AAEC/E497

AAEC/E497

(2)

AUSTRALIAN ATOMIC ENERGY COMMISSION

RESEARCH ESTABLISHMENT

LUCAS HEIGHTS

REMOTE TERMINAL ACCESS TO AAEC COMPUTER NETWORK FACILITIES

by

P.L. SANGER

July 1980

ISBN 0 642 59692 1

AUSTRALIAN ATOMIC ENERGY COMMISSION
RESEARCH ESTABLISHMENT
LUCAS HEIGHTS

REMOTE TERMINAL ACCESS TO AAEC COMPUTER NETWORK FACILITIES

by

P.L. SANGER

ABSTRACT

Installation of a MICRONOVA computer and four automatic answering modem lines supported by the MICRONET system has significantly improved remote terminal access to AAEC computer network facilities. This report describes the development of the MICRONET systems and the use of a novel combination of Dataway sequences to provide ACL-NOVA and non-ACL mode support.

National Library of Australia card number and ISBN 0 642 59692 1

CONTENTS

1. INTRODUCTION

Remote terminal access to the Australian Atomic Energy Commission's computer network facilities at Lucas Heights was first provided by means of a single, manual-answer modem connection to the Commission's NOVA 820 computer. The connection was supported by the Dataway Terminal Communication System (DATERCOM) [Sanger 1976, 1977]. It was used on a sharing basis by DECwriter terminals at the Coogee and Mascot Offices of the Commission and at the New South Wales Institute of Technology, by video display units at the AAEC Research Establishment Library and the Alexander Mackie College, Sydney, and by a number of portable acoustically-coupled terminals at remote locations.

To reduce competition for the existing line, to provide twenty-four hour per day access without the need for operator intervention, and to keep within the limited funds available at the time (< $5000), a 4K word MICRONOVA computer was purchased with four automatic answering modem connections. The modems were subsequently installed in the computer building at Lucas Heights by Telecom Australia with phone numbers allocated to make use of a rotary number system in which only the master number is called to get the next of four available lines. The MICRONOVA computer was then interfaced to the AAEC Dataway, completing the hardware phase of the project.

Software for the MICRONOVA computer was limited by the small amount of memory that was purchased, consequently the 10K word DATERCOM system could not be used to support the new modem connections. Instead, a new 3K word system called MICRONET was developed from DATERCOM to provide ACL-NOVA support by using Dataway communication with the NOVA 820 computer, and to provide full access to other network resources by using direct Dataway communication with the other nodes.

2. THE MICRONOVA COMPUTER

2.1 Hardware

The MICRONOVA computer consists of 4K 16-bit words of dynamic MOS read/write memory, a real time clock, five 4207 asynchronous interface boards and a 4210 general purpose interface board.

One of the asynchronous interface boards has the 4208 console debug option and is used to control the monitor teletype terminal. The other boards are used to operate the Telecom Australia Plan 40 automatic answering modem lines.

2.2 Dataway Interface

As the MICRONOVA input/output (I/O) bus is quite different to that of the NOVA computer series, a copy of the existing NOVA/Dataway interfaces could not be used to connect the MICRONOVA computer to the AAEC Dataway. However, the 4210 general purpose interface board provides a generalised front end to the MICRONOVA I/O bus, and was used as the basis for interfacing the computer to the Dataway [Reid 1979]. Accordingly, the Dataway instructions for the MICRONOVA computer are different to those for the NOVA computers; they are described in Appendix A.

2.3 Software Considerations

As the instruction set for the MICRONOVA computer contains the instruction set for the NOVA computer series, most of the code developed for the NOVA computers has been used as the basis of the software to be written for the MICRONOVA computer. This includes a locally modified binary loader program, Dataway test programs, the Dataway program loader, the Dataway dump program and the MICRONET system as discussed in Section 3.

Coding changes are mainly the result of different instruction operation codes for the I/O RESET and READ SWITCHES instructions, different Dataway instructions, a different way of operating the real time clock with its trap through location 2, the operation of the power fail circuitry to give an 'all 1's' device code to the interrupt acknowledge (INTA) instruction, and the availability of a last in/first out (LIFO) or push-down stack. The use of location 3 to trap the stack overflow condition, and location 47 (octal) for the TRAP instruction necessitated changes to the layout of page zero in the NOVA computer.

It was very tempting to consider extensive use of the new stack instructions to optimise the performance of the MICRONET system but, owing to development time constraints, it was used only to save and restore the four accumulators and location zero in the interrupt handlers (Appendix B).

2.4 Servicing the Modem Lines

To service the automatic answering modem lines, the MICRONET real time clock handler periodically reads the modem status register of each of the four lines (see Appendix C). If the ring indicator is set to one, then asserting the signal 'data terminal ready' will answer the call and allow communication to begin. The modem interface then functions in the same way as a standard terminal connected directly to the MICRONOVA computer.

If the break indicator is set, it indicates that the break character has been received by the modem interface; this can be used as a trigger to hang up the connection by clearing the 'data terminal ready' signal. In the MICRONET system, the END-SESSION condition is also used as a trigger to hang up the modem connection.

3. THE MICRONET SYSTEM

3.1 Design Considerations

With only 4K of dynamic MOS memory available on the MICRONOVA computer, it was impossible to use the existing 10K word DATERCOM system to support the new modem connections. One solution would have been to develop a new system that used the software simulation of restricted Dataway sequences [Sanger 1976], pseudo-remote teleprinter interface (RTI) communication, to call upon the DATERCOM system in the NOVA 820 computer to provide full access to the resources of the AAEC computer network. This approach would certainly have worked, for the scheme is used currently to support a video display unit attached to a PDP11 computer in one building, and a video display unit and two DECwriter terminals attached to a PDP11 computer in another. However, this method makes inefficient use of the AAEC Dataway by using single character data transfers for both ACL mode and non-ACL mode access to DATERCOM [Sanger 1977]. Also, non-ACL mode access under this scheme would require even further sharing of the main Dataway addresses of a NOVA 820 computer that is already heavily loaded.

An alternative solution was based on the premise that the bulk of the DATERCOM coding related to the ACL mode support, and consisted of the development of a system that not only provided ACL mode support by using pseudo-RTI communication with the NOVA 820 computer but also gave full access to the other resources of the AAEC computer network by using direct non-ACL mode communication to the other nodes of the network. The challenge, of course, was to fit this system into 4K words of memory.

3.2 Stage 1 - NETCOM

The first step consisted of removing the in-built ACL-NOVA support and modifying the RTI sequences so that they could be used to access the ACL-mode facilities provided by the DATERCOM system. This system, called NETCOM, used 3.7K words of memory and was tested by running it in the Commission's NOVA 1220 computer.

3.3 Stage 2 - MICROCOM

Memory requirements were reduced to 2.9K words by removing support for the serial multiple user terminals system SMUT [Ellis 1977]. This version, known as MICROCOM, was also checked out using the NOVA 1220 computer.

3.4 Stage 3 - MICRONET

Alteration of the machine-dependent parts of MICROCOM plus the addition of support for the automatic answering modem lines resulted in the generation of the 3K word MICRONET system for the MICRONOVA computer.

The machine-dependent changes resulted from the different Dataway instructions (Appendix A), the different way of operating the real time clock, and limited use of the LIFO stack (Appendix B). Details of the automatic answering modem support are given in Appendix C.

MICRONET was made available for general use on the 1st August 1979; it currently services fixed modem terminals at the New South Wales Institute of Technology, the Alexander Mackie College, the AAEC Research Establishment Library, the AAEC Mascot and Head Offices, and a number of acoustically-coupled portable terminals.

Instructions to remote terminal users in the use of the new automatic answering facilities are given in Appendix D, and operational details relevant to the smooth and efficient production use of the MICRONET system are discussed in Appendix E.

4. CONCLUSION

Installation of the MICRONOVA computer and development of the MICRONET system has provided efficient, low-cost, remote terminal access to the AAEC computer network facilities.

The availability of four automatic answering modem lines, connected using the rotary number system, provides the remote user with convenient twenty four hours per day access without the need for operator intervention, and has overcome problems associated with contention for the previous single manual answer modem line.

A novel combination of pseudo-RTI Dataway sequences to the NOVA 820 computer and direct Dataway sequences to the other nodes of the network has allowed the MICRONET system to provide full access to the resources of the AAEC computer network yet make efficient use of the AAEC Dataway.

The challenge of taking this design approach and still satisfying the 4K word memory requirements has obviously been met; this technique should prove to be useful for some of the other nodes of the AAEC computer network.

5. ACKNOWLEDGEMENTS

R.P. Backstrom wrote software to check some of the features of the MICRONOVA computer, including the functioning of the automatic answering modem interfaces and the console debug option. C.G. Laman speedily provided AAEC Dataway connections for the MICRONOVA computer, and D.J. Reid gave excellent service in developing the MICRONOVA-Dataway interface.

6. REFERENCES

Ellis, P.J. [1977] - SMUT - Serial Multiple-User Terminals System. AAEC/E433.

Jones, C.G. [1973] - Development of the NOVA-Dataway Interface. AAEC unpublished work.

Reid, D.J. [1979] - Development of the MICRONOVA-Dataway Interface. AAEC unpublished work.

Sanger, P.L. [1976] - The AAEC Dataway Terminal Communicatiom System. Proc. 7th Aust. Computer Conf., 2:610-622.

Sanger, P.L. .[1977] - Terminal Facilities Provided by the DATERCOM System. AAEC/E409.

Sanger, P.L., Jones, C.G. and Ellis, P.J. [1973] - Programming the NOVA Computer for Dataway Communication. AAEC/E268.

APPENDIX A
DATAWAY PROGRAMMING FOR THE MICRONOVA COMPUTER

A1. INTRODUCTION

The 4210 general purpose interface board provided a generalised front end to the MICRONOVA I/O bus, and was used as the basis to interface the computer to the AAEC Dataway [Reid 1979]. Use of the 4210 interface board greatly simplified the design of the MICRONOVA-Dataway interface, and allowed it to be developed in a fraction of the time that was required to develop the externally hardwired NOVA-Dataway interface [Jones 1973].

Only one device code (selected to be $(32)_8$ and referred to as DW in the instructions described below) can be used with the 4210 interface board; this limited the number of I/O instructions (DIA, DIB, DIC, DOA, DOB, DOC, NIO) and I/O signals (S, C, P) that could be used to program the MICRONOVA-Dataway interface compared to those available for the NOVA computers [Sanger et al. 1973]. Further, with the 4210 interface board being used to implement a data channel controller, the B and C registers are used as the memory address and word count registers respectively; thus the overall appearance of the MICRONOVA Dataway instructions is quite different to that for the NOVA computer.

The MICRONOVA Dataway instructions are more byte oriented having, for example, bits in the high order byte of the 16-bit accumulator (ac) in the DOA ac,DW instruction being used to indicate which byte of the 24-bit Dataway register is to be loaded.

A2. CONTROL SIGNALS

The I/O signals, S, C, P are used with the no I/O transfer instruction (NIO), to set up the raise address out (RAO, NIOS DW), the computer terminates (CTS, NIOC DW), and the raise end line (REL, NIOP DW) conditions respectively. The raise service line (RSL) condition is set up by having the high order bit (bit zero) of the accumulator set to one when the DOA ac, DW instruction is executed. The raise hold line (RHL) condition is set up by having the high order bit of the accumulator set to one when the DOB ac, DW instruction is executed.

A3. DATAWAY INTERRUPTS

Seven Dataway conditions can interrupt the MICRONOVA computer if priority line 8 (interrupt disable mask bit 8) is enabled; these are the same as those for the NOVA-Dataway interface. Again, device code $(32)_8$ is given in response to an interrupt acknowledge (INTA ac) instruction, but it is more efficient to execute the DIC ac, DW instruction and check if the high order bit of the accumulator is set to zero, as this instruction also reads the 3-bit Dataway interrupt code into bits 5, 6 and 7 of the accumulator. Execution of the DIC ac, DW instruction also clears the interrupt condition that was just read - there are no read interrupt register and clear interrupt register instructions as there were for the NOVA computers.

One special condition worth noting for the MICRONOVA computer is that 'powering up' can leave bits set in the Dataway interrupt register and all initialisation software should ensure that the register is cleared before attempting to handle any Dataway sequences. This is conveniently done by initially executing an I/O reset (IORST = DOAC 0,CPU) instruction followed by a loop to read and clear any spurious interrupts as follows:

```
START   DOAC   0,CPU      I/O Reset
LOOP    DIC    0,DW       Read interrupt register
        MOVL   0,0,SNC    skip if clear after power up
        JMP    LOOP       Br, check again
         .
         .
         .
```

A4. SKIP INSTRUCTIONS

The NOVA-Dataway interface skip instructions such as 'skip if Dataway error flag is zero (SKPDEZ)', could not be provided for the MICRONOVA computer, and the Dataway error flag condition must be detected by checking if bit 1 of the accumulator is set to one after executing the DIC ac,DW instruction.

A5. DATA TRANSFER INSTRUCTIONS

The MICRONOVA-Dataway input/output instructions are summarised below:

Load High Byte (DOA ac,DW)

If bit 5 of the accumulator is set to one, then the contents of the low order byte of the accumulator are loaded into the high order byte of the MICRONOVA-Dataway interface buffer register (B0-B7).

Load Middle Byte (DOA ac,DW)

If bit 6 of the accumulator is set to one then the contents of the low order byte of the accumulator are loaded into the middle byte of the buffer register (B8-B15).

Load Low Byte (DOA ac,DW)

If bit 7 of the accumulator is set to one, then the contents of the low order byte of the accumulator are loaded into the low order byte of the buffer register (B16-B23).

Set Word Count (DOC ac,DW)

The contents of the accumulator are loaded into the word count register. A negative word count must be specified.

Set Current Address (DOB ac,DW)

The contents of the accumulator are loaded into the current address register. If bit zero of the accumulator is set to one, then the interface will raise the 'hold in' (raise hold line) in response to a request for a data transfer sequence.

Read High Byte and Status (DIC ac,DW)

The contents of the high order byte of the data register (D0-D7) are read into the low order byte of the accumulator. The high order byte is set to contain the current state of the interrupt register and the Dataway error flag - bit zero set to zero if there is a Dataway interrupt, bit 1 set to one if there is a Dataway error, bits 2-4 always set to one, and bits 5-7 contain the interrupt code. Execution of this instruction also clears the interrupt request that was just read.

Read Low Data (DIA ac,DW)

The contents of the middle and low bytes of the data register (D8-D23) are read into the accumulator.

Read Current Address (DIB ac,DW)

The contents of the current address register are read into the accumulator.

A6. CONTROL INSTRUCTIONS

Raise Address Out (NIOS DW)

A 'start' pulse is sent to the interface to begin a data transfer sequence (raise address out) if the Dataway is not busy.

Raise Hold Line (DOB ac,DW)

If bit zero of the accumulator is set to one, then the interface will raise the 'hold in' (raise hold line) in response to a request for a data transfer sequence.

Raise Service Line (DOA ac,DW)

If bit zero of the accumulator is set to one then the interface will raise the appropriate service line - this instruction can be used to continue a data transfer sequence, but is normally used for RTI operations.

Raise End Line (NIOP DW)

A 'P' pulse is sent to the interface to raise the appropriate end line.

Computer Terminates (NIOC DW)

A 'clear' pulse is sent to the interface to reset the 'address in' or 'hold out' lines to complete the Dataway ending sequence.

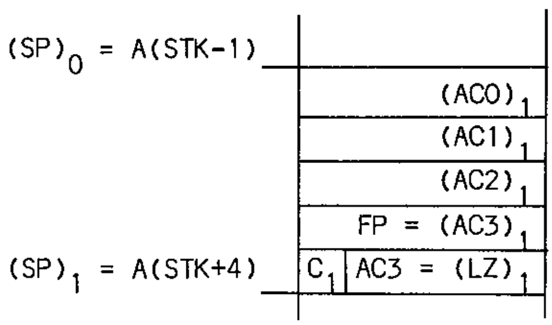
APPENDIX B
USE OF MICRONOVA STACK OPERATIONS FOR MICRONET INTERRUPT HANDLERS

It was very tempting to consider extensive use of the MICRONOVA stack operations (MICRONOVA programmers reference manual, DGC 015-000050) to optimise the performance of the MICRONET system. However, because of the development time constraints, the stack was only used for saving and restoring the four accumulators and location zero in the interrupt handler for two levels of interrupt. On the other hand, this gave rise to an efficient use of the SAV and RET instructions in the interrupt handler as described below.

MICRONET initially sets the stack pointer (SP) to the beginning address of the stack area minus one - $(SP)_0$ = A(STK-1). When an interrupt is taken, accumulator three is loaded into the frame pointer (FP), location zero (LZ) is loaded into accumulator three and the SAV instruction is executed, as follows:

```
INTRP   MTFP   3      FP=(AC3)1
        LDA    3,0    AC3=(LZ)1
        SAV           FP=A(STK+4),AC3=A(STK+4)
         .
         .
         .
         .
```

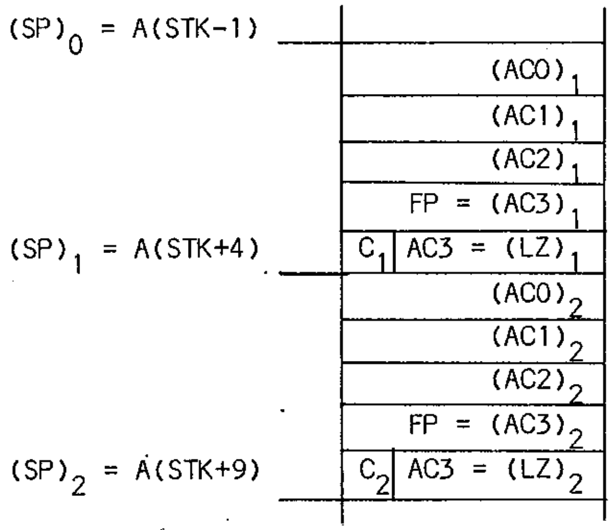
The return block 'pushed' onto the stack, assuming this to be a first level interrupt (subscript 1), is as follows:



Further, assuming that the interrupt handler detects that the first interrupt corresponded to an input character to be checked for validity, then

MICRONET turns the interrupt back on (INTEN) and gives control to the syntax analyser.

An interrupt at this point causes a second return block to be pushed onto the stack as follows,



with FP and AC3 being reset to the new $(SP)_2$ after execution of the SAV instruction.

With the syntax analyser already in use, there is a normal return from the interrupt handler with the interrupt being turned back on after execution of the RET instruction as follows:

```
EXIT    INTEN
        RET     SP=FP=A(STK+9),C=C2,
                PC=(LZ)2,AC3=(AC3)2,
                AC2=(AC2)2,AC1=(AC1)2,
                AC0=(AC0)2,SP=(SP)1=A(STK+4)
```

The RET instructions 'pops' the second return block from the stack to restore the accumulators and carry, and effectively executes an indirect jump through location zero by setting the program counter (PC) to contain the contents of location zero. The stack pointer is reset to point to the end of the first return block SP = $(SP)_1$.

When syntax analysis is complete, control is returned to the background program by turning the interrupt off, resetting FP to the current SP value before executing the RET instruction and turning the interrupt on as follows:

MFSP	3	$AC3=(SP)_1 = A(STK+4)$
MTFP	3	$FP = A(STK+4)$
INTEN		
RET		$SP=FP=A(STK+4), C=C_1,$ $PC=(LZ)_1, AC3=(AC3)_1,$ $AC2=(AC2)_1, AC1=(AC1)_1,$ $AC0=(AC0)_1, SP=(SP)_0=A(STK-1).$

APPENDIX C
SERVICING THE AUTOMATIC ANSWERING MODEM LINES

To service the automatic answering modem lines, the MICRONET real time clock handler periodically reads the modem status register of each of the four lines using the DIB ac,MODEM instruction.

If the ring indicator, i.e. low order bit 14 of the accumulator, is set to one, the call is answered by asserting the signal 'data terminal ready', by issuing the DOB ac,MODEM instruction with the low order bit 15 of the accumulator set to one. This allows communication to begin and the modem interface then functions in the same way as a standard directly connected terminal to the MICRONOVA computer.

If the break indicator is set, it indicates that the break character has been received by the modem interface; this is then treated as a cntrl/P (END-TASK or END-SESSION) request for an active user. The break indicator is cleared by issuing the DOB ac,MODEM instruction with low order bit 15 of the accumulator set to one to leave the data terminal ready until the cntrl/P request has been processed.

When the END-SESSION condition is completed in the MICRONET system, the modem connection is automatically hung up by clearing the 'data terminal ready' signal, i.e. by issuing the DOB ac,MODEM instruction with the low order bit 15 of the accumulator set to zero.

APPENDIX D

INSTRUCTIONS FOR USE OF AUTOMATIC ANSWERING FACILITIES OF THE AAEC COMPUTER NETWORK (CURRENT FROM AUGUST 1979)

D1. REMOTE TERMINAL USERS

1. Please note the change in the telephone number to be used to gain access to AAEC computer network facilities (the old number will be retained for a time to provide backup for the new system).

2. The new number gives access to the current automatic-answering line out of four that are available for use (the busy signal indicates that all four lines are in use).

3. When the call is automatically answered, a carrier signal tone can be heard; at this point the modem connection button should be pressed and the handpiece then put back in its normal position (the order is important or the line will be disconnected before a session can be established) or, for acoustic-coupled devices, the handpiece should be placed in the required position on the terminal.

4. Once the carrier signal is present (usually an indicator light is illuminated), the space character should be entered; this results in the colon character (:) being printed on a new line.

5. A password should now be entered (with the asterisk character (*) being printed for each password character entered) followed by the carriage return (RETURN or CR) character.

6. The message ID: is now printed requesting that the user account number be specified (with the asterisk character being printed for each user account number character entered) followed by carriage return.

7. When the user account number is validated, the dollar character ($) prompt is printed on a new line.

8. User specification of the required non-ACL mode task is now requested (such as N for access to the central computer Logon facility), or if the BEL(CNTRL/G) character is entered, then ACL mode (ACL-NOVA system) access is obtained.

9. When work within a session is completed, a carriage return response should be given to the dollar character ($) prompt; this results in the END-SESSION message being printed at the terminal <u>and</u> the telephone connection being hung up.

10. The user should now release the modem connection button or, for acoustic-coupled devices, return the handpiece to its normal position on the telephone.

<u>NOTE: Changes in the New System</u>

(a) No manual intervention or communication with the operator is required to answer the telephone call or to free the modem connection at the computer centre, therefore access is available twenty four hours per day (subject to availability of the various resources of the AAEC computer network).

(b) The password and user account number must be given before ACL mode access is made available.

(c) The telephone connection is automatically disconnected at END-SESSION time, and a new telephone call must be made to begin a new session.

APPENDIX E
OPERATIONAL DETAILS FOR THE MICRONET SYSTEM

Operational details relevant to the smooth and efficient use of the MICRONET system are discussed below.

E1. LOADING THE MICRONOVA COMPUTER AFTER A 'POWER OFF' SITUATION

The MICRONOVA computer has 4K MOS memory whose contents are lost when the power is turned off. To load the MICRONOVA computer in this case, the following steps should be followed:

(i) 'Power up', insert key and turn it from reset to run position to give control to the console debug option (this will give an exclamation mark (!) prompt at the monitor teletype, assuming it is 'powered on' and on-line).

(ii) Enter the characters 10L to have the bootstrap loader program automatically loaded into low core and started.

(iii) The MICRONOVA binary loader tape can now be read via the teletype paper tape reader. Control is automatically given back to the console debug option when the last frame of the binary loader tape has been read.

(iv) Enter the characters 7777R to start the binary loader program to allow it to read object programs via the teletype paper tape reader.

(v) Load the MICRONOVA Dataway program loader (set up for Dataway address X(58)) via the paper tape reader, and the message CALL will subsequently be printed on the monitor teletype.

(vi) Load the MICZERO program first to clear memory and then load MICRONET to provide access to AAEC computer network facilities.

E2. RELOAD CASE WITH POWER UP AND BINARY LOADER AND DATAWAY PROGRAM LOADER INTACT

(i) Insert key, move it back to the reset position, then forward to the run position to give control to the console debug option (this will given an exclamation mark (!) prompt at the monitor teletype).

(ii) Enter the characters 7200R to start the Dataway program loader, and the message CALL will be printed on the teletype.

(iii) Load the MICRONOVA Dataway dump program, MICDUMP, that will automatically dump the 4K contents of the MICRONOVA computer and then given control back to the console debug option.

(iv) Enter the characters 7200R again to start the Dataway program loader, and then use it to load the MICZERO program, to clear memory and, finally, to load the MICRONET program.

E3. DUMPING THE CONTENTS OF THE MICRONOVA COMPUTER

To dump the contents of the 4K memory of the MICRONOVA computer, follow steps (i) to (iii) of E2 above. Step (iv) of E2 should then be used to reload the MICRONOVA computer.

E4. ENTERING THE SYSTEM MESSAGE OR SYSTEM PASSWORD

The system message or system password may be entered at the MICRONOVA monitor teletype by entering an immediate hash colon (#:) or hash asterisk (#*) command (at the same level at which the space character is entered to bring up the ID: prompt - this is different to the DATERCOM case where these commands are entered under the ACL-NOVA system).

E5. IDENTIFYING TERMINALS LOGGED ONTO THE MICRONET SYSTEM

Terminals logged onto the MICRONET system (MICRONOVA computer) can be identified by using the NOVA task to interrogate the Dataway addresses X(59) to X(5B); that is 159 to 15B.