AAEC/E262

AUSTRALIAN ATOMIC ENERGY COMMISSION

RESEARCH ESTABLISHMENT
LUCAS HEIGHTS

IBM360 AND NOVA SOFTWARE DEVELOPED TO GIVE THE NOVA COMPUTER ACCESS TO THE RESOURCES OF THE IBM360 COMPUTER

by

P.L. SANGER
R.P. BACKSTROM

June 1973

ISBN 0 642 99573 7

AUSTRALIAN ATOMIC ENERGY COMMISSION

RESEARCH ESTABLISHMENT

LUCAS HEIGHTS

IBM360 AND NOVA SOFTWARE DEVELOPED TO GIVE THE NOVA COMPUTER ACCESS TO THE RESOURCES OF THE IBM360 COMPUTER

by

P.L. SANGER

R.P. BACKSTROM

ABSTRACT

IBM360 and NOVA software to allow NOVA programs stored as load modules on IBM360 disk storage to be loaded directly into the NOVA computer via the Dataway, and to allow the contents of NOVA core storage to be sent to the IBM360 computer and listed on the IBM1403 line printer, is described. This software can be used as the basis for the development of software to allow other Dataway computers to access the IBM360 computer. A scheme for adding interactive programs to the normal IBM360 Jobstream by writing the necessary JCL and SYSIN data to the HASP internal reader is also presented.

National Library of Australia card number and ISBN 0 642 99573 7

The following descriptors have been selected from the INIS Thesaurus to describe the subject content of this report for information retrieval purposes. For further details please refer to IAEA–INIS–12 (INIS: Manual for Indexing) and IAEA–INIS–13 (INIS: Thesaurus) published in Vienna by the International Atomic Energy Agency.

A CODES; DATA TRANSMISSION; IBM COMPUTERS; MAGNETIC DISKS; N CODES

CONTENTS

1. INTRODUCTION

The first computer on the AAEC Dataway was the NOVA computer which was connected via a NOVA-DATAWAY interface (C.G. Jones, AAEC Report to be published) and a Dataway Control Unit (C.G. Jones, AAEC Report to be published). The PDP9L computer was the next to be connected via a PDP9L-DATAWAY interface (C.G. Jones, AAEC Report to be published) and a Dataway Control Unit. Since the PDP9L computer is already attached to Selector Channel 2 of the IBM360 computer via an AAEC designed (Richardson 1970) and DEC built Link, completion of the PDP9L-NOVA Dataway link provided the path for NOVA access to the IBM360 computer.

The PDP9L computer essentially acts as the 'telephone exchange' for all Dataway communication with the IBM360 computer, and special software has been written for the PDP9L computer to control all AAEC network communications with the IBM360 computer (Richardson 1973). The PDP9L computer and all other computers on the Dataway appear to the IBM360 computer as 7-Track Magnetic Tape Control Units. The IBM360 Attention handling software conventions developed using the PDP9L (Richardson 1973) apply to all of the Dataway computers; this is discussed in Section 2.1. In the NOVA computer, software must be provided to fit in with the conventions of the IBM360 software; this is described in Section 3.

2. IBM360 SOFTWARE

2.1 Signalling Attention to the IBM360 Computer

To begin communicating with the IBM360 computer, the NOVA computer must carry out a Primary Write (Command Code X'05') sequence to the PDP9L computer via the Dataway. The PDP9L computer accepts 24 bytes from the NOVA, marks the NOVA address X'48' as being Ready and then presents Attention status to the IBM360 computer from address X'248'.

On receipt of the unsolicited interrupt from the device address X'248', the IBM360 Operating System† passes control to a 1st level Attention Handler AEINTATN which notes the Attention status and executes a special Supervisor Call. This routine performs a Diagnostic Read (Command Code X'06') of 24 bytes from the PDP9L computer to obtain the data sent by the NOVA computer. A check is next carried out on the 24 bytes and, if the first byte is X'FF', then control is given to a global program specified by the next 9 bytes of data. If the first byte was zero, then control is given to the program

---

† At the time of writing, Release 21.6 of the MVT version of OS/360 plus HASP (Houston Automatic Spooling System) version 3.1 was being used.

AEATR48. The last alternative applies when the first byte is not zero and not X'FF', in which case control is given to the AEATN48 program.

The point to be noted in these IBM360 Attention handling software conventions (Richardson 1973) is that when the first byte is X'FF', control is given to a program that can be used by all computers on the Dataway regardless of their address; otherwise, control is given to programs AEATN-- or AEATR-- that apply to a specific Dataway computer and Dataway address.

Sections 2.2 and 2.3 describe how the routines AEATN48 and AEATR48 have been used to allow the NOVA computer to access the resources of the IBM360 computer.

2.2 The AEATN48 Routine

2.2.1 General discussion

The AEATN48 routine allows NOVA programs that are stored in load module form as members of the partitioned data set AMC.NOVALIB to be sent to the NOVA computer. The first eight of the 24 bytes sent by the NOVA computer are translated from ASCII code to EBCDIC code and specifies the particular member of AMC.NOVALIB that is required. The next two bytes indicate whether the NOVA program is first to be relocated or not, as discussed in Section 2.2.2.

The NOVA program is loaded into IBM360 core storage using the DD-less I/O routine AELOAD (Richardson 1973) and is relocated if necessary. A series of EXCP (Execute Channel Program) operations is then carried out to send:

(i) a 24 byte Primary Write to the NOVA containing the Word Count (WC), Starting Address (CA) and Entry Point (EP) for the NOVA program,

(ii) a series of 800-byte Secondary Writes (Command Code X'09') containing the NOVA program

and (iii) a Rewind/Unload command (Command Code X'0F') to complete the program transfer operation.

2.2.2 Load module format and program relocation

Load module members of AMC.NOVALIB are created by passing control to the NOVAOBJ program (R.P. Backstrom, AAEC unpublished report) after using NOVASM (Sanger 1970) to assemble the NOVA program, and then executing the NOVALKED procedure. The load module format is shown in Figure 1. The first two bytes X'FF75' are used to indicate that the load module is a NOVA program and causes a system dump to occur if any attempt is made to use it as an IBM360 program.

The next two bytes contain an RLD count which is used for program relocation and indicates the number of locations in the NOVA program that must be relocated (X'FFFF' indicates that the module is not relocatable). For each location to be relocated, a two-byte RLD item appears at the end of the load

module as shown in Figure 1. Only relocation factors of $\pm$ 1 are allowed and bit zero of each RLD item is used to indicate if positive or negative relocation is required (zero for + ve relocation, one for - ve relocation).

Bytes 4 and 5 of the load module contain the number of words in the NOVA program. The next two bytes contain the Starting Address (CA) where the first word of the program is to be stored in the NOVA computer. Bytes 8 and 9 contain the program entry point. Bytes 10-27 are reserved for future use, while bytes 4-27 are used to set up the Primary Write for the NOVA computer. Bytes 28 onwards contain the NOVA program packed two 16-bit NOVA words per 32-bit IBM360 word. The NOVA text is followed by the RLD items.

Bytes 8 and 9 of the 24 bytes sent from the NOVA computer can be used to indicate the Starting Address required for the NOVA program (X'0000' indicates that the starting address in the load module should be used). If this value of CA is different from the value contained in the load module, then the appropriate relocation factor is calculated and the program is relocated in IBM360 core storage by using the RLD items at the end of the load module. The new starting address is stored in bytes 6 and 7 of the copy of the load module in IBM360 core storage, and the modified 24 bytes and relocated program are then sent to the NOVA computer.

2.3 The AEATR48 Routine

2.3.1 General discussion

The AEATR48 routine allows IBM360 programs associated with the NOVA computer to be added to the normal IBM360 Jobstream by writing the necessary Job Control Language (JCL) and data (SYSIN data) to the HASP internal reader. AEATR48 first of all checks for the presence of HASP and then examines the second byte of the 24 bytes sent from the NOVA to see if a dump of the contents of NOVA core storage is to be produced on the IBM1403 line printer, or if an interactive program is to be added to the Jobstream.

2.3.2 The AENVDUMP program

When the second byte of the 24 bytes sent from the NOVA is X'01' or X'02', this indicates that the contents of NOVA core storage are to be listed on the IBM1403 line printer in hexadecimal or octal respectively. Bytes 2 and 3 of the 24 bytes contain the number of words to be listed, while the next two bytes indicate the starting address for the dump.

In this case, the AEATR48 routine writes the records

```
// AENVDUMP JOB (CIF90072, NOVA), DATAWAY. USE,
//          CLASS = F, TIME=15, PRTY=9
```

// NOVA EXEC AENVDUMP, PARM = {HEX / OCT}

to the HASP internal reader. An EXCP sequence is then used to Read (Command Code X'02') 512 bytes from the NOVA computer. On completion of this Read, the NOVA data is written to the HASP internal reader as eight 80-byte records, each record having the first two bytes blank (to distinguish the records from JCL), the next 64 bytes containing the NOVA data and the remaining 14 bytes blank.

The EXCP Read sequence followed by writing to the HASP internal reader is continued until the NOVA sends immediate Unit Exception (UE) Status to the Read sequence. The AEATR48 routine then issues a Rewind/Unload command to terminate communication with the NOVA, and writes the last record /*EOF to the HASP internal reader.

The AENVDUMP program is subsequently executed under the normal Jobstream and produces a listing of the contents of NOVA core storage on the IBM1403 line printer.

2.3.3 Interactive programs

Setting the second byte of the 24 bytes sent from the NOVA to X'03' indicates that an interactive program is to be added to the IBM360 Jobstream. Bytes 2-23 of the 24 bytes are translated from ASCII code to EBCDIC code and are matched against entries in an authorised procedure list. If the 22-byte character string appears in the authorised procedure list, then a corresponding procedure name is taken from the list and added to the JCL that is then written to the HASP internal reader. The first byte of the authorised procedure list also indicates whether SYSIN data is required to follow the JCL or not.

For example, if the characters DISPLAY WTRC are sent from the NOVA, then the procedure name AENVPLT2 is added to the JCL and the records

```
// AENVPLT2   JOB   (CIF90072, NOVA), DATAWAY. USE,
//            CLASS=F, TIME=15, PRTY=9
// NOVA       EXEC  AENVPLT2
```

are written to the HASP internal reader.

Once all the necessary information has been written to the HASP internal reader, the interactive program subsequently runs under the normal Jobstream. This X'03' option has been used to run interactive display programs (Backstrom and Sanger 1973) and provides a very general interface between the NOVA and IBM360 computers.

2.3.4 Need for a time-sharing monitor

The above scheme of writing jobs to the HASP internal reader and running them under the normal Jobstream is not the best way of carrying out interactive computing from Dataway computers. Even though the CLASS=F queues are normally quite short, there is normally some delay in getting the job started; but once the program is executing the response from the IBM360 computer is very good. Interactive jobs run in this manner tie up one of the IBM360 initiators and require a minimum region of 60K, when they require only short bursts of IBM360 CPU activity over a considerable period of time.

The thought of running more than one interactive program in this way brings one to the realisation that such jobs should be run under a proper time-sharing monitor such as the TSO option of the IBM360 Operating System. However, there are a number of modifications that would have to be made to a system such as TSO to allow it to handle Dataway communications. Until such modifications can be carried out and a proper time-sharing system implemented, the above scheme does provide a convenient way of interacting with the IBM360 computer and assessing the advantages of the interaction.

3. NOVA SOFTWARE

3.1 Introduction

The first use made of the NOVA computer's access to the IBM360 computer was to load NOVA programs from IBM360 disk storage into the NOVA computer. Using this facility, programs can be loaded into the NOVA computer in a matter of seconds (it takes approximately 1½ seconds to load the 7K word ACL-NOVA system) and the NOVA software required for this purpose is described in Section 3.2.

The second application of the access to the IBM360 computer allows the contents of NOVA core storage to be dumped onto the IBM1403 line printer in an octal or hexadecimal representation as discussed in Section 3.3.

Combined use of these two software packages has greatly improved the flexibility of the NOVA system and has provided a powerful aid to program development on the NOVA computer. The software developed for these two applications can be used as a basis for software developed for other Dataway computers and for this reason, schematic diagrams of the required Dataway sequences and listings of the NOVA software are included in this report.

3.2 Loading the NOVA Computer from IBM360 Disk Storage

3.2.1 The NOVALOAD program

Loading programs from IBM360 disk storage into the NOVA computer begins

with the execution of the NOVALOAD program. The message CALL ␣† is printed on a new line at the monitor teletypewriter terminal attached to the NOVA computer indicating that a program name of up to eight characters may be specified.

Once the required program name followed by a Carriage Return (CR) has been typed at the terminal,†† then a 24-byte Primary Write is sent to the PDP9L computer with the first eight of the 24 bytes containing the program name and the next two bytes zero. The Dataway sequence for the Primary Write to the PDP9L is shown schematically in Figure 2. Figure 2(a) shows the normal Primary Write sequence, while Figure 2(b) shows the case where a previous Primary Write request was still pending in the PDP9L computer. In this second case, the PDP9L immediately sends Busy status to the NOVA computer. If the NOVA replies with Unit Check (UC) status, the pending request will be cancelled in the PDP9L. If the NOVA replies with Attention status, the pending request would be reactivated. The NOVALOAD program replies with UC status if Busy status is received during the Primary Write sequence to the PDP9L, and repeats the Primary Write to re-present the new request.

On receipt of the Primary Write from the NOVA, the PDP9L computer signals Attention to the IBM360 computer and marks the NOVA address as being Ready. The IBM360 computer performs a Diagnostic Read (Section 2.1) to the PDP9L to obtain the 24 bytes sent by the NOVA. Since a program name cannot start with X'00' or X'FF', the IBM360 routine AEATN48 is given control. AEATN48 loads the required NOVA program from the partitioned data set AMC.NOVALIB into IBM360 core storage and sends a 24-byte Primary Write to the NOVA containing the size, starting address and entry point of the NOVA program.

When the NOVA computer receives the Primary Write from the PDP9L computer, a check is made that the program will not overwrite NOVALOAD (WC + CA $\leqslant$ CA (for NOVALOAD)), and that the entry point is within the program (CA $\leqslant$ EP < CA + WC) or is zero. The NOVALOAD program then waits for a series of 800-byte Secondary Writes from the PDP9L computer.

The IBM360 computer now sends a series of 800-byte Secondary Writes containing the NOVA program to the NOVA computer. At the end of the program transfer a Rewind/Unload command is sent to the NOVA computer.

† ␣ represents space or blank in this report

†† If an error is made while entering the program name, typing ! will automatically restart the program.

When the NOVA computer receives the Rewind/Unload from the PDP9L computer, control is given to the new program (the computer executes a HALT instruction if a zero entry point is specified). Dataway sequences for Rewind/Unload from the PDP9L are shown schematically in Figure 4. The NOVALOAD program gives UC status in reply to Rewind/Unload at the end of the program transfer sequence so that the NOVA address is marked as Not Ready in the PDP9L computer. The only other Dataway sequence to be considered is the case of a Test I/O (TIO) command from the PDP9L; this is shown schematically in Figure 5.

The schematic diagrams of Dataway sequences shown in Figures 2 to 5 proved to be very useful for the writing of the NOVALOAD program since they show very clearly the various states that apply when each particular Dataway interrupt occurs. The NOVALOAD program works with Interrupt Off and uses a skip condition to test for Dataway Interrupts (Sanger, Jones and Ellis, 1973). A listing of the NOVALOAD program is given in Appendix A.

3.2.2 The PLSBDPL program

When the ACL-NOVA system (version ACLNOVA.RE72168) is being used on the NOVA computer only the 122 words between the end of the 5K ACL-NOVA work area and the start of the AAEC binary loader are spare (X'2EF4' to X'2F6E'). However, the NOVALOAD program is 214 words long, so a special version of NOVALOAD, called PLSBDPL, was written.

PLSBDPL was set up specifically to load the NOVALOAD program into the NOVA computer, and it was assumed that NOVALOAD was shorter than 400 words (so that only one Secondary Write is required) and that its starting address could be used as its entry point. The resulting PLSBDPL program is 116 words long and can now remain resident while the ACL-NOVA system is operating. Another advantage is that the program can be quickly loaded into the NOVA computer from a very short paper tape. A listing of the PLSBDPL program is given in Appendix B.

To load programs from IBM360 disk storage into the NOVA computer, the PLSBDPL program should first be executed by starting it at address X'2F00'. The NOVALOAD program is then loaded into the NOVA computer and started automatically at its entry point X'2E00'. The message CALL ƀ is printed on the monitor terminal and the required NOVA program can now be specified and loaded as described in Section 3.2.1. If additional NOVA programs are to be loaded, the NOVALOAD program should simply be restarted at the address X'2E00'.

3.3 Dumping the Contents of NOVA Core Storage on the IBM1403 Printer

Execution of the NOVADUMP program (starting address X'2B00') allows a

dump of the contents of NOVA core storage to be produced on the IBM1403 line printer. This program begins by sending a 24-byte Primary Write to the PDP9L computer with the first two bytes containing X'0001' or X'0002' to indicate that the dump is to be in hexadecimal or octal representation respectively, and with the next four bytes containing the word count and starting address for the dump. The parameters X'0001' (hexadecimal format), X'3000' (12K words) and X'0000' (starting address zero) have been assembled into the NOVADUMP program to obtain a full 12K hexadecimal dump of NOVA core storage, but they can be altered manually by changing the values of locations X'2B70', X'2B71', and X'2B72' of the program.

On receipt of the Primary Write from the NOVA, the PDP9L computer signals Attention to the IBM360 computer and marks the NOVA address as being Ready. The IBM360 computer performs a Diagnostic Read to the PDP9L to obtain the 24 bytes sent by the NOVA and, since the first byte is X'00', control is given to the IBM360 routine AEATR48. The AEATR48 routine then issues a series of 512-byte Reads to the NOVA computer to obtain the contents of NOVA core storage, and writes the JCL necessary to execute the AENVDUMP program (see Section 2.3.2) to the HASP internal reader followed by the contents of NOVA core storage as SYSIN data.

When the NOVA computer has sent all of the dump data to the IBM360 computer, Unit Exception status is given as immediate status in reply to the next Read request from the PDP9L computer (this is the status used as an end of file indication on the IBM360 computer). The Dataway sequences for Reads from the PDP9L computer are shown schematically in Figure 6. Figure 6(a) shows the responses for normal Read sequences, while Figure 6(b) shows the case where UE status is given as immediate status to the PDP9L computer.

On receipt of the UE status, the AEATR48 routine sends an end-of-file indication to the HASP internal reader and sends a Rewind/Unload to the NOVA computer.

The NOVADUMP program gives UC status to this final Rewind/Unload and executes a HALT instruction to indicate that the contents of NOVA core storage have been sent to the IBM360 computer (the NOVA address is now marked as Not Ready in the PDP9L computer). The schematic diagrams of the Dataway sequences shown in Figures 2,4, 5 and 6 proved to be very useful in the writing of the NOVADUMP program. A listing of the NOVADUMP program is given in Appendix C.

Subsequent execution of the IBM360 program AENVDUMP using the information written to the HASP internal reader produces the required hexadecimal or octal dump on the IBM1403 line printer.

4. CONCLUSIONS

The IBM360 and NOVA software described here shows how the NOVA computer can access the resources of the IBM360 computer. The NOVALOAD and NOVADUMP applications are just the first two uses made of this access, yet they have already improved the flexibility of the NOVA system and provided a powerful aid to program development on the NOVA computer.

It would be reasonably straightforward to adapt the software described here to apply to other computers on the Dataway. This software can thus be used as the basis for the development of software to allow other Dataway computers to access the IBM360 computer.

Until a proper time-sharing monitor can be set up for running a number of interactive jobs on the IBM360 computer, the scheme of adding interactive jobs to the normal Jobstream by writing the necessary JCL and SYSIN data to the HASP internal reader provides a convenient way of interacting with the IBM360 computer and assessing the advantages of the interaction.

5. ACKNOWLEDGEMENTS

The authors thank Dr. D.J. Richardson for valuable discussions and advice throughout this work particularly on the IBM360 Attention handling software and Dataway signalling conventions. The authors also record their appreciation of the work done by Mr. P.J. Ellis, Mr. C.G. Jones and Mr. C.G. Laman in setting up the Dataway hardware.

6. REFERENCES

Backstrom, R.P. and Sanger, P.L. (1973)- IBM360 and NOVA Software Developed to allow Plotter Output to be Displayed on the Tektronix T4002 Graphical Display Terminal. AAEC/E263.

Richardson, D.J. (1970) - The AAEC Computer Network. AAEC/TM576.

Richardson, D.J. (1973) - Signalling Conventions for the AAEC Computer Network. AAEC/E264.

Sanger, P.L. (1970) - NOVASM and NOVASIM - An Assembler and a Simulator for the NOVA and SUPERNOVA Computers, Written to run on an IBM360 Computer. AAEC/TM566.

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

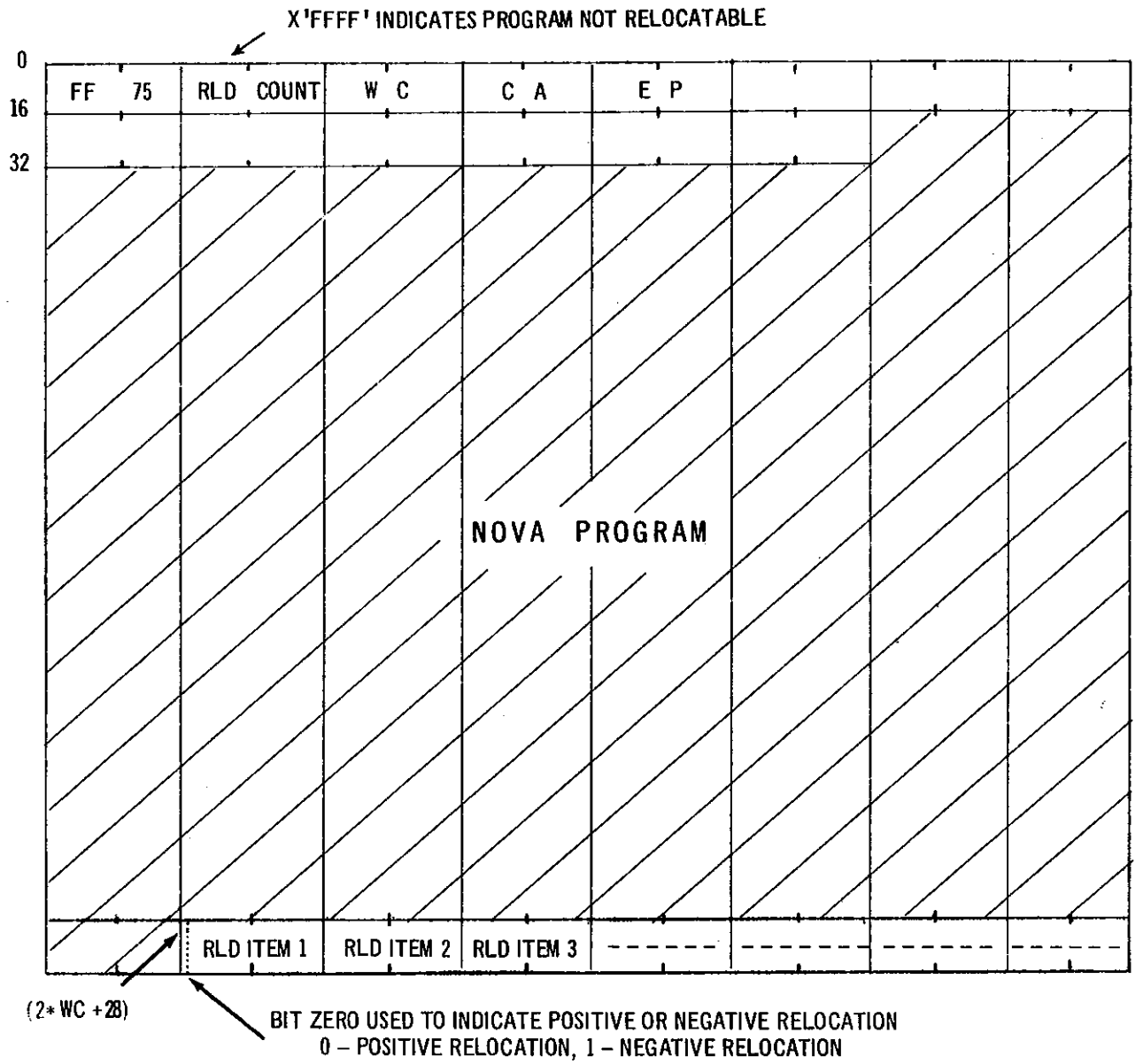
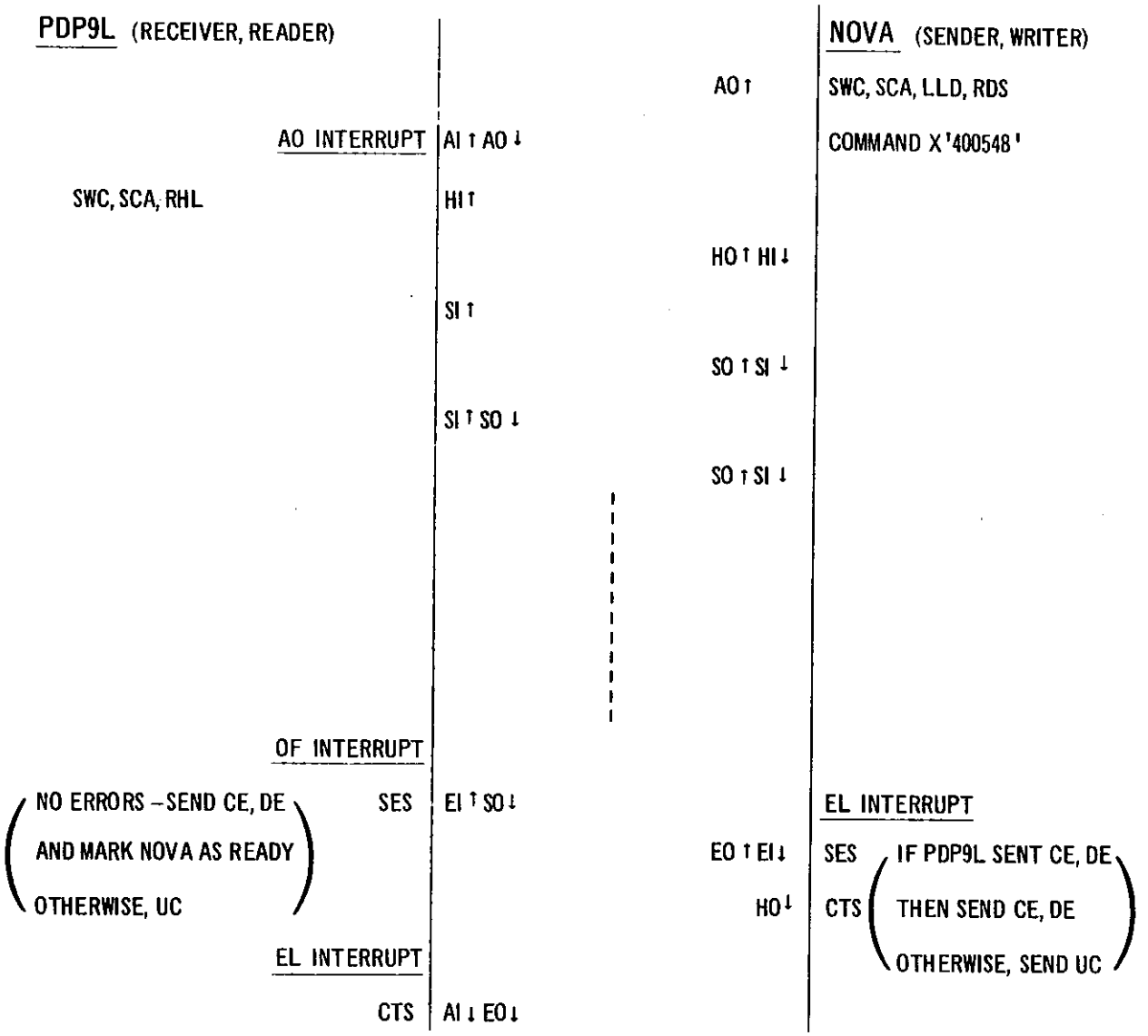
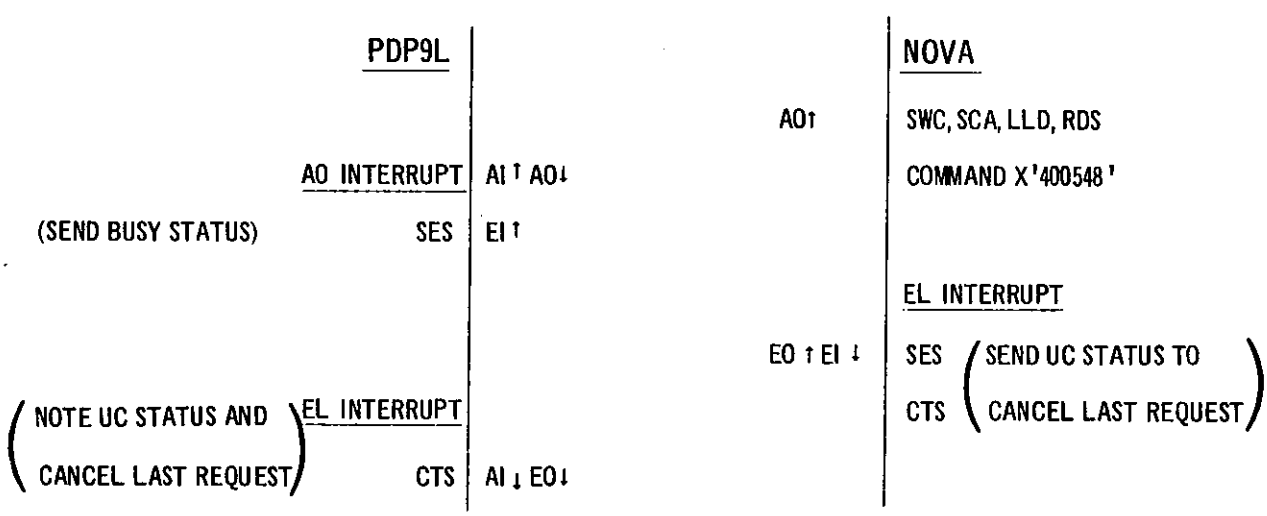


FIGURE 1 LOAD MODULE FORMAT OF NOVA PROGRAMS STORED AS MEMBERS OF AMC.NOVALIB



(a) NORMAL SEQUENCE (NO REQUEST OUTSTANDING)



(b) PRIMARY WRITE REQUEST OUTSTANDING

FIGURE 2 SCHEMATIC DIAGRAMS OF PRIMARY WRITE SEQUENCES TO THE PDP9L COMPUTER

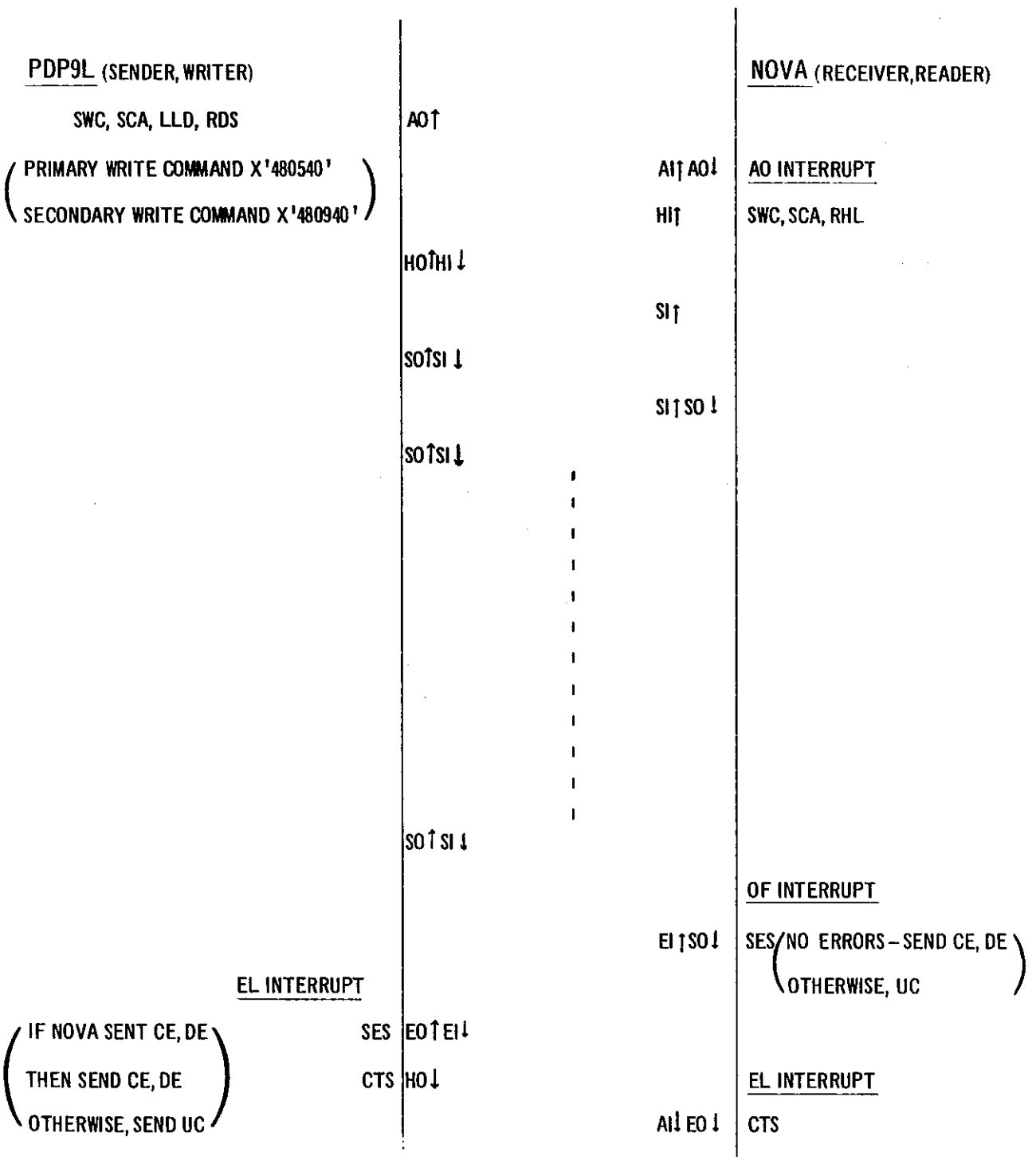


FIGURE 3 SCHEMATIC DIAGRAM OF PRIMARY AND SECONDARY WRITE SEQUENCES FROM THE PDP9L COMPUTER

(WORD COUNTS IN EACH COMPUTER IDENTICAL)

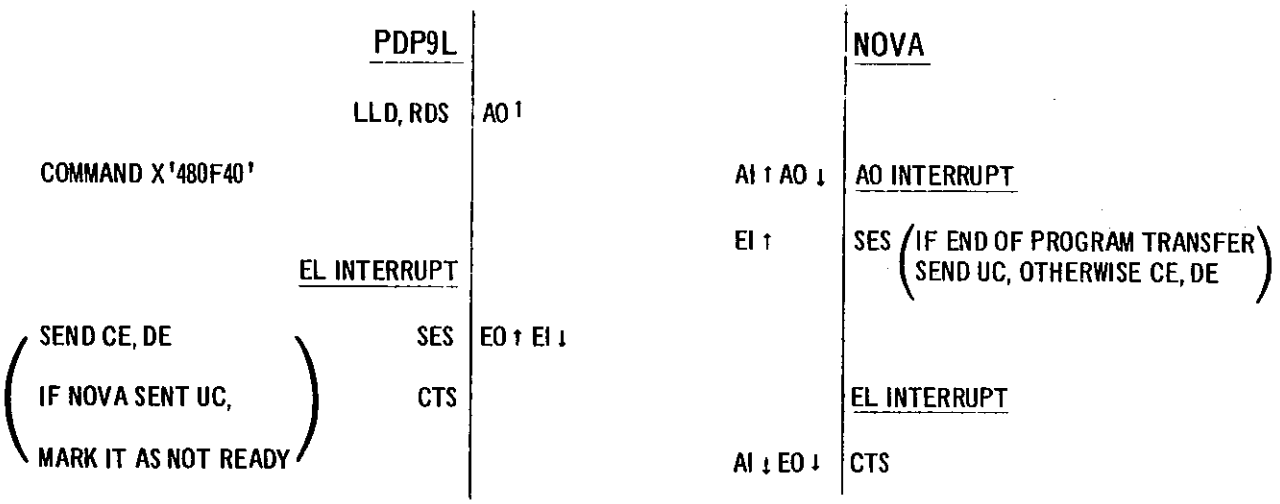


FIGURE 4 SCHEMATIC DIAGRAM OF REWIND/UNLOAD SEQUENCE FROM THE PDP9L COMPUTER

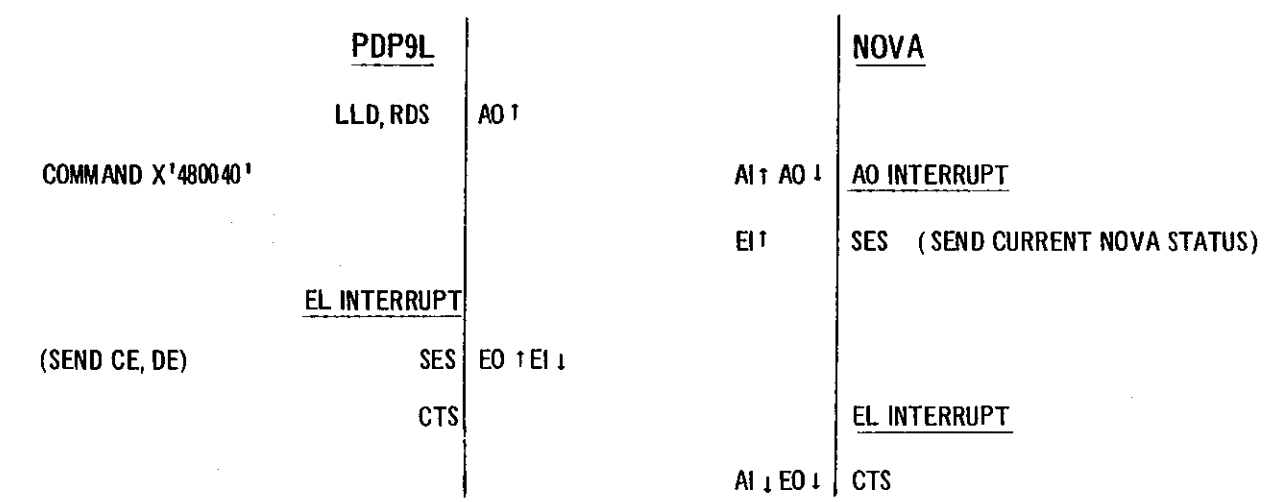
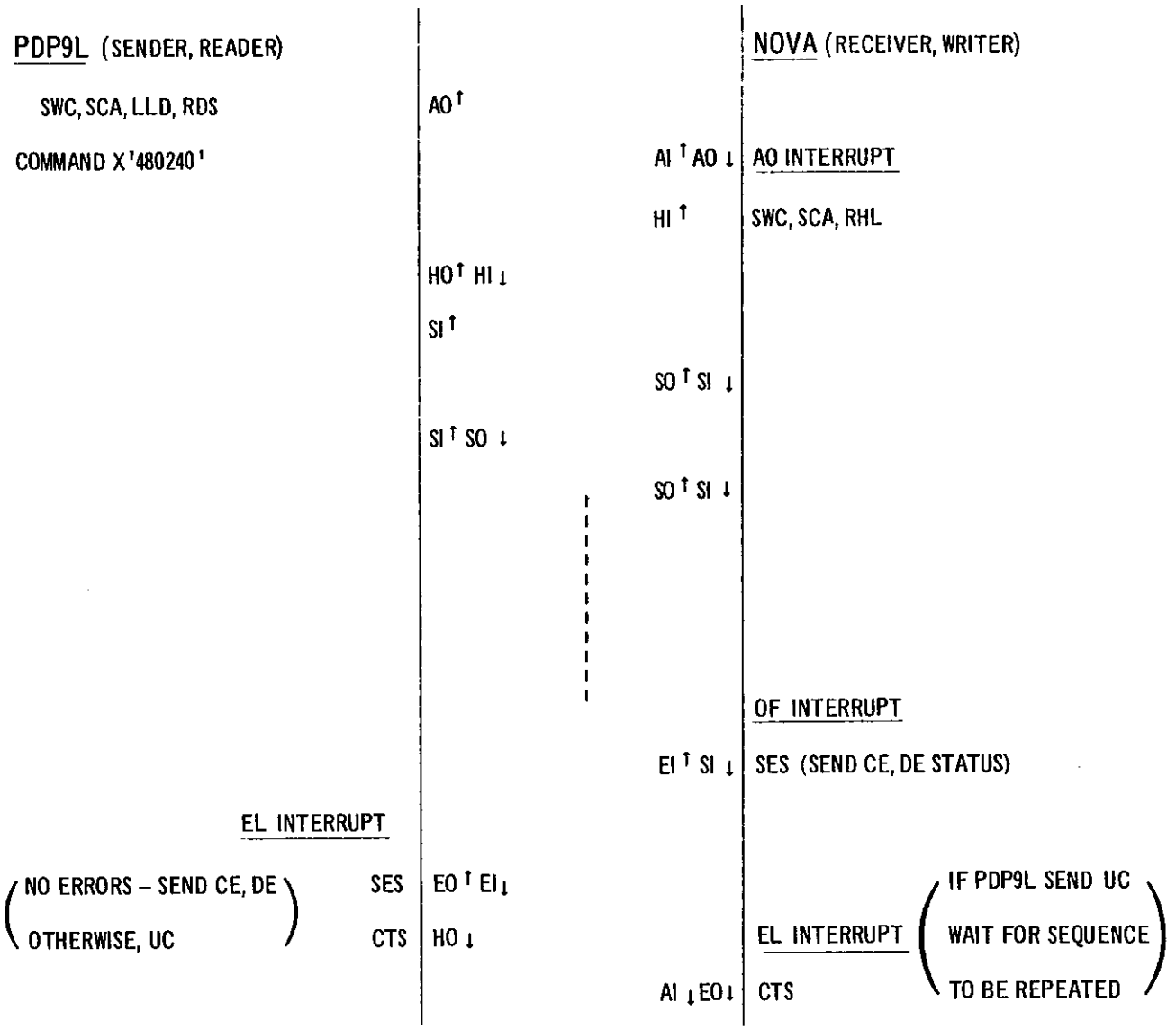
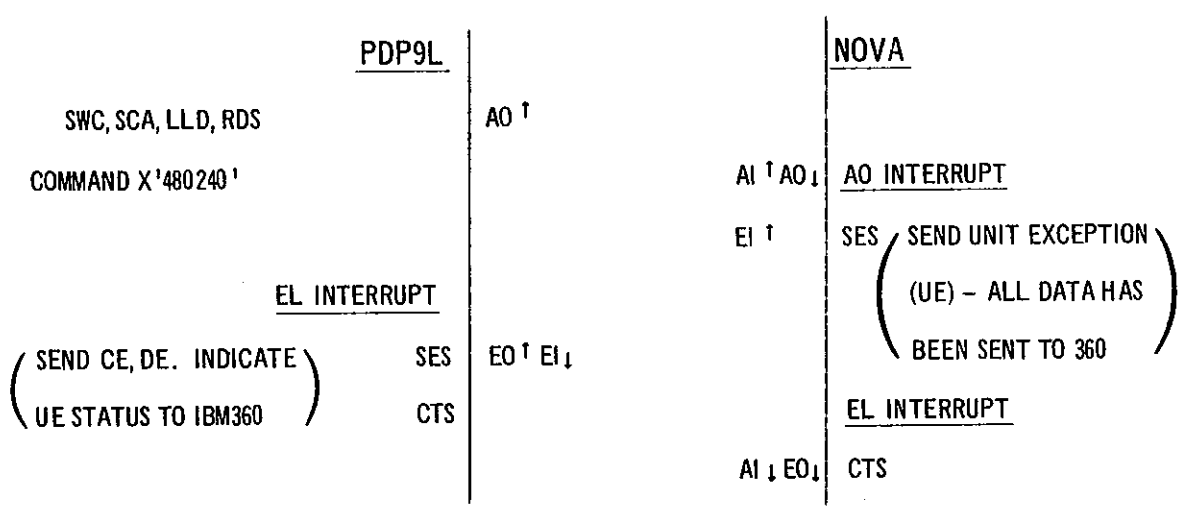


FIGURE 5 SCHEMATIC DIAGRAM OF TEST I/O SEQUENCE FROM THE PDP9L COMPUTER



(a) NORMAL SEQUENCE (DATA TRANSFERRED, NOVA HAS LOWER WORD COUNT)



(b) NO DATA TRANSFERRED

FIGURE 6 SCHEMATIC DIAGRAMS OF READ SEQUENCES FROM THE PDP9L COMPUTER

APPENDIX A

SOURCE LISTING OF THE NOVALOAD PROGRAM

```
//PLSDPL   JOB (C1C19267,B1),DR.P.L.SANGER,                          JOB  63
//             MSGCLASS=D,
//             CLASS=C,
//             TIME=1
//ASM  EXEC  NOVASM,PARM='LOAD=OBJ,DECK,NAME=NOVALOAD'
XXASM    EXEC  PGM=NOVASM,REGION=100K                               00000010
XXSTEPLIB  DD  DSN=AAELIB.LMODA,DISP=SHR                            00000020
XX         DD  DSN=RPB.ASM,DISP=SHR                                 00000030
XXSYSLOAD  DD  DSN=&&OBJ,UNIT=SYSDA,DISP=(NEW,PASS),                00000040
XX             SPACE=(TRK,(10,20))                                  00000050
//ASM.SYSPRINT DD SYSOUT=D
X/SYSPRINT DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210)      00000060
XXSYSPUNCH DD  SYSOUT=B,DCB=(RECFM=FB,BLKSIZE=1200,LRECL=80)        00000070
XXSYSUT1   DD  UNIT=SYSDA,SPACE=(TRK,(5,5))                         00000080
//ASM.SYSIN DD *
IEF236I ALLOC. FOR PLSDPL   ASM      ASM
IEF237I 136   ALLOCATED TO STEPLIB
IEF237I 133   ALLOCATED TO
IEF237I 133   ALLOCATED TO SYSLOAD
IEF237I 133   ALLOCATED TO SYSPRINT
IEF237I 332   ALLOCATED TO SYSPUNCH
IEF237I 135   ALLOCATED TO SYSUT1
IEF237I 313   ALLOCATED TO SYSIN

WORK AREA ALLOCATED (HEX BYTES)=00D000
                          CORE=12K
                          XREF=405
                        SYMTAB=135
```

```
 PC  OBJ CODE  ADDR    ERRORS   STMT    SOURCE STATEMENT

                                    1  *
                                    2  *    DATAWAY PROGRAM LOADER - NOVALOAD
                                    3  *
                                    4          PRINT     HEX
0008                                5  TTI     EQU       0(10)
0009                                6  TTO     EQU       0(11)
2E00                                7          ORG       X(2E00)
2E00  65BF                          8  DPL     IORST                 CLEAR ALL DEVICES AND INTDS
2E01  2929      2E2A                9  DPL1    LDA       1,DPL6C     LOAD CR
2E02  0921      2E23               10          JSR       DPL4        GIVE CR
2E03  2928      2E28               11          LDA       1,DPL7C     LOAD LF
2E04  091F      2E23               12          JSR       DPL4        GIVE LF
2E05  2922      2E27               13          LDA       1,DPL1C     LOAD 1ST WORD OF MESSAGE
2E06  091D      2E23               14          JSR       DPL4        PRINT C
2E07  AAC0                         15          MOVS      1,1         SET UP 2ND CHARACTER
2E08  0918      2E23               16          JSR       DPL4        PRINT A
2E09  291F      2E28               17          LDA       1,DPL1C+1   LOAD 2ND WORD OF MESSAGE
2E0A  0919      2E23               18          JSR       DPL4        PRINT L
2E0B  0918      2E23               19          JSR       DPL4        PRINT L
2E0C  AAC0                         20          MOVS      1,1         SET UP BLANK CHARACTER
2E0D  0916      2E23               21          JSR       DPL4        PRINT SPACE
2E0E  8530                         22          SUBC      0,0         ZERO AC0
2E0F  4160      2E6F               23          STA       0,DPL1S     SET 1ST 8 CHARACTERS TO ZERO
2E10  4160      2E70               24          STA       0,DPL1S+1
2E11  4160      2E71               25          STA       0,DPL1S+2
2E12  4160      2E72               26          STA       0,DPL1S+3
2E13  3916      2E29               27          LDA       3,DPL4C     LOAD CONSTANT 4
2E14  5967      2E78               28          STA       3,DPL2S     STORE THIS AS LOOP COUNTER
2E15  3118      2E2D               29          LDA       2,DPL3C     LOAD A(DPL1S)
2E16  0922      2E38               30  DPL2    JSR       DPL9        READ A CHARACTER
2E17  090C      2E23               31          JSR       DPL4        ECHO CHARACTER
2E18  A2C0                         32          MOVS      1,0         COPY CHAR INTO AC0 AND SWAP HALVES
2E19  091F      2E38               33          JSR       DPL9        READ ANOTHER CHARACTER
2E1A  0909      2E23               34          JSR       DPL4        ECHO CHARACTER
2E1B  A600                         35          ADD       1,0         FORM WORD TO BE STORED
2E1C  4200      0000               36          STA       0,0(2)      STORE THIS IN 24 BYTE AREA
2E1D  D300                         37          INC       2,2         ADD ONE TO ADDRESS
2E1E  8530                         38          SUBC      0,0         ZERO AC0
2E1F  195C      2E7B               39          DSZ       DPL2S       SKIP WHEN 8 CHARS HAVE BEEN READ
2E20  01F6      2E16               40          JMP       DPL2        OTHERWISE, CONTINUE
2E21  0917      2E38               41  DPL3    JSR       DPL9        READ A CHARACTER
2E22  01FF      2E21               42          JMP       DPL3        CONTINUE UNTIL CR IS READ
2E23  6A49                         43  DPL4    DOAS      1,TTO       PRINT CHARACTER
2E24  6789                         44  DPL5    SKPDN     TTO         SKIP WHEN CHAR IS PRINTED
2E25  01FF      2E24               45          JMP       DPL5        OTHERWISE, CHECK AGAIN
2E26  0300      0000               46          JMP       0(3)        RETURN
2E27  41C3                         47  DPL1C   DC        C'CAL
2E29  0004                         48  DPL4C   DC        F(4)
2E2A  008D                         49  DPL6C   DC        X(8D)       CR
2E2B  000A                         50  DPL7C   DC        X(0A)       LF
2E2C  0021                         51  DPL8C   DC        X(21)       EXCLAMATION MARK
2E2D  2E6F                         52  DPL3C   DC        A(DPL1S)
2E2E  4005                         53  DPL9C   DC        X(4005)     PRIMARY WRITE TO 9L COMMAND
2E2F  0048                         54  DPL1D   DC        X(48)       NOVA CORE ADDRESS
2E30  2ECF                         55  DPL2D   DC        A(DPL1T-1)  ADDR OF WORD BEFORE INTERRUPT TABLE
2E31  0040                         56  DPL3D   DC        X(40)       PDP9L ADDRESS
```

```
 PC   OBJ CODE  ADDR    ERRORS   STMT    SOURCE STATEMENT

                                    57  * SERVICE OUT
2E32   663F                         58  DPL6    HALT
2E33   01CD     2E00                59          JMP     DPL             RESTART IF CONTINUE IS PRESSED
                                    60  * DISCONNECT, REPEAT AND NO REPLY
2E34   394B     2E7F                61  DPL8    LDA     3,DPL6S         LOAD SWITCH 3
2E35   FA05                         62          MOV     3,3,SNR         SKIP IF NOT PRIMARY WRITE TO 9L
2E36   0111     2E47                63          JMP     DPL11           OTHERWISE, RESTART WRITE TO 9L
2E37   011A     2E51                64          JMP     DPL12           WAIT FOR NEW 9L SEQUENCE
2E38   5944     2E7C                65  DPL9    STA     3,DPL3S         SAVE RETURN ADDRESS
2E39   6788                         66  DPL10   SKPDN   TTI             SKIP WHEN CHARACTER IS AVAILABLE
2E3A   01FF     2E39                67          JMP     DPL10           OTHERWISE, CHECK AGAIN
2E3B   6988                         68          DIAC    1,TTI           READ CHARACTER INTO AC1
2E3C   39F0     2E2C                69          LDA     3,DPL8C         LOAD CODE FOR EXCLAMATION MARK
2E3D   BD05                         70          SUB     1,3,SNR         SKIP IF NOT EXCLAMATION MARK
2E3E   01C3     2E01                71          JMP     DPL1            OTHERWISE, RESTART PROGRAM
2E3F   39E8     2E2A                72          LDA     3,DPL6C         LOAD CODE FOR CR
2E40   BD04                         73          SUB     1,3,SZR         SKIP IF CR
2E41   053B     2E7C                74          JMP   I DPL3S           OTHERWISE, RETURN
2E42   4200     0000                75          STA     0,0(2)          STORE LAST WORD IN 24 BYTE AREA
2E43   593C     2E7F                76          STA     3,DPL6S         INDICATE PRIMARY WRITE TO PDP9L
2E44   09DF     2E23                77          JSR     DPL4            ECHO CR
2E45   29E6     2E2B                78          LDA     1,DPL7C         LOAD CODE FOR LF
2E46   09DD     2E23                79          JSR     DPL4            SEND LF
2E47   2137     2E7E                80  DPL11   LDA     0,DPL2C         LOAD CONSTANT 12
2E48   661B                         81          SWC     0               LOAD WC REGISTER
2E49   29E4     2E2D                82          LDA     1,DPL3C         LOAD A(DPL1S)
2E4A   6C1B                         83          SCA     1               LOAD CA REGISTER
2E4B   4130     2E7B                84          STA     0,DPL2S         SAVE COPY OF WORD COUNT
2E4C   4930     2E7C                85          STA     1,DPL3S         SAVE COPY OF CURRENT ADDRESS
2E4D   31E1     2E2E                86          LDA     2,DPL9C         LOAD PRIMARY WRITE TO 9L COMMAND
2E4E   39E1     2E2F                87          LDA     3,DPL1D         LOAD NOVA CORE ADDR
2E4F   7C1A                         88          LLD     3               LOAD THIS INTO DATA REGISTER
2E50   729A                         89          RDS     2               LOAD COMMAND AND RAO
2E51   FD30                         90  DPL12   SUBC    3,3             ZERO AC3
2E52   592B     2E7D                91          STA     3,DPL4S         INDICATE NOT WRITE FROM 9L
2E53   592D     2E80                92          STA     3,DPL8S         INDICATE NOVA DID NOT SEND 1ST STATUS
2E54   679A                         93  DPL13   SKPIRN                  SKIP IF DW INTERRUPT OCCURS
2E55   01FF     2E54                94          JMP     DPL13           OTHERWISE, CHECK AGAIN
2E56   711B                         95          RIR     2               READ INTERRUPT REGISTER
2E57   721B                         96          CIR     2               CLEAR INTERRUPT CONDITION
2E58   39D8     2E30                97          LDA     3,DPL2D         LOAD POINTER TO INTERRUPT TABLE
2E59   DE00                         98          ADD     2,3             FORM ADDRESS OF CORRECT ENTRY
2E5A   0700     0000                99          JMP   I 0(3)            PROCESS INTERRUPT
                                   100  * ADDRESS OUT
2E5B   6B1A                        101  DPL14   RLD     1               READ RETURN ADDRESS
2E5C   651A                        102          RAB     0               READ ADDRESS BYTE
2E5D   39D4     2E31               103          LDA     3,DPL3D         LOAD PDP9L ADDRESS
2E5E   ED05                        104          SUB     3,1,SNR         SKIP IF NOT PDP9L
2E5F   39D0     2E2F               105          LDA     3,DPL1D         OTHERWISE, LOAD NOVA CORE ADDR
2E60   E504                        106          SUB     3,0,SZR         SKIP IF NOVA CORE ADDR
2E61   0128     2E89               107          JMP     DPL18           OTHERWISE, SEND UNIT CHECK STATUS
2E62   731B                        108          RCB     2               READ COMMAND BYTE
2E63   2967     2ECA               109          LDA     1,DPL6D         LOAD SECONDARY WRITE COMMAND CODE
2E64   CD05                        110          SUB     2,1,SNR         SKIP IF NOT SECONDARY WRITE
2E65   0104     2E69               111          JMP     DPL15           OTHERWISE, START SECONDARY WRITE
2E66   3963     2EC9               112          LDA     3,DPL5D         LOAD PRIMARY WRITE COMMAND
```

```
 PC   OBJ CODE  ADDR    ERRORS   STMT     SOURCE STATEMENT

2E67   DD04                      113          SUB     2,3,SZR       SKIP IF PRIMARY WRITE COMMAND
2E68   011D    2E85              114          JMP     DPL17         OTHERWISE, CHECK FOR REWIND/UNLOAD
2E69   2913    2E7C              115  DPL15   LDA     1,DPL3S       LOAD REQUIRED CURRENT ADDRESS
2E6A   6C1B                      116          SCA     1             LOAD CURRENT ADDRESS REGISTER
2E6B   2110    2E7B              117          LDA     0,DPL2S       LOAD REQUIRED WORD COUNT
2E6C   6698                      118          SDT     0             LOAD WORD COUNT REGISTER AND RHL
2E6D   1110    2E7D              119          ISZ     DPL4S         INDICATE WRITE FROM 9L
2E6E   01E6    2E54              120          JMP     DPL13         WAIT FOR OF INTERRUPT
2E6F   0000                      121  DPL1S   DC      F(0)          24 BYTE AREA
2E70   0000                      122          REPEAT  11
2E7B   0000                      123  DPL2S   DC      F(0)          CURRENT WC
2E7C   0000                      124  DPL3S   DC      F(0)          CURRENT CA
2E7D   0000                      125  DPL4S   DC      F(0)          WRITE FROM 9L SWITCH
2E7E   000C                      126  DPL2C   DC      F(12)         CONSTANT 12 - CE,DE STATUS
2E7F   0000                      127  DPL6S   DC      F(0)          SWITCH 3 - COMMAND TYPE
2E80   0000                      128  DPL8S   DC      F(0)          SWITCH 2 - 1ST STATUS SWITCH
2E81   AD30                      129  DPL16   SUBC    1,1           ZERO AC1
2E82   6A1A                      130          LHD     1             CLEAR HIGH PART OF DATA REGISTER
2E83   64DA                      131          SES     0             SEND ENDING STATUS
2E84   0300    0000              132          JMP     0(3)          RETURN
2E85   D205                      133  DPL17   MOV     2,2,SNR       SKIP IF NOT TIO
2E86   2149    2ECF              134          LDA     0,DPL5S       OTHERWISE, LOAD NOVA STATUS
2E87   39F8    2E7F              135          LDA     3,DPL6S       LOAD COMMAND TYPE
2E88   FA96                      136          MOVZR   3,3,SEZ       SKIP IF REWIND/UNLOAD NOT EXPECTED
2E89   2142    2EC8              137  DPL18   LDA     0,DPL7D       LOAD UC STATUS
2E8A   11F6    2E80              138  DPL19   ISZ     DPL8S         INDICATE NOVA SENT 1ST STATUS
2E8B   09F6    2E81              139          JSR     DPL16         SEND ENDING STATUS
2E8C   01C8    2E54              140          JMP     DPL13         WAIT FOR EL INTERRUPT
                                 141  * END LINE
2E8D   19F3    2E80              142  DPL20   DSZ     DPL8S         SKIP IF NOVA SENT 1ST STATUS
2E8E   0127    2EB5              143          JMP     DPL25         OTHERWISE, MUST BE WRITE TO 9L
2E8F   60DB                      144          CTS                   TERMINATE DATAWAY SEQUENCE
2E90   29EF    2E7F              145          LDA     1,DPL6S       LOAD COMMAND TYPE
2E91   AA96                      146          MOVZR   1,1,SEZ       SKIP IF NOT REWIND/UNLOAD
2E92   012E    2EC0              147          JMP     DPL26         OTHERWISE, EXAMINE PGM ENTRY POINT
2E93   19EA    2E7D              148          DSZ     DPL4S         SKIP IF WRITE FROM 9L
2E94   01BD    2E51              149          JMP     DPL12         OTHERWISE, MUST BE TIO FROM 9L
2E95   39E9    2E7E              150          LDA     3,DPL2C       LOAD CE,DE STATUS
2E96   9D04                      151          SUB     0,3,SZR       SKIP IF CE,DE STATUS
2E97   01BD    2E54              152          JMP     DPL13         OTHERWISE, ALLOW 360 TO REPEAT SEQ
2E98   AA04                      153          MOV     1,1,SZR       SKIP IF PRIMARY WRITE FROM 9L
2E99   010A    2EA3              154          JMP     DPL21         OTHERWISE, MUST BE SECONDARY WRITE
2E9A   21D5    2E6F              155          LDA     0,DPL1S       LOAD TWC FOR SECONDARY WRITE
2E9B   31D5    2E70              156          LDA     2,DPL1S+1     LOAD ADDR FOR PROGRAM LOAD - APL
2E9C   C600                      157          ADD     2,0           FORM TWC+APL
2E9D   11E2    2E7F              158          ISZ     DPL6S         INDICATE SECONDARY WRITES EXPECTED
2E9E   392E    2ECC              159          LDA     3,DPL9D       LOAD ADDR OF START OF DPL
2E9F   9D13                      160          SUBZ    0,3,SNC       SKIP IF A(DPL).GE.TWC+APL
2EA0   0112    2EB2              161          JMP     DPL23         OTHERWISE, REJECT NEXT COMMAND
2EA1   392C    2ECD              162          LDA     3,DPL1E       LOAD CONSTANT 400
2EA2   59D9    2E7B              163          STA     3,DPL2S       STORE THIS AS WC FOR NEXT TRANSFER
2EA3   21CC    2E6F              164  DPL21   LDA     0,DPL1S       LOAD TWC REMAINING
2EA4   8205                      165          MOV     0,0,SNR       SKIP IF THIS IS NOT ZERO
2EA5   010E    2EB3              166          JMP     DPL24         OTHERWISE, SET UP FOR REW/UNLOAD
2EA6   3927    2ECD              167          LDA     3,DPL1E       LOAD CONSTANT 400
2EA7   E512                      168          SUBZ    3,0,SZC       SKIP IF TWC.LT.400
```

```
  PC  OBJ CODE  ADDR    ERRORS    STMT     SOURCE STATEMENT

2EA8  0104      2EAC              169              JMP       DPL22          OTHERWISE, UPDATE TWC
2EA9  E600                        170              ADD       3,0            FORM OLD TWC
2EAA  41D1      2E7B              171              STA       0,DPL2S        STORE THIS AS CURRENT WC
2EAB  8530                        172              SUBC      0,0            ZERO AC0
2EAC  41C3      2E6F              173  DPL22       STA       0,DPL1S        STORE NEW TWC
2EAD  21C3      2E70              174              LDA       0,DPL1S+1      LOAD APL
2EAE  41CE      2E7C              175              STA       0,DPL3S        STORE THIS AS CA FOR NEXT SEQUENCE
2EAF  E600                        176              ADD       3,0            FORM APL+400
2EB0  41C0      2E70              177              STA       0,DPL1S+1      UPDATE APL
2EB1  01A3      2E54              178              JMP       DPL13          WAIT FOR NEXT SEQUENCE FROM 9L
2EB2  49BF      2E71              179  DPL23       STA       1,DPL1S+2      ZERO PROGRAM ENTRY POINT
2EB3  11CC      2E7F              180  DPL24       ISZ       DPL6S          SET COMMAND TYPE TO 3
2EB4  01A0      2E54              181              JMP       DPL13          WAIT FOR 9L SEQUENCE
2EB5  731A                        182  DPL25       RLD       2              READ STATUS FROM DATA REGISTER
2EB6  21C8      2E7E              183              LDA       0,DPL2C        LOAD CE,DE STATUS
2EB7  3917      2ECE              184              LDA       3,DPL3E        LOAD BUSY STATUS
2EB8  DD05                        185              SUB       2,3,SNR        SKIP IF NOT BUSY STATUS
2EB9  2112      2ECB              186              LDA       0,DPL7D        OTHERWISE, LOAD UC STATUS
2EBA  09C7      2E81              187              JSR       DPL16          SEND ENDING STATUS
2EBB  600B                        188              CTS                      TERMINATE DATAWAY SEQUENCE
2EBC  9504                        189              SUB       0,2,SZR        SKIP IF NO ERROR IN DT SEQUENCE
2EBD  018A      2E47              190              JMP       DPL11          OTHERWISE, RESTART WRITE TO 9L
2EBE  11C1      2E7F              191              ISZ       DPL6S          INDICATE PRIMARY WRITE FROM 9L WANTED
2EBF  0192      2E51              192              JMP       DPL12          WAIT FOR 9L SEQUENCE
2EC0  39B1      2E71              193  DPL26       LDA       3,DPL1S+2      LOAD PROGRAM ENTRY POINT
2EC1  FA05                        194              MOV       3,3,SNR        SKIP IF THIS IS NONZERO
2EC2  663F                        195              HALT                     OTHERWISE, HALT
2EC3  0300      0000              196              JMP       0(3)           BRANCH TO PROGRAM ENTRY POINT
                                  197  * OVERFLOW
2EC4  21BA      2E7E              198  DPL27       LDA       0,DPL2C        LOAD CE,DE STATUS
2EC5  675A                        199              SKPDEZ                   SKIP IF NO ERROR IN DATA TRANSFER
2EC6  2105      2ECB              200              LDA       0,DPL7D        OTHERWISE, LOAD UNIT CHECK STATUS
2EC7  4108      2ECF              201              STA       0,DPL5S        SAVE NOVA STATUS FOR TIO FROM 9L
2EC8  01C2      2E8A              202              JMP       DPL19          SEND ENDING STATUS
2EC9  0005                        203  DPL5D       DC        X(05)          PRIMARY WRITE COMMAND
2ECA  0009                        204  DPL6D       DC        X(09)          SECONDARY WRITE COMMAND
2ECB  0002                        205  DPL7D       DC        X(02)          UNIT CHECK STATUS
2ECC  2E00                        206  DPL9D       DC        A(DPL)         ADDR OF PROGRAM
2ECD  0190                        207  DPL1E       DC        F(400)
2ECE  0010                        208  DPL3E       DC        X(10)          BUSY STATUS
2ECF  000C                        209  DPL5S       DC        X(000C)        NOVA STATUS FOR TIO FROM 9L
2ED0  2E34                        210  DPL1T       DC        A(DPL8)        DISCONNECT
2ED1  2E34                        211              DC        A(DPL8)        REPEAT
2ED2  2E34                        212              DC        A(DPL8)        NO REPLY
2ED3  2E5B                        213              DC        A(DPL14)       ADDRESS OUT
2ED4  2E32                        214              DC        A(DPL6)        SERVICE OUT
2ED5  2E8D                        215              DC        A(DPL20)       END LINE
2ED6  2EC4                        216              DC        A(DPL27)       OVERFLOW
2E00                              217              END       DPL
```

CROSS-REFERENCE

SYMBOL	VALUE	DEFN	REFERENCES											
DPL	2E00	0008	0059	0206	0217									
DPL1	2E01	0009	0071											
DPL1C	2E27	0047	0013	0017										
DPL1D	2E2F	0054	0087	0105										
DPL1E	2ECD	0207	0162	0167										
DPL1S	2E6F	0121	0023	0024	0025	0026	0052	0155	0156	0164	0173	0174	0177	0179	0193
DPL1T	2ED0	0210	0055											
DPL10	2E39	0066	0067											
DPL11	2E47	0080	0063	0190										
DPL12	2E51	0090	0064	0149	0192									
DPL13	2E54	0093	0094	0120	0140	0152	0178	0181						
DPL14	2E5B	0101	0213											
DPL15	2E69	0115	0111											
DPL16	2E81	0129	0139	0187										
DPL17	2E85	0133	0114											
DPL18	2E89	0137	0107											
DPL19	2E8A	0138	0202											
DPL2	2E16	0030	0040											
DPL2C	2E7E	0126	0080	0150	0183	0198								
DPL2D	2E30	0055	0097											
DPL2S	2E7B	0123	0028	0039	0084	0117	0163	0171						
DPL20	2E8D	0142	0215											
DPL21	2EA3	0164	0154											
DPL22	2EAC	0173	0169											
DPL23	2EB2	0179	0161											
DPL24	2EB3	0180	0166											
DPL25	2EB5	0182	0143											
DPL26	2EC0	0193	0147											
DPL27	2EC4	0198	0216											
DPL3	2E21	0041	0042											
DPL3C	2E2D	0052	0029	0082										
DPL3D	2E31	0056	0103											
DPL3E	2ECE	0208	0184											
DPL3S	2E7C	0124	0065	0074	0085	0115	0175							
DPL4	2E23	0043	0010	0012	0014	0016	0018	0019	0021	0031	0034	0077	0079	
DPL4C	2E29	0048	0027											
DPL4S	2E7D	0125	0091	0119	0148									
DPL5	2E24	0044	0045											
DPL5D	2EC9	0203	0112											
DPL5S	2ECF	0209	0134	0201										
DPL6	2E32	0058	0214											
DPL6C	2E2A	0049	0009	0072										
DPL6D	2ECA	0204	0109											
DPL6S	2E7F	0127	0061	0076	0135	0145	0158	0180	0191					
DPL7C	2E2B	0050	0011	0078										
DPL7D	2ECB	0205	0137	0186	0200									
DPL8	2E34	0061	0210	0211	0212									
DPL8C	2E2C	0051	0069											
DPL8S	2E80	0128	0092	0138	0142									
DPL9	2E38	0065	0030	0033	0041									
DPL9C	2E2E	0053	0086											
DPL9D	2ECC	0206	0159											
TTI	0008	0005	0066	0068										
TTO	0009	0006	0043	0044										

NO STATEMENTS FLAGGED IN THIS ASSEMBLY

```
IEF142I - STEP WAS EXECUTED - COND CODE 0000
IEF285I   AAELIB.LMODA                                    KEPT
IEF285I   VOL SER NOS= AAE002.
IEF285I   RPB.ASM                                         KEPT
IEF285I   VOL SER NOS= AAE004.
IEF285I   SYS73046.T181741.RV000.PLSDPL.OBJ               PASSED
IEF285I   VOL SER NOS= AAE004.
IEF285I   SYS73046.T181741.SV000.PLSDPL.R0000001          SYSOUT
IEF285I   VOL SER NOS= AAE004.
IEF285I   SYS73046.T181741.RV000.PLSDPL.R0000003          DELETED
IEF285I   VOL SER NOS= AAE008.
IEF373I STEP /ASM      / START 73046.1817
IEF374I STEP /ASM      / STOP  73046.1818 CPU   0MIN 05.58SEC MAIN 100K LCS   0K
*** CONDITION CODE =  000(HEX)
//LKED  EXEC  NOVALKED,NAME=NOVALOAD
XXNOVALKED PROC NAME=TEMPNAME                                          00000010
XXLKED   EXEC  PGM=IEWL,PARM='LIST,MAP',REGION=96K,COND=(0,NE)         00000020
XXSYSLIN   DD  DSN=&&OBJ,DISP=(OLD,DELETE)                             00000030
XXSYSLMOD  DD  DSN=AMC.NOVALIB(&NAME),DISP=SHR                         00000040
IEF653I SUBSTITUTION JCL - DSN=AMC.NOVALIB(NOVALOAD),DISP=SHR
//LKED.SYSPRINT DD SYSOUT=D
X/SYSPRINT DD  SYSOUT=A                                                00000050
XXSYSUT1   DD  DSN=&&UT1,UNIT=SYSDA,SPACE=(1700,(50,100))              00000060
//
IEF236I ALLOC. FOR PLSDPL   LKED     LKED
IEF237I 133   ALLOCATED TO SYSLIN
IEF237I 126   ALLOCATED TO SYSLMOD
IEF237I 133   ALLOCATED TO SYSPRINT
IEF237I 134   ALLOCATED TO SYSUT1

 F88-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED LIST,MAP
          DEFAULT OPTION(S) USED -  SIZE=(90112,6144)
                                                    MODULE MAP


  CONTROL SECTION                       ENTRY

    NAME     ORIGIN  LENGTH               NAME   LOCATION     NAME   LOCATION     NAME   LOCATION     NAME   LOCATION

  NOVATEXT       00     1CA


 ENTRY ADDRESS       00
 TOTAL LENGTH       1D0

****NOVALOAD  NOW REPLACED IN DATA SET


IEF142I - STEP WAS EXECUTED - COND CODE 0000
IEF285I   SYS73046.T181741.RV000.PLSDPL.OBJ               DELETED
IEF285I   VOL SER NOS= AAE004.
IEF285I   AMC.NOVALIB                                     KEPT
IEF285I   VOL SER NOS= AAE003.
IEF285I   SYS73046.T181741.SV000.PLSDPL.R0000005          SYSOUT
IEF285I   VOL SER NOS= AAE004.
IEF285I   SYS73046.T171303.RV015.INIT.UT1                 KEPT
IEF285I   VOL SER NOS= AAE014.
IEF373I STEP /LKED     / START 73046.1818
IEF374I STEP /LKED     / STOP  73046.1819 CPU   0MIN 01.52SEC MAIN  96K LCS   0K
*** CONDITION CODE =  000(HEX)
IEF375I  JOB /PLSDPL  / START 73046.1817
IEF376I  JOB /PLSDPL  / STOP  73046.1819 CPU   0MIN 07.10SEC
HIGHEST CONDN CODE =  000(HEX)
```

APPENDIX B

SOURCE LISTING OF THE PLSBDPL PROGRAM

```
IEF290I  PLSBDPL  SYSOUT=C,
//PLSBDPL  JOB (C1C19267,B1),OR,P.L.SANGER,                              JOB  64
//             MSGCLASS=D,
//             CLASS=C,
//             TIME=1
//DPLTPE  EXEC  PTAPE,PRG=NOVATPE
XXPTAPE  PROC  PRG=AEPTPNCH,PGMLIM=2500                                00000010
XXPGM    EXEC  PGM=AELINK,REGION=52K                                   00000020
XXAELINKUT DD  UNIT=SYSDA,SPACE=(TRK,2),DSN=&OBJ                       00000030
IEF653I SUBSTITUTION JCL - UNIT=SYSDA,SPACE=(TRK,2),DSN=&OBJ
XXCONPROG  DD  DISP=SHR,DSN=SYS2.LINKLIB(AEPTCON)                      00000040
XXDD1      DD  DISP=SHR,DSN=PDP.LINKLIB(AEPTPNCH)                      00000050
XXDD2      DD  DISP=SHR,DSN=PDP.LINKLIB(&PRG)                          00000060
IEF653I SUBSTITUTION JCL - DISP=SHR,DSN=PDP.LINKLIB(NOVATPE)
XXDD3      DD  SYSOUT=C                                                00000070
XXFT03F001 DD  SYSOUT=A,                                               00000080
XX             DCB=(BLKSIZE=1330,LRECL=133,RECFM=FBA),OUTLIM=&PGMLIM   00000090
IEF653I SUBSTITUTION JCL - DCB=(BLKSIZE=1330,LRECL=133,RECFM=FBA),OUTLIM=2500
XXFT06F001 DD  SYSOUT=A,                                               00000100
XX             DCB=(BLKSIZE=1330,LRECL=133,RECFM=FBA),OUTLIM=&PGMLIM   00000110
IEF653I SUBSTITUTION JCL - DCB=(BLKSIZE=1330,LRECL=133,RECFM=FBA),OUTLIM=2500
XXPNCHPROG DD  DISP=SHR,DSN=PDP.LINKLIB(AEBUFF)                        00000120
XXSYSIN    DD  DISP=SHR,DSN=SYSLDATA(AEPTDATA)                         00000130
XXSYSPRINT DD  SYSOUT=A                                                00000140
XXUSERPROG DD  DISP=SHR,DSN=PDP.LINKLIB(AEBUFF)                        00000150
IEF236I ALLOC. FOR PLSBDPL  PGM      DPLTPE
IEF237I 135   ALLOCATED TO AELINKUT
IEF237I 137   ALLOCATED TO CONPROG
IEF237I 126   ALLOCATED TO DD1
IEF237I 126   ALLOCATED TO DD2
IEF237I 134   ALLOCATED TO DD3
IEF237I 3A1   ALLOCATED TO FT03F001
IEF237I 3A4   ALLOCATED TO FT06F001
IEF237I 126   ALLOCATED TO PNCHPROG
IEF237I 136   ALLOCATED TO SYSIN
IEF237I 3A5   ALLOCATED TO SYSPRINT
IEF237I 126   ALLOCATED TO USERPROG
IEF142I - STEP HAS EXECUTED - COND CODE 0000
IEF285I   SYS73046.T181908.RV000.PLSBDPL.OBJ           DELETED
IEF285I   VOL SER NOS= AAE008.
IEF285I   SYS2.LINKLIB                                 KEPT
IEF285I   VOL SER NOS= AAE001.
IEF285I   PDP.LINKLIB                                  KEPT
IEF285I   VOL SER NOS= AAE003.
IEF285I   PDP.LINKLIB                                  KEPT
IEF285I   VOL SER NOS= AAE003.
IEF285I   SYS73046.T181908.SV000.PLSBDPL.R0000001      SYSOUT
IEF285I   VOL SER NOS= AAE014.
IEF285I   PDP.LINKLIB                                  KEPT
IEF285I   VOL SER NOS= AAE003.
IEF285I   SYSLDATA                                     KEPT
IEF285I   VOL SER NOS= AAE002.
IEF285I   PDP.LINKLIB                                  KEPT
IEF285I   VOL SER NOS= AAE003.
IEF373I STEP /PGM      / START 73046.1819
IEF374I STEP /PGM      / STOP  73046.1821 CPU   0MIN 00.48SEC MAIN  42K LCS   0K
*** CONDITION CODE =  000(HEX)
//ASM  EXEC  NOVASM,PARM=DECK
XXASM    EXEC  PGM=NOVASM,REGION=100K                                  00000010
XXSTEPLIB  DD  DSN=AAELIB.LMODA,DISP=SHR                               00000020
XX         DD  DSN=RP8.ASM,DISP=SHR                                    00000030
XXSYSLOAD  DD  DSN=&&OBJ,UNIT=SYSDA,DISP=(NEW,PASS),                   00000040
XX             SPACE=(TRK,(10,20))                                     00000850
//ASM.SYSPRINT DD SYSOUT=D
X/SYSPRINT DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210)         00000060
//ASM.SYSPUNCH DD SYSOUT=C
X/SYSPUNCH DD  SYSOUT=B,DCB=(RECFM=FB,BLKSIZE=1200,LRECL=80)           00000070
XXSYSUT1   DD  UNIT=SYSDA,SPACE=(TRK,(5,5))                            00000080
//ASM.SYSIN DD *
//
IEF236I ALLOC. FOR PLSBDPL  ASM      ASM
IEF237I 136   ALLOCATED TO STEPLIB
IEF237I 133 . ALLOCATED TO
IEF237I 135   ALLOCATED TO SYSLOAD
IEF237I 133   ALLOCATED TO SYSPRINT
IEF237I 135   ALLOCATED TO SYSPUNCH
IEF237I 133   ALLOCATED TO SYSUT1
IEF237I 313   ALLOCATED TO SYSIN

WORK AREA ALLOCATED (HEX BYTES)=007530
                          XREF=3000
                        SYMTAB=1000
```

```
 PC   OBJ CODE  ADDR    ERRORS   STMT     SOURCE STATEMENT

                                    1  *
                                    2  *    PLSBDPL
                                    3  *    BOOTSTRAP DATAWAY PROGRAM LOADER. THIS PROGRAM LOADS THE NOVALOAD
                                    4  *    PROGRAM FROM THE IBM 360 COMPUTER. IT IS ASSUMED THAT
                                    5  *    NOVALOAD IS SHORTER THAN 400 WORDS AND ITS STARTING ADDRESS
                                    6  *    CAN BE USED AS ITS ENTRY POINT.
                                    7  *
                                    8          PRINT     HEX
2EF4                                9          ORG       X(2EF4)
2EF4  4ECF                         10  DPL1S   DC        T'NOVALOAD'     NAME OF PROGRAM TO BE LOADED
2EF8  0000                         11  DPL2S   DC        F(0)            COMMAND TYPE
2EF9  0000                         12  DPL3S   DC        F(0)            WRITE FROM 9L SWITCH
2EFA  0000                         13  DPL4S   DC        F(0)            1ST STATUS SWITCH
2EFB  0000                         14  DPL5S   DC        F(0)            WORD COUNT FOR NOVALOAD
2EFC  0000                         15  DPL6S   DC        F(0)            STARTING ADDR AND EP FOR NOVALOAD
2EFD  000C                         16  DPL1C   DC        F(12)           CONSTANT 12 - CE,DE STATUS
2EFE  2EF4                         17  DPL2C   DC        A(DPL1S)        ADDRESS OF NOVALOAD PGM NAME
2EFF  4005                         18  DPL3C   DC        X(4005)         PRIMARY WRITE TO 9L COMMAND
2F00  65BF                         19  DPL     IORST                     CLEAR ALL DEVICES AND INTDS
2F01  FD30                         20          SUBC      3,3             ZERO AC3
2F02  59F6      2EF8               21          STA       3,DPL2S         INDICATE PRIMARY WRITE TO 9L
2F03  21FA      2EFD               22  DPL1    LDA       0,DPL1C         LOAD CONSTANT 12
2F04  6618                         23          SWC       0               LOAD WC REGISTER
2F05  29F9      2EFE               24          LDA       1,DPL2C         LOAD ADDR OF NOVALOAD PGM NAME
2F06  6C1B                         25          SCA       1               LOAD CA REGISTER
2F07  31F8      2EFF               26          LDA       2,DPL3C         LOAD PRIMARY WRITE TO 9L COMMAND
2F08  3951      2F59               27          LDA       3,DPL4C         LOAD NOVA CORE ADDRESS
2F09  7C1A                         28          LLD       3               LOAD THIS INTO DATA REGISTER
2F0A  729A                         29          RDS       2               LOAD COMMAND AND RAO
2F0B  FD30                         30  DPL2    SUBC      3,3             ZERO AC3
2F0C  59ED      2EF9               31          STA       3,DPL3S         INDICATE NOT WRITE FROM 9L
2F0D  59ED      2EFA               32          STA       3,DPL4S         INDICATE NOVA DID NOT SEND 1ST STATUS
2F0E  679A                         33  DPL3    SKPIRN                    SKIP IF DW INTERRUPT OCCURS
2F0F  01FF      2F0E               34          JMP       DPL3            OTHERWISE, CHECK AGAIN
2F10  7118                         35          RIR       2               READ INTERRUPT REGISTER
2F11  721B                         36          CIR       2               CLEAR INTERRUPT CONDITION
2F12  3948      2F5A               37          LDA       3,DPL5C         LOAD POINTER TO INTERRUPT TABLE
2F13  DE00                         38          ADD       2,3             FORM ADDRESS OF CORRECT ENTRY
2F14  0700      0000               39          JMP    I  0(3)            PROCESS INTERRUPT
                                   40  * DISCONNECT, REPEAT AND NO REPLY
2F15  39E3      2EF8               41  DPL5    LDA       3,DPL2S         LOAD COMMAND TYPE
2F16  FA05                         42          MOV       3,3,SNR         SKIP IF NOT PRIMARY WRITE TO 9L
2F17  01EC      2F03               43          JMP       DPL1            OTHERWISE, RESTART WRITE TO 9L
2F18  01F3      2F0B               44          JMP       DPL2            WAIT FOR NEW 9L SEQUENCE
                                   45  * SERVICE OUT
2F19  663F                         46  DPL6    HALT                      SO SHOULD NOT OCCUR
2F1A  01E6      2F00               47          JMP       DPL             RESTART IF CONTINUE IS PRESSED
                                   48  * ADDRESS OUT
2F1B  6B1A                         49  DPL7    RLD       1               READ RETURN ADDRESS
2F1C  651A                         50          RAB       0               READ ADDRESS BYTE
2F1D  3940      2F5D               51          LDA       3,DPL8C         LOAD PDP9L ADDRESS
2F1E  ED05                         52          SUB       3,1,SNR         SKIP IF NOT PDP9L
2F1F  393A      2F59               53          LDA       3,DPL4C         OTHERWISE, LOAD NOVA CORE ADDRESS
2F20  E504                         54          SUB       3,0,S2R         SKIP IF NOVA CORE ADDRESS
2F21  0118      2F39               55          JMP       DPL11           OTHERWISE, SEND UNIT CHECK STATUS
2F22  7318                         56          RCB       2               READ COMMAND BYTE
```

```
 PC   OBJ CODE  ADDR    ERRORS   STMT     SOURCE STATEMENT

2F23   39D8     2EFB             57           LDA     3,DPL5S      LOAD WC FOR SECONDARY WRITE
2F24   293A     2F5E             58           LDA     1,DPL9C      LOAD SECONDARY WRITE CODE
2F25   CD05                      59           SUB     2,1,SNR      SKIP IF NOT SECONDARY WRITE
2F26   0106     2F2C             60           JMP     DPL8         OTHERWISE, START SECONDARY WRITE
2F27   2934     2F5B             61           LDA     1,DPL6C      LOAD CA FOR PRIMARY WRITE
2F28   3937     2F5F             62           LDA     3,DPL1D      LOAD PRIMARY WRITE COMMAND CODE
2F29   F50C                      63           SUB#    3,2,SZR      SKIP IF PRIMARY WRITE
2F2A   010B     2F35             64           JMP     DPL10        OTHERWISE, CHECK FOR REWIND/UNLOAD
2F2B   FA91                      65           MOVZR   3,3,SKP      FORM WC FOR PRIMARY WRITE (2) & SKIP
2F2C   29D0     2EFC             66  DPL8     LDA     1,DPL6S      LOAD CA FOR SECONDARY WRITE
2F2D   6C1B                      67           SCA     1            LOAD CURRENT ADDRESS REGISTER
2F2E   7E9B                      68           SDT     3            LOAD WORD COUNT REGISTER AND RHL
2F2F   11CA     2EF9             69           ISZ     DPL3S        INDICATE WRITE FROM 9L
2F30   01DE     2F0E             70           JMP     DPL3         WAIT FOR OF INTERRUPT
2F31   AD30                      71  DPL9     SUBC    1,1          ZERO AC1
2F32   6A1A                      72           LHD     1            CLEAR HIGH PART OF DATA REGISTER
2F33   64DA                      73           SES     0            SEND ENDING STATUS
2F34   0300     0000             74           JMP     0(3)         RETURN
2F35   D205                      75  DPL10    MOV     2,2,SNR      SKIP IF NOT TIO
2F36   212B     2F61             76           LDA     0,DPL7S      OTHERWISE, LOAD NOVA STATUS
2F37   39C1     2EF8             77           LDA     3,DPL2S      LOAD COMMAND TYPE
2F38   FA96                      78           MOVZR   3,3,SEZ      SKIP IF REWIND/UNLOAD NOT EXPECTED
2F39   2127     2F60             79  DPL11    LDA     0,DPL2D      OTHERWISE, LOAD UNIT CHECK STATUS
2F3A   11C0     2EFA             80  DPL12    ISZ     DPL4S        INDICATE NOVA SENT 1ST STATUS
2F3B   09F6     2F31             81           JSR     DPL9         SEND ENDING STATUS
2F3C   01D2     2F0E             82           JMP     DPL3         WAIT FOR EL INTERRUPT
                                 83  * END LINE
2F3D   19BD     2EFA             84  DPL13    DSZ     DPL4S        SKIP IF NOVA SENT 1ST STATUS
2F3E   010B     2F49             85           JMP     DPL14        OTHERWISE, MUST BE PRIMARY WRITE
2F3F   60DB                      86           CTS                  TERMINATE DATAWAY SEQUENCE
2F40   39B8     2EF8             87           LDA     3,DPL2S      LOAD COMMAND TYPE
2F41   FA96                      88           MOVZR   3,3,SEZ      SKIP IF REWIND/UNLOAD NOT EXPECTED
2F42   058A     2EFC             89           JMP  I  DPL6S        OTHERWISE, ENTER NOVALOAD PROGRAM
2F43   19B6     2EF9             90           DSZ     DPL3S        SKIP IF WRITE FROM PDP9L
2F44   01C7     2F0B             91           JMP     DPL2         OTHERWISE, MUST BE TIO FROM 9L
2F45   39B8     2EFD             92           LDA     3,DPL1C      LOAD CE,DE STATUS
2F46   9D05                      93           SUB     0,3,SNR      SKIP IF NOVA DID NOT SEND CE,DE
2F47   11B1     2EF8             94           ISZ     DPL2S        OTHERWISE, ADD ONE TO COMMAND TYPE
2F48   01C6     2F0E             95           JMP     DPL3         WAIT FOR NEW 9L SEQUENCE
2F49   731A                      96  DPL14    RLD     2            READ 9L STATUS
2F4A   21B3     2EFD             97           LDA     0,DPL1C      LOAD CE,DE STATUS
2F4B   3911     2F5C             98           LDA     3,DPL7C      LOAD BUSY STATUS
2F4C   DD05                      99           SUB     2,3,SNR      SKIP IF NOT BUSY STATUS
2F4D   2113     2F60            100           LDA     0,DPL2D      OTHERWISE, LOAD UNIT CHECK STATUS
2F4E   09E3     2F31            101           JSR     DPL9         SEND ENDING STATUS
2F4F   60DB                     102           CTS                  TERMINATE DATAWAY SEQUENCE
2F50   9504                     103           SUB     0,2,SZR      SKIP IF NO ERROR IN DT SEQUENCE
2F51   01B2     2F03            104           JMP     DPL1         OTHERWISE, RESTART WRITE TO 9L
2F52   11A6     2EF8            105           ISZ     DPL2S        INDICATE WANT PRIMARY WRITE FROM 9L
2F53   0188     2F0B            106           JMP     DPL2         WAIT FOR 9L SEQUENCE
                                107  * OVERFLOW
2F54   21A9     2EFD            108  DPL15    LDA     0,DPL1C      LOAD CE,DE STATUS
2F55   675A                     109           SKPDEZ               SKIP IF THERE IS NO DATAWAY ERROR
2F56   210A     2F60            110           LDA     0,DPL2D      OTHERWISE, LOAD UNIT CHECK STATUS
2F57   410A     2F61            111           STA     0,DPL7S      SAVE NOVA STATUS FOR TIO FROM 9L
2F58   01E2     2F3A            112           JMP     DPL12        SEND NOVA STATUS
```

```
  PC   OBJ CODE  ADDR    ERRORS    STMT     SOURCE STATEMENT

2F59   0048                        113  DPL4C    DC       X(48)          NOVA CORE ADDRESS
2F5A   2F61                        114  DPL5C    DC       A(DPL1T-1)     POINTER TO INTERRUPT TABLE
2F5B   2EF8                        115  DPL6C    DC       A(DPL5S)       ADDRESS FOR STORING NOVALOAD WC & CA
2F5C   0010                        116  DPL7C    DC       X(10)          BUSY STATUS
2F5D   0040                        117  DPL8C    DC       X(40)          PDP9L ADDRESS
2F5E   0009                        118  DPL9C    DC       X(09)          SECONDARY WRITE COMMAND
2F5F   0005                        119  DPL1D    DC       X(05)          PRIMARY WRITE COMMAND
2F60   0002                        120  DPL2D    DC       X(02)          UNIT CHECK STATUS
2F61   000C                        121  DPL7S    DC       X(000C)        NOVA STATUS FOR TID FROM 9L
2F62   2F15                        122  DPL1T    DC       A(DPL5)        DISCONNECT
2F63   2F15                        123           DC       A(DPL5)        REPEAT
2F64   2F15                        124           DC       A(DPL5)        NO REPLY
2F65   2F18                        125           DC       A(DPL7)        ADDRESS OUT
2F66   2F19                        126           DC       A(DPL6)        SERVICE OUT
2F67   2F3D                        127           DC       A(DPL13)       END LINE
2F68   2F54                        128           DC       A(DPL15)       OVERFLOW
2F00                               129           END      DPL
```

```
                                                CROSS-REFERENCE

SYMBOL     VALUE  DEFN     REFERENCES

DPL        2F00   0019     0047  0129
DPL1       2F03   0022     0043  0104
DPL1C      2EFD   0016     0022  0092  0097  0108
DPL1D      2F5F   0119     0062
DPL1S      2EF4   0010     0017
DPL1T      2F62   0122     0114
DPL10      2F35   0075     0064
DPL11      2F39   0079     0055
DPL12      2F3A   0080     0112
DPL13      2F3D   0084     0127
DPL14      2F49   0096     0085
DPL15      2F54   0108     0128
DPL2       2F0B   0030     0044  0091  0106
DPL2C      2EFE   0017     0024
DPL2D      2F60   0120     0079  0100  0110
DPL2S      2EF8   0011     0021  0041  0077  0087  0094  0105
DPL3       2F0E   0033     0034  0070  0082  0095
DPL3C      2EFF   0018     0026
DPL3S      2EF9   0012     0031  0069  0090
DPL4C      2F59   0113     0027  0053
DPL4S      2EFA   0013     0032  0080  0084
DPL5       2F15   0041     0122  0123  0124
DPL5C      2F5A   0114     0037
DPL5S      2EFB   0014     0057  0115
DPL6       2F19   0046     0126
DPL6C      2F5B   0115     0061
DPL6S      2EFC   0015     0066  0089
DPL7       2F1B   0049     0125
DPL7C      2F5C   0116     0098
DPL7S      2F61   0121     0076  0111
DPL8       2F2C   0066     0060
DPL8C      2F5D   0117     0051
DPL9       2F31   0071     0081  0101
DPL9C      2F5E   0118     0058

NO STATEMENTS FLAGGED IN THIS ASSEMBLY


IEF142I - STEP WAS EXECUTED - COND CODE 0000
IEF285I   AAELIB.LMODA                                    KEPT
IEF285I   VOL SER NOS= AAE002.
IEF285I   RPB.ASM                                         KEPT
IEF285I   VOL SER NOS= AAE004.
IEF285I   SYS73046.T181908.RV000.PLSBDPL.OBJ              PASSED
IEF285I   VOL SER NOS= AAE008.
IEF285I   SYS73046.T181908.SV000.PLSBDPL.R0000005         SYSOUT
IEF285I   VOL SER NOS= AAE004.
IEF285I   SYS73046.T181908.SV000.PLSBDPL.R0000006         SYSOUT
IEF285I   VOL SER NOS= AAE008.
IEF285I   SYS73046.T181908.RV000.PLSBDPL.R0000007         DELETED
IEF285I   VOL SER NOS= AAE004.
IEF373I STEP /ASM      / START 73046.1821
IEF374I STEP /ASM      / STOP  73046.1821 CPU   0MIN 03.98SEC MAIN  82K LCS   0K
*** CONDITION CODE =  000(HEX)
IEF285I   SYS73046.T181908.RV000.PLSBDPL.OBJ              DELETED
IEF285I   VOL SER NOS= AAE008.
IEF375I  JOB /PLSBDPL / START 73046.1819
IEF376I  JOB /PLSBDPL / STOP  73046.1821 CPU   0MIN 12.46SEC
HIGHEST CONDN CODE =  000(HEX)
*********************************************************************************************************************************
```

APPENDIX C

SOURCE LISTING OF THE NOVADUMP PROGRAM

```
//PLSDMP   JOB (C1C19267,81),DR.P.L.SANGER,                          JOB   2
//             MSGCLASS=D,
//             CLASS=A,
//             TIME=1
//ASM EXEC  NOVASM,PARM='LOAD=OBJ'
XXASM    EXEC  PGM=NOVASM,REGION=120K                                00000010
XXSTEPLIB  DD  DSN=AAELIB.LNODA,DISP=SHR                             00000020
XX         DD  DSN=RP5.ASM,DISP=SHR                                  00000030
XXSYSLOAD  DD  DSN=&&OBJ,UNIT=SYSDA,DISP=(NEW,PASS),                 00000040
XX             SPACE=(TRK,(10,20))                                   00000050
//ASM.SYSPRINT DD SYSOUT=D
X/SYSPRINT DD  SYSOUT=A,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=1210)       00000060
XXSYSPUNCH DD  SYSOUT=D,DCB=(RECFM=FB,BLKSIZE=1200,LRECL=80)         00000070
XXSYSUT1   DD  UNIT=SYSDA,SPACE=(TRK,(5,5))                          00000080
//ASM.SYSIN DD *
IEF236I ALLOC. FOR PLSDMP   ASM      ASM
IEF237I 136   ALLOCATED TO STEPLIB
IEF237I 125   ALLOCATED TO
IEF237I 125   ALLOCATED TO SYSLOAD
IEF237I 134   ALLOCATED TO SYSPRINT
IEF237I 331   ALLOCATED TO SYSPUNCH
IEF237I 135   ALLOCATED TO SYSUT1
IEF237I 311   ALLOCATED TO SYSIN

WORK AREA ALLOCATED (HEX BYTES)=020000
                          CORE=12K
                          XREF=405
                        SYMTAB=135
```

```
 PC   OBJ CODE  ADDR    ERRORS   STMT    SOURCE STATEMENT

                                    1  *
                                    2  *    NOVADUMP
                                    3  *    GENERAL DUMP PROGRAM, USES WORD COUNT, CURRENT ADDRESS SCHEME.
                                    4  *
                                    5          PRINT     HEX
2B00                                6          ORG       X(2B00)
2B00  658F                          7  DMP     IORST                 CLEAR ALL DEVICES AND INTOS
2B01  8530                          8          SUBC      0,0         ZERO AC0
2B02  4161      2B65                9          STA       0,DMP1S     INDICATE PRIMARY WRITE TO 9L
2B03  2165      2B68               10  DMP1    LDA       0,DMP1C     LOAD CONSTANT 12
2B04  661B                         11          SWC       0           LOAD WC REGISTER
2B05  2964      2B69               12          LDA       1,DMP2C     LOAD A(DMP3C)
2B06  6C1B                         13          SCA       1           LOAD CA REGISTER
2B07  3163      2B6A               14          LDA       2,DMP4C     LOAD PRIMARY WRITE TO 9L COMMAND
2B08  3963      2B6B               15          LDA       3,DMP5C     LOAD NOVA CORE ADDRESS
2B09  7C1A                         16          LLD       3           LOAD THIS INTO DATA REGISTER
2B0A  729A                         17          RDS       2           LOAD COMMAND AND RAD
2B0B  FD30                         18  DMP2    SUBC      3,3         ZERO AC3
2B0C  595A      2B66               19          STA       3,DMP4S     INDICATE NOVA DID NOT SEND 1ST STATUS
2B0D  595A      2B67               20          STA       3,DMP5S     INDICATE NOT 9L READ
2B0E  679A                         21  DMP3    SKPIRN                SKIP IF DW INTERRUPT OCCURS
2B0F  01FF      2B0E               22          JMP       DMP3        OTHERWISE, CHECK AGAIN
2B10  6118                         23          RIR       0           READ INTERRUPT REGISTER
2B11  6218                         24          CIR       0           CLEAR INTERRUPT CONDITION
2B12  395A      2B6C               25          LDA       3,DMP6C     LOAD POINTER TO INTERRUPT TABLE
2B13  9E00                         26          ADD       0,3         FORM ADDRESS OF CORRECT ENTRY
2B14  0700      0000               27          JMP    I  0(3)        PROCESS INTERRUPT
                                   28  * DISCONNECT, REPEAT AND NO REPLY
2B15  394E      2B63               29  DMP5    LDA       3,DMP1S     LOAD COMMAND TYPE
2B16  FA05                         30          MOV       3,3,SNR     SKIP IF NOT PRIMARY WRITE TO 9L
2B17  01EC      2B03               31          JMP       DMP1        OTHERWISE, RESTART WRITE TO 9L
2B18  01F3      2B0B               32          JMP       DMP2        WAIT FOR NEW 9L SEQUENCE
                                   33  *  SERVICE OUT
2B19  663F                         34  DMP6    HALT                  SO SHOULD NOT OCCUR
2B1A  01E6      2B00               35          JMP       DMP         RESTART IF CONTINUE IS PRESSED
                                   36  *  ADDRESS OUT
2B1B  681A                         37  DMP7    RLD       1           READ RETURN ADDRESS
2B1C  751A                         38          RAB       2           READ ADDRESS BYTE
2B1D  3951      2B6E               39          LDA       3,DMP8C     LOAD POP9L ADDRESS
2B1E  ED05                         40          SUB       3,1,SNR     SKIP IF NOT POP9L
2B1F  394C      2B6B               41          LDA       3,DMP5C     OTHERWISE, LOAD NOVA CORE ADDRESS
2B20  DD04                         42          SUB       2,3,SZR     SKIP IF NOVA CORE ADDRESS
2B21  010F      2B30               43          JMP       DMP9        OTHERWISE, SEND UNIT CHECK STATUS
2B22  6318                         44          RCB       0           READ COMMAND BYTE
2B23  8294                         45          MOVZR     0,0,SZR     SHIFT COMMAND RIGHT AND SKIP IF ZERO
2B24  828C                         46          MOVR#     0,0,SZR     SKIP IF READ COMMAND
2B25  0109      2B2E               47          JMP       DMP8        OTHERWISE, CHECK FOR REWIND/UNLOAD
2B26  1141      2B67               48          ISZ       DMP5S       INDICATE 9L READ
2B27  3930      2B64               49          LDA       3,DMP2S     LOAD REQUIRED WORD COUNT
2B28  FA05                         50          MOV       3,3,SNR     SKIP IF THIS IS NONZERO
2B29  0108      2B31               51          JMP       DMP10       OTHERWISE, SEND UNIT EXCEPTION STATUS
2B2A  2938      2B65               52          LDA       1,DMP3S     LOAD REQUIRED CURRENT ADDRESS
2B2B  6C18                         53          SCA       1           LOAD CURRENT ADDRESS REGISTER
2B2C  7E98                         54          SCT       3           LOAD WORD COUNT REGISTER AND RHL
2B2D  01E1      2B0E               55          JMP       DMP3        WAIT FOR OF INTERRUPT
2B2E  2135      2B63               56  DMP8    LDA       0,DMP1S     LOAD COMMAND TYPE
```

```
 PC   OBJ CODE  ADDR    ERRORS   STMT    SOURCE STATEMENT
2B2F   8294                       57            MOVZR     0,0,SZR       SKIP IF REWIND/UNLOAD NOT EXPECTED
2B30   2143      2B73             58  DMP9      LDA       0,DMP1D       LOAD UNIT CHECK STATUS
2B31   1135      2B66             59  DMP10     ISZ       DMP4S         INDICATE NOVA SENT 1ST STATUS
2B32   0902      2B34             60            JSR       DMP11         SEND ENDING STATUS
2B33   0108      2B4E             61            JMP       DMP3          WAIT FOR EL INTERRUPT
2B34   AD30                       62  DMP11     SUBC      1,1           ZERO AC1
2B35   6A1A                       63            LHD       1             CLEAR HIGH PART OF DATA REGISTER
2B36   64DA                       64            SES       0             SEND ENDING STATUS
2B37   0300      2B04             65            JMP       0(3)          RETURN
                                  66  *  OVERFLOW
2B38   2130      2B69             67  DMP12     LDA       0,DMP1C       LOAD CE,DE STATUS
2B39   01F8      2B31             68            JMP       DMP10         SEND ENDING STATUS
                                  69  *  END LINE
2B3A   731A                       70  DMP13     RLD       2             READ PDP9L STATUS
2B3B   192B      2B66             71            DSZ       DMP4S         SKIP IF NOVA SENT 1ST STATUS
2B3C   0116      2B52             72            JMP       DMP14         OTHERWISE, MUST BE PRIMARY WRITE
2B3D   60D8                       73            CTS                     TERMINATE DATAWAY SEQUENCE
2B3E   2125      2B63             74            LDA       0,DMP1S       LOAD COMMAND TYPE
2B3F   A294                       75            MOVZR     0,0,SZR       SKIP IF REWIND/UNLOAD NOT EXPECTED
2B40   663F                       76            HALT                    OTHERWISE, HALT - END OF DUMP
2B41   D29D                       77            MOVZR#    2,2,SNR       SKIP IF NOT UNIT EXCEPTION STATUS
2B42   1121      2B63             78            ISZ       DMP1S         OTHERWISE, SHOW REW/UNLOAD EXPECTED
2B43   1924      2B67             79            DSZ       DMP5S         SKIP IF READ FROM 9L
2B44   01C7      2B7B             80            JMP       DMP2          OTHERWISE, WAIT FOR NEW 9L SEQUENCE
2B45   2923      2B68             81            LDA       1,DMP1C       LOAD CE,DE STATUS
2B46   CD04                       82            SUB       2,1,SZR       SKIP IF CE,DE STATUS
2B47   01C4      2B0B             83            JMP       DMP2          OTHERWISE, ALLOW 360 TO REPEAT SEQ
2B48   7D1B                       84            RCA       3             READ CONTENTS OF CA REGISTER
2B49   591C      2B65             85            STA       3,DMP3S       STORE THIS AS NEW CURRENT ADDRESS
2B4A   3128      2B72             86            LDA       2,DMP4D       LOAD STARTING ADDRESS FOR DUMP
2B4B   0D00                       87            SUB       2,3           FORM NO. OF WORDS TRANSFERRED
2B4C   3123      2B6F             88            LDA       2,DMP9C       LOAD CONSTANT 256
2B4D   2924      2B71             89            LDA       1,DMP3D       LOAD NO. OF WORDS TO BE DUMPED
2B4E   ED04                       90            SUB       3,1,SZR       SKIP IF END OF DUMP
2B4F   B512                       91            SUBZ      1,2,SZC       OTHERWISE, SKIP IF.GT.256 TO GO
2B50   4914      2C64             92            STA       1,DMP2S       STORE NEW WC
2B51   01BA      2B0B             93            JMP       DMP2          WAIT FOR READ FROM 9L
2B52   3918      2B60             94  DMP14     LDA       3,DMP7C       LOAD BUSY STATUS
2B53   2115      2B68             95            LDA       0,DMP1C       LOAD CE,DE STATUS
2B54   DD05                       96            SUB       2,3,SNR       SKIP IF NOT BUSY STATUS
2B55   211E      2B73             97            LDA       0,DMP1D       OTHERWISE, LOAD UNIT CHECK STATUS
2B56   09DE      2B34             98            JSR       DMP11         SEND ENDING STATUS
2B57   60DB                       99            CTS                     TERMINATE DATAWAY SEQUENCE
2B58   C504                      100            SUB       2,0,SZR       SKIP IF 9L SENT CE,DE STATUS
2B59   01AA      2B03            101            JMP       DMP1          OTHERWISE, RESTART WRITE TO 9L
2B5A   1109      2B63            102            ISZ       DMP1S         INDICATE READ FROM 9L EXPECTED
2B5B   2117      2B72            103            LDA       0,DMP4D       LOAD STARTING ADDRESS
2B5C   4109      2B65            104            STA       0,DMP3S       STORE THIS STARTING ADDRESS
2B5D   3112      2B6F            105            LDA       2,DMP9C       LOAD CONSTANT 256
2B5E   5106      2B64            106            STA       2,DMP2S       STORE THIS AS WC FOR DATA TRANSFER
2B5F   2912      2B71            107            LDA       1,DMP3D       LOAD NO. OF WORDS TO BE DUMPED
2B60   B512                      108            SUBZ      1,2,SZC       SKIP IF.GT.256 TO BE DUMPED
2B61   4903      2B64            109            STA       1,DMP2S       OTHERWISE, STORE THIS AS WC
2B62   01A9      2B0B            110            JMP       DMP2          WAIT FOR READ FROM 9L
2B63   0000                      111  DMP1S     DC        F(0)          COMMAND TYPE
2B64   0000                      112  DMP2S     DC        F(0)          CURRENT WC
```

NOVA ASSEMBLY

PC	OBJ CODE	ADDR	ERRORS	STMT	SOURCE STATEMENT			
2B65	0000			113	DMP3S	DC	F(0)	CURRENT CA
2B66	0000			114	DMP4S	DC	F(0)	1ST STATUS SWITCH
2B67	0000			115	DMP5S	DC	F(0)	9L READ SWITCH
2B68	000C			116	DMP1C	DC	F(12)	CONSTANT 12 - CE,DE STATUS
2B69	2B70			117	DMP2C	DC	A(DMP3C)	ADDR OF 24 BYTE AREA
2B6A	4005			118	DMP4C	DC	X(4005)	PRIMARY WRITE TO 9L COMMAND
2B6B	0048			119	DMP5C	DC	X(48)	NOVA CORE ADDRESS
2B6C	2B73			120	DMP6C	DC	A(DMP1T-1)	POINTER TO INTERRUPT TABLE
2B6D	0010			121	DMP7C	DC	X(0010)	BUSY STATUS
2B6E	0040			122	DMP8C	DC	X(40)	PDP9L ADDRESS
2B6F	0100			123	DMP9C	DC	F(256)	
2B70	0001			124	DMP3C	DC	X(0001)	INDICATE HEX DUMP TO 360
2B71	3000			125	DMP3D	DC	F(12288)	NO. OF WORDS TO BE DUMPED - 12K
2B72	0000			126	DMP4D	DC	F(0)	STARTING ADDRESS FOR DUMP
2B73	0002			127	DMP1D	DC	X(0002)	UNIT CHECK STATUS
2B74	2B15			128	DMP1T	DC	A(DMP5)	DISCONNECT
2B75	2B15			129		DC	A(DMP5)	REPEAT
2B76	2B15			130		DC	A(DMP5)	NO REPLY
2B77	2B18			131		DC	A(DMP7)	ADDRESS OUT
2B78	2B19			132		DC	A(DMP6)	SERVICE OUT
2B79	2B3A			133		DC	A(DMP13)	END LINE
2B7A	2B38			134		DC	A(DMP12)	OVERFLOW
0000				135		END		

```
                                                    CROSS-REFERENCE

SYMBOL     VALUE  DEFN     REFERENCES

DMP        2B00   [illegible]  0035
DMP1       2B03   0C13     0031  0131
DMP1C      2B68   0116     0010  0067  0081  0095
DMP1D      2B73   0127     0058  0097
DMP1S      2B63   0111     0009  0029  0056  0074  0078  0102
DMP1T      2B74   0128     0120
DMP10      2B31   0059     0051  0068
DMP11      2B34   0062     0060  0098
DMP12      2B38   0067     0134
DMP13      2B3A   0070     0133
DMP14      2B52   0094     0072
DMP2       2B0B   0016     0032  0080  0083  0093  0110
DMP2C      2B69   0117     0012
DMP2S      2B64   0112     0049  0092  0106  0109
DMP3       2B0E   0021     0022  0055  0061
DMP3C      2B70   0124     0117
DMP3D      2B71   0125     0089  0137
DMP3S      2B65   0113     0052  0085  0104
DMP4C      2B6A   0118     0014
DMP4D      2B72   0126     0086  0103
DMP4S      2B66   0114     0019  0059  0071
DMP5       2B15   0029     0128  0129  0130
DMP5C      2B6B   0119     0015  0041
DMP5S      2B67   0115     0020  0048  0079
DMP6       2B19   0034     0132
DMP6C      2B6C   0120     0025
DMP7       2B1B   0037     0131
DMP7C      2B60   0121     0094
DMP8       2B2E   0056     0047
DMP8C      2B6E   0122     0039
DMP9       2B30   0058     0043
DMP9C      2B6F   0123     0088  0105

NO STATEMENTS FLAGGED IN THIS ASSEMBLY


IEF142I - STEP WAS EXECUTED - COND CODE 0000
IEF285I   AAELIB.LMODA                                    KEPT
IEF285I   VOL SER NOS= AAE302.
IEF285I   RPB.ASM                                         KEPT
IEF285I   VOL SER NOS= AAE012.
IEF285I   SYS73137.T191158.RV000.PLSDMP.OBJ               PASSED
IEF285I   VOL SER NOS= AAE012.
IEF285I   SYS73137.T191158.SV000.PLSDMP.R0000001          SYSOUT
IEF285I   VOL SER NOS= AAE008.
IEF285I   SYS73137.T191158.RV000.PLSDMP.R0000003          DELETED
IEF285I   VOL SER NOS= AAE013.
IEF373I STEP /ASM      / START 73137.1912
IEF374I STEP /ASM      / STOP  73137.1912 CPU   0MIN 04.08SEC MAIN 100K LCS   0K
*** CONDITION CODE =  000(HEX)
//LKED  EXEC  NOVALKED,NAME=NOVADUMP
XXLKED   PROC  NAME='EMPNAME,DSN='AMC.NOVALIB'                          00000010
XXLKED   EXEC  PGM=.PNL,PARM='LIST,MAP',REGION=96K,COND=(0,NE)          00000020
XXSYSLIN   DD  DSN=&&OBJ,DISP=(OLD,DELETE)                              00000030
XXSYSLMOD  DD  DSN=&&SYM(&NAME),DISP=SHR                                00000040
IEF653I SUBSTITUTION JCL - DSN=AMC.NOVALIB(NOVADUMP),DISP=SHR
//LKED.SYSPRINT DD SYSOUT=0
X/SYSPRINT DD  SYSOUT=A                                                 00000050
XXSYSUT1   DD  DSN=&SUT1,UNIT=SYSDA,SPACE=(1700,(50,100))               00000060
//
IEF236I ALLOC. FOR PLSDMP   LKED     LKED
IEF237I 125   ALLOCATED TO SYSLIN
IEF237I 126   ALLOCATED TO SYSLMOD
IEF237I 125   ALLOCATED TO SYSPRINT
IEF237I 124   ALLOCATED TO SYSUT1
```

```
F88-LEVEL LINKAGE EDITOR OPTIONS SPECIFIED LIST,MAP
        DEFAULT OPTION(S) USED -  SIZE=(90112,6144)

                                                    MODULE MAP

 CONTROL SECTION                        ENTRY

   NAME     ORIGIN  LENGTH               NAME   LOCATION     NAME   LOCATION     NAME   LOCATION     NAME   LOCATION

 NOVATEXT       00     112

ENTRY ADDRESS       00
TOTAL LENGTH       118

****NOVADUMP  NOW REPLACED IN DATA SET



IEF142I - STEP WAS EXECUTED - COND CODE 0000
IEF285I   SYS73137.T191158.RV000.PLSDMP.OBJ            DELETED
IEF285I   VOL SER NOS= AAE012.
IEF285I   AMC.NOVALIB                                  KEPT
IEF285I   VOL SER NOS= AAE003.
IEF285I   SYS73137.T191158.SV000.PLSDMP.R0000005       SYSOUT
IEF285I   VOL SER NOS= AAE012.
IEF285I   SYS73137.T160811.RV008.INIT.UT1              KEPT
IEF285I   VOL SER NOS= AAE014.
IEF373I STEP /LKED    / START 73137.1912
IEF374I STEP /LKED    / STOP  73137.1912 CPU   0MIN 01.36SEC MAIN  96K LCS   0K
*** CONDITION CODE =  000(HEX)
IEF375I  JOB /PLSDMP  / START 73137.1912
IEF376I  JOB /PLSDMP  / STOP  73137.1912 CPU   0MIN 05.44SEC
HIGHEST CONDN CODE =  200(HEX)
```

C6.