CLM-PDN 1/73

MISER

A MINIMUM SIZE EXECUTIVE FOR THE MODULAR 1

by

M Calderbank

ABSTRACT

This report describes the MISER executive program for a Modular 1 mini-computer. The facilities available to a user program or operator are listed together with a description of the structure of the executive and its performance.

U.K.A.E.A. Research Group

Culham Laboratory

Abingdon

Berkshire

June 1973

CONTENTS

1. INTRODUCTION

MISER (MInimum Size Executive Routine) was written to economise on core space. It was designed to be as modular as possible within the constraints of size and efficiency and as adaptable as possible while still remaining compatible with the E2 assembler and loader, (See Appendix E). It can be reduced to a minimum of about 2K words although 4K is a usual size and is configurable for different I/O devices.

This report describes the executive in terms of its main components, the scheduler, the I/O system, the interrupt handler, and so on.

2. SYSTEM DESIGN PHILOSOPHY

A basic feature of the design is that parameter checking is not done where it leads to extensive code. Indeed, certain facilities are provided which if misused, can cause the executive to fail. The design aim is that nothing should prevent a program or suite of programs making efficient use of the hardware. One notable omission from MISER is that there is no store management system. Instead, it is up to the programmer or operator to decide where in store a program is to be loaded. Although it would not be difficult to add a store management routine to MISER it is felt that for the kind of dedicated systems for which MISER is best suited such a routine is superfluous and might detract from the flexibility of the executive. (See 'Modular 1 User Group Newsletter 2', November 1970).

The executive will be described in 5 broad areas

Program Management	...	See Section 3
Peripheral Management	...	See Section 4
Operator Control	...	See Section 5

Miscellaneous Extracodes ... See Section 6

Message Facilities ... See Section 7

3. PROGRAM MANAGEMENT

3.1 Catalogue of Programs

Programs or activities within MISER are catalogued in a table known as the PRNTABLE and each activity has a unique tag associated with it at load time, known as a PRN or Program Reference Number. Up to 16 PRNs can be loaded or catalogued at any time. A PRN has a CONTROL BLOCK associated with it which is accessed via the PRNTABLE in the following way:

3.1.1 Absolute word 4 of core contains the address of the PRNTABLE.

3.1.2 The nth entry in the PRNTABLE contains the address of the CONTROL BLOCK associated with the activity or program whose PRN is n.

3.1.3 If the entry is held negative then the PRN is free and the associated CONTROL BLOCK is unused. The PRN is said to be 'absent' or 'not loaded'.

3.1.4 If the entry is positive then the CONTROL BLOCK is occupied and the PRN must be considered as a candidate for running.

The CONTROL BLOCK consists of 16 consecutive words as follows:

WORD 0	=	P Counter	These eight words
WORD 1	=	A Register	hold the current
WORD 2	=	B Register	values of the PRN's
WORD 3	=	M Register	active registers.
WORD 4	=	W Register	If a PRN has not
WORD 5	=	X Register	yet been active
WORD 6	=	Y Register	then only the X
WORD 7	=	Z Register	register is defined

WORD 8 = Entry point of PRN relative to X segment
WORD 9 = Y Register } These two values are defined at
WORD 10 = Z Register } load time and may not be altered
WORD 11 = Program status word - see below for details
WORD 12 = CH0, CH1 } The Four Character
WORD 13 = CH2, CH3 } Identifier of the PRN
WORD 14 = PRN No./Main PRN No. - see below
WORD 15 = Four Fields used for queuing purposes - see below

Word 15 is used to implement the queuing mechanism. Four 4 bit fields are used.

Field 1 =0 if no PRNs waiting to run this one

=PRN which is at head of queue waiting to run this one.

Field 2 =0 if this PRN is either the tail of a queue of programs or is not in any queue at all.

=Next PRN in queue if this PRN is in a queue of PRNs.

Field 3 BIT 3 = 0 = RUNF required
1 = Run required

BITS 2-0 = Priority of PRN to be run

Field 4 = PRN to be run or RUNF.

All programs have at least one entry point with an entry address = 0 relative to X segment. Some programs have several side entry points with non zero entry addresses.

As far as the low level special state executive routines are concerned each entry point is a different PRN and several entry points of one program can be active simultaneously. However, higher level executive routines such as those in the TINT program regard one program as a collection of PRNs. For the purposes of such higher level routines word 9 of the control block holds not only the PRN of the entry point but the PRN of the first entry point or main entry point of the program.

Thus the operator command to remove PRN 5 (say) from the system will also remove other PRNs of the same program.

The most important item in the block from a scheduling point of view is the program status word (PSW). The use of this word is as follows:

BIT 15 = Active/Inactive bit. If set then the PRN may be scheduled

BIT 14 =
BIT 13 = } Current priority of PRN in range 1-7 (0 is reserved)
BIT 12 =

BIT 11 = Suspension marker. If set the PRN is suspended

BIT 10 =
BIT 9 = } I/O Count. This is the number of executive controlled transfers currently in progress for this PRN
BIT 8 =

BIT 7 =
BIT 6 =
BIT 5 = } I/O Sum. This is the sum of all the I/O identifiers for all transfers which have completed since the last SUS command
BIT 4 =
BIT 3 =

BIT 2 =
BIT 1 = } Debug qualifier - indicates reason for failure of PRN = 0 (program gave debug SVC), = 1 (Violation), = 2 (N.S store reject)
BIT 0 =

3.2 SCHEDULER Routine

The principle routine in the Program Management section is the low-level scheduler routine itself. This routine may be entered from another special state routine at any time providing that all normal state registers have been saved. It will select a normal state activity that can 'go' and enter it. The algorithm chosen is such that the highest priority active PRN is entered. If several PRNs are active at the same priority then successive entries to the scheduler will give each of them

control in turn.

The scheduling time varies from 80μs to 300μs depending upon the number of PRNs loaded, the number active, and their relative priority.

3.3 Loading and Freeing a Program

Programs are loaded into store and entries made in the PRNTABLE and associated Control Block by the LOAD program which is a permanent program in MISER and has a PRN of 2. The LOAD program requires several parameters in order to load a program. These may be passed internally or, alternatively it will interact with the operator via the executive console to obtain them. A complete description of the LOAD program is given in Appendix A.

Programs are removed from the system and the associated Control Block is released by means of a normal state routine which forms a part of PRN 1. (The TINT program). This facility can be called externally via the executive console (see section 5.3.5) or internally by issuing a RUNF call for PRN 1 with the M register set to contain the main entry PRN of the program to be freed.

Just as the LOAD program may allocate several PRNs to a single program, so the FREE routine may remove several PRNs which are side entries to a single program.

3.4 Activating a PRN

A PRN may be activated by another PRN issuing one of 3 executive calls. The A register must hold the PRN and the B register its priority.

3.4.1 'SCHED' Executive Call 132

This activates the PRN to run in parallel with the calling PRN.

3.4.2 'RUN' Executive Call 12

This suspends the calling PRN before activating the called PRN. It is used to call a service program to act on behalf of the calling program. If the called PRN is already active then the calling PRN is placed on a queue waiting to run the called PRN.

3.4.3 'RUNF' Executive Call 32

This executive call is identical to 'RUN' except that the Z segment and M register of the calling PRN are passed to the called PRN. This allows information to be passed in the Z segment with the M register pointing to the information area.

The following error numbers are placed in the A register if any of the above 3 executive calls are rejected.

error number	meaning
1	PRN not loaded
2	PRN already active (not RUN or RUNF)
3	PRN active and suspended (not RUN or RUNF)
9	Circular Q being formed (not SCHED)

Whenever a PRN is made active its registers assume the following initial values:

P = Entry point address in X segment (usually Ø for a main entry point)

A = PRN of program which called this PRN.

B = PRN of self.

M = Ø (SCHED or RUN calls)

= M register of calling PRN (RUNF)

W = Ø

X =)
) Load time segments
Y =)

Z = Load time Z segment (SCHED or RUN)

= Z segment of calling PRN (RUNF)

When a PRN has been called by a RUN or RUNF executive call it should always return control to the calling PRN by issuing a REVIVE1 or REVIVE2 call with its A register holding the PRN to be revived.

3.4.4 'REVIVE1' Executive Call 64
'REVIVE2' Executive Call 36

These two executive calls are identical except that REVIVE2 causes the suspended PRN's P counter to be incremented by 1 before it is unsuspended or revived. This enables the called PRN to chose between a success (REVIVE2) or failure (REVIVE1) return to the caller. The PRN issuing the REVIVE call is itself terminated. Errors causing the call to be rejected are

A = 1 PRN to be revived is not loaded

A = 2 PRN is active

A = 9 PRN is not suspended.

3.5 PRN Termination - 'TERM', 'DEBUG', 'RESET'

A PRN may terminate itself normally by issuing the 'TERM' (16) executive call. An alternative method of terminating is to issue the 'DEBUG' (40) executive call. This causes the PRN to be treated as having failed with a failure qualifier = Ø. If a PRN violates or the core store the PRN is using fails then the PRN is terminated with failure qualifiers 1 and 2 respectively. PRN failures cause the executive to print the following message on the executive console.

FAIL N Q P A B M W X Y Z

where N = PRN which has failed

Q = failure qualifier = Ø 'DEBUG' call
= 1 Violation
= 2 Normal state store failure

and P A B M W X Y Z are the current values of the PRNs registers (in hexadecimal notation).

A PRN may forcibly terminate another PRN using the 'RESET'(1) executive call with the PRN to be reset in its A register. Peripheral transfers which are under way when a PRN terminates complete in the normal way.

3.6 Entries to Scheduler

The low level scheduler is entered when any of the following events occur.

3.6.1 A PRN terminates or fails

3.6.2 An I/O operation terminates

3.6.3 A PRN suspends itself waiting for I/O to complete or for a message

3.6.4 A message is sent to a PRN

3.6.5 A clock interrupt occurs

3.6.6 A PRN issues a 'DROPOUT' executive call which is an explicit call on the scheduler

3.6.7 A PRN calls another PRN using the RUN or RUNF executive calls.

3.7 Changing Priority - 'PRIORITY' Executive Call 128

A PRN may change its priority (or that of any other PRN) by issuing the PRIORITY executive call. The A register holds the PRN and the B register the priority in the range 1-7.

On exit the top 4 bits of the PSW of the program in question before the change of priority are returned in the A register. (See section 3.4). If the PRN is not loaded then this call is rejected.

4. PERIPHERAL MANAGEMENT

4.1 Standard Peripherals 'READ', 'WRITE', 'SUS', 'CANCEL'

4.1.1 A class of peripherals which have the following characteristics is handled by I/O routines in MISER. (See also App. F and the description of the CNFG program in App. D).

4.1.1.1 The peripheral must interrupt the processor to transfer data.

4.1.1.2 The transfer of data must commence after some start command has been given and cease after a stop command.

4.1.2 Each peripheral device has a logical device number associated with it. A user program initiates a transfer by issuing an executive call 'READ/WRITE' (24) with its B register pointing to a 6 word parameter area in either the Y or Z segment. A negative address in B signifies the Y segment and a positive address, the Z segment. The following information must be set up in the parameter area.

4.1.2.1 Word Ø = Logical Device Number

4.1.2.2 Word 1 = Start address of transfer

4.1.2.3 Word 2 = No. of characters to be transferred

4.1.2.4 Word 3 = Termination Character

4.1.2.5 Word 4 = Segment Identifier

= Ø/1 (Z/Y segment)

4.1.2.6 Word 5 = Transfer Identifier.

If any of the parameters are faulty the 'READ/WRITE' call will be rejected with A=5. If the peripheral is busy or offline then the reject qualifier is A=4. Information is transferred, with characters occupying one word each, until either the

required number have been transferred, or the termination character is met. If the transfer is not to terminate when a particular character is met then -1 should be given as the termination character. In the case of BSI acceptors, Source Terminate also terminates the transfer.

4.1.3 Transfers from input devices always transfer -1 as the last character. This means that the actual number of characters obtained from the device is one less than the number specified.

4.1.4 A program may wait for a transfer to terminate by either issuing the 'SUS'(4) executive call or a 'WAIT' (140) executive call with the channel 8 bit set (see section 7.2). In either case the program will be suspended only if it has outstanding I/O transfers and no transfer identifiers to be returned. When the program is resumed its A register will contain the sum of the transfer identifiers of all the transfers which have completed since the last occasion it issued 'SUS' (or 'WAIT' on channel 8).

4.1.5 Miser restricts transfer identifiers to the range Ø-15 and no PRN may have more than 7 transfers outstanding at any one time.

4.1.6 B.S.I. source interfaces may be configured to receive Acceptor Error interrupts. If an acceptor error interrupt occurs during a transfer then the transfer is terminated after the next normal data interrupt, source terminate is set in the data interrupt dedlocks and bit 2^3 is set in the transfer identifier of the transfer under way.

4.1.7 At power on time all I/O transfers are cancelled. Furthermore when (X-ON) is typed on the executive console, devices Ø-2 have any I/O transfers cancelled. Any PRN may cancel I/O on any device by issuing a 'CANCEL' (108) executive call with its B register containing the logical device number. A cancelled I/O

transfer will normally terminate on the next character without informing the PRN which set up the transfer. The device is immediately made free for use.

4.1.8 For those peripherals which cannot be handled by the MISER I/O routines it is possible to specify a PRN number to which all 'READ/WRITE' executive calls should be passed. This has the advantage that other programs can access these peripherals in the standard way. The effect of issuing a READ/WRITE executive call to a logical device number which is handled by a PRN is as if the 'RUN' executive call had been given. The calling program is suspended and the called PRN is activated. If the called PRN is not loaded the transfer is rejected with A=5.

4.2 Disc Unit

Two versions of MISER are available with disc routines for handling the 1.542 and 1.541 disc units. The routines merely initiate disc transfers and return error qualifiers and were intended to be used with a filing system program. (See Section 9.2).

Two executive calls are provided to access the disc.

'DISCIO' Executive Call 68;

Parameters:

A Register	Bits 15-8 - Cylinder Address Bits 6-3 - Drive Number Bits 2-0 - 0/1 = R/W (Read or Write)
B Register	Bits 8-4 Track on Cylinder (1-20) Bits 4-0 Page on Track (0-13)
M Register	Store pages involved in transfer in absolute addresses. In the form 1st page/last page

The cylinder, track and page addresses may be calculated from the absolute disc page as follows:

Cylinder = Absolute Page/280 (save the remainder R1)

Track = R1/14 (save the remainder of the division R2)

Page on Track = R2

This executive call initiates the transfer and success causes the next instruction to be skipped. If the disc is busy control returns to the failure exit with a reject qualifier in the A Register (=4).

'DISCWAIT' Executive Call 72

Parameters:

None

The calling program is suspended until its disc I/O has completed. If Disc I/O has already completed control is passed back immediately.

The A register is loaded with the success/failure flag

Values taken are
- 0 = Successful transfer
- 1 = Not used
- 2 = Disc reports error while searching or transferring
- 3 = Disc rejects transfer after successful seek

The B register is loaded with the status word from the disc controller. Illegal cylinder address will give rise to a busy condition when the Read/Write is initiated.

Illegal track or page in track will give rise to Failure 3

Illegal store address will give rise to Failure 2.

4.3 Peripherals handled by User Programs

The executive has the following facilities to enable non standard peripherals to be handled by normal state user programs. This includes giving commands to the peripheral and receiving interrupts from a peripheral.

4.3.1 Linking to an interrupt

Peripherals interrupt to a series of 4 locations known as

dedlocks. Two executive calls are provided to manipulate the contents of dedlocks and connect interrupts to a particular PRN.

4.3.1.1 'LINK'executive call 8

The A register contains a PRN to which the dedlock is to be linked and the B register contains the absolute dedlock address. Any PRN may be linked to any dedlock in page 1 or 2 of store. If the dedlock address is outside this range the call will be rejected. This call inhibits the interrupt in question by setting bit 15 of the P dedlock to a 1. An 'endint' executive call must be given before an interrupt can be received.

4.3.1.2 'ENDINT'Executive call 56

The A register must contain a value which is to be used as the slave word of the next interrupt transfer and the B register contains the dedlock address. (The dedlock address must be in page 1 or 2). If the dedlock address is valid the interrupts are permitted by resetting bit 15 of the P dedlock.

When a linked interrupt occurs, the PRN in question is scheduled to run at priority 7 with all interrupts off. The A register will contain the master word from the transfer and the B register will contain the dedlock address. The PRN should permit interrupts by giving an executive call 'ION' or 'ENDINT' within a reasonably short time of being entered (400-800μs). As well as having global interrupt inhibition the dedlock in question is also inhibited from interrupting again and can only be cleared by the 'ENDINT' executive call for that dedlock. A single PRN may receive interrupts from several different dedlocks, but if this is done it is recommended that interrupts are not permitted again except after the interrupt received has been completely

serviced and the PRN should terminate immediately after the 'ENDINT' executive call. If a linked interrupt occurs for a PRN which is already active or suspended then it will be ignored and the interrupt will be inhibited.

4.3.2 Giving Peripheral Commands

4.3.2.1 'OBEYFN'- Executive call 92

Any function may be obeyed in special state using this executive facility. The function which is to be obeyed must be loaded into the M register. Any values which the special state A or B register should contain are loaded into the A and B registers. The executive call is issued. On return from the executive call the A and B registers will hold the same values as the special state A and B registers and if the function caused an instruction to be skipped in special state then an instruction will be skipped on return.

Example:

Supposing command 4 is to be given to a peripheral on channel 7 device 1 with a master word of 255. The following coding will suffice.

```
        LDA  L  255;        Master Word
        LDB  X  PADDR;      Peripheral Address
        LDM  X  PFUNCT;     Function to be obeyed
        EXEC 92;
        JMP  REJ;           Peripheral rejects call
        - - - - -
PADDR:  @ ||||||....|..... ; Channel 7 device 1
PFUNCT: STA  YB  4;     In SS   STA  B  4
```

4.3.2.2 'ION' and 'IOF' calls 52 and 48

These executive calls turn all interrupts on or off. They should be used with care.

5. OPERATOR CONTROL

5.1 The X-ON interrupt

The X-ON interrupt from the executive teletype is the standard mechanism for activating a program to communicate with the machine operator. The M dedlock of the X-ON interrupt is assumed to contain the PRN to be activated. All I/O on the executive console is first cancelled (devices Ø,1,2) and then the PRN in question is first reset and then scheduled with priority 5. If the PRN in question is not loaded in the machine then a standard PRN (=1), the TINT program, is used which forms a part of the executive. The remainder of this section will be concerned with the standard teletype interrupt program (TINT).

5.2 The TINT program

This program prints its name as soon as it is activated and requests a two character command in reply. The commands accepted cover the same areas as the internal MISER facilities. If an illegal command is given the prompt is repeated. Data for the command is requested in response to further prompts. The operator may type a reply in one of the following formats.

5.2.1 <positive integer in range Ø-32767>

5.2.1 <page no. in range Ø-255>, <positive integer in range Ø-32767>

5.2.2 @ <1-4 hexadecimal characters>

If a '?' is typed the number read in so far is ignored and input restarts.

The TINT program may itself be interrupted and reset by

typing <X-ON>.

5.3 TINT Program Commands

5.3.1 'NO'

This causes the TINT program to terminate.

5.3.2 'LD'

This causes the LOAD program to be run with priority 2. (See Appendix A).

5.3.3 'RN'

Prompts: PRN: <reply with PRN number>

PTY: <reply with priority>

The PRN in question is scheduled with the requested priority.

5.3.4 'AB'

Prompt: PRN: <reply with PRN>

The PRN in question is reset or aborted.

5.3.5 'FR'

Prompt: PRN: <reply with PRN>

The PRN in question is removed from the executive tables. Only a main PRN is removed and any side entry points are also removed.

5.3.6 'LI'

Prompt: PRN: <reply with a PRN>

This command causes the TINT program to list all PRNs in the executive lists, starting with the PRN referred to. The format is shown below.

<PRN> <NAME> <XO> <XN> <YO> <ZO> <ZN> <PSW>

where <XO>,<XN> etc. are the X,Y,Z segment bounds and

<PSW> = Program Status Word in hexadecimal notation.

5.3.7 'RP'

Prompt: DEV: <reply with a logical device number>

This command causes all I/O on the device in question to be cancelled and the device made available to any program. If the device is in use the using PRN is not notified of the terminated I/O transfer.

5.3.8 'RD'

Prompts: ADDR: <reply with an absolute core store address>

NO: <reply with a number n>

This command causes the TINT program to read the absolute location given and the following n locations and print them out in hexadecimal format 8 words to a line.

5.3.9 'ST'

Prompts: ADDR: <reply with an absolute core store address>

VAL: <reply with any 16 bit value>

This command stores the 16 bit value into the absolute address specified. It should be used with care, especially if addresses in the executive area of core are specified.

5.3.10 'MS'

Prompt: PRN: <reply with a PRN>

This command sends a message on channel 2 with a message word of Ø and no segment to the specified PRN. (See Section 7.0).

5.3.11 'DL'

'FL'

'EX'

These three commands are used only in the disc based versions of MISER. 'FL' causes the file systems teletype handler to be run (PRN=4, PTY=2) and 'DL' causes the LOAD program to be run to load a program from disc while 'EX' brings

into store the command execute program (see section 9.2).

6. MISCELLANEOUS EXTRACODES

A number of extracodes are provided to enable a user program to by pass the protection mechanism.

6.1 Read and Write to any part of store

'GET'	executive call	252
'PUT'		248

In both cases the absolute core store address must be in the B register. The A register contains the contents of that location after the GET operation and before the PUT operation. For the PUT operation to be successful the M register must hold the value 211. The GET operation is always obeyed and there is no reject branch from this extracode. The PUT operation may fail if the M register is not correctly set up and the reject branch entered.

6.2 Segment manipulation

'RELOC'	Executive call	88
'SETSEG'		84
'UNSETSEG'		80

None of these extracodes have reject branches. The first, RELOC, places the X, Y, and Z register values in the programs A, B and M registers. This enables a program to find the absolute addresses of its segments. A program may change its Y or Z segments from their load time values by setting the new Y register value in the A register, and the new Z register value in the B register, and executing the SETSEG extracode. If a register holds -1 then no change is made to the corresponding relocation register. On return from the extracode the A and B registers hold the previous values of the Y and Z

registers and in the case of a change of Y segment the W register is loaded with the absolute address of the first word of its old Y segment. (This latter facility is for those programs which set up their Y segments to span the whole of core and wish to continue to access their old Y segment using the YW addressing mode). Lastly UNSETSEG changes the Y and Z segments back to their load time values.

7. INTERNAL MESSAGE FACILITIES

Later versions of MISER have internal message facilities built in. These are modelled on those of the CTL E4 executive. Any PRN can receive messages on one of 3 channels selectively and can send messages to other PRNs on one of two channels. This provides a powerful communication and inter-task synchronisation facility.

A message consists of a 16 bit word plus, optionally the senders X, Y, or Z segment which becomes the recipients Z segment. The recipient can wait for messages on any channel or any combination of channels. Messages on channel 1 have priority over those on channel 2. The third channel (channel 8) has lowest priority. Multiple messages on a channel are received strictly in the order sent. Channel 8 is a channel down which I/O transfer completion messages are sent.

7.1 'SEND' Executive Call 136

The M register holds the PRN of the recipient and the B register the Message word.

The A register is used to specify the message parameters and is split into 3 fields.

Field 1 bits $2^{\emptyset}-2^{1}$ = Ø no segment
1 X segment
2 Y segment
3 Z segment

Field 2 bits 2^2-2^4 = channel number (1 or 2)

Field 3 bits 2^5-2^{15} = transmitted to recipient but not used by executive.

The registers are unaltered by this call. A reject branch will be entered if:

Recipient PRN is not loaded

Recipient PRN is not active or suspended

Channel number is invalid

Message queues are full.

7.2 'WAIT' Executive Call 140

This extracode is executed to wait for a message. The A register is assumed to contain a mask with bit 2^i set if a message is permitted on channel i.

Note that if bit 2^8 is set and there are no outstanding I/O transfers then the calling program will continue. On return from this extracode the calling PRNs registers will be set as follows:

7.2.1 Message on channel 1 or 2

A = Specification word as sent by Sender

B = Message Word

M = PRN of sender.

7.2.2 Message on channel 8

A = Sum of I/O transfer identifiers

B),), M) unchanged.

N.B. a WAIT with bit 2^8 of the A register set is identical to the SUS executive call (see section 4.1.4).

There is no reject branch for this extracode.

7.3 'CLEARMSG' Executive Call 144

This extracode has no reject branch and clears down all queued messages for the calling PRN. It is recommended that this call is made before a TERM or DEBUG and on initial startup of a program.

A power-on interrupt will also clear out all queued messages.

7.4 'CLOCK' Executive Call 124

This extracode allows a PRN to request a message to be sent to it after a given time interval has elapsed. (Only systems with a real time clock can support this facility).

Each PRN in the systen has a clock word which may be set by means of the above executive call. The executive call is never rejected and the clock word is taken from the user program A register.

The clock word is divided into three fields

bits Ø -7 = Count field

bits 8 - 14 = Message field

bit 15 = Ø/1 (clock on/off)

Once set, the count field is reduced by one at every clock interrupt (every second) until the count reaches zero. A message is then sent on channel 2 to the PRN in question. The message parameters are as follows

A Ø -1 = Ø (no segment)

A 2 -4 = 2 (channel number 2)

A 5 = 14 = Ø

A15 = 1 (non-I/O message)

B Ø-6 = Message field

B 7-15 = Ø

After the message has been sent the clock is switched off by setting bit 15 of the clock word.

A PRN may reset its clock word at any time including switching the clock off.

8. TIMINGS

8.1 Linked Interrupts

The time to enter a normal state PRN which is linked to an interrupt (see section 4.3), from the moment the interrupt is received to the entry to the PRN is about 60μs in fast core (0.75μs cycle time).

8.2 Character Interrupts

The time taken to service a single input character interrupt is about 30μs and for a single output character interrupt is about 25μs. This limits the upper transfer rate for interrupting devices to 30Kbytes/s.

8.3 Scheduler Time

The time taken by the scheduler to select a PRN to be entered is given by the following formula

$$T = 25 + 11x + 4y + 4z \quad \mu s$$

where

x = No. of PRNs loaded

y = No. of active PRNs

z = No. of times priority of PRN i is higher than priority of PRN i-1

The worst case when x=16, y=16, z=7 is about 300μs.

The shortest timing when x=3, y=1, z=0 is about 60μs. A more typical case with 8 PRNs loaded (x=8), 3 active (y=3) and z=1 gives a figure of about 130μs.

8.4 Special State Residency

Apart from Power On time, the longest time spent in

special state is on the termination of an I/O transfer and is about 340μs. Interrupts are turned off after about 150μs. Message sending takes about 150-200μs + time in scheduler. All paths longer than 150μs in special state have been protected by a temporary interrupt inhibition.

9. DIFFERENT VERSIONS OF MISER

It is comparatively easy to prepare different versions of MISER with different facilities. (See App. G). Consequently the version described here is not a static version but is liable to change from time to time. There are two major variants of the MISER described here:

9.1 Basic Laboratory Version

This version has no disc routines and no message facilities. The CLOCK facility has different parameters and different functions. Page 14 contains special extracodes of use to a laboratory computer and page 15 is free for use by user programs. Mr E G Murphy is responsible for this version.

9.2 Disc Based Version

This version is for use on systems with either a 28 Mbyte replaceable disc or 2Mbyte fixed head disc. A FILE program (see FEPD 15/72, 17/72) is included as part of the basic executive together with a disc bootstrap and enhancements to the TINT program. The FILE program enables files to be created, deleted, copied and edited, and programs can be assembled and dumped in hunkload format. Two extra devices become available, 16 and 17, which are input from a disc file and output to a disc file respectively. A command execute program is provided for obeying sequences of commands held in files on disc (see

FEPD 2/73). Appendix D gives a list of utility programs and documentation describing this system.

10. UTLITY PROGRAMS

Having obtained a version of MISER there are utility programs to configure it for different devices anddump out new versions of it. There is a special version of the CTL ASS4 assembler, and Miser versions of the CTL utilities MINImods and DUMP. These programs are described in Appendix D.

11. ACKNOWLEDGEMENTS

The author would like to thank Dr R J Dakin, Mr A Knight, Mr J R Morley and Mr B Weldon who have all helped to produce the MISER executive described in this report. Principal thanks, however, must go to Mr E G Murphy whose inspiration and enthusiasm led to the design and development of MISER.

25 June 1973

APPENDIX A

THE MISER LOAD PROGRAM

This program accepts E2 SLB from any character stream device and loads it into core under the control of 3 load time X, Y, Z registers. The X, Y, Z registers are specified either by the operator at the executive console or by a program which slaves the loader or by control information in the SLB stream. To accommodate this latter facility the definition of E2 SLB has been extended to include an extra control character:

Control Character Value = 8

The next 3 data words in the input stream contain load time X, Y, Z register values.

PROMPTS AND RESPONSES

PRN? <reply with the PRN to be assigned>

XYZ? <x>;<length>;<y>;<length>;<z>;<length>;

where <x>, <y> and <z> are the start page numbers for the X, Y and Z segments respectively and are followed by the lengths of these segments in pages.

If a null reply(';') is given to the XYZ prompt the LOAD program will expect the X, Y, Z register values to be specified in the E2 SLB input stream. If no X, Y, Z values have been obtained by the time the first data word has to be loaded into store then the XYZ? prompt will be repeated.

OUTPUT MESSAGES

If a successful load occurs the program name is output followed by a '!' when the loading has been completed. An error is signalled by the message

ERROR NN

where NN is the error number. The meaning of the error numbers is as follows:

Error No.	Meaning
A) LOADING ERRORS	
0	Error on tape-control character out of range
1	Error on tape-checksum failure
2	Error on tape-pointer mode illegal
4	Loading failure-address above X segment bound
5	Loading failure-address above Y segment bound
6	Loading failure-address above Z segment bound
B) PARAMETER ERRORS	
11	PRN out of range
12	PRN already allocated
17	IFR Device non operational or file not open.

SLAVING THE LOAD PROGRAM

The LOAD program may be slaved from another PRN by setting up a table of parameters in its Z segment and scheduling it.

Care must be taken to ensure that the LOAD program is inactive (PSW=Ø) before loading parameters into the Z segment. The following parameters must be passed.

Location (Page, Line)	Parameter
5, Ø	PRN to be allocated to loaded program
5, 1	Load time X register or -1 if X,Y,Z registers are to be taken from input stream
5,2	Y register value
5,3	Z register value
5,4	Ø/1 = do not schedule/schedule loaded program with priority =2

5,5	Logical device number of input stream. (N.B. this location is set to 3 at the end of the loading process)

APPENDIX B

CORE MAP

Page	Word	Use (Special state routine names in capitals)
0	0-30	Executive Global Variables
	[0	CURPRN
	1	CURSCAN
	2	NOPRNS
	3	CONADDR
	4	PRNBASE
	5	CURDEDS
	6	PLISTBASE]
		etc
	30-250	SCHEDULER
		SAVEREGS
		SETCONT
		RUN
		TERM*
		FINDPRN
		RESTORE
1	0-255	Software Dedlocks
2	0-127	Hardware Dedlocks
	128-255	REVIVE1&2*
		SCHED*
		DEBUG*
		POWER
3	0-255	READ/WRITE*
		RELOC*
		UNSETSEG*
		GET*
		SETSEG*
4	0-255	LINK*
		ENDINT*
		RESET*
		IOFF*
		ION*
		OBEYFN*
		PUT*
		PRIORITY*
		RUN/RUNF*
		X-ON
		SUS*
		DROPOUT*
5	0-63	Z segment of LOAD
	64-255	IDLE
		CLOCK*

Contd. Page	Word	Use (Special state routine names in capitals)
		PERIPHRESET BSWRAE CANCEL* MEND
6	0-255	X segment of TINT
7	0-255	X segment of TINT
8	0-100	X segment of TINT
	100-255	Y segment of TINT
9	0-255	Peripheral Lists
10	0-255	X segment of LOAD
11	0-255	X segment of LOAD
12	0-127	Message Queues
	128-255	Y segment of LOAD
13	0-255	PRN control blocks
14	0-255	DISCIO*
		DISCWAIT*
		SENDMSG*
15	0-255	WAIT MSG*
		CLEAR MSG*

*denotes that these routines service software interrupts of the same name.

APPENDIX C

SUMMARY OF EXECUTIVE CALLS

Value	Name	Parameters
0	RESET*	A = PRN, M=133+, (see section 3.5)
4	SUS	None (see section 4.1)
8	LINK*	A = PRN, B=Dedlocks (see section 4.3)
12	RUN*	A = PRN, B=Priority (see section 3.4)
16	TERM	None (see section 3.5)
24	READ/WRITE*	B=Information Area (see section 4.1)
32	RUNF*	A=PRN, B=Priority (see section 3.4)
36	REVIVE2*	A=PRN (see section 3.4)
40	DEBUG	None (see section 3.5)
48	IOF*	M=119+ (section 4.3.2.2)
52	ION	None (section 4.3.2.2)
56	ENDINT*	A=Adedlock B=dedlock (section 4.3)
64	REVIVE1*	A=PRN (section 3.4)
68	DISCIO*	see section 4.2
72	DISCWAIT	None (section 4.2)
80	UNSETSEG	None (section 6.2)
34	SETSEG	A=Y, B=Z (section 6.2)
88	RELOC	None (section 6.2)
92	OBEYFN	See section 4.3.2.1
104	DROPOUT	None (section 3.6)
108	CANCEL	B=Device Number (section 4.1)
120	WAIT(DROPOUT)	None (section 3.6)
124	CLOCK	A=block word (section 7.4)

Contd.

Value	Name	Parameters
128	PRIORITY*	A=PRN, B=New Priority (section 3.7)
132	SCHEDULE*	A=PRN B=Priority (section 3.4)
136	SEND MSG*	See section 7.1
140	WAIT MSG	A=Mask (see section 7.2)
144	CLEAR MSG	None (section 7.3)
248	PUT*	A=Value, B=address, M=211+ (section 6.1)
252	GET	B=address (section 6.1)

*signifies that this executive call uses a rejection slot.

+A security code.

APPENDIX D

LIST OF UTILITY PROGRAMS

1. Basic MISER Utilities

Name	Function
CNFG	This program configures a MISER executive for different peripherals and/or channel and device allocations. It assumes that the executive is already configured for an executive teletype (LDNs 0,1,2) on Channel 0 interrupting to dedlocks 2,32. It assumes that there is sufficient core space for up to 32 peripherals. It is the responsibility of the user to check his core allocation. (See Appendix F).

Prompts and Responses:

Dev: Reply with the logical device number in the range 3-31. N.B. Device Numbers in the range 16-31 will corrupt some versions of MISER. Device numbers in the range 0-2 cause CNFG to terminate. Device numbers > 31 will cause the prompt to be repeated.

Channel:

Reply with the interface channel on which the device is connected. If channel 0 is specified then the prompt sequence jumps to the prompt PRN:.

Sub-Channel: Reply with the Sub-channel address of the device

Dedlock Address: Reply with the address of the dedlocks to which the device will interrupt

Start Command: Give the start command for the device

Stop Command: Give the Stop Command

Input or Output or Block: Reply I or O or B

Acceptor Error: Reply Y for B.S. Source, otherwise N

The device has now been configured and may be used. The Prompt sequence goes back to Dev:.

With issues of MISER dated 16/6/72 and later a new facility becomes available whereby any peripheral may be linked to any PRN.
If channel zero is specified then the prompt 'PRN:' is output. Reply with the PRN to which the peripheral is to be linked. Any Read or

Contd.

Name	Function
	Write calls for that peripheral will then cause the specified PRN to be RUN. The program then prompts for 5 data words which are placed in words 1-5 of the peripheral list. If the PRN concerned does not use them reply ';' to each one.
MSLD	This program produces a loaderese binary tape of the MISER executive. It must be loaded (XYZ=1;1;1;) with its X segment adjacent to the top end of MISER. No other program must be run while the tape is being punched. The resulting tape will not contain MSLD. After running MSLD overwrites its first location with the 'TERM' executive call.

DUMP (Basic Version)

This program produces a DUMP of any program in systems loader binary.

Prompts: PRN? Reply with the PRN of the program
XWORDS? Reply with the number of words of X to be dumped
YWORDS? Reply with the number of words of Y to be dumped
ZWORDS? Reply with the number of words of Z to be dumped
EPOINTS? Reply with address of additional entry points required else zero.

BSSS

This program issues an EXC instruction to any peripheral passing any master word to that peripheral and printing the slave word or rejection qualifier.

Prompts and Responses:

CH?	<reply with 1.11 channel number to be used>
D?	<reply with device number on channel>
C?	<reply with command to be given>
M?	<give master word to be sent to device>
?	<reply either '1;' to repeat command or ';' to specify a new one>

N.B. All numbers must be positive decimal integers

The response takes the form <R><S> where <R> is either ACC or REJ, indicating acceptance or rejection of the command and <S> is the slave word or rejection qualifier in decimal.

Contd.

Name	Function
MINI (See write up of CTL version MINI mods)	
Commands available ..	Q Print store in integer format
	L Load into store (Integer or Instruction format)
	I Print store in instruction format
	P Change PRN being debugged
	R Print registers of program being debugged
	G Go-away (terminate)
	S Set a stop
	U Unset a stop (Max No. Stops=32)

2. Disc Based MISER Utilities

Name	Function
FILE	Filing program for disc based version (see FEPD 17/72)
TELE	Operator file command program (see FEPD 15/72)
DUMP	Disc based version of basic DUMP (see FEPD 15/72)
COPY	Copies files from device to device (see FEPD 15/72)
TIDY	Removes deleted files from the disc (see FEPD 15/72)
DCPY	Copies cylinders from one drive to another (see FEPD 25/72)
EXEC	Obeys a file of commands (see FEPD 2/73)

MSLD (Disc Version) - Dump Executive in Loaderese Format

This program is essentially the same as the paper tape program of the same name except that it outputs the loaderese stream to device 17 (a MISER disc file).
The file program must be present as PRN 3 and it is usual for MSLD to be PRN 4.
The following XYZ values must be used:

```
FILE    16:10:26:5:31:0:
MSLD    31:1:32:1:33:1:
```

Contd.

Name	Function

MSDD - Dump Executive on Executive area of the disc

This program has a 31 page Z segment which is used to hold a core image of the first 31 pages of store. It will read an executive into this area either from paper tape or disc in either loaderese or MISER reload format. It can carry out the clearing of the file programs INITFLAG. This flag is set by the file program when it has initialised itself. Since the dumped copy will usually be initialised it is necessary to clear it to force the reloaded file program to initialise itself. The resulting core image is then either copied to the executive area of the disc or punched out on paper tape as a reload format version.

The following facilities are available:

PROMPT: MODE ?
REPLY:
- G Go away (Terminates MSDD)
- L Read loaderese version of MISER into core
- R Read MISER reload version of MISER into core
- I Clear INITFLAG in file program
- P Punch MISER reload version of MISER
- D Dump core image of MISER onto executive area of disc

Other prompts are:

- IFR ? Give 3 for paper tape reader 16 for disc
- OFR ? Give 5 for paper tape punch or 17 for disc
- END ? Reply 'Y' if more loaderese tapes are to be loaded.
- NO. PAGES? Give the number of pages to be dumped (this is usually 31)

The following error messages can occur:

:::Control Character out of range
(Control characters lie in the range 0-2)
:::Address out of range
:::Page Number out of range
(MSDD loads into addresses 0-7936, Pages 0-30)
:::Checksum error
(There are checksums on the reload tapes)

Contd.

Name	Function
	::: Disc Error Disc transfer when writing the executive to disc
	The XYZ requirements of MSDD are 2:1:31: 1.542 Disc Drive The executive area of the disc is drive 0. Cylinder 200. Track 0-2 1.541 Fixed Head Disc The executive area of the disc is side 1, Track 95-96
BOOT	This is a bootstrap program to load MISER and the FILE program from disc. The following sequence of operations should be followed: 1. Ensure that system disc is loaded as Drive 0 2. All modules on except processor (1.11) 3. Load Bootstrap Program into core using Reader/Loader 4. Switch on processor - the character 'A' should be printed on the executive teletype followed by the normal MISER startup message. ERRORS: If any error occurs during the loading of the executive a single digit character is printed on the executive teletype and the processor is halted. The cause of the error must be corrected and the process repeated. The error digits are as follows: '1' Disc Unit switched off or offline '2' Seek Error - Disc may not be loaded or fault on drive '3' Transfer rejected by hardware - call engineer '4' Parity error - try again - if persistent then call engineer.

APPENDIX E

CORRECTING AN E2 PROGRAM TO RUN WITH MISER

1. Program Entry

E2 loads WORD Ø of the programs Y segment with the PRN allocated to the program and clears WORD Ø of the Z segment. MISER does not do this and the instruction 'STB Y Ø' should be inserted at the front of the program if use is made of YØ. (N.B. Programs in CLSD already generate this instruction). Similarly the location ZØ should be cleared if required. The A register is zero on entry from the E2 executive; MISER loads the calling PRN in the A register. Again it may be necessary to insert an instruction to clear the A register at the head of the program.

2. 'TERM', 'DEBUG', 'SUS', 'CANCEL', 'WAIT'

All these executive calls have identical effect in the two executives apart from the possibility that 'WAIT' may have a different result in MISER due to the different scheduling philosophy.

3. 'READ', 'WRITE'

This is the same as E2 unpacked transfers except that there are restrictions in MISER on the size of the I/O identifier and the total number of I/O transfers under way at any one time. (See Section 4.1).

4. 'RESET', 'RUN', 'REVIVE'

The MISER facilities of the same name are broadly similar but changes will have to be made to the parameters supplied (See section 3.4).

5. 'FREE', 'NAME', 'SETSEG', 'CONTEXT', 'SHAREX'
'SUBSET', 'INTASS', 'INTDIS', 'READI', 'WRITEI', 'EXITI'
'LOOK'

There are no exactly equivalent facilities in MISER but most of these facilities can be simulated by using other MISER facilities.

6. 'CHACESS', 'LINK', 'ENDINT'

Broadly similar facilities exist but the parameters to be supplied are different. (See section 4.3).

APPENDIX F

THE MISER PERIPHERAL LIST - HOW TO WRITE A PROGRAM TO HANDLE A PERIPHERAL

Absolute location 6 of core store holds the address of the MISER Peripheral list. Each logical device number occupies 6 consecutive words in this list known as the Peripheral Control Block. The first word of the control block for device n (say) is located 6n words from the start of the peripheral list. The layout of each control block is as follows:

WORD Ø bits $2^{Ø}$-2^{4}: Start command for peripheral

bits 2^{5}-2^{9}: Sub-Channel address of peripheral

bits 2^{10}-2^{12}: Channel address of peripheral

bits 2^{13}-2^{15} = 7

N.B. This word is normally negative. If it is positive then it is taken to be the PRN of the program to be RUN to handle this peripheral.

WORD 1 Address of B dedlock to which the peripheral will interrupt to transfer data.

WORD 2 bits $2^{Ø}$-2^{3} PRN of program requesting current transfer

bits 2^{4}-2^{7} Transfer Identifier of current transfer

bits 2^{8}-2^{12} Stop command for peripheral

bits 2^{13} = Ø/1 Echo/No Echo input device (Teletypes only)

bit 2^{14} = Ø/1 Input/Output device

bit 2^{15} = Ø/1 Free/Busy

WORD 3 Absolute address of next character in transfer

WORD 4 -(N-2) where N = no. of characters left to be transferred.

WORD 5 Termination character

If a user program is handling a peripheral it may use these locations in the same way as the executive I/O routines via the 'GET' and 'PUT' executive calls. The CNFG program (see Appendix D) allows information to be set into the control block at configuring time. The following rough rules apply when writing a program to handle a peripheral for a user PRN:

1. On entry compute address of calling programs Control Block from the calling PRN value. (See section 3.1).
2. Obtain users B register value and compute absolute address of READ/WRITE parameter area by adding in the start address of the Y or Z segment in which it is stored.
3. Obtain users parameters and validate them. If the transfer cannot be accepted place a reject qualifier in the users A register, decrement his P counter and return control to the calling PRN by means of the REVIVE1 executive call. (See section 3.4.4).
4. Otherwise start the transfer and increment the I/O count in the PSW. This should be done with interrupts off. Then return control to the accept branch by means of REVIVE1.
5. When the transfer has completed the I/O count should be decremented (again with interrupts off) and the I/O identifier should be added to the PSW.
6. If the user is active or suspended on a non I/O process (bit 15 or bit 11 of PSW set) then nothing further must be done to the user PRN.
7. If the user PRN is suspended on I/O (bits $2^{14}-2^{12}$ non zero) then it must be set active (set bit 15 of PSW) and the I/O identifier sum should be removed from the PSW and placed in

the users A register. Control should then be passed to the MISER scheduler via 'DROPOUT' or 'TERM'. This completes the handling of a single transfer.

APPENDIX G

NOTES ON PREPARING NEW VERSIONS OF MISER

There are 3 main components of MISER:

1. The KERNAL

This is an assembly language program written for the LAS5 assembler which contains all the special state coding of MISER together with the IDLE program. It may be amended and re-assembled and the resulting loaderese tape overwritten on top of an existing MISER executive.

2. THE TINT PROGRAM

This normal state program is written in CLSD. It may be amended and recompiled to produce an assembly language version. The assembly language version can be assembled with the ASS4 assembler and tested like a normal program. When it is ready to be incorporated into the executive it must be assembled with the LAS5 assembler together with a steering tape holding XBASE,YBASE and ZBASE values. With the current version this tape has 3 statements as follows.

```
XBASE  6;
YBASE  8;
ZBASE  8;
**
```

The resulting loaderese tape is used to overwrite the existing version.

3. THE LOAD PROGRAM

This is an assembly language program. It may be amended and tested as a normal program and then assembled to produce a loaderese tape in the same way as for the TINT program. The steering tape segment bases are:

```
XBASE  10;
YBASE  12;
ZBASE   5;
```

Care must be taken to ensure that modifications do not cause new coding to overwrite store occupied by other routines.