION HEIGHT/
01096 BCFC 0A      FCB      $A,$D
01097 BCFE 42      FCC      /BETWEEN LIMITS 10mm TO 100mm/
01098 BD1A 0A      FCB      $A,$D,4
01099      *
01100 BD1D 45      ENTP      FCC      /ENTER THE CALIBRATION PERIOD/
01101 BD39 0A      FCB      $A,$D
01102 BD3B 42      FCC      /BETWEEN LIMITS 700 TO 2000 msec/
01103 BD5A 0A      FCB      $A,$D,4
01104      *
01105 BD5D 49      LIMRAT      FCC      /INPUT LIMITS OR RATIO FAULT/
01106 BD78 0A      FCB      $A,$D,7,4
01107      *
01108 BD7C 43      STEPL      FCC      /CALIBRATOR MOTOR STEPS LOST/
01109 BD97 0A      FCB      $A,$D,7,4
01110      *
01111 BD9B 4A      ULM      FCC      /JIG EXCEEDS ITS BOTTOM LIMIT/
01112 BDB7 0A      FCB      $A,$D
01113 BDB9 50      FCC      /PUSH K;U:(RUN)/
01114 BDC7 0A      FCB      $A,$D,7,4
01115      *
01116 BDCB 4A      DLM      FCC      /JIG EXCEEDS ITS TOP LIMIT/
01117 BDE4 0A      FCB      $A,$D
01118 BDE6 50      FCC      /PUSH K;D:(RUN)/
01119 BDF4 0A      FCB      $A,$D,7,4
01120      *
01121 BDF8 50      COM      FCC      /PROBE CALIBRATOR      K;OPTIONS:(RUN)
01122 BE1C 0A      FCB      $A,$D
01123 BE1E 0A      FCB      $A,$D
01124 BE20 43      FCC      /CALIBRATOR OPTIONS ; U=PROBES MOVED
01125 BE5B 0A      FCB      $A,$D
01126 BE5D 43      FCC      /C=CYCLE PROBES,R=DISPLAY OR ENTER NE
01127 BE8D 0A      FCB      $A,$D,4
01128      *
01129 BE90 48      HEI      FCC      /HEIGHT IN mm = /
01130 BE9F 04      FCB      4
01131      *
01132 BEA0 50      PER      FCC      /PERIOD IN msec = /
01133 BEB1 04      FCB      4
01134      *
01135 BEB2 53      KEY      FCC      /STRIKE ANY KEY TO STOP PROBES/
01136 BECF 0A      FCB      $A,$D,$4
01137      *
01138 BED2 54      AKEY      FCC      /TO STOP THE PROBES ENTER      : ANY KE
01139 BEFD 0A      FCB      $A,$D,$4
01140      *
01141 BF00 44      QUES      FCC      /DO YOU WISH TO CHANGE THESE VALUES?.
01142 BF2E 0A      FCB      $A,$D,$4
01143      *

```


APPENDIX EMPC/TROLLEY CONTROLLER PROGRAM LISTING

The following is the listing of the MPC and the trolley controller program (both use the same program). It is inevitable that this program will be changed or added to, to correct any weaknesses found when these instruments are used more extensively.

Flow diagrams

Also included are flow diagrams of the MPC and trolley controller program.


```

00001
00002
00003
00004
00005
00006
00007
00008
00009
00010
00011
00012
00013
00014
00015
00016
00017
00018
00019
00020
00021
00022
00023
00024
00025
00026
00027
00028
00029
00030
00031
00032
00033
00034
00035
00036
00037
00038
00039
00040
00041
00042
00043
00044
00045
00046
00047
00048
00049
00050
00051
00052
00053
00054
00055
00056
00057
00058

```

		OPT	L	
		NAM	MPC	
		* MEASUREMENT BRIDGE CONTROLLER		
		* -----		
		* STANOP VERSION		
		* -----		
		* SYSTEM EQUATES		
		* -----		
		* -----		
	A000	PIAD1	EQU	\$A000
	A001	PIAC1	EQU	\$A001
	A002	PIAD2	EQU	\$A002
	A003	PIAC2	EQU	\$A003
	A004	PIAD3	EQU	\$A004
	A005	PIAC3	EQU	\$A005
	A006	PIAD4	EQU	\$A006
	A007	PIAC4	EQU	\$A007
		* -----		
	A098	PTMC13	EQU	\$A098
	A099	PTMC2	EQU	\$A099
	A09B	PT1LSB	EQU	\$A09B
	A09A	PT1MSB	EQU	\$A09A
	A09D	PT2LSB	EQU	\$A09D
	A09C	PT2MSB	EQU	\$A09C
	A09F	PT3LSB	EQU	\$A09F
	A09E	PT3MSB	EQU	\$A09E
		* -----		
		* ID EQUATES		
	A0B0	ESXCL	EQU	\$A0B0
	A0B1	ESXFL	EQU	\$A0B1
	A0B2	ESXCH	EQU	\$A0B2
	A0B3	ESXFH	EQU	\$A0B3
	A0B4	ESYCL	EQU	\$A0B4
	A0B5	ESYFL	EQU	\$A0B5
	A0B6	ESYCH	EQU	\$A0B6
	A0B7	ESYFH	EQU	\$A0B7
	A0B8	HOMCX	EQU	\$A0B8
	A0B9	HOMCY	EQU	\$A0B9
		* -----		
		* STANOP EQUATES		
	C0A1	COMBUF	EQU	\$C0A1
	C072	FF	EQU	\$C072
	C92F	VDU	EQU	\$C92F
	D996	VDUASC	EQU	\$D996
	C545	VDUOUT	EQU	\$C545
	C52F	VDURET	EQU	\$C52F
	C5F9	KBDCHK	EQU	\$C5F9
	D961	BINBCD	EQU	\$D961
	C5C3	CHARIN	EQU	\$C5C3
	EA7C	BUSY	EQU	\$EA7C
	D551	WRONG	EQU	\$D551
	C726	CHKOFF	EQU	\$C726
	E6F6	COMCLR	EQU	\$E6F6
	DE48	GETNMH	EQU	\$DE48
		* -----		
	A0A0	MODEMC	EQU	\$A0A0
	A0A1	MODEMD	EQU	\$A0A1

KEYBOARD/MOTOR DIR. PUSHBUTTON
 DISPLAY
 MODE LED'S
 MOTOR CONTROL/FINE POS.BD. CO

WRITE CR #1,#3
 WRITE CR #2
 WRITE #1 LATCHES
 WRITE MSB
 WRITE #2 LATCHES
 WRITE MSB
 WRITE #3 LATCHES
 WRITE MSB

COARSE LOWER ENDSTOP
 FINE
 COARSE HIGHER ENDSTOP
 FINE
 COARSE LOWER ENDSTOP
 FINE
 COARSE HIGHER ENDSTOP
 FINE
 HOME POSITION
 Y HOME POSITION

COMMAND BUFFER
 CHAR. STRING TO VDU
 CR,LF
 CHECKS KB. FOR INPUT
 BINARY TO BCD
 INPUTS CHAR FROM KB.

00059		*			
00060	A21F	TARGET EQU	\$A21F		
00061	A23F	TARG1 EQU	\$A23F		
00062	A23E	XSPFLG EQU	\$A23E		SPEED CONT TARGET WORD
00063		*			
00064		*			
00065		* SCRATCH PAD			
00066		* -----			
00067		*			
00068	0000	ORG	\$00		
00069	0000 0001	TEMP RMB	1		INFO
00070	0001 0004	XCPOS RMB	4		X-COARSE POSITION
00071	0005 0004	XFPOS RMB	4		X-FINE
00072	0009 0004	YCPOS RMB	4		Y-COARSE
00073	000D 0004	YFPOS RMB	4		Y-FINE
00074	0011 0001	XER RMB	1		INCORRECT FIX COUNTERS
00075	0012 0001	YER RMB	1		
00076	0013 0004	XCTARG RMB	4		
00077	0017 0004	XFTARG RMB	4		
00078	001B 0004	YCTARG RMB	4		
00079	001F 0004	YFTARG RMB	4		
00080	0023 0004	XCGOTO RMB	4		GOTO POS
00081	0027 0004	XFGOTO RMB	4		
00082	002B 0004	YCGOTO RMB	4		
00083	002F 0004	YFGOTO RMB	4		
00084	0033 0004	XCSCCL RMB	4		SCALE
00085	0037 0004	XFSCCL RMB	4		
00086	003B 0004	YCSCCL RMB	4		
00087	003F 0004	YFSCCL RMB	4		
00088	0043 0004	XCHOME RMB	4		HOME
00089	0047 0004	XFHOME RMB	4		
00090	004B 0004	YCHOME RMB	4		
00091	004F 0004	YFHOME RMB	4		
00092	0053 0004	XCSPD RMB	4		SPEED
00093	0057 0004	XFSPD RMB	4		
00094	005B 0004	YCSPD RMB	4		
00095	005F 0004	YFSPD RMB	4		
00096	0063 0001	XTEST RMB	1		X POSITION +,-1 TEST
00097	0064 0001	YTEST RMB	1		Y POSITION +,-1 TEST
00098	0065 0002	T1 RMB	2		MATH
00099	0067 0001	T2 RMB	1		COVERSIONS STORAGE
00100	0068 0001	T3 RMB	1		DITTO
00101	0069 0001	T4 RMB	1		EXPONENT REMEMBER
00102	006A 0001	T5 RMB	1		EXPONENT/PUSHBUTTON REMEMBER
00103	006B 0001	T6 RMB	1		SIGN REMEMBER
00104	006C 0001	T7 RMB	1		SIGN REMEMBER
00105	006D 0001	T8 RMB	1		TX ERROR CLEAR
00106	006E 0001	T9 RMB	1		X OR Y ERROR FLAG
00107	006F 0001	XSTEP RMB	1		MOTOR SPEED PRESET FLAGS
00108	0070 0001	YSTEP RMB	1		
00109	0071 0001	XSTEP C RMB	1		SPEED INC/DEC STEPS COUNTERS
00110	0072 0001	YSTEP C RMB	1		
00111	0073 0001	XINC RMB	1		INC/DEC FLAGS
00112	0074 0001	YINC RMB	1		
00113	0075 0002	REG RMB	2		Y DISPLAY REM
00114	0077 0002	REG1 RMB	2		X DISPLAY REM
00115	0079 0002	REG2 RMB	2		CHANGE REG
00116	007B 0002	REG3 RMB	2		MOVE SPEED REG REMEMBER

00117	007D	0002	REG4	RMB	2	SPEED TABLE REGISTERS
00118	007F	0002	REG5	RMB	2	
00119	0081	0002	STT	RMB	2	STACK STORAGE
00120	0083	0002	STT1	RMB	2	DITTO
00121	0085	0005	IREG	RMB	5	MATH CHANGE REG.'S
00122	008A	0005	IREG1	RMB	5	
00123	008F	0001	DIR	RMB	1	DIRECTION
00124	0090	0001	MODE	RMB	1	X MOVEMENT
00125	0091	0001	MODE1	RMB	1	Y MOVEMENT
00126	0092	0001	LAT	RMB	1	LED MODE REM
00127	0093	0001	FLA	RMB	1	INT/EXT FLAG
00128	0094	0001	FLAG	RMB	1	INFO INPUT
00129	0095	0001	FLAG1	RMB	1	DITTO
00130	0096	0001	FLAG2	RMB	1	CLEAR ROOUTINE
00131	0097	0001	FLAG3	RMB	1	MATH SIGN BIT
00132	0098	0001	FLAG4	RMB	1	X INPUT FLAG
00133	0099	0001	FLAG5	RMB	1	Y INPUT FLAG
00134	009A	0001	FLAGXY	RMB	1	X or Y SELECT
00135	009B	0001	AFLAG	RMB	1	ABORT FLAG
00136	009C	0001	SPEED1	RMB	1	NO OF SPEED INC. or DEC.
00137	009D	0001	DO0	RMB	1	MATH
00138	009E	0001	DO1	RMB	1	
00139	009F	0001	DO2	RMB	1	
00140	00A0	0001	DO3	RMB	1	
00141	00A1	0001	DO4	RMB	1	
00142	00A2	0001	DO5	RMB	1	
00143	00A3	0001	DO6	RMB	1	
00144	00A4	0001	XSCALE	RMB	1	X SCALE
00145	00A5	0001	YSCALE	RMB	1	Y SCALE
00146	00A6	0001	BC10VE	RMB	1	BC1 or BC2 OVERRIDE FLAG
00147	00A7	0002	XSPDCT	RMB	2	CONTROLLER SPEED REMEMBER
00148	00A9	0001	XSPMK2	RMB	1	MK2 SPEED CONTROLLED
00149	00AA	0001	XSPDFL	RMB	1	S/CONT. IN REST OR OPERATE MO
00150			*			
00151	A200			ORG	\$A200	
00152	A200	0010	XYPOS	RMB	16	POSITION FIXING MODULE
00153	A210	0001	XXER	RMB	1	
00154	A211	0001	YYER	RMB	1	
00155	A212	0001	DIR1	RMB	1	
00156			*			
00157	A220			ORG	\$A220	X SPEED CONTROL
00158	A220	0001	XSPDHI	RMB	1	PERIOD HI BYTE
00159	A221	0001	XSPDLO	RMB	1	PERIOD LO BYTE
00160	A222	0001	DIR2	RMB	1	DIECTION FLAG
00161			*			
00162			*			
00163			*			DIRECTION CONTROL WORDS
00164			*			\$00-----HALT
00165			*			\$11-----FORWARD
00166			*			\$22-----REVERSE
00167			*			\$44-----LEFT
00168			*			\$88-----RIGHT
00169			*			\$14-----FWD/LEFT
00170			*			\$18-----FWD/RIGHT
00171			*			\$24-----REV/LEFT
00172			*			\$28-----REV/RIGHT
00173			*			
00174			*			


```

00175          * STANOP CONTROL ROUTINE
00176          * -----
00177 1000          ORG      $1000
00178 1000 7E 1021    JMP      COMINT    COMMAND INTERPRETER
00179 1003 7E 102D    JMP      SETUP     SETUP MPC
00180 1006 7E 1010    JMP      PTEST
00181 1009 BD C545 ABC JSR      VDUOUT    MPC PROMPT
00182 100C 39          RTS
00183 100D 7E 110D    JMP      MPC        MPC MAIN ROUTINE
00184          *
00185 1010 7D 00A9 PTEST TST      XSPMK2    MK1/MK2 SPD. CONT?
00186 1013 27 06      BEQ      PTEST1
00187 1015 CE 2B07    LDX      #MT1      TROLLEY PROMPT
00188 1018 7E 1009    JMP      ABC
00189 101B CE 2A00 PTEST1 LDX      #MP1    MPC PROMPT
00190 101E 7E 1009    JMP      ABC
00191          *
00192 1021 36          COMINT PSH A
00193 1022 81 4D      CMP A    #'M        HAS MPC BEEN CALLED
00194 1024 0C          CLC                      M=RECOGNITION CHARACTER
00195 1025 26 04      BNE      COMNO        NO
00196 1027 BD 1C15    JSR      COMYES
00197 102A 0D          SEC
00198 102B 32          COMNO PUL A
00199 102C 39          RTS
00200          *
00201          *
00202          * START-UP ROUTINE
00203          * -----
00204 102D 86 00      SETUP LDA A    #$00        SET PIA'S
00205 102F B7 A001    STA A    PIAC1
00206 1032 B7 A003    STA A    PIAC2
00207 1035 B7 A005    STA A    PIAC3
00208 1038 B7 A007    STA A    PIAC4
00209 103B B7 A000    STA A    PIAD1        INPUT FROM KEYBOARD
00210 103E 86 FF      LDA A    #$FF
00211 1040 B7 A004    STA A    PIAD3        MODE LED'S
00212 1043 B7 A002    STA A    PIAD2        DISPLAY OUTPUT
00213 1046 B7 A006    STA A    PIAD4        MOTOR CONTROL/F.POS.BOARD
00214 1049 86 3C      LDA A    #$3C
00215 104B B7 A005    STA A    PIAC3
00216 104E B7 A007    STA A    PIAC4
00217 1051 86 07      LDA A    #$07
00218 1053 B7 A001    STA A    PIAC1
00219 1056 86 3E      LDA A    #$3E
00220 1058 B7 A003    STA A    PIAC2
00221 105B B6 A000    LDA A    PIAD1        RESET FLAGS
00222 105E B6 A002    LDA A    PIAD2
00223          *
00224 1061 86 34      LDA A    #$34
00225 1063 B7 A007    STA A    PIAC4        RESET FUNCTION BOARD
00226 1066 BD 11E8    JSR      DELAY        DELAY
00227 1069 86 3C      LDA A    #$3C
00228 106B B7 A007    STA A    PIAC4
00229 106E BD 11F0    JSR      DELAY1
00230          *
00231 1071 86 15      LDA A    #$15        SET PTM
00232 1073 B7 A099    STA A    PTMC2        CR #1 MAY BE WRITTEN

```


00233	1076	C6	97	LDA B	#\$97	
00234	1078	F7	A098	STA B	PTMC13	TIMERS PRESET/MODE
00235	107B	86	14	LDA A	#\$14	
00236	107D	B7	A099	STA A	PTMC2	CR#3 MAY BE WRITTEN
00237	1080	C6	96	LDA B	#\$96	
00238	1082	F7	A098	STA B	PTMC13	MODE
00239	1085	86	27	LDA A	#\$27	PULSE WIDTH
00240	1087	C6	FF	LDA B	#\$FF	LOW FREQ
00241	1089	F7	A09A	STA B	PT1MSB	
00242	108C	B7	A09B	STA A	PT1LSB	
00243	108F	F7	A09E	STA B	PT3MSB	
00244	1092	B7	A09F	STA A	PT3LSB	
00245	1095	C6	63	LDA B	#\$63	SPEED INC/DEC RATE
00246	1097	F7	A09C	STA B	PT2MSB	
00247	109A	86	C7	LDA A	#\$C7	PULSE WIDTH
00248	109C	B7	A09D	STA A	PT2LSB	
00249			*			
00250	109F	86	C1	LDA A	#\$C1	
00251	10A1	B7	A004	STA A	PIAD3	LIGHT LEDS(POS./X/Y)
00252	10A4	4F		CLR A		
00253	10A5	B7	A006	STA A	PIAD4	CLEAR OVERRIDE CONTROL
00254	10A8	5F		CLR B		
00255	10A9	CE	0000	LDX	#TEMP	
00256	10AC	A7	00	STA A	0,X	CLEAR INFO REGS.
00257	10AE	08		INX		
00258	10AF	5C		INC B		
00259	10B0	C1	F0	CMF B	#\$F0	
00260	10B2	26	F8	BNE	CL	
00261			*			
00262	10B4	CE	0009	LDX	#YCPOS	
00263	10B7	DF	75	STX	REG	SET DISPLAY REGISTER
00264	10B9	CE	0001	LDX	#XCPOS	
00265	10BC	DF	77	STX	REG1	
00266	10BE	86	AA	LDA A	#\$AA	
00267	10C0	97	9A	STA A	FLAGXY	00-X;FF-Y;AA-XY
00268	10C2	86	05	LDA A	#\$05	
00269	10C4	97	63	STA A	XTEST	
00270	10C6	97	64	STA A	YTEST	
00271	10C8	73	0098	COM	FLAG4	
00272	10CB	73	0099	COM	FLAG5	
00273	10CE	86	C0	LDA A	#\$C0	
00274	10D0	B7	A002	STA A	PIAD2	CLEAR ERROR DIGIT
00275			*			
00276	10D3	CE	0043	LDX	#XCHOME	LOAD HOME POSITION FROM ID BO
00277	10D6	B6	A0B8	LDA A	HOMCX	
00278	10D9	A7	03	STA A	3,X	
00279	10DB	BD	1241	JSR	CON	
00280	10DE	CE	004B	LDX	#YCHOME	
00281	10E1	B6	A0B9	LDA A	HOMCY	
00282	10E4	A7	03	STA A	3,X	
00283	10E6	BD	1241	JSR	CON	
00284			*			
00285	10E9	86	01	LDA A	#\$01	SET SCALE TO ONE
00286	10EB	97	35	STA A	XCSC+2	
00287	10ED	97	36	STA A	XCSC+3	
00288	10EF	97	3D	STA A	YCSC+2	
00289	10F1	97	3E	STA A	YCSC+3	
00290			*			


```

00291 10F3 B6 A23F      LDA A  TARG1      MK1 OR MK2 CONTROLLER
00292 10F6 81 55        CMP A  #$55      00-MK1  FF-MK2
00293 10F8 26 0B        BNE      MK1
00294 10FA 86 FF        LDA A  #$FF
00295 10FC 97 A9        STA A  XSPMK2
00296 10FE 7F 009A      CLR      FLAGXY      X AXIS ONLY
00297 1101 86 01        LDA A  #$01
00298 1103 97 58        STA A  XFSPD+1    SET TO MAX SPEED
00299 1105 0F          MK1 SEI
00300 1106 CE 2550      LDX      #IRQ
00301 1109 FF FFF8      STX      $FFFF      SET IRQ VECTOR
00302 110C 39          RTS
00303
00304
00305
00306
00307
00308 110D 9F 83      MPC STS      STT1      REMEMBER STACK POINTER
00309 110F 86 FF      LDA A  #$FF
00310 1111 97 93      STA A  FLA      (FF-INT;00-EXT)
00311 1113 BD 115B      JSR      GETPOS      READ POSITION FROM BC3
00312 1116 0E          CLI
00313 1117 BD 11B8      JSR      DISPLAY    DISPLAY POSITION
00314 111A BD 1181      JSR      FLAGP2     CHECK CB1 PIA2(BC2 OVERRIDE)
00315 111D 0F          MPC1 SEI
00316 111E 7F 0093      CLR      FLA
00317 1121 9E 83      LDS      STT1
00318 1123 39          RTS
00319
00320
00321
00322
00323
00324
00325 1124 B6 A21F      SPM LDA A  TARGET
00326 1127 81 55        CMP A  #$55      INVERTED BUS
00327 1129 26 F9        BNE      SPM
00328 112B 96 8F        LDA A  DIR
00329 112D B7 A212      STA A  DIR1      INTERMEDIATE MEMORY
00330 1130 B6 A21F      LDA A  TARGET
00331 1133 81 55        CMP A  #$55      RETEST TARGET WORD
00332 1135 26 ED        BNE      SPM
00333 1137 39          RTS
00334
00335
00336
00337
00338 1138 B6 A23F      SPM1 LDA A  TARG1      TARGET WORD
00339 113B 81 55        CMP A  #$55
00340 113D 26 F9        BNE      SPM1
00341 113F 96 8F        LDA A  DIR
00342 1141 B7 A222      STA A  DIR2
00343 1144 96 A7        LDA A  XSPDCT
00344 1146 B7 A220      STA A  XSPDHI
00345 1149 96 A8        LDA A  XSPDCT+1
00346 114B B7 A221      STA A  XSPDLO
00347 114E B6 A23E      LDA A  XSPFLG
00348 1151 97 AA        STA A  XSPDFL

```



```

00349 1153 B6 A23F      LDA A   TARG1      TEST TARGET WORD AGAIN
00350 1156 81 55        CMP A   #$55
00351 1158 26 DE        BNE     SPM1
00352 115A 39          RTS
00353                  *
00354                  */GETPOS/
00355                  *THIS SUB READS THE POSITION FROM THE
00356                  *POSITION FIXING MODULE
00357 115B 9F 81      GETPOS STS      STT
00358 115D CE A211    TTT      LDX     #XYPOS+17
00359 1160 8E 0012    LDS      #YER
00360 1163 C6 12      LDA B   #$12
00361 1165 B6 A21F    TT      LDA A   TARGET
00362 1168 81 55      CMP A   #$55      INVERTED BUS
00363 116A 26 F1      BNE     TTT
00364 116C A6 00      LDA A   0,X
00365 116E 36        PSH A
00366 116F 09        DEX
00367 1170 5A        DEC B
00368 1171 27 02      BEQ     FINN
00369 1173 20 F0      BRA     TT
00370 1175 9E 81      FINN     LDS     STT
00371 1177 96 90      LDA A   MODE
00372 1179 9A 91      ORA A   MODE1      DO NOT DO /TEST/ IF SPEED=0
00373 117B 27 03      BEQ     FINN1
00374 117D BD 1430    JSR     TEST
00375 1180 39        FINN1    RTS
00376                  *
00377                  */FLAGP2/
00378                  * THIS SUB. CHECKS IF FLAG CB1 PIA2 HAS
00379                  * BEEN SET(OVERRIDE MOVE FLAG)
00380 1181 B6 A003    FLAGP2 LDA A   PIAC2      IS PUSHB. STROBE SET?
00381 1184 84 80      AND A   #$80
00382 1186 27 2F      BEQ     FP2
00383 1188 0F        SEI
00384 1189 BD 11E8    JSR     DELAY
00385 118C B6 A000    LDA A   PIAD1      ZERO TEST
00386 118F 84 F0      AND A   #$F0
00387 1191 27 20      BEQ     FP3
00388 1193 7F 007B    FLAGO    CLR     REG3
00389 1196 7F 0076    CLR     REG+1
00390 1199 7F 0071    CLR     XSTEP C      STEP COUNTER
00391 119C 7F 0072    CLR     YSTEP C
00392 119F CE 0009    LDX     #YCPOS      SET DISPLAY REGISTERS
00393 11A2 DF 75      STX     REG
00394 11A4 CE 0001    LDX     #XCPOS
00395 11A7 DF 77      STX     REG1
00396 11A9 CE 3012    LDX     #SPEEDI      SPEED TABLE POINTER
00397 11AC DF 7D      STX     REG4      FOR X AXIS
00398 11AE DF 7F      STX     REG5      FOR Y AXIS
00399 11B0 BD 1263    JSR     OM02      OVERRIDE MOVE ROUTINE
00400 11B3 B6 A002    FP3     LDA A   PIAD2      RESET FLAG
00401 11B6 0E        CLI
00402 11B7 39        FP2     RTS
00403                  *
00404                  */DISPLAY/
00405                  * THIS SUB DISPLAYS INFO IN THE INDEX
00406                  * REG. AT POSITION REG

```


PAGE 008

```

00407 11B8 DE 75   DISPLA LDX      REG
00408 11BA C6 10           LDA B   #$10
00409 11BC A6 00   DISBOT LDA A   0,X
00410 11BE 1B           ABA
00411 11BF B7 A002           STA A   PIAD2   DISPLAY
00412 11C2 84 0F           AND A   #$0F
00413 11C4 B7 A002           STA A   PIAD2   SET DISPLAY STROBE
00414 11C7 CB 10           ADD B   #$10
00415 11C9 C1 60           CMP B   #$60
00416 11CB 27 14           BEQ      DISTOP
00417 11CD C1 C0           CMP B   #$C0
00418 11CF 27 16           BEQ      DISEND
00419 11D1 C1 40           CMP B   #$40
00420 11D3 27 07           BEQ      RR2
00421 11D5 C1 A0           CMP B   #$A0
00422 11D7 27 03           BEQ      RR2
00423 11D9 08           INX
00424 11DA 20 E0           BRA      DISBOT
00425 11DC 08   RR2      INX
00426 11DD 08           INX
00427 11DE 08           INX
00428 11DF 20 DB           BRA      DISBOT
00429 11E1 DE 77   DISTOP LDX      REG1
00430 11E3 CB 10           ADD B   #$10
00431 11E5 20 D5           BRA      DISBOT
00432 11E7 39   DISEND RTS
00433 *
00434 */DELAY/
00435 * DELAY ROUTINES
00436 11E8 86 7C   DELAY LDA A   #$7C   APPROX 1msec
00437 11EA 4A   D1      DEC A
00438 11EB 81 00           CMP A   #$00
00439 11ED 26 FB           BNE      D1
00440 11EF 39           RTS
00441 *
00442 11F0 C6 FF   DELAY1 LDA B   #$FF
00443 11F2 86 FF           LDA A   #$FF
00444 11F4 4A   DD1      DEC A
00445 11F5 81 00           CMP A   #$00
00446 11F7 26 FB           BNE      DD1
00447 11F9 5A           DEC B
00448 11FA C1 00           CMP B   #$00
00449 11FC 27 04           BEQ      DD2
00450 11FE 86 FF           LDA A   #$FF
00451 1200 20 F2           BRA      DD1
00452 1202 39   DD2      RTS
00453 *
00454 *CHEX/
00455 * THIS SUB. CHANGES A DECIMAL NO. TO
00456 * HEX NO.(VALUE IS IN THE X REGISTER)
00457 1203 6F 03   CHEX  CLR      3,X
00458 1205 6F 07           CLR      7,X
00459 1207 A6 00           LDA A   0,X   HUNDREDS
00460 1209 81 00   C1    CMP A   #$00
00461 120B 27 09           BEQ      TEN
00462 120D C6 64           LDA B   #$64
00463 120F EB 03           ADD B   3,X
00464 1211 E7 03           STA B   3,X

```


00465	1213	4A		DEC A	
00466	1214	20 F3		BRA C1	
00467	1216	A6 01	TEN	LDA A 1,X	TENS
00468	1218	81 00	C2	CMP A # \$00	
00469	121A	27 09		BEQ UNITS	
00470	121C	C6 0A		LDA B # \$0A	
00471	121E	EB 03		ADD B 3,X	
00472	1220	E7 03		STA B 3,X	
00473	1222	4A		DEC A	
00474	1223	20 F3		BRA C2	
00475	1225	E6 02	UNITS	LDA B 2,X	
00476	1227	EB 03		ADD B 3,X	
00477	1229	E7 03		STA B 3,X	
00478	122B	A6 05		LDA A 5,X	FINE TENS
00479	122D	81 00	C3	CMP A # \$00	
00480	122F	27 09		BEQ FUNITS	
00481	1231	C6 0A		LDA B # \$0A	
00482	1233	EB 07		ADD B 7,X	
00483	1235	E7 07		STA B 7,X	
00484	1237	4A		DEC A	
00485	1238	20 F3		BRA C3	
00486	123A	E6 06	FUNITS	LDA B 6,X	
00487	123C	EB 07		ADD B 7,X	
00488	123E	E7 07		STA B 7,X	
00489	1240	39		RTS	
00490			*		
00491			*/CON/		
00492			*THIS SUB CONVERTS A HEX NO.(IN X REG) TO DECIMAL		
00493	1241	86 00	CON	LDA A # \$00	
00494	1243	A7 00		STA A 0,X	CLEAR REG'S
00495	1245	A7 01		STA A 1,X	
00496	1247	A7 02		STA A 2,X	
00497	1249	A6 03		LDA A 3,X	LOAD VALUE
00498	124B	C6 0A		LDA B # \$0A	
00499	124D	10	TENC	SBA	TENS
00500	124E	81 00		CMP A # \$00	
00501	1250	2B 04		BMI UN	
00502	1252	6C 01		INC 1,X	
00503	1254	20 F7		BRA TENC	
00504	1256	1B	UN	ABA	UNITS
00505	1257	A7 02		STA A 2,X	
00506	1259	A6 01		LDA A 1,X	
00507	125B	10		SBA	
00508	125C	2B 04		BMI CON1	
00509	125E	A7 01		STA A 1,X	
00510	1260	6C 00		INC 0,X	
00511	1262	39	CON1	RTS	
00512			*		
00513			* OVERRIDE MOVE ROUTINE		
00514			*		
00515	1263	97 6A	OM02	STA A T5	DIRECTION REMEMBER
00516	1265	84 30		AND A # \$30	X DIRECTION TEST
00517	1267	27 2C		BEQ M01A	NEITHER FWD OR REV
00518	1269	7D 0090		TST MODE	IF SET DON'T CHANGE DIRECTION
00519	126C	26 2D		BNE M01	
00520	126E	81 10		CMP A # \$10	
00521	1270	26 06		BNE M02	
00522	1272	86 11		LDA A # \$11	FORWARD

00523	1274	97	90		STA A	MODE	STORE X DIRECTION
00524	1276	20	04		BRA	M03	
00525	1278	86	22	M02	LDA A	#\$22	REVERSE
00526	127A	97	90		STA A	MODE	
00527	127C	96	04	M03	LDA A	XCPOS+3	ANY X COARSE POSITION FIX
00528	127E	81	AA		CMP A	#\$AA	
00529	1280	27	06		BEQ	M05	NON
00530	1282	86	10		LDA A	#\$10	FAST SPEED
00531	1284	97	6F		STA A	XSTEP	INCREMENTED STEPS COUNTER
00532	1286	20	04		BRA	M04	
00533	1288	86	01	M05	LDA A	#\$01	SLOW SPEED
00534	128A	97	6F		STA A	XSTEP	INCREMENTED STEPS COUNTER
00535	128C	86	FF	M04	LDA A	#\$FF	
00536	128E	97	73		STA A	XINC	INC/DEC FLAG
00537	1290	BD	1878		JSR	DIRSET	SET DIRECTION LED, RELAY
00538	1293	20	06		BRA	M01	
00539	1295	7F	0073	M01A	CLR	XINC	RESET INC FLAG TO DEC
00540	1298	7F	006F		CLR	XSTEP	
00541	129B	96	6A	M01	LDA A	T5	RELOAD DIRECTION
00542	129D	84	C0		AND A	#\$C0	Y DIRECTION TEST
00543	129F	27	31		BEQ	M01B	NEITHER LEFT OR RIGHT
00544	12A1	7D	00A9		TST	XSPMK2	MK2 SP.CONT.?
00545	12A4	26	2C		BNE	M01B	YES, THEN X ONLY
00546	12A6	7D	0091		TST	MODE1	IF SET DO NOT CHANGE DIRECTIO
00547	12A9	26	2D		BNE	M05	
00548	12AB	81	40		CMP A	#\$40	
00549	12AD	26	06		BNE	M022	
00550	12AF	86	44		LDA A	#\$44	LEFT
00551	12B1	97	91		STA A	MODE1	Y DIRECTION STORE
00552	12B3	20	04		BRA	M033	
00553	12B5	86	88	M022	LDA A	#\$88	RIGHT
00554	12B7	97	91		STA A	MODE1	
00555	12B9	96	0C	M033	LDA A	YCPOS+3	ANY Y POS. FIX
00556	12BB	81	AA		CMP A	#\$AA	
00557	12BD	27	06		BEQ	M055	NON
00558	12BF	86	10		LDA A	#\$10	FAST SPEED
00559	12C1	97	70		STA A	YSTEP	INC. STEPS COUNTER
00560	12C3	20	04		BRA	M044	
00561	12C5	86	01	M055	LDA A	#\$01	SLOW SPEED
00562	12C7	97	70		STA A	YSTEP	
00563	12C9	86	FF	M044	LDA A	#\$FF	
00564	12CB	97	74		STA A	YINC	Y INC FLAG
00565	12CD	BD	1878		JSR	DIRSET	SET LED AND RELAY
00566	12D0	20	06		BRA	M05	
00567	12D2	7F	0074	M01B	CLR	YINC	
00568	12D5	7F	0070		CLR	YSTEP	ZERO Y SPEED
00569	12D8	BD	1124	M05	JSR	SPM	INFORM POS FIX MODULE
00570	12DB	86	15		LDA A	#\$15	
00571	12DD	B7	A099		STA A	PTMC2	CR #1 MAY BE WRITTEN
00572	12E0	86	96		LDA A	#\$96	
00573	12E2	B7	A098		STA A	PTMC13	ALL TIMERS OPERATE
00574	12E5	BD	115B	OM01	JSR	GETPOS	GET POSITION FROM BC2
00575	12E8	BD	11B8		JSR	DISPLAY	DISPLAY POSITION
00576	12EB	BD	1493		JSR	ESXY	CHECK BRIDGE LIMITS
00577	12EE	7D	00A9		TST	XSPMK2	MK2 SPEED CONT. FLAG
00578	12F1	27	03		BEQ	M054	
00579	12F3	7E	1344		JMP	SPDMK2	SET MK2 SPEED
00580	12F6	B6	A099	M054	LDA A	PTMC2	READ STATUS REG FOR TO

00581	12F9	84	02		AND A	#\$02	
00582	12FB	27	03		BEQ	MOS1	TIMER #2 TO NOT SET
00583	12FD	BD	1387		JSR	IDSP	INCREMENT/DECREMENT SPEED
00584	1300	7D	0090	MOS1	TST	MODE	ANY X MOVEMENT
00585	1303	26	07		BNE	MOS2	
00586	1305	7D	0091		TST	MODE1	ANY Y MOVEMENT
00587	1308	26	02		BNE	MOS2	
00588	130A	20	1F		BRA	MOS3	HALT MOTOR
00589	130C	7D	0093	MOS2	TST	FLA	FF-INT;00-EXT
00590	130F	27	0C		BEQ	MOS2A	
00591	1311	B6	A000		LDA A	PIAD1	TEST FOR MOVEMENT CHANGE
00592	1314	84	F0		AND A	#\$F0	
00593	1316	91	6A		CMP A	T5	
00594	1318	27	CB		BEQ	OM01	
00595	131A	7E	1263		JMP	OM02	
00596	131D	0C		MOS2A	CLC		
00597	131E	BD	C5F9		JSR	KBDCHK	STOP WHEN SPACE IS STRUCK
00598	1321	24	C2		BCC	OM01	NOT STRUCK
00599	1323	81	20		CMP A	#\$20	SPACE
00600	1325	26	BE		BNE	OM01	
00601	1327	4F			CLR A		
00602	1328	7E	1263		JMP	OM02	
00603	132B	86	15	MOS3	LDA A	#\$15	HALT
00604	132D	B7	A099		STA A	PTMC2	CR #1 MAY BE WRITTEN
00605	1330	86	97		LDA A	#\$97	
00606	1332	B7	A098		STA A	PTMC13	ALL TIMERS PRESET
00607	1335	86	C1		LDA A	#\$C1	
00608	1337	B7	A004		STA A	PIAD3	MODE LED'S
00609	133A	4F			CLR A		
00610	133B	B7	A006		STA A	PIAD4	RESET MOTOR CONTROL
00611	133E	97	94		STA A	FLAG	
00612	1340	BD	1A65		JSR	ER2	TEST FOR INCORRECT FIXES
00613	1343	39			RTS		
00614				*			
00615				*MK2 SPEED CONTROOLLER SETUP			
00616				*			
00617	1344	96	6F	SPDMK2	LDA A	XSTEP	MK1 STEP INC. REMEMBER
00618	1346	81	01		CMP A	#\$01	SLOW SPEED
00619	1348	27	11		BEQ	SSPEED	
00620	134A	96	73		LDA A	XINC	X INC/DEC FLAG
00621	134C	27	1A		BEQ	ZSPEED	ZERO SPEED
00622	134E	86	01		LDA A	#\$01	FAST SPEED
00623	1350	97	A7		STA A	XSPDCT	HI BYTE
00624	1352	86	4D		LDA A	#\$4D	
00625	1354	97	A8		STA A	XSPDCT+1	LO BYTE
00626	1356	BD	1138		JSR	SPM1	INFORM SPEED MODULE
00627	1359	20	29		BRA	MOS5	
00628	135B	86	07	SSPEED	LDA A	#\$07	SLOW SPEED
00629	135D	97	A7		STA A	XSPDCT	
00630	135F	86	D0		LDA A	#\$D0	
00631	1361	97	A8		STA A	XSPDCT+1	
00632	1363	BD	1138		JSR	SPM1	
00633	1366	20	1C		BRA	MOS5	
00634	1368	86	00	ZSPEED	LDA A	#\$00	ZERO SPEED
00635	136A	97	A7		STA A	XSPDCT	
00636	136C	97	A8		STA A	XSPDCT+1	
00637	136E	BD	1138		JSR	SPM1	
00638	1371	96	AA		LDA A	XSPDFL	MOTOR AT ZERO SPD. YET


```

00639 1373 26 0F          BNE      MOS5      NO
00640 1375 7F 0090        CLR      MODE      YES
00641 1378 7F 008F        CLR      DIR
00642 137B BD 1878        JSR      DIRSET
00643 137E BD 1138        JSR      SPM1
00644 1381 BD 1124        JSR      SPM
00645 1384 7E 1300 MOS5   JMP      MOS1
00646                      *
00647                      */IDSP/
00648                      *THIS SUB INCREMENTS OR DECREMENTS THE SPEED BY
00649                      *CHANGING THE FREQUENCY IN TIMERS #1 OR #3 AT THE
00650                      *RATE OF TIMER #2
00651                      *
00652 1387 B6 A09C IDSP    LDA A   PT2MSB   RESET TO
00653 138A B6 A09D        LDA A   PT2LSB
00654 138D 96 90          LDA A   MODE      X MOVEMENT FLAG
00655 138F 27 1A          BEQ      IDSP2     NO X MOVEMENT
00656 1391 96 73          LDA A   XINC      X INC DEC FLAG
00657 1393 27 05          BEQ      IDSP3
00658 1395 BD 13CA        JSR      XISP      INCREMENT SPEED
00659 1398 20 11          BRA      IDSP2
00660 139A BD 13FD IDSP3  JSR      XDSP      DECREMENT SPEED
00661 139D 7D 0071        TST      XSTEP C   X STEP COUNT
00662 13A0 26 09          BNE      IDSP2
00663 13A2 7F 0090        CLR      MODE
00664 13A5 BD 1878        JSR      DIRSET     SET NEW DIRECTION
00665 13A8 BD 1124        JSR      SPM        INFORM POS FIX MODULE
00666 13AB 96 91 IDSP2    LDA A   MODE1
00667 13AD 27 1A          BEQ      IDSP1     NO Y MOVEMENT
00668 13AF 96 74          LDA A   YINC      Y INC/DEC FLAG
00669 13B1 27 05          BEQ      IDSP4
00670 13B3 BD 13E4        JSR      YISP      INC SPEED
00671 13B6 20 11          BRA      IDSP1
00672 13B8 BD 1417 IDSP4  JSR      YDSP      DEC SPEED
00673 13BB 7D 0072        TST      YSTEP C   Y STEP COUNT
00674 13BE 26 09          BNE      IDSP1
00675 13C0 7F 0091        CLR      MODE1
00676 13C3 BD 1878        JSR      DIRSET     STOP Y MOVEMENT
00677 13C6 BD 1124        JSR      SPM        INFORM BC2
00678 13C9 39 IDSP1      RTS
00679                      *
00680                      */*ISP/
00681                      *THIS SUB INCREMENTS THE SPEED
00682                      *
00683 13CA 96 71 XISP      LDA A   XSTEP C   X INC STEP COUNTER
00684 13CC 91 6F          CMP A   XSTEP     PRESET VALUE FOR X SPEED
00685 13CE 27 2C          BEQ      ISP1
00686 13D0 DE 7D          LDX      REG4      X SPEED STORAGE
00687 13D2 A6 00          LDA A   0,X
00688 13D4 B7 A09A        STA A   PT1MSB
00689 13D7 86 27          LDA A   #27
00690 13D9 B7 A09B        STA A   PT1LSB     PULSE WIDTH
00691 13DC 08            INX
                                NEXT VALUE FROM SPEED TABLE
00692 13DD DF 7D          STX      REG4
00693 13DF 7C 0071        INC      XSTEP C   X INC COUNTER
00694 13E2 20 18          BRA      ISP1
00695 13E4 96 72 YISP      LDA A   YSTEP C   Y INC COUNTER
00696 13E6 91 70          CMP A   YSTEP     Y PRESET VALUE

```



```

00697 13E8 27 12      BEQ      ISP1
00698 13EA DE 7F      LDX      REG5
00699 13EC A6 00      LDA A    0,X
00700 13EE B7 A09E     STA A    PT3MSB
00701 13F1 86 27      LDA A    #27      PULSE WIDTH
00702 13F3 B7 A09F     STA A    PT3LSB
00703 13F6 08         INX              NEXT VALUE FROM SPEED TABLE
00704 13F7 DF 7F      STX      REG5
00705 13F9 7C 0072     INC      YSTEPC
00706 13FC 39         ISP1    RTS
00707                *
00708                */*DSP/
00709                *THIS SUB DECREMENTS THE SPEED
00710                *
00711 13FD 96 71      XDSP    LDA A    XSTEP C    X STEP COUNTER
00712 13FF 91 6F      CMP A    XSTEP    PRESET VALUE
00713 1401 27 2C      BEQ      DSP1
00714 1403 DE 7D      LDX      REG4
00715 1405 09         DEX              NEXT VALUE FROM SPEED TABLE
00716 1406 7A 0071     DEC      XSTEP C    DEC X STEP COUNTER
00717 1409 A6 00      LDA A    0,X
00718 140B B7 A09A     STA A    PT1MSB
00719 140E 86 27      LDA A    #27      PULSE WIDTH
00720 1410 B7 A09B     STA A    PT1LSB
00721 1413 DF 7D      STX      REG4
00722 1415 20 18      BRA      DSP1
00723 1417 96 72      YDSP    LDA A    YSTEP C    Y PRESET VALUE
00724 1419 91 70      CMP A    YSTEP
00725 141B 27 12      BEQ      DSP1
00726 141D DE 7F      LDX      REG5
00727 141F 09         DEX
00728 1420 7A 0072     DEC      YSTEP C
00729 1423 A6 00      LDA A    0,X
00730 1425 B7 A09E     STA A    PT3MSB
00731 1428 86 27      LDA A    #27      PULSE WIDTH
00732 142A B7 A09F     STA A    PT3LSB
00733 142D DF 7F      STX      REG5
00734 142F 39         DSP1    RTS
00735                *
00736                *
00737                */TEST/
00738                *THIS SUB TESTS TO SEE THAT X or Y COARSE
00739                *POSITION DOES NOT CHANGE BY MORE THAN 1
00740 1430 CE 0001     TEST    LDX      #XCPOS
00741 1433 E6 03      LDA B    3,X      XCOARSE POSITION
00742 1435 C1 AA      CMP B    #AA
00743 1437 27 29      BEQ      TE1      NO COARSE REGISTERED YET
00744 1439 7D 0098     TST      FLAG4    FIRST ENTRY
00745 143C 27 0B      BEQ      TE2
00746 143E BD 1AED     JSR      ER4      (IS FIRST FIX OK?)
00747 1441 73 0098     COM      FLAG4
00748 1444 96 04      LDA A    XCPOS+3
00749 1446 97 63      STA A    XTEST
00750 1448 39         RTS
00751 1449 96 63      TE2    LDA A    XTEST
00752 144B 10         SBA
00753 144C 2B 0E      BMI      TE3
00754 144E 81 01      CMP A    #01

```


00755	1450	2F 0E		BLE	TE4	
00756	1452	D7 63	TE5	STA B	XTEST	
00757	1454	86 AA		LDA A	#\$AA	
00758	1456	97 6E		STA A	T9	
00759	1458	7E 1A38		JMP	ER1	JUMP TO ALARM
00760	145B	39	TE6	RTS		
00761	145C	81 FF	TE3	CMP A	#\$FF	NEGATIVE
00762	145E	26 F2		BNE	TE5	
00763	1460	D7 63	TE4	STA B	XTEST	
00764	1462	E6 0B	TE1	LDA B	11,X	YCOARSE POSITION
00765	1464	C1 AA		CMP B	#\$AA	
00766	1466	27 F3		BEQ	TE6	NO COARSE POSITION BEEN REGIS
00767	1468	7D 0099		TST	FLAG5	
00768	146B	27 0B		BEQ	TE7	
00769	146D	BD 1AED		JSR	ER4	
00770	1470	73 0099		COM	FLAG5	
00771	1473	96 0C		LDA A	YCPOS+3	
00772	1475	97 64		STA A	YTEST	
00773	1477	39		RTS		
00774	1478	96 64	TE7	LDA A	YTEST	
00775	147A	10		SBA		
00776	147B	2B 06		BMI	TE8	
00777	147D	81 01		CMP A	#\$01	
00778	147F	2F 06		BLE	TE9	
00779	1481	20 07		BRA	TE10	
00780	1483	81 FF	TE8	CMP A	#\$FF	
00781	1485	26 03		BNE	TE10	
00782	1487	D7 64	TE9	STA B	YTEST	
00783	1489	39		RTS		
00784	148A	D7 64	TE10	STA B	YTEST	
00785	148C	86 55		LDA A	#\$55	
00786	148E	97 6E		STA A	T9	
00787	1490	7E 1A38		JMP	ER1	
00788			*			
00789			* /ESXY/			
00790			* THIS SUB TESTS AND SLOWS THE BRIDGE			
00791			* IF IT GETS CLOSER THAN 1 METER FROM			
00792			* THE ENDSTOPS AS INDECATED BY THE			
00793			* I.D. BOARD(OVERRIDE MODE ONLY)			
00794	1493	96 8F	ESXY	LDA A	DIR	TEST DIRECTION
00795	1495	27 4A		BEQ	ESO	
00796	1497	84 F0		AND A	#\$F0	MASK
00797	1499	81 20		CMP A	#\$20	
00798	149B	27 04		BEQ	EST2	TEST X LOWER
00799	149D	2D 43		BLT	EST3	TEST X HEIGHER
00800	149F	20 36		BRA	ESOK	
00801	14A1	96 04	EST2	LDA A	XCPOS+3	X COARSE POS
00802	14A3	81 AA		CMP A	#\$AA	NO POS REGISTERED
00803	14A5	27 30		BEQ	ESOK	
00804	14A7	F6 A0B0		LDA B	ESXCL	X LOW COARSE ENDSTOP
00805	14AA	10		SBA		POS-XCOARSE LOW
00806	14AB	2D 5F		BLT	XNOGO	
00807	14AD	81 01		CMP A	#\$01	
00808	14AF	27 0C		BEQ	F1	
00809	14B1	22 24		BHI	ESOK	>1
00810	14B3	96 08		LDA A	XFPOS+3	WITHIN 1 METER FROM ENDSTOP
00811	14B5	F6 A0B1		LDA B	ESXFL	
00812	14B8	10		SBA		

00813	14B9	23	51		BLS	XNOGO	
00814	14BB	20	0C		BRA	XESNOK	
00815	14BD	96	08	F1	LDA A	XFPOS+3	=1....FINE POSITION
00816	14BF	8B	64		ADD A	#\$64	100
00817	14C1	F6	A0B1		LDA B	ESXFL	LOWER X FINE
00818	14C4	10			SBA		
00819	14C5	81	64		CMP A	#\$64	
00820	14C7	22	0E		BHI	ESOK	>100cm FROM ENDSTOP
00821	14C9	96	6F	XESNOK	LDA A	XSTEP	
00822	14CB	27	04		BEQ	XXSTOP	IF ZERO DONT CHANGE
00823	14CD	86	01		LDA A	#\$01	SLOW SPEED
00824	14CF	97	6F		STA A	XSTEP	
00825	14D1	BD	1626	XXSTOP	JSR	XS5	SET INC/DEC FLAG
00826	14D4	BD	1B6B	ESNOK	JSR	ER5	WITHIN 1 METRE FROM LIMIT
00827	14D7	96	8F	ESOK	LDA A	DIR	
00828	14D9	84	0F		AND A	#\$0F	
00829	14DB	81	04		CMP A	#\$04	
00830	14DD	27	5F		BEQ	EST5	
00831	14DF	22	33		BHI	EST4	
00832	14E1	39		ESO	RTS		
00833	14E2	B6	A0B2	EST3	LDA A	ESXCH	HI X COARSE ENDSTOP
00834	14E5	D6	04		LDA B	XCPOS+3	X COARSE POSITION
00835	14E7	C1	AA		CMP B	#\$AA	
00836	14E9	27	EC		BEQ	ESOK	
00837	14EB	10			SBA		HI X ENDSTOP-POS
00838	14EC	2D	1E		BLT	XNOGO	
00839	14EE	81	01		CMP A	#\$01	
00840	14F0	27	0C		BEQ	F2	
00841	14F2	22	E3		BHI	ESOK	>1
00842	14F4	B6	A0B3		LDA A	ESXFH	WITHIN 1 METER FROM ENDSTOP
00843	14F7	D6	08		LDA B	XFPOS+3	
00844	14F9	10			SBA		
00845	14FA	23	10		BLS	XNOGO	
00846	14FC	20	CB		BRA	XESNOK	
00847	14FE	B6	A0B3	F2	LDA A	ESXFH	=1....HI X FINE ENDSTOP
00848	1501	8B	64		ADD A	#\$64	
00849	1503	D6	08		LDA B	XFPOS+3	
00850	1505	10			SBA		
00851	1506	81	64		CMP A	#\$64	
00852	1508	22	CD		BHI	ESOK	OK
00853	150A	20	BD		BRA	XESNOK	
00854	150C	7F	006F	XNOGO	CLR	XSTEP	
00855	150F	7F	0073		CLR	XINC	SET FLAG TO DEC
00856	1512	20	C0		BRA	ESNOK	
00857	1514	96	0C	EST4	LDA A	YCPOS+3	Y COARSE POSITION
00858	1516	81	AA		CMP A	#\$AA	
00859	1518	27	C7		BEQ	ESO	
00860	151A	F6	A0B4		LDA B	ESYCL	LOW Y COARSE ENDSTOP
00861	151D	10			SBA		POS- YCOARSE ES
00862	151E	2D	55		BLT	YNOGO	
00863	1520	81	01		CMP A	#\$01	
00864	1522	22	BD		BHI	ESO	
00865	1524	27	0A		BEQ	F3	
00866	1526	96	10		LDA A	YFPOS+3	WITHIN 1M FROM ENDSTOP
00867	1528	F6	A0B5		LDA B	ESYFL	
00868	152B	10			SBA		
00869	152C	23	47		BLS	YNOGO	
00870	152E	20	36		BRA	YESNOK	


```

00871 1530 96 10 F3 LDA A YFPOS+3 Y FINE POS
00872 1532 8B 64 ADD A #$64
00873 1534 F6 A0B5 LDA B ESYFL LOW X FINE ES
00874 1537 10 SBA
00875 1538 81 64 CMP A #$64
00876 153A 22 A5 BHI ESO
00877 153C 20 28 BRA YESNOK
00878 153E B6 A0B6 EST5 LDA A ESYCH HI Y COARSE ES
00879 1541 D6 0C LDA B YCPOS+3 Y COARSE
00880 1543 C1 AA CMP B #$AA
00881 1545 27 9A BEQ ESO
00882 1547 10 SBA
00883 1548 2D 2B BLT YNOGO
00884 154A 81 01 CMP A #$01
00885 154C 22 93 BHI ESO
00886 154E 27 0A BEQ F4
00887 1550 B6 A0B7 LDA A ESYFH
00888 1553 D6 10 LDA B YFPOS+3
00889 1555 10 SBA
00890 1556 23 1D BLS YNOGO
00891 1558 20 0C BRA YESNOK
00892 155A B6 A0B7 F4 LDA A ESYFH Y FINE HI ES
00893 155D 8B 64 ADD A #$64
00894 155F D6 10 LDA B YFPOS+3
00895 1561 10 SBA
00896 1562 81 64 CMP A #$64
00897 1564 22 0E BHI ES02
00898 1566 96 70 YESNOK LDA A YSTEP
00899 1568 27 04 BEQ YYSTOP IF ZERO DONT CHANGE
00900 156A 86 01 LDA A #$01 Y SLOW SPEED
00901 156C 97 70 STA A YSTEP
00902 156E BD 16C0 YYSTOP JSR YS5
00903 1571 BD 1B6B ES01 JSR ER5
00904 1574 39 ES02 RTS
00905 1575 7F 0070 YNOGO CLR YSTEP
00906 1578 7F 0074 CLR YINC SET FLAG TO DEC
00907 157B 20 F4 BRA ES01
00908 *
00909 * /ESX/
00910 * THIS SUB TESTS THE X AXIS ENSTOPS TO
00911 * THE GOTO POSITION(NON-OVERRIDE MODE ONLY)
00912 157D 96 90 ESX LDA A MODE TEST DIRECTION
00913 157F 84 F0 AND A #$F0 MASK
00914 1581 81 20 CMP A #$20
00915 1583 27 03 BEQ ESX2 TEST X LOW
00916 1585 2D 17 BLT ESX3 TEST X HI
00917 1587 39 RTS
00918 1588 96 26 ESX2 LDA A XCGOTO+3
00919 158A F6 A0B0 LDA B ESXCL X COARSE LOW ENDSTOP
00920 158D 10 SBA POS-COARSE X LOW ES
00921 158E 22 0D BHI ESXOK
00922 1590 2D 08 BLT ESXNOK
00923 1592 96 2A LDA A XFGOTO+3 FINE GOTO POS
00924 1594 F6 A0B1 LDA B ESXFL FINE X LOW ES
00925 1597 10 SBA
00926 1598 22 03 BHI ESXOK
00927 159A BD 1B82 ESXNOK JSR ER6
00928 159D 39 ESXOK RTS

```



```

00929 159E B6 A0B2 ESX3 LDA A ESXCH X COARSE HI ES
00930 15A1 D6 26 LDA B XCGOTO+3
00931 15A3 10 SBA
00932 15A4 22 F7 BHI ESXOK
00933 15A6 2D F2 BLT ESXNOK
00934 15A8 B6 A0B3 LDA A ESXFH X FINE ES
00935 15AB D6 2A LDA B XFGOTO+3
00936 15AD 10 SBA
00937 15AE 22 ED BHI ESXOK
00938 15B0 20 E8 BRA ESXNOK
00939 *
00940 * /ESY/
00941 * THIS SUB COMPARES THE Y ENDSTOPS TO
00942 * THE GOTO POSITION(NON-OVERRIDE MODE ONLY)
00943 15B2 96 91 ESY LDA A MODE1 TEST DIRECTION
00944 15B4 84 0F AND A #$0F
00945 15B6 81 04 CMP A #$04
00946 15B8 27 19 BEQ ESY5
00947 15BA 22 01 BHI ESY4 TEST Y HI
00948 15BC 39 RTS
00949 15BD 96 2E ESY4 LDA A YCGOTO+3
00950 15BF F6 A0B4 LDA B ESYCL Y COARSE LOWER ES
00951 15C2 10 SBA
00952 15C3 22 0D BHI ESYOK OK
00953 15C5 2D 08 BLT ESYNOK INCORRECT
00954 15C7 96 32 LDA A YFGOTO+3 FINE
00955 15C9 F6 A0B5 LDA B ESYFL
00956 15CC 10 SBA
00957 15CD 22 03 BHI ESYOK
00958 15CF BD 1B82 ESYNOK JSR ER6
00959 15D2 39 ESYOK RTS
00960 15D3 B6 A0B6 ESY5 LDA A ESYCH Y COARSE HEIGHER ENDSTOP
00961 15D6 D6 2E LDA B YCGOTO+3
00962 15D8 10 SBA
00963 15D9 22 F7 BHI ESYOK
00964 15DB 2D F2 BLT ESYNOK
00965 15DD B6 A0B7 LDA A ESYFH Y FINE
00966 15E0 D6 32 LDA B YFGOTO+3
00967 15E2 10 SBA
00968 15E3 22 ED BHI ESYOK
00969 15E5 20 E8 BRA ESYNOK
00970 *
00971 *XSPEED/
00972 *THIS SUB SELECTS THE MOTOR SPEED
00973 *IN THE X-AXIS (AND SPEED INCREMENTS)
00974 15E7 7D 00A9 XSPEED TST XSPMK2 MK1 or MK2 SP. CONTROL
00975 15EA 27 03 BEQ XS8 NO
00976 15EC 7E 1639 JMP SPE YES
00977 15EF 96 A3 XS8 LDA A D06 EXPONENT
00978 15F1 97 69 STA A T4
00979 15F3 81 80 CMP A #$80 ZERO TEST
00980 15F5 27 2B BEQ XS2
00981 15F7 81 0F CMP A #$0F TEST FOR NEG EXPONENT
00982 15F9 25 15 BCS XS1
00983 15FB 40 NEG A 2'S COMP
00984 15FC 81 05 CMP A #$05
00985 15FE 2E 22 BGT XS2 ZERO SPEED
00986 1600 81 01 CMP A #$01

```


00987	1602	2B	06		BMI	XS3	MED/SLOW SPEED
00988	1604	86	01		LDA A	#\$01	SLOW SPEED
00989	1606	97	6F		STA A	XSTEP	
00990	1608	20	1C		BRA	XS5	
00991	160A	86	06	XS3	LDA A	#\$06	MED/SLOW SPEED
00992	160C	97	6F		STA A	XSTEP	
00993	160E	20	16		BRA	XS5	
00994	1610	81	01	XS1	CMP A	#\$01	
00995	1612	2B	F6		BMI	XS3	
00996	1614	27	06		BEQ	XS4	
00997	1616	86	10		LDA A	#\$10	FAST SPEED
00998	1618	97	6F		STA A	XSTEP	
00999	161A	20	0A		BRA	XS5	
01000	161C	86	0B	XS4	LDA A	#\$0B	
01001	161E	97	6F		STA A	XSTEP	
01002	1620	20	04		BRA	XS5	
01003	1622	86	00	XS2	LDA A	#\$00	
01004	1624	97	6F		STA A	XSTEP	STOP MOTOR
01005	1626	D6	71	XS5	LDA B	XSTEP	SPEED
01006	1628	96	6F		LDA A	XSTEP	NEW VALUE OF SPEED
01007	162A	11			CBA		
01008	162B	27	0B		BEQ	XS6	NO CHANGE
01009	162D	2A	05		BPL	XS7	
01010	162F	7F	0073		CLR	XINC	SET FLAG TO DECREMENT
01011	1632	20	04		BRA	XS6	
01012	1634	86	FF	XS7	LDA A	#\$FF	SET FLAG TO INCREMENT
01013	1636	97	73		STA A	XINC	
01014	1638	39		XS6	RTS		
01015				*			
01016	1639	96	A3	SPE	LDA A	D06	MK2 SPEED SELECT
01017	163B	97	69		STA A	T4	
01018	163D	81	80		CMP A	#\$80	ZERO TEST
01019	163F	27	25		BEQ	SPE1	(0 SPEED)
01020	1641	81	0F		CMP A	#\$0F	NEG. EXP. ?
01021	1643	25	2A		BCS	SPE2	NO (FAST SPEED)
01022	1645	40			NEG A		
01023	1646	81	05		CMP A	#\$05	
01024	1648	2E	1C		BGT	SPE1	(0 SPEED)
01025	164A	81	02		CMP A	#\$02	MORE THAN 25cm FROM STOP
01026	164C	2B	21		BMI	SPE2	YES (FAST SPEED)
01027	164E	96	A7		LDA A	XSPDCT	SLOW SPEED
01028	1650	81	07		CMP A	#\$07	IS ENTERED SPEED SLOWER?
01029	1652	22	31		BHI	SPE4	YES
01030	1654	26	06		BNE	SPE5	NO
01031	1656	96	A8		LDA A	XSPDCT+1	EQUAL
01032	1658	81	D0		CMP A	#\$D0	
01033	165A	22	29		BHI	SPE4	
01034	165C	86	07	SPE5	LDA A	#\$07	SLOW SPEED
01035	165E	97	A7		STA A	XSPDCT	-----
01036	1660	86	D0		LDA A	#\$D0	
01037	1662	97	A8		STA A	XSPDCT+1	
01038	1664	20	1F		BRA	SPE4	
01039	1666	4F		SPE1	CLR A		ZERO SPEED
01040	1667	97	A7		STA A	XSPDCT	-----
01041	1669	97	A8		STA A	XSPDCT+1	
01042	166B	97	90		STA A	MODE	
01043	166D	20	16		BRA	SPE4	
01044	166F	96	A7	SPE2	LDA A	XSPDCT	FAST SPEED

01045	1671	81	01		CMP A	#\$01	ENTERED SPEED SLOWER ?
01046	1673	22	10		BHI	SPE4	YES
01047	1675	26	06		BNE	SPE6	NO
01048	1677	96	A8		LDA A	XSPDCT+1	EQUAL
01049	1679	81	4D		CMP A	#\$4D	TEST LSB
01050	167B	22	08		BHI	SPE4	
01051	167D	86	01	SPE6	LDA A	#\$01	FAST SPEED
01052	167F	97	A7		STA A	XSPDCT	-----
01053	1681	86	4D		LDA A	#\$4D	
01054	1683	97	A8		STA A	XSPDCT+1	
01055	1685	BD	1138	SPE4	JSR	SPM1	
01056	1688	39			RTS		
01057					*		
01058					*/YSPEED/		
01059					*THIS SUB SELECTS THE MOTOR SPEED		
01060					*AND SPEED INCREMENTS IN THE Y-AXIS		
01061	1689	96	A3		YSPEED LDA A	D06	EXPONENT
01062	168B	97	6A		STA A	T5	
01063	168D	81	80		CMP A	#\$80	ZERO TEST
01064	168F	27	2B		BEQ	YS2	
01065	1691	81	0F		CMP A	#\$0F	TEST FOR NEG EXP
01066	1693	25	15		BCS	YS1	
01067	1695	40			NEG A		2'S COMP
01068	1696	81	05		CMP A	#\$05	
01069	1698	2E	22		BGT	YS2	ZERO SPEED
01070	169A	81	01		CMP A	#\$01	
01071	169C	2B	06		BMI	YS3	MED/SLOW SPEED
01072	169E	86	01		LDA A	#\$01	SLOW SPEED
01073	16A0	97	70		STA A	YSTEP	
01074	16A2	20	1C		BRA	YS5	
01075	16A4	86	06	YS3	LDA A	#\$06	MED/SLOW SPEED
01076	16A6	97	70		STA A	YSTEP	
01077	16A8	20	16		BRA	YS5	
01078	16AA	81	01	YS1	CMP A	#\$01	
01079	16AC	2B	F6		BMI	YS3	
01080	16AE	27	06		BEQ	YS4	
01081	16B0	86	10		LDA A	#\$10	FAST SPEED
01082	16B2	97	70		STA A	YSTEP	
01083	16B4	20	0A		BRA	YS5	
01084	16B6	86	0B	YS4	LDA A	#\$0B	MED SPEED
01085	16B8	97	70		STA A	YSTEP	
01086	16BA	20	04		BRA	YS5	
01087	16BC	86	00	YS2	LDA A	#\$00	STOP Y MOTOR
01088	16BE	97	70		STA A	YSTEP	
01089	16C0	D6	72	YS5	LDA B	YSTEP	CURRENT SPEED
01090	16C2	96	70		LDA A	YSTEP	SPEED, NEW VALUE
01091	16C4	11			CBA		
01092	16C5	27	0B		BEQ	YS6	NO CHANGE
01093	16C7	2A	05		BPL	YS7	
01094	16C9	7F	0074		CLR	YINC	SET FLAG TO DECREMENT
01095	16CC	20	04		BRA	YS6	
01096	16CE	86	FF	YS7	LDA A	#\$FF	
01097	16D0	97	74		STA A	YINC	SET FLAG TO INCREMENT
01098	16D2	39		YS6	RTS		
01099					*		
01100					*/MOVE/		
01101					*THIS SUB CONTROLES THE NON-OVERRIDE		
01102					*MOVEMENT OF THE BRIDGE		

01103	16D3	BD	1878	MOVE	JSR	DIRSET	SELECT DIRECTIONS
01104	16D6	B6	01		LDA A	#\$01	
01105	16D8	B7	A099		STA A	PTMC2	CR#1 CAN BE WRITTEN
01106	16DB	B6	96		LDA A	#\$96	
01107	16DD	B7	A098		STA A	PTMC13	START TIMERS
01108	16E0	BD	1124		JSR	SPM	INFORM POS FIX MODULE
01109	16E3	7D	00A9		TST	XSPMK2	MK2 SP.CONT. PRESENT ?
01110	16E6	27	05		BEQ	XMOVE2	NO
01111	16E8	BD	1138		JSR	SPM1	INFORM CONTROLLER
01112	16EB	20	05		BRA	XMOVE	
01113	16ED	7D	0090	XMOVE2	TST	MODE	ANY X MOVEMENT
01114	16F0	27	58		BEQ	YMOVE	
01115	16F2	B6	A001	XMOVE	LDA A	PIAC1	KEYBOARD
01116	16F5	4D			TST A		IRQ FLAG
01117	16F6	2A	03		BPL	XMOVE1	NO
01118	16F8	7E	27C8		JMP	AC	ABORT(STOP MOTOR)
01119	16FB	B6	A0A0	XMOVE1	LDA A	MODEMC	
01120	16FE	47			ASR A		
01121	16FF	24	03		BCC	XMOVE3	
01122	1701	7E	27C8		JMP	AC	EXTERNAL ABORT
01123	1704	BD	115B	XMOVE3	JSR	GETPOS	READ POS FROM BC1
01124	1707	BD	11B8		JSR	DISPLAY	
01125	170A	7D	00A9		TST	XSPMK2	
01126	170D	26	0A		BNE	NXTO	
01127	170F	B6	A099		LDA A	PTMC2	TEST FOR TO(SPEED INC/DEC RAT
01128	1712	84	02		AND A	#\$02	
01129	1714	27	03		BEQ	NXTO	
01130	1716	BD	1387		JSR	IDSP	INC/DEC SPEED
01131	1719	DE	79	NXTO	LDX	REG2	
01132	171B	BD	1818		JSR	MIN1	CONVERT POS TO MATHPACK FORMA
01133	171E	CE	0085		LDX	#IREG	
01134	1721	BD	23E8		JSR	SUBTR	POS-GOTOPOS
01135	1724	96	97		LDA A	FLAG3	
01136	1726	91	6B		CMP A	T6	TEST FOR CHANGE IN SIGN
01137	1728	27	06		BEQ	MOVEX1	
01138	172A	86	FA		LDA A	#\$FA	FORCE MOTOR STOP
01139	172C	97	A3		STA A	D06	
01140	172E	20	0A		BRA	MOVEX2	
01141	1730	96	A3	MOVEX1	LDA A	D06	
01142	1732	81	02		CMP A	#\$02	TEST DIST FROM DEST
01143	1734	2C	07		BGE	MOVEXY	
01144	1736	91	69		CMP A	T4	
01145	1738	27	03		BEQ	MOVEXY	
01146	173A	BD	15E7	MOVEX2	JSR	XSPEED	SELECT NEW SPEED
01147	173D	7D	0091	MOVEXY	TST	MODE1	TEST MOVE FLAGS
01148	1740	26	08		BNE	YMOVE	
01149	1742	7D	0090		TST	MODE	
01150	1745	27	53		BEQ	HALT	
01151	1747	7E	16F2	XMO	JMP	XMOVE	
01152	174A	B6	A001	YMOVE	LDA A	PIAC1	KEYBOARD
01153	174D	4D			TST A		IRQ FLAG
01154	174E	2A	03		BPL	YMOVE1	
01155	1750	7E	27C8		JMP	AC	ABORT
01156	1753	B6	A0A0	YMOVE1	LDA A	MODEMC	
01157	1756	47			ASR A		
01158	1757	24	03		BCC	YMOVE2	
01159	1759	7E	27C8		JMP	AC	EXTERNAL ABORT
01160	175C	BD	115B	YMOVE2	JSR	GETPOS	

01161	175F	BD	11B8		JSR	DISPLAY	
01162	1762	B6	A099		LDA A	PTMC2	TEST FOR TO
01163	1765	84	02		AND A	#\$02	
01164	1767	27	03		BEQ	NYTO	NO Y TO
01165	1769	BD	1387		JSR	IDSP	INC/DEC SPEED
01166	176C	DE	7B	NYTO	LDX	REG3	
01167	176E	BD	1818		JSR	MIN1	CONVERT TO MATH FORMAT
01168	1771	CE	008A		LDX	#IREG1	
01169	1774	BD	23E8		JSR	SUBTR	POS-GOTOPOS
01170	1777	96	97		LDA A	FLAG3	
01171	1779	91	6C		CMP A	T7	CHECK FOR CHANGE IN SIGN
01172	177B	27	06		BEQ	MOVEY1	
01173	177D	86	FA		LDA A	#\$FA	FORCE STOP
01174	177F	97	A3		STA A	D06	
01175	1781	20	0A		BRA	MOVEY3	
01176	1783	96	A3	MOVEY1	LDA A	D06	
01177	1785	81	02		CMP A	#\$02	DIST FROM DEST?
01178	1787	2C	07		BGE	MOVEY2	
01179	1789	91	6A		CMP A	T5	
01180	178B	27	03		BEQ	MOVEY2	
01181	178D	BD	1689	MOVEY3	JSR	YSPEED	SELECT NEW SPEED
01182	1790	7D	0090	MOVEY2	TST	MODE	TEST MOVE FLAGS
01183	1793	26	B2		BNE	XMO	
01184	1795	7D	0091		TST	MODE1	
01185	1798	26	B0		BNE	YMOVE	
01186	179A	86	01	HALT	LDA A	#\$01	
01187	179C	B7	A099		STA A	PTMC2	CR#1 CAN BE WRITTEN
01188	179F	86	97		LDA A	#\$97	
01189	17A1	B7	A098		STA A	PTMC13	PRESET TIMERS
01190	17A4	C6	63		LDA B	#\$63	
01191	17A6	86	C7		LDA A	#\$C7	
01192	17A8	F7	A09C		STA B	PT2MSB	RELOAD FASTER SPEED INC/DEC R
01193	17AB	B7	A09D		STA A	PT2LSB	
01194	17AE	7F	009B		CLR	AFLAG	CLEAR ABORT FLAG
01195	17B1	7F	A006		CLR	PIAD4	RESET DIR RELAYS
01196	17B4	7D	00A9		TST	XSPMK2	MK1 or MK2 SP.CONTROLLER
01197	17B7	27	0D		BEQ	HALT1	MK1
01198	17B9	BD	1138	HALT2	JSR	SPM1	
01199	17BC	96	AA		LDA A	XSPDFL	MK2....AT 0 SPEED YET
01200	17BE	26	F9		BNE	HALT2	NO.....LOOP
01201	17C0	7F	008F		CLR	DIR	
01202	17C3	BD	1138		JSR	SPM1	INFORM SPEED MODULE
01203	17C6	7F	008F	HALT1	CLR	DIR	
01204	17C9	BD	1124		JSR	SPM	INFORM POSITION MODULE
01205	17CC	BD	1A65		JSR	ER2	TEST FOR INCORRECT POS FIXES
01206	17CF	39			RTS		
01207				*			
01208				* /MINX/			
01209				* THIS SUB SETS THE X MINUEND			
01210	17D0	86	FF	MINX	LDA A	#\$FF	
01211	17D2	97	9B		STA A	AFLAG	SET ABORT FLAG
01212	17D4	CE	0001		LDX	#XCPOS	
01213	17D7	DF	79		STX	REG2	
01214	17D9	BD	1818		JSR	MIN1	
01215	17DC	CE	0085		LDX	#IREG	
01216	17DF	BD	23E8		JSR	SUBTR	POS-GOTO POS
01217	17E2	96	97		LDA A	FLAG3	SIGN
01218	17E4	97	6B		STA A	T6	


```

01219 17E6 81 FF          CMP A   #$FF
01220 17E8 27 05          BEQ     FFF          BRANCH TO FORWARD
01221 17EA 86 22          LDA A   #$22
01222 17EC 97 90          STA A   MODE        SET X DIRECTION FLAG
01223 17EE 39             RTS
01224 17EF 86 11 FFF     LDA A   #$11
01225 17F1 97 90          STA A   MODE
01226 17F3 39             RTS
01227                      *
01228                      */MINY/
01229                      * THIS SUB SETS THE Y MINUEND
01230 17F4 86 FF MINY     LDA A   #$FF
01231 17F6 97 9B          STA A   AFLAG        SET ABORT FLAG
01232 17F8 CE 0009        LDX     #YCPOS
01233 17FB DF 7B          STX     REG3
01234 17FD BD 1818        JSR     MIN1        MINUEND
01235 1800 CE 008A        LDX     #IREG1
01236 1803 BD 23E8        JSR     SUBTR       POS - GOTO POS
01237 1806 96 97          LDA A   FLAG3
01238 1808 97 6C          STA A   T7
01239 180A 81 FF          CMP A   #$FF
01240 180C 27 05          BEQ     RRR          BRANCH TO RIGHT
01241 180E 86 88          LDA A   #$88
01242 1810 97 91          STA A   MODE1       SET Y DIRECTION FLAG
01243 1812 39             RTS
01244 1813 86 44 RRR     LDA A   #$44
01245 1815 97 91          STA A   MODE1
01246 1817 39             RTS
01247                      *
01248                      */MIN1/
01249                      * THIS SUB. SETS UP MINUEND AND STORES IT
01250                      * IN REGISTERS D03 TO D07
01251 1818 86 08 MIN1     LDA A   #$08        NO OF SHIFTS
01252 181A 97 A3          STA A   D06
01253 181C 4A            DEC A
01254 181D 97 67          STA A   T2
01255 181F A6 03          LDA A   3,X        COARSE
01256 1821 97 A0          STA A   D03
01257 1823 7F 00A2        CLR     D05
01258 1826 7F 0097        CLR     FLAG3     SIGN
01259 1829 7F 0068        CLR     T3
01260 182C BD 185D        JSR     MCON1
01261 182F 96 68          LDA A   T3
01262 1831 97 A1          STA A   D04
01263 1833 7F 0097        CLR     FLAG3
01264 1836 BD 23BF        JSR     NORM        NORMALISE RESULT
01265 1839 39             RTS
01266                      *
01267                      */MCON/
01268                      * THIS SUBROUTINE SETS UP THE SUBTRAHEND
01269                      * AND STORES IT IN IREG (INDEX REGISTER)
01270 183A BD 1818 MCON     JSR     MIN1
01271 183D CE 0085        LDX     #IREG
01272 1840 96 A0 MCO1     LDA A   D03        1ST MANTISSA
01273 1842 A7 00          STA A   0,X
01274 1844 96 A1          LDA A   D04        2ND MANTISSA
01275 1846 A7 01          STA A   1,X
01276 1848 96 A2          LDA A   D05        3RD MANTISSA

```



```

01277 184A A7 02          STA A 2,X
01278 184C 96 A3          LDA A D06      EXPONENT
01279 184E A7 03          STA A 3,X
01280 1850 96 97          LDA A FLAG3    SIGN
01281 1852 A7 04          STA A 4,X
01282 1854 39             RTS
01283                    *
01284                    */MCO/
01285                    *THIS SUB SETS UP THE Y-AXIS SUBRAHEND
01286                    *AND STORES IT IN INDEX REG IREG1
01287 1855 BD 1818 MCO     JSR      MIN1
01288 1858 CE 008A        LDX      #IREG1
01289 185B 20 E3         BRA      MCO1
01290                    *
01291                    */MCON1/
01292                    * THIS SUB COVERTS FINE POSITION TO
01293                    * MATH PACK DESIMAL
01294 185D A6 07          MCON1 LDA A 7,X
01295 185F C6 32          LDA B #32      (50)
01296 1861 10            DES4  SBA
01297 1862 24 03          BCC      DES1    CARRY CLEAR
01298 1864 1B            ABA
01299 1865 20 03          BRA      DES2
01300 1867 7C 0068 DES1  INC      T3
01301 186A 78 0068 DES2  ASL      T3
01302 186D 7A 0067        DEC      T2
01303 1870 27 05          BEQ      DES3
01304 1872 57            ASR B
01305 1873 27 F5          BEQ      DES2    $00 DOES NOT SET CARRY
01306 1875 20 EA          BRA      DES4
01307 1877 39            DES3  RTS
01308                    *
01309                    */DIRSET/
01310                    *THIS SUB SETS THE DIRECTIONS OF MOVEMENT
01311                    *AND DIRECTION LEDS
01312 1878 96 90          DIRSET LDA A MODE    X DIR
01313 187A 26 06          BNE      DS1
01314 187C D6 91          LDA B MODE1    Y DIR
01315 187E D7 8F          STA B DIR
01316 1880 20 0F          BRA      DS3
01317 1882 D6 91          DS1  LDA B MODE1    Y DIR
01318 1884 26 04          BNE      DS2
01319 1886 97 8F          STA A DIR      DIRECTION FLAG
01320 1888 20 07          BRA      DS3
01321 188A 84 F0          DS2  AND A #$F0
01322 188C C4 0F          AND B #$0F
01323 188E 1B            ABA
01324 188F 97 8F          STA A DIR
01325 1891 96 8F          DS3  LDA A DIR      SELECT DIR LEDS/RELAYS
01326 1893 81 14          CMP A #$14
01327 1895 26 07          BNE      DR1
01328 1897 86 55          LDA A #$55    FWD/LEFT
01329 1899 B7 A006        STA A PIAD4    LEDS/RELAY
01330 189C 20 49          BRA      DS4
01331 189E 81 18          DR1  CMP A #$18
01332 18A0 26 07          BNE      DR2
01333 18A2 86 99          LDA A #$99    FWD/RIGHT
01334 18A4 B7 A006        STA A PIAD4    LEDS/RELAY

```


01335	18A7	20	3E		BRA	DS4	
01336	18A9	81	24	DR2	CMP A	#\$24	
01337	18AB	26	07		BNE	DR3	
01338	18AD	86	66		LDA A	#\$66	REV/LEFT
01339	18AF	B7	A006		STA A	PIAD4	LEDS/RELAY
01340	18B2	20	33		BRA	DS4	
01341	18B4	81	28	DR3	CMP A	#\$28	
01342	18B6	26	07		BNE	DR4	
01343	18B8	86	AA		LDA A	#\$AA	REV/RIGHT
01344	18BA	B7	A006		STA A	PIAD4	LEDS/RELAY
01345	18BD	20	28		BRA	DS4	
01346	18BF	81	11	DR4	CMP A	#\$11	
01347	18C1	26	05		BNE	DR5	
01348	18C3	B7	A006		STA A	PIAD4	LEDS/RELAY
01349	18C6	20	1F		BRA	DS4	
01350	18C8	81	22	DR5	CMP A	#\$22	
01351	18CA	26	05		BNE	DR6	
01352	18CC	B7	A006		STA A	PIAD4	LEDS RELAY
01353	18CF	20	16		BRA	DS4	
01354	18D1	81	44	DR6	CMP A	#\$44	
01355	18D3	26	05		BNE	DR7	
01356	18D5	B7	A006		STA A	PIAD4	LEDS/RELAY
01357	18D8	20	0D		BRA	DS4	
01358	18DA	81	88	DR7	CMP A	#\$88	
01359	18DC	26	05		BNE	DR8	
01360	18DE	B7	A006		STA A	PIAD4	LEDS/RELAY
01361	18E1	20	04		BRA	DS4	
01362	18E3	4F		DR8	CLR A		
01363	18E4	B7	A006		STA A	PIAD4	
01364	18E7	39		DS4	RTS		
01365				*			
01366				*/ADDSCLE/			
01367				* THIS SUB	ADDS THE SCALE TO MOVE TO POSITION		
01368				*(ADDITION OF TWO POSITIONS IN THE X REGISTERS)			
01369	18E8	A6	06	ADDSCLE	LDA A	6,X	
01370	18EA	E6	16		LDA B	22,X	
01371	18EC	1B			ABA		
01372	18ED	C6	0A		LDA B	#\$0A	
01373	18EF	10			SBA		
01374	18F0	2B	06		BMI	AA1	
01375	18F2	A7	06		STA A	6,X	2nd DES
01376	18F4	6C	05		INC	5,X	
01377	18F6	20	03		BRA	A2	
01378	18F8	1B		AA1	ABA		
01379	18F9	A7	06		STA A	6,X	2nd DES
01380	18FB	A6	05	A2	LDA A	5,X	
01381	18FD	E6	15		LDA B	21,X	
01382	18FF	1B			ABA		
01383	1900	C6	0A		LDA B	#\$0A	
01384	1902	10			SBA		
01385	1903	2B	06		BMI	A3	
01386	1905	A7	05		STA A	5,X	1st DES
01387	1907	6C	02		INC	2,X	
01388	1909	20	03		BRA	A4	
01389	190B	1B		A3	ABA		
01390	190C	A7	05		STA A	5,X	1st DES
01391	190E	A6	02	A4	LDA A	2,X	
01392	1910	E6	12		LDA B	18,X	

01393	1912	1B		ABA		
01394	1913	C6	0A	LDA B	#\$0A	
01395	1915	10		SBA		
01396	1916	2B	06	BMI	A5	
01397	1918	A7	02	STA A	2,X	UNITS
01398	191A	6C	01	INC	1,X	
01399	191C	20	03	BRA	A6	
01400	191E	1B		A5	ABA	
01401	191F	A7	02	STA A	2,X	UNITS
01402	1921	A6	01	A6	LDA A	1,X
01403	1923	81	0A	CMP A	#\$0A	
01404	1925	2D	04	BLT	A7	
01405	1927	6C	00	INC	0,X	
01406	1929	6F	01	CLR	1,X	
01407	192B	39		A7	RTS	
01408				*		
01409				*/SUBSCL/		
01410				* THIS SUB SUBTRACTS THE SCALE FROM MOVE TO POSITION		
01411				* (VALUES ARE IN THE X REGISTERS)		
01412	192C	A6	06	SUBSCL	LDA A	6,X
01413	192E	E6	16		LDA B	22,X
01414	1930	10			SBA	
01415	1931	2B	04		BMI	S1
01416	1933	A7	06		STA A	6,X
01417	1935	20	24		BRA	S2
01418	1937	1B		S1	ABA	
01419	1938	C6	0A		LDA B	#\$0A
01420	193A	1B			ABA	
01421	193B	E6	16		LDA B	22,X
01422	193D	10			SBA	
01423	193E	A7	06		STA A	6,X
01424	1940	6A	05		DEC	5,X
01425	1942	2A	17		BPL	S2
01426	1944	86	09		LDA A	#\$09
01427	1946	A7	05		STA A	5,X
01428	1948	6A	02		DEC	2,X
01429	194A	2A	0F		BPL	S2
01430	194C	A7	02		STA A	2,X
01431	194E	6A	01		DEC	1,X
01432	1950	2A	09		BPL	S2
01433	1952	A7	01		STA A	1,X
01434	1954	6A	00		DEC	0,X
01435	1956	2A	03		BPL	S2
01436	1958	7E	19A4		JMP	NUL
01437	195B	A6	05	S2	LDA A	5,X
01438	195D	E6	15		LDA B	21,X
01439	195F	10			SBA	
01440	1960	2B	04		BMI	S3
01441	1962	A7	05		STA A	5,X
01442	1964	20	1E		BRA	S4
01443	1966	1B		S3	ABA	
01444	1967	C6	0A		LDA B	#\$0A
01445	1969	1B			ABA	
01446	196A	E6	15		LDA B	21,X
01447	196C	10			SBA	
01448	196D	A7	05		STA A	5,X
01449	196F	6A	02		DEC	2,X
01450	1971	2A	11		BPL	S4

01451	1973	86	09		LDA A	#\$09	
01452	1975	A7	02		STA A	2,X	
01453	1977	6A	01		DEC	1,X	
01454	1979	2A	09		BPL	S4	
01455	197B	A7	01		STA A	1,X	
01456	197D	6A	00		DEC	0,X	
01457	197F	2A	03		BPL	S4	
01458	1981	7E	19A4		JMP	NUL	
01459	1984	A6	02	S4	LDA A	2,X	
01460	1986	E6	12		LDA B	18,X	
01461	1988	10			SBA		
01462	1989	2B	04		BMI	S5	
01463	198B	A7	02		STA A	2,X	UNITS
01464	198D	20	1E		BRA	S6	
01465	198F	1B		S5	ABA		
01466	1990	C6	0A		LDA B	#\$0A	
01467	1992	1B			ABA		
01468	1993	E6	12		LDA B	18,X	
01469	1995	10			SBA		
01470	1996	A7	02		STA A	2,X	UNITS
01471	1998	6A	01		DEC	1,X	
01472	199A	2A	11		BPL	S6	
01473	199C	86	09		LDA A	#\$09	
01474	199E	A7	01		STA A	1,X	TENS
01475	19A0	6A	00		DEC	0,X	
01476	19A2	2A	09		BPL	S6	
01477	19A4	4F		NUL	CLR A		ZERO THE DISPLAY
01478	19A5	A7	00		STA A	0,X	
01479	19A7	08			INX		
01480	19AB	8C	0008		CPX	#\$0008	
01481	19AB	26	F7		BNE	NUL	
01482	19AD	39		S6	RTS		
01483				*			
01484				*/BUT/			
01485				*THIS SUB CHECKS IF KEYBOARD BUTTON HAS BEEN RELEASE			
01486	19AE	B6	A001	BUT	LDA A	PIAC1	
01487	19B1	84	40		AND A	#\$40	
01488	19B3	27	F9		BEQ	BUT	
01489	19B5	BD	11E8		JSR	DELAY	
01490	19B8	B6	A000		LDA A	PIAD1	RESET FLAG
01491	19BB	39			RTS		
01492				*			
01493				*/SPCON/			
01494				*THIS SUB FINDS THE SPEED PERIOD AND			
01495				*STORES IT IN XSPDCT(MK2 SPD CONTROLLER)			
01496	19BC	96	94	SPCON	LDA A	FLAG	
01497	19BE	81	01		CMP A	#\$01	HOME
01498	19C0	27	69		BEQ	FS3	IF HOME THEN FAST SPEED
01499	19C2	CE	0053		LDX	#XCSPD	
01500	19C5	BD	1203		JSR	CHEX	CHANGE TO A HEX VALUE
01501	19C8	A6	07		LDA A	7,X	
01502	19CA	27	69		BEQ	SPERR	NO SPEED ENTERED
01503	19CC	81	0A		CMP A	#\$0A	
01504	19CE	22	65		BHI	SPERR	ENTERED SPEED TO HI
01505	19D0	97	A0		STA A	D03	1st MANTISSA
01506	19D2	5F			CLR B		
01507	19D3	D7	A1		STA B	D04	2nd MANTISSA
01508	19D5	D7	A2		STA B	D05	3rd MANTISSA

01509	19D7	D7	97		STA B	FLAG3	SIGN BYTE
01510	19D9	86	08		LDA A	#\$08	
01511	19DB	97	A3		STA A	D06	EXPONENT
01512	19DD	BD	23BF		JSR	NORM	MATH NORMALISATION ROUTINE
01513	19E0	CE	008A		LDX	#IREG1	(IREG USED TO STORE GOTO POS)
01514	19E3	BD	1840		JSR	MC01	STORE IN INDEX REG
01515	19E6	86	09		LDA A	#\$09	
01516	19E8	97	A3		STA A	D06	EXPONENT
01517	19EA	86	93		LDA A	#\$93	294 NORMALISED
01518	19EC	97	A0		STA A	D03	
01519	19EE	D7	A1		STA B	D04	
01520	19F0	D7	A2		STA B	D05	
01521	19F2	D7	97		STA B	FLAG3	
01522	19F4	BD	2487		JSR	MULT	MULTIPLY(SPD MULT 294)
01523	19F7	CE	008A		LDX	#IREG1	
01524	19FA	BD	1840		JSR	MC01	RESULT INTO INDEX REG
01525	19FD	86	F4		LDA A	#\$F4	NORMALISED DIVIDEND(1000000)
01526	19FF	97	A0		STA A	D03	
01527	1A01	86	24		LDA A	#\$24	
01528	1A03	97	A1		STA A	D04	
01529	1A05	7F	00A2		CLR	D05	
01530	1A08	7F	0097		CLR	FLAG3	
01531	1A0B	86	14		LDA A	#\$14	EXPONENT OF 20
01532	1A0D	97	A3		STA A	D06	
01533	1A0F	BD	24FB		JSR	DIV	DIVISION ROUTINE
01534	1A12	D6	A3		LDA B	D06	EXPONENT
01535	1A14	86	10		LDA A	#\$10	16(24 IF VALUES ARE LARGER)
01536	1A16	10			SBA		
01537	1A17	27	09		BEQ	FS1	
01538	1A19	74	00A0	FS2	LSR	D03	DE-NORMALISE
01539	1A1C	76	00A1		ROR	D04	
01540	1A1F	4A			DEC A		
01541	1A20	26	F7		BNE	FS2	
01542	1A22	96	A0	FS1	LDA A	D03	HI BYTE
01543	1A24	D6	A1		LDA B	D04	LO BYTE
01544	1A26	97	A7		STA A	XSPDCT	X CONTROLLER HI BYTE
01545	1A28	D7	A8		STA B	XSPDCT+1	X CONTROLLER LO BYTE
01546	1A2A	39		SPCONF	RTS		
01547	1A2B	86	01	FS3	LDA A	#\$01	
01548	1A2D	C6	4D		LDA B	#\$4D	
01549	1A2F	97	A7		STA A	XSPDCT	FAST SPEED
01550	1A31	D7	A8		STA B	XSPDCT+1	
01551	1A33	20	F5		BRA	SPCONF	
01552	1A35	7E	1BAD	SPERR	JMP	ER7	ERROR
01553				*			
01554				*ALARM/ERROR			RELATED ROUTINES
01555				*			
01556	1A38	BD	1C07	ER1	JSR	SALARM	TRANSMIT FAULT
01557	1A3B	86	FF		LDA A	#\$FF	
01558	1A3D	97	6D		STA A	T8	TO RESET DISPLAY
01559	1A3F	96	6E		LDA A	T9	X/Y ERROR FLAG
01560	1A41	81	AA		CMP A	#\$AA	X
01561	1A43	26	0C		BNE	YERO	
01562	1A45	86	CA		LDA A	#\$CA	DISPLAY A
01563	1A47	B7	A002		STA A	PIAD2	
01564	1A4A	C6	36		LDA B	#\$36	
01565	1A4C	F7	A003		STA B	PIAC2	ILLUMINATE ERROR DIGET
01566	1A4F	20	0E		BRA	XYER	

01567	1A51	81	55	YERO	CMP	A	#55	
01568	1A53	26	0A		BNE		XYER	
01569	1A55	86	CB		LDA	A	#CB	DISPLAY B
01570	1A57	B7	A002		STA	A	PIAD2	
01571	1A5A	C6	36		LDA	B	#36	
01572	1A5C	F7	A003		STA	B	PIAC2	ILL. ERROR DIGET
01573	1A5F	84	0F	XYER	AND	A	#0F	
01574	1A61	B7	A002		STA	A	PIAD2	SET STROBE LOW
01575	1A64	39			RTS			
01576				*				
01577				*ANY				FAULTY POSITION FIXES
01578				*				
01579	1A65	7D	0093	ER2	TST		FLA	INT/EXT FLAG
01580	1A68	27	5A		BEQ		ER2B	EXTERNAL
01581	1A6A	B6	A004		LDA	A	PIAD3	
01582	1A6D	97	92		STA	A	LAT	REM. LED DISPLAY
01583	1A6F	96	11	ER21	LDA	A	XER	X ERROR COUNT
01584	1A71	27	1A		BEQ		ER23	
01585	1A73	C6	C0		LDA	B	#C0	
01586	1A75	1B			ABA			
01587	1A76	B7	A002		STA	A	PIAD2	DISPLAY
01588	1A79	C6	36		LDA	B	#36	
01589	1A7B	F7	A003		STA	B	PIAC2	ILL ERROR DIG
01590	1A7E	86	40		LDA	A	#40	
01591	1A80	B7	A004		STA	A	PIAD3	ILL X LED
01592	1A83	BD	1AAA		JSR		CLRCHK	
01593	1A86	BD	11F0		JSR		DELAY1	
01594	1A89	96	12	ER22	LDA	A	YER	Y ERROR COUNT
01595	1A8B	27	E2		BEQ		ER21	
01596	1A8D	96	12	ER23	LDA	A	YER	
01597	1A8F	27	18		BEQ		ER24	NO ERROR
01598	1A91	C6	C0		LDA	B	#C0	
01599	1A93	1B			ABA			
01600	1A94	B7	A002		STA	A	PIAD2	DISPLAY
01601	1A97	C6	36		LDA	B	#36	
01602	1A99	F7	A003		STA	B	PIAC2	ILL ERROR DIG
01603	1A9C	86	80		LDA	A	#80	
01604	1A9E	B7	A004		STA	A	PIAD3	ILL Y LED
01605	1AA1	BD	1AAA		JSR		CLRCHK	
01606	1AA4	BD	11F0		JSR		DELAY1	
01607	1AA7	20	C6		BRA		ER21	
01608	1AA9	39		ER24	RTS			
01609	1AAA	B6	A000	CLRCHK	LDA	A	PIAD1	KEYBD
01610	1AAD	84	0F		AND	A	#0F	
01611	1AAF	81	0F		CMP	A	#0F	ABORT/CLEAR
01612	1AB1	26	10		BNE		CC1	
01613	1AB3	86	3E		LDA	A	#3E	
01614	1AB5	B7	A003		STA	A	PIAC2	BLANK ERROR
01615	1AB8	96	92		LDA	A	LAT	
01616	1ABA	B7	A004		STA	A	PIAD3	RESTORE LED DISPLAY
01617	1ABD	7F	0011		CLR		XER	
01618	1AC0	7F	0012		CLR		YER	CLEAR ERROR COUNTERS
01619	1AC3	39		CC1	RTS			
01620	1AC4	96	11	ER2B	LDA	A	XER	X ER COUNT
01621	1AC6	27	10		BEQ		ER2B1	
01622	1AC8	CE	2FDE		LDX		HMP29	
01623	1ACB	BD	C545		JSR		VDUOUT	(X FIXES)
01624	1ACE	96	11		LDA	A	XER	

01625	1AD0	8A	30		DRA	A	#\$30	
01626	1AD2	BD	C92F		JSR		VDU	DISPLAY NO.
01627	1AD5	BD	C52F		JSR		VDURET	
01628	1AD8	96	12	ER2B1	LDA	A	YER	
01629	1ADA	27	10		BEQ		ER2B2	NO ERROR
01630	1ADC	CE	2FF8		LDX		#MP30	
01631	1ADF	BD	C545		JSR		VDUOUT	
01632	1AE2	96	12		LDA	A	YER	
01633	1AE4	8A	30		DRA	A	#\$30	
01634	1AE6	BD	C92F		JSR		VDU	DISPLAY NO.
01635	1AE9	BD	C52F		JSR		VDURET	
01636	1AEC	39		ER2B2	RTS			
01637				*				
01638				*			FIRST X OR Y AXES FIX OK?	
01639				*				
01640	1AED	7F	0073	ER4	CLR		XINC	X/Y POSITION OK?
01641	1AF0	7F	0074		CLR		YINC	
01642	1AF3	7F	006F		CLR		XSTEP	
01643	1AF6	7F	0070		CLR		YSTEP	
01644	1AF9	7D	00A9		TST		XSPMK2	MK2 SP.CONT.?
01645	1AFC	27	1E		BEQ		SMO	NO
01646	1AFE	7F	00A7	SMO4	CLR		XSPDCT	
01647	1B01	7F	00A8		CLR		XSPDCT+1	
01648	1B04	BD	1138		JSR		SPM1	INFORM SPEED CONTROL
01649	1B07	96	AA		LDA	A	XSPDFL	AT ZERO SPEED?
01650	1B09	26	F3		BNE		SMO4	NO
01651	1B0B	7F	0090		CLR		MODE	YES/CLEAR FLAGS
01652	1B0E	7F	008F		CLR		DIR	
01653	1B11	BD	1878		JSR		DIRSET	CLEAR LED
01654	1B14	BD	1138		JSR		SPM1	INFORM SPEED CONTROL(RELAY RE
01655	1B17	BD	1124		JSR		SPM	INFORM POSITION SENSE BOARD
01656	1B1A	20	17		BRA		SMO3	
01657	1B1C	BD	1387	SMO	JSR		IDSP	DECREMENT/STOP MOTORS
01658	1B1F	7D	0090		TST		MODE	X MOVEMENT
01659	1B22	26	F8		BNE		SMO	
01660	1B24	7D	0091		TST		MODE1	Y MOVEMENT
01661	1B27	26	F3		BNE		SMO	
01662	1B29	86	15		LDA	A	#\$15	
01663	1B2B	B7	A099		STA	A	PTMC2	CR#1 MAY BE WRITTEN
01664	1B2E	86	97		LDA	A	#\$97	
01665	1B30	B7	A098		STA	A	PTMC13	TIMERS HELD PRESET
01666	1B33	BD	115B	SMO3	JSR		GETPOS	GET LATEST POSITION
01667	1B36	BD	11B8		JSR		DISPLAY	AND DISPLAY
01668	1B39	86	CE		LDA	A	#\$CE	DISPLAY ONLY AN E ON ERROR DI
01669	1B3B	B7	A002		STA	A	PIAD2	
01670	1B3E	C6	36		LDA	B	#\$36	
01671	1B40	F7	A003		STA	B	PIAC2	ILL. ERROR DIGET
01672	1B43	84	0F		AND	A	#\$0F	
01673	1B45	B7	A002		STA	A	PIAD2	SET STROBE LOW
01674	1B48	7D	0093		TST		FLA	INT/EXT FLAG
01675	1B4B	27	1D		BEQ		SMO5	EXTENAL CONTROL
01676	1B4D	B6	A000	SMO1	LDA	A	PIAD1	
01677	1B50	84	0F		AND	A	#\$0F	
01678	1B52	81	0E		CMP	A	#\$0E	CLEAR
01679	1B54	27	0C		BEQ		SMO2	
01680	1B56	81	0F		CMP	A	#\$0F	INCORRECT VALUE
01681	1B58	26	F3		BNE		SMO1	
01682	1B5A	BD	19AE		JSR		BUT	


```

01683 1B5D BD 102D      JSR      SETUP
01684 1B60 20 08        BRA      SM05
01685 1B62 86 3E      SM02  LDA A   #$3E      CLEAR ERROR DIGET
01686 1B64 B7 A003      STA A   PIAC2
01687 1B67 BD 19AE      JSR      BUT
01688 1B6A 39          SM05  RTS
01689                  *
01690                  *PLATFORM 1 METRE FROM LIMIT
01691                  *
01692 1B6B BD 1C07      ER5   JSR      SALARM      1M FROM ENDSTOP
01693 1B6E 86 CC        LDA A   #$CC
01694 1B70 B7 A002      STA A   PIAD2      DISPLAY C
01695 1B73 C6 36        LDA B   #$36
01696 1B75 F7 A003      STA B   PIAC2      ILL. ERROR DIGET
01697 1B78 C6 FF        LDA B   #$FF
01698 1B7A D7 6D        STA B   T8
01699 1B7C 84 0F        AND A   #$0F
01700 1B7E B7 A002      STA A   PIAD2      SET STROBE LOW
01701 1B81 39          RTS
01702                  *
01703                  * MOVE TO POSITION EXCEEDS LIMIT
01704                  *
01705 1B82 BD 1C07      ER6   JSR      SALARM      GOTO POSITION OVERRIDES ENDST
01706 1B85 7D 0093      TST      FLA
01707 1B88 27 15        BEQ      EXER6
01708 1B8A B6 A004      LDA A   PIAD3      CURRENT LED DISPLAY
01709 1B8D 97 92        STA A   LAT
01710 1B8F 86 08        LDA A   #$08      FLASHING LED
01711 1B91 98 92        EOR A   LAT
01712 1B93 97 6E        STA A   T9
01713 1B95 BD 1BD6      JSR      EDIS
01714 1B98 86 04        LDA A   #$04
01715 1B9A 97 94        STA A   FLAG
01716 1B9C 7E 2597      JMP      SEL      REENTER GOTO POSITION
01717 1B9F CE 2E92      EXER6 LDX      HMP20
01718 1BA2 BD C545      JSR      VDUOUT
01719 1BA5 BD D551      JSR      WRONG      TRY AGAIN
01720 1BA8 31          INS
01721 1BA9 31          INS      ADJUST STACK
01722 1BAA 31          INS
01723 1BAB 31          INS
01724 1BAC 39          RTS
01725                  *
01726 1BAD BD 1C07      ER7   JSR      SALARM      NO OR WRONG SPEED ENTERED
01727 1BB0 86 50        LDA A   #$50      FLASH SPEED LED
01728 1BB2 97 92        STA A   LAT      REMEMBER DISPLAY
01729 1BB4 86 40        LDA A   #$40      FLASHING LED
01730 1BB6 97 6E        STA A   T9
01731 1BB8 BD 1BD6      JSR      EDIS
01732 1BBB 86 03        LDA A   #$03      SPEED SELECT
01733 1BBD 97 94        STA A   FLAG      INPUT MODE FLAG
01734 1BBF 7E 2597      JMP      SEL      SETUP FOR SPEED INPUT
01735                  *
01736 1BC2 BD 1C07      ER3   JSR      SALARM      (X OR Y)
01737 1BC5 B6 A004      LDA A   PIAD3      CURRENT LED DISPLAY
01738 1BC8 97 92        STA A   LAT
01739 1BCA 86 C0        LDA A   #$C0
01740 1BCC 98 92        EOR A   LAT

```



```

01741 1BCE 97 6E          STA A   T9
01742 1BD0 BD 1BD6        JSR     EDIS
01743 1BD3 7E 2844        JMP     RTI
01744                      *
01745 1BD6 86 CF          EDIS LDA A   #$CF      DISPLAY ERROR DIGET
01746 1BD8 B7 A002        STA A   PIAD2
01747 1BDB C6 36          LDA B   #$36
01748 1BDD F7 A003        STA B   PIAC2      ILL. ERROR DIGET
01749 1BE0 84 0F          AND A   #$0F
01750 1BE2 B7 A002        STA A   PIAD2      SET STROBE LOW
01751 1BE5 96 6E          LDA A   T9
01752 1BE7 B7 A004        STA A   PIAD3
01753 1BEA BD 11F0        JSR     DELAY1
01754 1BED 96 92          LDA A   LAT
01755 1BEF B7 A004        STA A   PIAD3      FLASHING LED
01756 1BF2 BD 11F0        JSR     DELAY1
01757 1BF5 B6 A000        LDA A   PIAD1
01758 1BF8 84 0F          AND A   #$0F
01759 1BFA 81 0F          CMP A   #$0F
01760 1BFC 26 D8          BNE     EDIS
01761 1BFE 86 3E          LDA A   #$3E      CLEAR ERROR
01762 1C00 B7 A003        STA A   PIAC2
01763 1C03 BD 19AE        JSR     BUT      DEBOUNCE
01764 1C06 39            RTS
01765                      *
01766 1C07 86 36          SALARM LDA A   #$36
01767 1C09 B7 A005        STA A   PIAC3      SOUND ALARM
01768 1C0C BD 11F0        JSR     DELAY1
01769 1C0F 86 3E          LDA A   #$3E
01770 1C11 B7 A005        STA A   PIAC3
01771 1C14 39            RTS
01772                      *
01773                      OPT      L
01774                      *
01775                      * STANOP RELATED ROUTINES
01776                      * -----
01777                      */COMYES---COMMAND INTERPRETER/
01778 1C15 B6 C0A5        COMYES LDA A   COMBUF+4 EXT PROG CONTROLLED
01779 1C18 27 03          BEQ     CYA      NO
01780 1C1A 7E 2309        JMP     EXPROG   ENTER EXT PROG VALUES
01781 1C1D 86 3E          CYA   LDA A   #$3E
01782 1C1F B7 A003        STA A   PIAC2      BLANK ERROR DIGIT
01783 1C22 96 04          LDA A   XCPOS+3
01784 1C24 81 AA          CMP A   #$AA      ANY POSITION FIX YET?
01785 1C26 26 03          BNE     CY1      YES
01786 1C28 7E 1C7A        JMP     FINDXY   GET POSITIONS
01787 1C2B B6 C0A3        CY1   LDA A   COMBUF+2
01788 1C2E 81 4C          CMP A   #'L      LEFT
01789 1C30 26 05          BNE     CY2
01790 1C32 86 40          LDA A   #$40
01791 1C34 7E 1C4F        JMP     OMO
01792 1C37 81 52          CY2   CMP A   #'R      RIGHT
01793 1C39 26 05          BNE     CY3
01794 1C3B 86 80          LDA A   #$80
01795 1C3D 7E 1C4F        JMP     OMO
01796 1C40 81 42          CY3   CMP A   #'B      REVERSE(BACKWARDS)
01797 1C42 26 05          BNE     CY4
01798 1C44 86 20          LDA A   #$20

```



```

01799 1C46 7E 1C4F      JMP      OMO
01800 1C49 81 46      CY4    CMP A    #'F      FORWARD
01801 1C4B 26 0D      BNE      CY5
01802 1C4D 86 10      LDA A    #$10
01803 1C4F 36      OMO      PSH A
01804 1C50 CE 2CD1      LDX      #MP8
01805 1C53 BD C545      JSR      VDUOUT
01806 1C56 32      PUL A
01807 1C57 7E 1193      JMP      FLAG0      OVERRIDE MOVE
01808 1C5A 81 47      CY5    CMP A    #'G
01809 1C5C 26 03      BNE      CY6
01810 1C5E 7E 1D69      JMP      MT0      NON-OVERRIDE MOVE
01811 1C61 81 50      CY6    CMP A    #'P
01812 1C63 26 03      BNE      CY7
01813 1C65 7E 1F7C      JMP      PAR      DISPLAY PARAMETERS
01814 1C68 81 49      CY7    CMP A    #'I
01815 1C6A 26 03      BNE      CY8
01816 1C6C 7E 221C      JMP      SINC      INCREMENT
01817 1C6F 81 44      CY8    CMP A    #'D
01818 1C71 26 03      BNE      CY9
01819 1C73 7E 2302      JMP      SDEC      DECREMENT
01820 1C76 BD D551      CY9    JSR      WRONG      TRY AGAIN
01821 1C79 39      RTS
01822      *
01823      *
01824      */FINDXY/
01825      *THIS ROUTINE MOVES THE PLATFORM TO THE 1ST
01826      *POSITION SENSOR. Y IS SKIPPED WITH MK2 SP.CONT.
01827 1C7A CE 2BF7      FINDXY LDX      #MP3
01828 1C7D BD C545      JSR      VDUOUT      ENTER F OR B
01829 1C80 0C      CLC
01830 1C81 BD C5F9      FX1    JSR      KBDCHK      CHECK KEYBD FOR ENTERY
01831 1C84 24 FB      BCC      FX1
01832 1C86 81 42      CMP A    #$42      (B)
01833 1C88 26 04      BNE      FX2
01834 1C8A 86 20      LDA A    #$20
01835 1C8C 20 0B      BRA      FX3
01836 1C8E 81 46      FX2    CMP A    #$46      (F)
01837 1C90 27 05      BEQ      FX2A
01838 1C92 BD D551      JSR      WRONG      TRY AGAIN
01839 1C95 20 E3      BRA      FINDXY      WRONG KEY
01840 1C97 86 10      FX2A   LDA A    #$10
01841 1C99 BD EA7C      FX3    JSR      BUSY
01842 1C9C BD C52F      JSR      VDURET
01843 1C9F BD 1193      JSR      FLAG0      OVERRIDE MOVE
01844 1CA2 CE 2C70      LDX      #MP5
01845 1CA5 BD C545      JSR      VDUOUT      X POS = (PROMPT)
01846 1CA8 BD 1D3E      JSR      DISNO      DISPLAY POSITION
01847 1CAB CE 2CAA      FX3A   LDX      #MP7
01848 1CAE BD C545      JSR      VDUOUT      OK?
01849 1CB1 0C      CLC
01850 1CB2 BD C5F9      FX4    JSR      KBDCHK      KEYB. ENTRY?
01851 1CB5 24 FB      BCC      FX4
01852 1CB7 81 59      CMP A    #'Y      Y IF OK
01853 1CB9 27 14      BEQ      FX5
01854 1CBB 81 4E      CMP A    #'N
01855 1CBD 27 05      BEQ      FX4A      OK
01856 1CBF BD D551      JSR      WRONG

```



```

01857 1CC2 20 E7          BRA    FX3A
01858 1CC4 BD 102D FX4A   JSR    SETUP    NOT OK
01859 1CC7 BD 115B       JSR    GETPOS
01860 1CCA BD 11B8       JSR    DISPLAY
01861 1CCD 20 AB         BRA    FINDXY    DO AGAIN
01862 1CCF 86 3E        FX5    LDA    A    #3E
01863 1CD1 B7 A003       STA    A    PIAC2    CLR ERROR DIG.
01864 1CD4 7D 00A9       TST     XSPMK2    MK1 OR MK2 SPEED CONT.
01865 1CD7 27 01        BEQ     FY
01866 1CD9 39          FXY    RTS
01867 1CDA CE 2C36 FY    LDX     #MP4
01868 1CDD BD C545       JSR    VDUOUT    L OR R?
01869 1CE0 0C          CLC
01870 1CE1 BD C5F9 FY1   JSR    KBDCHK
01871 1CE4 24 FB        BCC     FY1
01872 1CE6 81 4C        CMP    A    #4C    (L)
01873 1CE8 26 04        BNE     FY2
01874 1CEA 86 40        LDA    A    #40
01875 1CEC 20 0B        BRA     FY3
01876 1CEE 81 52        FY2    CMP    A    #52    (R)
01877 1CF0 27 05        BEQ     FY2A
01878 1CF2 BD D551       JSR    WRONG
01879 1CF5 20 E3        BRA     FY    DO AGAIN
01880 1CF7 86 80        FY2A   LDA    A    #80
01881 1CF9 BD EA7C FY3   JSR    BUSY
01882 1CFC BD C52F       JSR    VDURET
01883 1CFF BD 1193       JSR    FLAGO    OVERRIDE MOVE
01884 1D02 CE 2C8D       LDX     #MP6    Y POS =
01885 1D05 BD C545       JSR    VDUOUT
01886 1D08 96 0C        LDA    A    YCPOS+3
01887 1D0A 97 04        STA    A    XCPOS+3
01888 1D0C 96 10        LDA    A    YFPOS+3
01889 1D0E 97 08        STA    A    XFPOS+3
01890 1D10 BD 1D3E       JSR    DISNO    DISPLAY POSITION
01891 1D13 CE 2CAA FY3A  LDX     #MP7
01892 1D16 BD C545       JSR    VDUOUT    OK?
01893 1D19 0C          CLC
01894 1D1A BD C5F9 FY4   JSR    KBDCHK
01895 1D1D 24 FB        BCC     FY4
01896 1D1F 81 59        CMP    A    #'Y    Y FOR OK
01897 1D21 27 14        BEQ     FY5
01898 1D23 81 4E        CMP    A    #'N
01899 1D25 27 05        BEQ     FY4A
01900 1D27 BD D551       JSR    WRONG
01901 1D2A 20 E7        BRA     FY3A
01902 1D2C BD 102D FY4A  JSR    SETUP    NOT OK
01903 1D2F BD 115B       JSR    GETPOS
01904 1D32 BD 11B8       JSR    DISPLAY
01905 1D35 20 A3        BRA     FY
01906 1D37 86 3E        FY5    LDA    A    #3E
01907 1D39 B7 A003       STA    A    PIAC2    ERROR DIG. RESET
01908 1D3C 20 9B        BRA     FXY
01909
01910
01911
01912
01913 1D3E 7F 0003 DISNO  CLR     XCPOS+2
01914 1D41 CE 0003      LDX     #XCPOS+2 COARSE POSITION

```


01915	1D44	BD	D961		JSR	BINBCD	
01916	1D47	86	02		LDA A	#02	
01917	1D49	B7	C072		STA A	FF	LEADING 0'S
01918	1D4C	BD	D996		JSR	VDUASC	OUTPUT TO VDU
01919	1D4F	86	2C		LDA A	#',	
01920	1D51	BD	C92F		JSR	VDU	OUTPUT COMMA
01921	1D54	7F	0007		CLR	XFPOS+2	
01922	1D57	CE	0007		LDX	#XFPOS+2	FINE POSITION
01923	1D5A	BD	D961		JSR	BINBCD	
01924	1D5D	86	03		LDA A	#03	
01925	1D5F	B7	C072		STA A	FF	LEADING 0'S
01926	1D62	BD	D996		JSR	VDUASC	OUTPUT TO VDU
01927	1D65	BD	C52F		JSR	VDURET	CR;LF
01928	1D68	39			RTS		
01929				*			
01930				*/MTO/			
01931				*THIS IS THE	SETUP ROUTINE	FOR THE EXTERNAL	
01932				*NON-OVERRIDE	MOVE ROUTINE		
01933	1D69	B6	C0A5	MTO	LDA A	COMBUF+4	EXT PROG CONTROLLED
01934	1D6C	27	07		BEQ	MTOA	NO
01935	1D6E	96	1C		LDA A	XCTARG+9	X OR Y
01936	1D70	97	9A		STA A	FLAGXY	
01937	1D72	7E	1DF6		JMP	MTO11	
01938	1D75	B6	C0A4	MTOA	LDA A	COMBUF+3	TEST COMMAND
01939	1D78	81	48		CMF A	#'H	HOME
01940	1D7A	26	03		BNE	MTO1	
01941	1D7C	7E	1EE5		JMP	HOMETO	
01942	1D7F	81	54	MTO1	CMF A	#'T	TARGET POSITION
01943	1D81	26	03		BNE	MTO2	
01944	1D83	7E	1F3D		JMP	TARGTO	
01945	1D86	CE	2C70	MTO2	LDX	#MP5	
01946	1D89	BD	C545		JSR	VDUOUT	X POS =
01947	1D8C	BD	1D3E		JSR	DISNO	DISPLAY POSITION
01948	1D8F	CE	2CF0	MTO3	LDX	#MP10	
01949	1D92	BD	C545		JSR	VDUOUT	(ENTER Q OR X)
01950	1D95	0C			CLC		
01951	1D96	BD	C5F9	MTO4	JSR	KBDCHK	
01952	1D99	24	FB		BCC	MTO4	
01953	1D9B	81	51		CMF A	#'Q	QUIT
01954	1D9D	26	06		BNE	MTO5	
01955	1D9F	86	FF		LDA A	#FF	
01956	1DA1	97	9A		STA A	FLAGXY	Y AXIS ONLY
01957	1DA3	20	0E		BRA	MTO6	
01958	1DA5	81	58	MTO5	CMF A	#'X	ENTER X AXIS DESTINATION
01959	1DA7	26	E6		BNE	MTO3	DO AGAIN
01960	1DA9	CE	0023		LDX	#XCGOTO	
01961	1DAC	BD	1E25		JSR	ENTV	ENTER FROM KEYBD
01962	1DAF	86	AA		LDA A	#AA	
01963	1DB1	97	9A		STA A	FLAGXY	X AND Y AXIS
01964	1DB3	7D	00A9	MTO6	TST	XSPMK2	WHICH SPEED CONTROLLER
01965	1DB6	26	38		BNE	MTO10	DON'T DISPLAY Y AXIS IF MK2
01966	1DB8	CE	2C8D		LDX	#MP6	
01967	1DBB	BD	C545		JSR	VDUOUT	Y AXIS POS=
01968	1DBE	96	0C		LDA A	YCPOS+3	
01969	1DC0	97	04		STA A	XCPOS+3	
01970	1DC2	96	10		LDA A	YFPOS+3	
01971	1DC4	97	08		STA A	XFPOS+3	
01972	1DC6	BD	1D3E		JSR	DISNO	DISPLAY VALUE


```

01973 1DC9 CE 2F64 MT07 LDX #MP26
01974 1DCC BD C545 JSR VDUOUT Q OR Y
01975 1DCF 0C CLC
01976 1DD0 BD C5F9 MT08 JSR KBDCHK
01977 1DD3 24 FB BCC MT08
01978 1DD5 81 51 CMP A #'Q QUIT
01979 1DD7 27 0C BEQ MT09
01980 1DD9 81 59 CMP A #'Y
01981 1ddb 26 EC BNE MT07 DO AGAIN
01982 1DDD CE 002B LDX #YCGOTO
01983 1DE0 BD 1E25 JSR ENTV ENTER VALUE
01984 1DE3 20 11 BRA MT011
01985 1DE5 96 9A MT09 LDA A FLAGXY
01986 1DE7 7F 009A CLR FLAGXY
01987 1DEA 81 AA CMP A #$AA X AND Y
01988 1DEC 27 08 BEQ MT011
01989 1DEE 20 19 BRA MT012
01990 1DF0 BD 1E0A MT010 JSR XSP DISPLAY SPEED
01991 1DF3 7F 009A CLR FLAGXY X ONLY
01992 1DF6 86 04 MT011 LDA A #$04
01993 1DF8 97 94 STA A FLAG MODE
01994 1DFA BD 2597 JSR SEL SET LEDS AND REGS
01995 1DFD CE 2EAF LDX #MP21
01996 1E00 BD C545 JSR VDUOUT (STOP)
01997 1E03 BD 115B JSR GETPOS
01998 1E06 BD 2705 JSR EG MOVE
01999 1E09 39 MT012 RTS
02000 *
02001 1E0A CE 2D4E XSP LDX #MP12
02002 1E0D BD C545 JSR VDUOUT (SPEED =)
02003 1E10 96 58 LDA A XFSPD+1
02004 1E12 88 30 EOR A #$30 CONVERT TO ASCII
02005 1E14 BD C92F JSR VDU OUTPUT TENS TO VDU
02006 1E17 96 59 LDA A XFSPD+2
02007 1E19 88 30 EOR A #$30 CONVERT TO ASCII
02008 1E1B BD C92F JSR VDU OUTPUT UNITS
02009 1E1E CE 2D6B LDX #MP13
02010 1E21 BD C545 JSR VDUOUT (CM/SEC)
02011 1E24 39 RTS
02012 *
02013 */ENTV/
02014 *THIS ENTERS A 5 DIGET VALUE FROM THE KEYBOARD
02015 *AND STORES IT IN THE INDEX REG
02016 1E25 5F ENTV CLR B
02017 1E26 DF 7F STX REG5 REMEMBER X REG POINTER
02018 1E28 BD C5C3 ENTV1 JSR CHARIN
02019 1E2B 81 7F CMP A #$7F DELETE 1 CHAR
02020 1E2D 26 03 BNE ENTV1A
02021 1E2F 7E 1ECD JMP COR2
02022 1E32 81 0D ENTV1A CMP A #$0D RETURN
02023 1E34 26 03 BNE ENTV2
02024 1E36 7E 1E6D JMP COR
02025 1E39 81 2C ENTV2 CMP A #', COMMA
02026 1E3B 26 03 BNE ENTV3
02027 1E3D 7E 1E9D JMP COR1
02028 1E40 36 ENTV3 PSH A
02029 1E41 84 30 AND A #$30 TEST VALID INPUT
02030 1E43 81 30 CMP A #$30

```


02031	1E45	27	08		BEQ	ENTV4	
02032	1E47	32			PUL A		
02033	1E48	86	7F	ENTV3A	LDA A	#\$7F	
02034	1E4A	BD	C92F		JSR	VDU	DELETE LAST ENTRY ON VDU
02035	1E4D	20	D9		BRA	ENTV1	DO AGAIN
02036	1E4F	32		ENTV4	PUL A		
02037	1E50	84	0F		AND A	#\$0F	MASK
02038	1E52	A7	00		STA A	0,X	
02039	1E54	08			INX		
02040	1E55	5C			INC B		
02041	1E56	C1	05		CMP B	#\$05	END OF DESIMAL
02042	1E58	27	0D		BEQ	ENTV6	
02043	1E5A	C1	03		CMP B	#\$03	END OF INTEGER
02044	1E5C	26	CA		BNE	ENTV1	ENTER ANOTHER NUMBER
02045	1E5E	86	2C	ENTV5	LDA A	#',	
02046	1E60	BD	C92F		JSR	VDU	DISPLAY A COMMA
02047	1E63	08			INX		
02048	1E64	08			INX		
02049	1E65	20	C1		BRA	ENTV1	
02050	1E67	BD	C52F	ENTV6	JSR	VDURET	CR;LF
02051	1E6A	DE	7F		LDX	REG5	RESTORE X REGISTER
02052	1E6C	39			RTS		
02053				*			
02054	1E6D	DE	7F	COR	LDX	REG5	RESTORE X REG
02055	1E6F	5D			TST B		
02056	1E70	27	1E		BEQ	C01	NO VALUES ENTERED
02057	1E72	C1	01		CMP B	#\$01	HAS ONE DIGET BEEN ENTERED
02058	1E74	27	0A		BEQ	C02	
02059	1E76	C1	02		CMP B	#\$02	HAS 2 DIGETS BEEN ENTERED
02060	1E78	27	0C		BEQ	C03	
02061	1E7A	C1	03		CMP B	#\$03	HAS 3 BEEN ENTERED
02062	1E7C	27	18		BEQ	C04	
02063	1E7E	20	18		BRA	C05	
02064	1E80	A6	00	C02	LDA A	0,X	
02065	1E82	A7	02		STA A	2,X	PUT VALUE IN CORRECT REGISTER
02066	1E84	20	0C		BRA	C06	
02067	1E86	A6	01	C03	LDA A	1,X	
02068	1E88	A7	02		STA A	2,X	
02069	1E8A	A6	00		LDA A	0,X	
02070	1E8C	A7	01		STA A	1,X	
02071	1E8E	20	04		BRA	C07	
02072	1E90	6F	02	C01	CLR	2,X	
02073	1E92	6F	01	C06	CLR	1,X	
02074	1E94	6F	00	C07	CLR	0,X	
02075	1E96	6F	05	C04	CLR	5,X	
02076	1E98	6F	06	C05	CLR	6,X	
02077	1E9A	7E	1E67		JMP	ENTV6	
02078				*			
02079	1E9D	DE	7F	COR1	LDX	REG5	RESTORE X REG
02080	1E9F	5D			TST B		TEST WICH DIGET
02081	1EA0	27	1B		BEQ	C011	ONLY DESIMAL
02082	1EA2	C1	01		CMP B	#\$01	
02083	1EA4	27	07		BEQ	C022	ONE INTEGER
02084	1EA6	C1	02		CMP B	#\$02	
02085	1EA8	27	09		BEQ	C033	TWO INTEGERS
02086	1EAA	7E	1E48		JMP	ENTV3A	FAULT
02087	1EAD	A6	00	C022	LDA A	0,X	
02088	1EAF	A7	02		STA A	2,X	

02089	1EB1	20	0C		BRA	C044	
02090	1EB3	A6	01	C033	LDA A	1,X	
02091	1EB5	A7	02		STA A	2,X	
02092	1EB7	A6	00		LDA A	0,X	
02093	1EB9	A7	01		STA A	1,X	
02094	1EBB	20	04		BRA	C055	
02095	1EBD	6F	02	C011	CLR	2,X	PUT ZERO INTO LEADING REGISTE
02096	1EBF	6F	01	C044	CLR	1,X	
02097	1EC1	6F	00	C055	CLR	0,X	
02098	1EC3	C6	03		LDA B	#03	
02099	1EC5	08			INX		
02100	1EC6	08			INX		
02101	1EC7	08			INX		
02102	1EC8	08			INX		
02103	1EC9	08			INX		
02104	1ECA	7E	1E28		JMP	ENTV1	
02105				*			
02106	1ECD	5D		COR2	TST B		DELETE ROUTINE
02107	1ECE	27	08		BEQ	COR2A	
02108	1ED0	C1	03		CMP B	#03	TEST CHARACTER
02109	1ED2	27	07		BEQ	COR2B	
02110	1ED4	5A			DEC B		DEC. CHAR. COUNTER
02111	1ED5	09			DEX		
02112	1ED6	6F	00		CLR	0,X	CLEAR CHARACTER REGISTER
02113	1ED8	7E	1E28	COR2A	JMP	ENTV1	ENTER AGAIN
02114	1EDB	DE	7F	COR2B	LDX	REG5	RESTORE INDEX REG
02115	1EDD	08			INX		
02116	1EDE	08			INX		
02117	1EDF	6F	00		CLR	0,X	CLEAR CHARACTER
02118	1EE1	5A			DEC B		
02119	1EE2	7E	1E48		JMP	ENTV3A	DELETE COMMA
02120				*			
02121				*HOMETO/			
02122				*THIS IS THE	SETUP ROUTINE FOR THE CONSOLE CONTROLLE		
02123				*MOVE TO THE	'HOME' POSITION		
02124	1EE5	CE	2C70	HOMETO	LDX	#MP5	
02125	1EE8	BD	C545		JSR	VDUOUT	(X POS =)
02126	1EEB	BD	1D3E		JSR	DISNO	DISPLAY POSITION
02127	1EEE	CE	2D31		LDX	#MP11	
02128	1EF1	BD	C545		JSR	VDUOUT	
02129	1EF4	96	46		LDA A	XCHOME+3	
02130	1EF6	97	04		STA A	XCPOS+3	
02131	1EF8	96	4A		LDA A	XFHOME+3	
02132	1EFA	97	08		STA A	XFPOS+3	
02133	1EFC	BD	1D3E		JSR	DISNO	DISPLAY HOME POSITION
02134	1EFF	7D	00A9		TST	XSPMK2	
02135	1F02	26	22		BNE	HOX	BRANCH FOR MK2 SPD.CON.
02136	1F04	CE	2C8D		LDX	#MP6	
02137	1F07	BD	C545		JSR	VDUOUT	(Y POS=)
02138	1F0A	96	0C		LDA A	YCPOS+3	
02139	1F0C	97	04		STA A	XCPOS+3	
02140	1F0E	96	10		LDA A	YFPOS+3	
02141	1F10	97	08		STA A	XFPOS+3	
02142	1F12	BD	1D3E		JSR	DISNO	
02143	1F15	CE	2ED3		LDX	#MP22	
02144	1F18	BD	C545		JSR	VDUOUT	(Y HOME=)
02145	1F1B	96	4E		LDA A	YCHOME+3	
02146	1F1D	97	04		STA A	XCPOS+3	


```

02147 1F1F 96 52          LDA A   YFHOME+3
02148 1F21 97 08          STA A   XFPOS+3
02149 1F23 BD 1D3E        JSR      DISNO      DISPLAY VALUE
02150 1F26 CE 2EAF HOX    LDX      #MP21
02151 1F29 BD C545        JSR      VDUOUT    (ABORT)
02152 1F2C BD 115B        JSR      GETPOS    GET LATEST POSITION
02153 1F2F 86 AA          LDA A   #$AA
02154 1F31 97 9A          STA A   FLAGXY
02155 1F33 86 01          LDA A   #$01
02156 1F35 97 94          STA A   FLAG
02157 1F37 BD 2597        JSR      SEL
02158 1F3A 7E 2705        JMP      EG
02159                      *
02160 1F3D 5F              TARGTO CLR B
02161 1F3E CE 0013        LDX      #XCTARG
02162 1F41 A6 00          TARG   LDA A   0,X      TRANSFER TARGET TO MOVE TO
02163 1F43 A7 10          STA A   16,X
02164 1F45 08              INX
02165 1F46 5C              INC B
02166 1F47 C1 0F          CMP B   #$0F      HAS 16 BYTES BEEN TRANFERED?
02167 1F49 26 F6          BNE     TARG
02168 1F4B CE 2DF8        LDX      #MP18
02169 1F4E BD C545        JSR      VDUOUT    (DEST =)
02170 1F51 96 26          LDA A   XCGOTO+3
02171 1F53 97 04          STA A   XCPOS+3
02172 1F55 96 2A          LDA A   XFGOTO+3
02173 1F57 97 08          STA A   XFPOS+3
02174 1F59 BD 1D3E        JSR      DISNO      DISPLAY VALUE
02175 1F5C 7D 00A9        TST      XSPMK2
02176 1F5F 27 03          BEQ     TQ
02177 1F61 7E 1DF0        JMP      MT010
02178 1F64 CE 2F47 TQ     LDX      #MP25
02179 1F67 BD C545        JSR      VDUOUT    (Y TARG=)
02180 1F6A 96 2E          LDA A   YCGOTO+3
02181 1F6C 97 04          STA A   XCPOS+3
02182 1F6E 96 32          LDA A   YFGOTO+3
02183 1F70 97 08          STA A   XFPOS+3
02184 1F72 BD 1D3E        JSR      DISNO      DISPLAY
02185 1F75 86 AA          LDA A   #$AA
02186 1F77 97 9A          STA A   FLAGXY
02187 1F79 7E 1DF6        JMP      MT011
02188                      *
02189                      */PAR/
02190                      *THIS ROUTINE DISPLAYS AND CHANGES THE PLATFORM PARA
02191 1F7C 7F 0000 PAR      CLR      TEMP
02192 1F7F BD C52F          JSR      VDURET
02193 1F82 BD 115B          JSR      GETPOS
02194 1F85 CE 2C70          LDX      #MP5
02195 1F88 BD C545          JSR      VDUOUT    (POSITION =)
02196 1F8B BD 1D3E          JSR      DISNO      DISPLAY VALUE
02197 1F8E CE 2D97          LDX      #MP15
02198 1F91 BD C545          JSR      VDUOUT    (LIMITS)
02199 1F94 B6 A0B0          LDA A   ESXCL      X COARSE LOWER LIMIT
02200 1F97 97 04          STA A   XCPOS+3
02201 1F99 B6 A0B1          LDA A   ESXFL      X FINE LOWER LIMIT
02202 1F9C 97 08          STA A   XFPOS+3
02203 1F9E BD 1D3E          JSR      DISNO      DISPLAY VALUE
02204 1FA1 CE 2DB4          LDX      #MP15A

```


02205	1FA4	BD	C545		JSR	VDUOUT	(LIMITS HI)
02206	1FA7	B6	A0B2		LDA A	ESXCH	X COARSE HIGHER LIMIT
02207	1FAA	97	04		STA A	XCPOS+3	
02208	1FAC	B6	A0B3		LDA A	ESXFH	X FINE HIGHER LIMIT
02209	1FAF	97	08		STA A	XFPOS+3	
02210	1FB1	BD	1D3E		JSR	DISNO	
02211	1FB4	CE	2D31	PARA	LDX	#MP11	
02212	1FB7	BD	C545		JSR	VDUOUT	(HOME)
02213	1FBA	96	46		LDA A	XCHOME+3	
02214	1FBC	97	04		STA A	XCPOS+3	
02215	1FBE	96	4A		LDA A	XFHOME+3	
02216	1FC0	97	08		STA A	XFPOS+3	
02217	1FC2	BD	1D3E		JSR	DISNO	DISPLAY HOME VALUE
02218	1FC5	7D	0000		TST	TEMP	
02219	1FC8	27	2E		BEQ	PARR2	
02220	1FCA	CE	2DD1	PAR1	LDX	#MP16	
02221	1FCD	BD	C545		JSR	VDUOUT	(CHANGE?)
02222	1FD0	0C			CLC		
02223	1FD1	BD	C5F9	PAR1A	JSR	KBDCHK	
02224	1FD4	24	FB		BCC	PAR1A	
02225	1FD6	81	4E		CMP A	#'N	NO
02226	1FD8	27	1B		BEQ	PAR21	
02227	1FDA	81	59		CMP A	#'Y	YES
02228	1FDC	27	05		BEQ	PAR1B	
02229	1FDE	BD	D551		JSR	WRONG	
02230	1FE1	20	E7		BRA	PAR1	
02231	1FE3	BD	C52F	PAR1B	JSR	VDURET	
02232	1FE6	CE	2D31		LDX	#MP11	
02233	1FE9	BD	C545		JSR	VDUOUT	
02234	1FEC	CE	0043		LDX	#XCHOME	
02235	1FEF	BD	1E25		JSR	ENTV	ENTER VALUES
02236	1FF2	BD	1203		JSR	CHEX	CHANGE TO HEX
02237	1FF5	BD	C52F	PAR21	JSR	VDURET	
02238	1FF8	CE	2FA4	PARR2	LDX	#MP27	
02239	1FFB	BD	C545		JSR	VDUOUT	TARGET
02240	1FFE	96	16		LDA A	XCTARG+3	
02241	2000	97	04		STA A	XCPOS+3	
02242	2002	96	1A		LDA A	XFTARG+3	
02243	2004	97	08		STA A	XFPOS+3	
02244	2006	BD	1D3E		JSR	DISNO	DISPLAY VALUE
02245	2009	7D	0000		TST	TEMP	
02246	200C	27	2E		BEQ	PAR2	
02247	200E	CE	2DD1	PA1	LDX	#MP16	
02248	2011	BD	C545		JSR	VDUOUT	(CHANGE?)
02249	2014	0C			CLC		
02250	2015	BD	C5F9	PA2	JSR	KBDCHK	
02251	2018	24	FB		BCC	PA2	
02252	201A	81	4E		CMP A	#'N	
02253	201C	27	1B		BEQ	PA3	
02254	201E	81	59		CMP A	#'Y	
02255	2020	27	05		BEQ	PA4	
02256	2022	BD	D551		JSR	WRONG	TRY AGAIN
02257	2025	20	E7		BRA	PA1	
02258	2027	BD	C52F	PA4	JSR	VDURET	
02259	202A	CE	2FA4		LDX	#MP27	
02260	202D	BD	C545		JSR	VDUOUT	(TARGET=)
02261	2030	CE	0013		LDX	#XCTARG	
02262	2033	BD	1E25		JSR	ENTV	ENTER VALUE

02263	2036	BD	1203		JSR	CHEX	
02264	2039	BD	C52F	PA3	JSR	VDURET	
02265	203C	CE	2D7A	PAR2	LDX	#MP14	
02266	203F	BD	C545		JSR	VDUOUT	(SCALE)
02267	2042	96	36		LDA	A XCSCCL+3	X SCALE COARSE
02268	2044	97	04		STA	A XCPOS+3	
02269	2046	96	3A		LDA	A XFSCCL+3	
02270	2048	97	08		STA	A XFPOS+3	
02271	204A	BD	1D3E		JSR	DISNO	DISPLAY VALUE
02272	204D	7D	0000		TST	TEMP	
02273	2050	27	2E		BEQ	PAR4	
02274	2052	CE	2DD1	PAR3	LDX	#MP16	
02275	2055	BD	C545		JSR	VDUOUT	(CHANGE ?)
02276	2058	0C			CLC		
02277	2059	BD	C5F9	PAR3A	JSR	KBDCHK	TEST KEYBOARD
02278	205C	24	FB		BCC	PAR3A	
02279	205E	81	4E		CMP	A #'N	N....NO
02280	2060	27	1B		BEQ	PAR41	
02281	2062	81	59		CMP	A #'Y	YES
02282	2064	27	05		BEQ	PAR3B	
02283	2066	BD	D551		JSR	WRONG	
02284	2069	20	E7		BRA	PAR3	
02285	206B	BD	C52F	PAR3B	JSR	VDURET	
02286	206E	CE	2D7A		LDX	#MP14	
02287	2071	BD	C545		JSR	VDUOUT	
02288	2074	CE	0033		LDX	#XCSCCL	
02289	2077	BD	1E25		JSR	ENTV	
02290	207A	BD	1203		JSR	CHEX	
02291	207D	BD	C52F	PAR41	JSR	VDURET	
02292	2080	7D	00A9	PAR4	TST	XSPMK2	
02293	2083	26	03		BNE	NOYP	
02294	2085	7E	2114		JMP	YPAR	
02295	2088	BD	1E0A	NOYP	JSR	XSP	DISPLAY SPEED
02296	208B	7D	0000		TST	TEMP	
02297	208E	27	5E		BEQ	PAR6	
02298	2090	CE	2DD1	PAR4A	LDX	#MP16	
02299	2093	BD	C545		JSR	VDUOUT	(CHANGE ?)
02300	2096	0C			CLC		
02301	2097	BD	C5F9	PAR5	JSR	KBDCHK	TEST KEYBD
02302	209A	24	FB		BCC	PAR5	
02303	209C	81	4E		CMP	A #'N	NO
02304	209E	27	48		BEQ	PAR5G	
02305	20A0	81	59		CMP	A #'Y	
02306	20A2	27	05		BEQ	PAR5H	
02307	20A4	BD	D551		JSR	WRONG	
02308	20A7	20	E7		BRA	PAR4A	
02309	20A9	26	43	PAR5H	BNE	PAR6	
02310	20AB	CE	2D4E		LDX	#MP12	
02311	20AE	BD	C545		JSR	VDUOUT	
02312	20B1	5F			CLR	B	DIGET COUNTER
02313	20B2	BD	C5C3	PAR5A	JSR	CHARIN	
02314	20B5	36			PSH	A	
02315	20B6	84	30		AND	A #\$30	
02316	20B8	81	30		CMP	A #\$30	NUMBERS ONLY
02317	20BA	27	08		BEQ	PAR5B	
02318	20BC	32			PUL	A	
02319	20BD	86	7F	PAR5E	LDA	A #\$7F	
02320	20BF	BD	C92F		JSR	VDU	CLEAR LAST ENTERY

02321	20C2	20	EE		BRA	PAR5A	GET ANOTHER DIGET
02322	20C4	5D		PAR5B	TST B		2ND DIGET?
02323	20C5	26	16		BNE	PAR5C	YES
02324	20C7	32			PUL A		
02325	20C8	84	0F		AND A	#\$0F	
02326	20CA	27	0B		BEQ	PAR5D	
02327	20CC	81	01		CMP A	#\$01	10 CM/SEC MAX
02328	20CE	26	ED		BNE	PAR5E	CLEAR AND DO AGAIN
02329	20D0	97	58		STA A	XFSPD+1	
02330	20D2	7F	0059		CLR	XFSPD+2	
02331	20D5	20	0B		BRA	PAR5F	
02332	20D7	5C		PAR5D	INC B		INCREMENT DIGET COUNTER
02333	20D8	7F	0058		CLR	XFSPD+1	
02334	20DB	20	D5		BRA	PAR5A	GET ANOTHER INPUT
02335	20DD	32		PAR5C	PUL A		
02336	20DE	84	0F		AND A	#\$0F	
02337	20E0	97	59		STA A	XFSPD+2	
02338	20E2	CE	2D6B	PAR5F	LDX	#MP13	
02339	20E5	BD	C545		JSR	VDUOUT	
02340	20E8	7F	0000	PAR5G	CLR	TEMP	
02341	20EB	7E	1F7C		JMP	PAR	
02342	20EE	BD	C52F	PAR6	JSR	VDURET	
02343	20F1	CE	2DD1		LDX	#MP16	
02344	20F4	BD	C545		JSR	VDUOUT	CHANGE?
02345	20F7	0C			CLC		
02346	20F8	BD	C5F9	PAR6A	JSR	KBDCHK	
02347	20FB	24	FB		BCC	PAR6A	
02348	20FD	81	4E		CMP A	#'N	
02349	20FF	27	12		BEQ	PAR7	
02350	2101	81	59		CMP A	#'Y	
02351	2103	26	09		BNE	PAR6B	
02352	2105	BD	C52F		JSR	VDURET	
02353	2108	73	0000		COM	TEMP	
02354	210B	7E	1FB4		JMP	PARA	
02355	210E	BD	D551	PAR6B	JSR	WRONG	
02356	2111	20	DB		BRA	PAR6	
02357	2113	39		PAR7	RTS		
02358				*			
02359				*CHANGE Y PARAMETERS			
02360				*			
02361	2114	7D	0000	YPAR	TST	TEMP	
02362	2117	26	37		BNE	YPA	
02363	2119	CE	2C8D		LDX	#MP6	
02364	211C	BD	C545		JSR	VDUOUT	(Y POS=)
02365	211F	96	0C		LDA A	YCPOS+3	
02366	2121	97	04		STA A	XCPOS+3	
02367	2123	96	10		LDA A	YFPOS+3	
02368	2125	97	08		STA A	XFPOS+3	
02369	2127	BD	1D3E		JSR	DISNO	DISPLAY VALUE
02370	212A	CE	2F0D		LDX	#MP24	
02371	212D	BD	C545		JSR	VDUOUT	(Y LIMITS)
02372	2130	B6	A0B4		LDA A	ESYCL	
02373	2133	97	04		STA A	XCPOS+3	
02374	2135	B6	A0B5		LDA A	ESYFL	
02375	2138	97	08		STA A	XFPOS+3	
02376	213A	BD	1D3E		JSR	DISNO	DIS. VALUE
02377	213D	CE	2F2A		LDX	#MP24A	
02378	2140	BD	C545		JSR	VDUOUT	LIMITS

02379	2143	B6	A0B6		LDA	A	ESYCH	
02380	2146	97	04		STA	A	XCPOS+3	
02381	2148	B6	A0B7		LDA	A	ESYFH	
02382	214B	97	08		STA	A	XFPOS+3	
02383	214D	BD	1D3E		JSR		DISNO	
02384	2150	CE	2ED3	YPA	LDX		#MP22	
02385	2153	BD	C545		JSR		VDUOUT	(Y HOME=)
02386	2156	96	4E		LDA	A	YCHOME+3	
02387	2158	97	04		STA	A	XCPOS+3	
02388	215A	96	52		LDA	A	YFHOME+3	
02389	215C	97	08		STA	A	XFPOS+3	
02390	215E	BD	1D3E		JSR		DISNO	VALUE
02391	2161	7D	0000		TST		TEMP	
02392	2164	27	2E		BEQ		YP22	
02393	2166	CE	2DD1	YP1	LDX		#MP16	
02394	2169	BD	C545		JSR		VDUOUT	(CHANGE?)
02395	216C	0C			CLC			
02396	216D	BD	C5F9	YP1A	JSR		KBDCHK	
02397	2170	24	FB		BCC		YP1A	
02398	2172	81	4E		CMP	A	#'N	NO
02399	2174	27	1B		BEQ		YP1C	
02400	2176	81	59		CMP	A	#'Y	YES
02401	2178	27	05		BEQ		YP1B	
02402	217A	BD	D551		JSR		WRONG	TRY AGAIN
02403	217D	20	E7		BRA		YP1	
02404	217F	BD	C52F	YP1B	JSR		VDURET	
02405	2182	CE	2ED3		LDX		#MP22	
02406	2185	BD	C545		JSR		VDUOUT	Y HOME=)
02407	2188	CE	004B		LDX		#YCHOME	
02408	218B	BD	1E25		JSR		ENTV	ENTER NEW VALUE
02409	218E	BD	1203		JSR		CHEX	
02410	2191	BD	C52F	YP1C	JSR		VDURET	
02411	2194	CE	2FC1	YP22	LDX		#MP28	
02412	2197	BD	C545		JSR		VDUOUT	(Y TARG=)
02413	219A	96	1E		LDA	A	YCTARG+3	
02414	219C	97	04		STA	A	XCPOS+3	
02415	219E	96	22		LDA	A	YFTARG+3	
02416	21A0	97	08		STA	A	XFPOS+3	
02417	21A2	BD	1D3E		JSR		DISNO	DISPLAY VALUE
02418	21A5	7D	0000		TST		TEMP	
02419	21A8	27	2E		BEQ		YP2	
02420	21AA	CE	2DD1	YP11	LDX		#MP16	
02421	21AD	BD	C545		JSR		VDUOUT	(CHANGE?)
02422	21B0	0C			CLC			
02423	21B1	BD	C5F9	YP11A	JSR		KBDCHK	
02424	21B4	24	FB		BCC		YP11A	
02425	21B6	81	4E		CMP	A	#'N	NO
02426	21B8	27	1B		BEQ		YP11C	
02427	21BA	81	59		CMP	A	#'Y	YES
02428	21BC	27	05		BEQ		YP11B	
02429	21BE	BD	D551		JSR		WRONG	
02430	21C1	20	E7		BRA		YP11	
02431	21C3	BD	C52F	YP11B	JSR		VDURET	
02432	21C6	CE	2FC1		LDX		#MP28	
02433	21C9	BD	C545		JSR		VDUOUT	(TARG =)
02434	21CC	CE	001B		LDX		#YCTARG	
02435	21CF	BD	1E25		JSR		ENTV	ENTER VALUE
02436	21D2	BD	1203		JSR		CHEX	


```

02437 21D5 BD C52F YP11C JSR VDURET
02438 21D8 CE 2EF0 YP2 LDX HMP23
02439 21DB BD C545 JSR VDUOUT (SCALE=)
02440 21DE 96 3E LDA A YCSCL+3
02441 21E0 97 04 STA A XCPOS+3
02442 21E2 96 42 LDA A YFSCL+3
02443 21E4 97 08 STA A XFPOS+3
02444 21E6 BD 1D3E JSR DISNO DISPLAY VALUE
02445 21E9 7D 0000 TST TEMP
02446 21EC 27 2B BEQ YP4
02447 21EE CE 2DD1 YP3 LDX HMP16
02448 21F1 BD C545 JSR VDUOUT (CHANGE?)
02449 21F4 0C CLC
02450 21F5 BD C5F9 YP3A JSR KBDCHK
02451 21F8 24 FB BCC YP3A
02452 21FA 81 4E CMP A #'N NO
02453 21FC 27 18 BEQ YP3C
02454 21FE 81 59 CMP A #'Y YES
02455 2200 27 05 BEQ YP3B
02456 2202 BD D551 JSR WRONG TRY AGAIN
02457 2205 20 E7 BRA YP3
02458 2207 CE 2EF0 YP3B LDX HMP23
02459 220A BD C545 JSR VDUOUT SCALE=
02460 220D CE 003B LDX HYCSC
02461 2210 BD 1E25 JSR ENTV ENTER NEW VALUE
02462 2213 BD 1203 JSR CHEX
02463 2216 7E 20E8 YP3C JMP PAR5G
02464 2219 7E 20EE YP4 JMP PAR6
02465 *
02466 *THIS ROUTINE INCREASES THE GOTO POSITION
02467 *BY THE VALUE IN SCALE
02468 221C 7F 0000 SINC CLR TEMP
02469 221F B6 C0A5 SINCD LDA A COMBUF+4 EXT PROG CONTROLLED
02470 2222 27 14 BEQ SINC1 NO
02471 2224 96 1C LDA A XCTARG+9 X OR Y REQUEST
02472 2226 97 9A STA A FLAGXY
02473 2228 7D 0000 TST TEMP
02474 222B 26 05 BNE SINC2
02475 222D BD 26BA JSR INC INCREMENT
02476 2230 20 03 BRA SINC3
02477 2232 BD 26E1 SINC2 JSR DEC DECREMENT
02478 2235 7E 22A5 SINC3 JMP IX4
02479 2238 7D 00A9 SINC1 TST XSPMK2 MK1 OR MK2 SPEED CONT.
02480 223B 27 06 BEQ SINC4
02481 223D BD 1E0A JSR XSP DISPLAY SPEED
02482 2240 7E 2250 JMP IX IF MK2 X AXIS ONLY
02483 2243 B6 C0A4 SINC4 LDA A COMBUF+3
02484 2246 81 58 CMP A #$58 (X)
02485 2248 27 06 BEQ IX
02486 224A 81 59 CMP A #$59 (Y)
02487 224C 26 5D BNE IX5 WRONG KEY
02488 224E 20 73 BRA DX
02489 2250 BD 115B IX JSR GETPOS
02490 2253 CE 2C70 LDX HMP5
02491 2256 BD C545 JSR VDUOUT (X POS =)
02492 2259 BD 1D3E JSR DISNO
02493 225C 86 04 LDA A #$04 SET FLAGS
02494 225E 97 94 STA A FLAG

```


02495	2260	7F	009A		CLR	FLAGXY	
02496	2263	7D	0000	IX1	TST	TEMP	INC OR DEC
02497	2266	27	05		BEQ	IX1A	INC
02498	2268	BD	26E1		JSR	DEC	DECREMENT
02499	226B	20	03		BRA	IX1B	
02500	226D	BD	26BA	IX1A	JSR	INC	INCREMENT
02501	2270	CE	2DF8	IX1B	LDX	#MP18	
02502	2273	BD	C545		JSR	VDUOUT	(DEST. IS)
02503	2276	CE	0023		LDX	#XCGOTO	
02504	2279	BD	1203		JSR	CHEX	
02505	227C	96	26		LDA A	XCGOTO+3	
02506	227E	97	04		STA A	XCPOS+3	
02507	2280	96	2A		LDA A	XFGOTO+3	
02508	2282	97	08		STA A	XFPOS+3	
02509	2284	BD	1D3E		JSR	DISNO	DISPLAY VALUE
02510	2287	CE	2E15	IX2	LDX	#MP19	
02511	228A	BD	C545		JSR	VDUOUT	(STRIKE G;I;Q)
02512	228D	0C			CLC		
02513	228E	BD	C5F9	IX3	JSR	KBDCHK	
02514	2291	24	FB		BCC	IX3	
02515	2293	81	44		CMP A	#\$44	(D)
02516	2295	27	15		BEQ	IX6	DECREMENT AGAIN
02517	2297	81	49		CMP A	#\$49	(I)
02518	2299	27	1D		BEQ	IX7	INCREMENT AGAIN
02519	229B	81	47		CMP A	#\$47	(G)
02520	229D	27	06		BEQ	IX4	MOVE TO POSITON
02521	229F	81	51		CMP A	#\$51	(Q)
02522	22A1	27	08		BEQ	IX5	QUIT
02523	22A3	20	E2		BRA	IX2	WRONG KEY
02524	22A5	BD	115B	IX4	JSR	GETPOS	UPDATE POSITION
02525	22A8	BD	2705		JSR	EG	
02526	22AB	39		IX5	RTS		
02527	22AC	86	FF	IX6	LDA A	#\$FF	
02528	22AE	97	00		STA A	TEMP	
02529	22B0	7D	009A		TST	FLAGXY	
02530	22B3	26	0E		BNE	DX	
02531	22B5	7E	2250		JMP	IX	
02532	22B8	7F	0000	IX7	CLR	TEMP	
02533	22BB	7D	009A		TST	FLAGXY	
02534	22BE	26	1C		BNE	DX1	
02535	22C0	7E	2263		JMP	IX1	
02536				*			
02537	22C3	CE	2C8D	DX	LDX	#MP6	
02538	22C6	BD	C545		JSR	VDUOUT	(Y POS=)
02539	22C9	96	0C		LDA A	YCPOS+3	
02540	22CB	97	04		STA A	XCPOS+3	
02541	22CD	96	10		LDA A	YFPOS+3	
02542	22CF	97	08		STA A	XFPOS+3	
02543	22D1	BD	1D3E		JSR	DISNO	DISPLAY VALUE
02544	22D4	86	04		LDA A	#\$04	
02545	22D6	97	94		STA A	FLAG	SET MODE
02546	22D8	86	FF		LDA A	#\$FF	
02547	22DA	97	9A		STA A	FLAGXY	SET Y AXIS FLAG
02548	22DC	7D	0000	DX1	TST	TEMP	
02549	22DF	27	05		BEQ	DX1A	
02550	22E1	BD	26E1		JSR	DEC	DO DECREMENT ROUTINE
02551	22E4	20	03		BRA	DX1B	
02552	22E6	BD	26BA	DX1A	JSR	INC	DO INCREMENT ROUTINE


```

02553 22E9 CE 2DF8 DX1B LDX      #MP18
02554 22EC BD C545      JSR      VDUOUT      (Y DEST.=)
02555 22EF CE 002B      LDX      #YCGOTO
02556 22F2 BD 1203      JSR      CHEX
02557 22F5 96 2E        LDA A    YCGOTO+3
02558 22F7 97 04        STA A    XCPOS+3
02559 22F9 96 32        LDA A    YFGOTO+3
02560 22FB 97 08        STA A    XFPOS+3
02561 22FD BD 1D3E      JSR      DISNO      DISPLAY Y DESTINATION
02562 2300 20 85        BRA      IX2
02563                      *
02564 2302 86 FF      SDEC    LDA A    #$FF
02565 2304 97 00      STA A    TEMP      DEC SCALE FLAG
02566 2306 7E 221F      JMP      SINCD
02567                      *
02568                      *EXPORG THIS ENTERS VALUES FROM AN EXTERNAL
02569                      *PROGRAM AND STORES IT IN XCTARG
02570                      *
02571 2309 CE 0013 EXPROG LDX      #XCTARG
02572 230C BD DE48      JSR      GETNMM      GET 12 BYTES OF INFO
02573 230F CE 0013      LDX      #XCTARG
02574 2312 A6 00        LDA A    0,X      TRANSFER TO CORRECT REGISTERS
02575 2314 97 26        STA A    XCGOTO+3
02576 2316 A6 01        LDA A    1,X
02577 2318 97 2A        STA A    XFGOTO+3
02578 231A A6 02        LDA A    2,X
02579 231C 97 2E        STA A    YCGOTO+3
02580 231E A6 03        LDA A    3,X
02581 2320 97 32        STA A    YFGOTO+3
02582 2322 A6 04        LDA A    4,X
02583 2324 97 36        STA A    XCSCCL+3
02584 2326 A6 05        LDA A    5,X
02585 2328 97 3A        STA A    XFSCCL+3
02586 232A A6 06        LDA A    6,X
02587 232C 97 3E        STA A    YCSCCL+3
02588 232E A6 07        LDA A    7,X
02589 2330 97 42        STA A    YFSCCL+3
02590 2332 A6 09        LDA A    9,X
02591 2334 97 5A        STA A    XFSPD+3
02592 2336 CE 0023      LDX      #XCGOTO      CONVERT FROM HEX TO DES
02593 2339 BD 1241      JSR      CON
02594 233C CE 0027      LDX      #XFGOTO
02595 233F BD 1241      JSR      CON
02596 2342 CE 002B      LDX      #YCGOTO
02597 2345 BD 1241      JSR      CON
02598 2348 CE 002F      LDX      #YFGOTO
02599 234B BD 1241      JSR      CON
02600 234E CE 0033      LDX      #XCSCCL
02601 2351 BD 1241      JSR      CON
02602 2354 CE 0037      LDX      #XFSCCL
02603 2357 BD 1241      JSR      CON
02604 235A CE 003B      LDX      #YCSCCL
02605 235D BD 1241      JSR      CON
02606 2360 CE 003F      LDX      #YFSCCL
02607 2363 BD 1241      JSR      CON
02608 2366 CE 0057      LDX      #XFSPD
02609 2369 BD 1241      JSR      CON
02610 236C 7E 1C1D      JMP      CYA

```



```

02611          OPT      L
02612          *
02613          NAM      MATH
02614          *
02615          * IN THE FOLLOWING SUBROUTINES INVOLVING TWO
02616          * OPERANDS, THE FIRST OPERAND IS AT LOCATIONS D03 TO
02617          * D07 IN THE FOLLOWING FORMAT:
02618          *   LOCATION D03 M.S.BYTE OF MANTISSA
02619          *   D04 MIDDLE BYTE OF MANTISSA
02620          *   D05 L.S.BYTE OF MANTISSA
02621          *   D06 EXPONENT
02622          *   D07 SIGN OF MANTISSA
02623          * THE ADDRESS OF THE SECOND OPERAND IS SPECIFIED BY
02624          * THE INDEX REGISTER AS FOLLOWS:
02625          *   0,X M.S.BYTE OF MANTISSA
02626          *   1,X MIDDLE BYTE OF MANTISSA
02627          *   2,X L.S.BYTE OF MANTISSA
02628          *   3,X EXPONENT
02629          *   4,X SIGN OF MANTISSA
02630          * THE RESULT ALWAYS OVERWRITES THE FIRST OPERAND IN
02631          * LOCATIONS D03 TO D07 ; THE SECOND OPERAND REMAINS
02632          * UNALTERED
02633          *
02634          * IN THIS PROGRAM D07 IS FLAG3
02635          *
02636          * ADDITION ROUTINE. ADD
02637          * -----
02638          * THE ADDEND, THE ADDRESS OF WHICH IS SPECIFIED BY
02639          * THE INDEX REGISTER, IS ADDED TO THE AUGEND IN
02640          * LOCATIONS D03 TO D07.
02641          *
02642 236F A6 04 ADD LDA A 4,X TEST SIGN OF ADDEND
02643 2371 2A 04 BPL ADDP IF SIGN IS NEGATIVE THEN
02644 2373 BD 23F0 JSR SUBP ...SUBTRACT ADDEND
02645 2376 39 RTS
02646 2377 DF 65 ADDP STX T1 SAVE INDEX REGISTER
02647 2379 BD 23FA JSR ADJEXP ADJUST EXPONENTS
02648 237C 96 A0 LDA A D03
02649 237E 9A A1 ORA A D04
02650 2380 9A A2 ORA A D05
02651 2382 26 07 BNE A1 TEST IF MANTISSA=0
02652 2384 7F 0097 CLR FLAG3 SET SIGN TO POSITIVE
02653 2387 BD 2456 JSR GET
02654 238A 39 RTS
02655 238B 96 97 A1 LDA A FLAG3 TEST FLAG
02656 238D 27 03 BEQ L1
02657 238F BD 2431 JSR COMPL COMPLEMENT MANTISSA
02658 2392 BD 2443 L1 JSR ADD1 ADD MANTISSAS
02659 2395 25 0C BCS L2 CHECK IF OVERFLOW OCCURRED
02660 2397 96 97 LDA A FLAG3 TEST SIGN
02661 2399 27 19 BEQ OUT1
02662 239B BD 2431 JSR COMPL COMPLEMENT MANTISSA
02663 239E BD 23BF JSR NORM NORMALISE RESULT
02664 23A1 20 11 BRA OUT1
02665 23A3 96 97 L2 LDA A FLAG3 TEST SIGN
02666 23A5 26 10 BNE L3
02667 23A7 0D SEC SHIFT IN A '1'
02668 23A8 76 00A0 ROR D03

```



```

02669 23AB 76 00A1      ROR      D04
02670 23AE 76 00A2      ROR      D05
02671 23B1 7C 00A3      INC      D06      ADJUST EXPONENT
02672 23B4 DE 65      OUT1    LDX      T1      RESTORE INDEX REGISTER
02673 23B6 39          RTS
02674 23B7 7F 0097 L3    CLR      FLAG3    SET SIGN TO POSITIVE
02675 23BA BD 23BF      JSR      NORM    NORMALISE RESULT
02676 23BD 20 F5      BRA      OUT1
02677
02678      *
      * NORMALISATION ROUTINE
02679 23BF 96 A0      NORM    LDA      A      D03      CHECK IF NORMALISED
02680 23C1 2B 15      BMI      R2
02681 23C3 86 19      LDA      A      #25      SET MAX.NUMBER OF SHIFTS
02682 23C5 4A      L5      DEC      A
02683 23C6 27 18      BEQ      L4
02684 23C8 7A 00A3      DEC      D06      DECREMENT EXPONENT
02685 23CB 29 0C      BVS      ZERO    CHECK FOR 2'S COMPL.OVERFLOW
02686 23CD 78 00A2      ASL      D05      SHIFT MANTISSA LEFT
02687 23D0 79 00A1      ROL      D04
02688 23D3 79 00A0      ROL      D03
02689 23D6 2A ED      BPL      L5      REPEAT IF NOT YET NORMALISED
02690 23D8 39      R2      RTS
02691 23D9 4F      ZERO    CLR      A
02692 23DA 97 A0      STA      A      D03      SET MANTISSA=0
02693 23DC 97 A1      STA      A      D04
02694 23DE 97 A2      STA      A      D05
02695 23E0 86 80      L4      LDA      A      #80      SET EXPONENT TO LARGEST
02696 23E2 97 A3      STA      A      D06      ...NEGATIVE NUMBER ALLOWED
02697 23E4 7F 0097      CLR      FLAG3
02698 23E7 39      RTS
02699
02700      *
      * SUBTRACTION ROUTINE. SUBTR
02701      * -----
02702      * THE SUBTRAHEND, THE ADDRESS OF WHICH IS SPECIFIED
02703      * BY THE INDEX REGISTER, IS SUBTRACTED FROM THE
02704      * MINUEND IN LOCATIONS D03 TO D07.
02705      *
02706 23E8 A6 04      SUBTR    LDA      A      4,X      TEST SIGN OF SUBTRAHEND
02707 23EA 2A 04      BPL      SUBP      IF SIGN IS NEGATIVE, THEN
02708 23EC BD 2377      JSR      ADDP      ...ADD SUBTRAHEND
02709 23EF 39      RTS
02710 23F0 73 0097 SUBP    COM      FLAG3    CHANGE SIGN
02711 23F3 BD 2377      JSR      ADDP      ADD ABS.VALUE OF SUBTRAHEND
02712 23F6 73 0097      COM      FLAG3    CHANGE SIGN OF RESULT
02713 23F9 39      RTS
02714
02715      *
      * SUBROUTINE 'ADJEXP' OPERATES ON THE SMALLER OF
02716      * THE TWO OPERANDS. ITS MANTISSA IS SHIFTED RIGHT
02717      * AND ITS EXPONENT INCREASED UNTIL IT IS EQUAL TO
02718      * THE EXPONENT OF THE LARGER OPERAND.
02719      *
02720 23FA A6 03      ADJEXP    LDA      A      3,X      EXPONENT OF ADDEND
02721 23FC 91 A3      CMP      A      D06      EXPONENT OF AUGEND
02722 23FE 27 12      BEQ      RET1      TEST FOR EQUAL EXPONENTS
02723 2400 2D 11      BLT      RELOC      (3,X)<(D06)
02724 2402 74 00A0 ADJ1    LSR      D03      SHIFT MANTISSA TO RIGHT AND
02725 2405 76 00A1      ROR      D04      ...INCREASE EXPONENT
02726 2408 76 00A2      ROR      D05

```



```

02727 240B 7C 00A3      INC      D06
02728 240E 91 A3        CMP A    D06
02729 2410 26 F0        BNE      ADJ1
02730 2412 39          RET1     RTS
02731                  *
02732                  * SUBROUTINE 'RELOC' IS REQUIRED IN CASE THE ADDEND
02733                  * IS A CONSTANT IN ROM. IF ITS EXPONENT IS SMALLER
02734                  * THAN THAT OF THE AUGEND, IT MUST BE TRANSFERRED TO
02735                  * RAM SO THAT ITS MANTISSA CAN BE SHIFTED RIGHT AND
02736                  * ITS EXPONENT INCREASED UNTIL IT EQUALS THE
02737                  * EXPONENT OF THE AUGEND.
02738                  *
02739 2413 E6 00          RELOC    LDA B    0,X      COPY MANTISSA INTO D00,D01,D0
02740 2415 D7 9D          STA B    D00
02741 2417 E6 01          LDA B    1,X
02742 2419 D7 9E          STA B    D01
02743 241B E6 02          LDA B    2,X
02744 241D D7 9F          STA B    D02
02745 241F 74 009D      ADJ2    LSR      D00      SHIFT MANTISSA TO RIGHT AND
02746 2422 76 009E          ROR      D01      ...INCREASE EXPONENT
02747 2425 76 009F          ROR      D02
02748 2428 4C          INC A      A CONTAINS EXP.OF ADDEND
02749 2429 91 A3        CMP A    D06      COMPARE WITH EXP.OF AUGEND
02750 242B 26 F2        BNE      ADJ2
02751 242D CE 009D      LDX      #D00      LOAD X REG.WITH ADDR.
02752 2430 39          RTS          ...OF RELOCATED ADDEND.
02753                  *
02754                  * SUBROUTINE 'COMPL' FORMS THE 2'S COMPLEMENT OF
02755                  * THE 3-BYTE WORD IN LOCATIONS D03,D04 AND D05
02756 2431 4F          COMPL    CLR A
02757 2432 90 A2          SUB A    D05
02758 2434 97 A2          STA A    D05
02759 2436 86 00          LDA A    #00
02760 2438 92 A1          SBC A    D04
02761 243A 97 A1          STA A    D04
02762 243C 86 00          LDA A    #00
02763 243E 92 A0          SBC A    D03
02764 2440 97 A0          STA A    D03
02765 2442 39          RTS
02766                  *
02767                  * SUBROUTINE 'ADD1' ADDS THE MANTISSA OF THE ADDEND
02768                  * AT THE ADDRESS SPECIFIED BY THE INDEX REGISTER TO
02769                  * THAT OF THE AUGEND IN LOCATIONS D03,D04,D05 AND
02770                  * STORES THE RESULT IN D03,D04,D05
02771 2443 96 A2          ADD1    LDA A    D05
02772 2445 AB 02          ADD A    2,X
02773 2447 97 A2          STA A    D05
02774 2449 96 A1          LDA A    D04
02775 244B A9 01          ADC A    1,X
02776 244D 97 A1          STA A    D04
02777 244F 96 A0          LDA A    D03
02778 2451 A9 00          ADC A    0,X
02779 2453 97 A0          STA A    D03
02780 2455 39          RTS
02781                  *
02782                  *
02783                  * SUBROUTINE 'GET' TRANSFERS 4 BYTES OF DATA FROM
02784                  * LOCATIONS SPECIFIED BY THE INDEX REGISTER, TO

```



```

02785          * LOCATIONS D03,D04,D05,D06.
02786 2456 A6 00 GET    LDA A  0,X
02787 2458 97 A0        STA A  D03
02788 245A A6 01        LDA A  1,X
02789 245C 97 A1        STA A  D04
02790 245E A6 02        LDA A  2,X
02791 2460 97 A2        STA A  D05
02792 2462 A6 03        LDA A  3,X
02793 2464 97 A3        STA A  D06
02794 2466 39          RTS
02795          *
02796 2467 96 A2 SUB1   LDA A  D05
02797 2469 A0 02        SUB A  2,X
02798 246B 97 A2        STA A  D05
02799 246D 96 A1        LDA A  D04
02800 246F A2 01        SBC A  1,X
02801 2471 97 A1        STA A  D04
02802 2473 96 A0        LDA A  D03
02803 2475 A2 00        SBC A  0,X
02804 2477 97 A0        STA A  D03
02805 2479 39          RTS
02806          *
02807 247A 96 9D MOVEE  LDA A  D00
02808 247C 97 A0        STA A  D03
02809 247E 96 9E        LDA A  D01
02810 2480 97 A1        STA A  D04
02811 2482 96 9F        LDA A  D02
02812 2484 97 A2        STA A  D05
02813 2486 39          RTS
02814          *
02815 2487 A6 04 MULT   LDA A  4,X      TEST SIGN
02816 2489 2A 03        BPL     MULTP
02817 248B 73 0097      COM     FLAG3   CHANGE SIGN
02818 248E BD 24F1 MULTP JSR     ZTST
02819 2491 5F          CLR B
02820 2492 D7 9D        STA B  D00
02821 2494 D7 9E        STA B  D01
02822 2496 D7 9F        STA B  D02
02823 2498 C6 18        LDA B  #24     SHIFT COUNT
02824 249A 74 00A0      LSR     D03
02825 249D 76 00A1      ROR     D04
02826 24A0 76 00A2      ROR     D05
02827 24A3 24 12 M1     BCC     SHFT    C=1?
02828 24A5 A6 02        LDA A  2,X     YES
02829 24A7 9B 9F        ADD A  D02
02830 24A9 97 9F        STA A  D02
02831 24AB A6 01        LDA A  1,X
02832 24AD 99 9E        ADC A  D01
02833 24AF 97 9E        STA A  D01
02834 24B1 A6 00        LDA A  0,X
02835 24B3 99 9D        ADC A  D00
02836 24B5 97 9D        STA A  D00
02837 24B7 76 009D SHFT ROR     D00    NO
02838 24BA 76 009E      ROR     D01
02839 24BD 76 009F      ROR     D02
02840 24C0 76 00A0      ROR     D03
02841 24C3 76 00A1      ROR     D04
02842 24C6 76 00A2      ROR     D05

```


02843	24C9	5A		DEC B		CONT UNTIL ZERO
02844	24CA	26	D7	BNE	M1	
02845	24CC	A6	03	LDA A	3,X	
02846	24CE	9B	A3	ADD A	D06	
02847	24D0	29	26	BVS	Z1	2'S COMP OVERFLOW?
02848	24D2	97	A3	STA A	D06	
02849	24D4	96	9D	LDA A	D00	NORM?
02850	24D6	2B	11	BMI	M2	
02851	24D8	79	00A0	ROL	D03	
02852	24DB	79	009F	ROL	D02	
02853	24DE	79	009E	ROL	D01	
02854	24E1	79	009D	ROL	D00	
02855	24E4	7A	00A3	DEC	D06	
02856	24E7	29	0F	BVS	Z1	2'S COMP OVERFLOW?
02857	24E9	BD	247A M2	JSR	MOVEE	
02858	24EC	96	A0	LDA A	D03	
02859	24EE	27	08	BEQ	Z1	
02860	24F0	39		RTS		
02861			*			
02862	24F1	96	A0	ZTST	LDA A D03	
02863	24F3	26	02	BNE	R1	
02864	24F5	31		INS		
02865	24F6	31		INS		
02866	24F7	39	R1	RTS		
02867			*			
02868	24F8	7E	23D9	Z1	JMP ZERO	
02869			*			
02870	24FB	A6	04	DIV	LDA A 4,X	TEST SIGN OF DIVIDEND
02871	24FD	2A	03	BPL	DIVF	
02872	24FF	73	0097	COM	FLAG3	CH. SIGN
02873	2502	BD	24F1	DIVF	JSR ZTST	DIVIDEND=0?
02874	2505	5F		CLR B		
02875	2506	D7	9D	STA B	D00	
02876	2508	D7	9E	STA B	D01	
02877	250A	D7	9F	STA B	D02	
02878	250C	C6	19	LDA B	#25	SHIFT COUNT
02879	250E	BD	2467	COMP	JSR SUB1	DIVI > DIVISOR?
02880	2511	24	06	BCC	SHFT1	YES
02881	2513	BD	2443	JSR	ADD1	NO
02882	2516	0C		CLC		
02883	2517	20	01	BRA	SHFT0	
02884	2519	0D	SHFT1	SEC		
02885	251A	79	009F	SHFT0	ROL D02	
02886	251D	79	009E		ROL D01	
02887	2520	79	009D		ROL D00	
02888	2523	5A		DEC B		SHIFT COUNT=0?
02889	2524	27	10	BEQ	TC	YES
02890	2526	78	00A2	ASL	D05	
02891	2529	79	00A1	ROL	D04	
02892	252C	79	00A0	ROL	D03	
02893	252F	24	DD	BCC	COMP	C SET?
02894	2531	BD	2467	JSR	SUB1	YES
02895	2534	20	E3	BRA	SHFT1	
02896	2536	24	0C	TC	BCC SBEXP	
02897	2538	76	009D	ROR	D00	
02898	253B	76	009E	ROR	D01	
02899	253E	76	009F	ROR	D02	
02900	2541	7C	00A3	INC	D06	


```

02901 2544 96 A3 SBEXP LDA A D06
02902 2546 A0 03 SUB A 3,X SUBT. EXPONENTS
02903 2548 29 AE BVS Z1
02904 254A 97 A3 STA A D06
02905 254C BD 247A JSR MOVEE
02906 254F 39 RTS
02907 *
02908 * IRQ
02909 * ---
02910 * KEYBOARD INPUT FUNCTION SELECT
02911 2550 BD 11E8 IRQ JSR DELAY
02912 2553 B6 A000 LDA A PIAD1 FROM KEYBOARD
02913 2556 B4 0F AND A #$0F MASK 4MSB
02914 2558 97 00 STA A TEMP
02915 255A 7D 009B TST AFLAG TEST FOR ABORT
02916 255D 27 03 BEQ DB1
02917 255F 7E 27C8 JMP AC ABORT MOVEMENT
02918 2562 BD 19AE DB1 JSR BUT KEY RELEASE CHECK
02919 2565 96 00 INPUT LDA A TEMP CHECK ENTERED CHAR.
02920 2567 81 0C CMP A #$0C INPUT SELECT
02921 2569 26 03 BNE JP
02922 256B 7E 2594 JMP IS
02923 256E 81 0D JP CMP A #$0D X/Y SELECT
02924 2570 26 03 BNE JP1
02925 2572 7E 2698 JMP XYS
02926 2575 81 0A JP1 CMP A #$0A INCREMENT
02927 2577 26 03 BNE JP2
02928 2579 7E 26BA JMP INC
02929 257C 81 0B JP2 CMP A #$0B DECREMENT
02930 257E 26 03 BNE JP3
02931 2580 7E 26E1 JMP DEC
02932 2583 81 0E JP3 CMP A #$0E ENTER/GOTO
02933 2585 26 03 BNE JP4
02934 2587 7E 2705 JMP EG
02935 258A 81 0F JP4 CMP A #$0F ABORT/CLEAR
02936 258C 26 03 BNE JP5
02937 258E 7E 27C8 JMP AC
02938 2591 7E 2808 JP5 JMP INFO
02939 *
02940 * SELECT GOTO1,HOME,SPEED or SCALE
02941 2594 7C 0094 IS INC FLAG
02942 2597 96 94 SEL LDA A FLAG
02943 2599 7F 0095 CLR FLAG1 OVERFLOW/CLEAR/ABORT FLAG
02944 259C 7F 0096 CLR FLAG2
02945 259F 81 01 CMP A #$01
02946 25A1 27 32 BEQ HOME
02947 25A3 81 02 CMP A #$02
02948 25A5 27 62 BEQ SCALE
02949 25A7 81 03 CMP A #$03
02950 25A9 27 07 BEQ SPD
02951 25AB 81 04 CMP A #$04
02952 25AD 26 11 BNE POS
02953 25AF 7E 263D JMP GOTO1
02954 25B2 7D 00A9 SPD TST XSPMK2 MK1 or MK2 CONTROLLER?
02955 25B5 27 03 BEQ NOSPD
02956 25B7 7E 2681 JMP SPEED
02957 25BA 7C 0094 NOSPD INC FLAG
02958 25BD 7E 263D JMP GOTO1

```



```

02959          *
02960          *POSITION MODE
02961          *
02962 25C0 7F 0094 POS      CLR      FLAG
02963 25C3 CE 0009          LDX      #YCPOS      Y COARSE POSITION
02964 25C6 DF 75          STX      REG      DISPLAY REGISTER
02965 25C8 CE 0001          LDX      #XCPOS
02966 25CB DF 77          STX      REG1
02967 25CD 86 C1          LDA A      #C1
02968 25CF B7 A004          STA A      PIAD3      LIGHT X/POSITION LED'S
02969 25D2 7E 2844          JMP      RTI
02970          *
02971          *HOME MODE
02972          *
02973 25D5 96 9A      HOME      LDA A      FLAGXY      CHECK X/Y SELECT
02974 25D7 81 00          CMP A      #$00
02975 25D9 27 10          BEQ      H0
02976 25DB 81 AA          CMP A      #$AA
02977 25DD 27 18          BEQ      H01
02978 25DF 86 82          LDA A      #$82      Y ONLY
02979 25E1 B7 A004          STA A      PIAD3      LIGHT LED
02980 25E4 CE 004B          LDX      #YCHOME
02981 25E7 DF 79          STX      REG2      CHANGE REG
02982 25E9 20 11          BRA      H02
02983 25EB 86 42      H0      LDA A      #$42      X ONLY
02984 25ED B7 A004          STA A      PIAD3      LIGHT LED
02985 25F0 CE 0043          LDX      #XCHOME
02986 25F3 DF 79          STX      REG2
02987 25F5 20 05          BRA      H02
02988 25F7 86 C2      H01      LDA A      #C2
02989 25F9 B7 A004          STA A      PIAD3
02990 25FC CE 004B      H02      LDX      #YCHOME
02991 25FF DF 75          STX      REG
02992 2601 CE 0043          LDX      #XCHOME
02993 2604 DF 77          STX      REG1
02994 2606 7E 2844          JMP      RTI
02995          *
02996          *SCALE MODE
02997          *
02998 2609 96 9A      SCALE      LDA A      FLAGXY      CHECK X/Y SELECT
02999 260B 81 00          CMP A      #$00
03000 260D 27 10          BEQ      SC
03001 260F 81 AA          CMP A      #$AA
03002 2611 27 18          BEQ      SC1
03003 2613 86 84          LDA A      #$84
03004 2615 B7 A004          STA A      PIAD3      LIGHT LED
03005 2618 CE 003B          LDX      #YCSCCL
03006 261B DF 79          STX      REG2
03007 261D 20 11          BRA      SC2
03008 261F 86 44      SC      LDA A      #$44      X ONLY
03009 2621 B7 A004          STA A      PIAD3      LIGHT LED
03010 2624 CE 0033          LDX      #XCSCCL
03011 2627 DF 79          STX      REG2      CHANGE REG
03012 2629 20 05          BRA      SC2
03013 262B 86 C4      SC1      LDA A      #C4
03014 262D B7 A004          STA A      PIAD3
03015 2630 CE 003B      SC2      LDX      #YCSCCL
03016 2633 DF 75          STX      REG

```



```

03017 2635 CE 0033      LDX      #XCSCl
03018 2638 DF 77        STX      REG1
03019 263A 7E 2844      JMP      RTI
03020                      *
03021                      *MOVE TO MODE
03022                      *
03023 263D 96 9A      GOT01  LDA  A   FLAGXY      X OR Y
03024 263F 81 00      CMP  A   #$00
03025 2641 27 18      BEQ      GT
03026 2643 81 AA      CMP  A   #$AA
03027 2645 27 28      BEQ      GT1
03028 2647 86 88      LDA  A   #$88      Y ONLY
03029 2649 B7 A004      STA  A   PIAD3      LIGHT LED'S
03030 264C CE 002B      LDX      #YCGOTO
03031 264F DF 75      STX      REG      DISPLAY REG
03032 2651 DF 79      STX      REG2     CHANGE REG
03033 2653 CE 0009      LDX      #YCPOS
03034 2656 DF 77      STX      REG1     DISPLAY YPOS ON X DISPLAY
03035 2658 7E 2844      JMP      RTI
03036 265B 86 48      GT      LDA  A   #$48      X ONLY
03037 265D B7 A004      STA  A   PIAD3      LEDS
03038 2660 CE 0023      LDX      #XCGOTO
03039 2663 DF 77      STX      REG1
03040 2665 DF 79      STX      REG2     CHANGE REG
03041 2667 CE 0001      LDX      #XCPOS
03042 266A DF 75      STX      REG      DISPLAY XPOS ON Y DISPLAY
03043 266C 7E 2844      JMP      RTI
03044 266F CE 002B      GT1    LDX      #YCGOTO
03045 2672 DF 75      STX      REG
03046 2674 CE 0023      LDX      #XCGOTO
03047 2677 DF 77      STX      REG1
03048 2679 86 C8      LDA  A   #$C8
03049 267B B7 A004      STA  A   PIAD3
03050 267E 7E 2844      JMP      RTI
03051                      *
03052                      *ENTER SPEED MODE
03053                      *
03054 2681 CE 005B      SPEED  LDX      #YCSPD
03055 2684 DF 75      STX      REG      DISPLAY REG(MODE INDECATOR)
03056 2686 86 50      LDA  A   #$50      X ONLY
03057 2688 B7 A004      STA  A   PIAD3      LIGHT LEDS
03058 268B CE 0053      LDX      #XCSPD
03059 268E DF 77      STX      REG1     DISPLAY REG(MODE INDECATOR)
03060 2690 CE 0058      LDX      #XFSPD+1
03061 2693 DF 79      STX      REG2     CHANGE REG
03062 2695 7E 2844      JMP      RTI
03063                      *
03064                      * SELECT X or Y
03065                      *
03066 2698 7D 00A9      XYS     TST      XSPMK2      MK2 SP.CONT. ?
03067 269B 26 06      BNE      XYS3      YES  X ONLY
03068 269D 96 9A      LDA  A   FLAGXY      00=X:FF=Y:AA=XY
03069 269F 81 AA      CMP  A   #$AA
03070 26A1 26 06      BNE      XYS1
03071 26A3 7F 009A      XYS3    CLR      FLAGXY
03072 26A6 7E 2597      JMP      SEL
03073 26A9 81 00      XYS1    CMP  A   #$00
03074 26AB 26 06      BNE      XYS2

```



```

03075 26AD 73 009A      COM      FLAGXY
03076 26B0 7E 2597      JMP      SEL
03077 26B3 86 AA      XYS2  LDA A   #$AA
03078 26B5 97 9A      STA A   FLAGXY
03079 26B7 7E 2597      JMP      SEL
03080                      *
03081                      * INCREMENT X or Y MOVE TO POSITION
03082                      * -----
03083 26BA 96 94      INC      LDA A   FLAG      INC GOTO ONLY
03084 26BC 81 04      CMP A   #$04
03085 26BE 26 1B      BNE      RTI1
03086 26C0 96 9A      LDA A   FLAGXY      CHECK X or Y
03087 26C2 81 00      CMP A   #$00
03088 26C4 27 0F      BEQ      XX
03089 26C6 81 AA      CMP A   #$AA
03090 26C8 27 14      BEQ      XX1
03091 26CA CE 003B      LDX      #YCSCl
03092 26CD CE 002B      LDX      #YCGOTO
03093 26D0 BD 18E8      JSR      ADDSCLE      ADD SCALE
03094 26D3 20 06      BRA      RTI1
03095 26D5 CE 0023 XX      LDX      #XCGOTO
03096 26D8 BD 18E8      JSR      ADDSCLE
03097 26DB 7E 2844 RTI1      JMP      RTI
03098 26DE 7E 1BC2 XX1      JMP      ER3      X OR Y
03099                      *
03100                      * DECREMENT X or Y MOVE TO POSITION
03101                      * -----
03102 26E1 96 94      DEC      LDA A   FLAG      GOTO ONLY
03103 26E3 81 04      CMP A   #$04
03104 26E5 26 1B      BNE      RTI2
03105 26E7 96 9A      LDA A   FLAGXY      CHECK Xor Y
03106 26E9 81 00      CMP A   #$00
03107 26EB 27 0F      BEQ      XXX
03108 26ED 81 AA      CMP A   #$AA
03109 26EF 27 ED      BEQ      XX1
03110 26F1 CE 003B      LDX      #YCSCl
03111 26F4 CE 002B      LDX      #YCGOTO
03112 26F7 BD 192C      JSR      SUBSCL      SUBTRACT SCALE
03113 26FA 20 06      BRA      RTI2
03114 26FC CE 0023 XXX      LDX      #XCGOTO
03115 26FF BD 192C      JSR      SUBSCL
03116 2702 7E 2844 RTI2      JMP      RTI
03117                      *
03118                      * ENTER/GOTO
03119                      * -----
03120                      *
03121 2705 7F 006D EG      CLR      T8
03122 2708 7F 006E      CLR      T9
03123 270B 96 94      LDA A   FLAG
03124 270D 81 01      CMP A   #$01
03125 270F 27 6A      BEQ      MHOME      MOVE TO 'HOME' POSITION
03126 2711 81 04      GOTO      CMP A   #$04
03127 2713 26 63      BNE      RT
03128 2715 7D 00A9      TST      XSPMK2      MK1 or MK2 SP.CONTROLLER
03129 2718 27 05      BEQ      GOTO11
03130 271A BD 19BC      JSR      SPCON      CALCULATE MK2 SPEED
03131 271D 20 21      BRA      GOTO12
03132 271F CE 3012 GOTO11 LDX      #SPEEDI      SET SPEED INC/DEC REGISTERS

```


03133	2722	DF	7D		STX	REG4	
03134	2724	DF	7F		STX	REG5	
03135	2726	7F	006F		CLR	XSTEP	
03136	2729	7F	0070		CLR	YSTEP	
03137	272C	7F	0071		CLR	XSTEP C	
03138	272F	7F	0072		CLR	YSTEP C	
03139	2732	86	C7		LDA A	#\$C7	
03140	2734	B7	A09C		STA A	PT2MSB	SET LOWER INC/DEC RATE
03141	2737	B7	A09D		STA A	PT2LSB	
03142	273A	96	9A		LDA A	FLAGXY	
03143	273C	81	FF		CMP A	#\$FF	
03144	273E	27	17		BEQ	YGT	
03145	2740	CE	0023	GOTO12	LDX	#XCGOTO	
03146	2743	BD	1203		JSR	CHEX	CHANGE TO HEX
03147	2746	BD	183A		JSR	MCON	SET SUBRAHEND
03148	2749	BD	17D0		JSR	MINX	SET MINUEND(X)
03149	274C	BD	157D		JSR	ESX	TEST X-AXIS ENDSTOPS
03150	274F	BD	15E7		JSR	XSPEED	X SPEED SELECT
03151	2752	7D	009A		TST	FLAGXY	X AND/OR Y FLAG
03152	2755	27	12		BEQ	BGO	
03153	2757	CE	002B	YGT	LDX	#YCGOTO	
03154	275A	BD	1203		JSR	CHEX	CHANGE TO HEX
03155	275D	BD	1855		JSR	MCO	SET SUBRAHEND
03156	2760	BD	17F4		JSR	MINY	SET Y MINUEND
03157	2763	BD	15B2		JSR	ESY	TEST Y-AXIS ENDSTOPS
03158	2766	BD	1689		JSR	YSPEED	Y SPEED SELECT
03159	2769	B6	A0A0	BGO	LDA A	MODEMC	CLR RDRF FOR EX ABORT
03160	276C	BD	16D3		JSR	MOVE	
03161	276F	96	9A		LDA A	FLAGXY	X AND/OR Y FLAG
03162	2771	81	AA		CMP A	#\$AA	IF X/Y RETURN TO POSITION MOD
03163	2773	26	03		BNE	RT	
03164	2775	7F	0094		CLR	FLAG	
03165	2778	7E	2597	RT	JMP	SEL	SELECT MODE AND DISPLAY MODE
03166	277B	96	04	MHOME	LDA A	XCPOS+3	
03167	277D	81	AA		CMP A	#\$AA	NO FIXING YET
03168	277F	27	11		BEQ	MH2	
03169	2781	7D	00A9		TST	XSPMK2	SKIP YTESTS IF MK2
03170	2784	26	0F		BNE	MH1	
03171	2786	96	0C		LDA A	YCPOS+3	
03172	2788	81	AA		CMP A	#\$AA	
03173	278A	27	06		BEQ	MH2	
03174	278C	96	9A		LDA A	FLAGXY	
03175	278E	81	AA		CMP A	#\$AA	ONLY MOVE IF X AND Y LED'S AR
03176	2790	27	03		BEQ	MH1	
03177	2792	7E	1BC2	MH2	JMP	ER3	
03178	2795	CE	0043	MH1	LDX	#XCHOME	
03179	2798	BD	1203		JSR	CHEX	CHANGE TO HEX
03180	279B	BD	183A		JSR	MCON	SUBRAHEND
03181	279E	BD	17D0		JSR	MINX	X MINUEND
03182	27A1	BD	15E7		JSR	XSPEED	X SPEED SELECT
03183	27A4	7D	00A9		TST	XSPMK2	IF MK2 MOVE X ONLY
03184	27A7	26	0F		BNE	MH3	
03185	27A9	CE	004B		LDX	#YCHOME	
03186	27AC	BD	1203		JSR	CHEX	CHANGE TO HEX
03187	27AF	BD	1855		JSR	MCO	SUBRAHEND
03188	27B2	BD	17F4		JSR	MINY	Y MINUEND
03189	27B5	BD	1689		JSR	YSPEED	Y SPEED SELECT
03190	27B8	BD	16D3	MH3	JSR	MOVE	


```

03191 27BB CE 0009      LDX      WYCPOS      RESET DISPLAY REGISTERS
03192 27BE DF 75        STX      REG
03193 27C0 CE 0001      LDX      #XCPOS
03194 27C3 DF 77        STX      REG1
03195 27C5 7E 2844      JMP      RTI
03196
03197      *
03198      * ABORT and CLEAR routine
03199      * -----
03199 27C8 DE 79      AC      LDX      REG2
03200 27CA 96 9B      LDA      A      AFLAG      ABORT MOVEMENT FLAG
03201 27CC 81 FF      CMP      A      #$FF
03202 27CE 27 24      BEQ      ABORT
03203 27D0 7D 006D      TST      T8      CLEAR ERROR DIGIT
03204 27D3 26 29      BNE      AC3
03205 27D5 96 95      LDA      A      FLAG1
03206 27D7 81 00      CMP      A      #$00
03207 27D9 27 16      BEQ      AC2
03208 27DB 81 03      CMP      A      #$03
03209 27DD 26 08      BNE      AC1
03210 27DF 96 96      LDA      A      FLAG2
03211 27E1 81 02      CMP      A      #$02
03212 27E3 27 02      BEQ      AC1
03213 27E5 09      DEX
03214 27E6 09      DEX
03215 27E7 09      AC1      DEX
03216 27E8 86 00      LDA      A      #$00
03217 27EA A7 00      STA      A      0,X
03218 27EC 7A 0095      DEC      FLAG1
03219 27EF DF 79      STX      REG2
03220 27F1 7E 2844      AC2      JMP      RTI
03221 27F4 4F      ABORT      CLR      A
03222 27F5 BD 1263      JSR      OM02
03223 27F8 7F 009B      CLR      AFLAG
03224 27FB 7E 2597      JMP      SEL
03225 27FE 86 3E      AC3      LDA      A      #$3E
03226 2800 B7 A003      STA      A      PIAC2
03227 2803 7F 006D      CLR      T8
03228 2806 20 E9      BRA      AC2
03229
03230      *
03231      * ENTER INFO(NUMERICALS)
03232      * -----
03232 2808 96 94      INFO      LDA      A      FLAG      DISREGARD POS. MODE
03233 280A 27 37      BEQ      F01
03234 280C 81 03      CMP      A      #$03      SPEED
03235 280E 26 0A      BNE      IF
03236 2810 DE 79      LDX      REG2      LOAD LOWER TWO DIGETS ONLY
03237 2812 D6 95      LDA      B      FLAG1
03238 2814 C1 02      CMP      B      #$02
03239 2816 2C 2B      BGE      F01
03240 2818 20 1D      BRA      F0
03241 281A 96 9A      IF      LDA      A      FLAGXY
03242 281C 81 AA      CMP      A      #$AA
03243 281E 26 03      BNE      INF
03244 2820 7E 1BC2      JMP      ER3
03245 2823 DE 79      INF      LDX      REG2
03246 2825 D6 95      LDA      B      FLAG1
03247 2827 C1 05      CMP      B      #$05
03248 2829 2C 18      BGE      F01

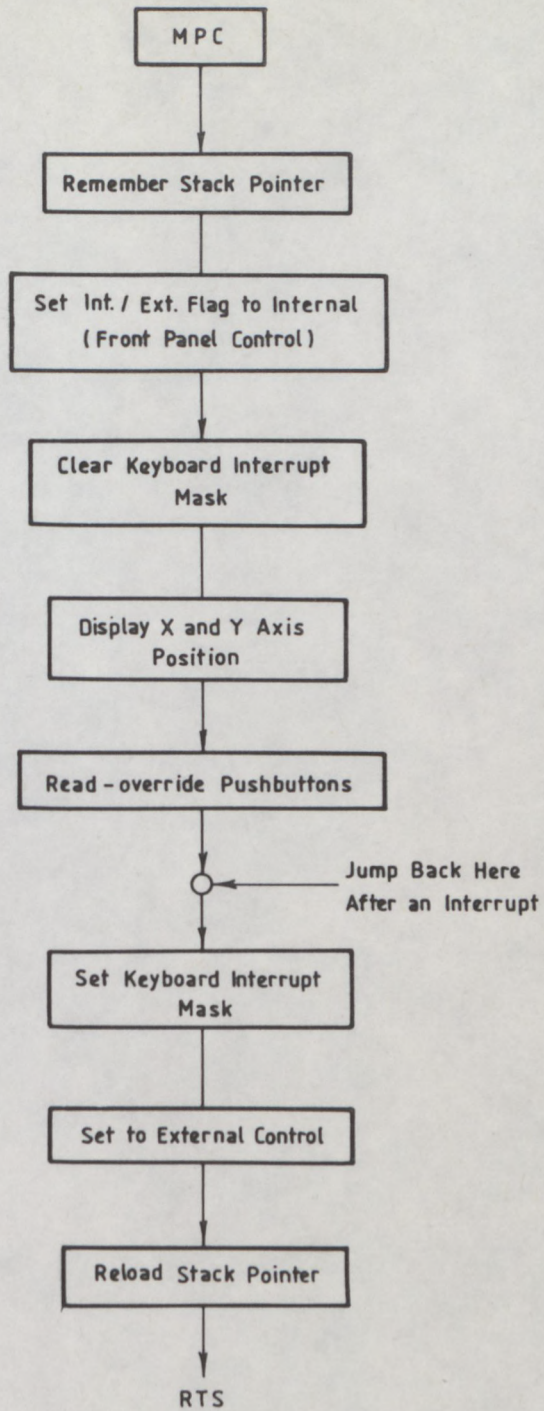
```

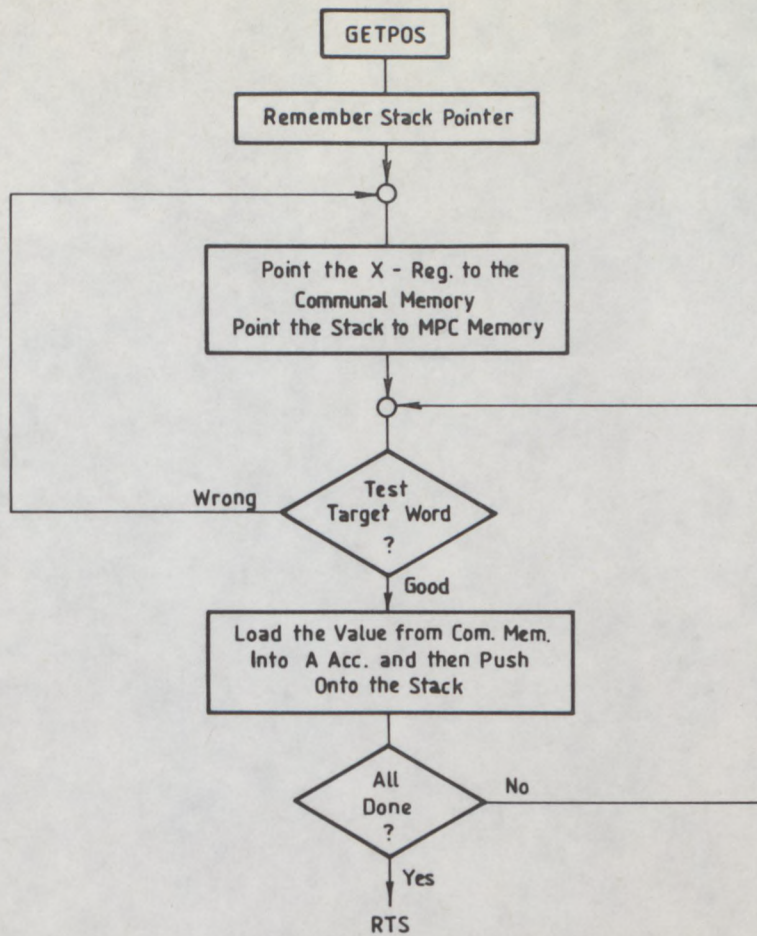
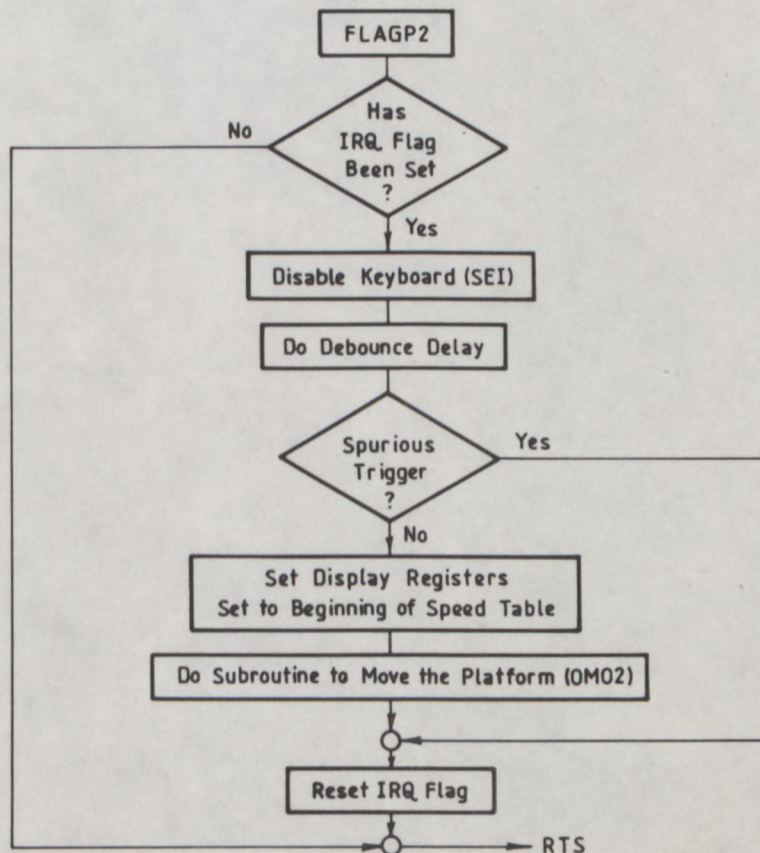

03249	282B	C1	03		CMP B	#03	
03250	282D	26	08		BNE	F0	
03251	282F	96	96		LDA A	FLAG2	
03252	2831	81	02		CMP A	#02	
03253	2833	26	02		BNE	F0	
03254	2835	08			INX		
03255	2836	08			INX		
03256	2837	96	00	F0	LDA A	TEMP	
03257	2839	A7	00		STA A	0,X	
03258	283B	D7	96		STA B	FLAG2	
03259	283D	7C	0095		INC	FLAG1	
03260	2840	08			INX		
03261	2841	DF	79		STX	REG2	
03262	2843	01		F01	NOP		
03263				*			
03264	2844	B6	A000	RTI	LDA A	PIAD1	RESET KEYBD. FLAGS
03265	2847	B6	A002		LDA A	PIAD2	
03266	284A	7D	0093		TST	FLA	INT OR EXT
03267	284D	27	03		BEQ	RTS	
03268	284F	7E	111D		JMP	MPC1	
03269	2852	B6	C0A5	RTS	LDA A	COMBUF+4	EXT PROG CONTROLLED
03270	2855	27	06		BEQ	RTS1	NO
03271	2857	BD	C726		JSR	CHKOFF	SEND 8 EOT'S
03272	285A	BD	E6F6		JSR	COMCLR	CLEAR COMMAND BUFFER
03273	285D	39		RTS1	RTS		
03274				*			
03275					OPT	L	
03276				*			
03277				*PROMPTS			
03278				*			
03279	2A00				ORG	\$2A00	
03280	2A00	4D		MP1	FCC	/MOVE MEAST PLATFORM	M;OPTIONS:(RUN)
03281	2A24	0A			FCB	\$A,\$D	
03282				*			
03283	2A26	0A		MP2	FCB	\$A,\$D	
03284	2A28	4D			FCC	/MEASUREMENT PLATFORM OPTIONS ; F=FOR	
03285	2A62	0A			FCB	\$A,\$D	
03286	2A64	52			FCC	/R=RIGHT,G=MOVE TO POSITION,IX=IY=INC	
03287	2A9D	0A			FCB	\$A,\$D	
03288	2A9F	44			FCC	/DX=DY=DECREMENT GRID POSITION,F=DISP	
03289	2AD9	0A			FCB	\$A,\$D	
03290	2ADB	47			FCC	/GH=MOVE 'HOME',GT=MOVE TO TARGET POS	
03291	2B04	0A			FCB	\$A,\$D,\$4	
03292				*			
03293	2B07	4D		MT1	FCC	/MOVE TROLLEY	M;OPTIONS:(RUN)
03294	2B2B	0A			FCB	\$A,\$D,\$A,\$D	
03295	2B2F	54			FCC	/TROLLEY CONTROLLER OPTIONS ; F=FORWA	
03296	2B61	0A			FCB	\$A,\$D	
03297	2B63	49			FCC	/IX=INCREMENT GRID POSITION,ID=DECREM	
03298	2B98	0A			FCB	\$A,\$D	
03299	2B9A	47			FCC	/G=MOVE TO ...,GH=MOVE 'HOME',GT=MOVE	
03300	2BCF	0A			FCB	\$A,\$D	
03301	2BD1	50			FCC	/P=DISPLAY\CHANGE TROLLEY PARAMETERS/	
03302	2BF4	0A			FCB	\$A,\$D,4	
03303				*			
03304	2BF7	0A		MP3	FCB	\$A,\$D	
03305	2BF9	45			FCC	/ENTER F(FORWARD) OR B(REVERSE) FOR 1	
03306	2C33	0A			FCB	\$A,\$D,\$4	

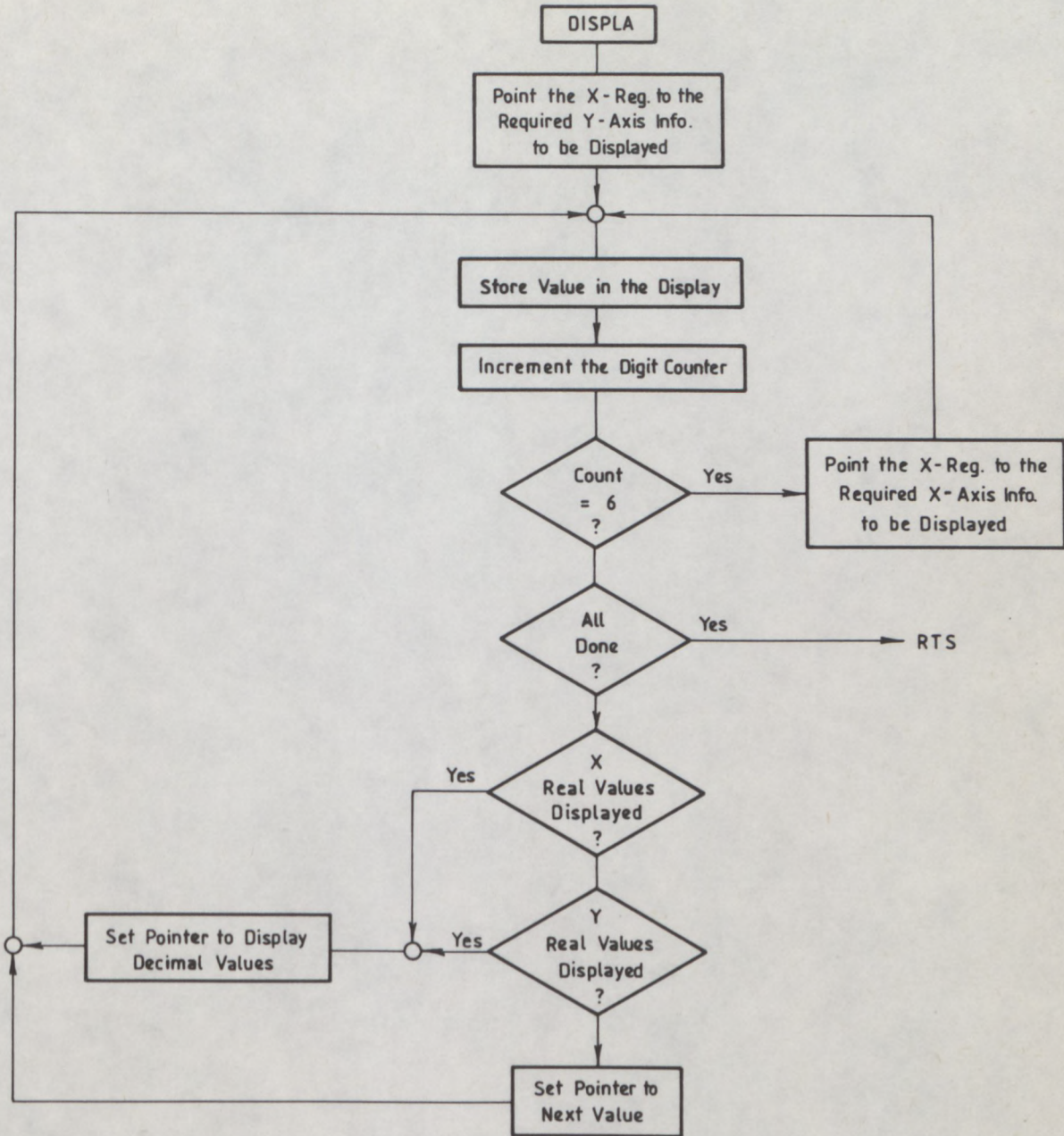
03307		*		
03308	2C36 0A	MP4	FCC	\$A,\$D
03309	2C38 45		FCC	/ENTER L(LEFT) OR R(RIGHT) FOR 1st Y
03310	2C6D 0A		FCB	\$A,\$D,\$4
03311		*		
03312	2C70 50	MP5	FCC	/PLATFORM X POSITION IS = /
03313	2C8C 04		FCB	\$4
03314		*		
03315	2C8D 50	MP6	FCC	/PLATFORM Y POSITION IS = /
03316	2CA9 04		FCB	\$4
03317		*		
03318	2CAA 49	MP7	FCC	/IS THIS POSITION CORRECT?.....Y OR N
03319	2CCE 0A		FCB	\$A,\$D,\$4
03320		*		
03321	2CD1 53	MP8	FCC	/STRIKE THE SPACE BAR TO STOP/
03322	2CED 0A		FCB	\$A,\$D,\$4
03323		*		
03324	2CF0 45	MP10	FCC	/ENTER..Q FOR NO X AXIS MOVEMENT/
03325	2D0F 0A		FCB	\$A,\$D
03326	2D11 4F		FCC	/OR.....X PLUS NEW DESTINATION/
03327	2D2E 0A		FCB	\$A,\$D,\$4
03328		*		
03329	2D31 58	MP11	FCC	/X AXIS 'HOME' POSITION IS = /
03330	2D4D 04		FCB	\$4
03331		*		
03332	2D4E 50	MP12	FCC	/PLATFORM X AXIS SPEED IS = /
03333	2D6A 04		FCB	\$4
03334		*		
03335	2D6B 20	MP13	FCC	/ CM PER SEC./
03336	2D77 0A		FCB	\$A,\$D,\$4
03337		*		
03338	2D7A 58	MP14	FCC	/X AXIS GRID SPACING IS = /
03339	2D96 04		FCB	\$4
03340		*		
03341	2D97 50	MP15	FCC	/PLATFORM X AXIS LIMITS ARE /
03342	2DB3 04		FCB	\$4
03343		*		
03344	2DB4 20	MP15A	FCC	/ AND /
03345	2DD0 04		FCB	\$4
03346		*		
03347	2DD1 44	MP16	FCC	/DO YOU WISH TO CHANGE THIS?...Y OR N
03348	2DF5 0A		FCB	\$A,\$D,\$4
03349		*		
03350	2DF8 4D	MP18	FCC	/MY X AXIS DESTINATION IS = /
03351	2E14 04		FCB	\$4
03352		*		
03353	2E15 45	MP19	FCC	/ENTER : G...MOVE TO DESTIATION/
03354	2E34 0A		FCB	\$A,\$D
03355	2E36 20		FCC	/ D...DECREMENT DESTINATION/
03356	2E58 0A		FCB	\$A,\$D
03357	2E5A 20		FCC	/ I...INCREMENT DESTINATION/
03358	2E7C 0A		FCB	\$A,\$D
03359	2E7E 20		FCC	/ Q...QUIT/
03360	2E8F 0A		FCB	\$A,\$D,\$4
03361		*		
03362	2E92 44	MP20	FCC	/DESTINATION EXCEEDS LIMITS/
03363	2EAC 0A		FCB	\$A,\$D,\$4
03364		*		

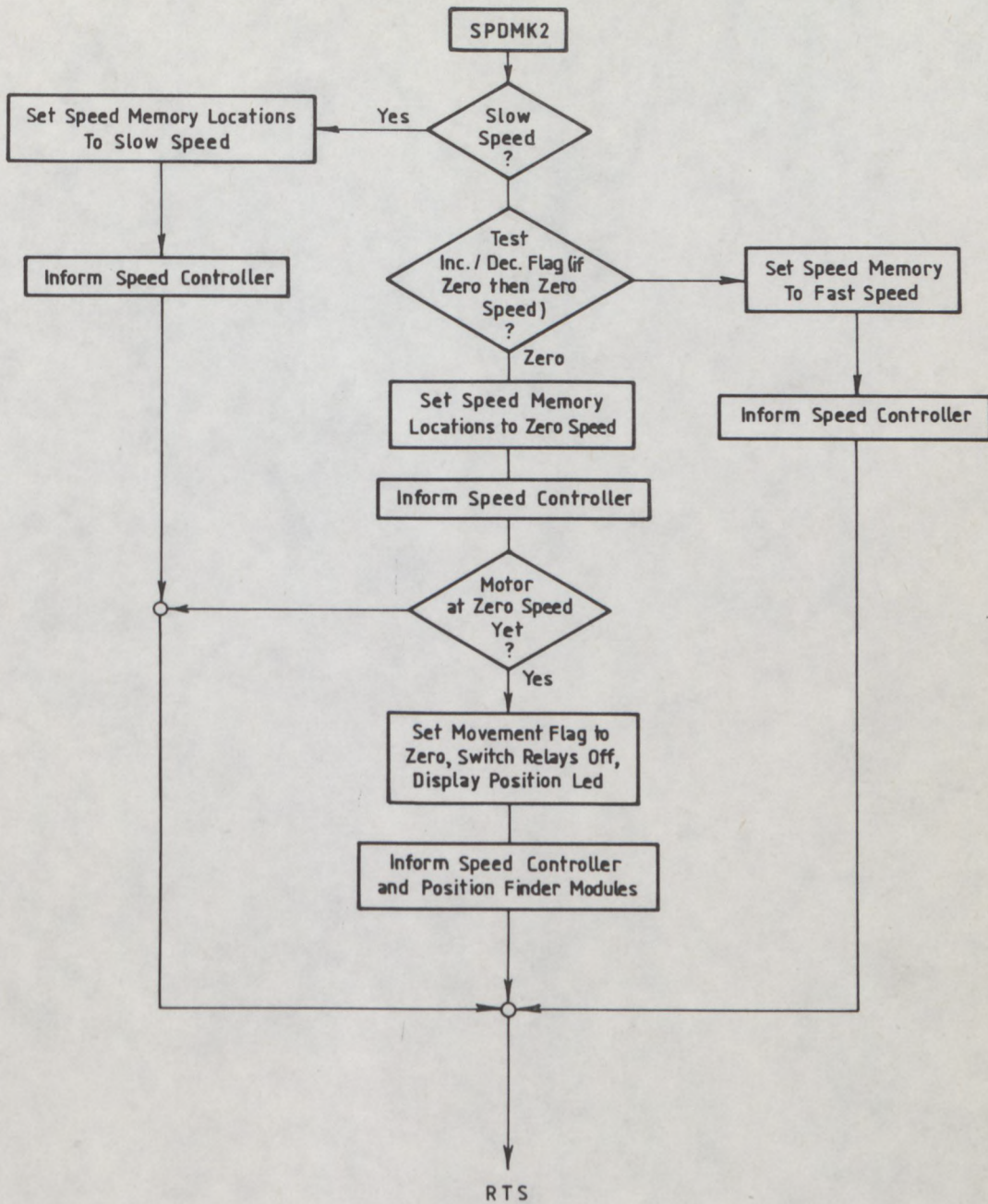
03365	2EAF	0A	MP21	FCB	\$A,\$D	
03366	2EB1	54		FCC	/TO ABORT ENTER	ANY KEY:(RUN)/
03367	2ED0	0A		FCB	\$A,\$D,\$4	
03368			*			
03369	2ED3	59	MP22	FCC	/Y AXIS 'HOME' POSITION IS =	/
03370	2EEF	04		FCB	\$4	
03371			*			
03372	2EF0	59	MP23	FCC	/Y AXIS GRID SPACING IS =	/
03373	2F0C	04		FCB	\$4	
03374			*			
03375	2F0D	50	MP24	FCC	/PLATFORM Y AXIS LIMITS ARE	/
03376	2F29	04		FCB	\$4	
03377			*			
03378	2F2A	20	MP24A	FCC	/	AND /
03379	2F46	04		FCB	\$4	
03380			*			
03381	2F47	4D	MP25	FCC	/MY Y AXIS DESTINATION IS =	/
03382	2F63	04		FCB	\$4	
03383			*			
03384	2F64	45	MP26	FCC	/ENTER..Q FOR NO Y AXIS MOVEMENT/	
03385	2F83	0A		FCB	\$A,\$D	
03386	2F85	4F		FCC	/OR.....Y PLUS NEW DESTIATION/	
03387	2FA1	0A		FCB	\$A,\$D,\$4	
03388			*			
03389	2FA4	4D	MP27	FCC	/MY X TARGET POSITION IS =	/
03390	2FC0	04		FCB	\$4	
03391			*			
03392	2FC1	4D	MP28	FCC	/MY Y TARGET POSITION IS =	/
03393	2FDD	04		FCB	\$4	
03394			*			
03395	2FDE	58	MP29	FCC	/X AXIS INCORRECT FIXES =	/
03396	2FF7	04		FCB	\$4	
03397			*			
03398	2FF8	59	MP30	FCC	/Y AXIS INCORRECT FIXES =	/
03399	3011	04		FCB	\$4	
03400			*			
03401			*			
03402	3012	3F	SPEEDI	FCB	\$3F,\$38,\$30,\$28,\$20,\$18,\$15,\$12	
03403	301A	0F		FCB	\$0F,\$0B,\$09,\$08,\$07,\$06,\$05,\$04	
03404			*			
03405		0000		END		

TOTAL ERRORS 00000

MAIN LOOPFig. E. 1

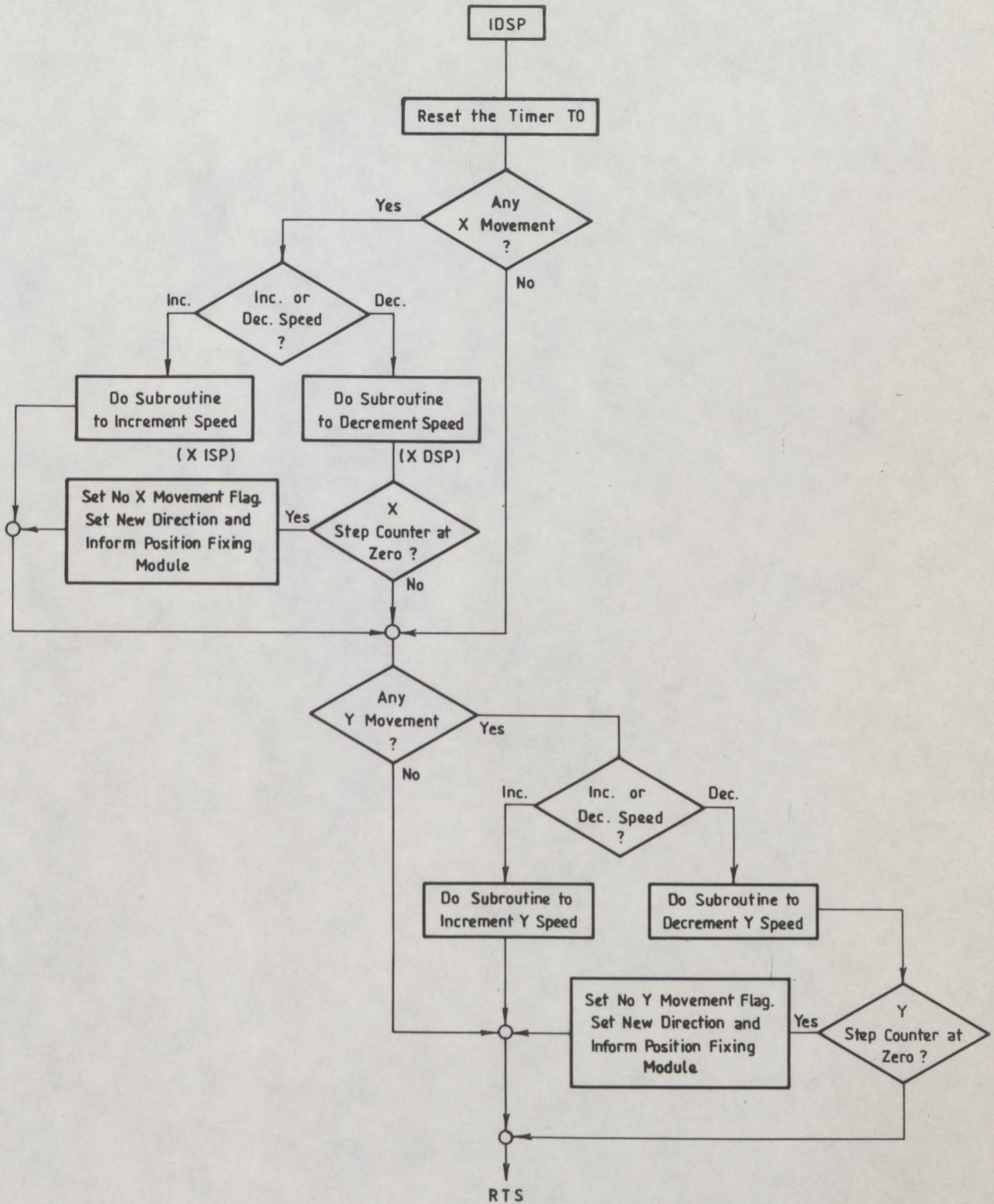
READ POSITION FROM POSITION MODULEFig. E.2TEST FOR ANY OVERRIDE REQUEST. (Motor Control Pushbuttons) Fig. E.3

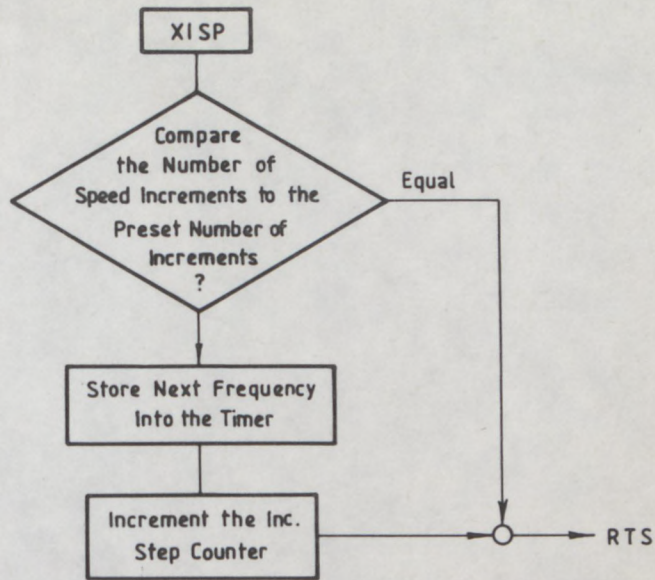




SETUP ROUTINE FOR MK2 SPEED CONTROLLER

Fig. E.6

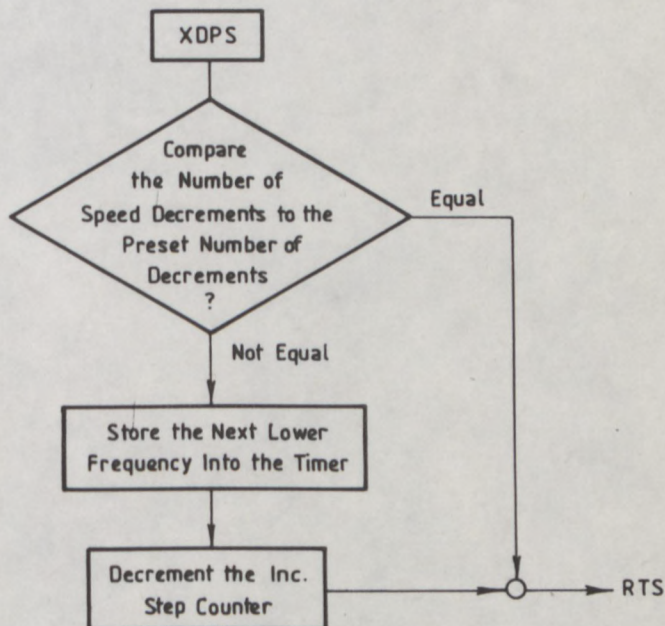




The Y-axis YISP is similar to the above X-axis XISP

INCREMENT SPEED

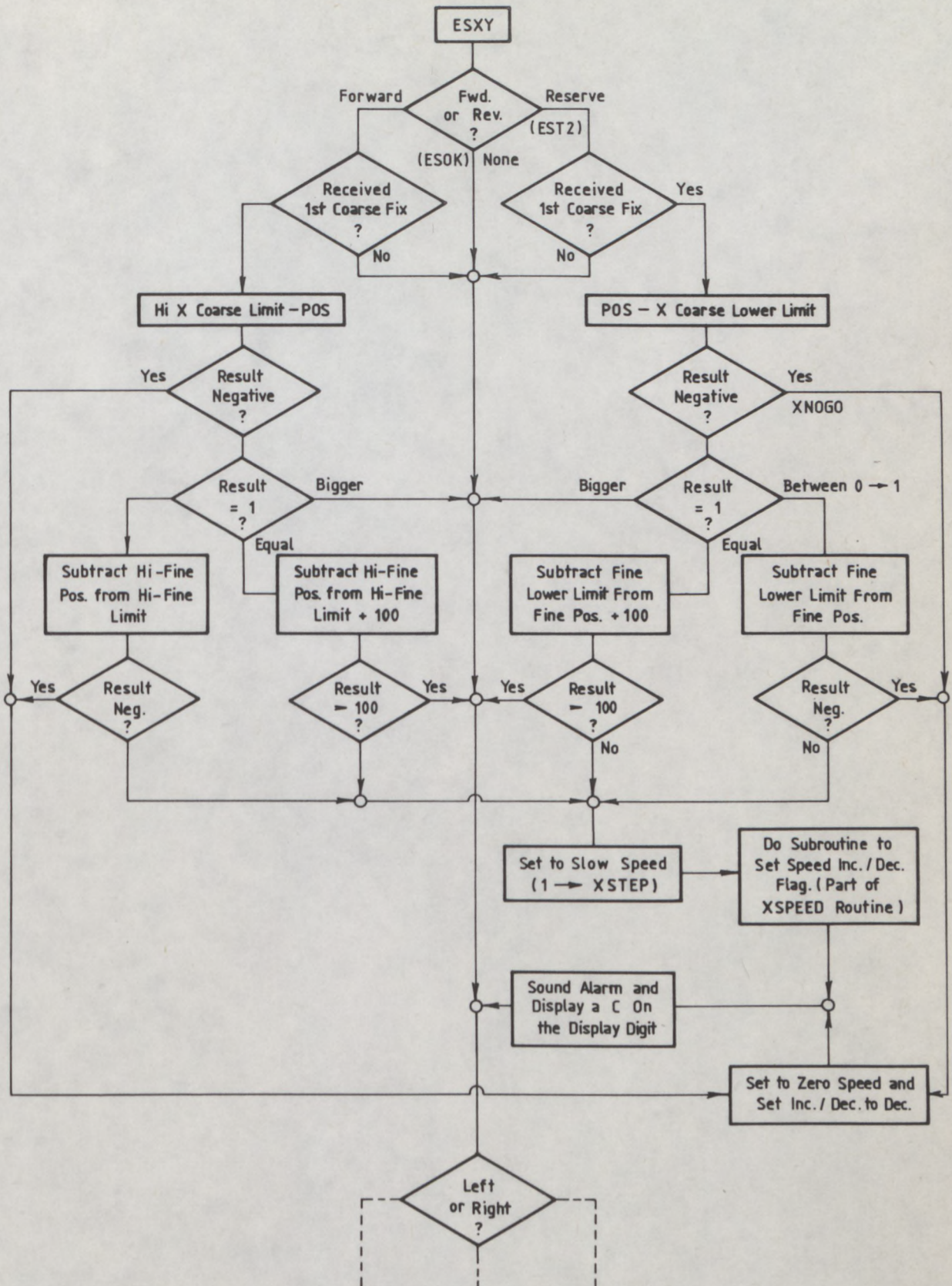
Fig. E.7 (b)



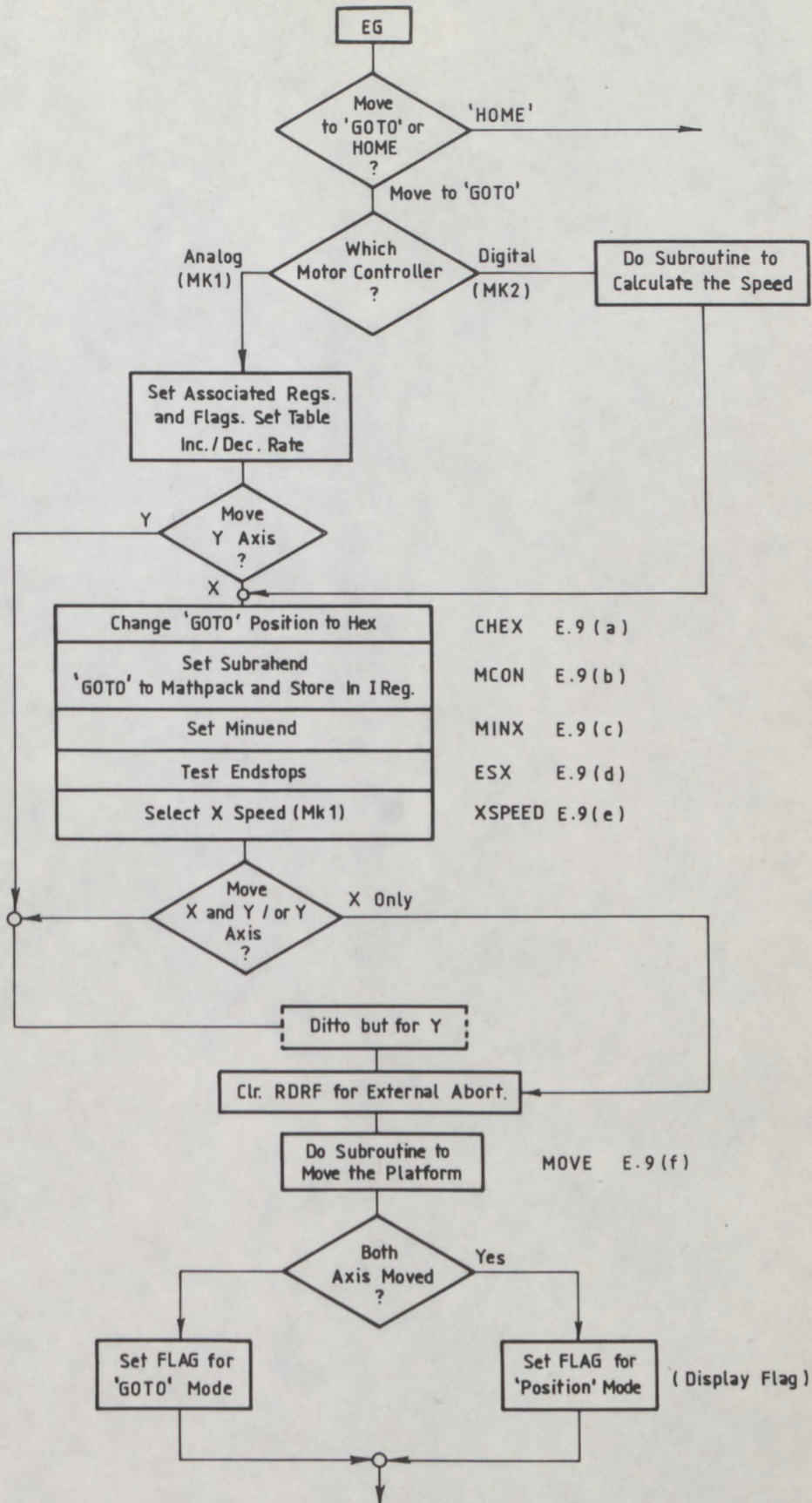
The Y-axis YDSP is similar to the above X-axis XDSP

DECREMENT SPEED

Fig. E.7 (c)

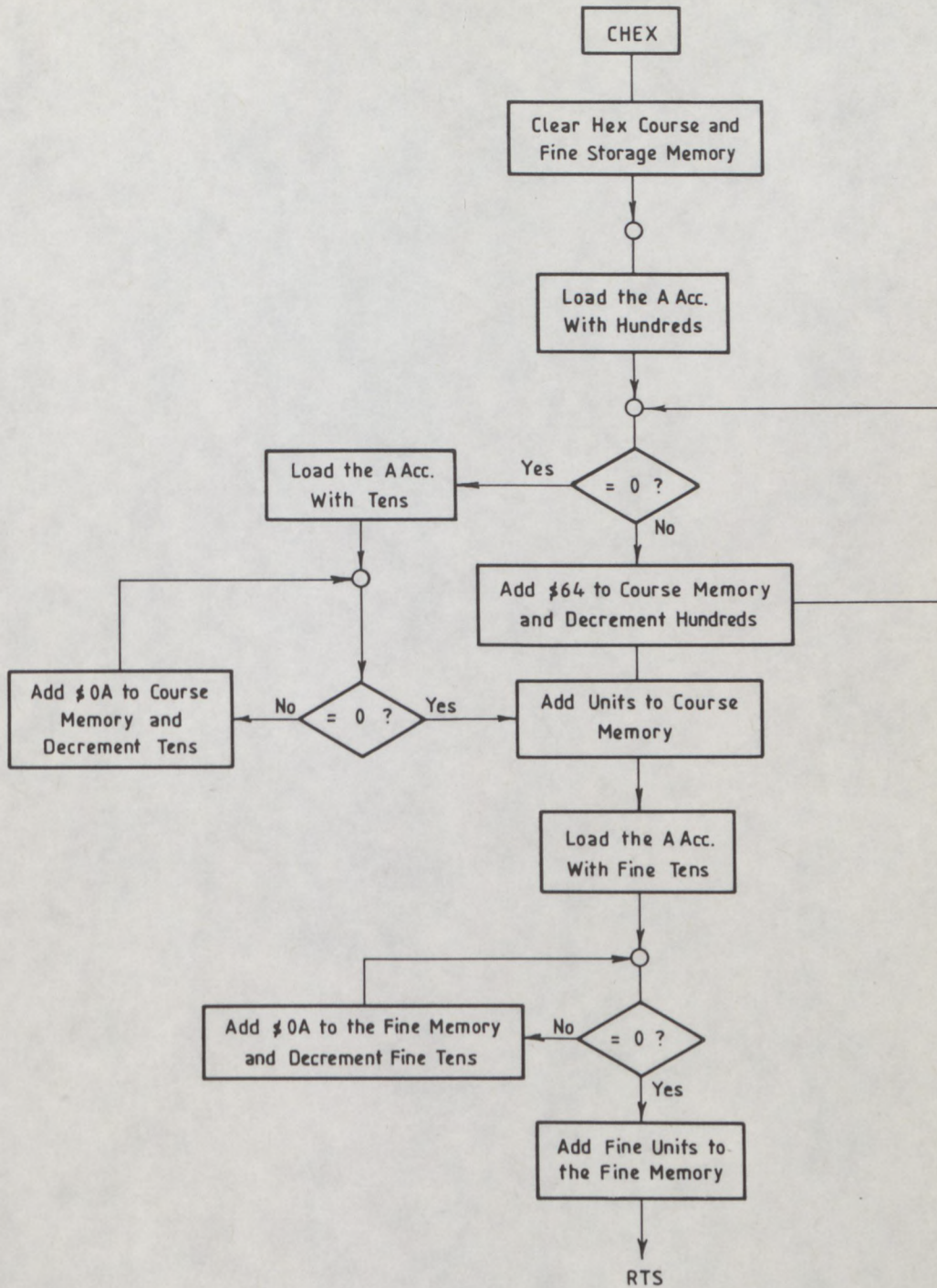


This Y Axis routine is similar to the X Axis routine above but uses different limits and scratchpad area.



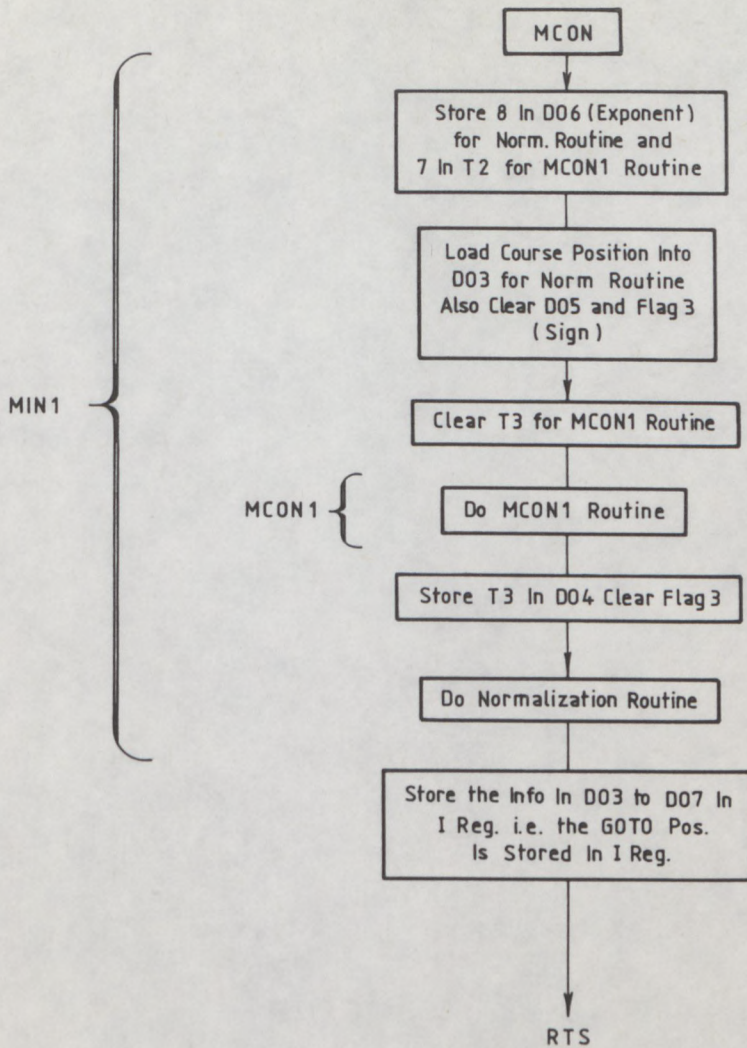
NON - OVERRIDE MOVE TO POSITION ROUTINE

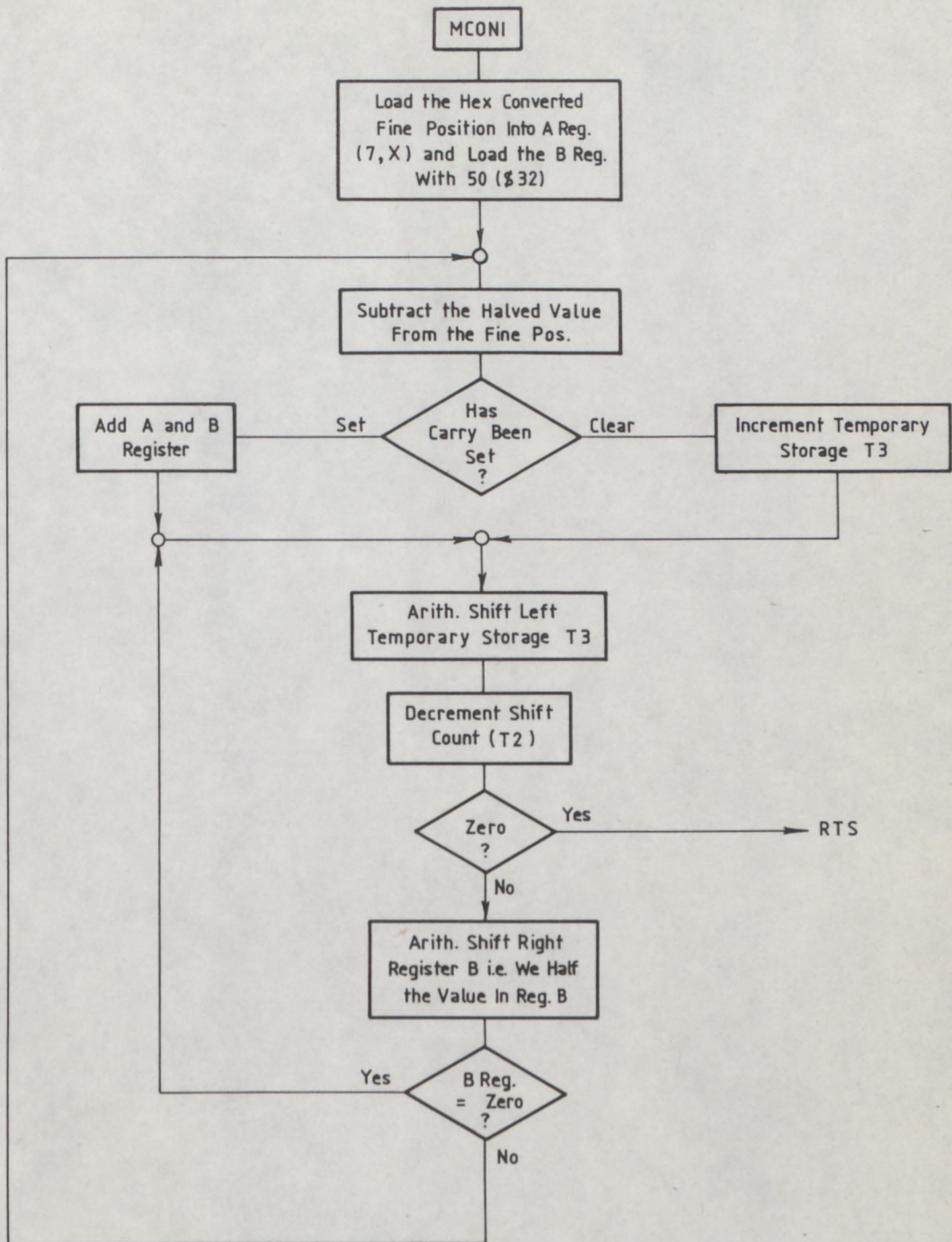
Fig. E.9



CHANGE THREE DIGIT DECIMAL TO HEX

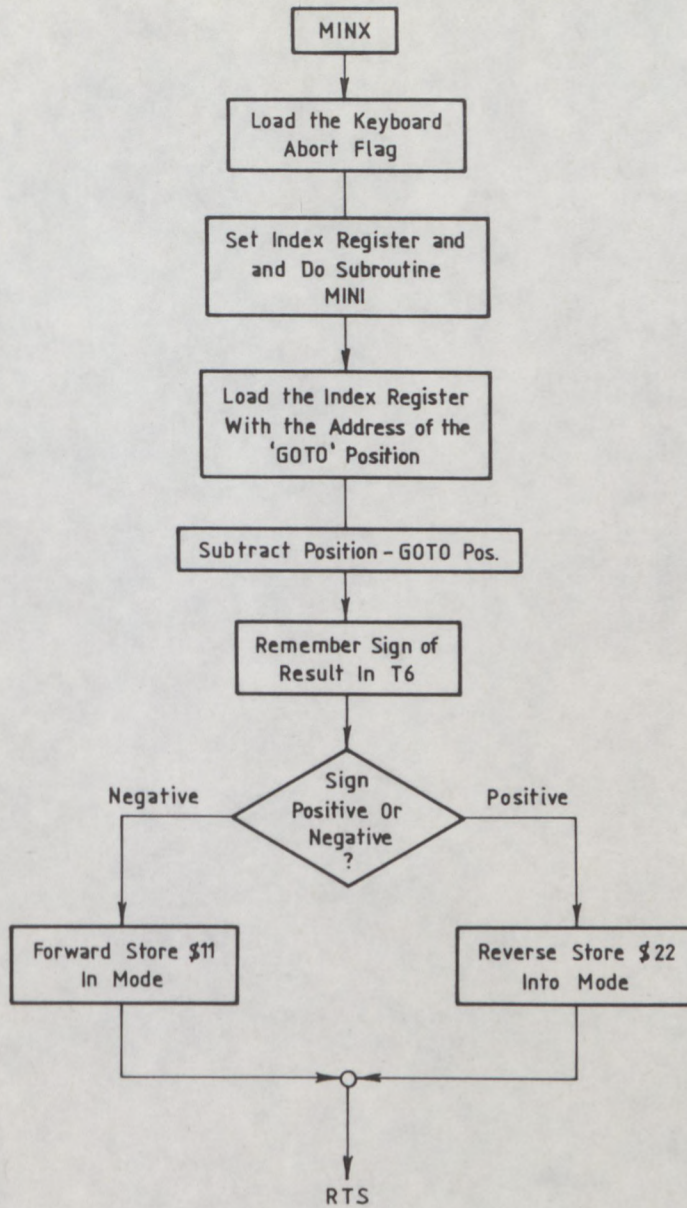
Fig. E.9 (a)



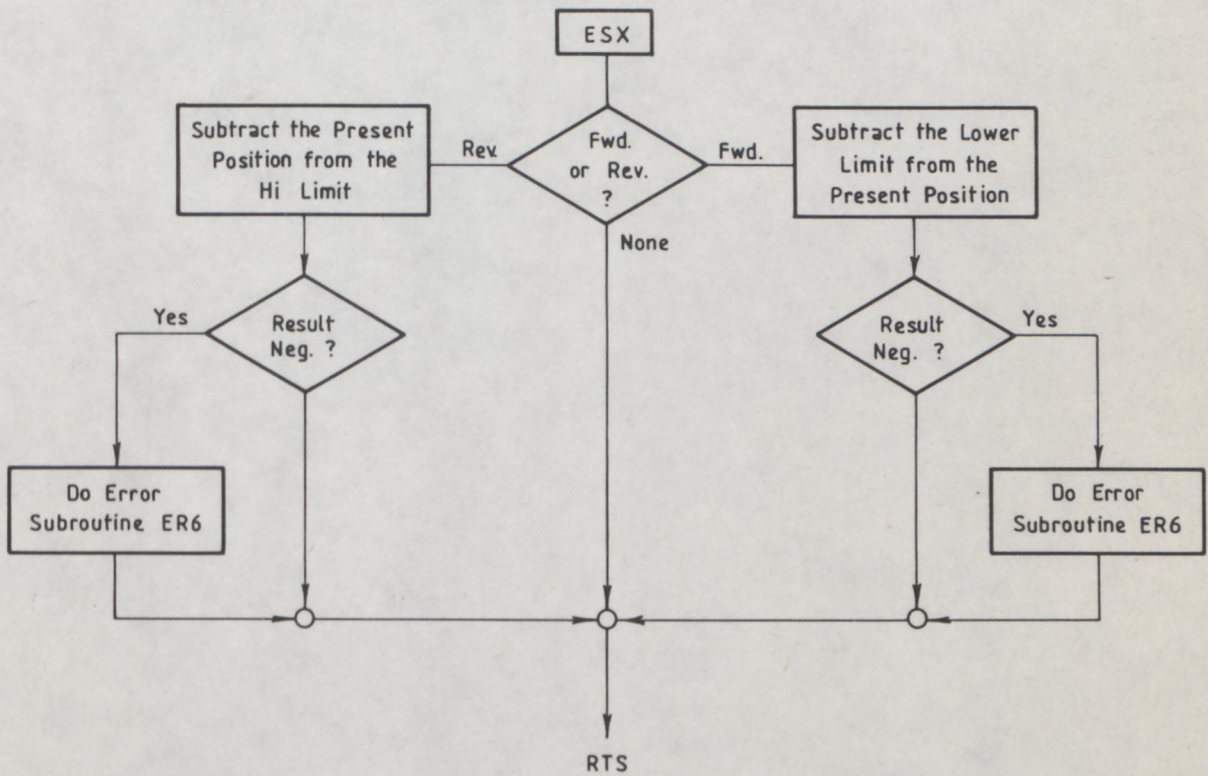


SET DECIMAL VALUE OF SUBRAHEND

Fig. E.9(b)(ii)



(MINY for Y - axis)



The routine for the Y axis (ESY) is similar

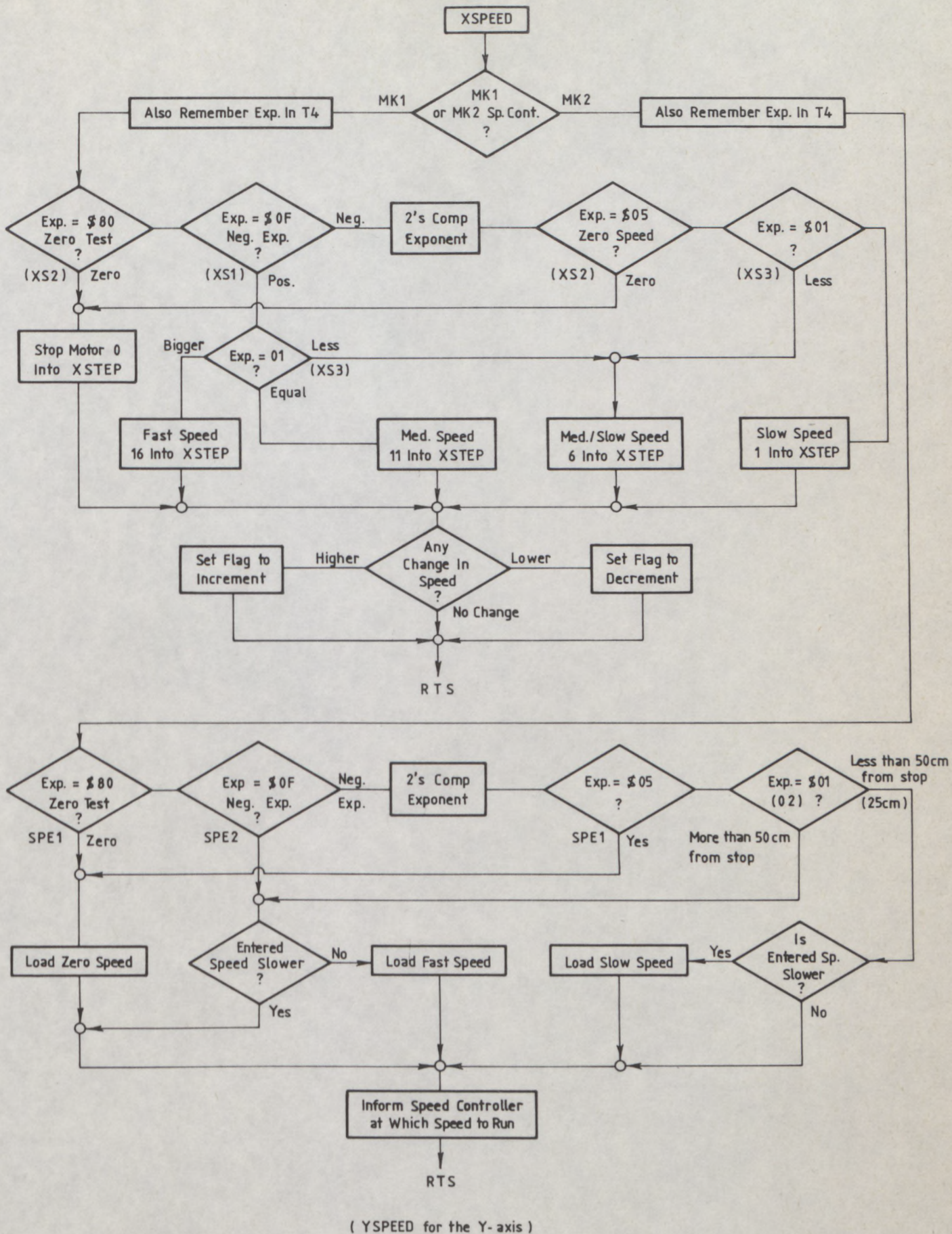
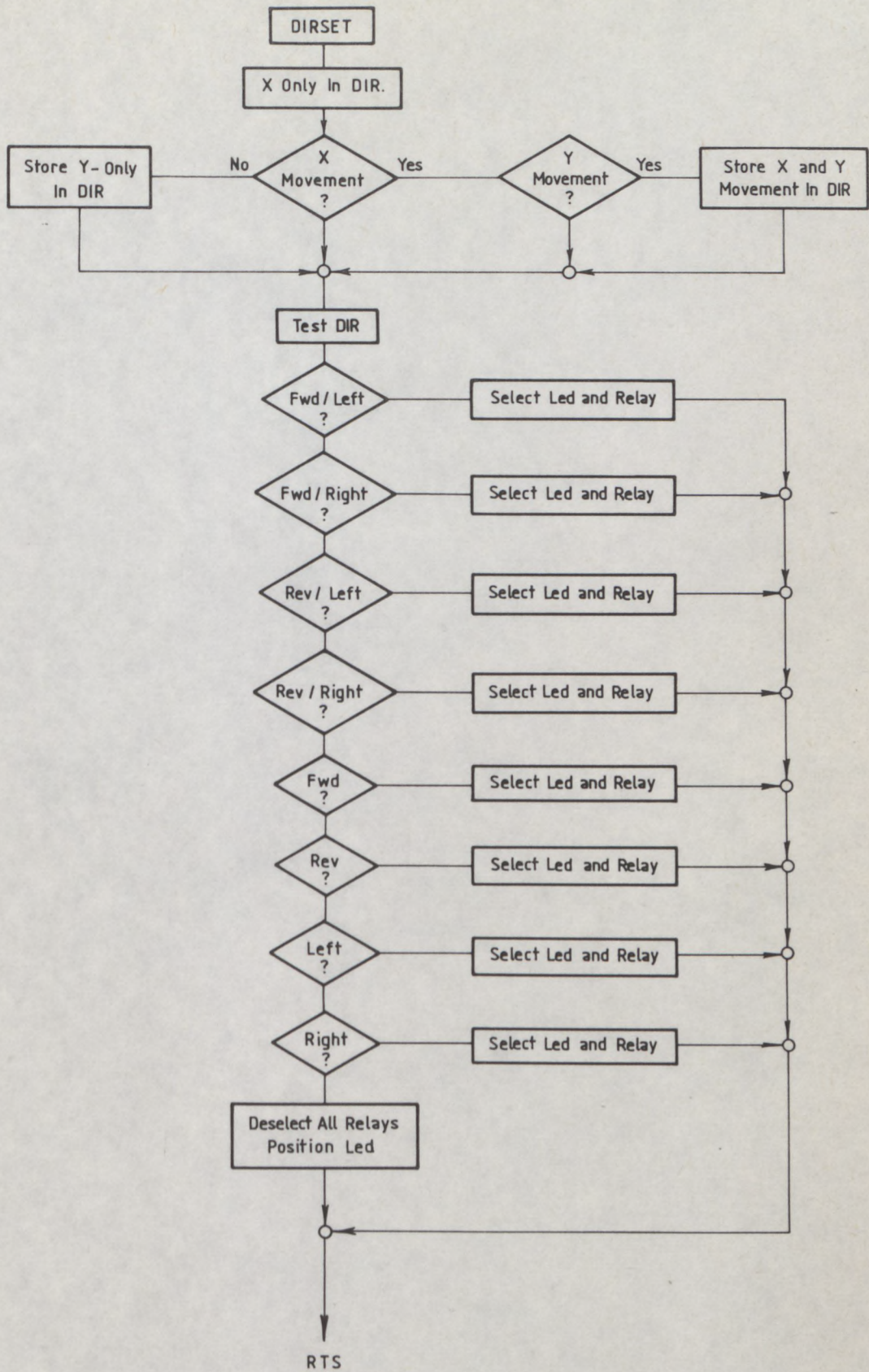
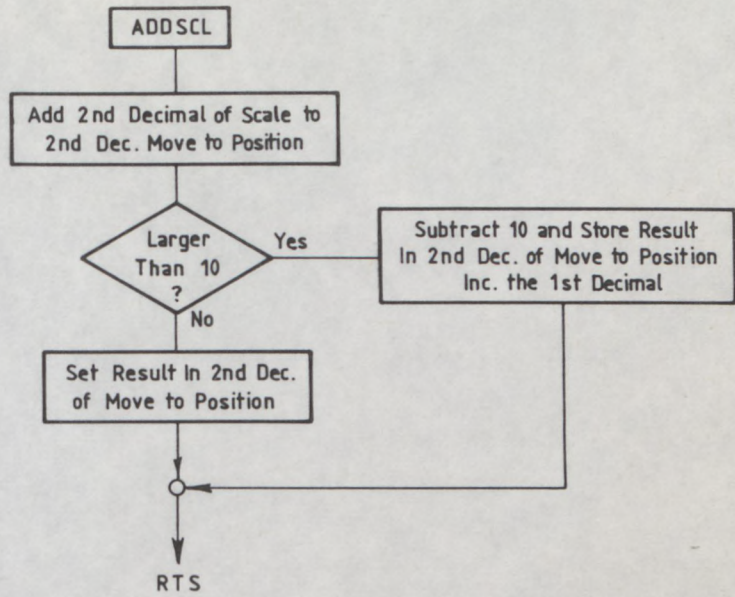


Fig. E.9 (e)



DIRECTION RELAY / LED SELECT ROUTINE

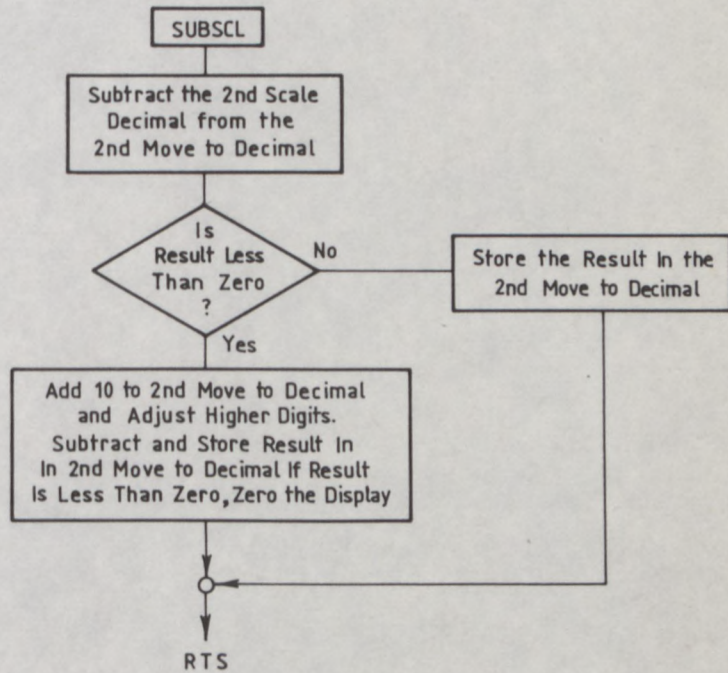
Fig. E. 10



(The same format is repeated for the 1st decimal and units)

ADD SCALE VALUE

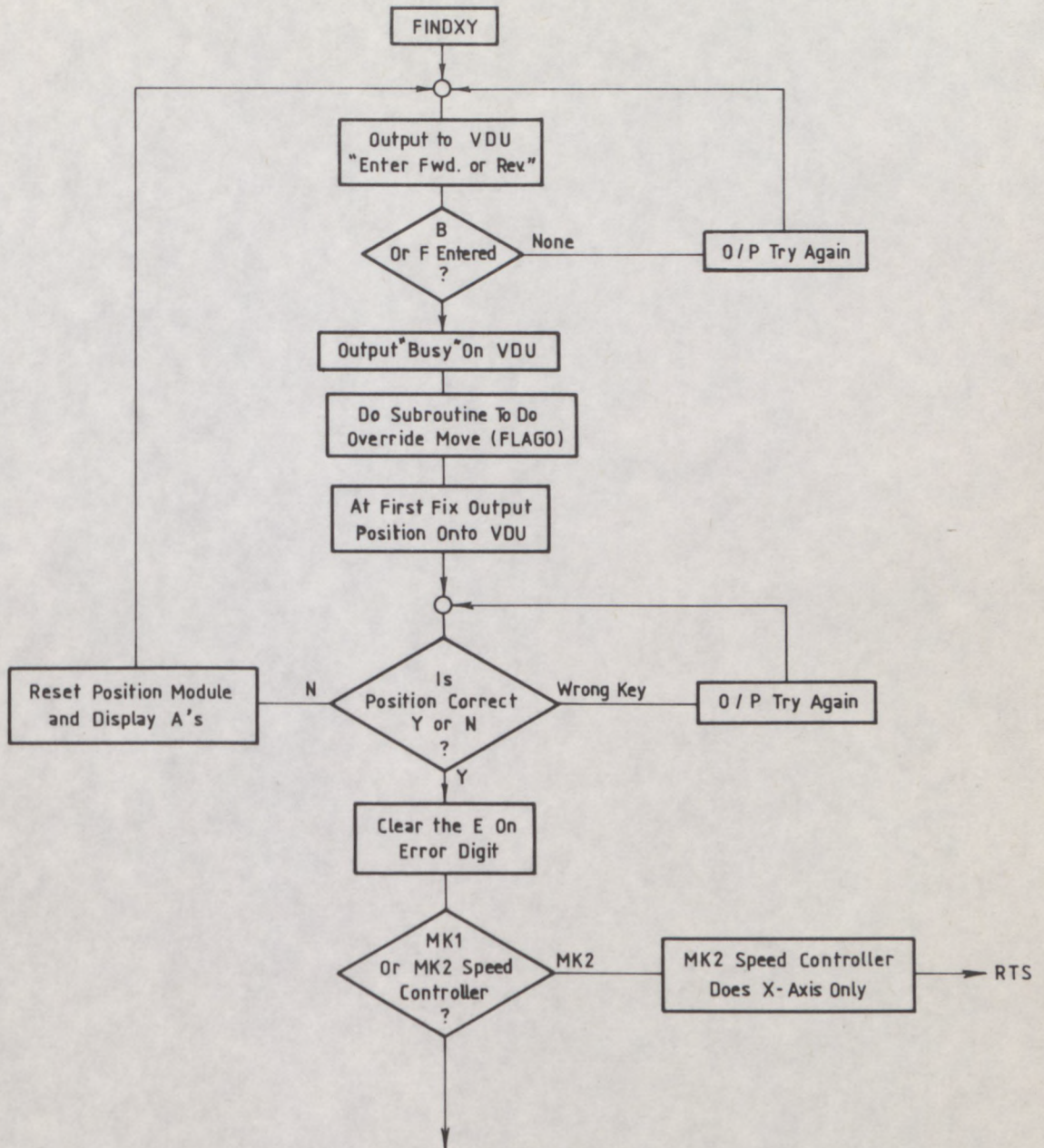
Fig. E.11 (a)



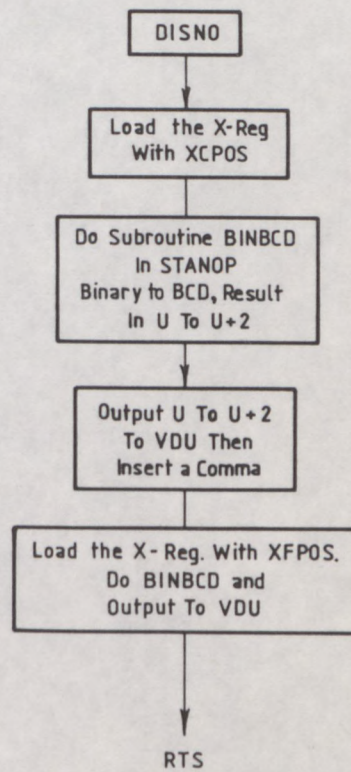
(The same format is repeated for the 1st decimal and units)

SUBTRACT SCALE VALUE

Fig. E.11 (b)

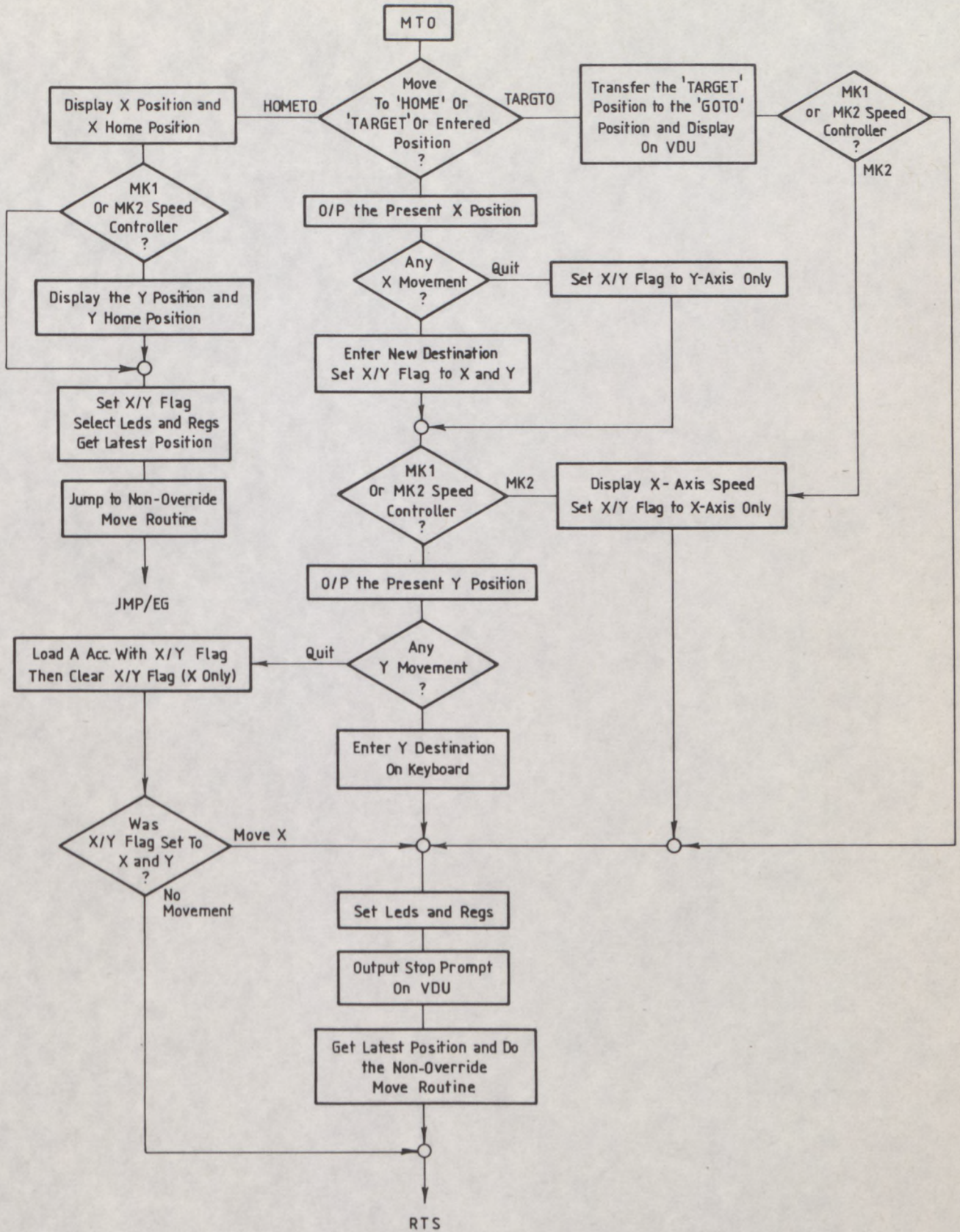


(To find the 1st Y-Axis fix a similar routine as the above 1st X - Axis fix routine is used.)



DISPLAY HEX VALUES IN XCPOS+3 AND XFPOS+3 ON VDU

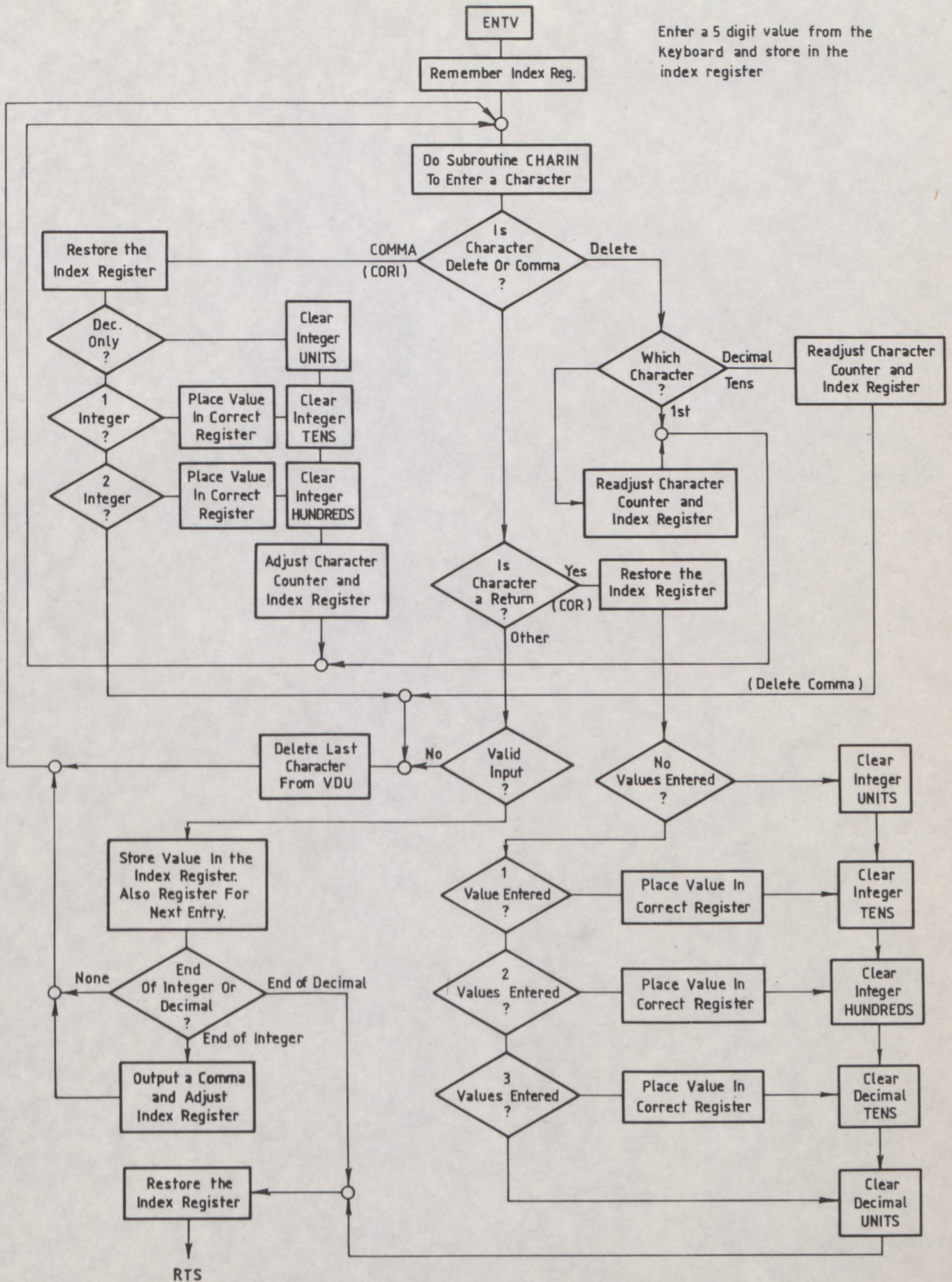
Fig. E.14

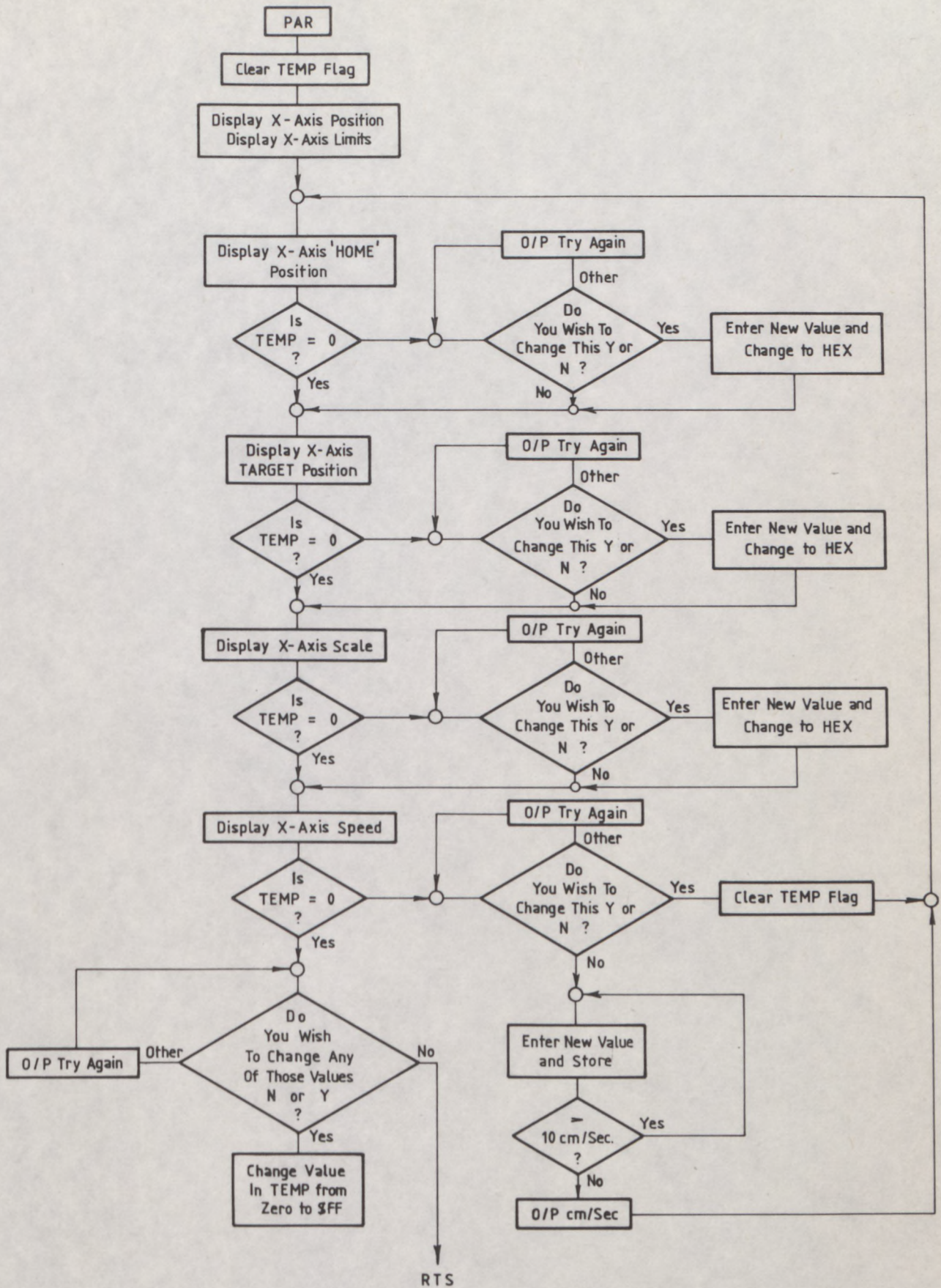


SETUP FOR EXTERNAL NON-OVERRIDE MOVE

Fig. E.15

Enter a 5 digit value from the
Keyboard and store in the
index register

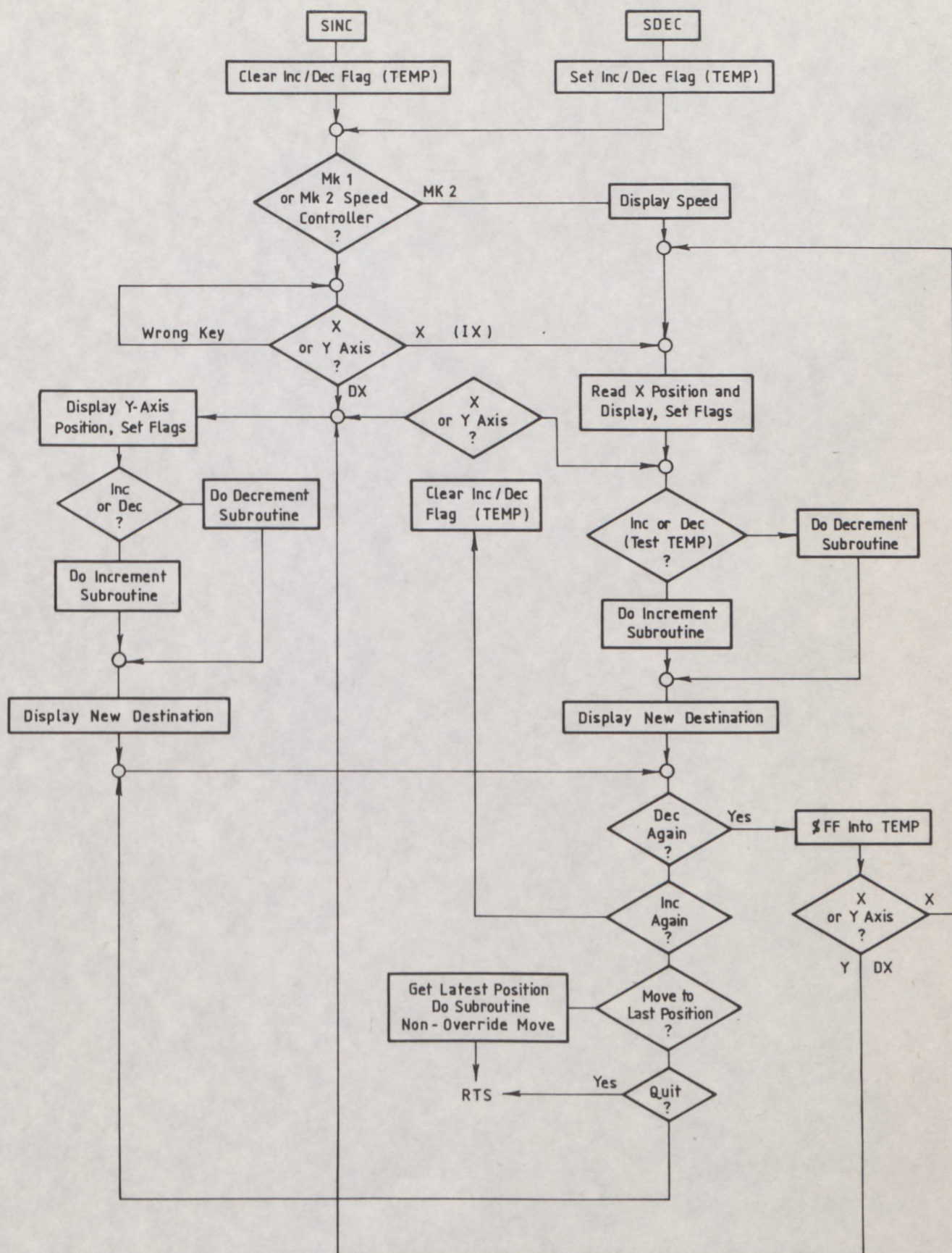




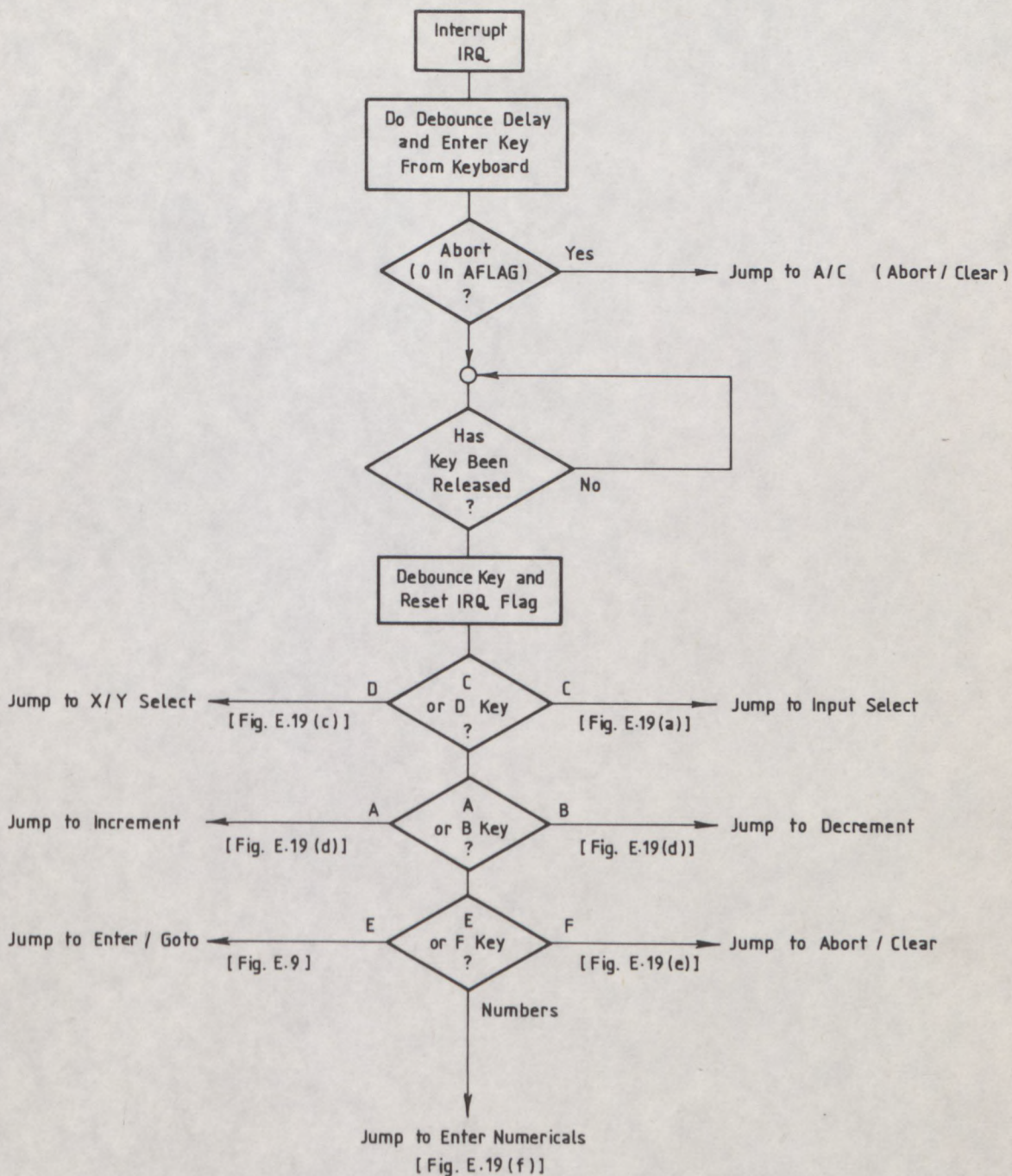
SET PARAMETERS

Fig. E.17

The routine for the Y - Axis is the same as for the above X - Axis



Increment or decrement 'GOTO' position by value stored in 'SCALE'.



MODE SELECT

Fig. E.19

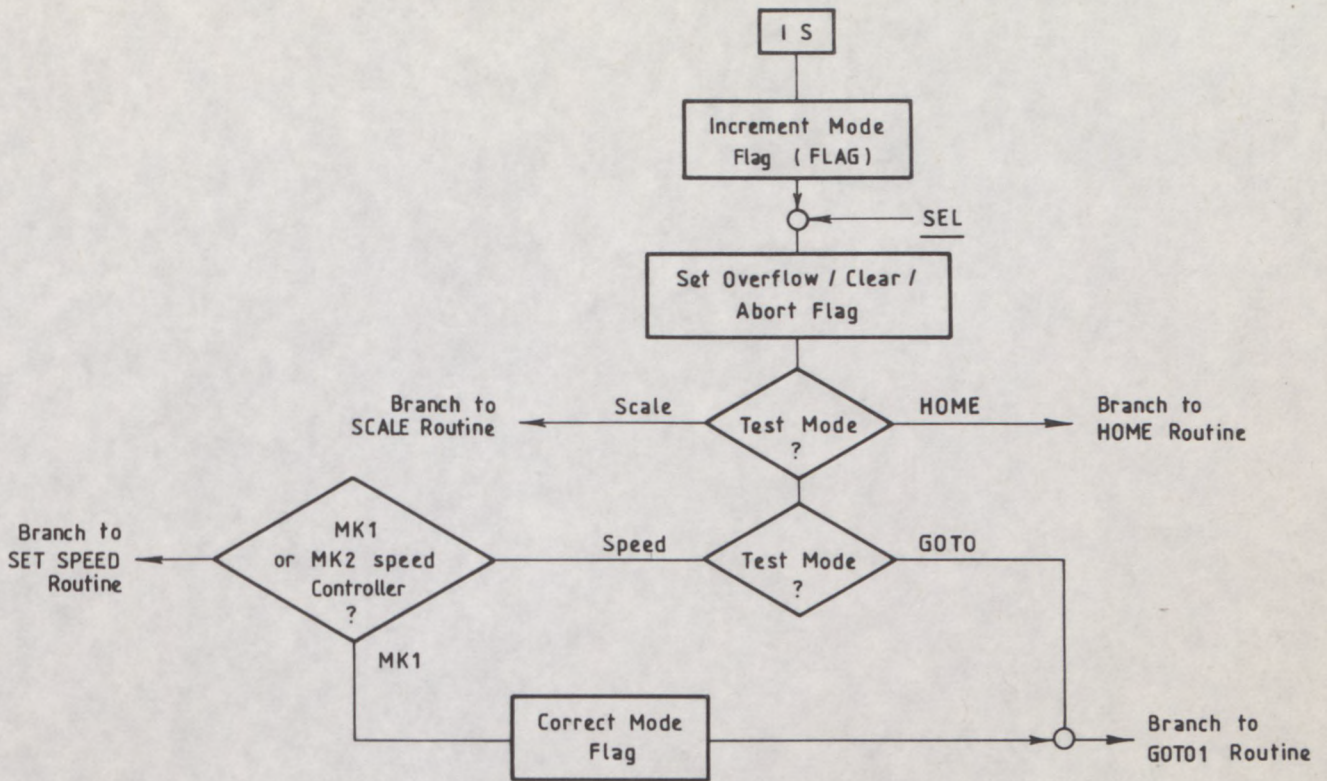
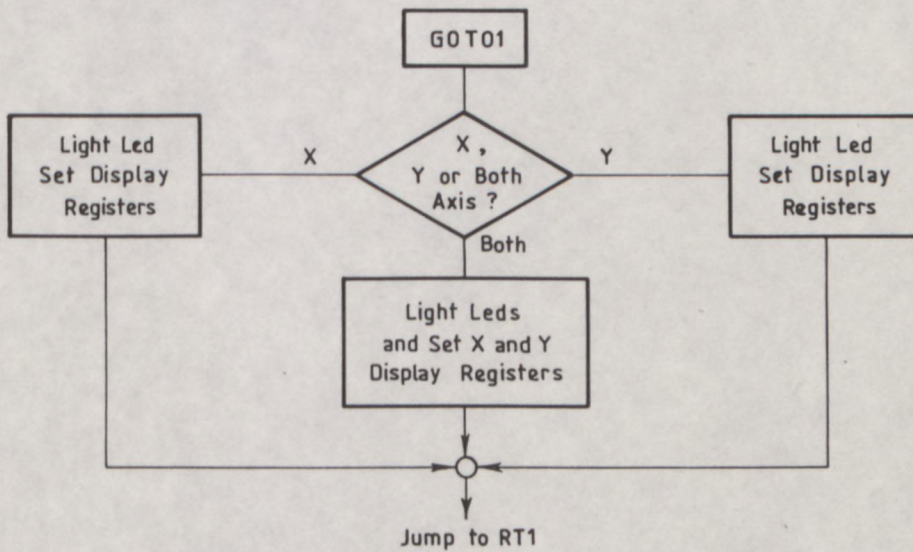
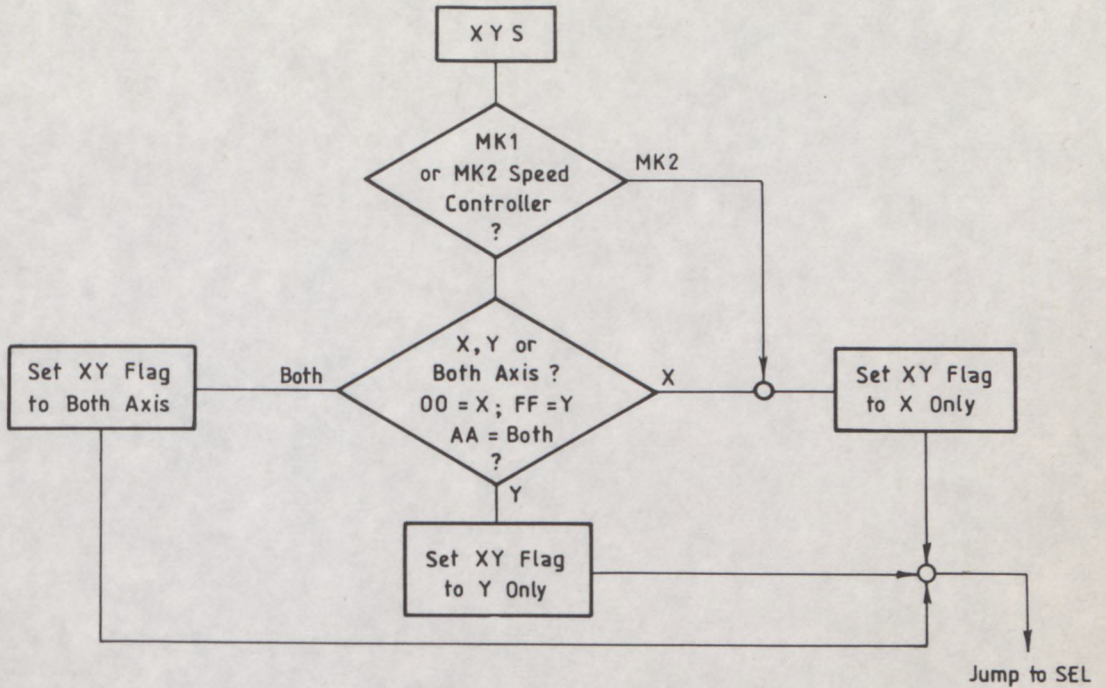
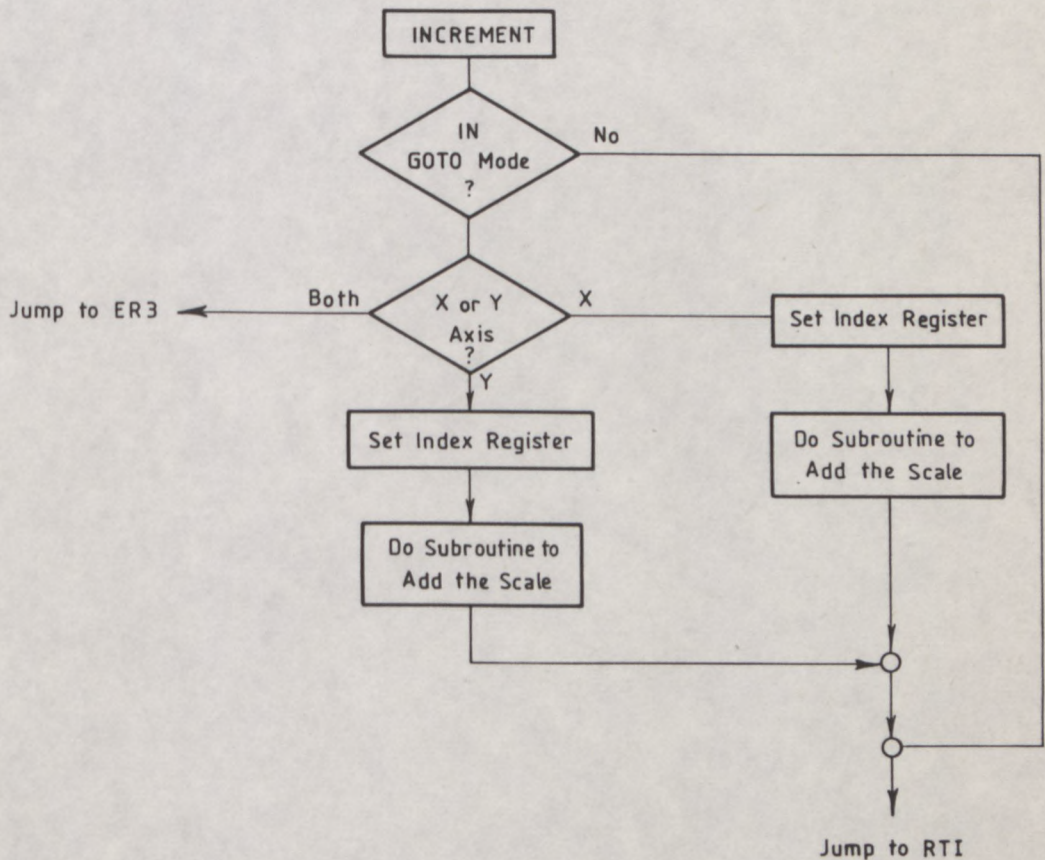


Fig. E.19 (a)

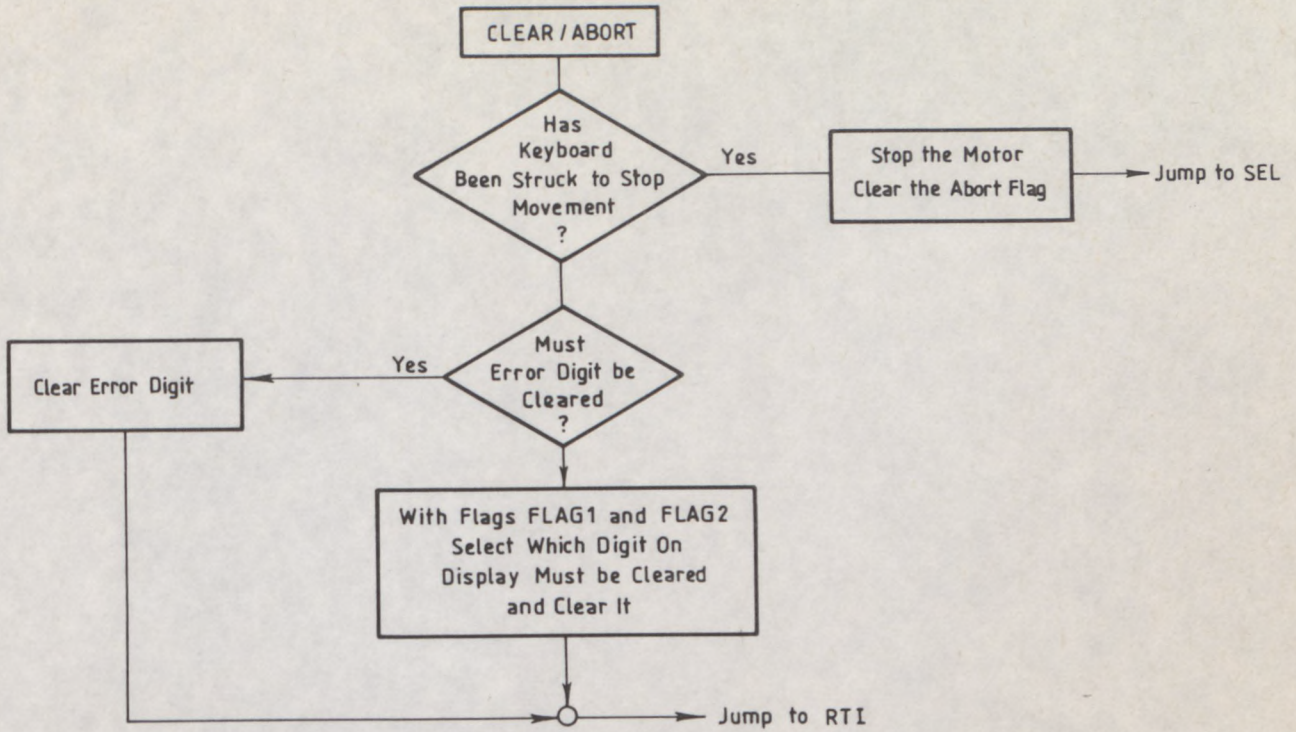
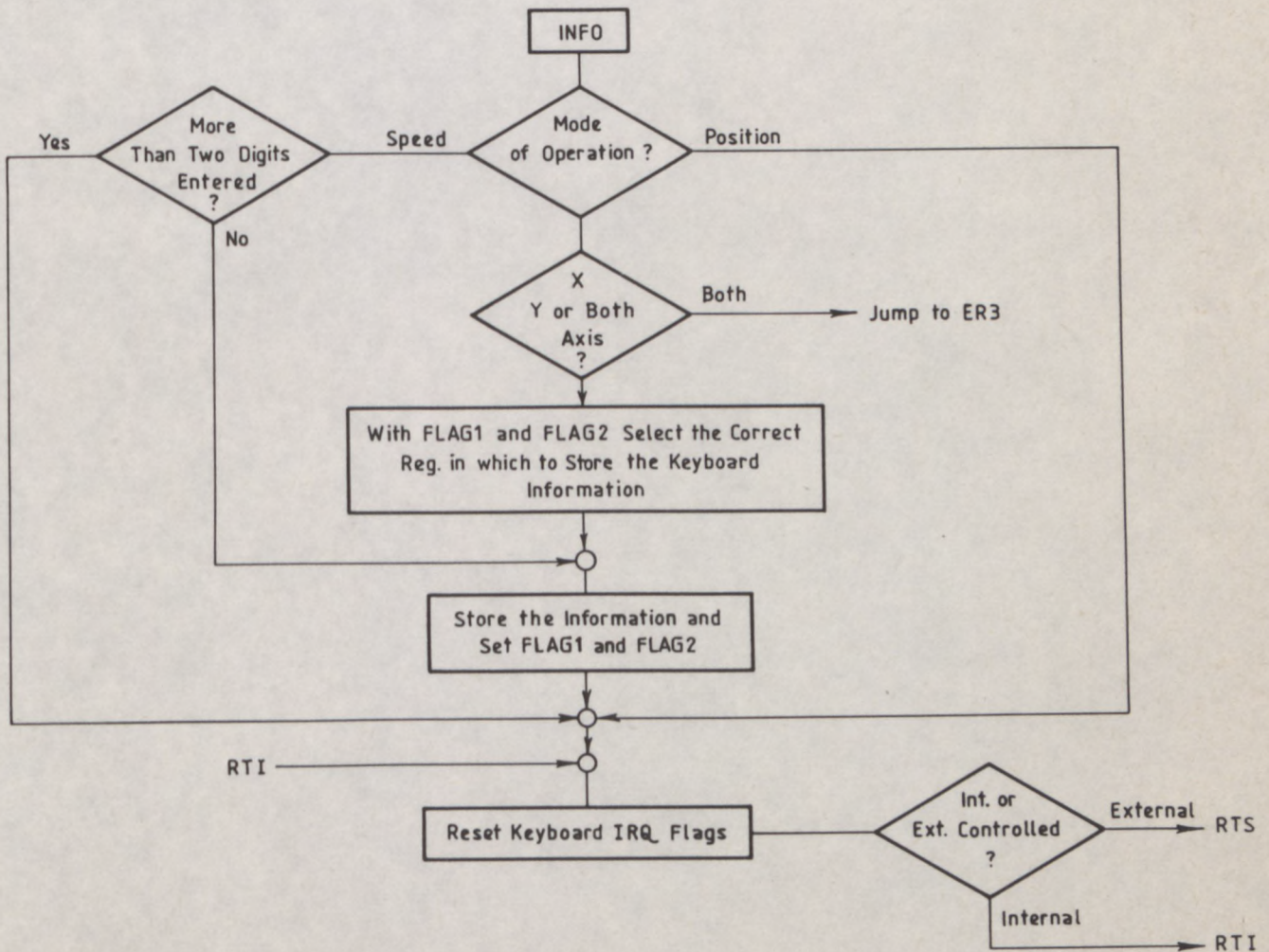


HOME and SCALE routines are the same as the above GOTO1 routine. The SPEED routine is the same but operates only on the X - axis.

X OR Y SELECTFig. E.19 (c)

Decrement is the same as Increment but does subroutine to subtract the scale

INCREMENTFig. E.19 (d)

CLEAR / ABORT ROUTINEFig. E.19 (e)ENTER NUMERICALSFig. E.19 (f)

APPENDIX FPOSITION-FIXING MODULE PROGRAM LISTING

The following is the assembler listing of the position-fixing module program. Position fixing is done with the aid of a one-board microcomputer, see Section 5.5.

Flow diagrams

Also included are flow diagrams of the position fixing module program.


```

00001          NAM      BC1
00002      * POSITION FIXING MODULE
00003      *
00004      * SYSTEM EQUATES
00005      00FF      STACK EQU      $FF
00006      001F      TARGET EQU     $1F      BUSY TEST
00007      *
00008      * PIA EQUATES
00009      1800      PIAD1 EQU      $1800      X AXIS
00010      1801      PIAC1 EQU      $1801
00011      1802      PIAD2 EQU      $1802      Y AXIS
00012      1803      PIAC2 EQU      $1803
00013      *
00014      *PTM EQUATES
00015      1400      CR13 EQU      $1400
00016      1401      CR2 EQU      $1401
00017      1402      T1M EQU      $1402
00018      1403      T1L EQU      $1403
00019      1404      T2M EQU      $1404
00020      1405      T2L EQU      $1405
00021      1406      T3M EQU      $1406
00022      1407      T3L EQU      $1407
00023      *
00024      *
00025      0080      ORG      $80
00026      *
00027      * RAM ALLOCATIOIS
00028      *
00029      0080 0004      XCPOS RMB      4      X-COARSE POSITION
00030      0084 0004      XFPOS RMB      4      X-FINE POSITION
00031      0088 0004      YCPOS RMB      4      Y-COARSE POSITION
00032      008C 0004      YFPOS RMB      4      Y-FINE POSITION
00033      0090 0001      XERR RMB      1      ERROR COUNTERS
00034      0091 0001      YERR RMB      1
00035      0092 0002      ST RMB      2      STACK REMEMBER
00036      0094 0001      TEMP2 RMB      1      FIRST X COARSE REG
00037      0095 0001      TEMP3 RMB      1      FIRST Y COARSE REG
00038      0096 0001      TEMP4 RMB      1
00039      0097 0001      TEMP5 RMB      1
00040      0098 0001      DIR RMB      1      DIRECTION FLAG
00041      0099 0001      DIRR RMB      1
00042      009A 0001      XIFLAG RMB      1      INTERRUPT FLAG
00043      009B 0001      XEFLAG RMB      1      ERROR FLAG
00044      009C 0001      YIFLAG RMB      1
00045      009D 0001      YEFLAG RMB      1
00046      009E 0001      XFLAG RMB      1      +/- 5CM FLAGS
00047      009F 0001      YFLAG RMB      1
00048      *
00049      0000      ORG      $00
00050      *
00051      0000 0004      XCPOS1 RMB      4      BC2/BC1 INTERMEDIATE X-COARSE
00052      0004 0004      XFPOS1 RMB      4      INTERMEDIATE X-FINE
00053      0008 0004      YCPOS1 RMB      4      INTERMEDIATE Y-COARSE
00054      000C 0004      YFPOS1 RMB      4      INTERMEDIATE Y-FINE
00055      0010 0001      XER RMB      1
00056      0011 0001      YER RMB      1
00057      0012 0001      DIR1 RMB      1      DIRECTION LINK
00058      *

```



```

00059      *
00060      *   DIRECTION CONTROL WORDS
00061      *   $00-----HALT
00062      *   $11-----FORWARD
00063      *   $22-----REVERSE
00064      *   $44-----LEFT
00065      *   $88-----RIGHT
00066      *   $14-----FWD/LEFT
00067      *   $18-----FWD/RIGHT
00068      *   $24-----REV/LEFT
00069      *   $28-----REV/RIGHT
00070      *
00071      *
00072      * START-UP PROGRAM
00073      *
00074 F800      ORG      $F800
00075 F800 8E 00FF START LDS      #STACK
00076 F803 0F      SEI
00077 F804 86 55      LDA A      #$55
00078 F806 97 1F      STA A      TARGET      SET FUCTION BOARD
00079 F808 86 00      LDA A      #$00
00080 F80A B7 1801     STA A      PIAC1      ACCESS DDR
00081 F80D B7 1803     STA A      PIAC2
00082 F810 B7 1800     STA A      PIAD1      INPUT PORTS
00083 F813 B7 1802     STA A      PIAD2
00084 F816 86 1F      LDA A      #$1F
00085 F818 B7 1801     STA A      PIAC1      MODE
00086 F81B 86 3F      LDA A      #$3F      CB2-HI      $37 FOR CB2-LOW
00087 F81D B7 1803     STA A      PIAC2
00088 F820 86 00      LDA A      #$00
00089 F822 B7 1401     STA A      CR2      SELECT CR3
00090 F825 C6 80      LDA B      #$80
00091 F827 F7 1400     STA B      CR13      TH3 MODE
00092 F82A 86 83      LDA A      #$83
00093 F82C B7 1401     STA A      CR2      TH2 MODE
00094 F82F F7 1400     STA B      CR13      TH1 MODE
00095 F832 CE 01F4     LDX      #$01F4
00096 F835 FF 1404     STX      T2M      1KHZ OSC
00097 F838 CE 0B00     LDX      #$0B00
00098 F83B FF 1402     STX      T1M      X TO DELAY
00099 F83E FF 1406     STX      T3M      Y TO DELAY
00100 F841 86 0A      LDA A      #$0A
00101 F843 5F      CLR B
00102 F844 CE 0000     LDX      #XCPOS1
00103 F847 A7 00      CL      STA A      0,X      CLEAR MEMORY
00104 F849 08      INX
00105 F84A 5C      INC B
00106 F84B C1 12      CMP B      #$12
00107 F84D 26 F8      BNE      CL
00108 F84F 5F      CLR B
00109 F850 CE 0080     LDX      #XCPOS
00110 F853 A7 00      CL1     STA A      0,X
00111 F855 08      INX
00112 F856 5C      INC B
00113 F857 C1 20      CMP B      #$20
00114 F859 26 F8      BNE      CL1
00115 F85B 86 AA      LDA A      #$AA      LOAD POS. MEM. WITH A'S
00116 F85D 97 03      STA A      XCPOS1+3

```



```

00117 F85F 97 0B      STA A   YCPOS1+3
00118 F861 97 83      STA A   XCPOS+3
00119 F863 97 8B      STA A   YCPOS+3
00120 F865 86 FF      LDA A   #$FF
00121 F867 97 12      STA A   DIR1
00122 F869 97 94      STA A   TEMP2
00123 F86B 97 95      STA A   TEMP3
00124 F86D 43         COM A
00125 F86E 97 98      STA A   DIR
00126 F870 97 99      STA A   DIRR
00127 F872 97 96      STA A   TEMP4
00128 F874 97 97      STA A   TEMP5
00129                 *
00130                 *
00131                 * MAIN LOOP
00132                 *
00133 F876 0F          BEGIN   SEI
00134 F877 BD F8C1     JSR     CREG      STORE POS. IN INTERMEDIATE ME
00135 F87A BD F8EA B1  JSR     CDIR      READ THE DIRECTION OF MOVEMEN
00136 F87D 7D 0098     TST     DIR      SET?
00137 F880 26 0E       BNE     NCLR      YES
00138 F882 BD FB76     JSR     DELAY     GIVE BC1 TIME TO READ MEM
00139 F885 7F 0099     CLR     DIRR
00140 F888 7F 0090     CLR     XERR
00141 F88B 7F 0091     CLR     YERR
00142 F88E 20 E6       BRA     BEGIN
00143 F890 96 98       NCLR    LDA A   DIR      X-AXIS
00144 F892 91 99       CMP A   DIRR
00145 F894 27 24       BEQ     NCLR3     NO DIRECTION CHANGE
00146 F896 D6 99       LDA B   DIRR     REMEMBERED DIRECTION
00147 F898 C4 30       AND B   #$30     MASK
00148 F89A 84 30       AND A   #$30     MASK
00149 F89C 11         CBA
00150 F89D 27 06       BEQ     NCLR1     NO CHANGE
00151 F89F 4D         TST A
00152 F8A0 27 03       BEQ     NCLR1     CHANGED TO ZERO
00153 F8A2 BD FB89     JSR     PUTX      COMPUTE NEXT X VALUE
00154 F8A5 96 98       NCLR1  LDA A   DIR      Y-AXIS
00155 F8A7 D6 99       LDA B   DIRR
00156 F8A9 C4 0C       AND B   #$0C
00157 F8AB 84 0C       AND A   #$0C
00158 F8AD 11         CBA
00159 F8AE 27 06       BEQ     NCLR2     NO CHANGE
00160 F8B0 4D         TST A
00161 F8B1 27 03       BEQ     NCLR2     CHANGED TO ZERO
00162 F8B3 BD FB9F     JSR     PUTY      COMPUTE NEXT Y VALUE
00163 F8B6 96 98       NCLR2  LDA A   DIR
00164 F8B8 97 99       STA A   DIRR     REMEMBER PRESENT DIRECTIONS
00165 F8BA 0E         NCLR3  CLI
00166 F8BB BD FB76     JSR     DELAY
00167 F8BE 0F         SEI
00168 F8BF 20 B9       BRA     B1
00169                 *
00170                 * SUBROUTINES
00171                 *
00172                 */CREG/
00173                 *THIS SUB STORES THE POSITION IN
00174                 *THE INTERMEDIATE MEMORY

```



```

00175 F8C1 86 3F    CREG    LDA A    #$3F
00176 F8C3 B7 1803    STA A    PIAC2    SET BUSY FLAG
00177 F8C6 86 55    LDA A    #$55    SET TARGET WORD
00178 F8C8 97 1F    STA A    TARGET
00179 F8CA 9F 92    STS    ST    STORE STACK
00180 F8CC 8E 0011    LDS    #YER
00181 F8CF C6 12    LDA B    #$12
00182 F8D1 CE 0091    LDX    #YERR
00183 F8D4 A6 00    AND    LDA A    0,X
00184 F8D6 43    COM A
00185 F8D7 36    PSH A    INVERTED BUS
00186 F8D8 09    DEX    LOAD INTO INTERMEDIATE MEMORY
00187 F8D9 5A    DEC B
00188 F8DA 27 02    BEQ    FIN
00189 F8DC 20 F6    BRA    AND
00190 F8DE 9E 92    FIN    LDS    ST    RELOAD STACK
00191 F8E0 86 AA    LDA A    #$AA
00192 F8E2 97 1F    STA A    TARGET    RESET TARGET WORD
00193 F8E4 86 37    LDA A    #$37
00194 F8E6 B7 1803    STA A    PIAC2    RESET BUSY FLAG
00195 F8E9 39    RTS
00196
00197    *
00198    */CDIR/
00199    *THIS SUB READS THE DIRECTION OF MOVEMENT
00200    *FROM THE INT. MEMORY
00200 F8EA 86 3F    CDIR    LDA A    #$3F
00201 F8EC B7 1803    STA A    PIAC2
00202 F8EF 86 55    LDA A    #$55
00203 F8F1 97 1F    STA A    TARGET    TARGET WORD
00204 F8F3 96 12    LDA A    DIR1
00205 F8F5 43    COM A    INVERTED BUS
00206 F8F6 97 98    STA A    DIR    TRANSFER TO BC1
00207 F8F8 86 AA    LDA A    #$AA
00208 F8FA 97 1F    STA A    TARGET    RESET TARGET WORD
00209 F8FC 86 37    LDA A    #$37
00210 F8FE B7 1803    STA A    PIAC2    RESET BUSY FLAG
00211 F901 39    RTS
00212
00213    *
00214    */XFS/
00215    * THIS SUB. SETS THE X AXIS FINE POSITION
00215 F902 7D 0094    XFS    TST    TEMP2    1st COARSE FIX FLAG
00216 F905 27 03    BEQ    XFS2
00217 F907 7E F9CA    XFS1    JMP    XFSR    NO COARSE FIX YET
00218 F90A 96 98    XFS2    LDA A    DIR    TEST DIRECTION
00219 F90C 84 30    AND A    #$30    MASK
00220 F90E 27 F7    BEQ    XFS1
00221 F910 81 10    CMP A    #$10    FORWARD?
00222 F912 27 54    BEQ    XFS7    YES
00223 F914 96 87    LDA A    XFPOS+3    NO...TEST FINE POSITION
00224 F916 27 30    BEQ    XFS4    ZERO
00225 F918 7D 009E    TST    XFLAG    5CM PLAY FLAG
00226 F91B 27 0E    BEQ    XFS3
00227 F91D 81 5E    CMP A    #$5E    BETWEEN 95 - 100
00228 F91F 24 0A    BCC    XFS3    YES
00229 F921 81 06    CMP A    #$06    BETWEEN 0 - 5
00230 F923 25 06    BCS    XFS3    YES
00231 F925 7F 009E    CLR    XFLAG
00232 F928 7A 0096    DEC    TEMP4

```


00233	F92B	7A	0087	XFS3	DEC	XFPOS+3	DEC FINE POS
00234	F92E	7D	009B		TST	XEFLAG	TEST ERROR FLAG
00235	F931	27	13		BEQ	XF3B	NOT SET
00236	F933	81	50		CMP A	#50	(80)
00237	F935	2A	0F		BPL	XF3B	
00238	F937	96	90		LDA A	XERR	
00239	F939	81	09		CMP A	#09	9 ERRORS MAX
00240	F93B	27	03		BEQ	XF3A	
00241	F93D	7C	0090		INC	XERR	INCREMENT ERROR COUNTER
00242	F940	7F	009B	XF3A	CLR	XEFLAG	CLEAR ERROR FLAG
00243	F943	7A	0096		DEC	TEMP4	NEXT COARSE VALUE
00244	F946	20	76	XF3B	BRA	XFS12	
00245	F948	7D	009E	XFS4	TST	XFLAG	
00246	F94B	26	09		BNE	XFS5	STARTED AT ZERO DONT SET ERRO
00247	F94D	7D	009A		TST	XIFLAG	TEST INTERRUPT FLAG
00248	F950	27	0D		BEQ	XFS6	0 IF SET
00249	F952	86	FF		LDA A	#FF	
00250	F954	97	9B		STA A	XEFLAG	SET ERROR FLAG
00251	F956	7A	0083	XFS5	DEC	XCPOS+3	DECREMENT COARSE POSITION
00252	F959	86	63		LDA A	#63	(99)
00253	F95B	97	87		STA A	XFPOS+3	MAX FINE VALUE
00254	F95D	20	5F		BRA	XFS12	
00255	F95F	86	FF	XFS6	LDA A	#FF	
00256	F961	97	9A		STA A	XIFLAG	RESET INT. FLAG
00257	F963	7F	009B		CLR	XEFLAG	CLEAR ERROR FLAG
00258	F966	20	EE		BRA	XFS5	
00259	F968	96	87	XFS7	LDA A	XFPOS+3	
00260	F96A	27	34		BEQ	XFS10	FINE POS = 0?
00261	F96C	7D	009E		TST	XFLAG	5CM PLAY FLAG
00262	F96F	27	0E		BEQ	XFS8	
00263	F971	81	5E		CMP A	#5E	BETWEEN 95-99CM
00264	F973	24	0A		BCC	XFS8	
00265	F975	81	06		CMP A	#06	BETWEEN 0-5CM
00266	F977	25	06		BCS	XFS8	
00267	F979	7F	009E		CLR	XFLAG	+/- FLAG CLEARED
00268	F97C	7C	0096		INC	TEMP4	NEXT COARSE POSITION
00269	F97F	81	63	XFS8	CMP A	#63	AT MAX FINE COUNT YET?
00270	F981	27	35		BEQ	XFS11	YES
00271	F983	7C	0087	XFS9	INC	XFPOS+3	INCREMENT FINE POSITION
00272	F986	7D	009B		TST	XEFLAG	
00273	F989	27	33		BEQ	XFS12	0=NOT SET
00274	F98B	81	14		CMP A	#14	20CM FROM NEXT COARSE UPDATE
00275	F98D	2B	2F		BMI	XFS12	
00276	F98F	96	90		LDA A	XERR	
00277	F991	81	09		CMP A	#09	MAX ERRORS
00278	F993	27	03		BEQ	XF9A	
00279	F995	7C	0090		INC	XERR	INCREMENT ERROR COUNTER
00280	F998	7F	009B	XF9A	CLR	XEFLAG	CLEAR ERROR FLAG
00281	F99B	7C	0096		INC	TEMP4	NEXT COARSE VALUE
00282	F99E	20	1E		BRA	XFS12	
00283	F9A0	7D	009E	XFS10	TST	XFLAG	
00284	F9A3	26	DE		BNE	XFS9	STARTED AT ZERO DONT SET ERRO
00285	F9A5	7D	009A		TST	XIFLAG	TEST INTERRUPT FLAG
00286	F9A8	26	08		BNE	XFS10A	
00287	F9AA	73	009A		COM	XIFLAG	
00288	F9AD	7F	009B		CLR	XEFLAG	CLEAR ERROR FLAG
00289	F9B0	20	D1		BRA	XFS9	
00290	F9B2	86	FF	XFS10A	LDA A	#FF	

00291	F9B4	97	9B		STA A	XEFLAG	SET ERROR FLAG
00292	F9B6	20	CB		BRA	XFS9	
00293	F9B8	7C	0083	XFS11	INC	XCPOS+3	INC COARSE POS
00294	F9BB	7F	0087		CLR	XFPOS+3	FINE POS TO ZERO
00295	F9BE	CE	0080	XFS12	LDX	#XCPOS	
00296	F9C1	BD	FA94		JSR	CON	HEX TO DECIMAL
00297	F9C4	CE	0084		LDX	#XFPOS	
00298	F9C7	BD	FA94		JSR	CON	
00299	F9CA	39		XFSR	RTS		
00300				*			
00301				*/YFS/			
00302				* THIS	SUB.	SETS THE Y	AXIS FINE POSITION
00303	F9CB	7D	0095	YFS	TST	TEMP3	1st COARSE FIX
00304	F9CE	27	03		BEQ	YFS2	
00305	F9D0	7E	FA93	YFS1	JMP	YFSR	
00306	F9D3	96	98	YFS2	LDA A	DIR	
00307	F9D5	84	0C		AND A	#\$0C	MASK
00308	F9D7	27	F7		BEQ	YFS1	
00309	F9D9	81	04		CMP A	#\$04	LEFT?
00310	F9DB	27	54		BEQ	YFS7	YES
00311	F9DD	96	8F		LDA A	YFPOS+3	NO-TEST FINE POSITION
00312	F9DF	27	30		BEQ	YFS4	ZERO
00313	F9E1	7D	009F		TST	YFLAG	
00314	F9E4	27	0E		BEQ	YFS3	
00315	F9E6	81	5E		CMP A	#\$5E	FREE-PLAY
00316	F9E8	24	0A		BCC	YFS3	
00317	F9EA	81	06		CMP A	#\$06	
00318	F9EC	25	06		BCS	YFS3	
00319	F9EE	7F	009F		CLR	YFLAG	
00320	F9F1	7A	0097		DEC	TEMP5	
00321	F9F4	7A	008F	YFS3	DEC	YFPOS+3	DECREMENT FINE POSITION
00322	F9F7	7D	009D		TST	YEFLAG	TESR ERROR FLAG
00323	F9FA	27	13		BEQ	YF3B	NOT SET
00324	F9FC	81	50		CMP A	#\$50	(80)
00325	F9FE	2A	0F		BPL	YF3B	
00326	FA00	96	91		LDA A	YERR	
00327	FA02	81	09		CMP A	#\$09	9 ERRORS MAX
00328	FA04	27	03		BEQ	YF3A	
00329	FA06	7C	0091		INC	YERR	ERROR COUNTER
00330	FA09	7F	009D	YF3A	CLR	YEFLAG	CLEAR ERROR FLAG
00331	FA0C	7A	0097		DEC	TEMP5	NEXT VALUE
00332	FA0F	20	76	YF3B	BRA	YFS12	
00333	FA11	7D	009F	YFS4	TST	YFLAG	
00334	FA14	26	09		BNE	YFS5	STARTED AT ZERO DONT SET ERRO
00335	FA16	7D	009C		TST	YIFLAG	
00336	FA19	27	0D		BEQ	YFS6	
00337	FA1B	86	FF		LDA A	#\$FF	
00338	FA1D	97	9D		STA A	YEFLAG	SET ERROR FLAG
00339	FA1F	7A	008B	YFS5	DEC	YCPOS+3	DECREMENT COARSE POSITION
00340	FA22	86	63		LDA A	#\$63	(99)
00341	FA24	97	8F		STA A	YFPOS+3	
00342	FA26	20	5F		BRA	YFS12	
00343	FA28	86	FF	YFS6	LDA A	#\$FF	
00344	FA2A	97	9C		STA A	YIFLAG	RESET IRQ FLAG
00345	FA2C	7F	009D		CLR	YEFLAG	
00346	FA2F	20	EE		BRA	YFS5	
00347	FA31	96	8F	YFS7	LDA A	YFPOS+3	
00348	FA33	27	34		BEQ	YFS10	

00349	FA35	7D	009F	TST	YFLAG		
00350	FA38	27	0E	BEQ	YFS8		
00351	FA3A	81	5E	CMP A	#5E		
00352	FA3C	24	0A	BCC	YFS8		
00353	FA3E	81	06	CMP A	#06	FREE-PLAY	
00354	FA40	25	06	BCS	YFS8		
00355	FA42	7F	009F	CLR	YFLAG		
00356	FA45	7C	0097	INC	TEMP5		
00357	FA48	81	63	YFS8	CMP A	#63	MAX FINE VALUE
00358	FA4A	27	35	BEQ	YFS11	YES	
00359	FA4C	7C	008F	YFS9	INC	YFPOS+3	INCREMENT FINE POSITION
00360	FA4F	7D	009D	TST	YEFLAG	TEST ERROR FLAG	
00361	FA52	27	33	BEQ	YFS12	NOT SET	
00362	FA54	81	14	CMP A	#14		
00363	FA56	2B	2F	BMI	YFS12		
00364	FA58	96	91	LDA A	YERR		
00365	FA5A	81	09	CMP A	#09	MAX ERRORS	
00366	FA5C	27	03	BEQ	YF9A		
00367	FA5E	7C	0091	INC	YERR	INC. ERROR COUNT	
00368	FA61	7F	009D	YF9A	CLR	YEFLAG	CLEAR ERROR FLAG
00369	FA64	7C	0097	INC	TEMP5	NEXT VALUE	
00370	FA67	20	1E	BRA	YFS12		
00371	FA69	7D	009F	YFS10	TST	YFLAG	
00372	FA6C	26	DE	BNE	YFS9	IF STARTED AT ZERO DO NOT SET	
00373	FA6E	7D	009C	TST	YIFLAG	TEST IRQ FLAG	
00374	FA71	26	08	BNE	YFS10A		
00375	FA73	73	009C	COM	YIFLAG		
00376	FA76	7F	009D	CLR	YEFLAG		
00377	FA79	20	D1	BRA	YFS9		
00378	FA7B	86	FF	YFS10A	LDA A	#FF	
00379	FA7D	97	9D	STA A	YEFLAG		
00380	FA7F	20	CB	BRA	YFS9		
00381	FA81	7F	008F	YFS11	CLR	YFPOS+3	ZERO FINE POSITION
00382	FA84	7C	008B	INC	YCPOS+3		
00383	FA87	CE	0088	YFS12	LDX	#YCPOS	
00384	FA8A	BD	FA94	JSR	CON	HEX TO DECIMAL	
00385	FA8D	CE	008C	LDX	#YFPOS		
00386	FA90	BD	FA94	JSR	CON		
00387	FA93	39		YFSR	RTS		
00388				*			
00389				*/CON/			
00390				* THIS	SUB. CONVERTS	HEX TO DECIMAL	
00391	FA94	86	00	CON	LDA A	#00	CONVERT TO DECIMAL
00392	FA96	A7	00		STA A	0,X	
00393	FA98	A7	01		STA A	1,X	
00394	FA9A	A7	02		STA A	2,X	
00395	FA9C	A6	03		LDA A	3,X	
00396	FA9E	C6	0A		LDA B	#0A	(10)
00397	FAA0	10		TEN	SBA		
00398	FAA1	81	00		CMP A	#00	
00399	FAA3	2B	04		BMI	UN	
00400	FAA5	6C	01		INC	1,X	TENS
00401	FAA7	20	F7		BRA	TEN	
00402	FAA9	1B		UN	ABA		
00403	FAAA	A7	02		STA A	2,X	UNITS
00404	FAAC	A6	01		LDA A	1,X	
00405	FAAE	10			SBA		
00406	FAAF	2B	04		BMI	CON1	


```

00407 FAB1 A7 01          STA A 1,X
00408 FAB3 6C 00          INC 0,X
00409 FAB5 39          CON1 RTS
00410                  *
00411                  */XCS/
00412                  * THIS SUB. ENTERS THE X AXIS COARSE POSITION
00413 FAB6 B6 1401 XCS     LDA A CR2      READ PTM STATUS REG
00414 FAB9 84 01          AND A #$01     TEST FOR TO
00415 FABB 27 4E          BEQ XCSR        RETURN IF NOT SET
00416 FABD D6 98          LDA B DIR      DIRECTION
00417 FABF C4 30          AND B #$30     MASK
00418 FAC1 27 48          BEQ XCSR        (SPURIOUS TRIG)
00419 FAC3 B6 1800        LDA A PIAD1    POSITION FROM LIGHT FORK
00420 FAC6 84 1F          AND A #$1F     MASK
00421 FAC8 27 41          BEQ XCSR        SPURIOUS TRIG
00422 FACA 81 1F          CMP A #$1F     (SPURIOUS TRIG)
00423 FACC 27 3D          BEQ XCSR
00424 FACE 7D 0094        TST TEMP2      1st COARSE FIX FLAG
00425 FAD1 26 39          BNE XCS4
00426 FAD3 91 96          CMP A TEMP4     COMPARE WITH COMPUTED VALUE
00427 FAD5 27 0B          BEQ XCS1        EQUAL
00428 FAD7 96 90          LDA A XERR
00429 FAD9 81 09          CMP A #$09      9 ERRORS MAX
00430 FADB 27 03          BEQ XCSA
00431 FADD 7C 0090        INC XERR        INCREMENT ERROR COUNTER
00432 FAE0 96 96          XCSA LDA A TEMP4
00433 FAE2 97 83          XCS1 STA A XCPOS+3  STORE COMPUTED VALUE
00434 FAE4 7F 0087        CLR XFPOS+3
00435 FAE7 7F 009E        CLR XFLAG      CLEAR FREE PLAY FLAG
00436 FAEA 7F 009A        CLR XIFLAG    CLEAR IRQ FLAG
00437 FAED C1 10          CMP B #$10     TEST DIRECTION
00438 FAEF 27 05          BEQ XCS2
00439 FAF1 7A 0096        DEC TEMP4      NEXT VALUE
00440 FAF4 20 03          BRA XCS3
00441 FAF6 7C 0096 XCS2   INC TEMP4      NEXT VALUE
00442 FAF9 CE 0B00 XCS3   LDX #$0B00
00443 FAFC FF 1402        STX T1M        X DELAY
00444 FAFF CE 0080        LDX #XCPOS
00445 FB02 BD FA94        JSR CON        HEX TO DES.
00446 FB05 CE 0084        LDX #XFPOS
00447 FB08 BD FA94        JSR CON
00448 FB0B 39          XCSR RTS
00449 FB0C 97 83          XCS4 STA A XCPOS+3
00450 FB0E 7F 0087        CLR XFPOS+3
00451 FB11 7F 0094        CLR TEMP2      1st FIX FLAG
00452 FB14 20 E3          BRA XCS3
00453                  *
00454                  */YCS/
00455                  * THIS SUB. ENTERS THE Y AXIS COARSE POSITION
00456 FB16 B6 1401 YCS     LDA A CR2
00457 FB19 84 04          AND A #$04     TEST FOR TO
00458 FB1B 27 4E          BEQ YCSR        RETURN IF NOT SET
00459 FB1D D6 98          LDA B DIR
00460 FB1F C4 0C          AND B #$0C     MASK
00461 FB21 27 48          BEQ YCSR        (SPURIOUS TRIG)
00462 FB23 B6 1802        LDA A PIAD2    LIGHT FORK
00463 FB26 84 7F          AND A #$7F     MASK
00464 FB28 27 41          BEQ YCSR        SPURIOUS TRIG

```


PAGE 009 BC1

00465	FB2A	81	7F		CMP A	#\$7F	
00466	FB2C	27	3D		BEQ	YCSR	(SPURIOUS TRIG)
00467	FB2E	7D	0095		TST	TEMP3	1st COARSE FIX
00468	FB31	26	39		BNE	YCS4	
00469	FB33	91	97		CMP A	TEMP5	COMPARE WITH COMPUTED VALUE
00470	FB35	27	0B		BEQ	YCS1	
00471	FB37	96	91		LDA A	YERR	
00472	FB39	81	09		CMP A	#\$09	9 ERRORS MAX
00473	FB3B	27	03		BEQ	YCSA	
00474	FB3D	7C	0091		INC	YERR	ERROR COUNTER
00475	FB40	96	97	YCSA	LDA A	TEMP5	
00476	FB42	97	8B	YCS1	STA A	YCPOS+3	LOAD COMPUTED VALUE
00477	FB44	7F	008F		CLR	YFPOS+3	
00478	FB47	7F	009F		CLR	YFLAG	
00479	FB4A	7F	009C		CLR	YIFLAG	IRQ FLAG
00480	FB4D	C1	04		CMP B	#\$04	
00481	FB4F	27	05		BEQ	YCS2	
00482	FB51	7A	0097		DEC	TEMP5	NEXT VALUE
00483	FB54	20	03		BRA	YCS3	
00484	FB56	7C	0097	YCS2	INC	TEMP5	NEXT VALUE
00485	FB59	CE	0B00	YCS3	LDX	#\$0B00	
00486	FB5C	FF	1406		STX	T3M	Y DELAY
00487	FB5F	CE	0088		LDX	#YCPOS	
00488	FB62	BD	FA94		JSR	CON	CONVERT TO DECIMAL
00489	FB65	CE	008C		LDX	#YFPOS	
00490	FB68	BD	FA94		JSR	CON	
00491	FB6B	39		YCSR	RTS		
00492	FB6C	97	8B	YCS4	STA A	YCPOS+3	STORE VALUE FROM SENSOR
00493	FB6E	7F	008F		CLR	YFPOS+3	
00494	FB71	7F	0095		CLR	TEMP3	
00495	FB74	20	E3		BRA	YCS3	
00496				*			
00497				* DELAY ROUTINE			
00498	FB76	36		DELAY	PSH A		
00499	FB77	37			PSH B		
00500	FB78	C6	FF		LDA B	#\$FF	
00501	FB7A	86	FF	D2	LDA A	#\$FF	
00502	FB7C	4A		D1	DEC A		
00503	FB7D	81	00		CMP A	#\$00	
00504	FB7F	26	FB		BNE	D1	
00505	FB81	5A			DEC B		
00506	FB82	C1	00		CMP B	#\$00	
00507	FB84	26	F4		BNE	D2	
00508	FB86	32			PUL A		
00509	FB87	33			PUL B		
00510	FB88	39			RTS		
00511				*			
00512				*/PUT/			
00513				*THIS SUB COMPUTES THE			NEXT COARSE VALUE
00514	FB89	7F	0090	PUTX	CLR	XERR	X ERROR COUNTER
00515	FB8C	7F	009B		CLR	XEFLAG	X ERROR FLAG
00516	FB8F	86	FF		LDA A	#\$FF	
00517	FB91	97	9A		STA A	XIFLAG	IRQ FLAG
00518	FB93	D6	83		LDA B	XCPOS+3	
00519	FB95	96	98		LDA A	DIR	
00520	FB97	84	30		AND A	#\$30	MASK
00521	FB99	81	10		CMP A	#\$10	FORWARD
00522	FB9B	27	19		BEQ	P1	

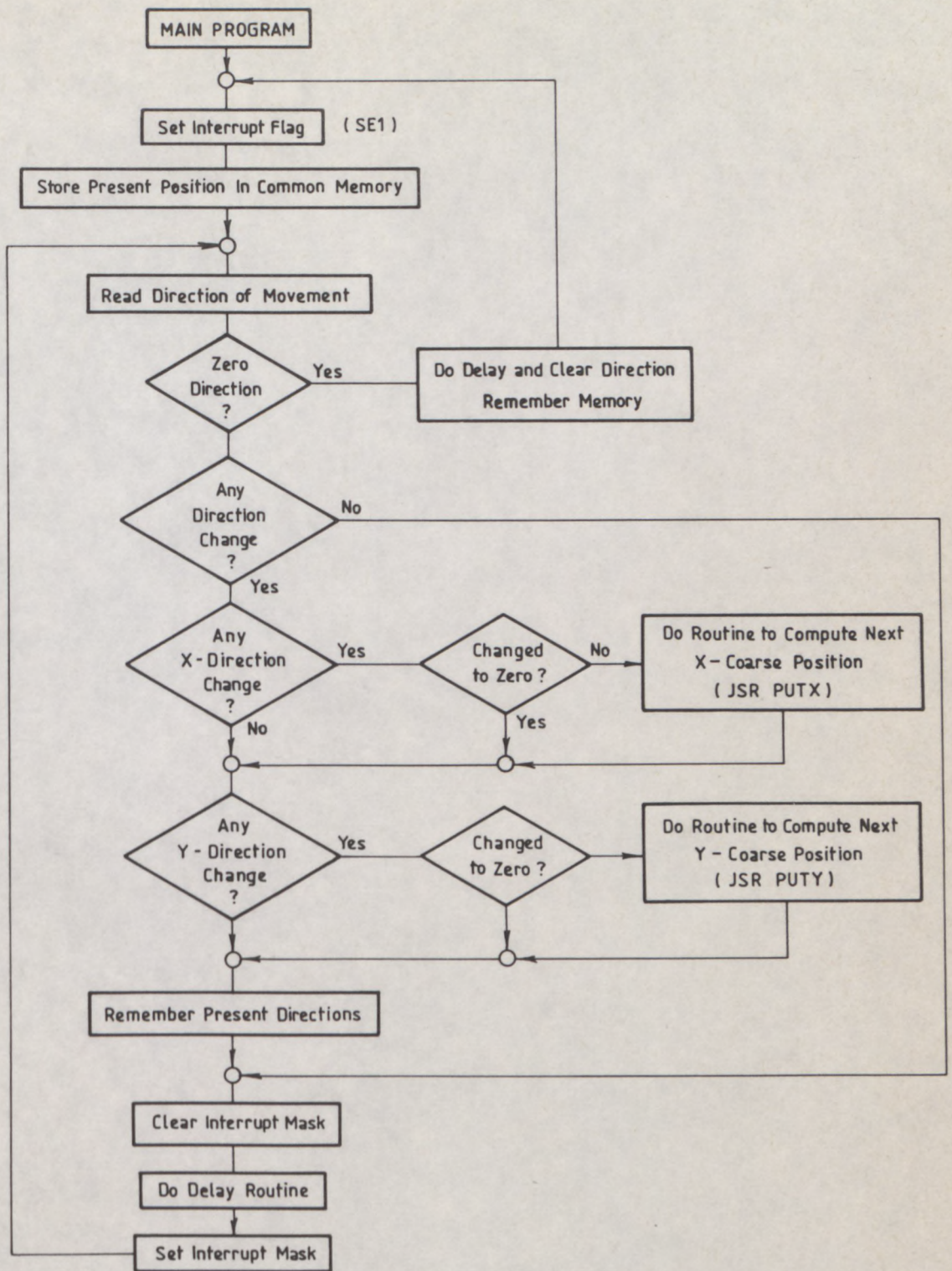
00523	FB9D	20	2D		BRA	P2	REVERSE
00524	FB9F	7F	0091	PUTY	CLR	YERR	CLEAR ASS. FLAGS
00525	FBA2	7F	009D		CLR	YEFLAG	
00526	FBA5	86	FF		LDA A	#\$FF	
00527	FBA7	97	9C		STA A	YIFLAG	
00528	FBA9	D6	8B		LDA B	YCPOS+3	
00529	FBAB	96	98		LDA A	DIR	
00530	FBAD	84	0C		AND A	#\$0C	MASK
00531	FBAF	81	04		CMP A	#\$04	LEFT
00532	FBB1	27	2C		BEQ	P3	
00533	FBB3	20	40		BRA	P4	RIGHT
00534	FBB5	39		P6	RTS		
00535	FBB6	86	FF	P1	LDA A	#\$FF	
00536	FBB8	97	9E		STA A	XFLAG	SET 5CM FREE PLAY FLAG
00537	FBBA	96	87		LDA A	XFPOS+3	TEST FINE POSITION
00538	FBBC	81	06		CMP A	#\$06	BETWEEN 0 - 5
00539	FBBE	25	08		BCS	P1A	YES
00540	FBC0	81	5E		CMP A	#\$5E	BETWEEN 95 - 99
00541	FBC2	24	03		BCC	P1B	YES
00542	FBC4	7F	009E		CLR	XFLAG	CLEAR FREE PLAY FLAG
00543	FBC7	5C		P1B	INC B		INCREMENT PRESENT POSITION
00544	FBC8	D7	96	P1A	STA B	TEMP4	REMEMBER POSITION
00545	FBCA	20	E9		BRA	P6	
00546	FBCB	96	87	P2	LDA A	XFPOS+3	TEST POSITION
00547	FBCE	81	06		CMP A	#\$06	BETWEEN 1 - 5 FROM COARSE UPD
00548	FBD0	25	05		BCS	P2A	YES
00549	FBD2	81	5E		CMP A	#\$5E	BETWEEN 95 - 99
00550	FBD4	25	05		BCS	P2B	NO
00551	FBD6	5C			INC B		INCREMENT POSITION
00552	FBD7	86	FF	P2A	LDA A	#\$FF	
00553	FBD9	97	9E		STA A	XFLAG	SET FREE PLAY FLAG
00554	FBDB	D7	96	P2B	STA B	TEMP4	REM. POS.
00555	FBDD	20	D6		BRA	P6	
00556	FBD F	86	FF	P3	LDA A	#\$FF	
00557	FBE1	97	9F		STA A	YFLAG	SET 5 CM PLAY FLAG
00558	FBE3	96	8F		LDA A	YFPOS+3	CHECK Y FINE POSITION
00559	FBE5	81	06		CMP A	#\$06	5cm IRQ FREE-PLAY
00560	FBE7	25	08		BCS	P3A	
00561	FBE9	81	5E		CMP A	#\$5E	
00562	FBEB	24	03		BCC	P3B	
00563	FBED	7F	009F		CLR	YFLAG	CLEAR FREE PLAY FLAG
00564	FBF0	5C		P3B	INC B		
00565	FBF1	D7	97	P3A	STA B	TEMP5	STORE PRESENT VALUE
00566	FBF3	20	C0		BRA	P6	
00567	FBF5	96	8F	P4	LDA A	YFPOS+3	
00568	FBF7	81	06		CMP A	#\$06	0-5
00569	FBF9	25	05		BCS	P4A	
00570	FBFB	81	5E		CMP A	#\$5E	95-99
00571	FBFD	25	05		BCS	P4B	
00572	FBFF	5C			INC B		
00573	FC00	86	FF	P4A	LDA A	#\$FF	
00574	FC02	97	9F		STA A	YFLAG	SET FREE PLAY FLAG
00575	FC04	D7	97	P4B	STA B	TEMP5	
00576	FC06	20	AD		BRA	P6	
00577				*			
00578				*			
00579				* INTERRUPTS			
00580				* -----			


```

00581          *
00582 FFB0          ORG      $FFB0
00583          *
00584 FFB0 B6 1803 IRQ    LDA A   PIAC2      ANY COARSE TRIGGER
00585 FFB3 84 80          AND A   #$80
00586 FFB5 27 18          BEQ      IR1        NO COARSE TRIG
00587 FFB7 B6 1800          LDA A   PIAD1      X OR Y
00588 FFBA 84 C0          AND A   #$C0        MASK LOWER BITS
00589 FFBC 81 80          CMP A   #$80
00590 FFBE 25 05          BCS      IR2
00591 FFC0 36          PSH A
00592 FFC1 BD FAB6          JSR      XCS        REMEMBER A REG
00593 FFC4 32          PUL A        SET X COARSE POSITION
00594 FFC5 84 40          AND A   #$40        RESTORE A REG
00595 FFC7 27 03          BEQ      IR4
00596 FFC9 BD FB16          JSR      YCS        SET Y COARSE POSITION
00597 FFCC B6 1802 IR4    LDA A   PIAD2      RESET IRQ FLAG
00598 FFCF B6 1801 IR1    LDA A   PIAC1      ANY FINE TRIG
00599 FFD2 84 C0          AND A   #$C0
00600 FFD4 27 15          BEQ      IR6
00601 FFD6 81 80          CMP A   #$80
00602 FFD8 25 05          BCS      IR3
00603 FFDA 36          PSH A
00604 FFDB BD F902          JSR      XFS        REMEMBER A ACC
00605 FFDE 32          PUL A        SET X FINE POSITION
00606 FFDF 84 40          AND A   #$40
00607 FFE1 27 03          BEQ      IR5
00608 FFE3 BD F9CB          JSR      YFS        SET Y FINE POSITION
00609 FFE6 B6 1800 IR5    LDA A   PIAD1      RESET IRQ FLAG
00610 FFE9 20 C5          BRA      IRQ
00611 FFEB 8E 00FF IR6    LDS      #STACK    RELOAD STACK POINTER
00612 FFEE 7E F876          JMP      BEGIN    JUMP TO MAIN LOOP
00613          *
00614          *
00615          * INTERRUPT VECTORS
00616          *
00617          *
00618 FFF8          ORG      $FFF8
00619 FFF8 FF          FCB      $FF,$B0,$00,$00,$00,$00,$00,$F8,$00

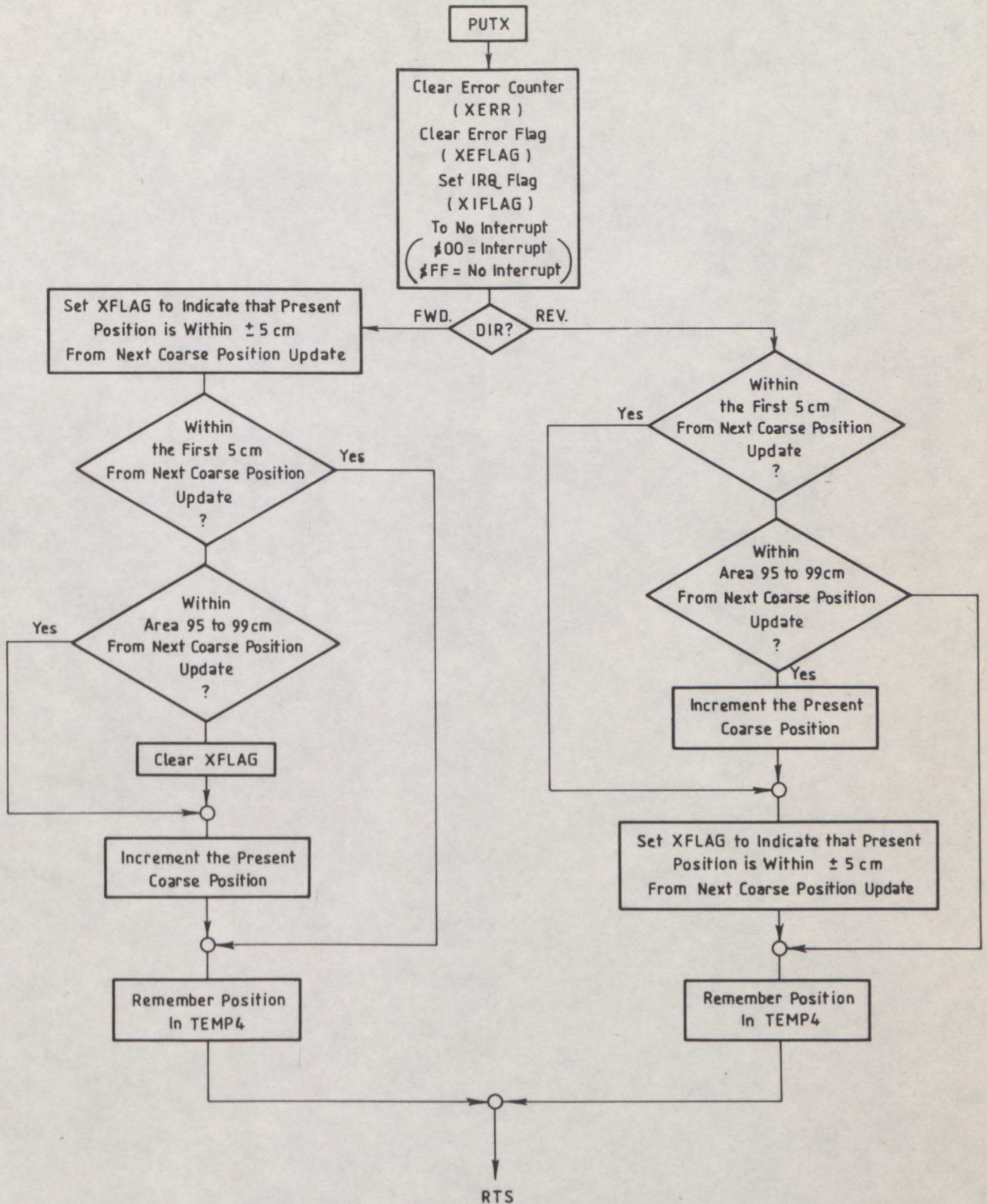
```

TOTAL ERRORS 00000



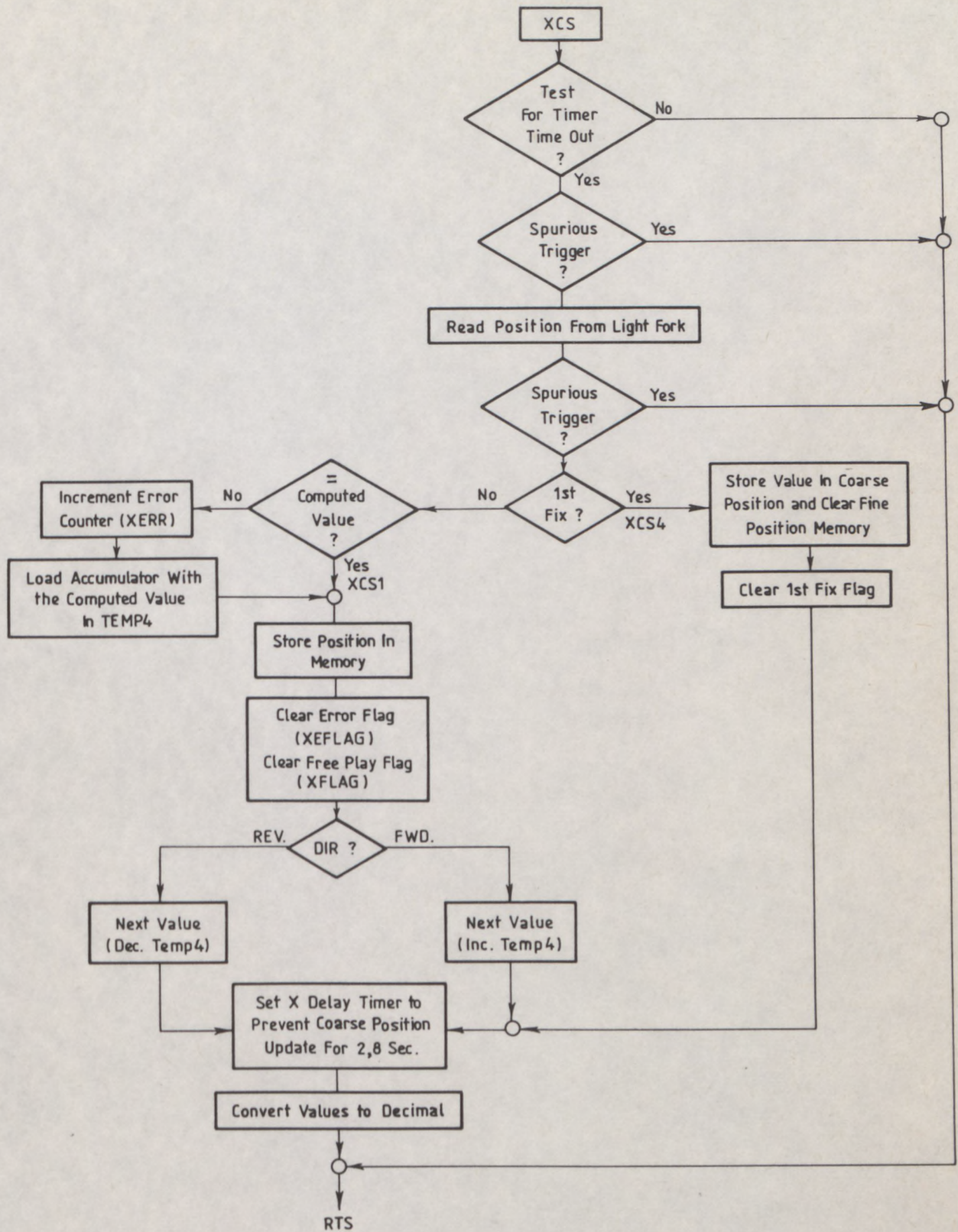
POSITION FIXING MODULE
CONTROL LOOP

Fig. F.1



COMPUTING THE NEXT COARSE POSITION

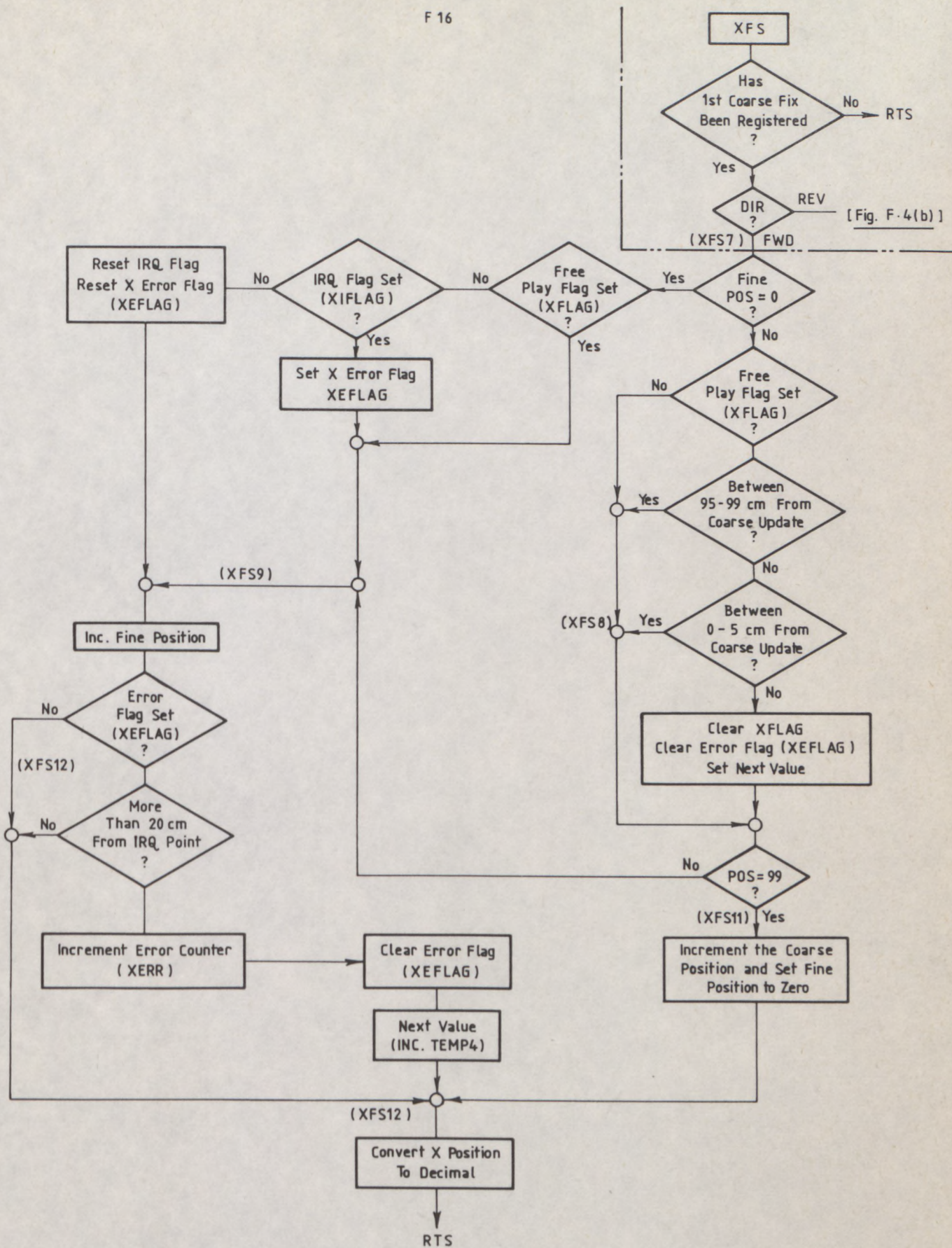
Fig. F.2



COARSE POSITION FIXING

(Y Axis similar to the X Axis routine)

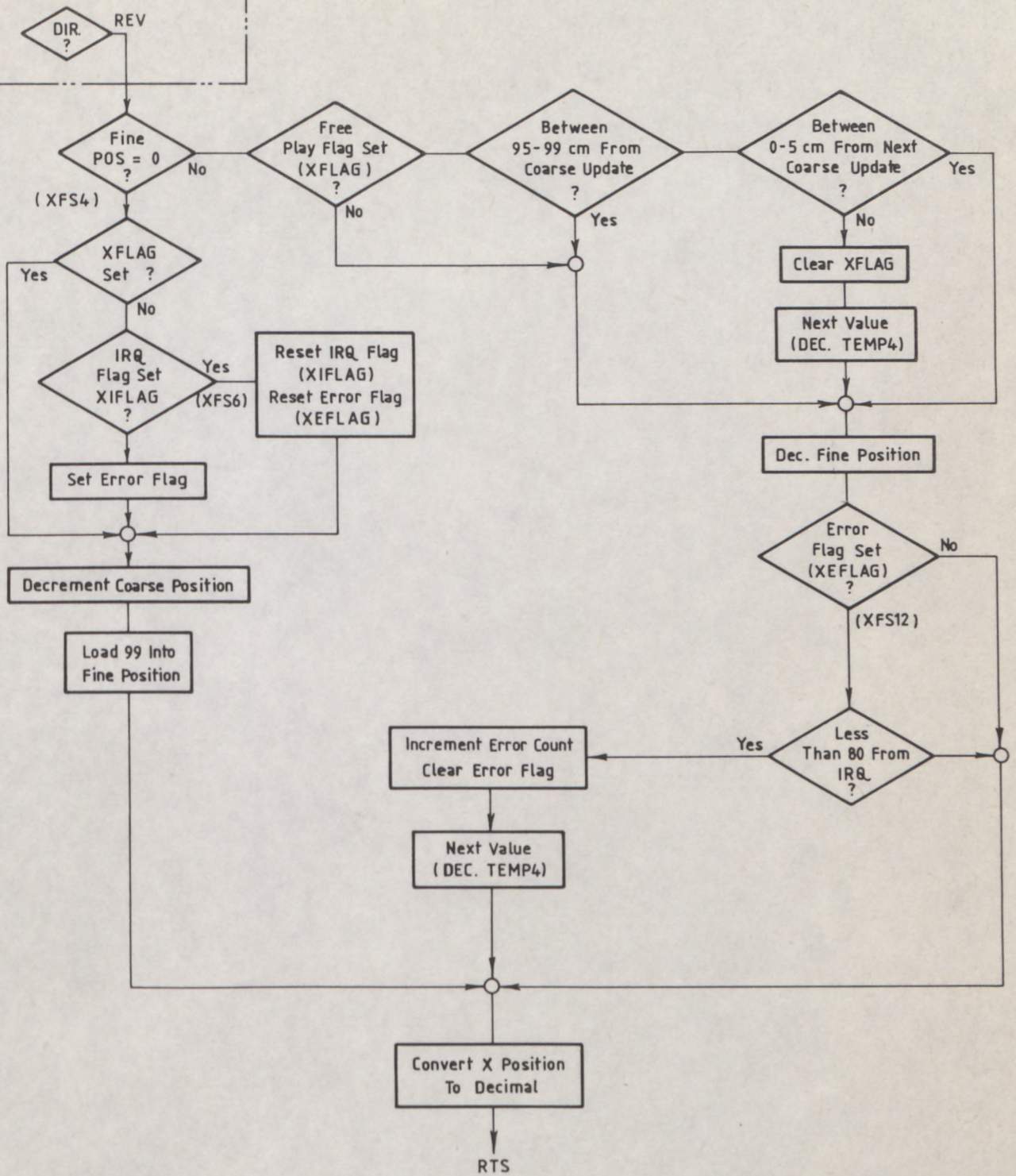
Fig. F.3



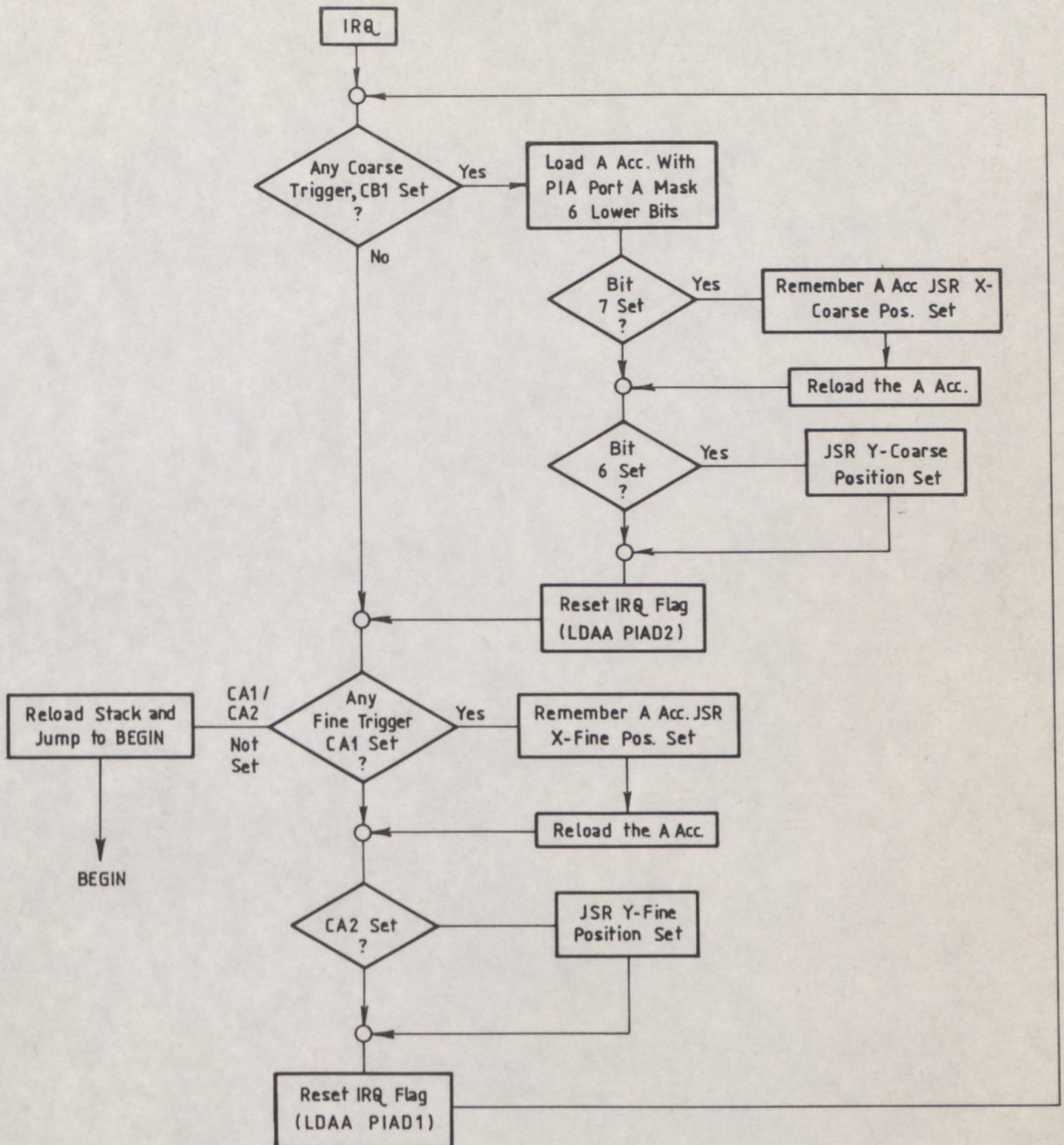
FORWARD FINE POSITION FIXING

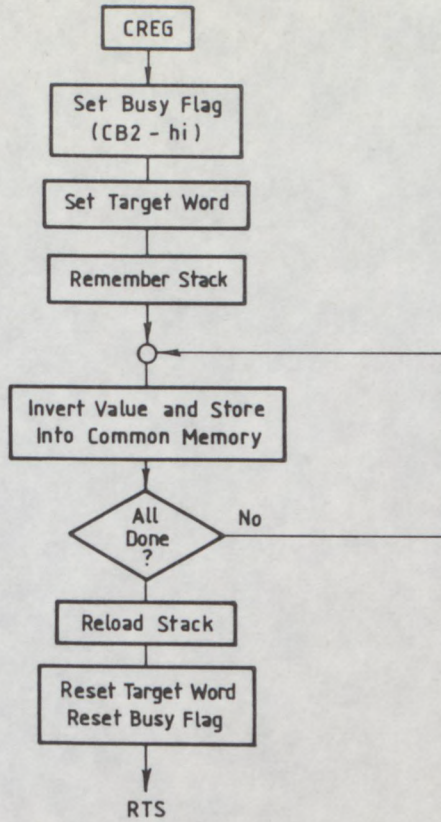
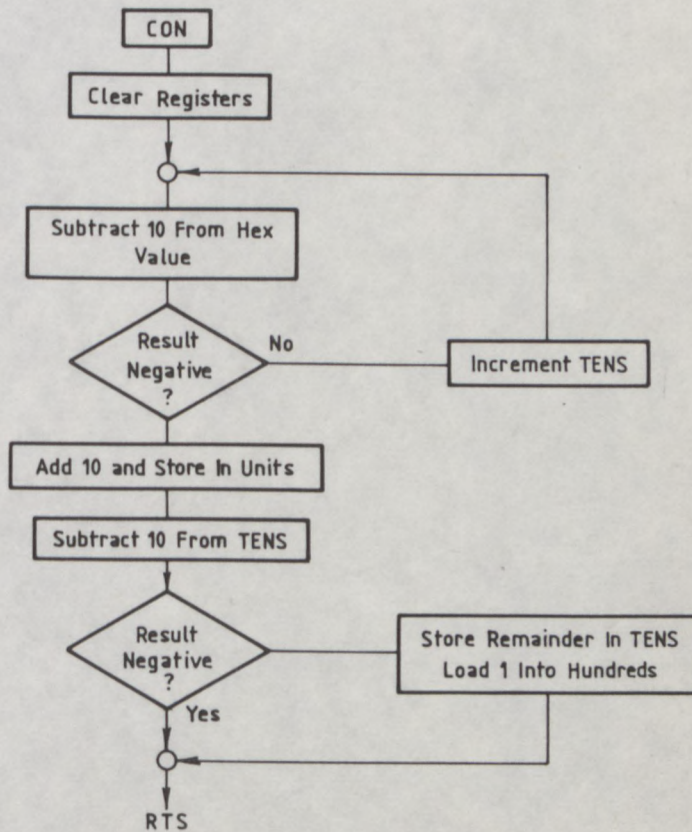
Fig. F.4 (a)

(Y Axis similar to X Axis)

**REVERSE FINE POSITION FIXING****Fig. F.4 (b)**

(Y Axis similar to the X Axis)

INTERRUPT ROUTINEFig. F.5

STORE IN COMMON MEMORYFig. F.6CONVERT TO DECIMALFig. F.7

APPENDIX GMK2 SPEED CONTROLLER PROGRAM LISTING

The following is the listing of the MK2 speed controller program (for convenience the speed table is not listed). The speed is controlled with the aid of a one-board microcomputer, see Section 6.5.

Flow diagrams

Also included are flow diagrams of the MK2 speed controller.


```

00001          NAM      MSPEED
00002          *
00003          *  MOTOR SPEED CONTROL
00004          *
00005          *
00006          *  SYSTEM EQUATES
00007          *
00008      1400      CR13      EQU      $1400      PTM
00009      1401      CR2       EQU      $1401
00010      1402      T1M       EQU      $1402
00011      1403      T1L       EQU      $1403
00012      1404      T2M       EQU      $1404
00013      1405      T2L       EQU      $1405
00014      1406      T3M       EQU      $1406
00015      1407      T3L       EQU      $1407
00016          *
00017      1800      PIAD1      EQU      $1800      PIA
00018      1801      PIAC1      EQU      $1801
00019      1802      PIAD2      EQU      $1802
00020      1803      PIAC2      EQU      $1803
00021      00FF      STACK     EQU      $FF
00022          *
00023      003F      TARG1      EQU      $003F      TARGET WORD
00024      003E      XSPFLG     EQU      $003E      REST/OPERATE FLAG
00025          *
00026          *  SCRATCH PAD
00027          *
00028      0020          ORG      $20
00029      0020      0001      SPDHI      RMB      1      SPEED HI BYTE
00030      0021      0001      SPDLO      RMB      1      SPEED LO BYTE
00031      0022      0001      DIR2       RMB      1      DIRECTION FLAG
00032          *
00033      0080          ORG      $80
00034      0080      0001      T1         RMB      1      SPEED REMEMBER
00035      0081      0001      T2         RMB      1      SPEED REMEMBER
00036      0082      0001      T3         RMB      1      SPEED RELOAD COUNTER
00037      0083      0001      T4         RMB      1      PERIOD AVERAGING COUNTER
00038      0084      0001      T5         RMB      1      INC/DEC TABLE REM.(RELOAD)
00039      0085      0001      T6         RMB      1      RELOAD TIMES
00040      0086      0002      REG1        RMB      2      TABLE POINTER
00041      0088      0002      REG2        RMB      2      AVERAGE VALUE
00042      008A      0002      REG3        RMB      2      SUM
00043      008C      0001      DIR         RMB      1      DIRECTION FOR RELAY
00044      008D      0001      SPFLG       RMB      1      REST/OPERATE FLAG
00045      008E      0001      SPLO        RMB      1
00046      008F      0001      SPHI        RMB      1
00047      0090      0001      FORK        RMB      1      LIGHT FORK FLAG
00048          *
00049          *      TH1.....PERIOD COUNTER
00050          *      TH2.....CONDUCTING ANGLE CONTROL
00051          *      TH3.....HI-FREQUENCY(SCR GATE TRIGGER)
00052          *
00053      F800          ORG      $F800
00054      F800      8E      00FF      START      LDS      #STACK
00055      F803      0F          SEI
00056          *
00057      F804      86      00          LDA      A      #$00      SETUP PTM
00058      F806      B7      1401          STA      A      CR2      SELECT CR3

```


00059	F809	C6	82		LDA B	#82	
00060	F80B	F7	1400		STA B	CR13	TH3 MODE
00061	F80E	C6	83		LDA B	#83	
00062	F810	F7	1401		STA B	CR2	TH2 MODE AND SELECT CR1
00063	F813	C6	01		LDA B	#01	
00064	F815	F7	1400		STA B	CR13	TH1 MODE TIMERS PRESET
00065	F818	CE	0031		LDX	#0031	49
00066	F81B	FF	1406		STX	T3M	
00067	F81E	CE	FFFF		LDX	#FFFF	
00068	F821	FF	1402		STX	T1M	
00069				*			
00070	F824	B7	1801		STA A	PIAC1	SETUP PIA'S
00071	F827	B7	1803		STA A	PIAC2	SELECT DDR'S
00072	F82A		43		COM A		
00073	F82B	B7	1802		STA A	PIAD2	OUTPUT PORT
00074	F82E	86	F0		LDA A	#F0	
00075	F830	B7	1800		STA A	PIAD1	LSB INPUT/MSB OUTPUT
00076	F833	86	3E		LDA A	#3E	CA2-HI
00077	F835	B7	1801		STA A	PIAC1	PIA1 MODE
00078	F838	86	34		LDA A	#34	
00079	F83A	B7	1803		STA A	PIAC2	PIA2 MODE
00080				*			
00081	F83D		4F		CLR A		CLEAR MEMORY
00082	F83E		5F		CLR B		
00083	F83F	CE	0080		LDX	#T1	
00084	F842	A7	00	CL	STA A	0,X	
00085	F844		08		INX		
00086	F845		5C		INC B		
00087	F846	C1	11		CMP B	#11	
00088	F848	26	F8		BNE	CL	
00089	F84A	CE	FC00		LDX	#TABLE	
00090	F84D	DF	86		STX	REG1	
00091	F84F	7F	0020		CLR	SPDHI	
00092	F852	7F	0021		CLR	SPDLO	
00093	F855		43		COM A		
00094	F856	97	22		STA A	DIR2	
00095				*			
00096				*	MAIN PROGRAM		
00097				*			
00098	F858	BD	F8FC	BEGIN	JSR	CREG	MOVEMENT?
00099	F85B	96	8C		LDA A	DIR	TEST DIRECTION
00100	F85D	26	08		BNE	MP7	
00101	F85F	BD	F93B		JSR	RELAY	NONE
00102	F862	BD	F958		JSR	DELAY	
00103	F865	20	F1		BRA	BEGIN	
00104				*			
00105	F867	96	8E	MP7	LDA A	SPLO	GET REQUIRED SPEED
00106	F869	26	04		BNE	MP2	
00107	F86B	D6	8F		LDA B	SPHI	MSB
00108	F86D	27	4C		BEQ	MP1	ZERO SPEED
00109	F86F	D6	8F	MP2	LDA B	SPHI	LOAD HI SPEED BYTE
00110	F871	90	80		SUB A	T1	
00111	F873	D2	81		SBC B	T2	
00112	F875	2D	14		BLT	MP4	
00113	F877	22	02		BHI	MP6	
00114	F879	20	DD		BRA	BEGIN	NO SPEED CHANGE
00115	F87B	7D	0080	MP6	TST	T1	IF MOTOR WAS AT 0 SPEED INCRE
00116	F87E	26	05		BNE	MP5	

00117	F880	7D	0081		TST	T2	
00118	F883	27	06		BEQ	MP4	
00119	F885	86	00	MP5	LDA A	#\$00	
00120	F887	97	84		STA A	T5	DEC SPEED
00121	F889	20	04		BRA	MP3	
00122	F88B	86	FF	MP4	LDA A	#\$FF	
00123	F88D	97	84		STA A	T5	INC SPEED
00124	F88F	0F		MP3	SEI		
00125	F890	96	8E		LDA A	SPLO	
00126	F892	D6	8F		LDA B	SPHI	
00127	F894	D7	81		STA B	T2	REMEMBER LAST SPEED
00128	F896	97	80		STA A	T1	
00129	F898	86	01		LDA A	#\$01	
00130	F89A	97	85		STA A	T6	RELOAD SPEED 1 TIME
00131	F89C	7D	008D		TST	SPFLG	FLAG SET TO OPERATE?
00132	F89F	26	17		BNE	MP8	YES
00133	F8A1	86	FF		LDA A	#\$FF	
00134	F8A3	97	8D		STA A	SPFLG	SET FLAG TO OPERATE
00135	F8A5	97	89		STA A	REG2+1	SET AVE PERIOD TO MAX
00136	F8A7	97	88		STA A	REG2	
00137	F8A9	7F	0090		CLR	FORK	L/FORK OPERATING FLAG
00138	F8AC	BD	F93B		JSR	RELAY	SET RELAY
00139	F8AF	86	02		LDA A	#\$02	
00140	F8B1	B7	1400		STA A	CR13	TIMERS OPERATE
00141	F8B4	5F			CLR B		
00142	F8B5	F7	1800		STA B	PIAD1	CLEAR ERROR SENSE PORT
00143	F8B8	0E		MP8	CLI		
00144	F8B9	20	9D		BRA	BEGIN	
00145	F8BB	C6	FF	MP1	LDA B	#\$FF	
00146	F8BD	F7	1800		STA B	PIAD1	SET ERROR SENSE PORT
00147	F8C0	86	01		LDA A	#\$01	
00148	F8C2	97	85		STA A	T6	RELOAD SPEED 1 TIME ONLY
00149	F8C4	7F	0084		CLR	T5	INC/DEC FLAG
00150	F8C7	D6	87		LDA B	REG1+1	TEST FOR TABLE START
00151	F8C9	C1	00		CMP B	#\$00	
00152	F8CB	22	8B		BHI	BEGIN	NO
00153	F8CD	D6	86		LDA B	REG1	MSB
00154	F8CF	C1	FC		CMP B	#\$FC	
00155	F8D1	22	85		BHI	BEGIN	
00156	F8D3	0F			SEI		
00157	F8D4	86	3E		LDA A	#\$3E	
00158	F8D6	B7	1801		STA A	PIAC1	WATCHDOG TIMER
00159	F8D9	86	01		LDA A	#\$01	
00160	F8DB	B7	1400		STA A	CR13	TIMERS PRESET
00161	F8DE	4F			CLR A		
00162	F8DF	97	80		STA A	T1	RESET FLAGS
00163	F8E1	97	81		STA A	T2	
00164	F8E3	97	8D		STA A	SPFLG	REST STATE
00165	F8E5	97	83		STA A	T4	
00166	F8E7	97	88		STA A	REG2	
00167	F8E9	97	89		STA A	REG2+1	
00168	F8EB	97	8A		STA A	REG3	
00169	F8ED	97	8B		STA A	REG3+1	
00170	F8EF	BD	F8FC	MP9	JSR	CREG	INFORM BC2 THAT AT ZERO SP.
00171	F8F2	BD	F93B		JSR	RELAY	RESET RELAYS
00172	F8F5	96	8C		LDA A	DIR	
00173	F8F7	26	F6		BNE	MP9	
00174	F8F9	7E	F858		JMP	BEGIN	


```

00175      *
00176      *      SUBROUTINES
00177      *
00178      */CREG/
00179      *THIS SUB TRANSFERES TO AND FROM DEDICATED
00180      *MEMORY TO INTERMEDIATE MEMORY
00181      F8FC 86 3C      CREG      LDA A    #$3C
00182      F8FE B7 1803      STA A    PIAC2      SET BUSY FLAG
00183      F901 86 55      LDA A    #$55
00184      F903 97 3F      STA A    TARG1      CHANGE TARGET WORD
00185      F905 96 21      LDA A    SPDLO
00186      F907 43      COM A
00187      F908 97 8E      STA A    SPLO
00188      F90A 96 20      LDA A    SPDHI
00189      F90C 43      COM A
00190      F90D 97 8F      STA A    SPHI
00191      F90F 96 22      LDA A    DIR2
00192      F911 43      COM A      INVERTED BUS
00193      F912 97 8C      STA A    DIR
00194      F914 96 8D      LDA A    SPFLG
00195      F916 43      COM A
00196      F917 97 3E      STA A    XSPFLG
00197      F919 86 AA      LDA A    #$AA
00198      F91B 97 3F      STA A    TARG1      RESET TARGET WORD
00199      F91D 86 34      LDA A    #$34
00200      F91F B7 1803      STA A    PIAC2      RESET BUSY FLAG
00201      F922 39      RTS
00202      *
00203      */WDOG/
00204      *WATCHDOG PULSE FOR MICRO RUN-AWAY
00205      F923 7D 008D      WDOG      TST      SPFLG      REST OR OPERATE?
00206      F926 27 06      BEQ      WD1
00207      F928 86 37      LDA A    #$37      OPERATE MODE
00208      F92A C6 3F      LDA B    #$3F
00209      F92C 20 04      BRA      WD2
00210      F92E 86 36      WD1      LDA A    #$36      REST MODE
00211      F930 C6 3E      LDA B    #$3E
00212      F932 B7 1801      WD2      STA A    PIAC1      CA2 NEGATIVE
00213      F935 08      INX
00214      F936 09      DEX
00215      F937 F7 1801      STA B    PIAC1      CA2 POSITIVE
00216      F93A 39      RTS
00217      *
00218      */RELAY/
00219      *THIS SUB SELECTS AND ACTIVATES THE
00220      *DIRECTION RELAYS
00221      F93B 96 8C      RELAY      LDA A    DIR
00222      F93D 81 11      CMP A    #$11      FORWARD
00223      F93F 27 0B      BEQ      RE2
00224      F941 81 22      CMP A    #$22
00225      F943 26 0E      BNE      RE1
00226      F945 86 02      LDA A    #$02
00227      F947 B7 1802      STA A    PIAD2      REVERSE RELAY
00228      F94A 20 0B      BRA      RE3
00229      F94C 86 01      RE2      LDA A    #$01
00230      F94E B7 1802      STA A    PIAD2      FORWARD RELAY
00231      F951 20 04      BRA      RE3
00232      F953 4F      RE1      CLR A      REST

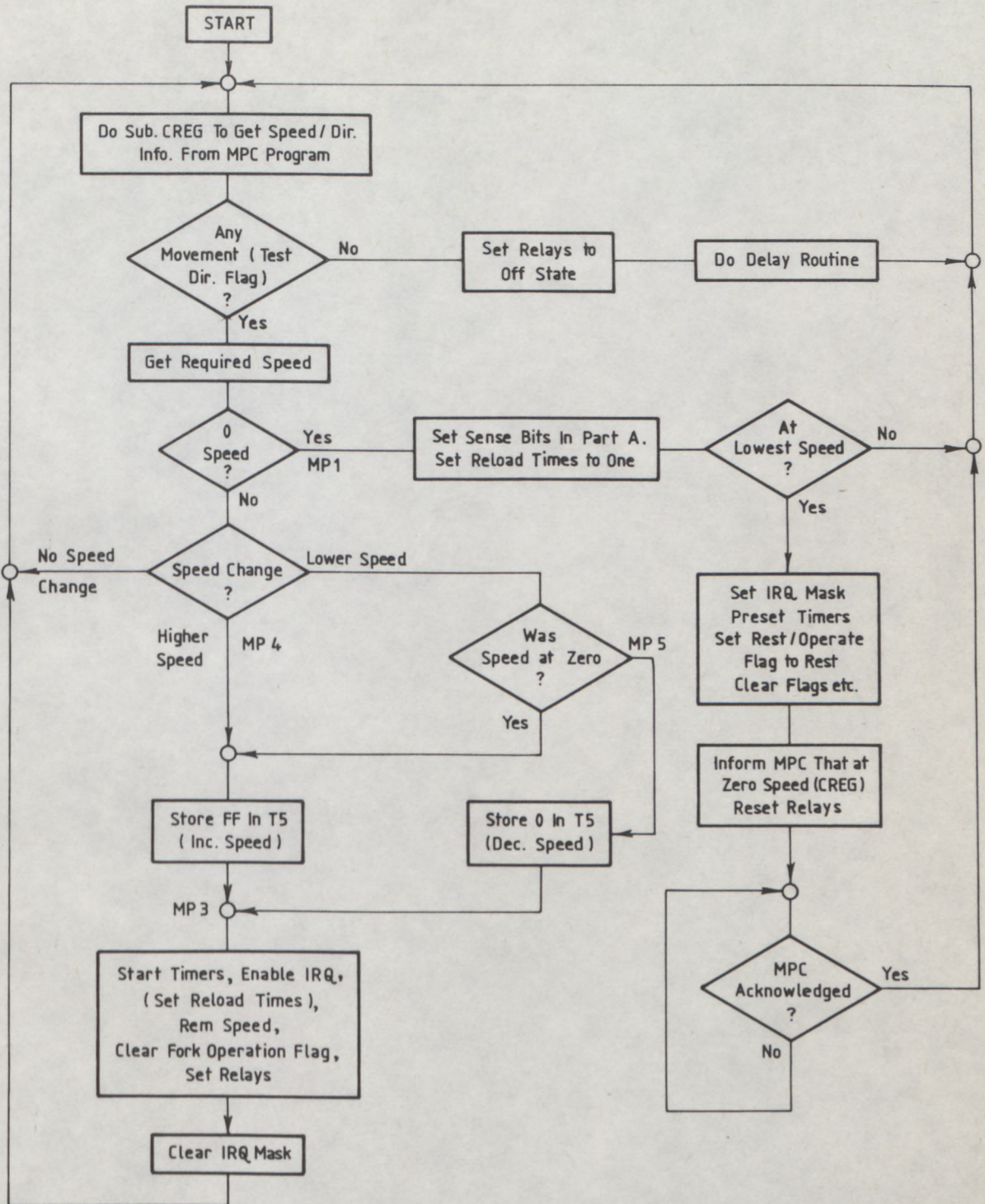
```


00233	F954	B7	1802		STA A	PIAD2	RESET RELAY
00234	F957	39		RE3	RTS		
00235				*			
00236				*/DELAY ROUTINE/			
00237	F958	86	FF	DELAY	LDA A	#\$FF	
00238	F95A	C6	02		LDA B	#\$02	
00239	F95C	4A		DD1	DEC A		
00240	F95D	81	00		CMP A	#\$00	
00241	F95F	26	FB		BNE	DD1	
00242	F961	5A			DEC B		
00243	F962	C1	00		CMP B	#\$00	
00244	F964	27	04		BEQ	DD2	
00245	F966	86	FF		LDA A	#\$FF	
00246	F968	20	F2		BRA	DD1	
00247	F96A	39		DD2	RTS		
00248				*			
00249				*			
00250	FA00				ORG	\$FA00	
00251				*			
00252				* INTERRUPTS			
00253				*			
00254				*/NMI/			
00255	FA00	DE	86	HZ100	LDX	REG1	RELOAD LAST TABLE POINTER
00256	FA02	7D	008D		TST	SPFLG	REST OR OPERATE?
00257	FA05	27	6E		BEQ	RT1	REST MODE
00258	FA07	B6	1800		LDA A	PIAD1	ERROR SENCE PORT
00259	FA0A	26	27		BNE	DEC2	DEC SPEED IF SET
00260	FA0C	96	90		LDA A	FORK	TEST FOR FORK PULSE
00261	FA0E	81	FF		CMP A	#\$FF	
00262	FA10	26	02		BNE	OK	
00263	FA12	20	1F		BRA	DEC2	SLOW MOTOR TO LOWEST SPEED
00264	FA14	7C	0090	OK	INC	FORK	
00265	FA17	96	82		LDA A	T3	
00266	FA19	81	00		CMP A	#\$00	
00267	FA1B	26	42		BNE	LOTIM	
00268	FA1D	96	89	MS1	LDA A	REG2+1	AVE LSB
00269	FA1F	D6	88		LDA B	REG2	AVE MSB
00270	FA21	90	80		SUB A	T1	DISC PERIOD - SET PERIOD
00271	FA23	D2	81		SBC B	T2	
00272	FA25	22	1E		BHI	INCS	INCREASE SPEED
00273	FA27	27	36		BEQ	LOTIM	
00274	FA29	96	84	DECS	LDA A	T5	
00275	FA2B	81	00		CMP A	#\$00	TEST IF AT SET SPEED
00276	FA2D	27	04		BEQ	DEC2	
00277	FA2F	86	10		LDA A	#\$10	
00278	FA31	97	85		STA A	T6	SPEED RELOAD FLAG
00279	FA33	96	87	DEC2	LDA A	REG1+1	DECREASE SPEED
00280	FA35	81	00		CMP A	#\$00	LOWEST SPEED (LONGEST PULSE)
00281	FA37	26	06		BNE	DEC1	
00282	FA39	96	86		LDA A	REG1	MSB
00283	FA3B	81	FC		CMP A	#\$FC	
00284	FA3D	23	20		BLS	LOTIM	
00285	FA3F	09		DEC1	DEX		NEXT VALUE FROM TABLE
00286	FA40	09			DEX		
00287	FA41	DF	86		STX	REG1	REMEMBER POINTER
00288	FA43	20	1A		BRA	LOTIM	
00289	FA45	96	84	INCS	LDA A	T5	
00290	FA47	81	FF		CMP A	#\$FF	

00291	FA49	27	04		BEQ	INC2	
00292	FA4B	86	10		LDA A	#\$10	
00293	FA4D	97	85		STA A	T6	SPEED RELOAD FLAG
00294	FA4F	96	87	INC2	LDA A	REG1+1	
00295	FA51	81	1E		CMP A	#\$1E	HIGHEST SPEED (SHORTEST PULSE
00296	FA53	26	06		BNE	INC1	
00297	FA55	96	86		LDA A	REG1	
00298	FA57	81	FF		CMP A	#\$FF	
00299	FA59	27	04		BEQ	LOTIM	
00300	FA5B	08		INC1	INX		NEXT VALUE FROM TABLE
00301	FA5C	08			INX		
00302	FA5D	DF	86		STX	REG1	REMEMBER TABLE POINTER
00303	FA5F	A6	00	LOTIM	LDA A	0,X	MSB
00304	FA61	E6	01		LDA B	1,X	LSB
00305	FA63	B7	1404		STA A	T2M	LOAD TIMER #2
00306	FA66	F7	1405		STA B	T2L	
00307	FA69	7C	0082		INC	T3	
00308	FA6C	96	82		LDA A	T3	
00309	FA6E	91	85		CMP A	T6	LOAD SAME VALUE X TIMES
00310	FA70	2D	03		BLT	RT1	
00311	FA72	7F	0082		CLR	T3	
00312	FA75	BD	F923	RT1	JSR	WDOG	TRIGGER WATCHDOG TIMER
00313	FA78	3B			RTI		
00314				*			
00315	FB00				ORG	\$FB00	
00316				*			
00317				*/IRQ/			
00318	FB00	7F	0090	PCOUNT	CLR	FORK	FORK IS OPERATING, FLAG
00319	FB03	B6	1401		LDA A	CR2	CHECK FOR COUNTER OVERFLOW
00320	FB06	84	01		AND A	#\$01	
00321	FB08	27	08		BEQ	PC1	
00322	FB0A	CE	FFFF		LDX	#\$FFFF	
00323	FB0D	FF	1402		STX	T1M	
00324	FB10	20	65		BRA	TR	
00325	FB12	F6	1402	PC1	LDA B	T1M	MSB COUNT
00326	FB15	B6	1403		LDA A	T1L	LSB COUNT
00327	FB18	CE	FFFF		LDX	#\$FFFF	
00328	FB1B	FF	1402		STX	T1M	
00329	FB1E	43			COM A		
00330	FB1F	53			COM B		
00331	FB20	C1	40		CMP B	#\$40	16384 MAX INPUT
00332	FB22	2A	53		BPL	TR	
00333	FB24	36			PSH A		
00334	FB25	37			PSH B		
00335	FB26	96	83		LDA A	T4	4 SUMS COUNT
00336	FB28	81	04		CMP A	#\$4	
00337	FB2A	27	23		BEQ	NAV	SET NEW AVERAGE
00338	FB2C	33			PUL B		ADD TO SUM
00339	FB2D	32			PUL A		
00340	FB2E	9B	8B		ADD A	REG3+1	ADD LSB
00341	FB30	D9	8A		ADC B	REG3	ADD MSB
00342	FB32	97	8B		STA A	REG3+1	
00343	FB34	D7	8A		STA B	REG3	
00344	FB36	7C	0083		INC	T4	INC 4 AVE COUNTER
00345	FB39	96	83		LDA A	T4	
00346	FB3B	81	04		CMP A	#\$04	
00347	FB3D	26	38		BNE	TR	NOT 4 YET
00348	FB3F	96	8B		LDA A	REG3+1	(LSB SUM)

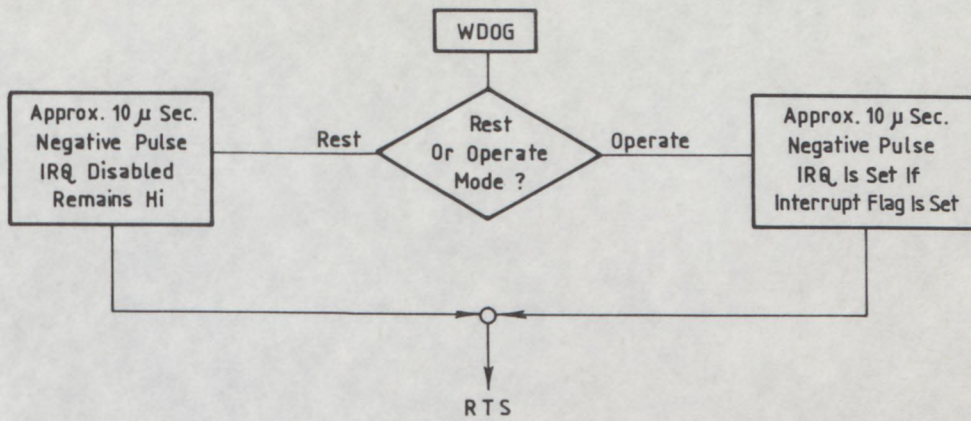
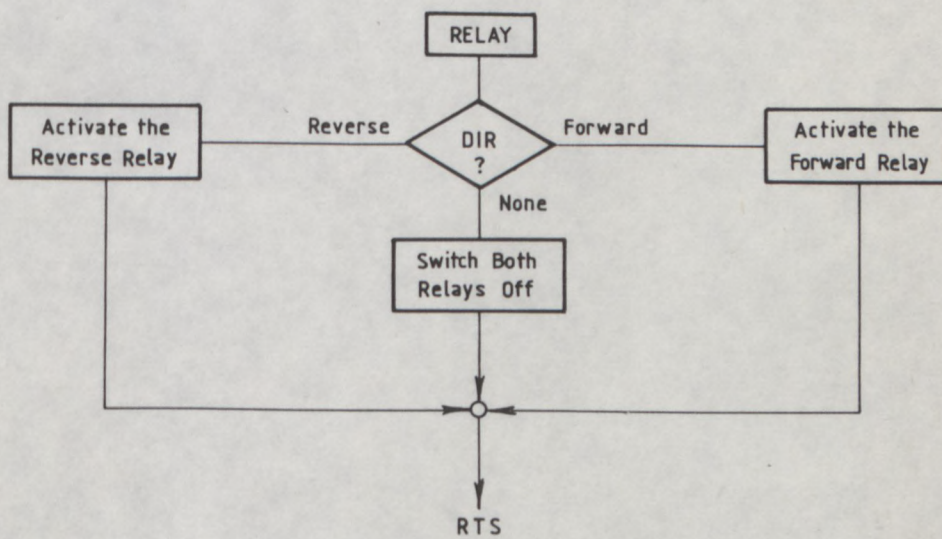
00349	FB41	D6	8A		LDA B	REG3	(MSB SUM)
00350	FB43	0C			CLC		COMPUTE AVERAGE
00351	FB44	54			LSR B		
00352	FB45	46			ROR A		
00353	FB46	0C			CLC		
00354	FB47	54			LSR B		
00355	FB48	46			ROR A		
00356	FB49	97	89		STA A	REG2+1	
00357	FB4B	D7	88		STA B	REG2	AVE MSB
00358	FB4D	20	28		BRA	TR	
00359	FB4F	96	8B	NAV	LDA A	REG3+1	
00360	FB51	D6	8A		LDA B	REG3	
00361	FB53	90	89		SUB A	REG2+1	SUM - AVE (LSB)
00362	FB55	D2	88		SBC B	REG2	SUM - AVE (MSB)
00363	FB57	97	8B		STA A	REG3+1	SUM OF 3
00364	FB59	D7	8A		STA B	REG3	
00365	FB5B	33			PUL B		NEW VALUE
00366	FB5C	32			PUL A		
00367	FB5D	9B	8B		ADD A	REG3+1	
00368	FB5F	D9	8A		ADC B	REG3	
00369	FB61	97	89		STA A	REG2+1	
00370	FB63	97	8B		STA A	REG3+1	NEW SUM
00371	FB65	D7	88		STA B	REG2	
00372	FB67	D7	8A		STA B	REG3	NEW SUM(MSB)
00373	FB69	0C			CLC		
00374	FB6A	74	0088		LSR	REG2	
00375	FB6D	76	0089		ROR	REG2+1	
00376	FB70	0C			CLC		
00377	FB71	74	0088		LSR	REG2	NEW AVERAGE
00378	FB74	76	0089		ROR	REG2+1	
00379	FB77	B6	1800	TR	LDA A	PIAD1	RESET IRQ FLAG
00380	FB7A	3B			RTI		
00381				*			
00382	FC00				ORG	\$FC00	
00443					OPT	L	
00444	FF18	13A6			FDB	5030,5020,5010,5000	
00445				*			
00446	FFF8				ORG	\$FFF8	
00447	FFF8	FB00		INTREQ	FDB	\$FB00	
00448	FFFA	0000		SOFINT	FDB	\$0000	
00449	FFFC	FA00		NOMINT	FDB	\$FA00	
00450	FFFE	F800		RESTAR	FDB	\$F800	

TOTAL ERRORS 00000

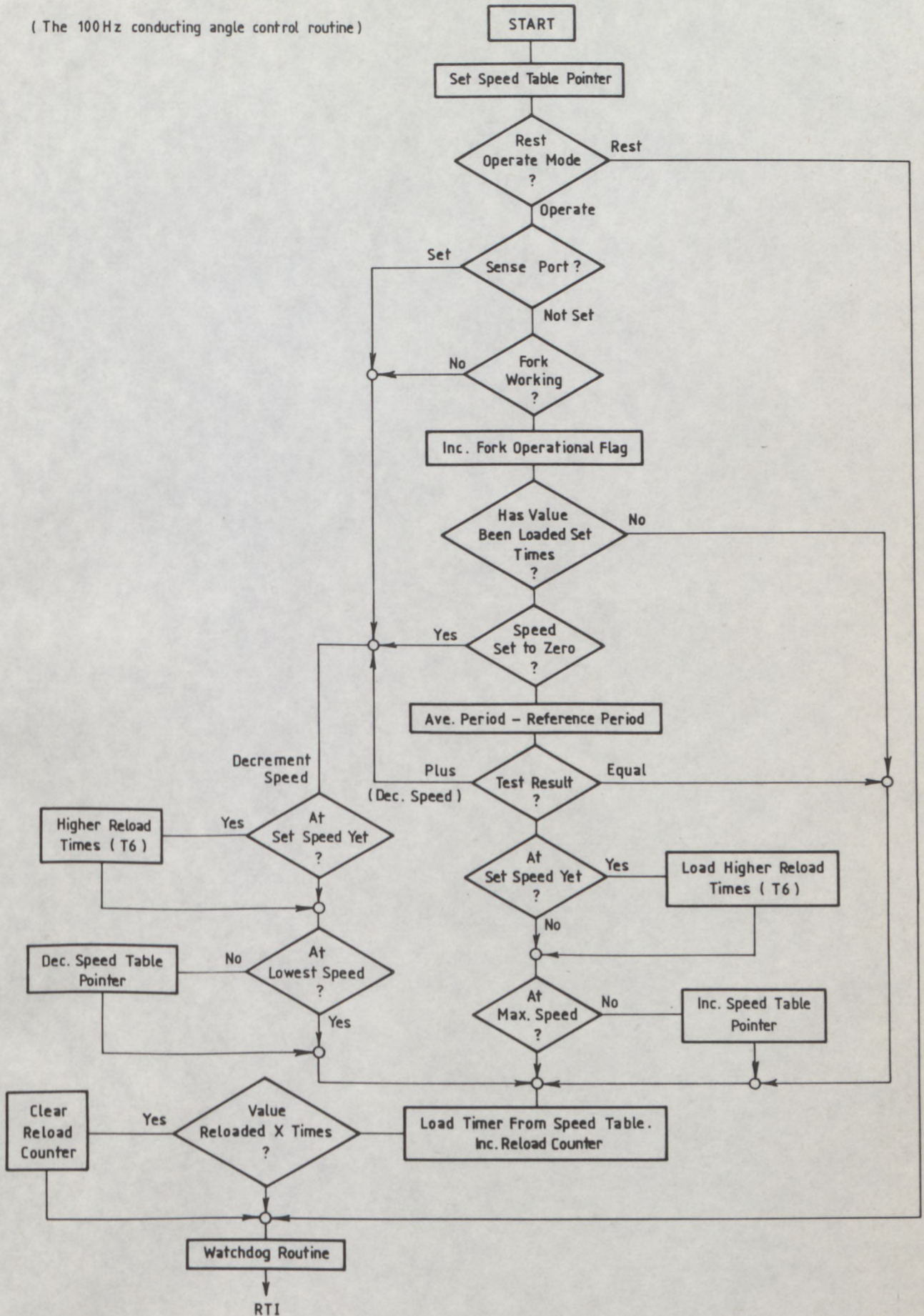


MK2 MOTOR SPEED CONTROLLER
MAIN PROGRAM

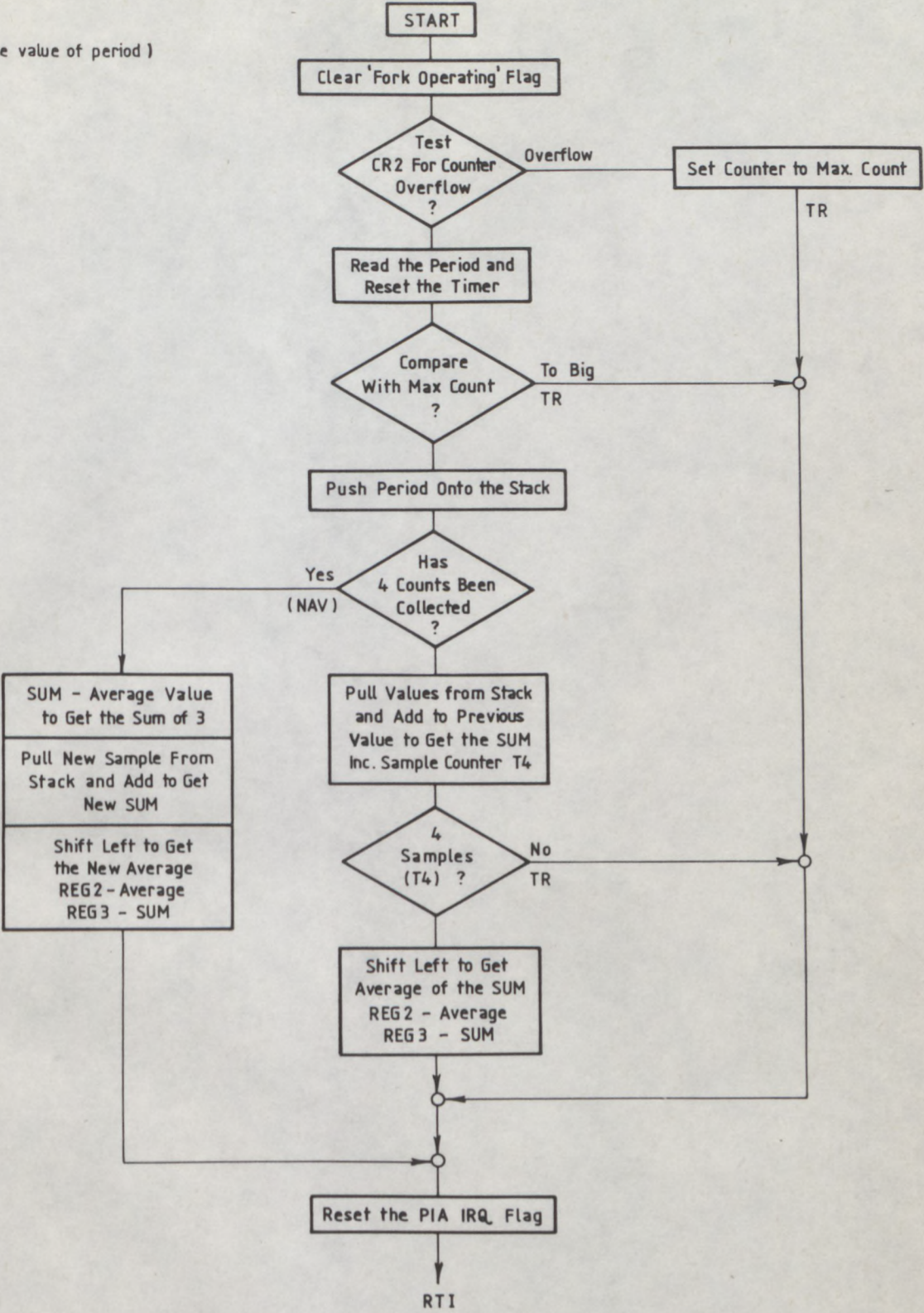
Fig. G.1

WATCHDOG TIMERFig. G.2 (a)RELAY CONTROLFig. G.2 (b)

(The 100Hz conducting angle control routine)



(Average value of period)



IRQ ROUTINE

Fig. G.4

