PERKIN-ELMER

COMMON
MULTI-MEDIA DIAGNOSTIC (MMD)
CROSS GENERATOR

Consists of:

Program Description	06-252M95R04A15
Program Listing	06-252M96R04A13

06-252 R04

The Perkin-Elmer Corporation, Computer Systems Division 2 Crescent Place, Oceanport, New Jersey 07757

COMMON MULTI-MEDIA DIAGNOSTIC (MMD) CROSS GENERATOR PROGRAM DESCRIPTION

1. GENERAL

The Common Multi-Media Diagnostic (MMD) Cross Generator, Program Number 06-252, is designed for media generation and verification. The program always generates or verifies a complete output media; no update facilities are provided.

The possible input and output devices are:

- 800 bpi magnetic tape
- 1600 bpi magnetic tape
- 10Mb disk
- 13.5Mb disk
- 80Mb disk
- 300Mb disk
- Floppy disk

The program automatically determines the nature of the input and output devices. If the output device is a magnetic tape or a floppy, the proper boot loader segment is written first so that the media can be loaded using the standard autoload sequence. If the output is a disk, it is created in such a way that it can be boot loaded using the same Loader Storage Unit (LSU) or Automatic Load Option (ALO) used for boot loading an OS disk.

NOTE

The High Performance Tape Drive (HPTD) Controller and 13.5Mb disk are supported on 32-bit systems only.

2. REQUIREMENTS

This is a stand-alone program available on all Multi-Media. The program can be executed on any Perkin-Elmer 16-bit or 32-bit processor that has 16kb or more of main memory.

A console device is required to use this program. The default console is a video display unit (VDU), such as the Model 550, etc., on the X'10'/X'11' PASLA type interface. See Appendix A.

Whatever the input device, the input media must be in the Multi-Media format standard for that media. See Section 6. If the input device is a floppy, a Selector Channel (SELCH) is not used. If the input device is a magnetic tape, SELCH operation is optional. If the input device is a disk, SELCH operation is mandatory.

If the output device is a disk, including floppy, the output media must have been previously formatted. The same rules regarding SELCH operation apply to the output device. Additionally, the output device must be online and not write protected.

3. LOADING PROCEDURES

The Common MMD Cross Generator is available only on Multi-Media.

3.1 Loading from a Multi-Media Magnetic Tape

- Manually key the X'50' sequence shown below into memory.

```
     LOCATION      CONTENTS

       0030          8800     IIPSW    DCX    8800,2000
       0032          2000
       0034          8000              DC     X'8000',Z(IIPSW)
       0036          0030

       0038          8800     MMPSW    DCX    8800,2000
       003A          2000
       003C          8000              DC     X'8000',Z(MMPSW)
       003E          0038

       0050          D500     LOAD     AL     X'CF'
       0052          00CF
       0054          4300              B      X'80'
       0056          0080

       0078          85A1     Magnetic tape device address/
                  or 8540     output command (HPTD)

       007A          0000
       007C          0000     If no SELCH
or     007C          00F0     SELCH address
```

- Select address X'50' and execute. The Multi-Media loader is loaded into memory. The magnetic tape automatically rewinds and the loader starts.

- If the processor is equipped with a hexadecimal display panel, it halts with 'FF' on the display indicators. Enter the sequence number 025 on the display panel and depress EXECUTE. If the processor does not have a hexadecimal panel, the MMD Loader outputs an identifying message to the system console and requests sequence number input. Type the sequence number 025 on the system console, followed by a carriage return (CR). Refer to the Multi-Media Diagnostic (MMD) Loader (32-Bit), publication number 06-176A15 for more details.

3.2 LOADING FROM A MULTI-MEDIA DISKETTE

- Manually key the X'50' sequence shown below into memory.

```
LOCATION   CONTENTS

  0030       8800      IIPSW    DCX    8800,2000
  0032       2000
  0034       8000               DC     X'8000',Z(IIPSW)
  0036       0030
  0038       8800      MMPSW    DCX    8800,2000
  003A       2000
  003C       8000               DC     X'8000',Z(MMPSW)
  003E       0038

  0050       D500      LOAD     AL     X'CF'
  0052       00CF
  0054       4300               B      X'80'
  0056       0080

  0078       C186      Floppy device address/output command
  007A       0000
  007C       0000
```

- Put the diskette containing the Cross Generator in the input floppy drive.

- Select address X'50' and execute. The floppy media loader is loaded into memory.

- If the processor is equipped with a hexadecimal display panel, it halts with X'FF' on the display indicators. Enter the sequence number X'025' on the display panel and depress EXECUTE.

If the processor does not have a hexadecimal display panel, the floppy loader outputs an identifying message to the system console and requests sequence number input. Type the sequence number 025 on the system console, followed by a CR. Refer to the Floppy Diagnostic Loader/Generator Test Program (16-and 32-Bit), publication number 06-225A15 for more details.

3.3 LOADING FROM A MULTI-MEDIA DISK

The Multi-Media disk pack is built in such a way as to be boot-loadable using the ALO or the LSU. This capability can be used on 16-bit or 32-bit systems. The following paragraphs describe the procedures. In all cases, the Multi-Media disk pack must be mounted and hardware write protected.

On processors with displays:

- Set up low memory in the same manner as loading an OS:

LOCATION	CONTENTS	
0030	0000	
0032	0000	
0034	0000	
0036	0050	
0050	D500	Autoload instruction
0052	00CF	
0054	4300	
0056	0080	
0078	1399	Paper tape device number/command
007A	C633	Disk address and OS device code
007C	B6F0	Controller address and SELCH address
007E	0111	OS extension (.111 means MMD)

- If the system does not have an LSU or ALO, mount the OS boot load paper tape (03-074M17 for 32-bit systems or 03-098M17 for 16-bit systems) in the paper tape reader. Select address X'30' and execute. If the system does have an LSU or ALO, only locations X'7A', X'7C', and X'7E' need be set up. Enable the LSU/ALO and depress INITIALIZE.
- As soon as the Initial Program Load (IPL) is complete, X'FFFF' appears on the display panel. Disable the LSU or the ALO.
- Enter the sequence number '025' on the display panel and depress EXECUTE.

On processors without display (1620, etc.):

- Enable the IPL and depress INITIALIZE. Observe that the following is output to the system console:

```
Nonextended Memory          Extended Memory

BASIC TEST COMPLETE         BASIC TEST COMPLETE
MEMORY OK                   MEMORY 00000-07FFF OK
                            MEMORY 08000-0FFFF OK
                                    .
                                    .
                                    .
SERIES SIXTEEN CPUnnKB      where nnKB is the memory size
LOAD DSC1.002?
```

- Type the letter N on the system console. N means no.
- The system then responds with available devices:

```
LOAD DEVICES
DSC2 OK
DSC1 OK
   .
   .
   .
ENTER DEVN.OSID
```

- Enter the following:

```
DSC1.111
```

- The following should be output:

```
MMDL-INPUT SEQUENCE NUMBER
```

- Memory locations X'7A' through X'7E' have been set up by the IPL sequence with default values for the 10Mb removable platter. If necessary, modify these locations and reexecute from address X'4000'.
- Type the sequence number 025 on the system console followed by a CR.

On processors without display (3200, etc.):

- Enable the IPL and depress INITIALIZE. Observe that the following is output to the system console:

```
3200 LSU LOADER R00-00
DEVS
MG85
MGC5
DS5R
DS5F
DS67
D256
FLPY
OTHR
DEVICE=
```

- If the system has default addresses, enter DS5R; otherwise, enter OTHR and the applicable addresses.

Example:

```
DEVICE = OTHR
DEV#   = C6
CODE   = 33
CTLR   = B6
SLCH   = F0
```

- The following should be output:

```
VOL = MMD,FILE =
```

- Enter:

```
OS32MDL2.111
```

- The following should be output:

```
MMDL-INPUT SEQUENCE NUMBER
```

- Memory locations X'7A' through X'7E' have been set up by the IPL sequence with default values for the 10Mb removable platter. If necessary, modify these locations and reexecute from address X'6000'.

- Type the sequence number 025 on the system console followed by a CR.

4. PROGRAM EXECUTION

When loading is complete, refer to Appendix A and set up the console and log device parameters if other than the default PASLA/VDU is desired. Select the program start address, X'A00', and begin execution. Observe that the following title is output to the console device:

```
COMMON MMD CROSS GENERATOR 06-252RXX
*
```

After the program title has been output, an asterisk (*) operator prompt character is output to indicate that the program is ready to receive operator commands. Refer to Appendix B for additional operator console instructions. Appendix C summarizes the possible commands.

4.1 INPUT DEVICE SELECTION

Regardless of the media used to load the Cross Generator itself, the input device defaults to magnetic tape device address X'85' with SELCH operation disabled. If this is the desired input configuration, no commands are required.

4.1.1 Magnetic Tape Input

Use the INDEV option to specify the device address of the MMD master input magnetic tape.

Example:

```
INDEV 95          Selects magnetic tape address X'95'
INDEV 85,F0       Selects magnetic tape address X'85'
                  on SELCH X'F0'
```

The SELCH address can be individually specified using the SELCH1 option. A value of zero causes the program to bypass the SELCH and use sense status program sequences with the magnetic tape.

Example:

```
SELCH1 0          Bypass SELCH operation.
SELCH1 F1         Use SELCH X'F1' for data transfers.
```

4.1.2 Floppy Diskette Input

Use the INDEV option to specify the input floppy address.

```
INDEV C1
```

Use the IDRIVE option to select the input drive. Possible values are 0, 1, 2, or 3.

```
IDRIVE 0
```

4.1.3 Disk Input

Use the INDEV option to specify the device address, controller address, and SELCH address of the MMD master input disk.

```
INDEV C6,B6,F0          X'C6' is the drive address.
                        X'B6' is the controller address.
                        X'F0' is the SELCH address.
```

4.1.4 Programming Notes

It is not necessary to tell the program what kind of device will be used for input. That determination is made automatically based on the device numbers provided by the INDEV command. If the input device is a disk, the kind of disk (10Mb, 13.5Mb, 80Mb, or 300Mb) is also automatically determined.

4.2 OUTPUT DEVICE SELECTION

There is no default output device; therefore, a selection must be made. The output device must be online, at load point if applicable, and not write protected.

4.2.1 Magnetic Tape Output

Use the OUTDEV option to specify the output device address.

Example:

```
OUTDEV 85               Selects magnetic tape address X'85'
OUTDEV C5,F1            Selects magnetic tape address X'C5'
                        on SELCH X'F1'
```

The SELCH address can be individually specified using the SELCH2 option. A value of zero causes the program to bypass the SELCH and use sense status program sequences with the magnetic tape.

Example:

```
SELCH2 0          Bypass SELCH operation.
SELCH2 F0         Use SELCH X'F0' for data transfers.
```

For magnetic tape output, the 0 drive option specifies the type of magnetic tape unit. A value of 0 specifies a standard 800 or 1600 bpi unit. A value of 1 specifies an HPTD controller unit.

4.2.2 Floppy Diskette Output

If the input device is a floppy, the output device is also forced to be a floppy. In this case, no OUTDEV selection is required. The output device has the same device address as the input device. The drive selection is made according to the following example:

Example:

```
INDEV = C1,IDRIVE = 0    OUTDEV = C1,ODRIVE = 1
          IDRIVE = 1                ODRIVE = 0
          IDRIVE = 2                ODRIVE = 3
          IDRIVE = 3                ODRIVE = 2
```

Again, if the input device is a floppy, output device selection, including drive number, is made automatically by the program.

If the input device is not a floppy, the OUTDEV option must be used to select floppy output.

Example:

```
OUTDEV C1
```

Use the ODRIVE option to select the output drive.

Example:

```
ODRIVE 1
```

4.2.3 Disk Output

Use the OUTDEV option to specify the device address, controller address, and SELCH address of the output disk.

```
OUTDEV FC,FB,F2          X'FC' is the drive address.
                         X'FB' is the controller address.
                         X'F2' is the SELCH address.
```

4.2.4 Programming Notes

It is not necessary to tell the program what kind of device will be used for output. That determination is made based on the device numbers provided by the OUTDEV command. If the output device is a disk, the kind of disk (10Mb, 13.5Mb, 80Mb, or 300Mb) is also automatically determined.

4.3 VOLUME NUMBER SELECTION

If the output device is a floppy diskette, use the VOLUME option to specify which diskette in the set is to be generated. The default volume is number 1. The following list shows the present diskette complement.

VOLUME NUMBER	DISKETTE PART NO.	TITLE
1	06-250F01M86	16-BIT TESTS VOLUME 1
2	06-250F02M86	COMMON TESTS VOLUME 1
3	06-250F03M86	COMMON TESTS VOLUME 2
4	06-250F04M86	32-BIT TESTS VOLUME 1
5	06-250F05M86	16-BIT TESTS VOLUME 2
6	06-250F06M86	32-BIT TESTS VOLUME 2
7	06-250F07M86	32-BIT TESTS VOLUME 3

Note that the VOLUMF option equals the functional variation field in the diskette part number. The input magnetic tape or disk contains a volume key for every program so that the generator can pick out the required entries for the selected diskette.

The volume option may also be selected as part of the BUILD or VERIFY commands. If the output device is not a floppy, the VOLUME option is ignored.

NOTE

If the input device is a floppy, there is no choice of output volume.

4.4 MEDIA BUILD

Once the INDEV and OUTDEV options are established, the BUILD command causes duplication to begin.

If the output device is a magnetic tape:

1. The tape is initially rewound.
2. The two boot loaders are written, followed by two file marks.
3. The first program to transfer is sought on the input device.
4. The output magnetic tape is given one backspace file mark command.
5. The Program Definition Block (PDB) is read from the input device. A message line constructed from information in the PDB is output to the list device, showing the progress of the generator. The program definition is then written to the output device.
6. The actual program image is read from the input device and is copied to the output device. As many as 1,024 bytes are transferred at one time.
7. When the complete image has been transferred, two file marks are written to the output tape.
8. The next program to transfer is sought on the input device and the sequence repeats from Step 4.

See Section 6 for details of Multi-Media magnetic tape format.

If the output device is a floppy disk:

1. The diskette capacity is determined. If fewer than 1,941 logical records (each logical record is 128 bytes) are available, the diskette is not usable. An advisory message is output, followed by an operator prompt.
2. The directory is initialized and the boot loader program is written to records 5 and 6.
3. The first program to transfer is sought on the input device and its PDB is read.
4. A message line constructed from information in the PDB is output to the list device showing the progress of the generator. When complete, this listing shows the program complement of the finished diskette.

5. The actual program image is then read from the input device and copied to the output diskette. As many as 1,024 bytes are transferred at a time, but never fewer than 128 bytes.

6. When the complete image has been transferred, the directory on the output diskette is updated.

7. The next program to transfer is sought on the input device and the sequence repeats from Step 4.

See Section 6 for details of Multi-Media diskette format.

If the output device is a disk:

1. The disk is initialized with an OS format volume discriptor and bit map.

2. The first two entries on the input media, sequences 001 and 002, are copied to the output disk. Each is expanded to resemble a linked OS. The first one is called OS16MDL2.111; the second one is called OS32MDL2.111. The OS format directory is then preset to show these two files plus a third file that begins at Cylinder 8 and extends for 12,000 sectors. This third file is called MMD.DAT and represents the actual MMD library. The input device is then rewound.

3. Cylinders 8 and 9 are initialized with zeros. Any defective sectors on Cylinders 8 or 9, or in the volume descriptor, bit map, or OS format directories cause the "defective media" message to be output. Cylinder 8 contains the MMD format directory. The program library starts in Cylinder 9.

4. The first program to transfer is sought on the input device and its PDB is read.

5. A message line constructed from information in the PDB is output to the list device, showing the progress of the generator. The PDB is then written to the output device.

6. The actual program image is then read from the input device and copied to the output device. As many as 1,024 bytes are transferred at a time, but never fewer than 256 bytes.

7. When the complete image has been transferred, the directory on the output disk is updated.

8. The next program to transfer is sought on the input device and the sequence repeats from Step 5.

4.5 MEDIA VERIFICATION

The steps involved in media verification are identical to the build sequence except that, instead of writing to the output device, the device is read and the data is compared to what should be there. It is not necessary to do media verification immediately after a build; several media can be built and then verified against a master.

5. MESSAGES

MESSAGE	MEANING
INVALID INDEV SPECIFICATION	The nature of the input device cannot be determined.
INVALID OUTDEV SPECIFICATION	The nature of the output device cannot be determined.
INDEV/OUTDEV CONFLICT	The input and output device are the same.
DEVICE UNAVAILABLE DEV ddd	Device number ddd is off-line or otherwise unavailable.
DEFECTIVE OUTPUT MEDIA	The output diskette has more than two bad tracks and cannot be used; or the output disk has a defective sector in an OS critical area, or in Cylinders 8 or 9, and cannot be used.
DEVICE WRITE PROTECTED	The output device is write protected.
UNRECOVERABLE ERROR DEV ddd STA ss	After five retrys, device ddd still exhibits bad status ss.
ERROR ON DIRECTORY UPDATE DEV ddd STA ss	Unrecoverable error exists on output diskette directory.
DIRECTORY READ ERROR	Input disk directory is invalid or unrecoverable error exists on diskette directory read.
EOV NOT FOUND	Output disk or diskette directory does not have the proper end of volume marker.
DISKETTE FULL	The current program will not fit on the output diskette.

MESSAGE	MEANING
DIRECTORY FULL	An attempt was made to add more than 127 programs to the output diskette.
DUPLICATE SEQUENCE NUMBER	A non-unique sequence number has been encountered. The problem is with the input media.
END OF VOLUME	End of volume or end of file on the input media, indicating the end of the job.
INVALID DIRECTORY ON OUTPUT DISK	Invalid directory means that the output disk is most likely not an MMD disk. This message occurs only on a VERIFY operation.
BOOT LOADER VERIFY ERROR	This means a nonverify on the output magnetic tape boot loader section or the output diskette boot loader section.
NO SUCH SEQUENCE ON OUTPUT MEDIA	During a verify, the input program cannot be found on the output disk.
PDB VERIFY ERROR	This means a nonverify on the output media in a PDB.
VOLUME DESCRIPTOR VERIFY ERROR	This means a nonverify in the output disk volume descriptor.
BIT MAP VERIFY ERROR	This means a nonverify in the output disk bit map.
OS IMAGE VERIFY ERROR	This means a nonverify in one of the linked MMD loader images.
VERIFY ERROR	This means a nonverify in a program image.

6. MULTI-MEDIA FORMATS

There are three different Multi-Media formats, depending on the media. Magnetic tape, including cassettes, are in one format; floppy diskettes are in another format; and the cartridge disks are in a third format.

6.1 MAGNETIC TAPE MMD FORMAT

Data on a magnetic tape is organized as a number of variable length records. See Figure 1.

The first record on the tape is the boot loader that gets loaded by the autoload sequence. This code loads from X'80' to X'CF'. Upon completion, control is transferred to the boot loader which takes over to load the next record on the tape. That second record is an interim loader that occupies memory from X'100' to X'26F'. The interim loader determines whether the host processor is 16-bit or 32-bit, thereby knowing whether to load the first real program on the tape or the second one. If the host is 16-bit, one forward file mark is issued to the magnetic tape. If it is 32-bit, two forward file mark commands are issued. After skipping the appropriate number of file marks, the interim loader loads the actual MMD Loader. Appendix E is a listing of the two boot loaders.

The MMD Loader is on the tape in the same format as all the other library entries. The first record is the 52-byte PDB. All the information shown on the library printout is contained in this record. The interim loader uses only the low and high address values from the PDB. Memory image of the program follows the PDB. Subsequent records are all 256 bytes except for the last record, which may be less. The last record always has at least four bytes. A file mark separates every program entry on the tape. The occurrence of two file marks in a row signifies the end of the volume.

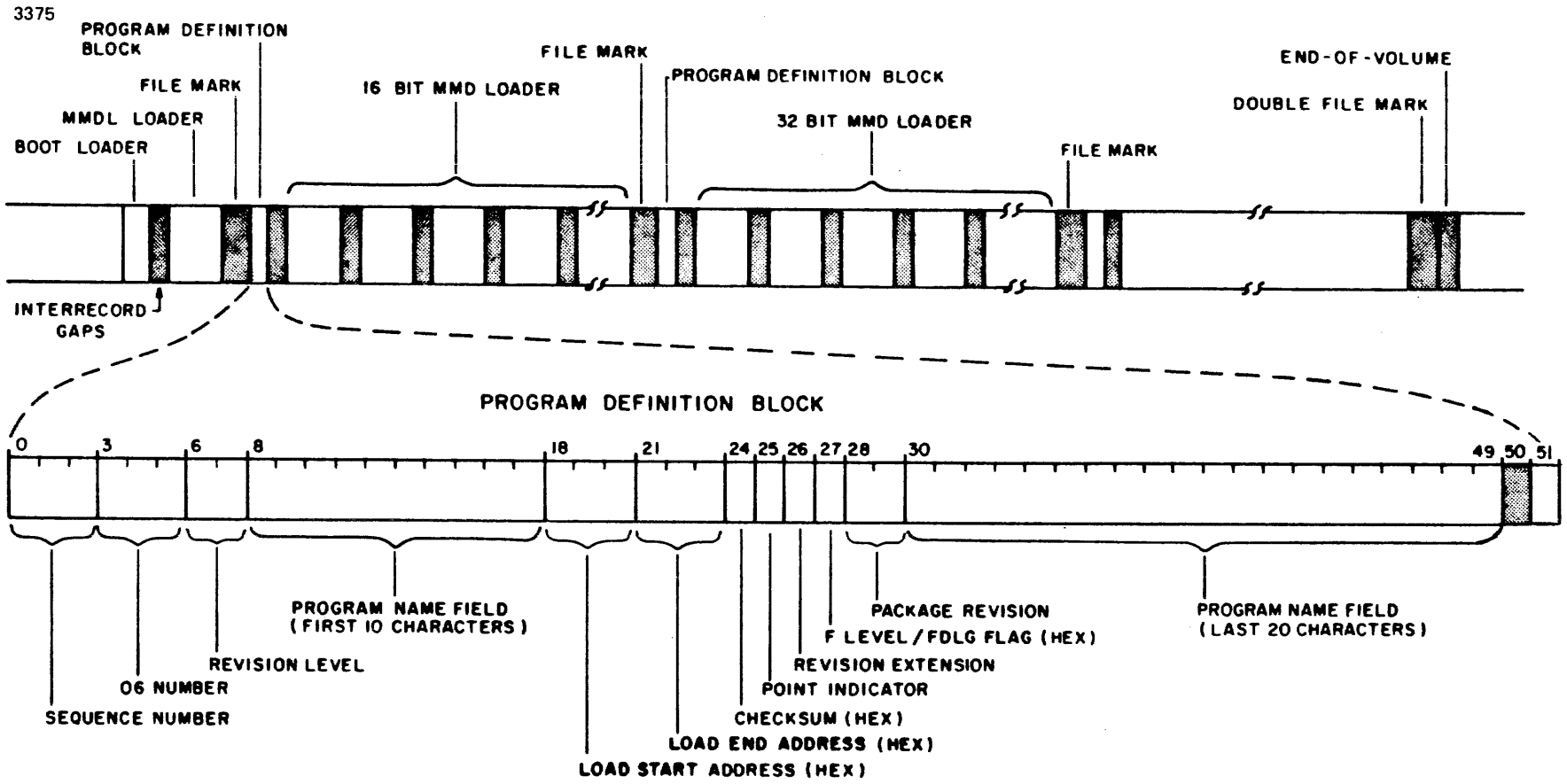

Figure 1 Multi-Media Diagnostic (MMD) Magnetic Tape Format

The PDB contains the following information fields:

INFORMATION FIELD	MEANING
SEQUENCE NUMBER	This is the library sequence number, in ASCII, for this program.
06 NUMBER	This is the test program part number, less the 06- prefix. For example, value in ASCII of 101 means 06-101.
REVISION LEVEL	This is the 2-digit revision level of the program object in ASCII.
PROGRAM NAME	This 30-byte field is separated into two fields of 10 and 20 bytes each, where the descriptive name for the program in ASCII is found.
LOAD START ADDRESS	This field contains the hexadecimal address of the first byte of the program. Up to six digits are allowed.
LOAD END ADDRESS	This field contains the hexadecimal address of the last byte of the program. It is always an odd address, and the difference between start and end addresses, modulo 256, is always greater than 3.
CHECKSUM	This field contains a 1-byte Exclusive-or checksum of every byte in the program image.
POINT INDICATOR	This field contains either zeros or X'2E', an ASCII period. If zero, the REVISION EXTENSION field is also zero. If equal to a period, the REVISION EXTENSION field contains an ASCII digit from 1 through 9.
REVISION EXTENSION	This field is used to indicate that the program image has a patch incorporated in it. The object revision then reads R01.2, for example, to indicate that two patches are incorporated. This object revision is also indicated on change page patch information sheets in the individual test program packages.

INFORMATION FIELD	MEANING
F LEVEL/FDLG FLAG	This byte is separated into two 4-bit fields. The first identifies the object functional variation. For example, a value of 0001 would identify 06-214F01. A value of zero states no functional variation. The second four bits identify the floppy diskette that this same program also appears on. A value of 00002 indicates that this program does not appear on any diskette; a value of 00012 states that this program appears on the F01 diskette. See Section 4.3. A value of 1111 indicates that this program goes on every diskette.
PACKAGE REVISION	This field identifies the revision level in ASCII of the individual test program package associated with this object.

NOTE

F LEVEL/FDLG FLAG and PACKAGE REVISION are not createable or modifiable by any of the present stand-alone Multi-Media generators. Special Multi-Medias built using the 06-177 or 06-225 generator will not contain these information fields.

6.2 FLOPPY DISK MMD FORMAT

Floppy disk is a directory device. It is not sequential like a magnetic tape; rather, there is a directory that tells the loader where each library entry begins. Refer to Figure 2.

The diskette surface is divided into 77 concentric tracks. Each track is divided into 26 sectors of 128 bytes each. Each sector constitutes one logical record. Logical records are numbered sequentially from 1 to a maximum of 2,002. Any defective tracks are excluded from the sequence; that is, if Track 2 is defective, logical record number (LRN) 53 is assigned to Track 3 instead. This is handled by the microprocessor controlled interface.

Track 0, LRN 1 and 2, is reserved for an OS volume descriptor (there is none on MMD). LRN 5 is where the boot loader starts. This boot loader is like the interim loader on the magnetic tape, except that it can be loaded directly with autoload instruction. When control is transferred to it, it first loads the rest of itself; it then determines if the host processor is 16- or 32-bit and, based on that determination, loads either program sequence number 1 or 2. Appendix F is a listing of the floppy boot loader.

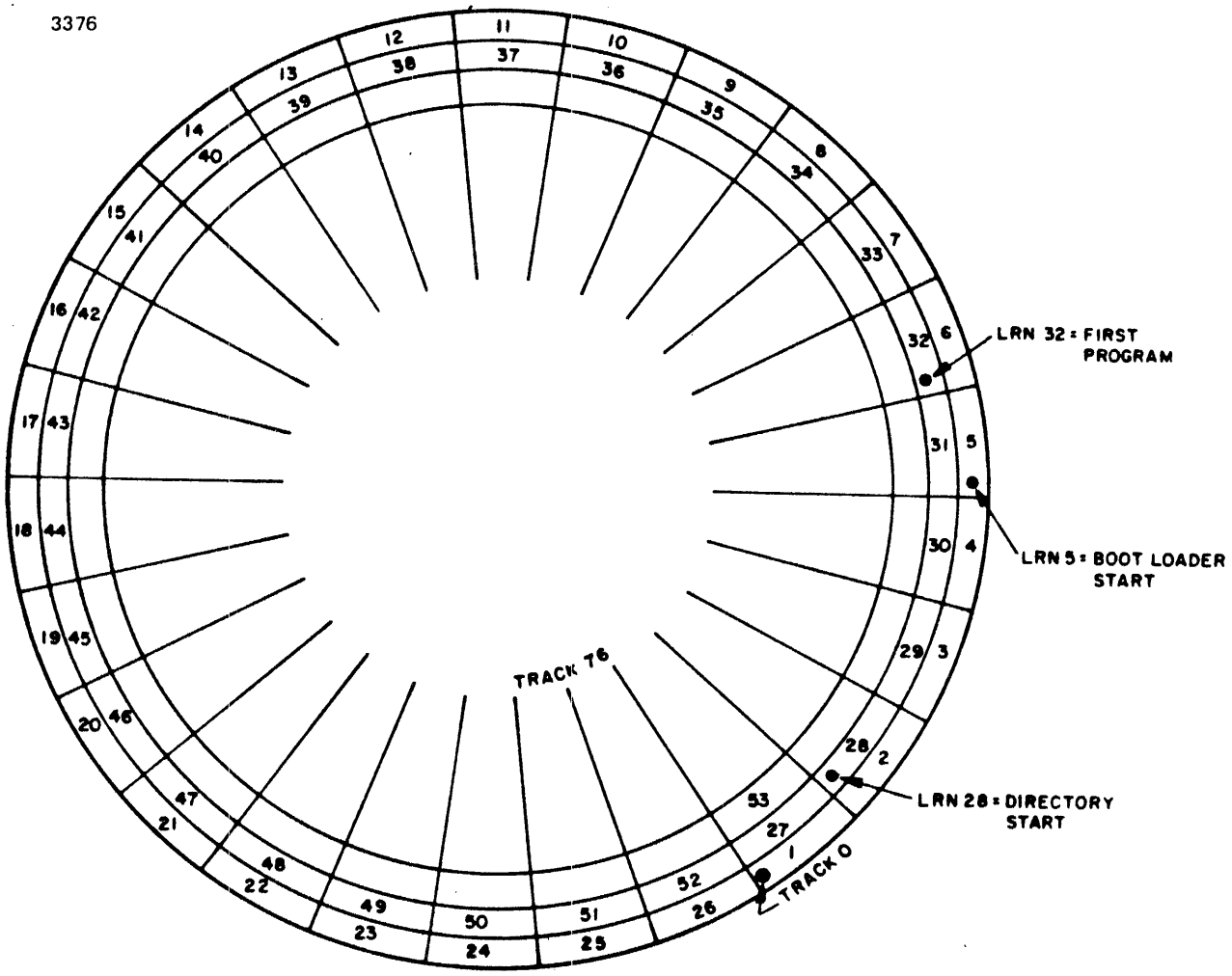

Figure 2 Floppy Disk MMD Format

LRN 28, 29, 30, and 31 constitute the directory. LRN 32 is where the first program goes. The directory consists of two halfwords per entry, allowing for a maximum of 127 programs. The last two halfwords have to be zero to mark the end of the directory. The first halfword of an entry contains the library sequence number in hexadecimal. The second halfword specifies the LRN where the corresponding program starts. Refer to Figure 3.

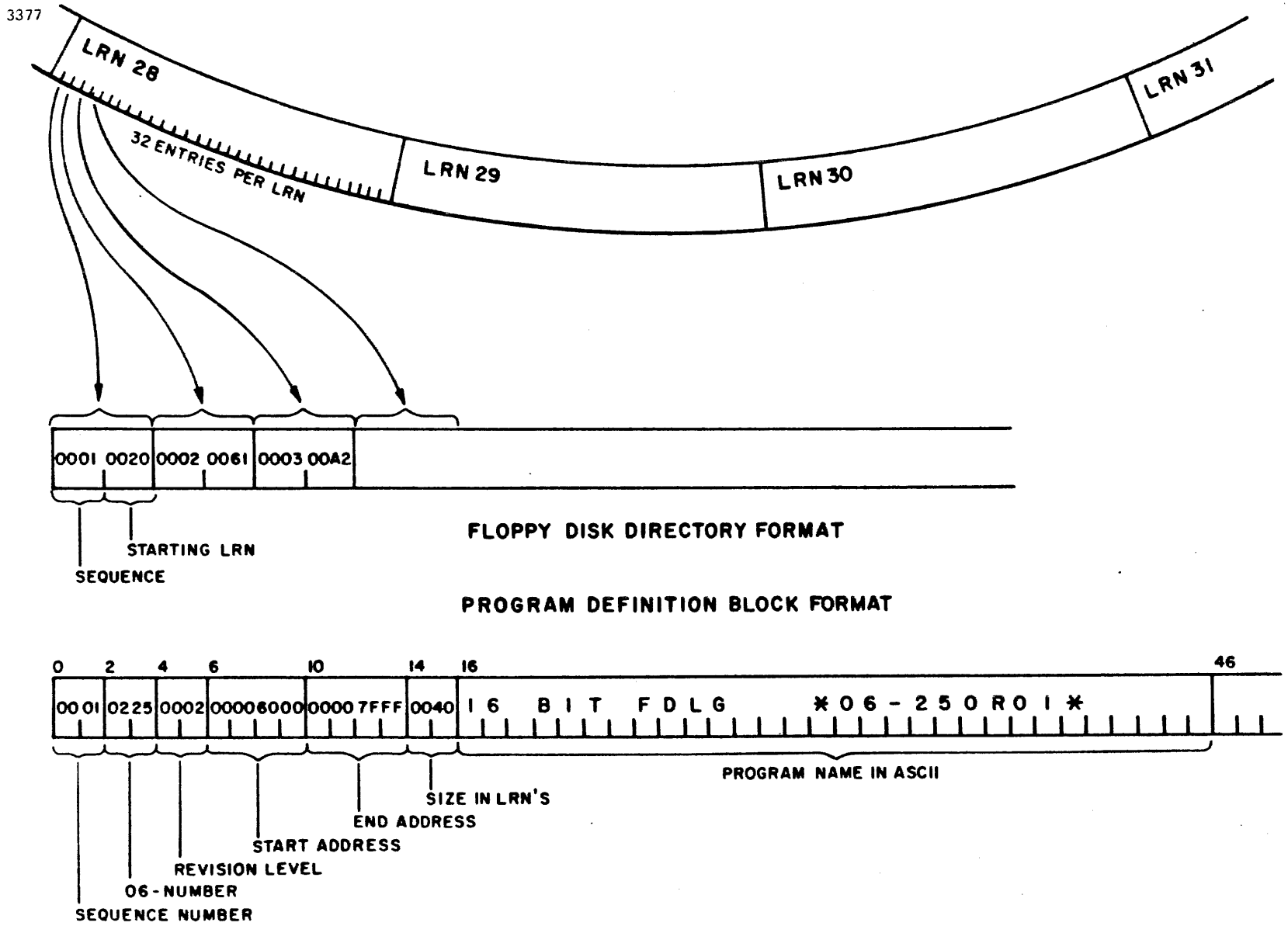

Figure 3 Floppy Disk Directory and PDB Format

Each program consists of some number of 128-byte LRNs. No partial LRNs are allowed. The first LRN of each program entry is the PDB. This PDB format is different from that used on magnetic tape.

Unlike the magnetic tape PDB, all information in the floppy PDB is in hexadecimal, except for the ASCII name. Only the first 46 bytes in the 128-byte LRN are used. The present Floppy Diagnostic Loader Generator does not allow more than 46 useful bytes; therefore, the additional information available in the magnetic tape PDB could not be included.

6.3 CARTRIDGE DISK MMD FORMAT

The cartridge disks are also directory devices. Instead of working with LRNs, as with floppy, the generator uses track, head, and sector number to identify areas on the disk. These are also translated into logical block addresses for use in the OS format part of the disk. The terms "track" and "cylinder" are often used interchangeably. A cylinder represents the same track on each recording surface of the disk assembly. There may be two or more recording surfaces, each with its own record/playback head. In practice, software writes a complete cylinder -- Track 0, Head 0; then Track 0, Head 1 -- before stepping to the next cylinder. This is done for timing reasons. It is faster to switch heads than to seek another track.

The generator builds the disk so that it can be boot loaded using the same LSU or ALO used to boot load an OS pack. This means that the first few cylinders have to be initialized to resemble an OS initialized pack. The generator puts the following items on the pack:

- Volume descriptor, one sector
- Bit map, many sectors
- OS directory, one sector
- OS inverted list, one sector
- OS secondary directory, one sector
- The program from the input media, whose sequence number is 001, expanded to look like a TET'ed 16-bit OS.
- The program from the input media, whose sequence number is 002, expanded to look like a TET'ed 32-bit OS.

Note that a program on MMD and the output of TET are both in image form, so the expansion involves filling in data to occupy Memory Location 0 up to the actual start address of the input program. The OS boot loader loads memory upward from Address 0.

Table 1 summarizes the data and its location in the OS format portion of the disk.

TABLE 1 OS DATA POSITIONS

3380

ITEM	DISK TYPE	SIZE IN SECTORS	STARTING LOGICAL BLOCK ADDRESS	STARTING CYLINDER	STARTING HEAD	STARTING SECTOR	NUMBER OF BITS SET (SECTORS USED)
Volume Descriptor	13.5Mb	1	0	0	0	0	
	10Mb	1	0	0	0	0	
	80Mb	1	0	0	0	0	
	300Mb	1	0	0	0	0	
Bit Map	13.5Mb	26	1	0	0	1	12,512
	10Mb	10	1	0	0	1	12,384
	80Mb	129	1	0	0	1	14,560
	300Mb	489	1	0	0	1	21,728
OS Directory	13.5Mb	1	27	0	0	27	
	10Mb	1	11	0	0	11	
	86Mb	1	130	0	2	2	
	300Mb	1	490	0	7	42	
OS Inverted List	13.5Mb	1	28	0	0	28	
	10Mb	1	12	0	0	12	
	80Mb	1	131	0	2	3	
	300Mb	1	491	0	7	43	
OS Secondary Directory	13.5Mb	1	29	0	0	29	
	10Mb	1	13	0	0	13	
	80Mb	1	132	0	2	4	
	300Mb	1	492	0	7	44	
First OS Image	13.5Mb	80	30	0	0	29	
	10Mb	80	14	0	0	14	
	80Mb	80	133	0	2	5	
	300Mb	80	493	0	7	45	
Second OS Image	13.5Mb	112	110	1	0	46	
	10Mb	112	94	1	1	22	
	80Mb	112	213	0	3	21	
	300Mb	112	573	0	8	61	
Third File Actual MMD Library	13.5Mb	12,194	122	3	0	30	
	10Mb	12,178	206	4	0	14	
	80Mb	14,235	325	1	0	5	
	300Mb	21,043	685	0	10	19	

Cylinder 8 on the disk is where the MMD format directory starts. See Figures 4 and 5. The present library needs only five sectors of directory. Each sector in the directory is divided into 32 8-byte blocks. The first block of the first sector contains all E's to indicate a properly initialized MMD disk. Each subsequent block contains an ASCII sequence number, followed by a cylinder, sector, and head number where that program physically starts on the disk. In this example, Program Sequence 001 starts at Cylinder 9, Sector 0, Head 0. Program Sequence 002 starts at Cylinder 9, Sector 17, Head 0.

When the MMD loader is asked to load Program Sequence 2, the disk is "seeked" to the appropriate starting point (Cylinder 9, Sector 17, Head 0) and loading begins. The first sector of a program entry contains the PDB. Only the first 52 bytes of the 256-byte sector are used and the PDB format is the same as for a magnetic tape. The loader uses the load start and end addresses contained in the PDB to determine how many sectors to read. All transfers involve 256 bytes. No partial sectors are transferred.

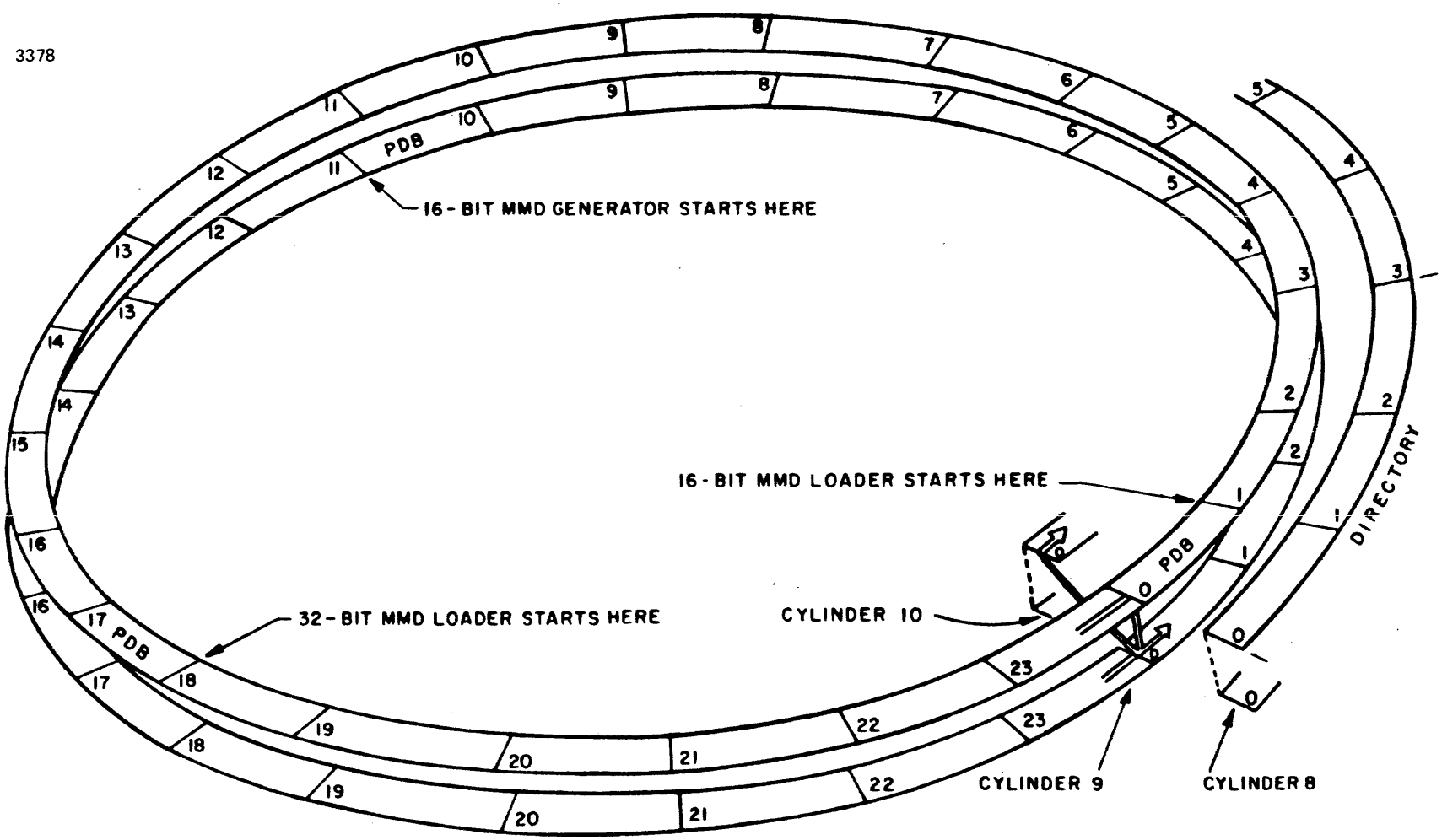

Figure 4 Multi-Media Diagnostic (MMD) 10Mb Disk Format

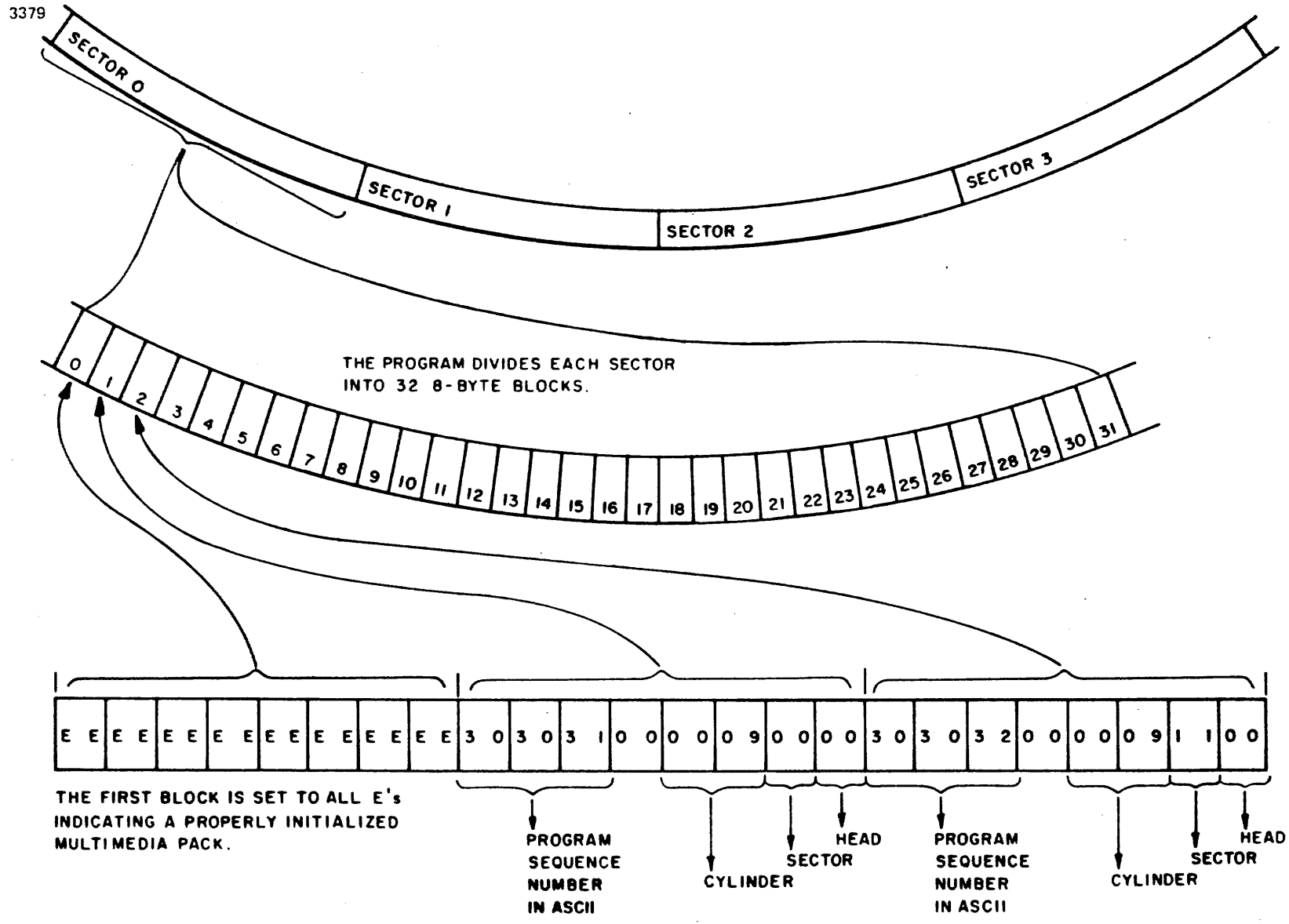

Figure 5 Multi-Media Diagnostic (MMD) Disk Directory Format

APPENDIX A
USER DEVICE DEFINITION

ASCII INPUT/OUTPUT DEVICE SUPPORT

The executive of the program uses the concept of console I/O device and list device. The console I/O device is an interactive device which is capable of logging messages and accepting commands and other user input. When the executive is accepting input from the user, or sending messages to the user, the console device is used. When the test program is running, the list device is used for logging messages.

IO HALFWORD CONTROL OF I/O DEVICE SELECTION

The list device and console device are specified to the executive by the contents of the halfword "IO" at ORIGIN1+X'10' (normally X'0A10'). The interpretation of this data is detailed in Table A-1. The executive allows only the identifiers shown and changes illegal identifiers to X'01'.

TABLE A-1 INPUT/OUTPUT IDENTIFIERS

	0 7	8 15
IO	CONSOLE DEVICE IDENTIFIER	LIST DEVICE IDENTIFIER
	X'01' - VDU on PASLA/PALM or COMM MUX interface	X'01' - VDU on PASLA/PALM or COMM MUX interface
	X'02' - Device on current loop interface	X'02' - Device on current loop interface
	X'03' - Reserved - Changed to X'01'	X'03' - Line printer on line printer interface
	X'04' - Device on PASLA/ PALM or COMM MUX interface	X'04' - Device on PASLA/ PALM or COMM MUX interface
	X'05' - VDU on micro-I/O bus interface	X'05' - VDU on micro-I/O bus interface

I/O DEVICE ADDRESSES AND CHARACTERISTICS

The device types implied by the values contained in the IO halfword are described in the following paragraphs. For each of the devices, including device type X'03', termination of an output line results in a carriage return (CR), line feed, and null character being output by the executive (X'0D', X'0A', X'00').

Devices identified by X'01' are assumed to be on a full-duplex asynchronous RS232C-type interface with addresses X'010' and X'011' for read and write sides, respectively. Examples of such interfaces are PASLA, PALM, and COMM MUX. The executive programs these devices for highest clock rate, seven data bits, two stop bits, and even parity. If the terminal is set up differently, location CRT2ND must be modified accordingly. Line break status is assumed to be indicated by framing-error status, with BUSY not active, and a zero character in the receive buffer. Offline status is assumed to be X'0C' (BUSY+EXAMINE STATUS).

Devices identified by X'02' are assumed to be on a Teletype-compatible current loop interface with address X'002'. The executive programs these devices for unblocked mode (echoplex). Line break status is assumed to be indicated by framing-error status. Offline status is assumed to be X'01' (device unavailable). If this bit is set, other status bits are "don't-cares".

The list device identified by X'03' is assumed to be a line printer on a line printer interface with address X'062'. Offline status is assumed to be X'01' (device unavailable). If this bit is set, other status bits are "don't-cares".

Devices indicated by X'04' are assumed to be attached as described for device type X'01', having the capability of transmitting DC4 and DC2 transmission pause and resume requests. An example of such a device is the Perkin-Elmer Carousel 300 terminal.

Devices indicated by X'05' are assumed to be on a micro-I/O bus interface with address X'0C0'. These devices are programmed for blocked mode (full-duplex). Line break is assumed to be indicated by framing-error status which is not testable if a character is in the interface read buffer. Offline status is assumed to be X'01' (device unavailable). If this bit is set, other status bits are "don't-cares".

The IO halfword, described above, controls which device identifiers are used when the program is started. The default data in this halfword is X'0101'. If this value does not indicate the desired type of I/O device, of the types supported, the data in the IO halfword may be modified before starting program execution.

If the default device addresses are not the addresses of the devices configured in the system, the table of device addresses found in the source program, adjacent to the IO halfword, may be modified. There are two halfword entries used for each type of device. The first is the read-side address, and the second is the write-side address. Both of these halfwords must be modified for any change required. If the device type has only one address (for example, a line printer), the device address must be placed in each of the two appropriate halfwords. The executive always uses the read-side address to test offline status.

APPENDIX B
COMMAND/OPTION INPUT

An asterisk (*) operator prompt is output to the console device to indicate that the program is waiting for user input. All option names must be typed in from the console, followed by a carriage return (CR) if there are no arguments, or if default arguments are to be used. If arguments are required, the option name must be followed by a space; then the desired argument or arguments are separated by commas. A CR must be used to signal the end of every option/command input.

An invalid command/option name or option value causes a question mark (?) to be output, followed by a CR, line feed, and an asterisk (*) prompt. If, during command/option input, a mistake is made, the hash mark (#) can be typed to delete the entire command line. A CR, line feed, and new prompt is output. The left arrow (<--) can be typed to delete the previously typed character, or a string of characters can be deleted by typing a left arrow for each character to be deleted. The backspace character and delete character are treated the same as a back arrow.

APPENDIX C
OPTION/COMMAND SUMMARY

Examine each option in the following list. If a default value is specified and is the value desired, no action is necessary; otherwise, the option must be entered.

OPTION	DEFAULT	DESCRIPTION
INDEV	0085	Specifies the MMD input device. One, two, or three arguments are allowed. If two arguments are input, the second argument is taken as the value for SELCH1. If three arguments are input, the second argument is taken as the disk controller address associated with the first argument drive address, and the third argument is taken as the value for SELCH1.
SELCH1	0000	Specifies the SELCH to use with INDEV. Note that a value for SELCH1 can also be specified with the INDEV option.
OUTDEV	0000	Specifies the output device. As with the INDEV option, one, two, or three arguments are allowed. If more than one argument is input, the last is taken as the value for SELCH2.
SELCH2	0000	Specifies the SELCH to use with OUTDEV. Note that a value for SELCH2 can also be specified with the OUTDEV option.
IDRIVE	0000	Specifies the input floppy drive number if the input device is a floppy. Ignored if the input is not a floppy.
ODRIVE	0000	Specifies the output floppy drive number if the output device is a floppy. The IDRIVE option is forced to a value equal to ODRIVE Exclusive-OR 1 when the input device is a Floppy.
VOLUME	0001	Specifies the test program complement to be written to the output diskette. Values from 1 to X'F' are accepted. Ignored if OUTDEV is not a floppy.

OPTION	DEFAULT	DESCRIPTION
OPTION	N/A	This command causes the program to display all options with their current value.
BUILD	VOLUME	This command starts the media duplication process. If no argument is given, the value of the VOLUME option specifies the diskette to build, if OUTDEV is a floppy. If an argument is given, its value replaces the value of the VOLUME option to specify which diskette to build.
VERIFY	VOLUME	This command starts the media verification process. As with the build option, if no argument is given, the value of VOLUME applies. If an argument is given, it replaces the value of VOLUME to specify which diskette to build.
CON	N/A	This command results in execution of a breakpoint instruction to return control to the console. If issued on a non-1600 or 3200 machine, an illegal instruction results.
BUILDV	VOLUME	This command combines the BUILD and VERIFY functions. Media duplication is followed immediately by media verification. No listing is generated until the verify phase.

APPENDIX D
SAMPLE COMMAND SEQUENCE

```
*OPTION

INDEV       0085
SELCH1      0000
OUTDEV      00C1
SELCH2      0000
IDRIVE      0000
ODRIVE      0000
VOLUME      0001

*BUILD 3

001  06-225F01R02    02   16-BIT FDLG                  006000 007FFF
002  06-225F02R02    02   32-BIT FDLG                  006000 007FFF
003  06-225F01R02    02   16-BIT FDLG (HIGH)           00D000 00FFFF
004  06-225F02R02    02   32-BIT FDLG (HIGH)           00D000 00FFFF
400  06-004    R09.1 10   COMMON TELETYPE BCT           000A00 001FFF
413  06-243    R00   00   MODEL 550 VDU TEST (F01)     000A00 0025FF
414  06-243    R00   00   MODEL 550 VDU TEST (F02)     000A00 0025FF
449  06-173F01R02    05   COMMON DISK TEST (16-BIT)    000A00 003FFF
450  06-173F01R02.1  05   CMN DISK TEST (32-BIT)       000A00 003FFF
451  06-173F02R02    05   COMMON DISK FORMATTER        000A00 003FFF
452  06-208    R00   00   COMMON 40 Mb FORMATTER       000A00 002BBF
453  06-198    R00.3 04   CMN FLOPPY TEST              000A00 003FFF
454  06-200F01R01.1  02   S16 MSM DISK TEST           000A00 003FFF
455  06-200F02R01.1  02   S32 MSM DISK TEST           000A00 003FFF
456  06-201    R02   03   COMMON MSM FORMATTER         000A00 003299
457  06-172    R03   07   COMMON MAG TAPE TEST         000A00 003FFF
458  06-171    R01   02   COMMON CASSETTE TEST         000A00 003CFF
500  06-127    R05.1 08   CMN PALS OFF-LINE TEST      000A00 0038FF
519  06-196    R00.1 01   CMN 360/370 INF (32-BIT)    000500 004471
520  06-196    R00   01   COMMON 360/370 INF TEST     000500 004471

END OF VOLUME
*
```

APPENDIX E
MAG TAPE BOOT LOADER


```
PROG= *NONE*      ASSEMBLED BY CAL 03-066R08-00   (32-BIT)

      MAG TAPE BOOT LOADER

          0000 0000              2  R0       EQU   0
          0000 0001              3  R1       EQU   1
          0000 0002              4  R2       EQU   2
          0000 0003              5  R3       EQU   3
          0000 0004              6  R4       EQU   4
          0000 0005              7  R5       EQU   5
          0000 0006              8  R6       EQU   6
          0000 0007              9  R7       EQU   7
          0000 0008             10  R8       EQU   8
          0000 0009             11  R9       EQU   9
          0000 000A             12  R10      EQU   10
          0000 000B             13  R11      EQU   11
          0000 000E             14  R14      EQU   14
          0000 000F             15  R15      EQU   15

   0000R                        17            ORG   X'80'
   0080   D310 0078             18  MTBOOT    LB    R1,X'78'          GET TAPE DEVICE ADDRESS
   0084   2420                 19            LIS   R2,0
   0086   D330 007D            20            LB    R3,X'7D'          PICK UP SELCH ADDRESS
   008A   9423                 21            EXBR  R2,R3             (R2)='SS00' OR '0000SS00'
   008C   EC20 0008            22            SRL   R2,8              IF 16 BIT, (R2,R3)='00SS,0000'
                                23  *                                 IF 32 BIT, (R2)=(R3)='000000SS'
   0090   9019                 24  MTBOOT1   SSR   R1,R9             STATUS CHECK
   0092   919C                 25            SLHLS R9,12             NO MOTION BIT TO CARRY
                                26  *                                 LS BYTE OF R9 = 00
   0094   2282                 27            BNCS  MTBOOT1           WAIT FOR NO-MOTION
   0096   C850 0100            28            LHI   R5,X'100'         LOAD START ADDRESS
   009A   2461                 29            LIS   R6,1              BXLE INCREMENT
   009C   C870 026F            30            LHI   R7,MTLOADE-MTLOADS+X'100' LOAD END ADDRESS
   00A0   C800 0030            31            LHI   R0,X'30'          SELCH READ COMMAND
   00A4   2448                 32            LIS   R4,8              SELCH STOP COMMAND
   00A6   9E24                 33            OCR   R2,R4             SELCH STOP
   00A8   9A39                 34            WDR   R3,R9             MS BYTE OF 3 BYTE ADDRESS
                                35  *                                 R3=SELCH ADDRESS IF 32 BIT HOST
   00AA   9825                 36            WHR   R2,R5             LS 2 BYTES OF START ADDRESS
   00AC   9A39                 37            WDR   R3,R9             MS BYTE OF 3 BYTE END ADDRESS
   00AE   9827                 38            WHR   R2,R7             LS 2 BYTES OF END ADDRESS
   00B0   DE10 0079            39            OC    R1,X'79'          MAG TAPE WRITE
   00B4   9E20                 40            OCR   R2,R0             SELCH GO
   00B6   2145                 41            BOS   MTBOOT2           FALSE SYNC = NO SELCH
   00B8   9D29                 42            SSR   R2,R9             ELSE, WAIT FOR SELCH NOT BUSY
   00BA   2081                 43            BTBS  8,1               LOOP ON BUSY
   00BC   9E24                 44            OCR   R2,R4             SELCH STOP
   00BE   0305                 45            BR    R5                BRANCH, START CODE JUST LOADED
   00C0   9D19                 46  MTBOOT2   SSR   R1,R9             MAG TAPE STATUS
   00C2   2081                 47            BTBS  8,1               LOOP ON BUSY
   00C4   DB15 0000            48            RD    R1,0(R5)          READ A BYTE
   00C8   C150 00C0            49            BXLE  R5,MTBOOT2        DECREMENT INDEX & LOOP
   00CC   4300 0100            50            B     X'100'            GO TO LOAD START ADRS
```

MAG TAPE BOOT LOADER

```
                                52  * THE FIRST PART OF THE BOOT LOADER RESIDES IN MEMORY
                                53  * FROM LOCATION X'80' TO X'CF'.  IT IS LOADED BY THE
                                54  * X'50' SEQUENCE.  WHEN CONTROL IS TRANSFERRED TO IT,
                                55  * THAT CODE READS IN THIS NEXT SECTION WHICH STARTS
                                56  * AT ADDRESS X'100'.

                                58  * ON INPUT, (R0)=SELCH READ COMMAND, X'30'
                                59  *           (R1)=MAG TAPE DEVICE ADDRESS
                                60  *           (R2)=SELCH ADDRESS
                                61  *           (R3)=0 IF 16 BIT HOST
                                62  *           (R3)=SELCH ADDRESS IF 32 BIT HOST
                                63  *           (R4)=SELCH STOP COMMAND, X'08'
                                64  *           (R6)=1
                                65  *
                                66  *
0000                            67          ORG   X'100'
0100  C810 8000                 68  MTLOADS LHI   R1,X'8000'          FOR TARGET CHECK
0104  0A11                      69          AAR   R1,R1                 =0000 OR FFFF0000
0106  0631                      70          OAR   R3,R1                 =0000 OR FFFF00SS
0108  0310 0078                71          LB    R1,X'78'              GET TAPE DEVICE ADDRESS
010C  9D19                      72  MTLOADS0 SSR  R1,R9                 MAG TAPE STATUS
010E  C390 0010                 73          THI   R9,X'10'              TEST FOR NO MOTION
0112  2253                      74          BZS   MTLOADS0              WAIT FOR IT
0114  C8F0 A023                75          LHI   R15,X'A023'           FF COMMANDS
0118  C8E0 C9C0                76          LHI   R14,X'C9C0'           DISARM COMMANDS
011C  0390 0079                77          LB    R9,X'79'              GET READ COMMAND
0120  C590 00A1                78          CLHI  R9,X'A1'              800/1600 BPI?
0124  2333                      79          BES   MTLOADS1              SKIP IF YES
0126  90F8                      80          SRHLS R15,8                 FF COMMAND FOR 6250
0128  90E8                      81          SRHLS R14,8                 DISARM COMMAND FOR 6250
012A  9E1E                      82  MTLOADS1 OCR  R1,R14                DISARM
012C  9E1F                      83          OCR   R1,R15                FORWARD FILE MARK
012E  9D19                      84  MTLOAD2 SSR   R1,R9
0130  C390 0010                 85          THI   R9,X'10'              NO MOTION?
0134  2233                      86          BZS   MTLOAD2               NO, WAIT
0136  0833                      87          LDAR  R3,R3                 CHECK HOST
0138  2336                      88          BZS   MTLOAD4               SKIP IF 16 BIT HOST
013A  9E1F                      89          OCR   R1,R15                ELSE, ANOTHER FORWARD FILE
013C  9D19                      90  MTLOAD3 SSR   R1,R9
013E  C390 0010                 91          THI   R9,X'10'              NO MOTION CHECK
0142  2233                      92          BZS   MTLOAD3
                                93  *                                   TAPE IS NOW IN POSITION AT THE
                                94  *                                   BEGINNING OF THE FIRST OR SECOND
                                95  *                                   PROGRAM, DEPENDING ON THE HOST.
                                96  *                                   NEXT, READ IN THE 52 BYTE
                                97  *                                   PROGRAM DEFINITION BLOCK (PDB)
                                98  *
0144  C850 0270                 99  MTLOAD4 LHI   R5,BOOTFN-MTLOADS+X'100' START ADRS
0148  C860 02A3                100          LHI   R6,BOOTFN-MTLOADS+X'133' END ADRS
014C  9E24                     101          OCR   R2,R4                 SELCH STOP
014E  2490                     102          LIS   R9,0
0150  9A39                     103          WDR   R3,R9                 MS BYTE OF 3 BYTE START ADRS
0152  9825                     104          WHR   R2,R5                 LS 2 BYTES OF START ADRS
```

MAG TAPE BOOT LOADER

```
0154  9A39            105           WDR   R3,R9              MS BYTE OF 3 BYTE END ADRS
0156  9826            106           WHR   R2,R6              LS 2 BYTES OF END ADDRESS
0158  DE10 0079       107           OC    R1,X'79'           START TAPE
015C  9E30            108           OCR   R3,R0              SELCH GO
015E  2145            109           BOS   MTLOAD5            SKIP IF FALSE SYNC
0160  9D29            110           SSR   R2,R9              ELSE WAIT FOR SELCH
0162  2081            111           BTBS  8,1                TO GO NON-BUSY
0164  9E24            112           OCR   R2,R4              THEN STOP THE SELCH
0166  230A            113           BS    MTLOAD6
0168  9D19            114 MTLOAD5   SSR   R1,R9              MAG TAPE STATUS
016A  2081            115           BTBS  8,1                WAIT FOR NON-BUST
016C  DB15 0000       116           RD    R1,0(R5)           READ
0170  2651            117           AIS   R5,1               BUMP INDEX
0172  0565            118           CLAR  R6,R5              DONE?
0174  2286            119           BNLS  MTLOAD5            NO, LOOP
0176  9D19            120           SSR   R1,R9              FINAL STATUS
0178  2170            121           BTFS  7,0                HANG ON ERROR
017A  9D19            122 MTLOAD6   SSR   R1,R9              TRANSFER COMPLETE
017C  C390 0010       123           THI   R9,X'10'           WAIT FOR NO MOTION
0180  2233            124           BZS   MTLOAD6            WAIT FOR NO MOTION
0182  D390 0282       125           LB    R9,BOOTEN-MTLOADS+X'100'+18
0186  D350 0283       126           LB    R5,BOOTEN-MTLOADS+X'100'+19
018A  9158            127           SLHLS R5,8               POSITION BITS 16-23
018C  D360 0284       128           LB    R6,BOOTEN-MTLOADS+X'100'+20
0190  0656            129           OAR   R5,R6              (R9,R5)=PROGRAM START ADDRESS
0192  0833            130           LDAR  R3,R3              TEST HOST
0194  2334            131           BZS   MTLOAD6A           SKIP IF 16 BIT
                      132 *         EXHR  R7,R9              (R7)='00XX0000'
0196  3479            133           DCX   3479
0198  0675            134           OAR   R7,R5              (R7)='00XXYYZZ'
019A  2302            135           BS    MTLOAD6B
019C  0875            136 MTLOAD6A  LDAR  R7,R5              SAVE START ADDRESS
019E  0857            137 MTLOAD6B  LDAR  R5,R7              (R5)=START ADDRESS
01A0  D3A0 0285       138           LB    R10,BOOTEN-MTLOADS+X'100'+21
01A4  D360 0286       139           LB    R6,BOOTEN-MTLOADS+X'100'+22
01A8  9466            140           EXBR  R6,R6
01AA  D380 0287       141           LB    R8,BOOTEN-MTLOADS+X'100'+23
01AE  0668            142           OAR   R6,R8              (R10,R6)=LOAD END ADDRESS
01B0  0833            143           LDAR  R3,R3              TEST HOST
01B2  2333            144           BZS   MTLOAD7            SKIP IF 16 BIT
                      145 *         EXHR  R11,R10            (R11)='00XX0000'
01B4  34BA            146           DCX   34BA
01B6  066B            147           OAR   R6,R11             (R6)='00XXYYZZ'
01B8  0885            148 MTLOAD7   LDAR  R8,R5              START ADRS
                      149 *                                  YYZZ OR 00XXYYZZ
01BA  EC80 0010       150           SRL   R8,16              IF 16 BIT HOST, (R8,R9)=0000,YYZZ
                      151 *                                  IF 32 BIT HOST, (R8)=000000XX
01BE  C885 00FF       152           LHI   R11,255(R5)        START ADDRESS + 255 EQUALS
                      153 *                                  END ADDRESS FOR THIS RECORD
01C2  056B            154           CLAR  R6,R11             COMPARE TO END ADDRESS
01C4  2382            155           BNLS  MTLOAD7A           SKIP IF NOT LESS
01C6  08B6            156           LDAR  R11,R6             IF YES, USE REAL END ADDRESS
01C8  9E24            157 MTLOAD7A  OCR   R2,R4              SELCH STOP
```

MAG TAPE ROOT LOADER

```
01CA 9A39           158          WDR   R3,R9              MS BYTE OF ADDRESS
01CC 9825           159          WHR   R2,R5              OUTPUT START ADRS
01CE 9A3A           160          WDR   R3,R10
01D0 982B           161          WHR   R2,R11             OUTPUT END ADRS
01D2 DE10 0079      162          OC    R1,X'79'           MAG TAPE START
01D6 9E20           163          OCR   R2,R0              SELCH START
01D8 2145           164          BOS   MTLOAD8            SKIP IF FALSE SYNC
01DA 9D29           165          SSR   R2,R9
01DC 2081           166          BTBS  8,1                WAIT ON SELCH BUSY
01DE 9E24           167          OCR   R2,R4              SELCH STOP
01E0 2308           168          BS    MTLOAD9
01E2 9D19           169 MTLOAD8  SSR   R1,R9              MAG TAPE STATUS
01E4 2081           170          BTBS  8,1                LOOP ON BUSY
01E6 DB15 0000      171          RD    R1,0(R5)           READ BYTES
01EA 2651           172          AIS   R5,1               BUMP ADDRESS
01EC 05B5           173          CLAR  R11,R5             DONE CHECK
01EE 2286           174          BNLS  MTLOAD8            LOOP ON RECORD
01F0 0390 0079      175 MTLOAD9  LB    R9,X'79'           GET READ COMMAND
01F4 C590 00A1      176          CLHI  R9,X'A1'           800/1600 BPI?
01F8 2334           177          BES   MTLOAD9B           SKIP IF YES
01FA 9D19           178 MTLOAD9A SSR   R1,R9              6250 STATUS
01FC 212D           179          BTCS  2,MTLOAD10         DONE IF EOF
01FE 2305           180          BS    MTLOAD9C           ELSE KEEP READING
0200 9D19           181 MTLOAD9B SSR   R1,R9
0202 C390 0040      182          THI   R9,X'40'           EOF?
0206 2138           183          BNZS  MTLOAD10           DONE IF YES
0208 9095           184 MTLOAD9C SRHLS R9,5               NO MOTION CHECK
020A 228D           185          BNCS  MTLOAD9            WAIT FOR IT
020C C85B 0001      186          LHI   R5,1(R11)          NEXT START ADRS
0210 0556           187          CLAR  R5,R6              COMPARE TO END ADDRESS/
0212 4280 01B8      188          BL    MTLOAD7-MTLOADS+X'100' LOOP
0216 0857           189 MTLOAD10 LDAR  R5,R7              START ADDRESS
                    190 *                                 (R6)=END ADDRESS
0218 24A0           191          LIS   R10,0              CHECKSUM ACCUMULATOR
021A D3B7 0000      192 MTLOAD11 LB    R11,0(R7)
021E 07AB           193          XAR   R10,R11            CALCULATE CHECKSUM
0220 2671           194          AIS   R7,1
0222 0567           195          CLAR  R6,R7
0224 2285           196          BNLS  MTLOAD11
0226 9D19           197 MTLOAD12 SSR   R1,R9
0228 C390 0010      198          THI   R9,X'10'           NO MOTION CHECK
022C 2233           199          BZS   MTLOAD12
022E C870 0038      200          LHI   R7,X'38'
0232 0390 0079      201          LB    R9,X'79'           GET READ COMMAND
0236 C590 00A1      202          CLHI  R9,X'A1'           800/1600 BPI?
023A 2333           203          BES   MTLOAD13           SKIP IF YES
023C C870 00E0      204          LHI   R7,X'E0'           RW COMMAND FOR 6250
0240 9E17           205 MTLOAD13 OCR   R1,R7              REWIND THE TAPE
0242 9D19           206 MTLOAD14 SSR   R1,R9
0244 C390 0010      207          THI   R9,X'10'           NO MOTION CHECK
0248 2233           208          BZS   MTLOAD14
024A D370 0288      209          LB    R7,BOOTFN-MTLOADS+X'100'+24
024E 057A           210          CLAR  R7,R10             CHECK CHECKSUM
```

MAG TAPE BOOT LOADER

```
0250   0335                    BER   R5          GO IF OK
0252   C810 00EE               LHI   R1,X'EE'
0256   24A1                    LIS   R10,1
0258   C8B0 0040               LHI   R11,X'40'
025C   9EAB                    OCR   R10,R11     DISPLAY IN INCREMENTAL MODE
025E   9AA1                    WDR   R10,R1
0260   24E0                    LIS   R14,0
0262   9AAE                    WDR   R10,R14
0264   9AAE                    WDR   R10,R14
0266   9AAE                    WDR   R10,R14
0268   C8B0 0080               LHI   R11,X'80'
026C   9EAB                    OCR   R10,R11     NORMAL MODE
026E   2200                    BS    *           HANG ON CHECKSUM ERROR
       0000 026F      MTLOADE  EQU   *-1
       0000 0270      BOOTEN   EQU   *
0270                           END
```

MAG TAPE BOOT LOADER

ASSEMBLED BY CAL 03-066R08-00 (32-BIT)

START OPTIONS: T=16,ERLST

NO CAL ERRORS
NO CAL WARNINGS
2 PASSES

TABLE SPACE USED : 2K

```
ABSTOP    0000 0270
ADC       0000 0002
BOOTEN    0000 0270
IMPTOP    0000 0000R
LADC      0000 0001
MTBOOT    0000 0080
MTBOOT1   0000 0090
MTBOOT2   0000 00C0
MTLOAD10  0000 0216
MTLOAD11  0000 021A
MTLOAD12  0000 0226
MTLOAD13  0000 0240
MTLOAD14  0000 0242
MTLOAD2   0000 012E
MTLOAD3   0000 013C
MTLOAD4   0000 0144
MTLOAD5   0000 0168
MTLOAD6   0000 017A
MTLOAD6A  0000 019C
MTLOAD6B  0000 019E
MTLOAD7   0000 01B8
MTLOAD7A  0000 01C8
MTLOAD8   0000 01E2
MTLOAD9   0000 01F0
MTLOAD9A  0000 01FA
MTLOAD9B  0000 0200
MTLOAD9C  0000 0208
MTLOADE   0000 026F
MTLOADS   0000 0100
MTLOADS0  0000 010C
MTLOADS1  0000 012A
PURETOP   0000 0000R
R0        0000 0000
R1        0000 0001
R10       0000 000A
R11       0000 000B
R14       0000 000E
R15       0000 000F
R2        0000 0002
R3        0000 0003
R4        0000 0004
R5        0000 0005
R6        0000 0006
R7        0000 0007
R8        0000 0008
R9        0000 0009
```

APPENDIX F
FMD BOOT LOADER

10:10:33 05/29/80

FMD BOOT LOADER

```
0000R
       0000 0080                    ORG   X'80'
0080   2440          STARTAD  EQU   *
0082   2303          BOOTST   LIS   R4,0
0084   4000                   BS    BOOT1
0086   4010                   DCX   4000
0088   4040 0022              DCX   4010
                     BOOT1    STH   R4,X'22'                 REGISTER SAVE POINTER(16BIT)
008C   C840 001C              LHI   R4,28                    LOAD DIRECTORY START LRN
0090   D310 0078              LB    R1,X'78'                 LOAD DEVICE ADDRESS
0094   D320 0079              LB    R2,X'79'
0098   C420 0030              NHI   R2,X'30'                 REMOVE DRIVE SELECT BITS
009C   C620 00C7              OHI   R2,X'C7'                 FORM STOP COMMAND
00A0   C850 00D0              LHI   R5,X'D0'                 START ADDRESS
00A4   C860 01F7              LHI   R6,ENDAD-STARTAD+X'D0'
                     * LOAD REST OF BOOT LOADER
00A8   9D13          BOOT1B   SSR   R1,R3
00AA   2081                   BTBS  8,1                      WAIT ON BUSY
00AC   D915 0000              RH    R1,0(R5)                 READ HALFWORDS
00B0   2652                   AIS   R5,2                     BUMP INDEX
00B2   0565                   CLAR  R6,R5
00B4   2286                   BNLS  BOOT1B                   LOOP
00B6   9D13                   SSR   R1,R3                    CHECK FINAL STATUS
00B8   2152                   BTFS  5,REDOBL                 RETRY BOOT LOAD
00BA   230E                   BS    STOPA
00BC   C850 D500     REDOBL   LHI   R5,X'D500'
00C0   4050 0050              STH   R5,X'50'
00C4   C850 00CF              LHI   R5,X'CF'
00C8   4050 0052              STH   R5,X'52'
00CC   9E12                   OCR   R1,R2                    STOP
00CE   9D13                   SSR   R1,R3
00D0   2221                   BFBS  2,1                      WAIT FOR IDLE
00D2   4300                   DCX   4300,0050                B   X'50'
00D4   0050
00D6   9E12          STOPA    OCR   R1,R2                    STOP COMMAND
00D8   9D13          IDLE     SSR   R1,R3
00DA   2221                   BFBS  2,1                      LOOP ON NOT IDLE
00DC   C850 01A8              LDAI  R5,LDBUF                 START ADDRESS
00E0   C860 01C8              LDAI  R6,LDBUF+32
00E4   9814                   WHR   R1,R4                    WRITE LRN TO CONTROLLER
00E6   2726                   SIS   R2,6                     FORM READ COMMAND
00E8   9E12                   OCR   R1,R2                    ISSUE READ COMMAND
                     * READ DIRECTORY
00EA   9D13          BOOT1C   SSR   R1,R3
00EC   2081                   BTBS  8,1
00EE   D915 0000              RH    R1,0(R5)
00F2   2652                   AIS   R5,2
00F4   0565                   CLAR  R6,R5
00F6   2286                   BNLS  BOOT1C                   LOOP
00F8   9D13                   SSR   R1,R3
00FA   2152                   BTFS  5,REDO                   ERROR, RETRY
00FC   2305                   BS    STOP1
00FE   2626          REDO     AIS   R2,6
0100   9E12                   OCR   R1,R2                    ISSUE STOP COMMAND
```

FMD BOOT LOADER

```
0102  4300 00D8          72            B     IDLE-STARTAD+X'80'  GO TO IDLE
0106  2626                73  STOP1     AIS   R2,6
0108  9E12                74            OCR   R1,R2               ISUE STOP COMMAND
010A  9D13                75            SSR   R1,R3
010C  2221                76            BFBS  2,1                 WAIT FOR IDLE
010E  C840 4000           77            LHI   R4,X'4000'          TEST PATTERN
0112  0A44                78            AAR   R4,R4
0114  2115                79            BMS   IS16                BRANCH, 16 BIT HOST
0116  2470                80            LIS   R7,0                SET 32 BIT FLAG
0118  4840 01AE           81            LH    R4,LDBUF+6          32 BIT PDB POINTER
011C  2304                82            BS    COM
011E  4840 01AA           83  IS16      LH    R4,LDBUF+2          16 BIT PDB POINTER
0122  2472                84            LIS   R7,2                16 BIT FLAG
0124  C850 01A8           85  COM       LDAI  R5,LDBUF            START ADDRESS
0128  C860 0227           86            LDAI  R6,LDBUF+127        END ADDRESS
012C  9814                87  COM1      WHR   R1,R4               PDB LRN TO CONTROLLER
012E  2726                88            SIS   R2,6
0130  9E12                89            OCR   R1,R2               ISSUE READ COMMAND
                          90  * READ PDB
0132  9D13                91  BOOT1D    SSR   R1,R3
0134  2081                92            BTBS  8,1                 WAIT ON BUSY
0136  D915 0000           93            RH    R1,0(R5)
013A  2652                94            AIS   R5,2
013C  0565                95            CLAR  R6,R5
013E  2286                96            BNLS  BOOT1D              LOOP
0140  9D13                97            SSR   R1,R3
0142  2152                98            BTFS  5,RED01             ERROR RETRY
0144  2307                99            BS    RDLDRGEN            ELSE READ PROGRAM
0146  2626               100  RED01     AIS   R2,6
0148  9E12               101            OCR   R1,R2               ISSUE STOP COMMAND
014A  9D13               102            SSR   R1,R3
014C  2221               103            BFBS  2,1                 WAIT FOR IDLE
014E  4300 0124          104            B     COM-STARTAD+X'80'   GO TO COM
0152  2626               105  RDLDRGEN  AIS   R2,6
0154  9E12               106            OCR   R1,R2               ISSUE STOP COMMAND
0156  9D13               107            SSR   R1,R3
0158  2221               108            BFBS  2,1                 WAIT FOR IDLE
015A  2726               109            SIS   R2,6
015C  2641               110            AIS   R4,1
015E  4950 01B0          111  RED03     LH    R5,LDBUF+8          LOAD LOW
0162  4050 01A0          112            STH   R5,BOOTEN16-STARTAD+X'80'
0166  4050 01A6          113            STH   R5,BOOTEN32-STARTAD+X'80'
016A  4860 01B4          114            LH    R6,LDBUF+12         LOAD HIGH
016E  9814               115            WHR   R1,R4               WRITE LRN TO CONTROLLER
0170  9E12               116            OCR   R1,R2               ISSUE READ COMMAND
                         117  * READ LOADER-GENERATOR PROGRAM
0172  9D13               118  BOOT1E    SSR   R1,R3
0174  2081               119            BTBS  8,1                 LOOP ON BUSY
0176  D915 0000          120            RH    R1,0(R5)
017A  2652               121            AIS   R5,2
017C  0565               122            CLAR  R6,R5
017E  2286               123            BNLS  BOOT1E              LOOP
0180  9D13               124            SSR   R1,R3
```

```
0182  2152                       BTFS  5,RED02                  ERROR RETRY
0184  2307                       BS    TURNOVER
0186  2626            RED02      AIS   R2,6
0188  9E12                       OCR   R1,R2                    ISSUE STOP COMMAND
018A  9D13                       SSR   R1,R3
018C  2221                       BFBS  2,1                      WAIT FOR IDLE
018E  4300 015E                  B     RED03-STARTAD+X'80'      GO TO RED03
0192  2626            TURNOVER   AIS   R2,6
0194  9E12                       OCR   R1,R2                    ISSUE STOP COMMAND
0196  9D13                       SSR   R1,R3
0198  2221                       BFBS  2,1                      WAIT FOR IDLE
019A  0877                       LDAR  R7,R7                    TEST HOST FLAG
019C  2333                       BZS   G032                     BRANCH, 32 BIT
019E  4300                       DCX   4300                     ELSE TAKE 16 BIT BRANCH
01A0  6000            BOOTEN16   DCX   6000
01A2  4300            G032       DCX   4300
01A4  4000                       DCX   4000                     RX3 BRANCH
01A6  6000            BOOTEN32   DCX   6000                     32 BIT START
      0000 01A7       ENDAD      EQU   *-1
                      *
      0000 01A8       LDBUF      EQU   *
```

PROG= 0625202 ASSEMBLED BY CAL 03-066R08-00 (32-BIT)

```
                    SCRAT
0625202             PROG  COMMON MMD CROSS GENERATOR  06-252R04M96A13
                    CROSS
                    SQUEZ 4
                    ERSQZ
*                   SQCHK
*
* COPYRIGHT C BY PERKIN-ELMER CORPORATION, JANUARY 1980
* REVISED JUNE, 1980


* THE COMMON MMD CROSS GENERATOR IS DESIGNED FOR MEDIA
* DUPLICATION, VERIFICATION, AND UPTADING.
* POSSIBLE INPUT OR OUTPUT DEVICES ARE 800, 1600, OR 6250 BPI
* MAGNETIC TAPE; 10, 13.5, 80, OR 300 MB
* DISK; OR FLOPPY DISKETTE.
*
* WHEN THE INPUT DEVICE IS A MAG TAPE OR A DISK AND
* THE OUTPUT DEVICE IS A FLOPPY, THE PROGRAM SELECTS
* THE INPUT LIBRARY ENTRIES TO BE TRANSFERRED TO THE
* FLOPPY.  THIS SELECTION IS BASED ON THE FLOPPY VOLUME
* NUMBER OR FUNCTIONAL VARIATION WHEN DOING A FULL DUPLICATION.
* WHEN DOING SELECTIVE OUTPUT, USER INPUT CHOOSES THE ENTRIES.
* THE VOLUME NUMBER CAN BE SPECIFIED WITH THE "VOLUME"
* OPTION, OR IN CONJUNCTION WITH THE "BUILD" OR "VERIFY"
* COMMANDS.
* IF THE INPUT DEVICE IS A FLOPPY, THEN THE OUTPUT DEVICE
* IS FORCED TO ALSO BE A FLOPPY.  VOLUME OPTIONS ARE IGNORED.
* THE PROGRAM AUTOMATICALLY DETERMINES IF THE INPUT AND
* OUTPUT DEVICE IS A MAG TAPE, 10, 80, OR 300 MB DISK,
* OR A FLOPPY.
* WHEN THE OUTPUT DEVICE IS A DISK, IT IS CREATED IN SUCH A WAY
* THAT IT CAN BE BOOT-LOADED USING THE SAME LSU OR ALO USED
* FOR BOOT-LOADING AN O.S. DISK.  THE DISK HAS AN O.S. FORMAT
* VOLUME DESCRIPTOR, BIT MAP, AND DIRECTORY.  THE FIRST TWO
* ENTRIES IN THE DIRECTORY ARE "OS16MDL2.111" & "OS32MDL2.111".
* THESE ARE ACTUALLY THE 16 BIT MMD LOADER AND THE 32 BIT MMD
* LOADER (LIBRARY SEQUENCES 001 AND 002), WRITTEN ON THE DISK
* TO RESEMBLE A TET'ED O.S.  THE THIRD DIRECTORY ENTRY IS
* CALLED "MMD.DAT".  IT STARTS IMMEDIATELY AFTER "OS32MDL2.111" AND
* EXTENDS TO 12,000 SECTORS BEYOND CYLINDER 8.  THIS FILE IS
* THE REGULAR MMD FORMAT LIBRARY.
```

EXEC - ETPE R05P1

```
                         43          NLSTC
                         44  *
                         45  $STRUC1  STRUC                  OPTION TABLE STRUCTURE
0000                     46  $OPTNAME DS    6                ASCII OPTION NAME
0006                     47  $CKROUT  DS    2                Z(CHECK ROUTINE)
0008                     48  $VALU1   DS    2                16-BIT VALUE
000A                     49  $VALU2   DS    2                SPARE
000C                     50          ENDS
                         51  *
      0000 0050          52  $BUFLEN  EQU   80               I/O BUFFER LENGTH
      0000 0038          53  MAXLINE  EQU   56               LIST DEVICE MAXIMUM LINE COUNT    *
      0000 0014          54  OPTLINES EQU   20               MAX LINES FOR OPTION PRINTOUT     *
                         55  *
                         56  *
                         57  *       CONDITIONAL ASSEMBLY PARAMETERS TO FOLLOW
                         58  *
                         59  *             IN ALL CASES, 0 EQUALS DELETE
                         60  *                           1 EQUALS INCLUDE
                         61  *
                         62  *                            FOR $CLOCK, FOLLOWING TIMERS INCLUDED
                         63  *                           1 EQUALS INCLUDE SOFTWARE
                         64  *                           2 EQUALS INCLUDE HARDWARE
                         65  *                           3 EQUALS INCLUDE BOTH
                         66  *             TIMER LABEL IS "TIMER" FOR SOFTWARE AND
                         67  *             HARDWARE, EXCEPT WHEN BOTH ARE INCLUDED.
                         68  *             THEN LABELS ARE "STIMER" AND "HTIMER"
                         69  *             RESPECTIVELY.
                         70  *
      0000 0000          71  $R5BIN   EQU   0
      0000 0000          72  $DECTAB  EQU   0
      0000 0000          73  $DECHEX  EQU   0
      0000 0000          74  $DECASC  EQU   0
      0000 0000          75  $KBINT   EQU   0
      0000 0001          76  $CLOCK   EQU   1
      0000 0001          77  $DISPLAY EQU   1
      0000 0000          78  $BUFIO   EQU   0
                         79  *
      0000 0000          80  R0       EQU   0
      0000 0001          81  R1       EQU   1
      0000 0002          82  R2       EQU   2
      0000 0003          83  R3       EQU   3
      0000 0004          84  R4       EQU   4
      0000 0005          85  R5       EQU   5
      0000 0006          86  R6       EQU   6
      0000 0007          87  R7       EQU   7
      0000 0008          88  R8       EQU   8
      0000 0009          89  R9       EQU   9
      0000 000A          90  R10      EQU   10
      0000 000B          91  R11      EQU   11
      0000 000C          92  R12      EQU   12
      0000 000D          93  R13      EQU   13
      0000 000E          94  R14      EQU   14
      0000 000F          95  R15      EQU   15
```

```
0000R                          97            ORG   X'A00'
0A00   4300 0A5E               98  ORIGIN1   B     START          START HERE FOR 32-BIT PROCESSOR
0A04                           99            IFZ   ADC-2
0A04   4300 0A5E              100  ORIGIN2   B     START          START HERE FOR 16-BIT PROCESSOR
0A08   4300 0A72              101  ORIGIN3   B     START3         SPECIAL 32-BIT PROCESSOR START
0A0C   4300 0A72              102  ORIGIN4   B     START4         SPECIAL 16-BIT PROCESSOR START
                              103            ELSE
                              107            ENDC
                              108  *
                              109  *--------------------------
                              110  *   TEST CONSTANTS                          *
                              111  *
       0000 0006              112  $MAXIO    EQU   6              > MAX VALID IDENTIFIER
0A10   0101                   113  IO        DC    X'0101'        I/O DEVICE(S) IDENTIFIER
                              114  *
0A12   0010                   115  PASLADR   DC    X'0010'        PALSA/PALM READ ADDRESS
0A14   0011                   116            DC    X'0011'        PASLA/PALM WRITE ADDRESS
0A16   0002                   117  CLIFADR   DC    X'0002'        CURRENT LOOP INTERFACE READ ADDRES
0A18   0002                   118            DC    X'0002'        CURRENT LOOP INTERFACE WRITE ADDRE
0A1A   0062                   119  LPADR     DC    X'0062'        DUMMY FOR LINE PRINTER
0A1C   0062                   120            DC    X'0062'        WRITE ADDRESS
0A1E   0010                   121  C300ADR   DC    X'0010'        CAROUSEL/PASLA READ ADDRESS
0A20   0011                   122            DC    X'0011'        CAROUSEL/PASLA WRITE ADDRESS
0A22   00C0                   123  MICROBUS  DC    X'00C0'        MICROBUS READ ADDRESS
0A24   00C0                   124            DC    X'00C0'        MICROBUS WRITE ADDRESS
0A26   0000                   125            DCX   0              PROVISION FOR SPECIAL DEVICE (READ
0A28   0000                   126            DCX   0              WRITE ADDRESS
                              127  *
                              128  * IO =   0101  FOR CRT ON PASLA
                              129  *        0202  FOR TELETYPE, CAROUSEL 15/30
                              130  *        XX03  FOR LINE PRINTER
                              131  *        0404  FOR CAROUSEL 300
                              132  *        0505  FOR MICROBUS
                              133  *
                              134  *-------------------------------------------------------
                              135  *  ETPE IO COMMANDS
                              136  *
0A2A   0000                   137  CONRADR   DCX   0              CONSOLE DEVICE READ ADDRESS
0A2C   0000                   138  CONWADR   DCX   0              CONSOLE DEVICE WRITE ADDRESS
                              139  *
0A2E   0000                   140  CONRD     DCX   0              CONSOLE READ/WRITE COMMANDS
       0000 0A2F              141  CONWRT    EQU   CONRD+1
0A30   0000                   142  CON2ND    DCX   0
       0000 0A31              143  CONENRD   EQU   CON2ND+1
0A32   0000                   144  CONCMD    DCX   0              DUMMY HW AS POINTER
0A34   A1A3                   145  CRTRD     DCX   A1A3           FOR CRT
0A36   EE61                   146  CRT2ND    DCX   EE61
0A38   E4E8                   147  CLIFRD    DCX   E4E8           *   CURRENT LOOP INTERFACE   P1 10/79
0A3A   0044                   148  CLIF2ND   DCX   0044           *                              *
0A3C   0080                   149  LPWRT     DCX   0080           *   LINE PRINTER
0A3E   0000                   150            DCX   0              DUMMY FOR LP
0A40   A1A3                   151  CARRD     DCX   A1A3           *   CAROUSEL 300
0A42   F061                   152  CAR2ND    DCX   F061
```

EXEC - ETPE R05P1

```
0A44  8202                153  MREADC   DCX   8202              *   MICROBUS
0A46  0000                154           DCX   0                 DUMMY HW FOR MICROBUS
                          155  *
                          156  *-------------------------------------------------
0A48  00                  157  CONRQ2S  DB    0                 CONSOLE REQUEST TO SEND CMD
0A49  23                  158  CRTRQ2S  DB    X'23'             FOR CRT
0A4A  00                  159           DB    0                 DUMMY BYTE FOR CLI
0A4B  00                  160           DB    0                 *   DUMMY BYTE FOR LP
0A4C  23                  161  CARRQ2S  DB    X'23'             *   CAROUSEL 300
0A4D  00                  162           DB    0                 *   DUMMY BYTE FOR MICROBUS
0A4E                      163           DB    *                 (ALIGN ON HW BOUNDRY)
0A4E  0000                164           DCX   0                 RESERVED
0A50  30F0                165  PSW      DCX   30F0              PSW USED IN PROGRAM
0A52  30F0                166  PSW2     DCX   30F0              PSW USED IN EXEC
0A54  70F0                167  PSW3     DCX   70F0              PSW USED IN INTERRUPT TESTS
0A56  0000                168           DCX   0                 RESERVED
0A58  0000                169           DCX   0                 RESERVED
0A5A  7FFF                170  $TIMVAL  DCX   7FFF              TIMEOUT CONSTANT
0A5C  8800                171  $CON     DCX   8800              BREAKPOINT INSTRUCTION
                          172  *-----------------------------
                          173  *
0A5E  48E0 0A52           174  START    LH    R14,PSW2          NEW PSW FOR ILLEGAL INTERRUPT
0A62  C8F0 0A72           175           LOAI  R15,STARTA        AND NEW LOC
0A66  D0E0 0034           176           STM   R14,X'34'         FOR SERIES 16
0A6A  D0E0 0030           177           STM   R14,X'30'         FOR SERIES 32
0A6E  0000                178           DCX   0                 TAKE AN ILLEGAL INSTRUCTION INT
0A70  2200                179           BS    *                 HALT IF II NOT TAKEN
                          180  *
*  0A72                   181  START3   B     STARTA            INSERT SPECIAL ROUTINE HERE
   0A72                   182           IFZ   ADC-2
*  0A72                   183  START4   B     STARTA            INSERT SPECIAL ROUTINE HERE
                          184           ENDC
0A72  C800 8000           185  STARTA   LHI   R0,X'8000'
0A76  4000 155C           186           STH   R0,ISITERR        FORCE TITLE PRINT
0A7A  EC00 0010           187           SRL   R0,16             REGISTER PAIR SHIFTED, SERIES 16
0A7E  4000 1548           188           STH   R0,MOD32          SIGN EXTENSION, SERIES 32.
                          189  *
0A82  41E0 1152           190           BAL   R14,STCON         SET UP CONSOLE
0A86  41F0 1304           191           BAL   R15,LCORE         SET UP LOW CORE
0A8A  2400                192  STARTA1  LIS   R0,0
0A8C  4000 1A16           193           STH   R0,NOSTOP         RESET AUTO VERIFY FLAG          **
0A90  4000 1688           194           STH   R0,NOMSG+$VALU1   FORCE 'NOMSG 0' AT START
0A94  4000 1558           195           STH   R0,$BRKFLG        NO BREAK KEY YET
0A98  41F0 0FB0           196           BAL   R15,CRLF
0A9C  41F0 0FBA           197           BAL   R15,$PRINT        PRINT TEST PROGRAM TITLE
0AA0  1780                198           DAC   TITLE
0AA2  48F0 1566           199           LH    R15,$WASDU        WAS DEVICE SEEN DU ?
0AA6  4230 0CE2           200           BNZ   HALT9             PRINT TOTAL, TOTERR
                          201  *
                          202  *
                          203  *  KEYBOARD INPUT ROUTINE
                          204  *
      0000 0AAA           205  OPTIN    EQU   *
```

```
  0AAA  41F0 12BC                   BAL   R15,SETKB          ESTABLISH CONSOLE
  0AAE  41F0 0FB0                   BAL   R15,CRLF
  0AB2  4820 0A52        OPTIN1     LH    R2,PSW2            SPEC'D AS X'30F0'
  0AB6  4020 155C                   STH   R2,ISITERR         FORCE EXEC MESSAGE PRINT
  0ABA  9512                        EPSR  R1,R2              NO INT. REG SET 15
  0ABC  2400                        LIS   R0,0               NO BRK TERM QUEUE, NOMSG>0   P1 10/79
  0ABE  4000 16DE                   STH   R0,$LINCNT         CLEAR LISTING LINE COUNT         *
  0AC2  4000 1558                   STH   R0,$BRKFLG         *                            P1 10/79
  0AC6  41F0 12BC                   BAL   R15,SETKB          ESTABLISH CONSOLE
  0ACA  D340 1636                   LB    R4,AMSG            OUTPUT AN * TO INDICATE
  0ACE  41F0 1054                   BAL   R15,OUTCHR         COMMAND MODE ESTABLISHED
  0AD2  2541                        LCS   R4,1               X'FF'
  0AD4  41F0 1054                   BAL   R15,OUTCHR
  0AD8  2400                        LIS   R0,0
  0ADA  4840 1A16                   LH    R4,NOSTOP          TEST AUTO-VERIFY FLAG
  0ADE  4000 1A16                   STH   R0,NOSTOP          RESET FLAG
  0AE2  4230 17C8                   BNZ   @VERIFY            VERIFY MODE
  0AE6  41F0 10CC                   BAL   R15,$READ          GET INPUT RECORD
                         *
                         * -------------------------------------
                         *
                         * COMMAND DECODE
                         *
  0AEA  C8C0 11D6        $LOOK      LDAI  R12,QUESTN         GLOBAL ERROR ROUTINE
  0AEE  C810 1650                   LDAI  R1,OPT-$STRUC1     TO START AT OPTION TABLE
* 0AF2  261C             $LOOK.0    AHI   R1,$STRUC1         ADVANCE TO NEXT TABLE ENTRY
  0AF4  2430             $LOOK.1    LIS   R3,0               CLEAR BUFFER INDEX
  0AF6  4851 0000                   LH    R5,0(R1)           END OF TABLE ?
  0AFA  021C                        BMR   R12                IF MINUS, THEN NO MATCH => ERROR.
  0AFC  0861                        LDAR  R6,R1              START OF OPTION ENTRY
  0AFE  D343 360C        $LOOK.2    LB    R4,$INBUF(R3)      GET INPUT BYTE
  0B02  D356 0000                   LB    R5,0(R6)           GET OPTION NAME BYTE
  0B06  2631                        AIS   R3,1               ADVANCE TO NEXT BYTE
  0B08  C550 0020                   CLHI  R5,C' '            OPTION NAME SPACE IN TABLE ?
  0B0C  233A                        BES   $LOOK.3            BRANCH: YES.
  0B0E  0545                        CLAR  R4,R5              INPUT, OPTION BYTES MATCH ?
  0B10  203F                        BNES  $LOOK.0            BRANCH: NO.
  0B12  2661                        AIS   R6,1               INDEX OPTION POINTER
  0B14  C530 0006                   CLHI  R3,$CKROUT         WHOLE OPTION NAME MATCHED ?
  0B18  208D                        BLS   $LOOK.2            BRANCH: NOT YET.
  0B1A  D343 360C                   LB    R4,$INBUF(R3)      GET BYTE FOLLOWING OPTION
  0B1E  2631                        AIS   R3,1               INCREMENT BUFFER POINTER
  0B20  C540 0020        $LOOK.3    CLHI  R4,C' '            OPTION FOLLOWED BY SPACE ?
  0B24  2336                        BES   $LOOK.4            BRANCH: YES.
  0B26  C540 000D                   CLHI  R4,X'0D'           CARRIAGE RETURN ?
  0B2A  4230 0AF2                   BNE   $LOOK.0            BRANCH: NO MATCH
  0B2E  2731                        SIS   R3,1               POINT TO CARRIAGE RETURN
  0B30  C510 16D4        $LOOK.4    CLAI  R1,OPTION          'OPTION' CMD ?
  0B34  4330 0BEA                   BE    $OPTPRT            BRANCH: YES.
  0B38  C510 16EC                   CLAI  R1,BUILD           'BUILD' COMMAND?                  *
  0B3C  4330 17C0                   BE    @BUILD             BRANCH: YES.                      *
  0B40  C510 16F8                   CLAI  R1,VERIFY          VERIFY COMMAND?                   *
  0B44  4330 17C8                   BE    @VERIFY            *                                 *
```

EXEC - ETPE R05P1

```
0B48  C510 1704          259          CLAI  R1,BUILDV          BUILD & VERIFY COMMAND?          *
0B4C  4330 17B6          260          BE    @BUILDV           *                                *
0B50  C510 16E0          261          CLAI  R1,CON            'CON' CMD ?
0B54  4330 0A5C          262          BE    $CON              BRANCH: YES.
                         263  *
                         264  * TO PROCESS COMMANDS WHICH MUST HAVE HEXADECIMAL INPUT VALUE
                         265  *
0B58  C540 0020          266  $LOOK.5  CLHI  R4,C' '           OPTION FOLLOWED BY SPACE ?
0B5C  023C               267          BNER  R12               IF NO, ERROR.
0B5E  41E0 0EBA          268          BAL   R14,OPTVAL        GET OPTION VALUE IN R6
0B62  48E1 0006          269          LH    R14,$CKROUT(R1)   GET A(OPTION CHECK ROUTINE)
0B66  2334               270          BZS   $LOOK.6           BRANCH: NO SPECIAL ROUTINE.
0B68  01FE               271          BALR  R15,R14           LINK OPTION CHECK ROUTINE
0B6A  274D               272          SIS   R4,X'0D'          TERMINATED BY CR ?
0B6C  023C               273          BNZR  R12               IF NO, ERROR.
0B6E  4061 0008          274  $LOOK.6  STH   R6,$VALU1(R1)    STORE OPTION VALUE
0B72  4300 0AB2          275          B     OPTIN1            TO ACCEPT NEXT COMMAND
                         276  *
                         277  *
                         278  *
                         279  * ------------------------------------
                         280  * OPTION CHECK ROUTINES
                         281  *
0B76  C360 FFFE          282  ZERONE   THI   R6,X'FFFE'        IGNORE LSB
0B7A  033F               283          BZR   R15               OKAY
0B7C  030C               284          BR    R12               ERROR RETURN
                         285  *
0B7E  C360 FC00          286  ADR      THI   R6,X'FC00'        R6 = 10 BIT DEVICE ADDRESS
0B82  023C               287          BNZR  R12               ERROR RETURN
0B84  030F               288          BR    R15               OK
                         289  *
0B86  40F0 1A6E          290  ADR.XXX  STH   R15,R15SAVE       *                                *
0B8A  C360 FC00          291          THI   R6,X'FC00'        (R6) = 10 BIT DEVICE ADDRESS     *
0B8E  023C               292          BNZR  R12               *                                *
0B90  24E0               293          LIS   R14,0             *                                *
0B92  40E1 0008          294          STH   R14,$VALU1(R1)    CLEAR FIRST VALUE                *
0B96  40E1 000A          295          STH   R14,$VALU2(R1)    CLEAR SECOND VALUE               *
0B9A  40E1 0014          296          STH   R14,$VALU1+$STRUC1(R1) CLEAR THIRD VALUE           *
0B9E  C540 002C          297          CLHI  R4,C','           COMMA?                           *
0BA2  023F               298          BNER  R15               EXIT IF NO, ONLY ONE VALUE       *
0BA4  4061 0008          299          STH   R6,$VALU1(R1)     STORE THIS FIRST VALUE           *
0BA8  41E0 0EBA          300          BAL   R14,OPTVAL        LOOK FOR SECOND VALUE            *
0BAC  C360 FC00          301          THI   R6,X'FC00'        10 BITS?                         *
0BB0  023C               302          BNZR  R12               ERROR IF NO                      *
0BB2  C540 002C          303          CLHI  R4,C','           COMMA FOLLOWS THIS ONE?          *
0BB6  2138               304          BNES  ADR.001           IF NO, NO CONTROLLER ADDRESS     *
0BB8  4061 000A          305          STH   R6,$VALU2(R1)     STORE CONTROLLER ADDRESS         *
0BBC  41E0 0EBA          306          BAL   R14,OPTVAL        GO GET SELCH VALUE               *
0BC0  C360 FC00          307          THI   R6,X'FC00'        10 BITS?                         *
0BC4  023C               308          BNZR  R12               ERROR IF OVER                    *
* 0BC6  261C              309  ADR.001  AHI   R1,$STRUC1       POINT TO SELCH OPTION            *
0BC8  48F0 1A6E          310          LH    R15,R15SAVE       *                                *
0BCC  030F               311          BR    R15               GO STORE SELCH VALUE             *
```

```
                                312  *
0BCE   C360 FFF0                313  LEVEL    THI   R6,X'FFF0'          (R6) = INTERRUPT LEVEL HEX DIGIT
0BD2   033F                     314           BZR   R15                 RETURN TO LOOK5
0BD4   030C                     315           BR    R12
                                316  *
0BD6   48E0 1548                317  BIGVALUE LH    R14,MOD32           TEST HOST FLAG
0BDA   2333                     318           BZS   BIGVAL1             SKIP IF 16 BIT
0BDC   0876                     319           LDAR  R7,R6               R7 GETS THE LS 16 BITS
                                320  *        EXBR  R6,R6
0BDE   3466                     321           DCX   3466                R6 GETS THE MS 16 BITS
0BE0   4061 0008                322  BIGVAL1  STH   R6,$VALU1(R1)       STORE MS 16 BITS
0BE4   4071 000A                323           STH   R7,$VALU2(R1)       STORE LS 16 BITS
0BE8   030F                     324           BR    R15                 RETURN
                                352  *----------------------------
                                353  *  TO PROCESS INPUT COMMAND 'OPTION'
                                354  *
0BEA   C540 000D                355  $OPTPRT  CLHI  R4,X'0D'            OPTION (CR) ?
0BEE   233A                     356           BES   $OPT.0              BRANCH: YES.
0BF0   41E0 0EBA                357           BAL   R14,OPTVAL          NO. GET OPTION DEV. PRINTOUT NUM.
0BF4   C560 0006                358           CLHI  R6,$MAXIO           DEVICE NUMBER VALID ?
0BF8   038C                     359           BNLR  R12                 BRANCH: NO.
0BFA   0866                     360           LDAR  R6,R6               OPTION ZERO ?
0BFC   033C                     361           BZR   R12                 BRANCH: YES. INVALID INPUT.
0BFE   D260 155B                362           STB   R6,IOSAVE+1         CHANGE THE LIST DEVICE
0C02   4820 16DA                363  $OPT.0   LH    R2,OPTION+$CKROUT   SPECIAL PRINTOUT ROUTINE ?
0C06   2332                     364           BZS   OPTRTN              BRANCH: NO.
0C08   01F2                     365           BALR  R15,R2              LINK USER ROUTINE
0C0A   C830 165C                366  OPTRTN   LDAI  R3,OPT              START OF OPTION TABLE
0C0E   2440                     367  $OPT.A   LIS   R4,0                *                                   *
0C10   4040 16DE                368           STH   R4,$LINCNT          CLEAR LISTING LINE COUNT            *
0C14   2410                     369  $OPT.B   LIS   R1,0
0C16   0823                     370           LDAR  R2,R3               START OF OPTION ENTRY
0C18   D302 0000                371  $OPT.2   LB    R0,0(R2)            GET OPTION NAME BYTE
0C1C   D201 358C                372           STB   R0,$OUTBUF(R1)      MOVE TO OUTPUT BUFFER
0C20   2611                     373           AIS   R1,1
0C22   2621                     374           AIS   R2,1
0C24   C510 0006                375           CLHI  R1,$CKROUT          WHOLE NAME MOVED ?
0C28   2088                     376           BLS   $OPT.2              BRANCH: NO.
0C2A   C840 2020                377           LHI   R4,C'  '            SPACES
0C2E   4040 35C2                378           STH   R4,$OUTBUF+$CKROUT
                                379  *
                                380  * PROCESSING OPTIONS WITH 4-DIGIT HEX VALUES.
                                381  * OPTION NAME ALREADY IN OUTPUT BUFFER.
                                382  *
0C32   4813 0008                383  $OPT.3   LH    R1,$VALU1(R3)       OPTION VALUE HALFWORD
0C36   2404                     384           LIS   R0,4
0C38   C820 35C3                385           LDAI  R2,$OUTBUF+$CKROUT+1 BUFFER OFFSET
0C3C   41F0 0F18                386           BAL   R15,HEXASC          WRITE OPTION VALUE IN HEX (4 DIGITS
0C40   2400                     387           LIS   R0,0                MARK END OF MESSAGE
0C42   D200 35C7                388           STB   R0,$OUTBUF+$CKROUT+5 INSERT TO BUFFER
0C46   4813 000A                389           LH    R1,$VALU2(R3)       SECOND VALUE?                       *
*  0C4A   233D                  390           BZ    $OPT.4              SKIP IF NONE                        *
0C4C   2404                     391           LIS   R0,4                4 DIGITS                            *
```

EXEC - ETPE R05P1

```
 0C4E  C820 35C8          392          LDAI  R2,$OUTBUF+$CKROUT+6 WHERE IT GOES                *
 0C52  41F0 0F18          393          BAL   R15,HEXASC          CONVERT                       *
 0C56  C800 002C          394          LHI   R0,C','             COMMA                         *
 0C5A  D200 35C7          395          STB   R0,$OUTBUF+$CKROUT+5 SEPARATES TWO VALUES         *
 0C5E  2400               396          LIS   R0,0                MARK END OF MESSAGE           *
 0C60  D200 35CC          397          STB   R0,$OUTBUF+$CKROUT+10 *
 0C64  41F0 0FC8          398 $OPT.4   BAL   R15,@PRINT          OUTPUT PRINT BUFFER           *
 0C68  41F0 0FB0          399          BAL   R15,CRLF            DO CARRIAGE RETURN/LINE FEED  *
*0C6C  263C               400 $OPT.5   AHI   R3,$STRUC1          LENGTH OF TABLE ENTRY
 0C6E  C530 16B0          401          CLAI  R3,OPTEND2          DONE ALL PRINTING OPTIONS ?
 0C72  4380 0AB2          402          BNL   OPTIN1              BRANCH: YES.
 0C76  D300 155B          403          LB    R0,IOSAVE+1         CURRENT LIST ID
 0C7A  D400 0A10          404          CLB   R0,IO               SAME AS CONSOLE ?
 0C7E  4230 0C0E          405          BNE   $OPT.A              BRANCH: YES. NO LINE CNT TEST.
 0C82  4800 16DE          406          LH    R0,$LINCNT          *                             *
 0C86  C500 0014          407          CLHI  R0,OPTLINES         COMPARE TO PAGE LIMIT         *
 0C8A  4280 0C14          408          BL    $OPT.B              BRANCH: SCREEN NOT FULL       *
 0C8E  41F0 10CC          409          BAL   R15,$READ           GET (CR) OR (LF) TO CONTINUE
 0C92  D340 360C          410          LB    R4,$INBUF           FIRST CHARACTER
 0C96  274D               411          SIS   R4,X'0D'            CARRIAGE RETURN ?
 0C98  4330 0AB2          412          BZ    OPTIN1              BRANCH: YES. DONE.
 0C9C  2643               413          AIS   R4,X'03'            LINE FEED (X'0A') ?
 0C9E  4230 0AEA          414          BNZ   $LOOK               BRANCH: NO. ATTEMPT DECODE.
 0CA2  4300 0C0E          415 $OPT.6   B     $OPT.A              BRANCH: CONTINUE.
                          416 *-------------------------
                          417 * 'RUN' COMMAND HAS BEEN ENTERED
                          418 *
 0CA6  274D               419 $RUNIT   SIS   R4,X'0D'            CARRIAGE RETURN ENTERED ?
 0CA8  023C               420          BNZR  R12                 BRANCH: INPUT ERROR.
*0CAA                     421          B     $KEEP.3             *                          **
 0CAA  41F0 1304          422 $KEEP.3  BAL   R15,LCORE           SET UP LOW CORE            **
                          423 *
 0CAE  41F0 17EE          424          BAL   R15,INIT            LINK USER INITIALIZATION ROUTINE
 0CB2  41F0 12C6          425 INITRET  BAL   R15,SETLST          SELECT LIST DEVICE
 0CB6  2400               426          LIS   R0,0
 0CB8  4000 1568          427          STH   R0,TOTAL            RESET TOTAL
 0CBC  4000 156A          428          STH   R0,TOTERR           RESET TOTERR
 0CC0  41F0 11E4          429 $KEEP3   BAL   R15,TSTBRK          CHECK BREAK KEY
 0CC4  2400               430          LIS   R0,0
 0CC6  4000 155C          431          STH   R0,ISITERR          RESET ERROR FLAG
 0CCA  48E0 0A50          432          LH    R14,PSW             SPEC'D AS X'30F0'
 0CCE  48F0 156C          433          LH    R15,BTESTNO         BINARY TEST NUMBER
 0CD2  91F1               434          SLLS  R15,LADC            CONVERT TO OFFSET
 0CD4  24F0               435          LIS   R15,0               FORCE ZERO                 **
 0CD6  48FF 177C          436          LDA   R15,TESTS(R15)      POINTER TO TEST MODULE
 0CDA  D0E0 1538          437          STM   R14,NEWPSW
 0CDE  C200 1538          438          LPSW  NEWPSW              GO TO TEST, WITH INTERRUPTS ENABLED
                          439 *-------------------------
       0000 0CE2          440 HALT9    EQU   *                   STOP MACHINE FOR ERROR PRINT
 0CE2  41F0 1274          441          BAL   R15,TSTDU           CHECK IF LIST DEVICE OFF-LINE
 0CE6  2336               442          BZS   $KEEP7              BRANCH: ON-LINE NOW.
 0CE8  C810 080F          443 HALT9A   LHI   R1,X'080F'          **
 0CEC  9114               444          SLHLS R1,4                R1 = X'80F0'
```

```
0CEE  9501                445          EPSR  R0,R1              STOP PROCESSOR. WHEN 'EXE/RUN' DEP
0CF0  2207                446          BS    HALT9              CHECK IF LIST DEVICE ON-LINE.
                          447  *
                          448  * LIST DEVICE WAS OFF-LINE. PRINT TOTAL, TOTERR
                          449  *
      0000 0CF2           450  $KEEP7   EQU   *
0CF2  2400                451          LIS   R0,0
0CF4  4000 1566           452          STH   R0,$WASOU          RESET DU FLAG
0CF8  41F0 0FBA           453          BAL   R15,$PRINT
0CFC  1655                454          DAC   NULLMSG            OUTPUT NULL STRING, CRLF
0CFE  41F0 0FBA           455          BAL   R15,$PRINT
0D02  15A3                456          DAC   TOTMSG             'TOTAL   TOTERR'
0D04  C840 2020           457          LHI   R4,C'  '           SPACES
0D08  4040 35C0           458          STH   R4,$OUTBUF+4
0D0C  4040 35C2           459          STH   R4,$OUTBUF+6
0D10  2404                460          LIS   R0,4
0D12  C820 35BC           461          LDAI  R2,$OUTBUF         DESTINATION
0D16  4810 1568           462          LH    R1,TOTAL
0D1A  41F0 0F18           463          BAL   R15,HEXASC         CONVERT TOTAL
0D1E  4810 156A           464          LH    R1,TOTERR
0D22  2628                465          AIS   R2,8               DESTINATION
0D24  41F0 0F18           466          BAL   R15,HEXASC         CONVERT TOTERR
0D28  240D                467          LIS   R0,X'0D'           CARRIAGE RETURN
0D2A  0202 0004           468          STB   R0,4(R2)           TO TERMINATE MESSAGE.
0D2E  41F0 0FC8           469          BAL   R15,@PRINT         PRINT CONTENTS OF BUFFER:
                          470  *                                TOTAL    TOTERR
                          471  *                                XXXX     YYYY
0D32  4300 0A72           472          B     STARTA             PRINT TITLE, ACCEPT COMMAND.
                          473  * ************************************
0D36                      474          IFZ   $DISPLAY-1
                          475  *
0D36  2401                476  DISPLAY  LIS   R0,1               DISPLAY PANEL ADDRESS
0D38  DE00 154E           477          OC    R0,INCR            INCREMENTAL MODE
0D3C  481F 0002           478          LH    R1,2(R15)          GET 2ND PARAMETER ADDRESS
0D40  4811 0000           479          LH    R1,0(R1)           GET DATA
0D44  9411                480          EXBR  R1,R1
0D46  9801                481          WHR   R0,R1              WRITE DATA
0D48  481F 0000           482          LH    R1,0(R15)          GET 1ST PARAMETER ADDRESS
0D4C  4811 0000           483          LH    R1,0(R1)           GET DATA
0D50  9411                484          EXBR  R1,R1
0D52  9801                485          WHR   R0,R1              WRITE DATA TO D1,D2
0D54  DE00 154D           486          OC    R0,NORM            NORMAL MODE
0D58  430F 0004           487          B     4(R15)             RETURN
                          488  *
                          489  * ************************************
                          490          ENDC
                          491  *
                          492  * ERROR ROUTINES                 (OVERRIDE NOMSG OPTION)
                          493  * RETURN LINK R15; NO REGISTERS MODIFIED.
                          494  *
0D5C  D0F0 35A8           495  ERR      STM   R15,$R15SAV        SAVE LINK
0D60  41F0 0DAA           496          BAL   R15,ERRCOM         'ERROR TTNN'
0D64  0DD6                497          DAC   ERRCOM1            EXIT
```

```
                                  498  *
0D66   D0F0 35A8                  499  ERRD     STM   R15,$R15SAV       SAVE LINK
0D6A   41F0 0DAA                  500           BAL   R15,ERRCOM        'ERROR TTNN'
0D6E   0E0C                       501           DAC   ERRD1             'DEV DDD'
0D70   0DD6                       502           DAC   ERRCOM1           EXIT
                                  503  *
0D72   D0F0 35A8                  504  ERRS     STM   R15,$R15SAV       SAVE LINK
0D76   41F0 0DAA                  505           BAL   R15,ERRCOM        'ERROR TTNN'
0D7A   0E1E                       506           DAC   ERRS1             'STA SS'
0D7C   0DD6                       507           DAC   ERRCOM1           EXIT
                                  508  *
0D7E   D0F0 35A8                  509  ERRDS    STM   R15,$R15SAV       SAVE LINK
0D82   41F0 0DAA                  510           BAL   R15,ERRCOM        'ERROR TTNN'
0D86   0E0C                       511           DAC   ERRD1             'DEV DDD'
0D88   0E1E                       512           DAC   ERRS1             'STA SS'
0D8A   0DD6                       513           DAC   ERRCOM1           EXIT
                                  514  *
0D8C   D0F0 35A8                  515  ERRL     STM   R15,$R15SAV       SAVE LINK
0D90   D0E0 1530                  516           STM   R14,OLDPSW        STORE CALLER'S PSW, LOC
0D94   41F0 0DAA                  517           BAL   R15,ERRCOM        'ERROR TTNN'
0D98   0E72                       518           DAC   ERRL1             'LOC LLLL'
0D9A   0DD6                       519           DAC   ERRCOM1           EXIT
                                  520  *
0D9C   D0F0 35A8                  521  ERRALL   STM   R15,$R15SAV       SAVE LINK
0DA0   41F0 0DAA                  522           BAL   R15,ERRCOM        'ERROR TTNN'
0DA4   0E30                       523           DAC   ERRDS1            'DEV DDD STA SS'
0DA6   0E50                       524           DAC   ERRPL1            'PSW PPPP  LOC LLLL'
0DA8   0DD6                       525           DAC   ERRCOM1           EXIT
                                  526  *
                                  527  *  COMMON ERROR ROUTINE
                                  528  *
0DAA   D000 19D0                  529  ERRCOM   STM   R0,ERRSAVE        STORE USER REGISTER SET
0DAE   4810 0A52                  530           LH    R1,PSW2           SPEC'D AS X'30F0'
0DB2   9501                       531           EPSR  R0,R1             DISABLE INT. @ PROCESSOR LEVEL
0DB4   4800 1590                  532           LH    R0,MTESTNO        MASTER TEST NUMBER
0DB8   4000 159A                  533           STH   R0,ETESTNO        MOVE TO MESSAGE
0DBC   4000 155C                  534           STH   R0,ISITERR        TO FORCE ERROR PRINT
0DC0   26F1                       535           AIS   R15,ADC-1
0DC2   C4F0 FFFE                  536           NHI   R15,0-ADC
0DC6   48CF 0000                  537           LDA   R12,0(R15)        FIRST PARAMETER
0DCA   48DF 0002                  538           LDA   R13,ADC(R15)      SECOND PARAMETER
0DCE   41E0 0E00                  539           BAL   R14,ERR1          'ERROR TTNN'
0DD2   01EC                       540           BALR  R14,R12           GO TO FIRST ROUTINE,
0DD4   01ED                       541           BALR  R14,R13           SECOND ROUTINE,
                                  542  *
0DD6   2400                       543  ERRCOM1  LIS   R0,0
0DD8   4000 155C                  544           STH   R0,ISITERR        RESET ERROR PRINT FLAG
0DDC   2411                       545           LIS   R1,1
0DDE   4010 155E                  546           STH   R1,NOERR          SUPPRESS THAT PRINT
0DE2   6110 156A                  547           AHM   R1,TOTERR         INCREMENT TOTERR
0DE6   2138                       548           BNZS  ERRCOM2           BRANCH: STILL COUNTING,
0DE8   2511                       549           LCS   R1,1              65,535 ERRORS REPORTED
0DEA   4010 156A                  550           STH   R1,TOTERR
```

```
0DEE   41F0 1274   551           BAL   R15,TSTDU          LIST DEVICE OFF-LINE ?
0DF2   4230 0CE2   552           BNZ   HALT9              BRANCH: YES.
0DF6   D100 19D0   553 ERRCOM2   LM    R0,ERRSAVE         RESTORE REGISTERS
0DFA   D1F0 35A8   554           LM    R15,$R15SAV        RESTORE LINK
0DFE   030F        555           BR    R15                RETURN TO CALLER.
                   556 *--------------------------
                   557 * MESSAGE PRINT ROUTINES          (DO NOT OVERRIDE NOMSG OPTION)
                   558 * RETURN LINK R14; REGISTERS MODIFIED R0,R1,R2,R5.
                   559 *
                   560 *  TO PRINT 'ERROR TTNN'
                   561 *
0E00               562           CNOP  ADC                ALIGN PARAMETER
0E00   D0E0 35AC   563 ERR1      STM   R14,$R14SAV        SAVE LINK
0E04   C850 1594   564           LDAI  R5,ERRMSG          PRINT MESSAGE                *
0E08   4300 0EB0   565           B     $MSGPRT1           'ERROR TTNN'                 *
                   566 *                                  TT FROM MTESTNO, NN FROM ERRNO
                   567 *
                   568 *  TO PRINT 'DEV DDD'
                   569 *
0E0C   D0E0 35AC   570 ERRD1     STM   R14,$R14SAV        SAVE LINK
0E10   2403        571           LIS   R0,3               SET UP DIGITS = 3
0E12   4810 154A   572           LH    R1,ERRDEV          R1 = ERROR DEV # IN BINARY
0E16   41E0 0E9C   573           BAL   R14,$MSGPRT        PRINT 'DEV DDD'
0E1A   15D4        574           DAC   ASCIDEV2           HEXASC DESTINATION
0E1C   15D0        575           DAC   DEVMSG2            A(MESSAGE)
                   576 *
                   577 *  TO PRINT 'STA SS'
                   578 *
0E1E   D0E0 35AC   579 ERRS1     STM   R14,$R14SAV        SAVE LINK
0E22   2402        580           LIS   R0,2               SET UP DIGITS = 2
0E24   D310 154C   581           LB    R1,ERRSTA          R1 = ERROR STATUS
0E28   41E0 0E9C   582           BAL   R14,$MSGPRT        PRINT 'STA SS'
0E2C   15CB        583           DAC   ASCISTA            HEXASC DESTINATION
0E2E   15C7        584           DAC   STAMSG             A(MESSAGE)
                   585 *
                   586 *  TO PRINT 'DEV DDD STA SS'
                   587 *
0E30   D0E0 35AC   588 ERRDS1    STM   R14,$R14SAV        SAVE LINK
0E34   2403        589           LIS   R0,3               SET UP DIGITS = 3
0E36   4810 154A   590           LH    R1,ERRDEV          R1 = ERROR DEV #
0E3A   C820 15C3   591           LDAI  R2,ASCIDEV         HEXASC DESTINATION
0E3E   41F0 0F18   592           BAL   R15,HEXASC         CONVERT IT TO ASCII
0E42   2402        593           LIS   R0,2               SET UP DIGITS = 2
0E44   D310 154C   594           LB    R1,ERRSTA          R1 = ERROR STATUS
0E48   41E0 0E9C   595           BAL   R14,$MSGPRT        PRINT 'DEV DDD STA SS'
0E4C   15CB        596           DAC   ASCISTA            HEXASC DESTINATION
0E4E   15BF        597           DAC   DEVMSG             A(MESSAGE)
                   598 *
                   599 *  TO PRINT 'PSW PPPP  LOC LLLL'
                   600 *
0E50   D0E0 35AC   601 ERRPL1    STM   R14,$R14SAV        SAVE REGISTERS
0E54   D1E0 1530   602           LM    R14,OLDPSW         R14 = PSW, R15 = LOC
0E58   081E        603           LDAR  R1,R14             PSW TO PRINT REGISTER
```

```
0E5A                            IFZ   ADC-2
0E5A   2404                     LIS   R0,4           ASSUME SERIES 16
0E5C   4850 1548                LH    R5,MOD32
0E60   2332                     BZS   ERRPL1A
                                ENDC
0E62   2406                     LIS   R0,6           SERIES 32
0E64   C820 15F2       ERRPL1A  LDAI  R2,ASCIPSW     DESTINATION
0E68   C850 15EE                LDAI  R5,PSWMSG
0E6C   41F0 0F18                BAL   R15,HEXASC     CONVERT PSW
0E70   2305                     BS    ERRPL1B        GO CONVERT LOC
                       *
                       *  TO PRINT 'LOC LLLL'
                       *
0E72   D0E0 35AC       ERRL1    STM   R14,$R14SAV    SAVE REGISTERS
0E76   C850 15FA                LDAI  R5,LOCMSG      A(MESSAGE)
0E7A   D1E0 1530       ERRPL1B  LM    R14,OLDPSW     R15 = OLD LOC TO PRINT
0E7E   081F                     LDAR  R1,R15         DATA TO PRINT REGISTER
0E80                            IFZ   ADC-2
0E80   2404                     LIS   R0,4           ASSUME SERIES 16
0E82   48F0 1548                LH    R15,MOD32
0E86   2332                     BZS   ERRL1A
                                ENDC
0E88   2406                     LIS   R0,6           SERIES 32
0E8A   C820 15FE       ERRL1A   LDAI  R2,ASCILOC     DESTINATION
0E8E   41F0 0F18                BAL   R15,HEXASC     CONVERT
0E92   41F0 0FD2                BAL   R15,PRINT      PRINT
0E96   D1E0 35AC                LM    R14,$R14SAV    RESTORE LINK
0E9A   030E                     BR    R14            RETURN
                       *
                       * ROUTINE IS CALLED BY MESSAGE PRINT ROUTINES
                       *
0E9C   26E1            $MSGPRT  AIS   R14,ADC-1
0E9E   C4E0 FFFE                NHI   R14,0-ADC
0EA2   482E 0000                LDA   R2,0(R14)      HEXASC DESTINATION
0EA6   41F0 0F18                BAL   R15,HEXASC     CONVERT DATA TO HEXADECIMAL
0EAA   26E2                     AIS   R14,ADC
0EAC   485E 0000                LDA   R5,0(R14)      A(MESSAGE TO PRINT)              *
0EB0   41F0 0FD2       $MSGPRT1 BAL   R15,PRINT      PRINT SPECIFIED MESSAGE          *
0EB4   D1E0 35AC                LM    R14,$R14SAV
0EB8   030E                     BR    R14            RETURN TO ORIGINAL CALLER
                       * ----------------------------
                       *
                       * TO OBTAIN OPTION VALUE IN R6    (R7:R6, TARGT 16)
                       * RETURNS WHEN SPECIAL CHARACTER FOUND. IGNORES SPACES.
                       *
0EBA   2460            OPTVAL   LIS   R6,0           INITIALIZE ACCUMULATOR
0EBC                            IFZ   ADC-2
0EBC   2470                     LIS   R7,0           HIGH-ORDER BITS, TARGT 16
                                ENDC
0EBE   D343 360C       $OPTV.0  LB    R4,$INBUF(R3)  GET NEXT INPUT CHARACTER
0EC2   C530 0050                CLHI  R3,$BUFLEN     AT END OF INPUT BUFFER ?
0EC6   038E                     BNLR  R14            RETURN IF YES.
0EC8   2631                     AIS   R3,1           ADVANCE BUFFER POINTER
```

```
0ECA  C540 0020          657            CLHI  R4,C' '            SPACE ?
0ECE  2238               658            BES   $OPTV.0           BRANCH: YES. IGNORE.
0ED0  C540 0030          659            CLHI  R4,C'0'           LESS THAN ZERO ?
0ED4  028E               660            BLR   R14               RETURN IF SPECIAL CHARACTER
0ED6  24FF               661            LIS   R15,15
0ED8  D44F 157A          662  $OPTV.2   CLB   R4,HEXTAB(R15)    SCAN TABLE
0EDC  2334               663            BES   $OPTV.3           MATCH
0EDE  27F1               664            SIS   R15,1
0EE0  2214               665            BNMS  $OPTV.2
0EE2  030C               666            BR    R12               ERROR: VALUE NOT IN TABLE.
0EE4  EB60 0004          667  $OPTV.3   RLL   R6,4               (R6:R7), SERIES 16
0EE8  066F               668            OAR   R6,R15            OR IN CURRENT DIGIT
0EEA  4300 0EBE          669            B     $OPTV.0
                         670  *-------------------------------
                         671  * TO CONVERT (R6) FROM BINARY TO UNARY PATTERN, IN R3
                         672  *
0EEE  2438               673  UNARY     LIS   R3,8              BIT TO SHIFT
0EF0  913C               674            SLHLS R3,12             R3 = '8000'
0EF2  CC36 0000          675            SRHL  R3,0(R6)          SHIFT TO DESIRED POSITION
0EF6  030E               676            BR    R14               AND RETURN.
                         677  *
                         678  *-----------------------------
                         679  *
0EF8                     680            IFNZ  $CLOCK            0 = INCLUDE NO TIMERS
      0000 0EF8          681  $TIMER    EQU   *
0EF8                     682            IFZ   $CLOCK-3          3 = INCLUDE BOTH
                         684            ELSE
      0000 0EF8          685  TIMER     EQU   *
                         686            ENDC
                         687  *  TO PROVIDE # OF MILLISECONDS DELAY SPECIFIED BY R0
                         688  *
0EF8  D000 409C          689            STM   R0,RSAVE          SAVE REGISTERS
0EFC  2410               690  $STIM1    LIS   R1,0
0EFE  2421               691            LIS   R2,1
0F00  4830 0A5A          692            LH    R3,$TIMVAL        (R3) = CONSTANT FOR 1 MSEC DELAY
0F04  C110 0F04          693            BXLE  R1,*
0F08  2701               694            SIS   R0,1
0F0A  2037               695            BNZS  $STIM1            LOOP TILL SPECIFIED DELAY
*0F0C                    696            B     $TIMRET           RELOAD REGISTERS, RETURN (R15)
                         697  *
                         698  *-------------------------------
                         699  *
0F0C  D100 409C          721  $TIMRET  LM    R0,RSAVE          RESTORE USER'S REGISTERS
0F10  030F               722  $TIMXT   BR    R15               AND RETURN.
                         724  * ---------------------------
                         725  * ROUTINE RESTORES REGISTERS SAVED ON ENTRY TO CALLING ROUTINE
                         726  * AND RETURNS BY R15
                         727  *
0F12  D100 409C          728  $RSAVRET LM    R0,RSAVE
0F16  030F               729            BR    R15               RETURN TO ORIGINAL CALLER
                         730  ****** THIS IS WHERE TO IMPLEMENT STACK
                         731  *
                         753  *---------------------------
```

```
                                754  *  TO CONVERT HEXADECIMAL DATA IN R1 TO ASCII CHAR & STORE @ 0(R2)
                                755  * OUTPUTS UP TO 4 DIGITS (8 DIGITS, SERIES 32)
                                756  *
0F18   D000 409C                757  HEXASC   STM   R0,RSAVE          STORE REGISTERS
0F1C   0830                     758           LDAR  R3,R0             R3 = DIGITS
0F1E   9132                     759           SLLS  R3,2
0F20   2734                     760           SIS   R3,4              R3 = 4(DIGITS)-4
0F22   0841                     761  $HEXA.1  LDAR  R4,R1             R4 = HEX DATA
0F24   EC43 0000                762           SRL   R4,0(R3)
0F28   C440 000F                763           NHI   R4,15             R4 = HEX DIGIT TO BE CONVERTED
0F2C   D344 157A                764           LB    R4,HEXTAB(R4)
0F30   D242 0000                765           STB   R4,0(R2)          STORE ASCII CHAR
0F34   2621                     766           AIS   R2,1
0F36   2734                     767           SIS   R3,4
0F38   221B                     768           BNMS  $HEXA.1           LOOP TILL ALL DIGITS
0F3A   4300 0F12                769           B     $RSAVRET          RESTORE REGISTERS, RETURN (R15)
                                796  * -------------------------------
                                797  * TO OUTPUT LIST OF BITS IN ASCENDING NUMERIC ORDER,
                                798  * STARTING FROM HIGH-ORDER BIT AS BIT 0
                                799  * DOES NOT OVERLAY OPTION NAME IN $OUTBUF.
                                800  *
       0000 0F3E                801  $LSTBIT  EQU   *
0F3E   D000 19D0                802           STM   R0,ERRSAVE        SAVE REGISTERS
0F42   2401                     803           LIS   R0,1              DIGITS TO OUTPUT
0F44   2410                     804           LIS   R1,0              STARTING WITH NUMBER 0
0F46   2470                     805           LIS   R7,0              PRINT FLAG
0F48   4835 0000                806           LH    R3,0(R5)          LOW-NUMBERED PARAMETER BITS
0F4C   2136                     807           BNZS  $LSTB.B           BRANCH: ONE SET
0F4E   4835 0002                808  $LSTB.A  LH    R3,2(R5)          HIGH-NUMBERED PARAMETER BITS
0F52   2402                     809           LIS   R0,2              2 DIGITS NEEDED FOR HEXASC
0F54   C810 0010                810           LHI   R1,X'10'          BIT NUMBER BASE
0F58   2428                     811  $LSTB.B  LIS   R2,$CKROUT+2      NO OVERLAY OF OPTION NAME
0F5A   9131                     812  $LSTB.0  SLHLS R3,1              TEST LEFTMOST HALFWORD BIT
0F5C   4380 0F7E                813           BNC   $LSTB.2A          BRANCH: ZERO.
0F60   C520 0008                814           CLHI  R2,$CKROUT+2      ANY OUTPUT YET ?
0F64   2336                     815           BES   $LSTB.1           BRANCH: NO
0F66   C840 002C                816           LHI   R4,C','           COMMA
0F6A   D242 35BC                817           STB   R4,$OUTBUF(R2)    INSERT IN BUFFER
0F6E   2621                     818           AIS   R2,1
0F70   08D2                     819  $LSTB.1  LDAR  R13,R2            SAVE BUFFER OFFSET
0F72   CA20 35BC                820           AHI   R2,$OUTBUF        HEXASC DESTINATION
0F76   41F0 0F18                821           BAL   R15,HEXASC        CONVERT BIT NUMBER
0F7A   082D                     822           LDAR  R2,R13            GET OFFSET
0F7C   0A20                     823  $LSTB.2  AAR   R2,R0             INCREMENT BUFFER POINTER
0F7E   2611                     824  $LSTB.2A AIS   R1,1              INCREMENT BIT NUMBER
0F80   C310 000F                825           THI   R1,15             HALFWORD COMPLETED ?
0F84   4230 0F5A                826           BNZ   $LSTB.0           BRANCH: NO.
0F88   244D                     827           LIS   R4,X'0D'          CARRIAGE RETURN
0F8A   D242 35BC                828           STB   R4,$OUTBUF(R2)    INSERT IN BUFFER
0F8E   0672                     829           OAR   R7,R2             ACCUMULATE HIGHEST BYTE COUNT
0F90   C520 0008                830           CLHI  R2,$CKROUT+2      ANY OUTPUT THIS TIME ?
0F94   2333                     831           BES   $LSTB.2B          BRANCH: NO.
0F96   41F0 0FC8                832           BAL   R15,@PRINT        PRINT THE BUFFER.
```

EXEC - ETPE R05P1

```
0F9A  C510 0020      833  $LSTB.2B CLHI  R1,32            FULLWORD COMPLETED ?
0F9E  4280 0F4E      834           BL    $LSTB.A          BRANCH: NO.
0FA2  2778           835           SIS   R7,$CKROUT+2     ANY OUTPUT DONE ?
0FA4  2133           836           BNZS  $LSTB.2C         BRANCH: YES.
0FA6  41F0 0FC8      837           BAL   R15,@PRINT       PRINT OPTION NAME IN BUFFER.
0FAA  D100 19D0      838  $LSTB.2C LM    R0,ERRSAVE
0FAE  030F           839           BR    R15              RETURN
                     840  * ----------------------------
                     856  *  TO OUTPUT CR,LF TO LIST DEVICE
                     857  *
0FB0  D000 409C      858  CRLF     STM   R0,RSAVE         SAVE REGISTERS
0FB4  C850 1659      859           LDAI  R5,CRLFMSG       CR, LF
* 0FB8  230F         860           B     $P1              GO PRINT LINE.
                     861  *
0FBA  26F1           862  $PRINT   AIS   R15,ADC-1
0FBC  C4F0 FFFE      863           NHI   R15,0-ADC
0FC0  485F 0000      864           LDA   R5,0(R15)        A(MESSAGE TO PRINT)
0FC4  26F2           865           AIS   R15,ADC
0FC6  2306           866           BS    $P0
                     867  *
0FC8  D000 409C      868  @PRINT   STM   R0,RSAVE         SAVE REGISTERS
0FCC  C850 35BC      869           LDAI  R5,$OUTBUF       TO PRINT OUTPUT BUFFER
0FD0  2303           870           BS    $P1
                     871  *
      0000 0FD2      872  PRINT    EQU   *                TO PRINT THE ASCII MESSAGE
0FD2  D000 409C      873  $P0      STM   R0,RSAVE         STORE REGISTERS
0FD6  2400           874  $P1      LIS   R0,0
0FD8  4000 1562      875           STH   R0,$LINEPOS      RESET BUFFER
0FDC  41F0 1274      876           BAL   R15,TSTDU        IS DEVICE UNAVAILABLE ?
0FE0  4230 0F12      877           BNZ   $RSAVRET         IF YES, RELOAD REGISTERS, RETURN.
                     878  *
0FE4  4810 1566      879           LH    R1,$WASDU        WAS DEVICE EVER SEEN DU ?
0FE8  4230 0CE2      880           BNZ   HALT9            OUTPUT TOTAL, TOTERR.
                     881  *
0FEC  4800 155C      882           LH    R0,ISITERR       AN ERROR MESSAGE ?
0FF0  4500 16B8      883           CLH   R0,NOMSG+$VALU1  IF SO, CAN BE SUPPRESSED ?
0FF4  4280 0F12      884           BL    $RSAVRET         BRANCH: MESSAGE IS SUPPRESSED.
0FF8  4800 16DE      885  $PRT.1   LH    R0,$LINCNT       *                                *
0FFC  2601           886           AIS   R0,1             INCREMENT LISTING LINE COUNT     *
0FFE  4000 16DE      887           STH   R0,$LINCNT       *                                *
1002  C500 0038      888           CLHI  R0,MAXLINE       AT LIMIT?                        *
1006  2184           889           BLS   $PRT.2           SKIP IF NO                       *
1008  2400           890           LIS   R0,0             *                                *
100A  4000 16DE      891           STH   R0,$LINCNT       RESET COUNT TO ZERO              *
                     892  *
100E  D345 0000      893  $PRT.2   LB    R4,0(R5)         GET A MESSAGE BYTE
1012  41F0 1054      894           BAL   R15,OUTCHR       OUTPUT IT
1016  0844           895           LDAR  R4,R4            ZERO BYTE?                       *
1018  233A           896           BZS   $PRT.3           MSG OVER
101A  2651           897           AIS   R5,1
101C  C350 0002      898           THI   R5,2             TIME TO CHECK BREAK ?
1020  2239           899           BZS   $PRT.2           BRANCH: NO.
1022  4050 1564      900           STH   R5,$PRTFLG       TO DEFER BREAK ACKNOWLEDGE
```

EXEC - ETPE R05P1

```
1026  41F0 11E4          901            BAL   R15,TSTBRK
102A  220E               902            BS    $PRT.2              LOOP FOR NEXT CHAR
                         903  *
                         904  *$PRT.3   LIS   R4,X'0A'           LF
                         905  *         BAL   R15,OUTCHR         LF
                         906  *         LIS   R4,0                ASCII 'NUL'
                         907  *         BAL   R15,OUTCHR         TERMINAL CHARACTER
102C  41F0 11E4          908  $PRT.3    BAL   R15,TSTBRK
1030  4040 1564          909            STH   R4,$PRTFLG          RE-ENABLE BREAK ACKNOWLEDGE
1034  48F0 1558          910            LH    R15,$BRKFLG
1038  4040 1558          911            STH   R4,$BRKFLG          BREAK BEING ACKNOWLEDGED
103C  4330 0F12          912            BZ    $RSAVRET            RESTORE REGISTERS, RETURN (R15)
1040  40F0 155C          913            STH   R15,ISITERR         FORCE MESSAGE PRINT
1044  C550 1654          914            CLAI  R5,$BRKEND          PRINTING 'BRK TERM' MESSAGE ?
1048  2334               915            BES   $PRT.4              BRANCH: YES.
104A  41F0 0FBA          916            BAL   R15,$PRINT          'RECURSIVE' CALL
104E  1639               917            DAC   BRKMSG              'BREAK TERMINATION'
1050  4300 0AB2          918  $PRT.4    B     OPTIN1              TO CMD PROCESSOR
                         919  *-------------------------------
                         920  *  TO OUTPUT A CHARACTER TO THE LIST DEVICE
                         921  *
1054  40F0 1574          922  OUTCHR    STA   R15,OUT.SAV        SAVE RETURN ADDRESS
1058  D310 1558          923            LB    R1,IOSAVE+1
105C  2714               924            SIS   R1,4
105E  4230 108E          925            BNZ   $OTC.4              BRANCH IF NOT CAROUSEL
1062  4010 1570          926  $OTC.0    STH   R1,$PAUSE           ZERO $PAUSE FLAG
1066  41F0 1274          927  $OTC.1    BAL   R15,TSTOU           ON LINE ?
106A  4230 10C6          928            BNZ   $OTC.7              BRANCH: OFFLINE, EXIT.
106E  9D21               929            SSR   R2,R1               GET CAROUSEL STATUS
1070  2385               930            BFFS  8,$OTC.3            BRANCH IF CHAR. IS TO BE READ
1072  4810 1570          931  $OTC.2    LH    R1,$PAUSE           PAUSED NOW ?
1076  2038               932            BNZS  $OTC.1              YES, LOOP
1078  230B               933            BS    $OTC.4              NO, GO OUTPUT CHARACTER
107A  9B21               934  $OTC.3    RDR   R2,R1               GET CAROUSEL CHARACTER
107C  C410 007F          935            NHI   R1,X'7F'
1080  C510 0014          936            CLHI  R1,X'14'            DC4 ?
1084  4330 1062          937            BE    $OTC.0              DC4, SET $PAUSE FLAG.
1088  CB10 0012          938            SHI   R1,X'12'            DC2 ?
108C  203D               939            BNZS  $OTC.2              BRANCH: NO, CHECK IF PAUSED NOW.
                         940  *
108E  4010 1570          941  $OTC.4    STH   R1,$PAUSE           RESET FLAG
1092  4110 12EC          942            BAL   R1,$SETUP           SET UP FOR OUTPUT
1096  9D01               943  $OTC.5    SSR   R0,R1               WAIT FOR NOT BUSY
1098  4230 10C6          944            BTC   3,$OTC.7            BRANCH IF OFF-LINE
109C  C510 0048          945            CLHI  R1,X'48'            NOT CL2S OR PF                    *
10A0  4330 10C6          946            BE    $OTC.7              RETURN IF YES                     *
10A4  C410 00FC          947            NHI   R1,X'FC'
10A8  C510 000C          948            CLHI  R1,X'0C'            HDX PASLA OFF-LINE ?
10AC  233D               949            BES   $OTC.7              BRANCH: YES.
10AE  C310 0008          950            THI   R1,8                BUSY ?
10B2  203E               951            BNZS  $OTC.5              WAIT FOR NOT BUSY.
10B4  9A04               952            WDR   R0,R4               OUTPUT DATA BYTE
10B6  9D01               953  $OTC.6    SSR   R0,R1               WAIT FOR NOT BUSY
```

EXEC - ETPE R05P1

```
10B8  2177           954          BTFS  7,$OTC.7          BRANCH IF OFF-LINE (PASLA HANGS)
10BA  C510 0048      955          CLHI  R1,X'48'          NOT CL2S OR PF                  *
10BE  2334           956          BES   $OTC.7            RETURN IF YES                   *
10C0  C310 0008      957          THI   R1,8              WAIT FOR NON BUSY               *
10C4  2037           958          BNZS  $OTC.6            *                               *
10C6  48F0 1574      959  $OTC.7  LDA   R15,OUT.SAV
10CA  030F           960          BR    R15               RETURN
                     961  * ---------------------------------
                     979  * ROUTINE GETS INPUT RECORD
                     980  *
10CC  D000 409C      981  $READ    STM   R0,RSAVE          SAVE REGISTERS
10D0  25D1           982  $RD.1   LCS   R13,1             INITIALIZE
10D2  26D1           983  $RD.2   AIS   R13,1             INCREMENT BUFFER POINTER
10D4  40D0 1562      984          STH   R13,$LINEPOS      ADDRESS OF CURRENT BYTE
10D8  4140 12D0      985  $RD.3   BAL   R4,KBREAD         PUT DEVICE IN READ MODE
10DC  9D04           986          SSR   R0,R4
10DE  2081           987          BTBS  8,1               IF BUSY, LOOP (POSSIBLE HANG)
10E0  9B04           988          RDR   R0,R4             READ A CHAR IN R4
10E2  D390 0A10      989          LB    R9,IO             WHAT TYPE DEVICE ?          P1 10/79
10E6  2792           990          SIS   R9,2              TYPE 2 ?                    P1 10/79
10E8  2338           991          BZS   $RD.3A            BRANCH: YES, E-PLEX ON.     P1 10/79
10EA  4890 0A2C      992          LH    R9,CONWADR        GET WRITE ADDRESS
10EE  DE90 0A2F      993          OC    R9,CONWRT         TURN DEVICE AROUND
10F2  9D93           994          SSR   R9,R3
10F4  2081           995          BTBS  8,1               WAIT FOR BUSY NOT
10F6  9A94           996          WDR   R9,R4             ECHO RECEIVED BYTE
      0000 10F8      997  $RD.3A  EQU   *                 P1 10/79
10F8  C440 007F      998          NHI   R4,X'7F'          REMOVE PARITY BIT
10FC  C540 0040      999          CLHI  R4,C'@'           COMMERCIAL AT SIGN?             **
1100  4330 0A5C     1000          BE    $CON              BREAK OUT IF YES                **
1104  C540 0060     1001          CLHI  R4,X'60'          UPPER-CASE CHARACTER ?
1108  2183          1002          BLS   $RD.4             BRANCH: NO.
110A  CB40 0020     1003          SHI   R4,X'20'          CONVERT TO LOWER-CASE
110E  C540 0023     1004  $RD.4   CLHI  R4,X'23'          HASH-MARK ?
1112  4330 0AAA     1005          BE    OPTIN             BRANCH: YES, GO TO CMD PROC.
1116  C540 0018     1006          CLHI  R4,X'18'          ASCII 'CANCEL' CHARACTER ?
111A  4330 0AAA     1007          BE    OPTIN             BRANCH: YES.
111E  C540 005F     1008          CLHI  R4,X'5F'          BACKARROW, UNDERLINE, DELETE ?
1122  2334          1009          BES   $RD.5             BRANCH: DELETE LAST CHARACTER
1124  C540 0008     1010          CLHI  R4,X'08'          BACKSPACE ?
1128  2136          1011          BNES  $RD.6             BRANCH: NO.
112A  27D2          1012  $RD.5   SIS   R13,2             TO DELETE LAST CHARACTER
112C  4210 10D0     1013          BM    $RD.1             BRANCH: NO UNDERFLOW ALLOWED.
1130  4300 10D2     1014          B     $RD.2             GET ANOTHER CHARACTER
1134  D24D 360C     1015  $RD.6   STB   R4,$INBUF(R13)    STORE CURRENT INPUT BYTE
1138  C540 000D     1016          CLHI  R4,X'0D'          CARRIAGE RETURN ?
113C  2135          1017          BNES  $RD.7             BRANCH: NOT YET.
113E  C850 1659     1018          LOAI  R5,CRLFMSG
1142  4300 0FD6     1019          B     $P1               OUTPUT (CR),(LF) TO CONSOLE, RETURN.
1146  C5D0 004F     1020  $RD.7   CLHI  R13,$BUFLEN-1     BUFFER AT MAX ?
114A  4280 10D2     1021          BL    $RD.2             BRANCH: NOT YET.
114E  4300 10D8     1022          B     $RD.3             BRANCH: FORCE OVERLAY OF LAST CHARACT
                    1023  *
```

EXEC - ETPE R05P1

```
                        1024  * ------------------------------
                        1025  * SET UP FOR CONSOLE, LIST I/O DEVICES
                        1026  *
1152  0310 0A10         1027  STCON    LB    R1,IO              GET I/O IDENTIFIERS
1156  0320 0A11         1028           LB    R2,IO+1
115A  2436              1029           LIS   R3,$MAXIO          IDENTIFIER CAN BE 1,2,3,4,5
115C  0513              1030           CLAR  R1,R3
115E  2182              1031           BLS   $STC.1             BRANCH IF KB IDENTIFIER OK
1160  2411              1032           LIS   R1,1               ELSE FORCE CRT
1162  0523              1033  $STC.1   CLAR  R2,R3
1164  2182              1034           BLS   $STC.2             SAME TEST FOR LIST DEVICE
1166  2421              1035           LIS   R2,1
1168  0210 0A10         1036  $STC.2   STB   R1,IO              REESTABLISH VALUES
116C  0220 0A11         1037           STB   R2,IO+1
1170  0362 0A48         1038           LB    R6,CONRQ2S(R2)
1174  4060 1554         1039           STH   R6,$LSTPAS         SET PASLA FLAG (LIST DEVICE)
1178  0866              1040           LDAR  R6,R6
117A  2336              1041           BZS   $STC.3             SKIP IF NOT PASLA
117C  9122              1042           SLHLS R2,2
117E  4802 0A10         1043           LH    R0,IO(R2)
1182  DE02 0A32         1044           OC    R0,CONCMD(R2)      ISSUE 2ND COMMAND (TO LIST DEVICE***
                        1045  *
1186  41F0 12BC         1046  $STC.3   BAL   R15,SETKB          ESTABLISH KEYBOARD DEVICE (& IOSAVE)
118A  9310              1047           LBR   R1,R0              (R1) = 1,2,4,5 ; (R0 = KBIDENT)
118C  9112              1048           SLHLS R1,2               (R1)=4,8,16,20
118E  2712              1049           SIS   R1,2
1190  4831 0A10         1050           LH    R3,IO(R1)
1194  4030 0A2A         1051           STH   R3,CONRADR         SET UP CONSOLE DEVICE READ ADDRESS
1198  4831 0A12         1052           LH    R3,IO+2(R1)
119C  4030 0A2C         1053           STH   R3,CONWADR         SET UP CONSOLE WRITE ADDRESS
11A0  4821 0A32         1054           LH    R2,CONCMD(R1)
11A4  4020 0A2E         1055           STH   R2,CONRD           SET UP R/W COMMANDS
11A8  4821 0A34         1056           LH    R2,CONCMD+2(R1)
11AC  4020 0A30         1057           STH   R2,CON2ND          2ND CMD: ENABLE READ CMD
11B0  9310              1058           LBR   R1,R0
11B2  D341 0A48         1059           LB    R4,CONRQ2S(R1)
11B6  D240 0A48         1060           STB   R4,CONRQ2S         CONSOLE REQUEST TO SEND
11BA  4040 1552         1061           STH   R4,$CONPAS         SET PASLA FLAG (CONSOLE)
11BE  0844              1062           LDAR  R4,R4
11C0  2333              1063           BZS   $STC.4             SKIP 2ND OC IF NOT PASLA DEVICE
11C2  9422              1064           EXBR  R2,R2
11C4  9E32              1065           OCR   R3,R2              ISSUE 2ND COMMAND (TO CONSOLE)
11C6  DE30 0A2E         1066  $STC.4   OC    R3,CONRD          PUT CONSOLE IN READ MODE
11CA  9832              1067           RDR   R3,R2              READ A DUMMY CHARACTER (SET BUSY)
11CC  0844              1068           LDAR  R4,R4              CONSOLE PASLA DEVICE ?
11CE  2333              1069           BZS   $STC.5             BRANCH: NO.
11D0  DE30 0A48         1070           OC    R3,CONRQ2S         REQUEST TO SEND (KEEP ON-LINE)
      0000 11D4         1071  $STC.5   EQU   *
11D4  030E              1072           BR    R14                RETURN
                        1073  *------------------------------
                        1074  *  TO OUTPUT '?' TO CONSOLE
                        1075  *
11D6  41F0 12BC         1076  QUESTN   BAL   R15,SETKB          SELECT KEYBOARD DEVICE
```

EXEC - ETPE R05P1

```
11DA   41F0 0FBA          1077            BAL   R15,$PRINT
11DE   1630               1078            DAC   QMSG              QUESTION MARK, CRLF
11E0   4300 0AB2          1079            B     OPTIN1            ACCEPT NEXT COMMAND
                          1080  *-----------------------------
                          1081  * IF BREAK KEY DEPRESSED, GO TO 'OPTIN' OR (BRKVECT); ELSE RETURN.
                          1082  *
       0000 11E4          1083  TSTBRK  EQU   *
11E4   D0E0 35A0          1084            STM   R14,$TBRKSV       SAVE REGISTERS
11E8   48F0 1688          1085            LH    R15,NOMSG+$VALU1  (R15) = 15 IF IGNORING I/O
11EC   46F0 1558          1086            OH    R15,$BRKFLG       (R15) = 15 IF BRK ALREADY SEEN
11F0   27FF               1087            SIS   R15,15            LOOK FOR BREAK ?
11F2   2137               1088            BNZS  $TSTB.2           BRANCH: YES.
                          1089  *
11F4   24F0               1090  $TSTB.1 LIS   R15,0
11F6   40F0 1556          1091            STH   R15,BRKVECT       CANCEL BREAK VECTOR
11FA   D1E0 35A0          1092  $TSTB.1A LM   R14,$TBRKSV       RELOAD REGISTERS,
11FE   030F               1093            BR    R15               RETURN TO CALLER.
                          1094  *
1200   48E0 0A2A          1095  $TSTB.2 LH    R14,CONRADR       READ SIDE ADDRESS FOR TERMINAL
1204   D3F0 0A10          1096            LB    R15,IO            CONSOLE ID CODE
1208   C5F0 0002          1097            CLHI  R15,2
120C   2333               1098            BES   $TSTB.3           BRANCH: TTY
120E   C5F0 0005          1099            CLHI  R15,5
1212   4330 1250          1100  $TSTB.3 BE    $TSTB.5           BRANCH: MICRO-I/O BUS
1216   9DEF               1101            SSR   R14,R15
1218   4280 11F4          1102            BTC   8,$TSTB.1         BRANCH: BSY = NO BRK
121C   C3F0 0020          1103            THI   R15,X'20'
1220   4330 11F4          1104            BZ    $TSTB.1           BRANCH: NO FRERR = NO BRK
1224   9BEF               1105            RDR   R14,R15
1226   08FF               1106            LDAR  R15,R15
1228   4230 11F4          1107            BNZ   $TSTB.1           BRANCH: NONZERO CHAR = NO BRK
                          1108  *
       0000 122C          1109  $TSTB.4 EQU   *                   IT IS BREAK
122C   24FF               1110            LIS   R15,15
122E   40F0 1558          1111            STH   R15,$BRKFLG       SET FLAG
1232   48F0 1564          1112            LH    R15,$PRTFLG       PRINTING NOW ?
1236   4230 11FA          1113            BNZ   $TSTB.1A          BRANCH: YES.
123A   24E0               1114            LIS   R14,0
123C   48F0 1556          1115            LH    R15,BRKVECT       SPECIFIED VECTOR
1240   40E0 1556          1116            STH   R14,BRKVECT       CANCEL VECTOR
1244   023F               1117            BNZR  R15               BUT TAKE IMMEDIATELY IF NON-ZERO.
1246   41F0 0FBA          1118            BAL   R15,$PRINT
124A   1639               1119            DAC   BRKMSG            'BREAK TERMINATION'
124C   4300 0AB2          1120            B     OPTIN1
                          1121  *
1250   9DEF               1122  $TSTB.5 SSR   R14,R15
1252   C3F0 0020          1123            THI   R15,X'20'
1256   2134               1124            BNZS  $TSTB.6           BRANCH: BRK.
1258   9BEF               1125            RDR   R14,R15           IF BRK QUEUED, SEE IT NEXT TIME.
125A   4300 11F4          1126            B     $TSTB.1           BRANCH: NO FRERR = NO BRK
125E   9BEF               1127  $TSTB.6 RDR   R14,R15           READ BREAK CHAR
1260   C8F0 8000          1128            LHI   R15,X'8000'
1264   26F1               1129  $TSTB.7 AIS   R15,1
```

EXEC - ETPE R05P1

```
1266  2031           1130          BNZS  $TSTB.7
1268  9DEF           1131          SSR   R14,R15
126A  C3F0 0020      1132          THI   R15,X'20'          BRK KEY STILL DOWN ?
126E  2038           1133          BNZS  $TSTB.6            BRANCH: YES.
1270  4300 122C      1134          B     $TSTB.4            GO SERVICE BREAK
                     1135  *--------------------------
                     1136  * SEE IF CURRENT LIST DEVICE IS OFF-LINE (R1 & CC NON-ZERO IF OFF)
                     1137  *
1274  241F           1138  TSTDU   LIS   R1,15
1276  4510 1688      1139          CLH   R1,NOMSG+$VALU1    IGNORING I/O ?
127A  4330 12AC      1140          BE    $IS.DU             BRANCH: CONSIDER AS DU.
127E  4800 1552      1141          LH    R0,$CONPAS
1282  D310 155B      1142          LB    R1,IOSAVE+1        LIST DEVICE ID
1286  D410 0A10      1143          CLB   R1,IO              SAME AS CONSOLE DEVICE ?
128A  2333           1144          BES   $TSTDU.1           BRANCH: YES.
128C  4800 1554      1145          LH    R0,$LSTPAS         NON-ZERO IF LIST DEVICE ON PASLA.
1290  9112           1146  $TSTDU.1 SLLS  R1,2
1292  4821 0AUE      1147          LH    R2,PASLADR-4(R1)   'READ SIDE' ADDRESS
1296  9D21           1148          SSR   R2,R1              GET DEVICE STATUS
1298  211A           1149          BTFS  1,$IS.DU
129A  0800           1150          LDAR  R0,R0              DEVICE ON PASLA ?
129C  2336           1151          BZS   $NOT.DU
129E  C410 00FC      1152          NHI   R1,X'FC'
12A2  C510 000C      1153          CLHI  R1,X'0C'           PASLA DU IF BSY+EX SET HERE
12A6  2333           1154          BES   $IS.DU             BRANCH: DU.
12A8  2410           1155  $NOT.DU  LIS   R1,0
12AA  2302           1156          BS    $DU.X
12AC  2511           1157  $IS.DU   LCS   R1,1
12AE  4800 1566      1158  $DU.X    LH    R0,$WASDU          GET OLD FLAG
12B2  0601           1159          OAR   R0,R1
12B4  4000 1566      1160          STH   R0,$WASDU          ACCUMULATE
12B8  0811           1161          LDAR  R1,R1              SET CC <> 0 IF DU
                     1162  *                                OR CC = 0 IF NOT DU
12BA  030F           1163          BR    R15                RETURN
                     1164  * --------------------------------
                     1165  * TO DIRECT INPUT AND OUTPUT TO CONSOLE DEVICE
                     1166  *
12BC  D300 0A10      1167  SETKB    LB    R0,IO              GET KEYBOARD DEVICE
12C0  D200 155B      1168          STB   R0,IOSAVE+1        SET LIST TO KEYBOARD
12C4  030F           1169          BR    R15                RETURN
                     1170  * --------------------------
                     1171  * TO RESELECT USER'S I/O CHOICE
                     1172  *
12C6  4800 0A10      1173  SETLST   LH    R0,IO
12CA  4000 155A      1174          STH   R0,IOSAVE
12CE  030F           1175          BR    R15                RETURN
                     1176  *--------------------------
                     1177  *  TO PUT KEYBOARD DEVICE IN READ MODE
                     1178  *
12D0  4800 0A2A      1179  KBREAD   LH    R0,CONRADR
12D4  DE00 0A2E      1180          OC    R0,CONRD           OC CONSOLE - READ COMMAND
12D8  0800 1550      1181          RD    R0,SINK            READ A DUMMY CHARACTER (SET BUSY)
12DC  4890 1552      1182          LH    R9,$CONPAS         PASLA ?
```

```
12E0  4200 12E0          1183           NOP   *                FOR SPECIAL KB DEVICE
12E4  2333               1184           BZS   $KBR.1           NO, BRANCH TO EXIT
12E6  DE00 0A48          1185           OC    R0,CONRQ2S       YES, OC (REQUEST TO SEND)
12EA  0304               1186  $KBR.1   BR    R4               RETURN
                         1199  *---------------------------
                         1200  *  LIST DEVICE SET UP ROUTINE
                         1201  *
12EC  4010 1578          1202  $SETUP   STA   R1,SET.RTN
12F0  D310 155B          1203           LB    R1,IOSAVE+1      GET LIST DEVICE IDENTIFIER
12F4  9112               1204           SLHLS R1,2             HW INDEX
12F6  4801 0A10          1205           LH    R0,IO(R1)        GET LIST DEVICE WRITE ADDRESS
12FA  DE01 0A31          1206           OC    R0,CONCMD-1(R1)
12FE  4810 1578          1207           LDA   R1,SET.RTN
1302  0301               1208           BR    R1               RETURN
                         1209  * ***************************
                         1210  *  LOW CORE SET UP ROUTINE
                         1211  *
1304  D0E0 35AC          1212  LCORE    STM   R14,$R14SAV      SAVE REGISTERS
1308  2400               1213           LIS   R0,0
130A  C810 004E          1214           LHI   R1,X'4E'
130E  4001 0000          1215  $LCOR1   STH   R0,0(R1)         ZERO MEMORY FROM X'0000'-X'004F'
1312  4001 0080          1216           STH   R0,X'80'(R1)     ZERO MEMORY FROM X'0080'-X'00CF'
1316  2712               1217           SIS   R1,2
1318  2215               1218           BNMS  $LCOR1
                         1219  *
131A                     1220           IFZ   ADC-2
131A  4800 1548          1221           LH    R0,MOD32         SERIES 32 ?
131E  2333               1222           BZS   $LCOR2           BRANCH: NO.
                         1223           ENDC
1320  C800 1304          1224           LHI   R0,$XI32         32-BIT I/O HANDLER
1324  C810 07FE          1225  $LCOR2   LHI   R1,1023*2        FOR MAX I/O SERVICE TABLE
1328  4001 00D0          1226  $LCOR3   STH   R0,X'D0'(R1)     VECTORS TO MEMORY X'00D0'-X'08CE'
132C  2712               1227           SIS   R1,2             ARE ZERO FOR SERIES 16
132E  2213               1228           BNMS  $LCOR3
                         1229  *
1330  C8E0 3000          1230           LHI   R14,X'3000'      ARITH FAULT, MALFUNCTION, ONLY.
1334  C8F0 149E          1231           LDAI  R15,$ERRF2       ILLEGAL INSTRUCTION HANDLER
1338                     1232           IFZ   ADC-2
1338  D0E0 0034          1233           STM   R14,X'34'        FOR SERIES 16
                         1234           ENDC
133C  D0E0 0030          1235           STM   R14,X'30'        FOR SERIES 32
                         1236  *
1340  24E0               1237           LIS   R14,0            TO ZERO MMF BIT IN NEW PSW
1342  C8F0 1438          1238           LDAI  R15,$ERRF3       MACHINE MALFUNCTION NEW LOC
1346                     1239           IFZ   ADC-2
1346  D0E0 003C          1240           STM   R14,X'3C'        FOR SERIES 16
                         1241           ENDC
134A  D0E0 0038          1242           STM   R14,X'38'        FOR SERIES 32
                         1243  *
134E  C8E0 3000          1244           LHI   R14,X'3000'      ARITH FAULT, MALFUNCTION, ONLY.
1352  C8F0 14AE          1245           LDAI  R15,$ERRF1
1356                     1246           IFZ   ADC-2
1356  4800 1548          1247           LH    R0,MOD32
```

```
135A  2133                1248            BNZS  $LCOR3A            BRANCH: PROTECT X'50' SEQUENCE
135C  D0E0 004C           1249            STM   R14,X'4C'          FIXED-POINT DIV FAULT HDLR, S16
      0000 1360           1250  $LCOR3A   EQU   *
                          1251            ENDC
1360  D0E0 0048           1252            STM   R14,X'48'          ARITHMETIC FAULT HDLR, S32
                          1253  *
1364  40E0 009A           1254            STH   R14,X'9A'          SVC NEW PSW
1368  241E                1255            LIS   R1,14
136A  C800 149A           1256            LHI   R0,$ERRF9          SVC INTERRUPT HDLR
136E  4001 009C           1257  $LCOR4    STH   R0,X'9C'(R1)       SVC INTPT NEW LOC'S
1372  2712                1258            SIS   R1,2
1374  2213                1259            BNMS  $LCOR4             DO ALL 16
                          1260  *
1376  C840 4108           1261            LHI   R4,PSWSAVE+X'FF'&X'FF00'+8 PPF REG SAVE AREA
137A                      1262            IFZ   ADC-2
137A  4810 1548           1263            LH    R1,MOD32
137E  213D                1264            BNZS  $LCOR5
                          1265  *
                          1266  * SET UP ADDITIONAL LOW CORE FOR 16-BIT MACHINE ONLY
                          1267  *
1380  4040 0022           1268            STH   R4,X'22'           REG SAVE POINTER
1384  C8F0 14A6           1269            LHI   R15,$ERRF5A
1388  D0E0 002C           1270            STM   R14,X'2C'          S16 FLOAT-POINT INTPT NEW PSW
                          1271  *
138C  C8F0 13C6           1272            LHI   R15,$XI16          S16 I/O HANDLER
1390  D0E0 0044           1273            STM   R14,X'44'          S16 EXTERNAL INTPT NEW PSW
1394  4300 13C0           1274            B     $LCORXIT           RESTORE R14:R15, RETURN (R15)
                          1275  *
                          1276  * SET UP ADDITIONAL LOW CORE FOR 32-BIT MACHINE ONLY
                          1277  *
                          1278            ENDC
1398  24F0                1279  $LCOR5    LIS   R15,0
139A  D0F0 0040           1280            STM   R15,X'40'          ZERO MALFUNCTION STATUS WORD, S320
                          1281  *
139E  4040 0086           1282            STH   R4,X'86'           S32 PPF REG SAVE POINTER
13A2  2748                1283            SIS   R4,8
13A4  4040 0084           1284            STH   R4,X'84'           S32 PPF PSW SAVE POINTER
                          1285  *                                  FOR S3200, IS ONE 24-BIT ADDRESS.
                          1286  *
13A8  C8F0 1496           1287            LDAI  R15,$ERRF8
13AC  D0E0 0088           1288            STM   R14,X'8A'          SYSTEM QUEUE INTPT NEW PSW
                          1289  *
13B0  C8F0 14AA           1290            LDAI  R15,$ERRF5
13B4  D0E0 0090           1291            STM   R14,X'90'          RELOC/PROTECT INTPT NEW PSW
                          1292  *
13B8  C8F0 1492           1293            LDAI  R15,$ERRF7
13BC  D0E0 00C8           1294            STM   R14,X'C8'          DATA FORMAT FAULT NEW PSW
                          1295  *
13C0  01E0 35AC           1296  $LCORXIT  LM    R14,$R14SAV        RESTORE REGISTERS
13C4  030F                1297            BR    R15                AND RETURN.
                          1298  *
                          1324  *
                          1325  * ***********************************************
```

```
                                1326  *  EXTERNAL INTERRUPT HANDLER
 13C6                           1327           IFZ   ADC-2
 13C6   D000 40DC               1328  $XI16    STM   R0,INTSAV          FOR 16-BIT PROCESSOR
 13CA   9F23                    1329           ACKR  R2,R3              ACKNOWLEDGE THE INTERRUPT
 13CC   D1E0 0040               1330           LM    R14,X'40'          OLD PSW, EXTERNAL INTERRUPT
 13D0   24A0                    1331           LIS   R10,0              AVOID $ERRF6 ON SERIES 16
 13D2   2306                    1332           BS    $XI16A
                                1333  *
                                1334  *                                FOR 32-BIT PROCESSOR
 13D4   95AA                    1335  $XI32    EPSR  R10,R10            PSW AFTER INTERRUPT
 13D6   50A0                    1336           DC    X'50A0',Z(INTPSW)  * ST R10,INTPSW
 13D8   1540
 13DA   08E0                    1337           LDAR  R14,R0             OLD PSW
 13DC   08F1                    1338           LDAR  R15,R1             OLD LOC
        0000 13DE               1339  $XI16A   EQU   *
                                1340           ELSE
                                1345           ENDC
 13DE   4020 154A               1346           STH   R2,INTDFV          INTERRUPTING DEVICE ADDRESS
 13E2   D230 154C               1347           STB   R3,INTSTA          INTERRUPTING DEVICE STATUS
 13E6   D0E0 1530               1348           STM   R14,OLDPSW
 13EA   4520 0A2A               1349           CLH   R2,CONRADR         CONSOLE READ-SIDE INTERRUPT ?
 13EA                           1350           IFZ   $KBINT-1
                                1352           ELSE
*13EE   2332                    1353           BE    RETOPSW            IGNORE (FOR 1610,20,30)
                                1354           ENDC
                                1355  *
*13F0   2308                    1356  $XI1     B     $ERRF4             INTERRUPT NOT EXPECTED
                                1357  *
                                1358  * ------------------------------------------
                                1359  * TO RETURN ON OLD PSW FOLLOWING I/O INTERRUPT
                                1360  *
        0000 13F2               1361  RETOPSW  EQU   *
 13F2                           1362           IFZ   ADC-2
 13F2   4800 1548               1363           LH    R0,MOD32           SERIES 32 ?
 13F6   2133                    1364           BNZS  RETOPSW1           BRANCH: NO.
 13F8   D100 40DC               1365           LM    R0,INTSAV          RESTORE USER REGISTER, SERIES 16
                                1366           ENDC
        0000 13FC               1367  RETOPSW1 EQU   *
 13FC   C200 1530               1368           LPSW  OLDPSW
                                1369  * ------------------------------
                                1370  *  EXTERNAL INTERRUPT ERROR ROUTINE
                                1371  *
 1400   2464                    1372  $ERRF4   LIS   R6,4               ERROR TTF4
 1402   2302                    1373           BS    XIERR1
                                1374  *------------------------------
                                1375  * DEVICE INTERRUPTED IN WRONG INTERRUPT LEVEL
                                1376  *
 1404   2466                    1377  $ERRF6   LIS   R6,6               ERROR TTF6
 1406   C660 4630               1378  XIERR1   OHI   R6,C'F0'          CONVERT TO ASCII
 140A   4060 159C               1379           STH   R6,ERRNO
 140E   D3AA 157A               1380           LB    R10,HEXTAB(R10)    CONVERT LEVEL TO ASCII
 1412   D2A0 161C               1381           STB   R10,ERRLVL         AND STORE IN MESSAGE
 1416   4810 0A52               1382           LH    R1,PSW2            SPEC'D AS X'30F0'
```

EXEC - ETPE R05P1

```
141A  9501                1383            EPSR  R0,R1              ENSURE USER REGISTER SET
141C  41F0 0D9C           1384            BAL   R15,ERRALL          'ERROR TTFN', 'DEV DOD STA SS'
                          1385  *                                   'PSW PPPP  LOC LLLL'
1420  4860 159C           1386            LH    R6,ERRNO
1424  C560 4636           1387            CLHI  R6,C'F6'            WRONG INTERRUPT LEVEL ?
1428  2136                 1388            BNES  XIERR2              BRANCH: NO.
142A  4060 155C           1389            STH   R6,ISITERR          FORCE PRINT
142E  41F0 0FBA           1390            BAL   R15,$PRINT
1432  1607                 1391            DAC   INTLVLM             'INTERRUPTED IN LEVEL N'
1434  4300 0A82           1392  XIERR2    B     OPTIN1              ENTER COMMAND MODE.
                          1393  *---------------------------
                          1394  *  SPURIOUS INTERRUPT HANDLERS
                          1395  *
                          1396  *  MACHINE MALFUNCTION INTERRUPT TRAP
                          1397  *
1438  950D                 1398  $FRRF3    EPSR  R13,R13             PSW AT ENTRY TO HANDLER & SAME CC
143A  D1E0 0020           1399            LM    R14,X'20'           S32 MALFUNCTION OLD PSW
143E                      1400            IFZ   ADC-2
143E  4800 1548           1401            LH    R0,MOD32            SERIES 32 ?
1442  233A                 1402            BZS   $MM16.1             BRANCH: NO..
1444  50D0                 1403            DC    X'50D0',Z(MMSW)     * ST R13,MMSW
1446  1544
1448  5800                 1404            DC    X'5800',X'0040'     * L R0,X'40'
144A  0040
*  144C  233D              1405            BZ    $MM.1               BRANCH: NOT S3200
144E  5000                 1406            DC    X'5000',Z(MMSW)     * ST R0,MMSW
1450  1544
*  1452  211D              1407            BM    $MM.2               BRANCH: S3200 POWER FAIL
1454  4300 147C           1408            B     $MM.3               BRANCH: OTHER S3200 MALFUNCTION
                          1409  *
1458  D1E0 0038           1410  $MM16.1   LM    R14,X'38'           S16 MALFUNCTION OLD PSW
145C  40D0 1546           1411            STH   R13,MMSW+2          PSW STATUS AT INTERRUPT
1460  2400                 1412            LIS   R0,0
1462  4000 1544           1413            STH   R0,MMSW             LEADING ZEROS
1466  C3D0 0001           1414  $MM.1     THI   R13,X'0001'         POWER FAIL ?
146A  2339                 1415            BZS   $MM.3               BRANCH: NO.
146C  C800 147C           1416  $MM.2     LHI   R0,$MM.3
1470  4000 003E           1417            STH   R0,X'3E'            CHANGE INTERRUPT NEW LOC
                          1418            ELSE
                          1431            ENDC
1474  4810 0A52           1432            LH    R1,PSW2             SPEC'D AS X'30F0'
1478  9501                 1433            EPSR  R0,R1               RE-ENABLE MALFUNCTION
147A  2200                 1434            BS    *                   AND WAIT FOR POWER RESTORE.
                          1435  *
                          1436  * AT THIS POINT, WE KNOW IT IS NOT A POWER FAIL.
                          1437  * POWER RESTORE REPORTS 'POWER FAIL' AS REASON FOR INTERRUPT.
                          1438  *
147C  C810 1438           1439  $MM.3     LDAI  R1,$ERRF3           RESTORE INTERRUPT VECTOR
1480                      1440            IFZ   ADC-2
1480  4010 003E           1441            STH   R1,X'3E'
1484  4800 1548           1442            LH    R0,MOD32
1488  2333                 1443            BZS   $MM.3A
148A  5010                 1444            DC    X'5010',X'003C'     * ST R1,X'3C'
```

```
148C   003C
       0000 148E          1445  $MM.3A   EQU   *
                          1446           ELSE
                          1448           ENDC
                          1449  *
148E   2463               1450           LIS   R6,3              ERROR TTF3
1490   2304               1451           BS    $BS.COMM
                          1452  * -------------------------
                          1453  * DATA FORMAT FAULT INTERRUPT
                          1454  *
1492   2467               1455  $ERRF7   LIS   R6,7              ERROR TTF7
1494   2302               1456           BS    $BS.COMM
                          1457  * -------------------------
                          1458  * SYSTEM QUEUE SERVICE INTERRUPT
                          1459  *
1496   2468               1460  $ERRF8   LIS   R6,8              ERROR TTF8
1498   230E               1461  $BS.COMM BS    COMM
                          1462  * -------------------------
                          1463  * SUPERVISOR CALL INTERRUPT
                          1464  *
149A   2469               1465  $ERRF9   LIS   R6,9              ERROR TTF9
149C   230C               1466           BS    COMM
                          1467  * -------------------------
                          1468  *  ILLEGAL INSTRUCTION INTERRUPT TRAP
                          1469  *
149E   2462               1470  $ERRF2   LIS   R6,2              ERROR TTF2
14A0                      1471           IFZ   ADC-2
14A0   C820 0030          1472           LHI   R2,X'30'          WHERE TO FIND OLD PSW, SERIES 16
                          1473           ENDC
14A4   2308               1474           BS    COMM
                          1475  * -------------------------
14A6                      1476           IFZ   ADC-2
                          1477  *  FLOATING-PT ARITH FAULT INT TRAP  (16 BIT PROCESSOR)
                          1478  *
14A6   C820 0028          1479  $ERRF5A  LHI   R2,X'28'          WHERE TO FIND OLD PSW, SERIES 16
                          1480  * -------------------------
                          1481           ENDC
                          1482  *  RELOCATION/PROTECTION INT TRAP
                          1483  *
14AA   2465               1484  $ERRF5   LIS   R6,5              ERROR TTF5
14AC   2304               1485           BS    COMM
                          1486  * -------------------------
                          1487  *  ARITHMETIC FAULT INT (32-BIT PROCESSOR) TRAP
14AE                      1488           IFZ   ADC-2
                          1489  *  FIXED-PT DIVIDE FAULT INT (16-BIT PROCESSOR) TRAP
                          1490           ENDC
                          1491  *
14AE   2461               1492  $ERRF1   LIS   R6,1              ERROR TTF1
14B0                      1493           IFZ   ADC-2
14B0   C820 0048          1494           LHI   R2,X'48'          WHERE TO FIND OLD PSW, SERIES 16
                          1495           ENDC
                          1496  *
                          1497  * ERROR TTFN PRINTOUT ROUTINE. EXPECTS USER REGISTER SET SELECTED.
```

```
                                1498  *
        0000 1484               1499  COMM     EQU   *
1484                            1500           IFZ   ADC-2
1484    4800 1548               1501           LH    R0,MOD32          SERIES 16 ?
1488    2136                    1502           BNZS  $COMM1            BRANCH: NO.
148A    C560 0003               1503           CLHI  R6,3              FROM $ERRF3 ?
148E    2333                    1504           BES   $COMM1            BRANCH: YES.
14C0    D1E2 0000               1505           LM    R14,0(R2)         GET INTERRUPT OLD PSW
                                1506           ENDC
14C4    D0E0 1530               1507  $COMM1   STM   R14,OLDPSW        OLD PSW, OLD LOC
14C8    C560 0002               1508           CLHI  R6,2              ILLEGAL INSTRUCTION?          **
14CC    2135                    1509           BNES  $COMM1A           SKIP IF NO                    **
14CE    C5F0 0A5C               1510           CLAI  R15,$CON          ON BREAKPOINT?                **
14D2    4330 0CE8               1511           BE    HALT9A            YES, HALT PROCESSOR           **
14D6    C660 4630               1512  $COMM1A  OHI   R6,C'F0'          CONVERT ERROR NUMBER TO ASCII
14DA    4060 159C               1513           STH   R6,ERRNO          ERROR NUMBER
14DE    4060 155C               1514           STH   R6,ISITERR        FORCE ERROR MESSAGE PRINT
14E2    4810 0A52               1515           LH    R1,PSW2           SPEC'D AS X'30F0'
14E6    9501                    1516           EPSR  R0,R1             ENSURE USER REGISTER SET
14E8    41E0 1152               1517  $COMM2   BAL   R14,STCON         SET UP & SELECT KEYBOARD DEVICE
14EC    41F0 1274               1518           BAL   R15,TSTOU         TEST IF KEYBOARD OFF-LINE
14F0    2034                    1519           BNZS  $COMM2            WAIT FOR ON-LINE.
14F2    41F0 0FB0               1520           BAL   R15,CRLF          SEND LINE FEED
14F6    41F0 0D5C               1521           BAL   R15,ERR           PRINT 'ERROR XXFN'
14FA    4860 159C               1522           LH    R6,ERRNO          GET ERROR NUMBER
14FE    4060 155C               1523           STH   R6,ISITERR        FORCE PRINT
1502    41E0 0E50               1524           BAL   R14,ERRPL1        PRINT 'PSW PPPP  LOC LLLL'
1506    C560 4633               1525           CLHI  R6,C'F3'          MACHINE MALFUNCTION ?
150A    4230 0AB2               1526           BNE   OPTIN1            BRANCH: NO.
                                1527  *
150E                            1528           IFZ   ADC-2
150E    4810 1546               1529           LH    R1,MMSW+2         ASSUME SERIES 16
1512    4800 1548               1530           LH    R0,MOD32          SERIES 32 ?
1516    2333                    1531           BZS   $COMM3            BRANCH: NO.
1518    5810                    1532           DC    X'5810',Z(MMSW)   * L R1,MMSW
151A    1544
        0000 151C               1533  $COMM3   EQU   *
                                1534           ELSE
                                1536           ENDC
151C    2408                    1537           LIS   R0,8              DIGIT COUNT
151E    C820 15E3               1538           LDAI  R2,ASCIMSW        DESTINATION
1522    41F0 0F18               1539           BAL   R15,HEXASC        CONVERT 3200 MMSW FOR PRINT
1526    41F0 0FBA               1540           BAL   R15,$PRINT
152A    15DA                    1541           DAC   MMSWMSG           'STATUS = XXXXXXXX'
152C    4300 0AAA               1542           B     OPTIN             GET COMMAND INPUT
                                1543  * ****************************
                                1544  *  ETPE CONSTANTS & TABLES
1530                            1545           ALIGN 8
                                1546  *---------------------
1530    0000                    1547  OLDPSW   DCX   0000,0000,0000,0000
1532    0000
1534    0000
1536    0000
```

```
1538  0000                1548  NEWPSW   DCX   0000,0000,0000,0000
153A  0000
153C  0000
153E  0000
1540  0000 0000          1549  INTPSW   DCY   0                    (SERIES 32 ONLY)
1544  0000                1550  MMSW     DCX   0000,0000           MACHINE MALFUNCTION STATUS
1546  0000
                          1551  *------------------------
1548  0000                1552  MOD32    DCX   0                    NON-ZERO, SERIES 32
154A  0000                1553  INTDEV   DCX   0                    INTERRUPTING DEV ADR
      0000 154A           1554  ERRDEV   EQU   INTDEV              ERROR DEVICE #
154C  00                  1555  INTSTA   DB    0                    INTERRUPTING DEV STATUS
      0000 154C           1556  ERRSTA   EQU   INTSTA              ERRONEOUS STATUS
154D  80                  1557  NORM     DB    X'80'                CONSOLE NORMAL MODE
154E  40                  1558  INCR     DB    X'40'                CONSOLE INCREMENTAL MODE
154F  E0                  1559  $CLKSTRT DB    X'E0'                PIC CMD DISARM+START
1550                      1560           DB    *                    (ALIGN ON HW BOUNDARY)
1550  0000                1561  SINK     DC    0                    BIT BUCKET
1552  0000                1562  $CONPAS  DCX   0                    SET WHEN CONSOLE ON PASLA/PALM
1554  0000                1563  $LSTPAS  DCX   0                    SET WHEN LIST DEVICE ON PASLA
                          1564  * --------------------
1556  0000                1568  BRKVECT  DC    Z(0)                 BREAK KEY VECTOR
1558  0000                1569  $BRKFLG  DCX   0                    SET IF BREAK KEY DETECTED
155A  0000                1570  IOSAVE   DCX   0                    CURRENT I/O IDENTIFIERS
155C  0000                1571  ISITERR  DCX   0                    MESSAGE LEVEL
155E  0000                1572  NOERR    DCX   0                    ZERO = 'NO ERROR'
1560  0000                1573  SELTST   DCX   0                    HIGHEST SELECTED TEST #
1562  0000                1574  $LINEPOS DCX   0                    CURRENT $OUTBUF POSITION
1564  0000                1575  $PRTFLG  DCX   0                    FLAG USED FOR DEFERRING BRK ACKNOWLED
1566  0000                1576  $WASDU   DCX   0                    ZERO IF I/O DEVICE ON-LINE
1568  0000                1577  TOTAL    DCX   0                    TIMES WHOLE TEST RAN
156A  0000                1578  TOTERR   DCX   0                    TOTAL ERRORS DETECTED
156C  0000                1579  BTESTNO  DCX   0                    CURRENT TEST # IN BINARY
156E  0000                1580  COUNT    DCX   0                    TIMES CURRENT TEST RAN
1570  0000                1581  $PAUSE   DCX   0                    SET DURING TRANSMISSION PAUSE
1572  0000                1582  $SHUTDWN DAC   0                    A(USER-DEFINED SHUTDOWN ROUTINE)
1574  0000                1583  OUT.SAV  DAC   0                    OUTCHR RETURN ADDRESS SAVE
1576  0000                1584  BRK.SAV  DAC   0                    TSTBRK RETURN ADDRESS SAVE
1578  0000                1585  SET.RTN  DAC   0                    $SETUP RETURN ADDRESS SAVE
                          1586  *
157A  3031 3233 3435 3637 1590  HEXTAB   DB    C'0123456789ABCDEF'  HEXADECIMAL DIGITS
1582  3839 4142 4344 4546
                          1591  *--------------------------------
                          1592  *  ETPE MESSAGES
                          1593  *
158A  5445 5354 2020 2A2A 1594  TSTMSG   DB    C'TEST  **',X'0D'
1592  0D
      0000 1590           1595  MTESTNO  EQU   TSTMSG+6            MASTER TEST NUMBER (ASCII)
1594                      1596           ALIGN 2
1594  4552 524F 5220 2A2A 1597  ERRMSG   DB    C'ERROR ****  ',13,10,0
159C  2A2A 2020 0D0A 00
      0000 159A           1598  ETESTNO  EQU   ERRMSG+6            STORED BY ETPE
      0000 159C           1599  ERRNO    EQU   ERRMSG+8            STORE ERRNO AS CHAR CONSTANT
```

```
15A3  544F 5441 4C20 2020  1600  TOTMSG   DB    C'TOTAL   TOTERR',13,10,0
15AB  544F 5445 5252 0D0A
15B3  00
15B4  4E4F 2045 5252 4F52  1601  NOERMSG  DB    C'NO ERROR',13,10,0
15BC  0D0A 00
15BF  4445 5620 2A2A 2A20  1602  DEVMSG   DB    C'DEV *** STA **',13,10,0
15C7  5354 4120 2A2A 0D0A
15CF  00
      0000 15C3            1603  ASCIDEV  EQU   DEVMSG+4
      0000 15C7            1604  STAMSG   EQU   DEVMSG+8
      0000 15CB            1605  ASCISTA  EQU   DEVMSG+12
15D0  4445 5620 2A2A 2A0D  1606  DEVMSG2  DB    C'DEV ***',13,10,0
15D8  0A00
      0000 15D4            1607  ASCIDEV2 EQU   DEVMSG2+4
15DA  5354 4154 5553 203D  1608  MMSWMSG  DB    C'STATUS =          ',13,10,0
15E2  2020 2020 2020 2020
15EA  200D 0A00
      0000 15E3            1609  ASCIMSW  EQU   MMSWMSG+9
15EE  5053 5720 2020 2020  1610  PSWMSG   DB    C'PSW         LOC       ',13,10,0
15F6  2020 2020 4C4F 4320
15FE  2020 2020 2020 0D0A
1606  00
      0000 15F2            1611  ASCIPSW  EQU   PSWMSG+4
      0000 15FA            1612  LOCMSG   EQU   PSWMSG+12
      0000 15FE            1613  ASCILOC  EQU   PSWMSG+16
1607  494E 5445 5252 5550  1614  INTLVLM  DB    C'INTERRUPTED IN LEVEL *',13,10,0
160F  5445 4420 494E 204C
1617  4556 454C 202A 0D0A
161F  00
      0000 161C            1615  ERRLVL   EQU   INTLVLM+21
1620  454E 4420 4F46 2054  1616  EOTMSG   DB    C'END OF TEST',13,10,0
1628  4553 540D 0A00
1630                       1617             ALIGN 4
1630  0D0A 3F0D 0A00       1618  QMSG     DB    X'0D',X'0A',C'?',13,10,0 CR,LF,?,CR
1636  2A20 00              1619  AMSG     DB    C'*',X'20',0
1639  FFFF FFFF FFFF FFFF  1620  BRKMSG   DB    -1,-1,-1,-1,-1,-1,-1,-1
1641  4252 4541 4B20 5445  1621             DB    C'BREAK TERMINATION',13,10,0
1649  524D 494E 4154 494F
1651  4E0D 0A00
      0000 1654            1622  $BRKEND  EQU   *-1
1655  FFFF FFFF 0D0A 00    1623  NULLMSG  DB    -1,-1,-1,-1,13,10,0
      0000 1659            1624  CRLFMSG  EQU   NULLMSG+4
165C                       1625             DB    *                    HALFWORD ALIGN
```

DATA CONSTANTS & CHECK ROUTINES

```
                                1627  *-----------------------------
                                1628  *  OPTION/COMMAND TABLE
                                1629  *  STRUCTURE DEFINED BY '$STRUC1' AT TOP OF LISTING
165C                            1630          ALIGN 4
                                1631  *
        0000 165C               1632  OPT     EQU   *
165C    494E 4445 5620          1633  INDEV   DC    C'INDEV ',Z(ADR,XXX),X'0085',X'0000'
1662    0B86
1664    0085
1666    0000
1668    5345 4C43 4831          1634  SELCH1  DC    C'SELCH1',Z(ADR),X'0000',X'0000'
166E    0B7E
1670    0000
1672    0000
1674    4F55 5444 4556          1635  OUTDEV  DC    C'OUTDEV',Z(ADR,XXX),X'0000',X'0000'
167A    0B86
167C    0000
167E    0000
1680    5345 4C43 4832          1636  SELCH2  DC    C'SELCH2',Z(ADR),X'0000',X'0000'
1686    0B7E
1688    0000
168A    0000
168C    4944 5249 5645          1637  IDRIVE  DC    C'IDRIVE',Z(LEVEL),X'0000',X'0000'
1692    0BCE
1694    0000
1696    0000
1698    4F44 5249 5645          1638  ODRIVE  DC    C'ODRIVE',Z(LEVEL),X'0000',X'0000'
169E    0BCE
16A0    0000
16A2    0000
16A4    564F 4C55 4D45          1639  VOLUME  DC    C'VOLUME',Z(LEVEL),X'0001',X'0000'
16AA    0BCE
16AC    0001
16AE    0000
        0000 1680               1640  OPTEND2 EQU   *                   END OF PRINTING OPTIONS
1680    4E4F 4D53 4720          1641  NOMSG   DC    C'NOMSG ',Z(LEVEL),X'0000',X'0000'
1686    0BCE
1688    0000
168A    0000
168C    4453 4352 5054          1642  DSCRPT  DC    C'DSCRPT',Z(BIGVALUE),C'MM',C'D '
16C2    0BD6
16C4    4D4D
16C6    4420
16C8    4F53 4944 2020          1643  OSID    DC    C'OSID  ',Z(BIGVALUE),C'MD',C'L2'
16CE    0BD6
16D0    4D44
16D2    4C32
        0000 16D4               1644  OPTEND  EQU   *                   END OF OPTIONS WITH VALUES
16D4    4F50 5449 4F4E          1645  OPTION  DC    C'OPTION',X'0000',X'0000',X'0000'
16DA    0000
16DC    0000
16DE    0000
        0000 16DE               1646  $LINCNT EQU   OPTION+$VALU2       PRINTOUT LINE COUNTER
```

DATA CONSTANTS & CHECK ROUTINES

```
    16E0   434F 4E20 2020        1647  CON       DC    C'CON   ',X'0000',X'0000',X'0000'
    16E6   0000
    16E8   0000
    16EA   0000
    16EC   4255 494C 4420        1648  BUILD     DC    C'BUILD ',Z(LEVEL),X'0000',X'0000'
    16F2   0BCE
    16F4   0000
    16F6   0000
    16F8   5645 5249 4659        1649  VERIFY    DC    C'VERIFY',Z(LEVEL),X'0000',X'0000'
    16FE   0BCE
    1700   0000
    1702   0000
    1704   4255 494C 4456        1650  BUILDV    DC    C'BUILDV',Z(LEVEL),X'0000',X'0000'
    170A   0BCE
    170C   0000
    170E   0000
    1710   5345 5155 2020        1651  SEQUENCE  DC    C'SEQU  ',Z(ADR),X'0000',X'0000'
    1716   0B7E
    1718   0000
    171A   0000
    171C   4E55 4D42 4552        1652  NUMBER    DC    C'NUMBER',Z(06NUM),X'0000',X'0000'
    1722   0000F
**  U002  **     06NUM
    1724   0000
    1726   0000
    1728   4E41 4D45 2020        1653  NAME      DC    C'NAME  ',Z(PNAME),X'0000',X'0000'
    172E   0000F
**  U002  **     PNAME
    1730   0000
    1732   0000
    1734   464C 4147 2020        1654  FLAG      DC    C'FLAG  ',Z(LEVEL),X'0000',X'0000'
    173A   0BCE
    173C   0000
    173E   0000
    1740   5048 4720 2020        1655  PACKAGE   DC    C'PKG   ',Z(ADR),X'0000',X'0000'
    1746   0B7E
    1748   0000
    174A   0000
    174C   5044 4220 2020        1656  PDBCMD    DC    C'PDB   ',Z(PDB.CMD),X'0000',X'0000'
    1752   0000F
**  U002  **     PDB.CMD
    1754   0000
    1756   0000
    1758   4C49 4D49 5453        1657  LIMITS    DC    C'LIMITS',Z(PLIMITS),X'0000',X'0000'
    175E   0000F
**  U002  **     PLIMITS
    1760   0000
    1762   0000
    1764   5752 4954 4520        1658  WRITE     DC    C'WRITE ',Z(WRT.CMD),X'0000',X'0000'
    176A   0000F
**  U002  **     WRT.CMD
    176C   0000
    176E   0000
```

DATA CONSTANTS & CHECK ROUTINES

```
1770  FFFF                           DCX   FFFF               END OF TABLE
                            *
1772  FFFF                 DEVSADR   DCX   FFFF               INTERRUPTING DEVICE TABLE
1774  FFFF                 DEVINT    DCX   FFFF               INTERRUPTING DEVICES
1776  0000                 INTLVL    DCX   0000
1778  8000                 DEFTESTS  DCX   8000,0000
177A  0000
177C  1B8C                 TESTS     DAC   RUN.CMD
177E  0001                 MAXTST    DCX   1
                            *
1780  434F 4D4D 4F4E 204D  TITLE     DC    C'COMMON MMD CROSS GENERATOR 06-252R04        '
1788  4D44 2043 524F 5353
1790  2047 454E 4552 4154
1798  4F52 2030 362D 3235
17A0  3252 3034 2020 2020
17A8  2020 2020
17AC  5052 454C 494D                 DC    C'PRELIM'
17B2  0D0A                           DC    X'0D0A',X'0000'
17B4  0000
                            *
                            *
17B6  240D                 @BUILDV   LIS   R0,X'0D'           NON-ZERO (CARRIAGE RETURN)
17B8  4000 1A16                      STH   R0,NOSTOP          SET AUTO VERIFY FLAG
17BC  2400                           LIS   R0,0               ZERO FOR BUILD PASS
17BE  2306                           BS    @BV                TO COMMON SEQUENCE
17C0  2400                 @BUILD    LIS   R0,0               ZERO TO BUILD
17C2  4000 1A16                      STH   R0,NOSTOP          KILL AUTO-VERIFY FLAG
17C6  2302                           BS    @BV
17C8  2401                 @VERIFY   LIS   R0,1               ONE TO VERIFY
17CA  4000 1A18            @BV       STH   R0,VERIFLAG        SET VERIFY FLAG
17CE  C540 0020                      CLHI  R4,X'20'           SPACE FOLLOWS COMMAND?
17D2  4230 0CA6                      BNE   $RUNIT             DO CARRIAGE RETURN CHECK IF NO
17D6  41E0 0EBA                      BAL   R14,OPTVAL         GO GET VOLUME INFORMATION
17DA  48E1 0006                      LH    R14,$CKROUT(R1)
17DE  2332                           BZS   @BV1
17E0  01FE                           BALR  R15,R14            PROCESS OPTION
17E2  4061 0008            @BV1      STH   R6,$VALU1(R1)      STORE VALUE
17E6  4060 16AC                      STH   R6,VOLUME+$VALU1   SET VOLUME OPTION TOO
17EA  4300 0CA6                      B     $RUNIT             GO DO CR CHECK
                            *
                            *
                            *
                            *     INFLAG  -  DISKTYP1          OUTFLAG  -  DISKTYP2
                            *       1                             1                  MAGTAPE
                            *       2                             2                  FLOPPY
                            *       3          3                  3          3       10 MB
                            *       4          4                  4          4       80 MB
                            *       5          6                  5          6       300 MB
                            *
                            *     VERIFY INDEV AND OUTDEV
                            *
17EE  40F0 1A6E            INIT      STH   R15,R15SAVE
17F2  C8F0 2A2A                      LHI   R15,C'**'
```

DATA CONSTANTS & CHECK ROUTINES

```
   17F6  40F0 159C        1705           STH   R15,ERRNO          NO TEST NUMBERS
   17FA  24F0             1706           LIS   R15,0              DEVICE CODE PRESET
   17FC  40F0 16DE        1707           STH   R15,$LINCNT        CLEAR LISTING LINE COUNT
   1800  40F0 1A70        1708           STH   R15,FIRSTHDI       FIRST HEAD ON INPUT DISK          R04
   1804  40F0 1A72        1709           STH   R15,FIRSTHDO       FIRST HEAD ON OUTPUT DISK         R04
   1808  4820 1664        1710           LH    R2,INDEV+$VALU1    INPUT DEVICE ADDRESS
   180C  4830 1666        1711           LH    R3,INDEV+$VALU2    POSSIBLE CONTROLLER ADDRESS
   1810  4840 1670        1712           LH    R4,SELCH1+$VALU1   POSSIBLE SELCH ADDRESS
   1814  41E0 199E        1713           BAL   R14,FMDCHK
   1818  4330 187C        1714           BE    INIT003F           ACCEPT FLOPPY AS INPUT AND OUTPUT
   181C  41E0 18C2        1715  INIT0001 BAL   R14,MTCHCK         TEST IF MAG TAPE
*  1820  2338              1716           BE    INIT0002           ACCEPTABLE MAG TAPE SPEC
   1822  41E0 18DE        1717           BAL   R14,DISKCHK        TEST IF DISK
*  1826  2335              1718           BE    INIT0002           ACCEPTABLE DISK SPEC
   1828  C850 3478        1719           LDAI  R5,SPECERR2        'INVALID INDEV SPECIFICATION'
   182C  4300 1C50        1720           B     CONMSG1
   1830  40F0 1A4E        1721  INIT0002 STH   R15,INFLAG         SAVE INPUT DEVICE CODE
   1834  C4F0 0007        1722           NHI   R15,7
   1838  D30F 1AE0        1723           LB    R0,DTAB(R15)
   183C  4000 1A6A        1724           STH   R0,DISKTYP1
   1840  C500 0008        1725           CLHI  R0,8               CDD DISK?                         R04
*  1844  2188              1726           BL    INITOUT            SKIP IF NO                        R04
   1846  C500 000C        1727           CLHI  R0,12              *                                 R04
*  184A  2385              1728           BNL   INITOUT            *                                 R04
   184C  C800 0010        1729           LHI   R0,X'10'           ON CDD REMOVABLE,                 R04
   1850  4000 1A70        1730           STH   R0,FIRSTHDI        FIRST HEAD IS HEAD '10'           R04
   1854  24F0             1731  INITOUT  LIS   R15,0              CLEAR DEVICE CODE
   1856  4820 167C        1732           LH    R2,OUTDEV+$VALU1   OUTPUT DEVICE ADDRESS
   185A  4830 167E        1733           LH    R3,OUTDEV+$VALU2   POSSIBLE CONTROLLER ADDRESS
   185E  4840 1688        1734           LH    R4,SELCH2+$VALU1   POSSIBLE SELCH
   1862  41E0 18C2        1735           BAL   R14,MTCHCK         TEST IF MAG TAPE
*  1866  233F              1736           BE    INIT0003           ACCEPT MAG TAPE OUTPUT
   1868  41E0 199E        1737           BAL   R14,FMDCHK         TEST IF FLOPPY
*  186C  233C              1738           BE    INIT0003           ACCEPT FLOPPY OUTPUT
   186E  41E0 18DE        1739           BAL   R14,DISKCHK        TEST IF DISK OUTPUT
*  1872  2339              1740           BE    INIT0003           SKIP IF YES
   1874  C850 33BC        1741           LDAI  R5,SPECERR         'INVALID OUTDEV SPECIFICATION'
   1878  4300 1C50        1742           B     CONMSG1
   187C  40F0 1A4E        1743  INIT003F STH   R15,INFLAG         STORE INPUT DEVICE CODE
   1880  4020 167C        1744           STH   R2,OUTDEV+$VALU1   FORCE OUTDEV=INDEV
   1884  40F0 1A50        1745  INIT0003 STH   R15,OUTFLAG        SAVE OUTPUT DEVICE CODE
   1888  D30F 1AE0        1746           LB    R0,DTAB(R15)
   188C  4000 1A6C        1747           STH   R0,DISKTYP2        OUTPUT DEVICE DISK TYPE
   1890  C500 0008        1748           CLHI  R0,8               CDD REMOVABLE?                    R04
*  1894  2188              1749           BL    INIT0004           SKIP IF NO                        R04
   1896  C500 000C        1750           CLHI  R0,12              *                                 R04
*  189A  2385              1751           BNL   INIT0004           *                                 R04
   189C  C800 0010        1752           LHI   R0,X'10'           *                                 R04
   18A0  4000 1A72        1753           STH   R0,FIRSTHDO        *                                 R04
   18A4  4800 16A0        1754  INIT0004 LH    R0,ODRIVE+$VALU1   OUTPUT DRIVE NUMBER
   18A8  C700 0001        1755           XHI   R0,1               FLIP TO GET INPUT DRIVE NUMBER
   18AC  4000 1694        1756           STH   R0,IDRIVE+$VALU1   STORE IT
   18B0  48F0 1A6E        1757           LH    R15,R15SAVE
```

DATA CONSTANTS & CHECK ROUTINES

```
18B4  4520 1664          1758            CLH   R2,INDEV+$VALU1    INDEV & OUTDEV CANNOT BE SAME   R04
18B8  023F               1759            BNER  R15                *                               R04
18BA  C850 3463          1760            LDAI  R5,SPECERR1        'INDEV/OUTDEV CONFLICT'
18BE  4300 1C50          1761            B     CONMSG1
                         1762  *
                         1763  *
18C2  0802               1764  MTCHCK    LDAR  R0,R2              COPY DEVICE NUMBER
18C4  C400 008E          1765            NHI   R0,X'8E'           ACCEPT 84, 94,,,D4,E4,F4
18C8  C500 0084          1766            CLHI  R0,X'84'           ACCEPT 85, 95,,,D5,E5,F5
18CC  023E               1767            BNER  R14                NOT A MAG TAPE
18CE  0802               1768            LDAR  R0,R2              COPY DEVICE NUMBER
18D0  DD00 154C          1769            SS    R0,ERRSTA
18D4  4210 19C4          1770            BTC   1,BADSTA           LEAVE IF DU
18D8  24F1               1771            LIS   R15,1              DEVICE CODE 1 FOR MAG TAPE
18DA  0833               1772            LDAR  R3,R3              DON'T ACCEPT IF CONTROLLER
18DC  030E               1773            BR    R14                ADDRESS SPECIFIED
                         1774  *
18DE  0802               1775  DISKCHK   LDAR  R0,R2              COPY DEVICE NUMBER
18E0  24F3               1776            LIS   R15,3              POSSIBLE DEVICE CODE FOR DISK
18E2  0523               1777  DISKCHK1  CLAR  R2,R3
18E4  2333               1778            BES   DISKCHK2           TROUBLE IF DRIVE=CONTROLLER
18E6  0534               1779            CLAR  R3,R4
18E8  2133               1780            BNES  DISKCHK3           TROUBLE IF CONTROLLER = SELCH
18EA  08EE               1781  DISKCHK2  LDAR  R14,R14
18EC  030E               1782            BR    R14                LEAVE: CC NOT ZERO
18EE  0803               1783  DISKCHK3  LDAR  R0,R3              HAS TO BE A CONTROLLER
18F0  2233               1784            BZS   DISKCHK2           IF NO, CAN'T BE A DISK
18F2  DE00 1A35          1785            OC    R0,DRESET          CONTROLLER RESET
18F6  DD00 154C          1786            SS    R0,ERRSTA
18FA  4240 19C4          1787            BTC   4,BADSTA           EXIT IF BAD STATUS
18FE  0804               1788            LDAR  R0,R4              HAS TO BE A SELCH
1900  223B               1789            BZS   DISKCHK2           NOT A DISK IF NO SELCH
1902  DE00 1A30          1790            OC    R0,STOPS           STOP SELCH
1906  DD00 154C          1791            SS    R0,ERRSTA
190A  4240 19C4          1792            BTC   4,BADSTA           BAD STATUS EXIT
190E  2450               1793            LIS   R5,0
1910  DE20 1A38          1794            OC    R2,RESTOR          DRIVE RESTORE
1914  0865               1795  RPS       LDAR  R6,R5              LAST RPS
1916  9B25               1796            RDR   R2,R5              THIS RPS
1918  0556               1797            CLAR  R5,R6              COMPARE
191A  2283               1798            BNLS  RPS                LOOP UNTIL SEE SMALLER RPS
191C  C560 0019          1799            CLHI  R6,25              CHECK SIZE OF LAST RPS
1920  2383               1800            BNLS  MBBIG              NOT A 10 MB DISK
                         1801  *
1922  2400               1802  DISKCHK4  LIS   R0,0               DEVICE CODE IN R15 = 3
1924  030E               1803            BR    R14                CLEAR CONDITION CODE
1926  24F4               1804  MBBIG     LIS   R15,4              DEVICE CODES 4 THROUGH 9
                         1805  *                                  R15 = 4 FOR 80 MB DISK
                         1806  *                                  R15 = 5 FOR 300 MB DISK
                         1807  *                                  R15 = 6 FOR 16 MB CDD REMOVABLE
                         1808  *                                  R15 = 7 FOR 16 MB CDD FIXED
                         1809  *                                  R15 = 8 FOR 48 MB CDD FIXED
                         1810  *                                  R15 = 9 FOR 80 MB CDD FIXED
```

DATA CONSTANTS & CHECK ROUTINES

```
1928  2480           1811          LIS   R8,0            CYLINDER
192A  2491           1812          LIS   R9,1            HEAD
192C  24A0           1813          LIS   R10,0           SECTOR
192E  41D0 284C      1814          BAL   R13,DISKWAIT
1932  41D0 2820      1815          BAL   R13,FILESET     SET UP FILE TO SEEK
1936  DE20 1A39      1816          OC    R2,SEEK         DO A SEEK TO HEAD 1
193A  9D35           1817          SSR   R3,R5
193C  2221           1818          BFBS  2,1             CONTROLLER IDLE
193E  9A3A           1819          WDR   R3,R10          WRITE SECTOR NUMBER TO CONTROLLER
1940  0819           1820          LDAR  R1,R9           HEAD NUMBER
1942  911A           1821          SLLS  R1,10           POSITION BITS
1944  0618           1822          OAR   R1,R8           COMBINE WITH CYLINDER NUMBER
1946  9831           1823          WHR   R3,R1           SEND TO CONTROLLER
1948  DE30 1A34      1824          OC    R3,RCHECK       DO A READCHECK
194C  9D35           1825          SSR   R3,R5           WAIT FOR CONTROLLER IDLE
194E  2221           1826          BFBS  2,1
1950  C350 0060      1827          THI   R5,X'60'        SEEK INC OR ILL ADR?
*1954 2337           1828          BZ    DISKCHK5        BRANCH IF NO, NOT CDD
1956  DE20 0000F     1829          OC    R2,MSMCLFT      CLEAR FAULT STATUS BITS
**0002 **    MSMCLFT
                     1830  *
195A  9D35           1831          SSR   R3,R5           CONTROLLER IDLE
195C  2221           1832          BFBS  2,1
195E  4300 1922      1833          B     DISKCHK4        RETURN
1962  2480           1834  DISKCHK5 LIS  R8,0            CYLINDER
1964  2498           1835          LIS   R9,8            HEAD
1966  24A0           1836          LIS   R10,0           SECTOR
1968  41D0 284C      1837          BAL   R13,DISKWAIT
196C  41D0 2820      1838          BAL   R13,FILESET     SET UP FILE TO SEEK
1970  DE20 1A39      1839          OC    R2,SEEK         DO A SEEK TO HEAD 8
1974  9D35           1840          SSR   R3,R5
1976  2221           1841          BFBS  2,1             CONTROLLER IDLE
1978  9A3A           1842          WDR   R3,R10          WRITE SECTOR NUMBER TO CONTROLLER
197A  0819           1843          LDAR  R1,R9           HEAD NUMBER
197C  911A           1844          SLLS  R1,10           POSITION BITS
197E  0618           1845          OAR   R1,R8           COMBINE WITH CYLINDER NUMBER
1980  9831           1846          WHR   R3,R1           SEND TO CONTROLLER
1982  DE30 1A34      1847          OC    R3,RCHECK       DO A READCHECK
1986  9D35           1848          SSR   R3,R5           WAIT FOR CONTROLLER IDLE
1988  2221           1849          BFBS  2,1
198A  C350 0060      1850          THI   R5,X'60'        SEEK INC OR ILL ADR?
198E  2132           1851          BNZS  DISKCHK6        BRANCH IF YES (80 MB)
1990  26F1           1852          AIS   R15,1           CODE 5 - 300 MB DISK
1992  DE20 1A35      1853  DISKCHK6 OC   R2,DRESET
1996  9D35           1854          SSR   R3,R5           CONTROLLER IDLE
1998  2221           1855          BFBS  2,1
199A  4300 1922      1856          B     DISKCHK4        RETURN
                     1857  *
199E  0833           1858  FMDCHK   LDAR  R3,R3          LEAVE IF CONTROLLER SPECIFIED
19A0  023E           1859          BNZR  R14
19A2  0802           1860          LDAR  R0,R2           COPY DEVICE NUMBER
19A4  C400 00CF      1861          NHI   R0,X'CF'        ACCEPT C1,D1,E1,F1
19A8  C500 00C1      1862          CLHI  R0,X'C1'        CHECK IF FLOPPY
```

PREVIOUS ERROR ON PAGE 30

DATA CONSTANTS & CHECK ROUTINES

```
19AC   023E             1863           BNER   R14              NO,RETURN
19AE   2407             1864           LIS    R0,7
19B0   9E20             1865  FMDCHK1  OCR    R2,R0            ISSUE STOP COMMAND
19B2   9D20             1866           SSR    R2,R0
19B4   2221             1867           BFBS   2,1              WAIT FOR CONTROLLER IDLE
19B6   2408             1868           LIS    R0,8
19B8   9E20             1869           OCR    R2,R0            ISSUE RESET COMMAND
19BA   9D20             1870           SSR    R2,R0
19BC   2221             1871           BFBS   2,1              WAIT FOR CONTROLLER IDLE
19BE   24F2             1872           LIS    R15,2            DEVICE CODE FOR FLOPPY
19C0   2400             1873           LIS    R0,0             CLEAR CONDITION CODE
19C2   030E             1874           BR     R14
                        1875  *
                        1876  *
19C4   4000 154A        1877  BADSTA   STH    R0,ERRDEV        SAVE DEVICE NUMBER
19C8   41F0 0D7E        1878           BAL    R15,ERR0S        ERRONEOUS STATUS
19CC   4300 0AAA        1879           B      OPTIN
```

O.S. FORMAT EQUATES

```
                    1881  * EQUATES DEFINING OFFSETS WITHIN THE VOLUME DESCRIPTOR
                    1882  *
0000 0000           1883  VD.VOL    EQU   0                   VOLUME NAME
0000 0004           1884  VD.ATRB   EQU   4                   ATTRIBUTE
0000 0008           1885  VD.FDB    EQU   8                   DIRECTORY LOGICAL BLOCK ADDRESS
0000 000C           1886  VD.OSP    EQU   12                  O.S. POINTER
0000 0010           1887  VD.OSS    EQU   16                  O.S.SIZE
0000 0014           1888  BM.LBA    EQU   20                  BIT MAM LBA
0000 0018           1889  ILS.LBA   EQU   24                  INVERTED LIST LBA
0000 001C           1890  SDR.LBA   EQU   28                  SECONDARY DIRECTORY LBA
                    1891  *
                    1892  * EQUATES DEFINING OFFSETS WITHIN THE DIRECTORY
                    1893  *
0000 0000           1894  FNAME     EQU   0                   FILE NAME
0000 0008           1895  EXT       EQU   8                   EXTENSION
0000 0008           1896  ACT       EQU   11                  ACCOUNT NUMBER
0000 000C           1897  FLBA      EQU   12                  FIRST LOGICAL BLOCK ADDRESS
0000 0010           1898  LLBA      EQU   16                  LAST LOGICAL BLOCK ADDRESS
0000 0014           1899  KEYS      EQU   20                  PROTECT KEYS
0000 0016           1900  LRCL      EQU   22                  LOGICAL RECORD LENGTH
0000 0018           1901  DATE      EQU   24                  DATE FILE ALLOCATED
0000 001C           1902  LUSE      EQU   28                  DATE LAST ASSIGNED
0000 0020           1903  WCNT      EQU   32                  WRITE COUNT
0000 0022           1904  RCNT      EQU   34                  READ COUNT
0000 0024           1905  ATTR      EQU   36                  ATTRIBUTES
0000 0025           1906  BKSZ      EQU   37                  BLOCK SIZE
0000 0026           1907  INBS      EQU   38                  INDEX BLOCK SIZE
0000 0028           1908  CSEC      EQU   40                  CUR SEC OR NO. LOGICAL RECORDS
0000 002C           1909  PSWD      EQU   44                  PASSWORD
0000 0030           1910  DSIZE     EQU   48                  DIRECTORY ENTRY SIZE (IN BYTES)
                    1911  *
0000 3860           1912  DIRECT1   EQU   INBUF+4            WHERE FIRST ENTRY GOES
0000 3890           1913  DIRECT2   EQU   INBUF+4+DSIZE       WHERE SECOND ENTRY GOES
0000 38C0           1914  DIRECT3   EQU   INBUF+4+DSIZE+DSIZE WHERE THIRD ENTRY GOES
                    1915  *
```

SAVE AREAS

```
19D0                        1917           ALIGN 4
19D0                        1918  ERRSAVE  DS    64
      0000 01A8             1919  LDBUF    EQU   ENDAD-STARTAD+X'81'&X'FFFE'
1A10  1B                    1920  STDIRM   DB    27
1A11  1C                    1921  STDIR    DB    28                  START OF DIRECTORY (LRN)
1A12  1F                    1922  ENDDIR   DB    31                  END OF DIRECTORY (LRN)
1A13  20                    1923  STSAV    DB    32                  START OF PROGRAM (LRN)
1A14  08                    1924  DIRSTART DB    8                   DISK DIRECTORY STARTS AT CYLINDER 8
1A15  09                    1925  PROGSTRT DB    9                   FIRST PROGRAM STARTS AT CYLINDER 9
1A16                        1926           DB    *
                            1927  *
                            1928  *
1A16  0000                  1929  NOSTOP   DCX   0                   AUTO VERIFY FLAG
1A18  0000                  1930  VERIFLAG DCX   0                   VERIFY MODE FLAG
1A1A  0000                  1931  NEXTCYL  DCX   0                   OUTPUT DISK PARAMETERS
1A1C  0000                  1932  NEXTHEAD DCX   0
1A1E  0000                  1933  NEXTSECT DCX   0
1A20  0000                  1934  FMDCMDI  DCX   0000                INPUT FLOPPY COMMAND
1A22  0000                  1935  FMDCMDO  DCX   0000                OUTPUT FLOPPY COMMAND
1A24  0000                  1936  FMDCMD   DCX   0000                FLOPPY COMMAND BYTE MODEL
1A26  0000                  1937  MAXLRN   DCX   0000                MAXIMUM LRN
1A28  0000                  1938  NEXTLRN  DCX   0000                OUTPUT FMD LRN POINTER
1A2A  0000                  1939  THISLRN  DCX   0000                INPUT FMD LRN POINTER
1A2C  C020                  1940  DISABLE  DCX   C020                COMMAND DISABLE INTERRUPTS
      0000 1A2D             1941  CLEAR    EQU   DISABLE+1           MAG TAPE CLEAR
1A2E  2123                  1942  MTREAD   DCX   2123                MAG TAPE READ
      0000 1A2F             1943  FORWARD  EQU   MTREAD+1            MAG TAPE FORWARD FILE MARK
1A30  0801                  1944  STOPS    DCX   0801                SELCH STOP
      0000 1A31             1945  DREAD    EQU   STOPS+1             DISK READ COMMAND
1A32  0210                  1946  DWRITE   DCX   0210
      0000 1A33             1947  SWRITE   EQU   DWRITE+1            WRITE TO THE SELCH
1A34  03C8                  1948  RCHECK   DCX   03C8                READCHECK THE DISK
      0000 1A35             1949  DRESET   EQU   RCHECK+1            RESET THE DISK
1A36  1020                  1950  SETCYL   DCX   1020                SET DISK CYLINDER
      0000 1A37             1951  SETHEAD  EQU   SETCYL+1            SET DISK HEAD
1A38  C1C2                  1952  RESTOR   DCX   C1C2
      0000 1A39             1953  SEEK     EQU   RESTOR+1
1A3A  1338                  1954  BKSP     DCX   1338                MAG TAPE BACKSPACE FILE MARK
      0000 1A3B             1955  REWIND   EQU   BKSP+1              MAG TAPE REWIND
1A3C  3070                  1956  SREAD    DCX   3070                SELCH READ
      0000 1A3D             1957  MSMCLFLT EQU   SREAD+1             MSM CLEAR FAULT
1A3E  0411                  1958  RESHEAD  DCX   0411                MSM RESET HEAD
      0000 1A3F             1959  BKSPRCRD EQU   RESHEAD+1           MAG TAPE BACK SPACE RECORD
1A40  3022                  1960  WFILEMK  DCX   3022
      0000 1A41             1961  MTWRITE  EQU   WFILEMK+1
1A42  0000                  1962  BITCOUNT DCX   0
1A44  0000                  1963  DIR.LBA  DCX   0,0                 O.S. DIRECTORY LBA
1A46  0000
1A48  0000                  1964  OSDIR    DCX   0,0,0               CYLINDER,HEAD,SECTOR
1A4A  0000
1A4C  0000
1A4E  0000                  1965  INFLAG   DCX   0                   INPUT DEVICE CODE
1A50  0000                  1966  OUTFLAG  DCX   0                   OUTPUT DEVICE CODE
```

SAVE AREAS

```
1A52 0000          1967 DRWFLAG DCX  0          READ OR WRITE FLAG
1A54 0000          1968 SELERR  DCX  0          SELCH ERROR FLAG
1A56 0000          1969 RETRY   DCX  0          RETRY COUNT
1A58 0000          1970 IOIRHEAD DCX 0          INPUT DISK DIRECTORY POINTERS/
1A5A 0000          1971 IOIRPNT DCX  0          DISK DIRECTORY POINTER
1A5C 0000          1972 IOIRLRN DCX  0          INPUT FLOPPY DIRECTORY POINTER
1A5E 0000          1973 IOIRSEC DCX  0          DISK DIRECTORY SECTOR
1A60 0000          1974 INCYL   DCX  0          INPUT DISK WORK PARAMETERS/
1A62 0000          1975 INHEAD  DCX  0
1A64 0000          1976 INSECT  DCX  0
1A66 0000          1977 OSFLAG  DCX  0
1A68 0000          1978 DISKTYPE DCX 0          10, 80, OR 300 MB
1A6A 0000          1979 DISKTYP1 DCX 0          INPUT DISK TYPE
1A6C 0000          1980 DISKTYP2 DCX 0          OUTPUT DISK TYPE
1A6E 0000          1981 R15SAVE DCX  0
1A70 0000          1982 FIRSTHDI DCX 0          *                              R04
1A72 0000          1983 FIRSTHDO DCX 0          *                              R04
1A74 0000          1984 DIRDAT  DCX  0,0,0      DIRECTORY UPDATE DATA
1A76 0000
1A78 0000
1A7A 0000          1985 STARTLBA DCX 0,0,0,0    STARTING LBA FOR THE OS'S
1A7C 0000
1A7E 0000
1A80 0000
1A82 0000          1986 FINALLBA DCX 0,0,0,0    FINAL LBA FOR THE OS'S
1A84 0000
1A86 0000
1A88 0000
1A8A 0000          1987 BLOCK   DCX  0          SECTOR COUNTER FOR OS BUILD
                   1988 *                       NUMBER OF SECTORS PER TRACK
1A8C 0040          1989 SECTAB  DC   H'64'      0...13.5 MB REMOVABLE
1A8E 0018          1990         DC   H'24'      2...10 MB DISK
1A90 0040          1991         DC   H'64'      4...80 MB DISK
1A92 0040          1992         DC   H'64'      6...300 MB DISK
1A94 0040          1993         DC   H'64'      8...13.5 MB FIXED              R04
1A96 0040          1994         DC   H'64'      A...40.5 MB FIXED              R04
1A98 0040          1995         DC   H'64'      C...67.5 MB FIXED              R04
                   1996 *                       NUMBER OF HEADS
1A9A 0001          1997 HDTAB   DC   H'1'
1A9C 0002          1998         DC   H'2'
1A9E 0005          1999         DC   H'5'
1AA0 0013          2000         DC   H'19'
1AA2 000B          2001         DC   H'11'
1AA4 000D          2002         DC   H'13'
1AA6 000F          2003         DC   H'15'
                   2004 *                       TOTAL NUMBER OF CYLINDERS
1AA8 0337          2005 CYLTAB  DC   H'823'
1AAA 0198          2006         DC   H'408'
1AAC 0337          2007         DC   H'823'
1AAE 0337          2008         DC   H'823'
1AB0 0337          2009         DC   H'823'
1AB2 0337          2010         DC   H'823'
1AB4 0337          2011         DC   H'823'
```

SAVE AREAS

```
                                2012  *                                        NUMBER OF SECTORS PER CYLINDER
1AB6   0040                     2013  LBATAB   DC    H'64'                     64 X 1
1AB8   0030                     2014           DC    H'48'                     24 X 2
1ABA   0140                     2015           DC    H'320'                    64 X 5
1ABC   04C0                     2016           DC    H'1216'                   64 X 19
1ABE   0040                     2017           DC    H'64'                     64 X 1
1AC0   00C0                     2018           DC    H'192'                    64 X 3
1AC2   0140                     2019           DC    H'320'                    64 X 5
                                2020  *
                                2021  *                                        BIT MAP SIZE = NUMBER OF SECTORS
1AC4   001A                     2022  BMAPSIZE DC    H'26'                     64 X 822 / 2048
1AC6   000A                     2023           DC    H'10'                     48 X 407 / 2048
1AC8   0081                     2024           DC    H'129'                    320 X 822 / 2048
1ACA   01E9                     2025           DC    H'489'                    1216 X 822 / 2048
1ACC   001A                     2026           DC    H'26'                     64 X 822 / 2048
1ACE   004E                     2027           DC    H'78'                     192 X 822 / 2048
1AD0   0081                     2028           DC    H'129'                    320 X 822 / 2048
                                2029  *
                                2030  *                                        BITS TO SET IN BIT MAP SO THAT
                                2031  *                                        EVERY SECTOR FROM CYLINDER 0
                                2032  *                                        THRU CYLINDER 7, PLUS AN ADDITIONAL
                                2033  *                                        12,000 CYLINDERS IS MARKED USED.
                                2034  *                                        VALUE IN TABLE IS NUMBER OF BYTES
                                2035  *                                        OF X'FF' TO PUT IN BIT MAP.
                                2036  *
1AD2   061C                     2037  USEDBITS DC    H'1564'                   (64 X 8 +12,000)/8
1AD4   060C                     2038           DC    H'1548'                   (48 X 8 +12,000)/8
1AD6   071C                     2039           DC    H'1820'                   (320 X 8 +12,000)/8
1AD8   0A9C                     2040           DC    H'2716'                   (1216 X 8 +12,000)/8
1ADA   061C                     2041           DC    H'1564'                   (64 X 8 +12,000)/8
1ADC   069C                     2042           DC    H'1692'                   (192 X 8 +12,000)/8
1ADE   071C                     2043           DC    H'1820'                   (320 X 8 +12,000)/8
                                2044  *
                                2045  *
1AE0   0000 0002 0406 0008      2046  DTAB     DB    0,0,0,2,4,6,0,8,10,12
1AE8   0A0C
1AEA   003B                     2047  DEVCODES DCX   3B,33,35,36,0,0,0
1AEC   0033
1AEE   0035
1AF0   0036
1AF2   0000
1AF4   0000
1AF6   0000
                                2048  *
                                2049  *
1AF8                            2050           ALIGN 4
1AF8   0000                     2051  SIZEI    DCX   0000,0000
1AFA   0000
                                2052  *
                                2053  *                                        PROGRAM DEFINITION BLOCK
1AFC   0000                     2054  SEQNUM   DCX   0000                      HEX SEQUENCE NUMBER
1AFE   0000                     2055  PARTNO   DCX   0000                      HEX PART NUMBER
1B00   0000                     2056  REVLEV   DCX   0000                      REV LEVEL
```

SAVE AREAS

```
1B02 0000          2057  LOW       DCX   0000,0A00        LOAD START ADDRESS
1B04 0A00
1B06 0000          2058  HIGH      DCX   0000,3FFF        LOAD END ADDRESS
1B08 3FFF
1B0A 0000          2059  LRNS      DCX   0000             LRN TALLY, THIS PROGRAM
1B0C               2060  FFORM     DS    30               PROGRAM NAME FIELD
1B2A               2061            DO    41
1B2A 0000          2062            DCX   0000             REST OF PDB = ZERO
1B2C 0000          2062            DCX   0000             REST OF PDB = ZERO
1B2E 0000          2062            DCX   0000             REST OF PDB = ZERO
1B30 0000          2062            DCX   0000             REST OF PDB = ZERO
1B32 0000          2062            DCX   0000             REST OF PDB = ZERO
1B34 0000          2062            DCX   0000             REST OF PDB = ZERO
1B36 0000          2062            DCX   0000             REST OF PDB = ZERO
1B38 0000          2062            DCX   0000             REST OF PDB = ZERO
1B3A 0000          2062            DCX   0000             REST OF PDB = ZERO
1B3C 0000          2062            DCX   0000             REST OF PDB = ZERO
1B3E 0000          2062            DCX   0000             REST OF PDB = ZERO
1B40 0000          2062            DCX   0000             REST OF PDB = ZERO
1B42 0000          2062            DCX   0000             REST OF PDB = ZERO
1B44 0000          2062            DCX   0000             REST OF PDB = ZERO
1B46 0000          2062            DCX   0000             REST OF PDB = ZERO
1B48 0000          2062            DCX   0000             REST OF PDB = ZERO
1B4A 0000          2062            DCX   0000             REST OF PDB = ZERO
1B4C 0000          2062            DCX   0000             REST OF PDB = ZERO
1B4E 0000          2062            DCX   0000             REST OF PDB = ZERO
1B50 0000          2062            DCX   0000             REST OF PDB = ZERO
1B52 0000          2062            DCX   0000             REST OF PDB = ZERO
1B54 0000          2062            DCX   0000             REST OF PDB = ZERO
1B56 0000          2062            DCX   0000             REST OF PDB = ZERO
1B58 0000          2062            DCX   0000             REST OF PDB = ZERO
1B5A 0000          2062            DCX   0000             REST OF PDB = ZERO
1B5C 0000          2062            DCX   0000             REST OF PDB = ZERO
1B5E 0000          2062            DCX   0000             REST OF PDB = ZERO
1B60 0000          2062            DCX   0000             REST OF PDB = ZERO
1B62 0000          2062            DCX   0000             REST OF PDB = ZERO
1B64 0000          2062            DCX   0000             REST OF PDB = ZERO
1B66 0000          2062            DCX   0000             REST OF PDB = ZERO
1B68 0000          2062            DCX   0000             REST OF PDB = ZERO
1B6A 0000          2062            DCX   0000             REST OF PDB = ZERO
1B6C 0000          2062            DCX   0000             REST OF PDB = ZERO
1B6E 0000          2062            DCX   0000             REST OF PDB = ZERO
1B70 0000          2062            DCX   0000             REST OF PDB = ZERO
1B72 0000          2062            DCX   0000             REST OF PDB = ZERO
1B74 0000          2062            DCX   0000             REST OF PDB = ZERO
1B76 0000          2062            DCX   0000             REST OF PDB = ZERO
1B78 0000          2062            DCX   0000             REST OF PDB = ZERO
1B7A 0000          2062            DCX   0000             REST OF PDB = ZERO
1B7C 0000          2063            DCX   0000
1B7E 0000          2064  RECORDS   DCX   0                NUMBER OF 256 BYTE RECORDS
1B80 0000          2065  RECORDSV  DCX   0                SAME INFO FOR VERIFY PHASE
1B82 0000          2066  LEFTOVER  DCX   0                NUMBER OF BYTES IN LAST RECORD
1B84 0000          2067  R5SAVE    DCX   0
```

SAVE AREAS

```
1B86    0000                2068  PROGSIZE DCX   0              NUMBER OF BYTES WRITTEN
1B88    0000                2069  PDBSTART DCX   0              START LRN FOR PDB
1B8A    0000                2070  UTILITY  DCX   0              PRINT UTILITY FLAG
        0000 1B2A           2071  FMDFLAG  EQU   FFORM+30       DISKETTE FLAG
        0000 1B2C           2072  PKGREV   EQU   FFORM+32       PACKAGE REVISION
```

MEDIA INITIALIZATION

```
1B8C  2501           2074  RUN.CMD  LCS   R0,1
1B8E  4000 1A66      2075           STH   R0,OSFLAG          FORCE OS FLAG SET
1B92  4840 1A4E      2076  RUN.0000 LH    R4,INFLAG
1B96  2741           2077           SIS   R4,1               MAG TAPE INPUT?
* 1B98  2139         2078           BNZ   RUN.0001           SKIP IF NO
1B9A  4820 1664      2079           LH    R2,INDEV+$VALU1
1B9E  DE20 1A2D      2080           OC    R2,CLEAR
1BA2  41FC 2EBC      2081           BAL   R15,NOMOTION
1BA6  DE20 1A38      2082           OC    R2,REWIND          REWIND THE MAG TAPE
1BAA  4820 167C      2083  RUN.0001 LH    R2,OUTDEV+$VALU1   OUTPUT DEVICE NUMBER
1BAE  4830 1A6C      2084           LH    R3,DISKTYP2
1BB2  4030 1A68      2085           STH   R3,DISKTYPE        DISKTYPE SET TO OUTPUT DISK
1BB6  2400           2086           LIS   R0,0               "REWIND" INPUT DISK
1BB8  4000 1A58      2087           STH   R0,IDIRHEAD        POINTERS ALL SET TO ZERO
1BBC  4000 1A5E      2088           STH   R0,IDIRSEC
1BC0  4000 1A5A      2089           STH   R0,IDIRPNT         DIRECTORY BUFFER INDEX TO ZERO
1BC4  D340 1A11      2090           LB    R4,STDIR           "REWIND" INPUT DISKETTE
1BC8  4040 1A5C      2091           STH   R4,IDIRLRN         PRESET INPUT DIRECTORY LRN
1BCC  D310 1A15      2092           LB    R1,PROGSTRT
1BD0  4010 1A1A      2093           STH   R1,NEXTCYL         INITIAL POINTERS FOR OUTPUT DISK
1BD4  4000 1A1C      2094           STH   R0,NEXTHEAD
1BD8  4000 1A1E      2095           STH   R0,NEXTSECT
1BDC  4830 1A50      2096           LH    R3,OUTFLAG
1BE0  2731           2097           SIS   R3,1               MAG TAPE OUTPUT?  (CODE 1)
1BE2  4330 1CF2      2098           BZ    INITMAG            BRANCH IF YES
1BE6  2731           2099           SIS   R3,1               FLOPPY OUTPUT?    (CODE 2)
1BE8  4230 1D88      2100           BNZ   INITDISK           NO, MUST BE DISK
                     2101  *
                     2102  * INITIALIZE THE FLOPPY
                     2103  *
1BEC  2400           2104           LIS   R0,0
1BEE  4000 1A66      2105           STH   R0,OSFLAG          KILL OS FLAG
1BF2  4830 16A0      2106           LH    R3,ODRIVE+$VALU1   PICK UP DRIVE NUMBER
1BF6  9134           2107           SLLS  R3,4               POSITION DRIVE SELECT BITS
1BF8  C630 00C0      2108           OHI   R3,X'C0'           OR IN DISARM BITS
1BFC  4030 1A24      2109           STH   R3,FMDCMD          SAVE COMMAND MODEL
1C00  4030 1A22      2110           STH   R3,FMDCMDO
1C04  41D0 2B0E      2111           BAL   R13,FMDIDLE        IDLE CHECK
1C08  C840 0800      2112           LHI   R4,2048            MAX LRN VALUE PLUS 26
1C0C  4810 1A24      2113  FINDMAX  LH    R1,FMDCMD
1C10  CB40 001A      2114           SHI   R4,26              DECREMENT ONE TRACK
1C14  2611           2115           AIS   R1,1               FORM READ COMMAND
1C16  41C0 2ABC      2116           BAL   R12,SULH56         PRESET ADDRESS REGISTERS
1C1A  9824           2117           WHR   R2,R4              WRITE LRN TO CONTROLLER
1C1C  9E21           2118           OCR   R2,R1              ISSUE READ COMMAND
1C1E  9D23           2119           SSR   R2,R3
1C20  2081           2120           BTBS  8,1                WAIT FOR NON-BUSY
1C22  D925 0000      2121  FINDMAX2 RH    R2,0(R5)          READ 2 BYTES AT A TIME
1C26  2652           2122           AIS   R5,2
1C28  0556           2123           CLAR  R5,R6
1C2A  2084           2124           BLS   FINDMAX2           LOOP
1C2C  9D21           2125           SSR   R2,R1
1C2E  C310 0050      2126           THI   R1,X'50'           DEFECTIVE SECTOR STATUS?
```

MEDIA INITIALIZATION

```
1C32  2137          2127           BNZS  FINDMAX3          BACK DOWN IF YES
1C34  41D0 2B12     2128           BAL   R13,STOP          ELSE FOUND MAX LRN
1C38  4040 1A26     2129           STH   R4,MAXLRN         SAVE VALUE
1C3C  4300 1C66     2130           B     INIT.002          AND LEAVE
                    2131  *
1C40  41D0 2B12     2132  FINDMAX3 BAL   R13,STOP
1C44  C540 079E     2133           CLHI  R4,1950           AT MINIMUM?
1C48  4380 1C0C     2134           BNL   FINDMAX           TRY NEXT TRACK
                    2135  *
1C4C  C850 3314     2136  MDERR001 LDAI  R5,MEDMSG         "DEFECTIVE MEDIA"
                    2137  *
1C50  41F0 12BC     2138  CONMSG1  BAL   R15,SETKB         SELECT CONSOLE FOR OUTPUT
1C54  C8F0 0AAA     2139           LDAI  R15,OPTIN         SET RETURN TO COMMAND MODE
                    2140  *
1C58  D000 405C     2141  CONMSG   STM   R0,REGISTER       SAVE REGISTERS
1C5C  41F0 0FD2     2142           BAL   R15,PRINT         PRINT MESSAGE
1C60  D100 405C     2143           LM    R0,REGISTER       RESTORE REGISTERS
1C64  030F          2144           BR    R15               RETURN
                    2145  *
                    2146  *
1C66  4830 1A18     2147  INIT.002 LH    R3,VERIFLAG
1C6A  4230 1CC6     2148           BNZ   BOOTCHCK          BRANCH IF VERIFY COMMAND
1C6E  41D0 280E     2149           BAL   R13,FMDIDLE       IDLE CHECK
1C72  41C0 2ABC     2150           BAL   R12,SULIMITS      LIMITS INBUF,INBUF+127
1C76  41C0 2AD0     2151           BAL   R12,ZEROBUFF      SET INBUF TO ZERO
1C7A  D340 1A11     2152           LB    R4,STDIR          DIRECTORY STARTS AT LRN 28
1C7E  41E0 2D00     2153           BAL   R14,WRLRN         WRITE FIRST RECORD
1C82  2641          2154           AIS   R4,1              NEXT LRN
1C84  41C0 2ABC     2155           BAL   R12,SULH56
1C88  41E0 2D00     2156           BAL   R14,WRLRN         WRITE SECOND RECORD
1C8C  2641          2157           AIS   R4,1              NEXT LRN
1C8E  41C0 2ABC     2158           BAL   R12,SULH56
1C92  41E0 2D00     2159           BAL   R14,WRLRN         WRITE THIRD RECORD
1C96  2541          2160           LCS   R4,1
1C98  4040 38DA     2161           STH   R4,INBUF+126      MARK DIRECTORY END
1C9C  D340 1A12     2162           LB    R4,ENDDIR         LAST LRN
1CA0  41C0 2ABC     2163           BAL   R12,SULH56
1CA4  41E0 2D00     2164           BAL   R14,WRLRN         WRITE LAST DIRECTORY RECORD
                    2165  *
1CA8  2445          2166           LIS   R4,5              POINT TO LRN 5
1CAA  C850 3018     2167           LDAI  R5,BOOTST         BOOT START ADDRESS
1CAE  C860 313F     2168           LDAI  R6,ENDAD          BOOT END ADDRESS
1CB2  41E0 2D00     2169           BAL   R14,WRLRN         WRITE FLOPPY BOOT
1CB6  D340 1A13     2170  FLOPPSET LB    R4,STSAV          START LRN FOR FIRST PROGRAM
1CBA  4040 1A28     2171           STH   R4,NEXTLRN        WORKING LRN POINTER
1CBE  4040 1B88     2172           STH   R4,PDBSTART       STARTING LRN FOR FIRST PDB
1CC2  4300 1F7C     2173           B     FINDNEXT
                    2174  *
                    2175  *
1CC6  2445          2176  BOOTCHCK LIS   R4,5              LRN 5
1CC8  C850 385C     2177           LDAI  R5,INBUF
1CCC  C865 0127     2178           LHI   R6,ENDAD-BOOTST(R5)
1CD0  41E0 2CA0     2179           BAL   R14,RDLRN         READ THE FLOPPY BOOTLOADER
```

MEDIA INITIALIZATION

```
1CD4  C850 3018  2180           LDAI  R5,BOOTST           PATTERN START ADDRESS
1CD8  C860 0128  2181           LHI   R6,ENDAD+1-BOOTST   NUMBER OF BYTES
1CDC  41E0 2EA4  2182           BAL   R14,COMPARE         COMPARE PATTERN TO INBUF
1CE0  41F0 1CE8  2183           BAL   R15,VERR.001        TEST IF ERROR
1CE4  4300 1CB6  2184           B     FLOPPSET            CONTINUE
1CE8  033F       2185  VERR.001 BER   R15                 RETURN IF GOOD COMPARE
1CEA  C850 33A1  2186           LDAI  R5,VERRMSG1         "BOOT LOADER VERIFY ERROR"
1CEE  4300 1C58  2187           B     CONMSG              RETURN FROM CONMSG ON R15


                 2189  *  INITIALIZE THE MAG TAPE
                 2190  *
1CF2  2400       2191  INITMAG LIS   R0,0
1CF4  4000 1A66  2192          STH   R0,OSFLAG           KILL OS FLAG
1CF8  4840 1688  2193          LH    R4,SELCH2+$VALU1    PICK UP SELCH ADDRESS
1CFC  DE40 1A30  2194          OC    R4,STOPS
1D00  DE20 1A2D  2195          OC    R2,CLEAR
1D04  41F0 2EBC  2196          BAL   R15,NOMOTION
1D08  DE20 1A39  2197          OC    R2,REWIND           REWIND THE OUTPUT TAPE
1D0C  41F0 2EBC  2198          BAL   R15,NOMOTION        WAIT FOR NOMOTION
1D10  4830 1A18  2199          LH    R3,VERIFLAG         TEST IF DOING A VERIFY
1D14  4230 1D4C  2200          BNZ   VMAGBOOT            BRANCH IF YES
1D18  DE20 1A40  2201          OC    R2,WFILEMK          WRITE A FILE MARK AT START OF
1D1C  41F0 2EBC  2202          BAL   R15,NOMOTION        TAPE, THEN BACK SPACE OVER IT
1D20  DE20 1A3A  2203          OC    R2,BKSP             TO GET AWAY FROM BOT
1D24  C850 3140  2204          LDAI  R5,MTBOOT           AUTOLOAD START
1D28  C865 004F  2205          LHI   R6,X'4F'(R5)        AUTOLOAD SECTION END
1D2C  41D0 2B62  2206          BAL   R13,WRITEMAG        WRITE THE AUTOLOAD PIECE
1D30  C850 3194  2207          LDAI  R5,MTLOADS          BOOT LOADER START ADDRESS
1D34  C860 32F7  2208          LDAI  R6,MTLOADE          BOOT LOADER END ADDRESS
1D38  41D0 2B62  2209          BAL   R13,WRITEMAG        WRITE THE BOOT LOADER
1D3C  DE20 1A40  2210  MAGFIN  OC    R2,WFILEMK          WRITE A FILE MARK
1D40  41F0 2EBC  2211          BAL   R15,NOMOTION
1D44  DE20 1A40  2212          OC    R2,WFILEMK          WRITE ANOTHER FILE MARK
1D48  4300 1F7C  2213          B     FINDNEXT            GO START LIBRARY
                 2214  *
1D4C  C850 385C  2215  VMAGBOOT LDAI  R5,INBUF           START ADDRESS
1D50  C860 38AB  2216          LHI   R6,INBUF+X'4F'      END ADDRESS
1D54  41D0 2B6A  2217          BAL   R13,READMAG         READ THE AUTOLOAD PIECE
1D58  C850 3140  2218          LDAI  R5,MTBOOT           PATTERN START ADDRESS
1D5C  C860 0050  2219          LHI   R6,X'50'            NUMBER OF BYTES
1D60  41E0 2EA4  2220          BAL   R14,COMPARE         COMPARE PATTERN TO INBUF
1D64  41F0 1CE8  2221          BAL   R15,VERR.001        "BOOT LOADER VERIFY ERROR"
1D68  C850 385C  2222          LDAI  R5,INBUF            START ADDRESS
1D6C  C865 0164  2223          LHI   R6,MTLOADE+1-MTLOADS(R5) END ADDRESS
1D70  41D0 2B6A  2224          BAL   R13,READMAG
1D74  C850 3194  2225          LDAI  R5,MTLOADS          PATTERN START ADDRESS
1D78  C860 0164  2226          LHI   R6,MTLOADE+1-MTLOADS NUMBER OF BYTES
1D7C  41E0 2EA4  2227          BAL   R14,COMPARE         COMPARE PATTERN TO INBUF
1D80  41F0 1CE8  2228          BAL   R15,VERR.001        "BOOT LOADER VERIFY ERROR"
1D84  4300 1F7C  2229          B     FINDNEXT
```

MEDIA INITIALIZATION

```
                                2231  *  INITIALIZE THE DISK  *
                                2232  *
1D88  4830 167E                 2233  INITDISK LH    R3,OUTDEV+$VALU2    GET CONTROLLER ADDRESS
1D8C  4840 1688                 2234           LH    R4,SELCH2+$VALU1    GET SELCH ADDRESS
1D90  DE40 1A30                 2235           OC    R4,STOPS
1D94  DE30 1A35                 2236           OC    R3,DRESET           RESET THE CONTROLLER
1D98  9D21                      2237  SENSTA1  SSR   R2,R1                DRIVE STATUS
1D9A  C310 0010                 2238           THI   R1,X'10'
*1D9E 2033                      2239           BNZ   SENSTA1             WAIT FOR ADDRESS INTERLOCK
                                2240  *
1DA0  4870 1A68                 2241           LH    R7,DISKTYPE
1DA4  2774                      2242           SIS   R7,4
1DA6  2183                      2243           BLS   NOTMSM
1DA8  DE20 1A3D                 2244           OC    R2,MSMCLFLT         CLEAR FAULT STATUS BITS
1DAC  4870 1A66                 2245  NOTMSM   LH    R7,OSFLAG           TEST IF DOING OS FRONT-END
1DB0  4330 1EC4                 2246           BZ    INITDSK1            SKIP IF NO
1DB4  2480                      2247  INITOS   LIS   R8,0                CYLINDER 0
1DB6  4890 1A72                 2248           LH    R9,FIRSTHDO         FIRST OUTPUT HEAD
1DBA  24A0                      2249           LIS   R10,0               SECTOR 0
1DBC  40A0 1B8A                 2250           STH   R10,UTILITY         CLEAR UTILITY FLAG
1DC0  41D0 2B20                 2251           BAL   R13,FILESET
1DC4  DE20 1A38                 2252           OC    R2,RESTOR
                                2253  *                                 SET UP THE BIT MAP FIRST
                                2254  *
1DC8  4870 1A68                 2255           LH    R7,DISKTYPE
1DCC  4807 1AD2                 2256           LH    R0,USEDBITS(R7)     NUMBER OF 'FF' BYTES
1DD0  2480                      2257           LIS   R8,0                BIT MAP GOES AT LBA 1
1DD2  2490                      2258           LIS   R9,0                (CYLINDER 0,HEAD 0,SECTOR 1)
1DD4  24A1                      2259           LIS   R10,1
1DD6  2410                      2260  BMAP0001 LIS   R1,0                INDEX
1DD8  25B1                      2261           LCS   R11,1               FOXES
1DDA  2701                      2262  BMAP0002 SIS   R0,1                DECREMENT BYTE COUNT
1DDC  4320 1E00                 2263           BNP   BMAP0003            DONE WITH FOXES
1DE0  D2B1 385C                 2264           STB   R11,INBUF(R1)       SET 8 BITS
1DE4  2611                      2265           AIS   R1,1                INCREMENT INDEX
1DE6  C510 0100                 2266           CLHI  R1,256              ONE SECTOR?
1DEA  2088                      2267           BLS   BMAP0002            LOOP IF NO
1DEC  4000 1A42                 2268           STH   R0,BITCOUNT         YES, SAVE COUNT
1DF0  41F0 2FD8                 2269           BAL   R15,DISKSET         WRITE THE SECTOR
1DF4  41F0 1EA6                 2270           BAL   R15,DISKERR2        NON VERIFY ERROR
1DF8  4800 1A42                 2271           LH    R0,BITCOUNT
1DFC  4300 1DD6                 2272           B     BMAP0001            LOOP
                                2273  *
1E00  24B0                      2274  BMAP0003 LIS   R11,0               FILL OUT REST OF SECTOR WITH ZERO
1E02  D2B1 385C                 2275           STB   R11,INBUF(R1)
1E06  2611                      2276           AIS   R1,1
1E08  C510 0100                 2277           CLHI  R1,256
1E0C  2086                      2278           BLS   BMAP0003
1E0E  41F0 2FD8                 2279           BAL   R15,DISKSET         WRITE THIS SECTOR
1E12  41F0 1EA6                 2280           BAL   R15,DISKERR2        NON VERIFY ERROR
1E16  41C0 2AC6                 2281           BAL   R12,SULH56,2
1E1A  41C0 2AD0                 2282           BAL   R12,ZEROBUFF
1E1E  4807 1AD2                 2283           LH    R0,USEDBITS(R7)
```

MEDIA INITIALIZATION

```
1E22  CA00 00FF  2284            AHI   R0,255              ROUND IT UP
1E26  9008       2285            SRLS  R0,8                NUMBER OF SECTORS WRITTEN
1E28  4507 1AC4  2286  BMAP0004 CLH   R0,BMAPSIZE(R7)      END OF MAP?
1E2C  4000 1A42  2287            STH   R0,BITCOUNT
*1E30 2389       2288            BNL   BMAPDONE            DONE WITH BITMAP
1E32  41F0 2FD8  2289            BAL   R15,DISKSET         WRITE THIS SECTOR
1E36  41F0 1EA6  2290            BAL   R15,DISKERR2
1E3A  4800 1A42  2291            LH    R0,BITCOUNT
1E3E  2601       2292            AIS   R0,1
*1E40 220C       2293            B     BMAP0004
                 2294  *
1E42  4080 1A48  2295  BMAPDONE STH   R8,OSDIR            SAVE CYLINDER, HEAD, AND SECTOR
1E46  4090 1A4A  2296            STH   R9,OSDIR+2          DIRECTORY FOLLOWS BIT MAP
1E4A  40A0 1A4C  2297            STH   R10,OSDIR+4
1E4E  41C0 2AC6  2298            BAL   R12,SULH56,2
1E52  41C0 2AD0  2299            BAL   R12,ZEROBUFF
1E56  4870 16C4  2300            LH    R7,DSCRPT+$VALU1    VOLUME = 'MMD '
1E5A  4070 385C  2301            STH   R7,INBUF+VD.VOL
1E5E  4870 16C6  2302            LH    R7,DSCRPT+$VALU2
1E62  4070 385E  2303            STH   R7,INBUF+VD.VOL+2
1E66  2471       2304            LIS   R7,1                BIT MAP LOGICAL BLOCK ADDRESS
1E68  4070 3872  2305            STH   R7,INBUF+BM.LBA+2
1E6C  41F0 2FA8  2306            BAL   R15,FORMLBA         CHANGE 8,9,10 TO LBA
1E70  4000 3864  2307            STH   R0,INBUF+VD.FDB     STORE DIRECTORY LBA
1E74  4010 3866  2308            STH   R1,INBUF+VD.FDB+2
1E78  41F0 2F8A  2309            BAL   R15,NEXTDISK        NEXT LBA VALUE
1E7C  4080 1A1A  2310            STH   R8,NEXTCYL          WHERE FIRST OS GOES
1E80  4090 1A1C  2311            STH   R9,NEXTHEAD
1E84  40A0 1A1E  2312            STH   R10,NEXTSECT
1E88  2480       2313            LIS   R8,0                CYLINDER 0
1E8A  4890 1A72  2314            LH    R9,FIRSTHDO         FIRST OUTPUT HEAD
1E8E  24A0       2315            LIS   R10,0               SECTOR ZERO
1E90  41F0 2FD8  2316            BAL   R15,DISKSET         WRITE THE VOLUME DESCRIPTOR
1E94  41F0 1E9C  2317            BAL   R15,DISKERR1        "VOLUME DESCRIPTOR VERIFY ERROR"
1E98  4300 1F7C  2318            B     FINDNEXT            GO DO FIRST PROGRAM
                 2319  *
1E9C  033F       2320  DISKERR1 BER   R15                 RETURN IF GOOD COMPARE
1E9E  C850 3442  2321            LDAI  R5,OSERRM3          "VOLUME DESCRIPTOR VERIFY ERROR"
1EA2  4300 1C58  2322            B     CONMSG              RETURN FROM CONMSG ON R15
                 2323  *
1EA6  033F       2324  DISKERR2 BER   R15                 RETURN IF GOOD COMPARE
                 2325  *                                 IF BAD, CYLINDER-HEAD-SECTOR
                 2326  *                                 VALUES HAVE NOT BEEN INCREMENTED
1EA8  4850 1B8A  2327            LH    R5,UTILITY          TEST UTILITY FLAG.  IF NOT ZERO,
                 2328  *                                 ERROR HAS ALREADY BEEN REPORTED.
1EAC  4230 2F8A  2329            BNZ   NEXTDISK            EXIT FROM NEXTDISK ON R15
1EB0  40F0 1B8A  2330            STH   R15,UTILITY         SET UTILITY FLAG
1EB4  C850 342B  2331            LDAI  R5,OSERRM2          "BIT MAP VERIFY ERROR"
1EB8  41F0 1C58  2332            BAL   R15,CONMSG          OUTPUT THE MESSAGE ONE TIME
1EBC  48F0 1B8A  2333            LH    R15,UTILITY         RESTORE R15
1EC0  4300 2F8A  2334            B     NEXTDISK            RETURN FROM NEXTDISK ON R15
                 2335  *
1EC4  D380 1A14  2336  INITDSK1 LB    R8,DIRSTART         CYLINDER 8
```

MEDIA INITIALIZATION

```
1EC8  4890 1A72          2337             LH    R9,FIRSTHDO       FIRST OUTPUT HEAD
1ECC  24A0               2338             LIS   R10,0             SECTOR 0
1ECE  41D0 2B20          2339             BAL   R13,FILESET
1ED2  DE20 1A38          2340             OC    R2,RESTOR         DRIVE RESTORE
1ED6  41D0 2B4C          2341             BAL   R13,DISKWAIT      CONTROLLER IDLE
1EDA  41C0 2AC6          2342             BAL   R12,SULH56,2       LIMITS INBUF,INBUF+255
1EDE  41C0 2AD0          2343             BAL   R12,ZEROBUFF      CLEAR INBUF
1EE2  C870 EEEE          2344             LHI   R7,X'EEEE'        FILL FIRST 8 BYTES WITH 'E'
1EE6  4070 385C          2345             STH   R7,INBUF
1EEA  4070 385E          2346             STH   R7,INBUF+2
1EEE  4070 3860          2347             STH   R7,INBUF+4
1EF2  4070 3862          2348             STH   R7,INBUF+6
1EF6  4890 1A72          2349  INITCYL    LH    R9,FIRSTHDO       FIRST OUTPUT HEAD
1EFA  41D0 2B20          2350             BAL   R13,FILESET
1EFE  DE20 1A39          2351             OC    R2,SEEK
1F02  41D0 2B4C          2352             BAL   R13,DISKWAIT
1F06  24A0               2353  INITTRK    LIS   R10,0             SECTOR 0
1F08  41E0 2C70          2354  INITSEC    BAL   R14,READCHCK      READ CHECK THE SECTOR
1F0C  C310 0020          2355             THI   R1,X'20'
1F10  4230 1C4C          2356             BNZ   MDERR001          'DEFECTIVE SECTOR'
1F14  41C0 2AC6          2357             BAL   R12,SULH56,2
1F18  4810 1A18          2358             LH    R1,VERIFLAG       TEST VERIFY FLAG
1F1C  433C 1F44          2359             BZ    INITSEC1          SKIP IF RESET
1F20  41E0 2DD6          2360             BAL   R14,READSECT      READ FIRST DIRECTORY SECTOR
1F24  4800 385C          2361             LH    R0,INBUF          EEEE
1F28  4A00 385E          2362             AH    R0,INBUF+2        +EEEE=DDDC
1F2C  4A00 3860          2363             AH    R0,INBUF+4        +EEEE=CCCA
1F30  4A00 3862          2364             AH    R0,INBUF+6        +EEEE=BBB8
1F34  C500 BBB8          2365             CLHI  R0,X'BB88'        CHECK IT
1F38  4330 1F7C          2366             BE    FINDNEXT          GO ON IF MATCH
1F3C  C850 33D8          2367  DIRECTER   LDAI  R5,ERRMSG1        "INVALID DIRECTORY ON OUTPUT DRIVE"
1F40  4300 1C50          2368             B     CONMSG1           GETURN TO COMMAND MODE
1F44  41E0 2DDA          2369  INITSEC1   BAL   R14,WRITSECT       WRITE A SECTOR
1F48  2400               2370             LIS   R0,0              CLEAR BUFFER FOR NEXT SECTOR
1F4A  4000 385C          2371             STH   R0,INBUF
1F4E  4000 385E          2372             STH   R0,INBUF+2
1F52  4000 3860          2373             STH   R0,INBUF+4
1F56  4000 3862          2374             STH   R0,INBUF+6
1F5A  4870 1A68          2375             LH    R7,DISKTYPE
1F5E  26A1               2376             AIS   R10,1             INCREMENT TO NEXT SECTOR
1F60  45A7 1A8C          2377             CLH   R10,SECTAB(R7)
1F64  4280 1F08          2378             BL    INITSEC           READ CHECK NEXT SECTOR
1F68  2691               2379             AIS   R9,1              INCREMENT TO NEXT HEAD
1F6A  4597 1A9A          2380             CLH   R9,HDTAB(R7)
1F6E  4280 1F06          2381             BL    INITTRK
1F72  2681               2382             AIS   R8,1              INCREMENT TO NEXT CYLINDER
1F74  D480 1A15          2383             CLB   R8,PROGSTRT       CYLINDER 9?
1F78  4320 1EF6          2384             BNP   INITCYL           ZERO OUT CYLINDERS 8 & 9
```

FIND A PROGRAM ON THE INPUT

```
1F7C  4830 1A4E  2386  FINDNEXT LH    R3,INFLAG          LOOK AT INPUT DEVICE CODE
1F80  4820 1664  2387           LH    R2,INDEV+$VALU1    GET INPUT DEVICE NUMBER
1F84  C530 0001  2388           CLHI  R3,1               MAG TAPE ?
*1F88 2337       2389           BE    MAGFF
1F8A  C530 0002  2390           CLHI  R3,2               FLOPPY DEVICE?
1F8E  4330 203E  2391           BE    FLOPPY
1F92  4300 1FB8  2392           B     DISK               DISK DEVICE
                 2393  *
1F96  4840 1670  2394  MAGFF    LH    R4,SELCH1+$VALU1   GET SELCH ADDRESS
1F9A  2333       2395           BZS   MAGFF1             SKIP IF NO SELCH
1F9C  DE40 1A30  2396           OC    R4,STOPS           SELCH STOP
1FA0  41F0 2EBC  2397  MAGFF1   BAL   R15,NOMOTION       CHECK FOR NOMOTION
1FA4  DE20 1A2F  2398           OC    R2,FORWARD         FORWARD FILE MARK
1FA8  C850 375C  2399           LDAI  R5,PDB
1FAC  C865 0032  2400           LHI   R6,50(R5)
1FB0  41D0 2B6A  2401           BAL   R13,READMAG        READ THE FIRST PDB
1FB4  4300 20A4  2402           B     TRYTHIS
                 2403  *
1FB8  4830 1666  2404  DISK     LH    R3,INDEV+$VALU2    CONTROLLER ADDRESS
1FBC  4840 1670  2405           LH    R4,SELCH1+$VALU1   SELCH ADDRESS
1FC0  4850 1A6A  2406           LH    R5,DISKTYP1
1FC4  4050 1A68  2407           STH   R5,DISKTYPE        SELECT INPUT DRIVE
1FC8  48B0 1A5A  2408           LH    R11,IDIRPNT        DIRECTORY POINTER
1FCC  4230 1FF0  2409           BNZ   INCRIDP
1FD0  C850 365C  2410           LHI   R5,IDIRBLK         STARTING ADDRESS
1FD4  C865 00FF  2411           LHI   R6,255(R5)         ENDING ADDRESS
1FD8  D380 1A14  2412           LB    R8,DIRSTART        'CYLINDER'
1FDC  4890 1A58  2413           LH    R9,IDIRHEAD        'HEAD'
1FE0  48A0 1A5E  2414           LH    R10,IDIRSEC        DIRECTORY SECTOR
1FE4  41E0 2DD6  2415           BAL   R14,READSECT       GET 256 BYTES OF DIRECTORY
1FE8  08EA       2416           LDAR  R14,R10
1FEA  06E9       2417           OAR   R14,R9             TEST IF HEAD 0, SECTOR ZERO
1FEC  2132       2418           BNZS  INCRIDP            NO, NOT FIRST DIRECTORY BLOCK
1FEE  26B8       2419           AIS   R11,8              SKIP OVER E'S IN FIRST BLOCK
1FF0  26B8       2420  INCRIDP  AIS   R11,8              INCREMENT INPUT DIRECTORY POINTER
1FF2  40B0 1A5A  2421           STH   R11,IDIRPNT        FOR NEXT TIME
1FF6  C5B0 0100  2422           CLHI  R11,256
1FFA  2187       2423           BLS   DECRIPNT
1FFC  24A0       2424           LIS   R10,0
1FFE  40A0 1A5A  2425           STH   R10,IDIRPNT        RESET IF END OF BLOCK
2002  24A1       2426           LIS   R10,1
2004  61A0 1A5E  2427           AHM   R10,IDIRSEC        POINT TO NEXT SECTOR
2008  27B8       2428  DECRIPNT SIS   R11,8              DECREMENT POINTER
200A  48EB 365C  2429           LH    R14,IDIRBLK(R11)   FETCH PDB PARAMETERS
200E  4330 2A42  2430           BZ    ENDOVOL1           ZERO MEANS END OF VOLUME
2012  488B 3660  2431           LH    R8,IDIRBLK+4(R11)  CYLINDER
2016  D39B 3663  2432           LB    R9,IDIRBLK+7(R11)  HEAD
201A  D3AB 3662  2433           LB    R10,IDIRBLK+6(R11) SECTOR
201E  C850 375C  2434           LDAI  R5,PDB             PDB BUFFER LIMITS
2022  C865 00FF  2435           LHI   R6,255(R5)
2026  41E0 2DD6  2436           BAL   R14,READSECT       READ THE PDB
202A  41F0 2F8A  2437           BAL   R15,NEXTDISK       BUMP POINTERS
202E  4080 1A60  2438           STH   R8,INCYL           SAVE INPUT DISK POSITION
```

FIND A PROGRAM ON THE INPUT

```
2032  4090 1A62          2439          STH   R9,INHEAD
2036  40A0 1A64          2440          STH   R10,INSECT
203A  4300 20A4          2441          B     TRYTHIS
                         2442  *
203E  4830 1694          2443  FLOPPY  LH    R3,IORIVE+$VALU1     INPUT DRIVE SELECTION
2042  9134               2444          SLLS  R3,4
2044  C630 00C0          2445          OHI   R3,X'C0'            FORM COMMAND MODEL
2048  4030 1A20          2446          STH   R3,FMDCMDI          SAVE IT
204C  4030 1A24          2447          STH   R3,FMDCMD
2050  48B0 1A5A          2448          LH    R11,IDIRPNT         CURRENT DIRECTORY BLOCK INDEX
*2054 213C               2449          BNZ   FLOPPY1             SKIP IF NOT ZERO
2056  4840 1A5C          2450          LH    R4,IDIRLRN
205A  C850 365C          2451          LHI   R5,IDIRBLK          SET UPPER
205E  C865 007F          2452          LHI   R6,127(R5)          AND LOWER LIMITS
2062  41E0 2CA0          2453          BAL   R14,RDLRN           READ 127 BYTES OF DIRECTORY
2066  2441               2454          LIS   R4,1                INCREMENT FOR NEXT TIME
2068  6140 1A5C          2455          AHM   R4,IDIRLRN
206C  26B4               2456  FLOPPY1 AIS   R11,4
206E  40B0 1A5A          2457          STH   R11,IDIRPNT
2072  C5B0 0080          2458          CLHI  R11,128             AT LIMIT?
2076  2184               2459          BLS   FLOPPY2
2078  24F0               2460          LIS   R15,0               RESET TO ZERO IF YES
207A  40F0 1A5A          2461          STH   R15,IDIRPNT
207E  27B4               2462  FLOPPY2 SIS   R11,4
2080  484B 365C          2463          LH    R4,IDIRBLK(R11)     GET SEQ NUM FROM DIRECTORY
2084  4330 2A3E          2464          BZ    ENDOVOL             ZERO MEANS END OF VOLUME
2088  484B 365E          2465          LH    R4,IDIRBLK+2(R11)   GET START LRN
208C  2641               2466          AIS   R4,1                PLUS 1 IS WHERE PROGRAM STARTS
208E  4040 1A2A          2467          STH   R4,THISLRN          SAVE INPUT FLOPPY LRN
2092  2741               2468          SIS   R4,1                POINT BACK TO PDB
2094  C850 1AFC          2469          LDAI  R5,SEQNUM
2098  C865 007F          2470          LHI   R6,127(R5)
209C  41E0 2CA0          2471          BAL   R14,RDLRN           READ THE PDB
20A0  4300 2186          2472          B     SIZEIT
                         2473  *
                         2474  *
                         2475  *
20A4  48F0 1A50          2476  TRYTHIS LH    R15,OUTFLAG         WHAT IS THE OUTPUT?
20A8  C5F0 0002          2477          CLHI  R15,2               IF FLOPPY, SELECTIVE OUTPUT FOR
*20AC 213E               2478          BNE   THISONE             EACH DISKETTE, ELSE DO ALL
20AE  03D0 3777          2479          LB    R13,PDB+27          PICK OUT THE FLAG DIGIT
20B2  C4D0 000F          2480          NHI   R13,X'0F'           ZERO MEANS THIS PROGRAM
20B6  4330 1F7C          2481          BZ    FINDNEXT            GOES ON NO DISKETTE.  A
                         2482  *                                 VALUE OF X'F' MEANS IT GOES ON
20BA  C5D0 000F          2483          CLHI  R13,X'0F'           EVERY DISKETTE.  OTHER VALUES
20BE  2335               2484          BES   THISONE             MUST MATCH THE VOLUME OPTION.
20C0  45D0 16AC          2485          CLH   R13,VOLUME+$VALU1
20C4  4230 1F7C          2486          BNE   FINDNEXT            TRY NEXT IF NO MATCH
```

CONVERT PDB TO FLOPPY FORMAT

```
20C8  03D0 376E  2488 THISONE  LB    R13,PDB+18        PICK UP START ADDRESS
20CC  40D0 1802  2489          STH   R13,LOW           BITS 8:15
20D0  03D0 376F  2490          LB    R13,PDB+19
20D4  02D0 1804  2491          STB   R13,LOW+2         BITS 16:23
20D8  03D0 3770  2492          LB    R13,PDB+20
20DC  C4D0 00FE  2493          NHI   R13,X'FE'
20E0  02D0 1805  2494          STB   R13,LOW+3         BITS 24:31
20E4  03D0 3771  2495          LB    R13,PDB+21        PICK UP END ADDRESS
20E8  40D0 1806  2496          STH   R13,HIGH          BITS 8:15
20EC  48D0 3772  2497          LH    R13,PDB+22
20F0  40D0 1808  2498          STH   R13,HIGH+2        BITS 16:31
20F4  C830 375C  2499          LDAI  R3,PDB            CONVERT TO FLOPPY TYPE PDB
20F8  2423       2500          LIS   R2,3              3 DIGITS
20FA  41D0 2C52  2501          BAL   R13,PACK          CONVERT SEQUENCE NUMBER
20FE  03D0 3777  2502          LB    R13,PDB+27        IF FLOPPY OUTPUT, TEST PDB FLAG
2102  40D0 182A  2503          STH   R13,FMDFLAG       STORE IN FLOPPY PDB
2106  4820 1A50  2504          LH    R2,OUTFLAG        CHECK THE OUTPUT DEVICE
210A  2722       2505          SIS   R2,2              IS IT A FLOPPY?
*210C 2138       2506          BNZ   SEQADJ            IF NO, DON'T ADJUST SEQUENCE
210E  C4D0 00OF  2507          NHI   R13,X'0F'         LS DIGIT IDENTIFIES TARGET VOLUME
2112  C5D0 00OF  2508          CLHI  R13,X'0F'         'F' MEANS EVERY DISKETTE
2116  2133       2509          BNES  SEQADJ            SKIP IF NOT ONE OF THE FDLG'S
2118  CB40 0020  2510          SHI   R4,X'20'          ADJUST FDLG SEQUENCE NUMBER
211C  4040 1AFC  2511 SEQADJ   STH   R4,SEQNUM         SET FLOPPY SEQUENCE NUMBER
2120  C830 375F  2512          LDAI  R3,PDB+3          POINT TO 06- NUMBER
2124  2423       2513          LIS   R2,3              THREE DIGITS
2126  41D0 2C52  2514          BAL   R13,PACK          PACK 06 PART NUMBER
212A  4040 1AFE  2515          STH   R4,PARTNO         STORE IN FLOPPY PDB
212E  C830 3762  2516          LDAI  R3,PDB+6          POINT TO REV LEVEL
2132  2422       2517          LIS   R2,2              2 DIGITS
2134  41D0 2C52  2518          BAL   R13,PACK          PACK REV LEVEL
2138  4040 1800  2519          STH   R4,REVLEV         STORE IN PDB
213C  0340 3775  2520          LB    R4,PDB+25         LOOK FOR POINT
2140  C540 002E  2521          CLHI  R4,C'.'
2144  213A       2522          BNES  NOPOINT           NO REV EXTENSION
2146  0340 3776  2523          LB    R4,PDB+26         ELSE GET EXTENSION DIGIT
214A  C440 000F  2524          NHI   R4,X'F'           MASK SINGLE HEX DIGIT
214E  914C       2525          SLLS  R4,12             POSITION OVER MS DIGIT OF REV
2150  4640 1800  2526          OH    R4,REVLEV         COMBINE
2154  4040 1800  2527          STH   R4,REVLEV         STORE IN FLOPPY PDB
2158  4840 3778  2528 NOPOINT  LH    R4,PDB+28         GET PACKAGE REV
215C  4040 182C  2529          STH   R4,PKGREV         STORE IN FLOPPY PDB
2160  C830 3764  2530          LHI   R3,PDB+8          POINT TO NAME FIELD
2164  2440       2531          LIS   R4,0              INDEX
2166  4803 0000  2532 NAMELOOP LH    R0,0(R3)          COPY 30 BYTE NAME FIELD
216A  4004 180C  2533          STH   R0,FFORM(R4)      INTO FLOPPY PDB
216E  2632       2534          AIS   R3,2              BUMP POINTERS
2170  2642       2535          AIS   R4,2
2172  C540 000A  2536          CLHI  R4,10             FIRST 10 BYTES?
2176  2088       2537          BLS   NAMELOOP          LOOP, NOT YET
2178  2134       2538          BNES  NAMELP01          BRANCH IF PASSED IT
217A  C830 377A  2539          LHI   R3,PDB+30         NAME IS SPLIT IN PDB
217E  220C       2540          BS    NAMELOOP          POINT TO LAST PIECE OF NAME FIELD
```

CONVERT PDB TO FLOPPY FORMAT

```
  2180   C540 001E          2541  NAMELP01 CLHI  R4,30          DONE 30 BYTES?
* 2184   208F               2542           BL    NAMELOOP       LOOP IF NO
```

CALCULATE SIZE OF INPUT PROGRAM

```
2186  48D0 1808  2544  SIZEIT   LH    R13,HIGH+2
218A  48D0 1804  2545           SH    R13,LOW+2          END MINUS START
218E  40D0 1AFA  2546           STH   R13,SIZFI+2
2192  2389       2547           BNCS  SIZEIT2
2194  48D0 1806  2548           LH    R13,HIGH
2198  2701       2549           SIS   R13,1
219A  48D0 1802  2550  SIZEIT1  SH    R13,LOW
219E  40D0 1AF8  2551           STH   R13,SIZFI
21A2  2304       2552           BS    SIZEIT3
21A4  48D0 1806  2553  SIZEIT2  LH    R13,HIGH
21A8  2207       2554           BS    SIZEIT1
21AA  24D1       2555  SIZEIT3  LIS   R13,1
21AC  61D0 1AFA  2556           AHM   R13,SIZFI+2
21B0  2383       2557           BNCS  SIZEIT4            SKIP IF NO CARRY
21B2  61D0 1AF8  2558           AHM   R13,SIZEI          PROPOGATE
21B6  48E0 1AF8  2559  SIZEIT4  LH    R14,SIZFI          R14,R15=SIZE OF INPUT PROGRAM
21BA  48F0 1AFA  2560           LH    R15,SIZFI+2
21BE  D2F0 1882  2561           STB   R15,LEFTOVER       SAVE MODULO 256 REMAINDER
21C2  932F       2562           LBR   R2,R15             R2 HOLDS 2**7 BIT
21C4  92EF       2563           STBR  R14,R15            R15 = SIZEI(16:23),(8:15)
21C6  943F       2564           EXBR  R3,R15             R3 = SIZEI(8:15),(16:23)
                 2565  *                                 SIZEI HAS BEEN SHIFTED RIGHT
                 2566  *                                 8 PLACES (LS 24 BITS ONLY)
21C8  4030 187E  2567           STH   R3,RECORDS         NO OF 256 BYTE RECORDS
21CC  4030 1880  2568           STH   R3,RECORDSV        SAVE FOR VERIFY PHASE
21D0  0A33       2569           AAR   R3,R3              SHIFT BACK LEFT ONE PLACE
21D2  9027       2570           SRLS  R2,7               POSITION MISSING BIT
21D4  0A32       2571           AAR   R3,R2              ADD IT BACK IN
21D6  D320 1882  2572           LB    R2,LEFTOVER        R3 = NO OF 128 BYTE RECORDS
21DA  C320 007F  2573           THI   R2,X'7F'           TEST MODULO 128 REMAINDER
21DE  2332       2574           BZS   EVEN               SKIP IF REMAINDER ZERO
21E0  2631       2575           AIS   R3,1               ROUND QUOTIENT UP
21E2  4030 180A  2576  EVEN     STH   R3,LRNS            SAVE NUMBER OF LOGICAL RECORDS
```

OUTPUT PDB DATA TO LIST DEVICE

```
21E6  C800 2020          2578 SHOWPDB   LHI   R0,X'2020'
21EA  C850 0046          2579           LHI   R5,70
21EE  4005 3550          2580 SHOWPDB0  STH   R0,PRINTOUT(R5)     BLANK OUT PRINT BUFFER
21F2  2752               2581           SIS   R5,2
21F4  2283               2582           BNLS  SHOWPDB0
21F6  4810 1AFC          2583           LH    R1,SEQNUM          GET FLOPPY SEQUENCE NUMBER
21FA  4800 1A66          2584           LH    R0,OSFLAG          DOING ONE OF THE O.S.'S?
21FE  4330 223E          2585           BZ    SHOWPDB1           SKIP IF NO
2202  C800 3136          2586           LHI   R0,C'16'
2206  2711               2587           SIS   R1,1               WHICH ONE?
2208  2333               2588           BZS   SHOW.OS1           THE FIRST ONE
220A  C800 3332          2589           LHI   R0,C'32'           THE SECOND ONE
220E  C810 4F53          2590 SHOW.OS1  LHI   R1,C'OS'           FIRST IS "OS16MDL2.111"
2212  4010 3554          2591           STH   R1,PRINTOUT+4      SECOND IS "OS32MDL2.111"
2216  4000 3556          2592           STH   R0,PRINTOUT+6      STORE "OS16" OR "OS32"
221A  4800 16D0          2593           LH    R0,OSID+$VALU1     GET OPTIONAL CHARACTERS
221E  4810 16D2          2594           LH    R1,OSID+$VALU2
2222  4000 3558          2595           STH   R0,PRINTOUT+8      STORE "MDL2"
2226  4010 355A          2596           STH   R1,PRINTOUT+10
222A  C800 2E31          2597           LHI   R0,C'.1'
222E  4000 355C          2598           STH   R0,PRINTOUT+12     STORE ".111"
2232  D200 355E          2599           STB   R0,PRINTOUT+14
2236  D200 355F          2600           STB   R0,PRINTOUT+15
223A  4300 22AC          2601           B     SHOWPDB2           GO DO TITLE FIELD
223E  C820 3550          2602 SHOWPDB1  LDAI  R2,PRINTOUT        WHERE SEQUENCE NUMBER GOES
2242  2403               2603           LIS   R0,3               THREE DIGITS
2244  41F0 0F18          2604           BAL   R15,HEXASC         CONVERT HEX TO ASCII
2248  C840 0030          2605           LHI   R4,C'0'
224C  D240 3555          2606           STB   R4,PRINTOUT+5      FILL IN CHARACTERS 06-
2250  C840 362D          2607           LHI   R4,C'6-'
2254  4040 3556          2608           STH   R4,PRINTOUT+6
2258  4810 1AFE          2609           LH    R1,PARTNO          GET 06- PART NUMBER
225C  C820 3558          2610           LDAI  R2,PRINTOUT+8      WHERE IT GOES
2260  2403               2611           LIS   R0,3               3 DIGITS
2262  41F0 0F18          2612           BAL   R15,HEXASC         CONVERT
2266  4840 1B2A          2613           LH    R4,FMDFLAG         GO FOR F LEVEL
226A  9044               2614           SRLS  R4,4               POSITION FIELD
226C  2339               2615           BZS   NOFVARI            SKIP IF ZERO
226E  C640 3030          2616           OHI   R4,C'00'           CONVERT TO ASCII
2272  4040 355C          2617           STH   R4,PRINTOUT+12     STORE IT
2276  C840 0046          2618           LHI   R4,C'F'
227A  D240 355B          2619           STB   R4,PRINTOUT+11
227E  C840 0052          2620 NOFVARI   LHI   R4,C'R'
2282  D240 355E          2621           STB   R4,PRINTOUT+14     PUT CHARACTER R IN OUTPUT
2286  4810 1B00          2622           LH    R1,REVLEV          GET REV LEVEL
228A  C820 355F          2623           LDAI  R2,PRINTOUT+15     WHERE IT GOES
228E  2402               2624           LIS   R0,2               2 DIGITS
2290  41F0 0F18          2625           BAL   R15,HEXASC         CONVERT
2294  D310 1B00          2626           LB    R1,REVLEV
2298  9014               2627           SRLS  R1,4               LOOK FOR REV EXTENSION
229A  2339               2628           BZS   SHOWPDB2           SKIP IF NONE
229C  D311 157A          2629           LB    R1,HEXTAB(R1)      CONVERT TO ASCII
22A0  C840 002E          2630           LHI   R4,C'.'
```

OUTPUT PDB DATA TO LIST DEVICE

```
22A4  D240 3561          2631           STB   R4,PRINTOUT+17      PUT POINT IN THE OUTPUT
22A8  D210 3562          2632           STB   R1,PRINTOUT+18
22AC  4850 1B2C          2633  SHOWPDB2 LH    R5,PKGREV           TEST FOR PACKAGE REV
22B0  2334               2634           BZS   SHOWPDB3            SKIP IF ZERO
22B2  4050 3564          2635           STH   R5,PRINTOUT+20
22B6  2450               2636           LIS   R5,0
22B8  4845 180C          2637  SHOWPDB3 LH    R4,FFORM(R5)        COPY FROM FLOPPY PDB
22BC  4045 3568          2638           STH   R4,PRINTOUT+24(R5)
22C0  2652               2639           AIS   R5,2
22C2  C550 001E          2640           CLHI  R5,30
22C6  2087               2641           BLS   SHOWPDB3            LOOP THRU 30 BYTES
                         2642  *                                  COPY LOW ADDRESS
22C8  2402               2643           LIS   R0,2                MS 2 DIGITS
22CA  4810 1B02          2644           LH    R1,LOW
22CE  C820 3588          2645           LHI   R2,PRINTOUT+56
22D2  41F0 0F18          2646           BAL   R15,HEXASC
22D6  2404               2647           LIS   R0,4                LS 4 DIGITS
22D8  4810 1B04          2648           LH    R1,LOW+2
22DC  2622               2649           AIS   R2,2
22DE  41F0 0F18          2650           BAL   R15,HEXASC
22E2  2402               2651           LIS   R0,2
22E4  4810 1B06          2652           LH    R1,HIGH
22E8  C820 358F          2653           LHI   R2,PRINTOUT+63
22EC  41F0 0F18          2654           BAL   R15,HEXASC
22F0  2404               2655           LIS   R0,4
22F2  4810 1B08          2656           LH    R1,HIGH+2
22F6  2622               2657           AIS   R2,2
22F8  41F0 0F18          2658           BAL   R15,HEXASC
22FC  4840 1B2A          2659           LH    R4,FMDFLAG          GET FMD FLAG
2300  C440 000F          2660           NHI   R4,X'F'             LS DIGIT = FLOPPY NUMBER
2304  D344 157A          2661           LB    R4,HEXTAB(R4)       CONVERT TO ASCII
2308  9148               2662           SLLS  R4,8
230A  264D               2663           AIS   R4,X'0D'            ADD A CARRIAGE RETURN
230C  4040 3596          2664           STH   R4,PRINTOUT+70      STORE IN PRINT BUFFER
2310  C840 0A00          2665           LHI   R4,X'0A00'          LINE FEED & ZERO
2314  4040 3598          2666           STH   R4,PRINTOUT+72      MARK END OF MESSAGE
2318  4800 1A66          2667           LH    R0,OSFLAG           BUILDING TET IMAGE?
*231C 2339               2668           BZ    SHOWPDB4            SKIP IF NO
231E  C800 3030          2669           LHI   R0,C'00'            START ADDRESS IS ACTUALLY ZERO
2322  4000 3588          2670           STH   R0,PRINTOUT+56
2326  4000 358A          2671           STH   R0,PRINTOUT+58
232A  4000 358C          2672           STH   R0,PRINTOUT+60
232E  48F0 1A16          2673  SHOWPDB4 LH    R15,NOSTOP          TEST IF BUILD PHASE OF BUILDV
2332  2338               2674           BZS   SHOWPDB5            IF NO, OUTPUT TO LIST DEVICE
2334  C840 0D00          2675           LHI   R4,X'0D00'          ELSE, OUTPUT JUST THE
2338  4040 3554          2676           STH   R4,PRINTOUT+4       SEQUENCE NUMBER TO THE
233C  41F0 12BC          2677           BAL   R15,SETKB           CONSOLE DEVICE
*2340 230E               2678           B     SHOWPDB6
2342  0300 155B          2679  SHOWPDB5 LB    R0,IOSAVE+1         GET LIST DEVICE ID                *
2346  2703               2680           SIS   R0,3                TEST IF LINE PRINTER              *
*2348 213A               2681           BNZ   SHOWPDB6            SKIP IF NO                        *
234A  4800 16DE          2682           LH    R0,$LINCNT          IF YES, TEST IF PAGE FULL         *
*234E 2137               2683           BNZ   SHOWPDB6            SKIP IF NO                        *
```

OUTPUT PDB DATA TO LIST DEVICE

```
2350   41F0 0FBA          2684            BAL   R15,$PRINT        IF YES, ISSUE FORM FEED          *
2354   3500               2685            DAC   HEADER            FOLLOWED BY HEADER MESSAGE       *
2356   24F2               2686            LIS   R15,2
2358   40F0 16DE          2687            STH   R15,$LINCNT       PRESET LINE COUNT
235C   41F0 0FBA          2688  SHOWPDB6  BAL   R15,$PRINT        PRINT LISTING
2360   3550               2689            DAC   PRINTOUT
```

POSITION OUTPUT MEDIA & TRANSFER PDB

```
2362  2420                2691  POSITION LIS   R2,0
2364  4020 1B8A           2692           STH   R2,UTILITY        CLEAR UTILITY FLAG
2368  4820 167C           2693           LH    R2,OUTDEV+$VALU1  OUTPUT DEVICE NUMBER
236C  4830 1A50           2694           LH    R3,OUTFLAG        LOOK AT OUTPUT DEVICE
2370  2731                2695           SIS   R3,1              MAG TAPE?
2372  4330 2428           2696           BZ    OUTMAG            BRANCH IF YES
2376  2731                2697           SIS   R3,1              FLOPPY?
2378  4230 246E           2698           BNZ   OUTDISK           NO, A DISK
237C  4830 1A22           2699           LH    R3,FMDCMDO
2380  4030 1A24           2700           STH   R3,FMDCMD         SET UP FOR OUTPUT DISKETTE
2384  41E0 2D48           2701           BAL   R14,RDIRECT       READ FLOPPY DIRECTORY
                          2702  *                                ON RETURN, R3 CONTAINS EOV FLAG
                          2703  *                                OR A MATCHING SEQUENCE NUMBER.
                          2704  *                                R5 IS INDEX TO DIRECTORY BLOCK.
2388  4530 1AFC           2705           CLH   R3,SEQNUM         SEQUENCE NUMBER CHECK
238C  4330 23F8           2706           BE    THERE1            DUPLICATE SEQUENCE NUMBER!!!
2390  48F0 1A18           2707           LH    R15,VERIFLAG      TEST IF DOING A VERIFY
2394  4230 262E           2708           BNZ   MISSING           NON PRESENT SEQUENCE!
2398  0833                2709           LDAR  R3,R3
239A  4230 2420           2710           BNZ   EOVRTN            EOV NOT FOUND!!
239E  0855                2711  THERE1A  LDAR  R5,R5             FIRST ENTRY IN THE BLOCK?
* 23A0  2336               2712           BZ    FIRST             BRANCH IF YES
23A2  4845 385A           2713           LH    R4,INBUF-2(R5)    LOAD PREVIOUS PDB
23A6  41C0 2ABC           2714           BAL   R12,SULH56
* 23AA  230F               2715           B     PRIOR
23AC  D4A0 1A11           2716  FIRST    CLB   R10,STOIR         FIRST BLOCK?
23B0  4330 23D6           2717           BE    FRST.BLK          SKIP IF YES
23B4  084A                2718           LDAR  R4,R10
23B6  2741                2719           SIS   R4,1              DECREMENT BLOCK POINTER
23B8  41C0 2ABC           2720           BAL   R12,SULH56
23BC  41E0 2CA0           2721           BAL   R14,RDLRN         READ PREVIOUS DIRECTORY BLOCK
23C0  4840 38DA           2722           LH    R4,INBUF+126      LAST ENTRY IN THAT BLOCK
23C4  41C0 2ABC           2723           BAL   R12,SULH56
23C8  41E0 2CA0           2724  PRIOR    BAL   R14,RDLRN         READ PDB OF LAST PROGRAM
23CC  4850 386A           2725           LH    R5,INBUF+14       GET SIZE OF LAST PROGRAM
23D0  0A54                2726           AAR   R5,R4             ADD START LRN OF LAST PROGRAM
23D2  2651                2727           AIS   R5,1              PLUS 1 FOR THE PDB
23D4  2303                2728           BS    NEWSTART
23D6  D350 1A13           2729  FRST.BLK LB    R5,STSAV          START LRN FOR FIRST PROGRAM
23DA  4050 1A28           2730  NEWSTART STH   R5,NEXTLRN        = START LRN FOR THIS PROGRAM
23DE  4050 1B88           2731           STH   R5,PDBSTART       WHERE THE PDB STARTS
23E2  2651                2732           AIS   R5,1              PLUS 1 FOR THE PDB
23E4  4A50 1B0A           2733           AH    R5,LRNS           PLUS SIZE OF THIS PROGRAM
23E8  4550 1A26           2734           CLH   R5,MAXLRN         COMPARE TO CEILING
23EC  4280 263A           2735           BL    DISKETTE          GO ON, OK!
                          2736  *                                ELSE, TOO BIG
23F0  C850 3364           2737           LDAI  R5,TBIGMSG        DISKETTE FULL MESSAGE TO CONSOLE
23F4  4300 1C50           2738           B     CONMSG1           DEVICE...RETURN TO OPTIN.
                          2739  *
23F8  48C0 1A18           2740  THERE1   LH    R12,VERIFLAG      TEST VERIFY FLAG
23FC  4230 239E           2741           BNZ   THERE1A           RETURN IF SET
2400  230C                2742           BS    THERE             ELSE ERROR
2402  48C0 1A18           2743  THERE2   LH    R12,VERIFLAG      TEST VERIFY FLAG
```

POSITION OUTPUT MEDIA & TRANSFER PDB

```
2406  2339                2744            BZS   THERE              ERROR IF NOT VERIFY
2408  4885 3860           2745            LH    R8,INBUF+4(R5)     GET CYLINDER NUMBER
240C  D395 3863           2746            LB    R9,INBUF+7(R5)     GET HEAD NUMBER
2410  D3A5 3862           2747            LB    R10,INBUF+6(R5)    GET SECTOR NUMBER
2414  4300 25D6           2748            B     THERE2A
                          2749  *
2418  C850 3385           2750  THERE     LDAI  R5,GOTMSG          DUPLICATE SEQUENCE NUMBER
241C  4300 1C50           2751            B     CONMSG1
                          2752  *
2420  C850 3354           2753  EOVRTN    LDAI  R5,EOVRTNM         EOV NOT FOUND
2424  4300 1C50           2754            B     CONMSG1
                          2755  *
2428  4840 1688           2756  OUTMAG    LH    R4,SELCH2+$VALU1   PICK UP SELCH ADDRESS
242C  41F0 2EBC           2757            BAL   R15,NOMOTION       WAIT FOR NOMOTION
2430  4870 1A18           2758            LH    R7,VERIFLAG        TEST VERIFY FLAG
*2434  213B                2759            BNZ   VMAGPDB            BRANCH IF SET
2436  C850 375C           2760            LDAI  R5,PDB             START ADDRESS
243A  C865 0032           2761            LHI   R6,50(R5)          END ADDRESS
243E  DE20 1A3A           2762            OC    R2,BKSP            BACK SPACE OVER FILE MARK
2442  41D0 2B62           2763            BAL   R13,WRITEMAG       WRITE THE PDB
2446  4300 2670           2764            B     DUPEPROG           DO REST OF PROGRAM
                          2765  *
244A  DE20 1A2F           2766  VMAGPDB   OC    R2,FORWARD         SKIP FORWARD PASSED FILE MARK
244E  C850 385C           2767            LDAI  R5,INBUF
2452  C865 0032           2768            LHI   R6,50(R5)
2456  41D0 2B6A           2769            BAL   R13,READMAG        READ THE PDB
245A  C850 375C           2770            LDAI  R5,PDB             PATTERN START ADDRESS
245E  C860 0032           2771            LHI   R6,50              NUMBER OF BYTES
2462  41E0 2EA4           2772            BAL   R14,COMPARE        COMPARE PATTERN TO INBUF
2466  41F0 2624           2773            BAL   R15,VERR.002       "PDB VERIFY ERROR"
246A  4300 2670           2774            B     DUPEPROG
                          2775  *
                          2776  *
246E  4830 167E           2777  OUTDISK   LH    R3,OUTDEV+$VALU2   CONTROLLER ADDRESS
2472  4840 1688           2778            LH    R4,SELCH2+$VALU1   SELCH ADDRESS
2476  4870 1A6C           2779            LH    R7,DISKTYP2
247A  4070 1A68           2780            STH   R7,DISKTYPE
247E  4870 1A66           2781            LH    R7,OSFLAG          TEST IF BUILDING OS
2482  4330 25B0           2782            BZ    OUTDISK1           BRANCH IF NO
                          2783  *                                  OSFLAG WILL BE SET WHILE WE'RE
                          2784  *                                  COPYING LIBRARY SEQUENCE NUMBERS
                          2785  *                                  001 AND 002 TO THE OUTPUT DISK.
2486  41C0 2AC6           2786            BAL   R12,SULH56.2       INSTEAD OF WRITING THE PROGRAM
248A  41C0 2AD0           2787            BAL   R12,ZEROBUFF       DEFINITION BLOCK, OUTPUT DATA
248E  4880 1A1A           2788            LH    R8,NEXTCYL         TO MAKE THIS PROGRAM LOOK LIKE A
2492  4890 1A1C           2789            LH    R9,NEXTHEAD        TET'ED O.S.
2496  48A0 1A1E           2790            LH    R10,NEXTSECT       STARTING CYLINDER, HEAD, SECTOR
249A  41F0 2FA8           2791            BAL   R15,FORMLBA        CONVERT TO LOGICAL BLOCK ADDRESS
249E  4870 1AFC           2792            LH    R7,SEQNUM          SEQUENCE 001 OR 002
24A2  C470 0002           2793            NHI   R7,2
24A6  0A77                2794            AAR   R7,R7              R7 = 0 OR 4
24A8  4007 1A7A           2795            STH   R0,STARTLBA(R7)    SAVE FOR DIRECTORY
24AC  4017 1A7C           2796            STH   R1,STARTLBA+2(R7)
```

POSITION OUTPUT MEDIA & TRANSFER PDB

```
24B0  4870 1AFC          2797            LH    R7,SEQNUM          GET SEQUENCE NUMBER AGAIN
24B4  2771               2798            SIS   R7,1               R7 = 0 OR 1
                         2799  *                                  0=FIRST O.S.  1=SECOND O.S.
24B6  2332               2800            BZS   TET.001            SKIP IF DOING THE FIRST O.S.
24B8  2571               2801            LCS   R7,1               FOR THE SECOND O.S. ONLY, THERE
24BA  4070 1A8A          2802  TET.001   STH   R7,BLOCK           WILL BE A "LIB" RECORD
24BE  4310 24F6          2803            BNM   TET.002            BRANCH IF FIRST O.S.
24C2  C8E0 0101          2804            LHI   R14,X'0101'
24C6  40E0 385C          2805            STH   R14,INBUF          TYPE + NO. OF LIB RECORDS
24CA  D2E0 387A          2806            STB   R14,INBUF+30
24CE  D3E0 1B08          2807            LB    R14,HIGH+2         MS END ADDRESS = XX
24D2  D2E0 38C2          2808            STB   R14,INBUF+102      CTOP = XXFE
24D6  26E1               2809            AIS   R14,1              R14 = YY
24D8  40E0 386A          2810            STH   R14,INBUF+14       SECTOR COUNT = 00YY
24DC  D2E0 38C6          2811            STB   R14,INBUF+106      UTOP = YY00
24E0  25E2               2812            LCS   R14,2
24E2  D2E0 38C3          2813            STB   R14,INBUF+103
24E6  25E1               2814            LCS   R14,1
24E8  40E0 3872          2815            STH   R14,INBUF+22       MXSP
24EC  24EF               2816            LIS   R14,X'F'           MS DIGIT OF MXSP
24EE  D2E0 3871          2817            STB   R14,INBUF+21
24F2  4300 2546          2818            B     SECTOR             GO WRITE THIS FIRST RECORD
24F6  4870 1AFC          2819  TET.002   LH    R7,SEQNUM          SEQUENCE 001 OR 002
24FA  C470 0002          2820            NHI   R7,2               KEY LOCATIONS:
24FE  48D0 1B04          2821            LH    R13,LOW+2
2502  C8E0 4000          2822            LHI   R14,X'4000'        SEQ 001          SEQ 002
2506  C8F0 2302          2823            LHI   R15,X'2302'        (16 BIT)         (32 BIT)
250A  40F0 38BC          2824            STH   R15,INBUF+X'60'    0060 = 2302      0060 = 2302
250E  C8F0 0100          2825            LHI   R15,X'0100'
2512  40F0 38BE          2826            STH   R15,INBUF+X'62'    0062 = 0100      0062 = 0100
2516  C8F0 4300          2827            LHI   R15,X'4300'
251A  40F0 38C0          2828            STH   R15,INBUF+X'64'    0064 = 4300      0064 = 4300
251E  40E0 38C2          2829            STH   R14,INBUF+X'66'    *                0066 = 4000
2522  40D7 38C2          2830            STH   R13,INBUF+X'66'(R7) 0066 = START    0068 = START
2526  D220 38D6          2831            STB   R2,INBUF+X'7A'     DEVICE NUMBER
252A  D230 38D8          2832            STB   R3,INBUF+X'7C'     CONTROLLER ADDRESS
252E  D240 38D9          2833            STB   R4,INBUF+X'7D'     SELCH ADDRESS
2532  4870 1A6C          2834            LH    R7,DISKTYP2
2536  4877 1AEA          2835            LH    R7,DEVCODES(R7)    GET O.S. DEVICE CODE
253A  D270 38D7          2836            STB   R7,INBUF+X'7B'     STORE DEVICE CODE
253E  C870 0111          2837            LHI   R7,X'0111'
2542  4070 38DA          2838            STH   R7,INBUF+X'7E'     O.S. EXTENSION
                         2839  *
2546  4870 1A18          2840  SECTOR    LH    R7,VERIFLAG        TEST VERIFY FLAG
254A  4230 2584          2841            BNZ   VERI.OS            SKIP IF SET
254E  41F0 2F3C          2842            BAL   R15,WNXTSECT       WRITE 256 BYTES
2552  4080 1A1A          2843  SECTOROK  STH   R8,NEXTCYL         SAVE POINTERS
2556  4090 1A1C          2844            STH   R9,NEXTHEAD
255A  40A0 1A1E          2845            STH   R10,NEXTSECT
255E  4870 1A8A          2846            LH    R7,BLOCK
2562  2671               2847            AIS   R7,1               INCREMENT SECTOR COUNT
2564  4070 385C          2848            STH   R7,INBUF
2568  4070 1A8A          2849            STH   R7,BLOCK
```

POSITION OUTPUT MEDIA & TRANSFER PDB

```
  256C  4330 24F6                      BZ    TET.002          DONE LIB. DO FIRST DATA RECORD
  2570  D470 1B04                      CLB   R7,LOW+2         AT REAL START ADDRESS YET?
  2574  4380 2670                      BNL   DUPEPROG         YES. PUT OUT THE PROGRAM IMAGE
  2578  41C0 2AC6                      BAL   R12,SULH56.2     NO. DO 256 MORE
  257C  41C0 2AD0                      BAL   R12,ZEROBUFF
  2580  4300 2546                      B     SECTOR
  2584  C850 3C5C            VERI.OS   LDAI  R5,CHKBUF        NEW START
  2588  C865 00FF                      LHI   R6,255(R5)
  258C  41F0 2F38                      BAL   R15,RNXTSECT     READ A SECTOR
  2590  41E0 2E9C                      BAL   R14,COMPAREX     COMPARE TO INBUF
  2594  41F0 259C                      BAL   R15,OSERROR      "O.S. IMAGE VERIFY ERROR"
  2598  4300 2552                      B     SECTOROK         BRANCH IF MATCH
                             *
  259C  033F                 OSERROR   BER   R15              RETURN IF GOOD COMPARE
  259E  4850 1B8A            OSERROR1  LH    R5,UTILITY       TEST UTILITY FLAG
  25A2  023F                           BNZR  R15              RETURN IF ALREADY SET
  25A4  C850 3411                      LDAI  R5,OSERRM        O.S. IMAGE VERIFY ERROR
  25A8  40F0 1B8A                      STH   R15,UTILITY      SET UTILITY FLAG
  25AC  4300 1C58                      B     CONMSG           RETURN FROM CONMSG ON R15
                             *
                             *
  25B0  41F0 2ECC            OUTDISK1  BAL   R15,DIRSERCH     READ THE DISK DIRECTORY
  25B4  4560 1AFC                      CLH   R6,SEQNUM        COMPARE TO SEQUENCE NUMBER
  25B8  4330 2402                      BE    THERE2           DUPLICATE
  25BC  48F0 1A18                      LH    R15,VERIFLAG     TEST IF DOING A VERIFY
  25C0  4230 262E                      BNZ   MISSING          IF YES. NO SEQ NUM MATCH
  25C4  0866                           LDAR  R6,R6
  25C6  4230 2420                      BNZ   EOVRTN
  25CA  4880 1A1A                      LH    R8,NEXTCYL       WHERE THIS ONE GOES
  25CE  4890 1A1C                      LH    R9,NEXTHEAD
  25D2  48A0 1A1E                      LH    R10,NEXTSECT
  25D6  4080 1A74            THERE2A   STH   R8,DIRDAT        SAVE INFO FOR DIRECTORY UPDATE
  25DA  4090 1A76                      STH   R9,DIRDAT+2
  25DE  40A0 1A78                      STH   R10,DIRDAT+4
  25E2  C850 375C                      LDAI  R5,PDB           SET BUFFER LIMITS
  25E6  C865 00FF                      LHI   R6,255(R5)
  25EA  4870 1A18                      LH    R7,VERIFLAG      CHECK VERIFY FLAG
* 25EE  213D                           BNZ   VERIPDB          SKIP IF SET
  25F0  41E0 2DDA                      BAL   R14,WRITSECT     ELSE WRITE THE PDB
  25F4  41F0 2F8A            GO.DISK   BAL   R15,NEXTDISK     BUMP POINTERS
  25F8  4080 1A1A                      STH   R8,NEXTCYL       POINTING TO WHERE THE PROGRAM STARTS
  25FC  4090 1A1C                      STH   R9,NEXTHEAD
  2600  40A0 1A1E                      STH   R10,NEXTSECT
  2604  4300 2670                      B     DUPEPROG         GO DO THE PROGRAM
                             *
  2608  41C0 2AC6            VERIPDB   BAL   R12,SULH56.2
  260C  41E0 2DD6                      BAL   R14,READSECT     READ THE PDB
  2610  C850 375C                      LDAI  R5,PDB           PATTERN START ADDRESS
  2614  C860 0032                      LHI   R6,50            NUMBER OF BYTES
  2618  41E0 2EA4                      BAL   R14,COMPARE      COMPARE PATTERN TO INBUF
  261C  41F0 2624                      BAL   R15,VERR.002     "PDB VERIFY ERROR"
  2620  4300 25F4                      B     GO.DISK          BUMP POINTERS & DO THE PROGRAM
                             *
```

POSITION OUTPUT MEDIA & TRANSFER PDB

```
2624  033F            2903  VERR.002 BER   R15              RETURN IF GOOD COMPARE
2626  C850 33FE       2904           LDAI  R5,VERRMSG2      "PDB VERIFY ERROR"
262A  4300 1C58       2905           B     CONMSG           RETURN FROM CONMSG ON R15
                      2906  *
262E  C850 34DD       2907  MISSING  LDAI  R5,ABSENTM       "NO SUCH SEQUENCE ON OUTPUT MEDIA"
2632  41F0 1C58       2908           BAL   R15,CONMSG       PRINT MESSAGE
2636  4300 1F7C       2909           B     FINDNEXT         LOOK FOR NEXT PROGRAM


263A  4840 1A28       2911  DISKETTE LH    R4,NEXTLRN       START LRN FOR THIS PROGRAM
263E  4850 1A18       2912           LH    R5,VERIFLAG
* 2642  233E          2913           BZ    DISKETT1         BRANCH IF NOT VERIFY
2644  41C0 2ABC       2914           BAL   R12,SULH56
2648  41E0 2CA0       2915           BAL   R14,RDLRN        READ IN THE PDB
264C  C850 1AFC       2916           LDAI  R5,SEQNUM        PATTERN START ADDRESS
2650  C860 002E       2917           LHI   R6,46            NUMBER OF BYTES
2654  41E0 2EA4       2918           BAL   R14,COMPARE
2658  41F0 2624       2919           BAL   R15,VERR.002     PDB VERIFY ERROR
* 265C  2307          2920           B     DISKETT2
265E  C850 1AFC       2921  DISKETT1 LHI   R5,SEQNUM        SET POINTERS TO THE PDB
2662  C865 007F       2922           LHI   R6,127(R5)
2666  41E0 2D00       2923           BAL   R14,WRLRN        WRITE THE PDB
266A  2441            2924  DISKETT2 LIS   R4,1
266C  6140 1A28       2925           AHM   R4,NEXTLRN       BUMP WORKING PDB POINTER
```

TRANSFER ACTUAL PROGRAM

```
2670  C860 03FF          2927  DUPEPROG LHI   R6,1023          SEE IF WE CAN DO 1KB
2674  4850 1B7E          2928           LH    R5,RECORDS       NO. OF 256 BYTE RECORDS TO DO
2678  C550 0004          2929           CLHI  R5,4             ARE THERE 4 OR MORE?
267C  2383               2930           BNLS  GETSOME          YES, DO 4 RECORDS (1024 BYTES)
267E  C860 00FF          2931           LHI   R6,255           IF NO, JUST DO 256 BYTES
2682  C850 385C          2932  GETSOME  LDAI  R5,INBUF         START ADDRESS
2686  4060 35B8          2933           STH   R6,R6SAVE
268A  C865 03FF          2934           LHI   R6,1023(R5)
268E  41C0 2AD0          2935           BAL   R12,ZEROBUFF     CLEAR THE INPUT BUFFER
2692  4860 35B8          2936           LH    R6,R6SAVE
2696  0A65               2937           AAR   R6,R5            FORM FINAL ADDRESS
2698  4820 1664          2938           LH    R2,INDEV+$VALU1
269C  4830 1A4E          2939           LH    R3,INFLAG        LOOK AT INPUT DEVICE CODE
26A0  C530 0001          2940           CLHI  R3,1             MAG TAPE?
* 26A4  2139              2941           BNE   NOTMAG           BRANCH IF NO
26A6  4840 1670          2942           LH    R4,SELCH1+$VALU1 YES MAG TAPE..R4 = SELCH ADDRESS
26AA  41D0 2B6A          2943           BAL   R13,READMAG      READ 1024,256,OR FEWER BYTES
26AE  41C0 2AE6          2944           BAL   R12,HOWMANY      SEE HOW MANY BYTES CAME IN
26B2  4300 2714          2945           B     WRITEIT          GO WRITE TO OUTPUT
26B6  C530 0002          2946  NOTMAG   CLHI  R3,2             FLOPPY?
26BA  4230 26DC          2947           BNE   NOTFMD           BRANCH IF NO
26BE  4830 1A20          2948           LH    R3,FMDCMDI       YES FLOPPY...GET INPUT COMMAND MODEL
26C2  4030 1A24          2949           STH   R3,FMDCMD        COMMAND MODEL FOR INPUT DRIVE
26C6  4840 1A2A          2950           LH    R4,THISLRN       CURRENT INPUT DRIVE POSITION
26CA  41E0 2CA0          2951           BAL   R14,RDLRN        READ 124, 256, OR 1024 BYTES
26CE  41C0 2AE6          2952           BAL   R12,HOWMANY      SEE HOW MANY CAME IN
26D2  9077               2953           SRLS  R7,7             DIVIDED BY 128 = LRNS
26D4  6170 1A2A          2954           AHM   R7,THISLRN       INCREMENT CURRENT POSITION
26D8  4300 2714          2955           B     WRITEIT
26DC  4830 1A6A          2956  NOTFMD   LH    R3,DISKTYP1      YES DISK...GET INPUT DISK TYPE
26E0  4030 1A68          2957           STH   R3,DISKTYPE      SELECT TYPE FOR INPUT DISK
26E4  4830 1666          2958           LH    R3,INDEV+$VALU2  CONTROLLER ADDRESS
26E8  4840 1670          2959           LH    R4,SELCH1+$VALU1 SELCH ADDRESS
26EC  4880 1A60          2960           LH    R8,INCYL         CURRENT INPUT DISK POSITION
26F0  4890 1A62          2961           LH    R9,INHEAD
26F4  48A0 1A64          2962           LH    R10,INSECT
26F8  41F0 2F38          2963  GET256   BAL   R15,RNXTSECT     READ 256 BYTES AT A TIME
26FC  CA50 0100          2964           AHI   R5,256           BUMP START ADDRESS
2700  0556               2965           CLAR  R5,R6            FINISHED?
2702  2085               2966           BLS   GET256           READ ANOTHER SECTOR IF NO
2704  4080 1A60          2967           STH   R8,INCYL         UPDATE CURRENT DISK POSITION
2708  4090 1A62          2968           STH   R9,INHEAD
270C  40A0 1A64          2969           STH   R10,INSECT
2710  41C0 2AE6          2970           BAL   R12,HOWMANY      SEE HOW MANY WE READ
                         2971  *
2714  4860 1B86          2972  WRITEIT  LH    R6,PROGSIZE      NUMBER OF BYTES MODULO 128
2718  2761               2973           SIS   R6,1             MINUS 1 FOR END ADDRESS
271A  C850 385C          2974           LDAI  R5,INBUF         R5 = START ADDRESS
271E  0A65               2975           AAR   R6,R5            R6 = END ADDRESS
2720  4820 167C          2976           LH    R2,OUTDEV+$VALU1 R2 GETS OUTPUT DEVICE NUMBER
2724  4830 1A50          2977           LH    R3,OUTFLAG       CHECK OUTPUT DEVICE CODE
2728  2731               2978           SIS   R3,1             MAG TAPE?
272A  4330 279C          2979           BZ    COPYMAG
```

TRANSFER ACTUAL PROGRAM

```
272E  2731                2980            SIS   R3,1          FLOPPY OUTPUT DEVICE?
2730  4230 278E           2981            BNZ   DISKOUT       NO, HAS TO BE A DISK
2734  4840 1A28           2982            LH    R4,NEXTLRN    CURRENT OUTPUT DRIVE POSITION
2738  4830 1A22           2983            LH    R3,FMDCMD0
273C  4030 1A24           2984            STH   R3,FMDCMD     COMMAND MODEL FOR OUTPUT DRIVE
2740  4830 1A18           2985            LH    R3,VERIFLAG
*2744 213B                2986            BNZ   VERIFMD2      BRANCH IF VERIFY COMMAND
2746  41E0 2D00           2987            BAL   R14,WRLRN     WRITE 128,256 OR 1024
274A  4860 1B86           2988            LH    R6,PROGSIZE   R6= NUMBER OF BYTES WRITTEN
274E  9067                2989            SRLS  R6,7          CHANGE TO NO OF LRN'S
2750  0A46                2990            AAR   R4,R6         INCREMENT WORKING LRN NUMBER
2752  4040 1A28           2991            STH   R4,NEXTLRN
2756  4300 281E           2992            B     OUTTEST
                          2993  *
275A  0B65                2994  VERIFMD2  SAR   R6,R5         NUMBER OF BYTES MINUS 1
275C  C850 3C5C           2995            LDAI  R5,CHKBUF     NEW START ADDRESS
2760  0A65                2996            AAR   R6,R5         NEW END ADDRESS
2762  41E0 2CA0           2997            BAL   R14,RDLRN     READ LRNS INTO CHECK BUFFER
2766  4860 1B86           2998            LH    R6,PROGSIZE   R6= NUMBER OF BYTES WRITTEN
276A  9067                2999            SRLS  R6,7          CHANGE TO NO OF LRN'S
276C  0A46                3000            AAR   R4,R6         INCREMENT WORKING LRN NUMBER
276E  4040 1A28           3001            STH   R4,NEXTLRN
2772  4860 1B86           3002  VERIFY01  LH    R6,PROGSIZE   NO. OF BYTES THIS TIME
2776  C8F0 281E           3003            LDAI  R15,OUTTEST   SET RETURN POINTER
277A  41E0 2EA0           3004            BAL   R14,COMPAREY  COMPARE INBUF:CHKBUF
277E  4330 281E           3005            BE    OUTTEST       GO SEE IF FINISHED
2782  4870 1A66           3006            LH    R7,OSFLAG
2786  4230 259E           3007            BNZ   OSERROR1      "O.S. IMAGE VERIFY ERROR"
278A  4850 1B8A           3008            LH    R5,UTILITY
278E  023F                3009            BNZR  R15
2790  C850 33AD           3010            LDAI  R5,CHKERRM    "VERIFY ERROR"
2794  4050 1B8A           3011            STH   R5,UTILITY
2798  4300 1C58           3012            B     CONMSG        RETURN FROM CONMSG ON R15
                          3013  *
279C  4840 1688           3014  COPYMAG   LH    R4,SELCH2+$VALU1
27A0  4870 1A18           3015            LH    R7,VERIFLAG
*27A4 2135                3016            BNZ   VERIMAG2
27A6  41D0 2B62           3017            BAL   R13,WRITEMAG
27AA  4300 281E           3018            B     OUTTEST
27AE  0B65                3019  VERIMAG2  SAR   R6,R5         BYTE COUNT
27B0  C850 3C5C           3020            LDAI  R5,CHKBUF
27B4  0A65                3021            AAR   R6,R5
27B6  41D0 286A           3022            BAL   R13,READMAG   READ INTO CHECKBUF
27BA  4300 2772           3023            B     VERIFY01
                          3024  *
27BE  4830 167E           3025  DISKOUT   LH    R3,OUTDEV+$VALU2  CONTROLLER ADDRESS
27C2  4840 1688           3026            LH    R4,SELCH2+$VALU1  SELCH ADDRESS
27C6  4880 1A1A           3027            LH    R8,NEXTCYL        CURRENT POSITION ON OUTPUT DISK
27CA  4890 1A1C           3028            LH    R9,NEXTHEAD
27CE  48A0 1A1E           3029            LH    R10,NEXTSECT
27D2  4870 1A6C           3030            LH    R7,DISKTYP2
27D6  4070 1A68           3031            STH   R7,DISKTYPE       SELECT OUTPUT DISK TYPE
27DA  4870 1A18           3032            LH    R7,VERIFLAG
```

TRANSFER ACTUAL PROGRAM

```
27DE 4330 2806          3033           BZ    WRITE256          SKIP IF NOT VERIFY
27E2 0B65               3034           SAR   R6,R5             BYTE COUNT MINUS 1
27E4 C850 3C5C          3035           LOAI  R5,CHKBUF         NEW START ADDRESS
27E8 0A65               3036           AAR   R6,R5             NEW END ADDRESS
27EA 41F0 2F38          3037 VERIF256  BAL   R15,RNXTSECT       READ A SECTOR
27EE CA50 0100          3038           AHI   R5,256            BUMP START ADDRESS
27F2 0556               3039           CLAR  R5,R6
27F4 2085               3040           BLS   VERIF256          READ ANOTHER SECTOR
27F6 4080 1A1A          3041           STH   R8,NEXTCYL        SAVE CURRENT POSITION
27FA 4090 1A1C          3042           STH   R9,NEXTHEAD
27FE 40A0 1A1E          3043           STH   R10,NEXTSECT
2802 4300 2772          3044           B     VERIFY01          GO VERIFY
                        3045 *
2806 41F0 2F3C          3046 WRITE256  BAL   R15,WNXTSECT       WRITE A SECTOR
280A CA50 0100          3047           AHI   R5,256            BUMP START ADDRESS
280E 0556               3048           CLAR  R5,R6
2810 2085               3049           BLS   WRITE256          WRITE ANOTHER SECTOR
2812 4080 1A1A          3050           STH   R8,NEXTCYL        SAVE CURRENT POSITION
2816 4090 1A1C          3051           STH   R9,NEXTHEAD
281A 40A0 1A1E          3052           STH   R10,NEXTSECT
                        3053 *
                        3054 *
281E 4850 1B7E          3055 OUTTEST   LH    R5,RECORDS         NO. OF 256 BYTE RECORDS TO DO
2822 D360 1B86          3056           LB    R6,PROGSIZE        NO. OF RECORDS DONE THIS TIME
2826 0866               3057           LDAR  R6,R6              LESS THAN 256 BYTES DONE?
2828 2132               3058           BNZS  OUTTEST1           SKIP IF NO
282A 2461               3059           LIS   R6,1               IF YES, FORCE DECREMENT OF 1
282C 0856               3060 OUTTEST1  SAR   R5,R6              DECREMENT TALLY BY NUMBER DONE
282E 4050 1B7E          3061           STH   R5,RECORDS
2832 4220 2670          3062           BP    DUPEPROG           MORE RECORDS TO DO IF PLUS
2836 2116               3063           BMS   FINISH             DONE IF MINUS
2838 D360 1B82          3064           LB    R6,LEFTOVER        IF ZERO, CHECK LEFTOVER
283C 2761               3065           SIS   R6,1               R6 ADJUST FOR FINAL ADDRESS
                        3066 *                                  IF LEFTOVER = ZERO, NO MORE BYTES
283E 4310 2682          3067           BNM   GETSOME            GO FOR LAST RECORD
                        3068 *                                  WHICH HAS LESS THAN 256 BYTES
```

PROGRAM WRAP-UP, DIRECTORY UPDATE

```
2842  4860 1A66          3070 FINISH   LH    R6,OSFLAG
*2846 2338               3071          BZ    FINISH.0          SKIP IF NOT OS
2848  4860 1AFC          3072          LH    R6,SEQNUM
284C  2762               3073          SIS   R6,2              CLEAR OS FLAG AFTER PROGRAM 002
284E  4060 1A66          3074          STH   R6,OSFLAG
2852  4300 28FE          3075          B     OSDISKX           GO DO DIRECTORY IF LAST O.S.
2856  4860 1A18          3076 FINISH.0 LH    R6,VERIFLAG       VERIFY?
285A  4230 1F7C          3077          BNZ   FINDNEXT          GO FOR NEXT PROGRAM IF YES
285E  4860 1A50          3078          LH    R6,OUTFLAG        LOOK AT OUTPUT DEVICE
2862  2761               3079          SIS   R6,1              MAG TAPE?
2864  4330 1D3C          3080          BE    MAGFIN            YES, WRITE FILE MARKS
2868  2761               3081          SIS   R6,1              FLOPPY?
286A  4230 28C0          3082          BNZ   DISKFIN           FINISH UP ON DISK OUTPUT
286E  41E0 2D48          3083          BAL   R14,RDIRECT        READ THE DIRECTORY
2872  4860 1AFC          3084          LH    R6,SEQNUM         GET THIS ONE'S SEQUENCE NUMBER
2876  4065 385C          3085          STH   R6,INBUF(R5)      PUT IT IN THE DIRECTORY
287A  4860 1B88          3086          LH    R6,PDBSTART       STARTING PDB FOR THIS PROGRAM
287E  4065 385E          3087          STH   R6,INBUF+2(R5)    INTO THE DIRECTORY TOO
2882  084A               3088          LDAR  R4,R10            BLOCK POINTER
2884  C550 007D          3089          CLHI  R5,125            ROOM FOR EOV MARKERS IN THIS BLOCK?
2888  238C               3090          BNLS  FINISH.1          SKIP IF NO
288A  2460               3091 FINISH.2 LIS   R6,0
288C  4065 3860          3092          STH   R6,INBUF+4(R5)    STUFF EOV MARKERS
2890  4065 3862          3093          STH   R6,INBUF+6(R5)
2894  41C0 2ABC          3094          BAL   R12,SULH56
2898  41E0 2D00          3095          BAL   R14,WRLRN         WRITE DIRECTORY BLOCK
289C  4300 1F7C          3096          B     FINDNEXT          GO FOR NEXT PROGRAM
28A0  D440 1A12          3097 FINISH.1 CLB   R4,ENDDIR         AT MAXIMUM?
28A4  4330 2A60          3098          BE    DIRERR            ERROR IF YES
28A8  41C0 2ABC          3099          BAL   R12,SULH56
28AC  41E0 2D00          3100          BAL   R14,WRLRN         WRITE THIS BLOCK
28B0  2641               3101          AIS   R4,1              INCREMENT TO NEXT
28B2  41C0 2A8C          3102          BAL   R12,SULH56
28B6  41E0 2CA0          3103          BAL   R14,RDLRN         READ NEXT DIRECTORY BLOCK
28BA  2450               3104          LIS   R5,0
28BC  4300 288A          3105          B     FINISH.2          STUFF EOV MARKERS
                         3106 *
                         3107 *
28C0  41F0 2ECC          3108 DISKFIN  BAL   R15,DIRSERCH      LOOK IN DIRECTORY
28C4  0866               3109          LOAR  R6,R6
28C6  4230 2A60          3110          BNZ   DIRERR            DIRECTORY ERROR
28CA  4860 375C          3111          LH    R6,PDB
28CE  4065 385C          3112          STH   R6,INBUF(R5)      COPY SEQUENCE NUMBER
28D2  D360 375E          3113          LB    R6,PDB+2
28D6  D265 385E          3114          STB   R6,INBUF+2(R5)
28DA  4860 1A74          3115          LH    R6,DIRDAT
28DE  4065 3860          3116          STH   R6,INBUF+4(R5)    COPY CYLINDER NUMBER
28E2  4860 1A76          3117          LH    R6,DIRDAT+2
28E6  D265 3863          3118          STB   R6,INBUF+7(R5)    COPY HEAD NUMBER
28EA  4860 1A78          3119          LH    R6,DIRDAT+4
28EE  D265 3862          3120          STB   R6,INBUF+6(R5)    COPY SECTOR NUMBER
28F2  41C0 2AC6          3121          BAL   R12,SULH56,2
28F6  41E0 2DDA          3122          BAL   R14,WRITSECT      REWRITE DIRECTORY SECTOR
```

PROGRAM WRAP-UP, DIRECTORY UPDATE

```
28FA  4300 1F7C        3123          B     FINDNEXT          GO FOR NEXT PROGRAM


28FE  41F0 2FA8        3125  OSDISKX BAL   R15,FORMLBA       FORM FINAL LBA FOR THIS ONE
2902  2711             3126          SIS   R1,1             DECREMENT BY 1
2904  2382             3127          BNCS  OSDISKX1
2906  2701             3128          SIS   R0,1
2908  4870 1AFC        3129  OSDISKX1 LH   R7,SEQNUM        SEQUENCE 001 OR 002
290C  C470 0002        3130          NHI   R7,2
2910  0A77             3131          AAR   R7,R7             R7 = 0 OR 4
2912  4007 1A82        3132          STH   R0,FINALLBA(R7)   SAVE FOR DIRECTORY
2916  4017 1A84        3133          STH   R1,FINALLBA+2(R7)
291A  0877             3134          LDAR  R7,R7             LAST ONE?
291C  4330 1F7C        3135          BZ    FINDNEXT          NO, DO ANOTHER ONE
2920  41C0 2AC6        3136          BAL   R12,SULH56,2
2924  41C0 2AD0        3137          BAL   R12,ZEROBUFF      PRESET TO ZERO
2928  C860 4F53        3138          LHI   R6,C'OS'
292C  C870 3136        3139          LHI   R7,C'16'
2930  4060 3860        3140          STH   R6,DIRECT1+FNAME  FIRST ENTRY = "OS16MDL2"
2934  4070 3862        3141          STH   R7,DIRECT1+FNAME+2
2938  C870 3332        3142          LHI   R7,C'32'
293C  4060 3890        3143          STH   R6,DIRECT2+FNAME  SECOND ENTRY = "OS32MDL2"
2940  4070 3892        3144          STH   R7,DIRECT2+FNAME+2
2944  4860 16D0        3145          LH    R6,OSID+$VALU1    GET OPTIONAL CHARACTERS
2948  4870 16D2        3146          LH    R7,OSID+$VALU2
294C  4060 3864        3147          STH   R6,DIRECT1+FNAME+4  STORE "MDL2"
2950  4070 3866        3148          STH   R7,DIRECT1+FNAME+6
2954  4060 3894        3149          STH   R6,DIRECT2+FNAME+4
2958  4070 3896        3150          STH   R7,DIRECT2+FNAME+6
295C  C860 4D4D        3151          LHI   R6,C'MM'
2960  C870 4420        3152          LHI   R7,C'D '
2964  4060 38C0        3153          STH   R6,DIRECT3+FNAME  THIRD ENTRY = "MMD "
2968  4070 38C2        3154          STH   R7,DIRECT3+FNAME+2
296C  C870 2020        3155          LHI   R7,X'2020'        FILL OUT WITH SPACES
2970  4070 38C4        3156          STH   R7,DIRECT3+FNAME+4
2974  4070 38C6        3157          STH   R7,DIRECT3+FNAME+6
2978  C860 3131        3158          LHI   R6,C'11'
297C  4060 3868        3159          STH   R6,DIRECT1+EXT    EXTENSION 111 FOR FIRST ENTRY
2980  D260 386A        3160          STB   R6,DIRECT1+EXT+2
2984  4060 3898        3161          STH   R6,DIRECT2+EXT    EXTENSION 111 FOR SECOND ENTRY
2988  D260 389A        3162          STB   R6,DIRECT2+EXT+2
298C  C860 4441        3163          LHI   R6,C'DA'
2990  C870 0054        3164          LHI   R7,C'T'
2994  4060 38C8        3165          STH   R6,DIRECT3+EXT    EXTENSION DAT FOR THIRD ENTRY
2998  D270 38CA        3166          STB   R7,DIRECT3+EXT+2
299C  C870 0010        3167          LHI   R7,X'10'
29A0  D270 3884        3168          STB   R7,DIRECT1+ATTR   EACH ENTRY HAS ATTRIBUTE '10'
29A4  D270 3884        3169          STB   R7,DIRECT2+ATTR
29A8  D270 38E4        3170          STB   R7,DIRECT3+ATTR
29AC  2571             3171          LCS   R7,1
29AE  4070 3874        3172          STH   R7,DIRECT1+KEYS   EACH HAS PROTECTION KEYS 'FFFF'
29B2  4070 38A4        3173          STH   R7,DIRECT2+KEYS
```

PROGRAM WRAP-UP, DIRECTORY UPDATE

```
29B6  4070 38D4          3174           STH   R7,DIRECT3+KEYS
29BA  4860 1A7A          3175           LH    R6,STARTLBA
29BE  4870 1A7C          3176           LH    R7,STARTLBA+2
29C2  4060 386C          3177           STH   R6,DIRECT1+FLBA      FIRST LOGICAL BLOCK ADDRESS
29C6  4070 386E          3178           STH   R7,DIRECT1+FLBA+2    FOR FIRST DIRECTORY ENTRY
29CA  4860 1A82          3179           LH    R6,FINALLBA
29CE  4870 1A84          3180           LH    R7,FINALLBA+2
29D2  4060 3870          3181           STH   R6,DIRECT1+LLBA      LAST LOGICAL BLOCK ADDRESS
29D6  4070 3872          3182           STH   R7,DIRECT1+LLBA+2    FOR FIRST ENTRY
29DA  4860 1A7E          3183           LH    R6,STARTLBA+4
29DE  4870 1A80          3184           LH    R7,STARTLBA+6
29E2  4060 389C          3185           STH   R6,DIRECT2+FLBA      FIRST LOGICAL BLOCK ADDRESS
29E6  4070 389E          3186           STH   R7,DIRECT2+FLBA+2    FOR SECOND ENTRY
29EA  4860 1A86          3187           LH    R6,FINALLBA+4
29EE  4870 1A88          3188           LH    R7,FINALLBA+6
29F2  4060 38A0          3189           STH   R6,DIRECT2+LLBA      LAST LOGICAL BLOCK ADDRESS
29F6  4070 38A2          3190           STH   R7,DIRECT2+LLBA+2    FOR SECOND ENTRY
29FA  41F0 2FA8          3191           BAL   R15,FORMLBA          TRANSLATE R8,R9,R10 TO LBA
29FE  4000 38CC          3192           STH   R0,DIRECT3+FLBA      FIRST LOGICAL BLOCK ADDRESS
2A02  4010 38CE          3193           STH   R1,DIRECT3+FLBA+2    FOR THE LIBRARY ITSELF
2A06  D380 1A14          3194           LB    R8,DIRSTART          POINT TO CYLINDER 8
2A0A  2490               3195           LIS   R9,0                 HEAD 0
                         3196  *                                   WATCH THIS AREA!!!!!!!!!!!!!!!!!
2A0C  24A0               3197           LIS   R10,0                SECTOR 0
2A0E  41F0 2FA8          3198           BAL   R15,FORMLBA          CHANGE TO LBA
2A12  CA10 2EE0          3199           AHI   R1,12000             12,000 SECTORS TOTAL
2A16  2382               3200           BNCS  OSDISKX2
2A18  2601               3201           AIS   R0,1
2A1A  4000 38D0          3202  OSDISKX2 STH   R0,DIRECT3+LLBA      LAST LOGICAL BLOCK ADDRESS
2A1E  4010 38D2          3203           STH   R1,DIRECT3+LLBA+2    FOR THE LIBRARY
2A22  4880 1A48          3204           LH    R8,OSDIR             CYLINDER,HEAD,SECTOR
2A26  4890 1A4A          3205           LH    R9,OSDIR+2           WHERE THE DIRECTORY GOES
2A2A  48A0 1A4C          3206           LH    R10,OSDIR+4
2A2E  41F0 2FD8          3207           BAL   R15,DISKSET          WRITE THE DIRECTORY
2A32  4230 1F3C          3208           BNE   DIRECTER             VERIFY ERROR
2A36  41F0 0FB0          3209           BAL   R15,CRLF             CARRIAGE RETURN, LINE FEED
2A3A  4300 1B92          3210           B     RUN.0000             RESET INPUT AND DO LIBRARY
```

ADVISORY AND ERROR MESSAGE ROUTINES

```
2A3E 41D0 2B12  3212 ENDOVOL  BAL  R13,STOP          FLOPPY STOP
2A42 C850 332D  3213 ENDOVOL1 LDAI R5,EOVMSG2        "END OF VOLUME"
2A46 4300 1C50  3214          B    CONMSG1
2A4A 41D0 2B12  3215 WRERR1   BAL  R13,STOP          FLOPPY STOP
2A4E 4020 154A  3216          STH  R2,ERRDEV
2A52 C850 32F8  3217          LDAI R5,DIRER
2A56 D440 1A13  3218          CLB  R4,STSAV          WHERE WERE WE?
*2A5A 2183      3219          BL   DIRERR            DIRECTORY ERROR
2A5C C850 34C7  3220          LDAI R5,UNRECOV
2A60 41F0 12BC  3221 DIRERR   BAL  R15,SETKB
2A64 41F0 0FD2  3222          BAL  R15,PRINT
2A68 41E0 0E30  3223 DIRERR1  BAL  R14,ERRDS1        DEV DD STA SS
2A6C 4300 0AAA  3224          B    OPTIN
2A70 41D0 2B12  3225 RDIRERR1 BAL  R13,STOP          FLOPPY STOP
2A74 C850 333D  3226          LDAI R5,RDEMSG         "DIRECTORY READ ERROR"
2A78 4300 1C50  3227          B    CONMSG1
                3228 *
2A7C C850 3374  3229 EOD      LDAI R5,EODMSG         DIRECTORY FULL
2A80 4300 1C50  3230          B    CONMSG1
                3231 *
2A84 4000 154A  3232 DU       STH  R0,ERRDEV         STORE DEVICE
2A88 41F0 12BC  3233          BAL  R15,SETKB         DIRECT PRINTOUT TO CONSOLE
2A8C 41F0 0FBA  3234          BAL  R15,$PRINT
2A90 34B2       3235          DAC  DEVUNA            'DEVICE UNAVAILABLE'
2A92 41E0 0E0C  3236          BAL  R14,ERRD1         'DEV=DD'
2A96 4300 0AAA  3237          B    OPTIN
                3238 *
2A9A C850 3499  3239 WPROT    LDAI R5,WPROTMSG       "DEVICE WRITE PROTECTED"
2A9E 4300 1C50  3240          B    CONMSG1
2AA2 4000 154A  3241 MTERXY   STH  R0,ERRDEV
2AA6 D210 154C  3242          STB  R1,ERRSTA
2AAA 41F0 12BC  3243          BAL  R15,SETKB
2AAE 41F0 0FBA  3244          BAL  R15,$PRINT
2AB2 34C7       3245          DAC  UNRECOV           "UNRECOVERABLE ERROR"
2AB4 41E0 0E30  3246          BAL  R14,ERRDS1        DEV DDD STA SS
2AB8 4300 0AAA  3247          B    OPTIN
```

LEVEL 12 SUBROUTINES

```
                        3249  * S U B R O U T I N E   S U L H 5 6
                        3250  *
2ABC  C850 385C         3251  SULH56    LDAI  R5,INBUF          START ADDRESS
2AC0  C865 007F         3252            LHI   R6,127(R5)        END ADDRESS
2AC4  030C              3253            BR    R12


                        3255  * S U B R O U T I N E   S U L H 5 6 . 2
                        3256  *
2AC6  C850 385C         3257  SULH56.2  LDAI  R5,INBUF          START ADDRESS
2ACA  C865 00FF         3258            LHI   R6,255(R5)        END ADDRESS
2ACE  030C              3259            BR    R12               RETURN


                        3261  * S U B R O U T I N E   Z E R O B U F F
                        3262  *
2AD0  2400              3263  ZEROBUFF  LIS   R0,0              CLEAR RETRY COUNT
2AD2  4050 1B84         3264            STH   R5,R5SAVE         SAVE START ADDRESS
2AD6  4005 0000         3265  ZEROBUF1  STH   R0,0(R5)          CLEAR THE INPUT BUFFER FIRST
2ADA  2652              3266            AIS   R5,2
2ADC  0556              3267            CLAR  R5,R6
2ADE  2084              3268            BLS   ZEROBUF1
2AE0  4850 1B84         3269            LH    R5,R5SAVE
2AE4  030C              3270            BR    R12


                        3272  * S U B R O U T I N E   H O W M A N Y
                        3273  *
2AE6  0876              3274  HOWMANY   LDAR  R7,R6             INBUF END ADDRESS
2AE8  CB70 385C         3275            SHI   R7,INBUF          MINUS START ADDRESS
2AEC  2671              3276            AIS   R7,1              PLUS 1
2AEE  4070 1B86         3277            STH   R7,PROGSIZE       ACTUAL COUNT
2AF2  4870 1A50         3278            LH    R7,OUTFLAG        IF OUTPUT DEVICE IS A MAG TAPE
2AF6  2771              3279            SIS   R7,1              USE THE ACTUAL BYTE COUNT
2AF8  033C              3280            BZR   R12
2AFA  4870 1B86         3281            LH    R7,PROGSIZE       OTHERWISE, ROUND UP
2AFE  2771              3282            SIS   R7,1              TO NEXT MULTIPLE OF 128
2B00  CA70 0080         3283            AHI   R7,X'80'
2B04  C470 FF80         3284            NHI   R7,X'FF80'
2B08  4070 1B86         3285            STH   R7,PROGSIZE       ROUNDED TO MODULO 128
2B0C  030C              3286            BR    R12
```

LEVEL 13 SUBROUTINES

```
                        3288  * S U B R O U T I N E   F M D I D L E
                        3289  *
2B0E  9D21              3290  FMDIDLE  SSR   R2,R1              SENSE STATUS
2B10  022D              3291           BTCR  2,R13               EXIT IF IDLE


                        3293  * S U B R O U T I N E   S T O P
                        3294  *
2B12  2417              3295  STOP     LIS   R1,7
2B14  4610 1A24         3296           OH    R1,FMDCMD
2B18  9E21              3297           OCR   R2,R1               ISSUE STOP COMMAND
2B1A  9D21              3298           SSR   R2,R1
2B1C  2221              3299           BFBS  2,1                 WAIT FOR IDLE
2B1E  030D              3300           BR    R13                 RETURN


                        3302  * S U B R O U T I N E   F I L E S E T
                        3303  *
2B20  4870 1A68         3304  FILESET  LH    R7,DISKTYPE
2B24  DE20 1A35         3305           OC    R2,DRESET           RESET GATED ATTENTION
2B28  9D31              3306           SSR   R3,R1               WAIT FOR CONTROLLER IDLE
2B2A  2221              3307           BFBS  2,1
2B2C  C570 0002         3308           CLHI  R7,2
*2B30  233C              3309           BE    FILESET1            10 MB
2B32  9829              3310           WHR   R2,R9               'HEAD'
2B34  DE20 1A37         3311           OC    R2,SETHEAD
2B38  9D31              3312           SSR   R3,R1
2B3A  2221              3313           BFBS  2,1
2B3C  9828              3314           WHR   R2,R8               'CYLINDER'
2B3E  DE20 1A36         3315           OC    R2,SETCYL
2B42  9D31              3316           SSR   R3,R1
2B44  2221              3317           BFBS  2,1
2B46  030D              3318           BR    R13
2B48  9828              3319  FILESET1 WHR   R2,R8               SEND CYLINDER
2B4A  030D              3320           BR    R13


                        3322  * S U B R O U T I N E   D I S K W A I T
                        3323  *
2B4C  9D31              3324  DISKWAIT SSR   R3,R1
2B4E  2221              3325           BFBS  2,1                 IDLE?
2B50  0802              3326           LDAR  R0,R2               R0 GETS ERROR DEVICE NUMBER
2B52  9D21              3327  SENSED   SSR   R2,R1
2B54  4210 2A84         3328           BTC   1,DU                DEVICE UNAVAILABLE
2B58  2083              3329           BTCS  8,SENSED            LOOP ON BUSY
2B5A  030D              3330           BR    R13


                        3332  * S U B R O U T I N E   C O N T I D L E
```

LEVEL 13 SUBROUTINES

```
                          3333  *
2B5C   9D31               3334  CONTIDLE SSR   R3,R1
2B5E   2221               3335           BFBS  2,1                IDLE
2B60   030D               3336           BR    R13


                          3338  * S U B R O U T I N E   W R I T E M A G
                          3339  *
2B62   2471               3340  WRITEMAG LIS   R7,1
2B64   4070 1A52          3341           STH   R7,DRWFLAG
2B68   2304               3342           BS    MAGINOUT


                          3344  * S U B R O U T I N E   R E A D M A G
                          3345  *
2B6A   2470               3346  READMAG  LIS   R7,0
2B6C   4070 1A52          3347           STH   R7,DRWFLAG
2B70   2405               3348  MAGINOUT LIS   R0,5              CLEAR RETRY COUNT
2B72   4000 1A56          3349           STH   R0,RETRY
2B76   4050 1B84          3350  MAGIO1   STH   R5,R5SAVE         SAVE RECORD START ADDRESS
2B7A   C500 1D88          3351           CLHI  R13,INITDISK       IF DOING MAG TAPE BOOTS,
2B7E   2185               3352           BLS   MAGIO1A            SKIP RECORD SIZE CHECKING
2B80   C805 00FF          3353           LHI   R0,255(R5)         ELSE, MAX RECORD SIZE IS 256 BYTES
2B84   0506               3354           CLAR  R0,R6              COMPARE TO ACTUAL END ADDRESS
2B86   2182               3355           BLS   MAGIO2             USE START + 255 IF LESS THAN ACTUAL
2B88   0806               3356  MAGIO1A  LDAR  R0,R6              ELSE USE ACTUAL END ADDRESS
2B8A   9D21               3357  MAGIO2   SSR   R2,R1             WAIT FOR NO MOTION BEFORE
2B8C   C310 0010          3358           THI   R1,X'10'           READING OR WRITING
2B90   2233               3359           BZS   MAGIO2
2B92   0844               3360           LDAR  R4,R4              SELCH?
2B94   4230 2BCA          3361           BNZ   MAGIO5             YES
2B98   4870 1A52          3362           LH    R7,DRWFLAG
2B9C   4230 2BFA          3363           BNZ   MAGWRIT1
2BA0   DE20 1A2E          3364           OC    R2,MTREAD          ISSUE READ COMMAND
2BA4   9D21               3365  READMAG1 SSR   R2,R1
2BA6   2081               3366           BTBS  8,1                LOOP ON BUSY
2BA8   D825 0000          3367           RD    R2,0(R5)
2BAC   2651               3368           AIS   R5,1
2BAE   0505               3369           CLAR  R0,R5
*  2BB0   2286               3370           BNL   READMAG1           LOOP IF NOT DONE
2BB2   9D21               3371  MAGIO4   SSR   R2,R1
2BB4   C310 0010          3372           THI   R1,X'10'
2BB8   2233               3373           BZS   MAGIO4             WAIT FOR NO MOTION
2BBA   C310 00C1          3374           THI   R1,X'C1'           ERROR?
2BBE   4230 2C1C          3375           BNZ   MTERR
2BC2   0565               3376           CLAR  R6,R5              DONE?
2BC4   4380 2B76          3377           BNL   MAGIO1
2BC8   030D               3378           BR    R13                RETURN
2BCA   DE40 1A30          3379  MAGIO5   OC    R4,STOPS           STOP SELCH
2BCE   9845               3380           WHR   R4,R5              OUTPUT START ADDRESS
2BD0   9840               3381           WHR   R4,R0              OUTPUT END ADDRESS
```

LEVEL 13 SUBROUTINES

```
2BD2  4870 1A52          3382          LH    R7,DRWFLAG
2BD6  4230 2C10          3383          BNZ   MAGWRIT2
2BDA  DE20 1A2E          3384          OC    R2,MTREAD          MAG TAPE READ
2BDE  DE40 1A3C          3385          OC    R4,SREAD           SELCH READ
                         3386  *
2BE2  9D41               3387  MAGIO6  SSR   R4,R1             WAIT ON SELCH BUSY
2BE4  2081               3388          BTBS  8,1
2BE6  DE40 1A30          3389          OC    R4,STOPS
2BEA  9941               3390          RHR   R4,R1              UNLOAD FINAL ADDRESS
2BEC  0501               3391          CLAR  R0,R1              COMPARE TO EXPECTED FINAL ADRESS
2BEE  4230 2C1C          3392          BNE   MTERR
2BF2  C851 0001          3393          LHI   R5,1(R1)           NEXT START ADDRESS
2BF6  4300 2BB2          3394          B     MAGIO4
                         3395  *
2BFA  DE20 1A41          3396  MAGWRIT1 OC   R2,MTWRITE
2BFE  9D21               3397  WRITMAG1 SSR  R2,R1
2C00  2081               3398          BTBS  8,1
2C02  DA25 0000          3399          WD    R2,0(R5)
2C06  2651               3400          AIS   R5,1
2C08  0505               3401          CLAR  R0,R5
*2C0A 2286               3402          BNL   WRITMAG1
2C0C  4300 2BB2          3403          B     MAGIO4
                         3404  *
2C10  DE20 1A41          3405  MAGWRIT2 OC   R2,MTWRITE
2C14  DE40 1A33          3406          OC    R4,SWRITE
2C18  4300 2BE2          3407          B     MAGIO6
                         3408  *
2C1C  9D21               3409  MTERR   SSR   R2,R1
2C1E  C310 0010          3410          THI   R1,X'10'           WAIT FOR NO MOTION
2C22  2233               3411          BZS   MTERR
2C24  9D21               3412          SSR   R2,R1
2C26  C310 0040          3413          THI   R1,X'40'           EOF?
2C2A  4230 2A42          3414          BNZ   ENDOVOL1           DONE IF YES
                         3415  *
2C2E  0802               3416  MTERR1  LDAR  R0,R2              ERROR DEVICE
2C30  4830 1A56          3417          LH    R3,RETRY
2C34  2731               3418          SIS   R3,1               DECREMENT RETRY COUNT
2C36  4030 1A56          3419          STH   R3,RETRY
2C3A  4210 2AA2          3420          BM    MTERXY             UN-RECOVERABLE ERROR
2C3E  DE20 1A3F          3421          OC    R2,BKSPRCRD        BACKSPACE RECORD
2C42  9D21               3422  MTERR2  SSR   R2,R1              WAIT FOR NO MOTION
2C44  C310 0010          3423          THI   R1,X'10'
2C48  2233               3424          BZS   MTERR2
2C4A  4850 1B84          3425          LH    R5,R5SAVE          RESTORE START ADDRESS
2C4E  4300 2B76          3426          B     MAGIO1             TRY AGAIN, THIS RECORD


                         3428  * S U B R O U T I N E   P A C K
                         3429  *
2C52  2440               3430  PACK    LIS   R4,0
2C54  D303 0000          3431  PACK.001 LB   R0,0(R3)
2C58  C500 0041          3432          CLHI  R0,C'A'
```

LEVEL 13 SUBROUTINES

```
2C5C   2182            3433           BLS   PACK.002
2C5E   2707            3434           SIS   R0,7
2C60   C400 000F       3435  PACK.002 NHI   R0,X'F'
2C64   9144            3436           SLLS  R4,4          SHIFT ACCUMULATOR
2C66   0640            3437           OAR   R4,R0         OR IN NEW DIGIT
2C68   2631            3438           AIS   R3,1          INCREMENT INDEX
2C6A   2721            3439           SIS   R2,1          DECREMENT COUNT
*2C6C  203C            3440           BNZ   PACK.001      LOOP
2C6E   030D            3441           BR    R13
```

LEVEL 14 SUBROUTINES

```
                                3443  * S U B R O U T I N E   R E A D C H C K
                                3444  *
 2C70   C570 0002               3445  READCHCK CLHI  R7,2
*2C74   233D                    3446           BE    READCHK1
 2C76   41D0 2B20               3447           BAL   R13,FILESET
 2C7A   9A3A                    3448           WDR   R3,R10            'SECTOR'
 2C7C   0819                    3449           LDAR  R1,R9             'HEAD'
 2C7E   911A                    3450           SLLS  R1,10
 2C80   0618                    3451           OAR   R1,R8             'CYLINDER'
 2C82   9831                    3452           WHR   R3,R1
 2C84   DE30 1A34               3453  SETUP    OC    R3,RCHECK
 2C88   41D0 2B5C               3454           BAL   R13,CONTIDLE
 2C8C   030E                    3455           BR    R14               RETURN
 2C8E   41D0 2B4C               3456  READCHK1 BAL   R13,DISKWAIT
 2C92   41D0 2B20               3457           BAL   R13,FILESET
 2C96   0819                    3458           LDAR  R1,R9             'HEAD'
 2C98   9115                    3459           SLLS  R1,5
 2C9A   061A                    3460           OAR   R1,R10            'SECTOR'
 2C9C   9A31                    3461           WDR   R3,R1
*2C9E   220D                    3462           B     SETUP


                                3464  * S U B R O U T I N E   R D L R N
                                3465  *
 2CA0   2400                    3466  RDLRN    LIS   R0,0              CLEAR RETRY COUNTER
 2CA2   4050 1B84               3467           STH   R5,R5SAVE         SAVE START ADDRESS
 2CA6   41D0 2B0E               3468  RDLRN.01 BAL   R13,FMDIDLE       IDLE CHECK
 2CAA   4810 1A24               3469           LH    R1,FMDCMD
 2CAE   2611                    3470           AIS   R1,1              FORM READ COMMAND
 2CB0   9824                    3471           WHR   R2,R4             LRN TO CONTROLLER
 2CB2   9E21                    3472           OCR   R2,R1             ISSUE READ COMMAND
 2CB4   9D23                    3473  RDLRN.02 SSR   R2,R3
 2CB6   2081                    3474           BTBS  8,1               WAIT ON BUSY
 2CB8   C870 007E               3475           LHI   R7,126            LOOP COUNT
 2CBC   D925 0000               3476  RDLRN.03 RH    R2,0(R5)         READ HALFWORDS
 2CC0   2652                    3477           AIS   R5,2              BUMP INDEX
 2CC2   2772                    3478           SIS   R7,2              DECREMENT COUNT
 2CC4   2284                    3479           BNLS  RDLRN.03          LOOP THRU 128 BYTES
 2CC6   0556                    3480           CLAR  R5,R6             DONE YET?
 2CC8   208A                    3481           BLS   RDLRN.02          LOOP IF NO
                                3482  *
 2CCA   DD20 154C               3483           SS    R2,ERRSTA
*2CCE   2154                    3484           BTC   5,RDLRNE          READ ERROR
 2CD0   41D0 2B12               3485           BAL   R13,STOP
 2CD4   030E                    3486           BR    R14               RETURN
 2CD6   41D0 2B12               3487  RDLRNE   BAL   R13,STOP
 2CDA   C500 0005               3488           CLHI  R0,5              MAX RETRY?
 2CDE   2386                    3489           BNLS  RDLRNE1           HARD ERROR IF YES
 2CE0   2601                    3490           AIS   R0,1              ELSE, BUMP RETRY COUNT
 2CE2   4850 1B84               3491           LH    R5,R5SAVE         RESTORE START ADDRESS
 2CE6   4300 2CA6               3492           B     RDLRN.01          TRY AGAIN
 2CEA   4020 154A               3493  RDLRNE1  STH   R2,ERRDEV
```

LEVEL 14 SUBROUTINES

```
2CEE  41F0 128C  3494          BAL   R15,SETKB
2CF2  41F0 0FBA  3495          BAL   R15,$PRINT
2CF6  34C7       3496          DAC   UNRECOV          READ ERROR
2CF8  41E0 0E30  3497          BAL   R14,ERRnS1       "DEV DDD STA SS"
2CFC  4300 0AAA  3498          B     OPTIN


                 3500  * S U B R O U T I N E  W R L R N
                 3501  *
2D00  2400       3502  WRLRN    LIS   R0,0             CLEAR RETRY COUNTER
2D02  4050 1B84  3503          STH   R5,R5SAVE        SAVE START ADDRESS
2D06  41D0 2B0E  3504  WRLRN.01 BAL   R13,FMDIDLE     IDLE CHECK
2D0A  2412       3505          LIS   R1,2
2D0C  4610 1A24  3506          OH    R1,FMDCMD        FORM WRITE COMMAND
2D10  9824       3507          WHR   R2,R4            WRITE LRN TO CONTROLLER
2D12  9E21       3508          OCR   R2,R1            ISSUE WRITE COMMAND
2D14  9D23       3509  WRLRN.02 SSR   R2,R3
2D16  2081       3510          BTBS  8,1              WAIT FOR NOT BUSY
2D18  C870 007E  3511          LHI   R7,126
2D1C  D825 0000  3512  WRLRN.03 WH    R2,0(R5)        WRITE TO DISKETTE
2D20  2652       3513          AIS   R5,2             BUMP INDEX
2D22  2772       3514          SIS   R7,2             DECREMENT COUNT
2D24  2284       3515          BNLS  WRLRN.03         WRITE 128 BYTES
                 3516  *
2D26  0556       3517          CLAR  R5,R6            DONE?
*2D28 208A       3518          BL    WRLRN.02         NO, ANOTHER 128
2D2A  DD20 154C  3519          SS    R2,ERRSTA        YES, CHECK STATUS OF XFER
*2D2E 2154       3520          BTC   5,WBERR          WRITE ERROR
2D30  41D0 2B12  3521          BAL   R13,STOP         NO ERROR, STOP
2D34  030E       3522          BR    R14              RETURN
2D36  C500 0005  3523  WBERR    CLHI  R0,5             MAX RETRY?
2D3A  4380 2A4A  3524          BNL   WBERR1           YES, HARD ERROR
2D3E  2601       3525          AIS   R0,1            NO, BUMP RETRY COUNT
2D40  4850 1B84  3526          LH    R5,R5SAVE        RESTORE START ADDRESS
2D44  4300 2D06  3527          B     WRLRN.01         TRY AGAIN


                 3529  * S U B R O U T I N E  R D I R E C T
                 3530  *
2D48  2400       3531  RDIRECT  LIS   R0,0             CLEAR RETRY COUNTER
2D4A  D3A0 1A10  3532          LB    R10,STDIRM       DIRECTORY START LRN MINUS 1
2D4E  D340 1A11  3533          LB    R4,STDIR         START LRN FOR DIRECTORY
2D52  41D0 2B0E  3534  RDLOOP   BAL   R13,FMDIDLE     IDLE CHECK
2D56  41C0 2ABC  3535          BAL   R12,SULH56
2D5A  4890 1AFC  3536          LH    R9,SEQNUM        CURRENT OUTPUT SEQUENCE NUMBER
2D5E  2411       3537          LIS   R1,1
2D60  4610 1A24  3538          OH    R1,FMDCMD        FORM READ COMMAND
2D64  9824       3539          WHR   R2,R4            SEND LRN TO CONTROLLER
2D66  9E21       3540          OCR   R2,R1            ISSUE READ COMMAND
2D68  26A1       3541  RDIRECT2 AIS   R10,1           BUMP CURRENT LRN POINTER
2D6A  D4A0 1A13  3542          CLB   R10,STSAV        OUT OF DIRECTORY?
```

LEVEL 14 SUBROUTINES

```
2D6E 4330 2DC0 3543          BE    RDIRERR          DIRECTORY ERROR IF YES
2D72 9D23      3544          SSR   R2,R3
2D74 2081      3545          BTBS  8,1              BUSY CHECK
2D76 D925 0000 3546  RDHL    RH    R2,0(R5)         READ HALFWORDS
2D7A 2652      3547          AIS   R5,2
2D7C 0556      3548          CLAR  R5,R6
2D7E 2084      3549          BLS   RDHL             LOOP
2D80 9D23      3550          SSR   R2,R3
2D82 4250 2DC0 3551          BTC   5,RDIRERR        READ ERROR
2D86 2450      3552          LIS   R5,0             SET UP BXLE REGISTERS
2D88 2464      3553          LIS   R6,4
2D8A C870 007C 3554          LHI   R7,124
2D8E 4835 385C 3555  RDIRECT1 LH   R3,INBUF(R5)     LOAD A SEQUENCE NUMBER
2D92 4210 2DB8 3556          BM    EOD1             END OF DIRECTORY
2D96 2339      3557          BZ    EOV1             END OF VOLUME
2D98 0539      3558          CLAR  R3,R9            MATCH SEQUENCE NUMBER?
2D9A 233C      3559          BE    RDIRECT3         BRANCH IF YES
2D9C C150 2D8E 3560          BXLE  R5,RDIRECT1      LOOP THRU DIRECTORY
2DA0 41C0 2ABC 3561          BAL   R12,SULH56
2DA4 4300 2D68 3562          B     RDIRECT2         GO TO NEXT BLOCK
               3563  *
2DA8 2541      3564  EOV1    LCS   R4,1
2DAA 2430      3565          LIS   R3,0             EOV FLAG
2DAC 41D0 2B12 3566  EOVEX   BAL   R13,STOP
2DB0 030E      3567          BR    R14
2DB2 4845 385E 3568  RDIRECT3 LH   R4,INBUF+2(R5)   GET PDB POINTER
2DB6 2205      3569          BS    EOVEX            EXIT
2DB8 41D0 2B12 3570  EOD1    BAL   R13,STOP
2DBC 4300 2A7C 3571          B     EOD
2DC0 C500 0005 3572  RDIRERR CLHI  R0,5              MAX RETRY?
2DC4 4380 2A70 3573          BNL   RDIRERR1         HARD ERROR IF YES
2DC8 2601      3574          AIS   R0,1             BUMP RETRY COUNTER
2DCA 41D0 2B12 3575          BAL   R13,STOP
2DCE 084A      3576          LDAR  R4,R10           RESET LRN POINTER
2DD0 27A1      3577          SIS   R10,1            DECREMENT CURRENT
2DD2 4300 2D52 3578          B     RDLOOP


               3580  * S U B R O U T I N E  R E A D S E C T
               3581  *
2DD6 2470      3582  READSECT LIS   R7,0
2DD8 2302      3583          B     SECT
               3584  * S U B R O U T I N E  W R I T S E C T
2DDA 2471      3585  WRITSECT LIS   R7,1
2DDC 4070 1A52 3586  SECT    STH   R7,DRWFLAG
2DE0 41D0 2B5C 3587          BAL   R13,CONTIDLE     WAIT FOR CONTROLLER IDLE
2DE4 41D0 2B20 3588          BAL   R13,FILESET      SET UP FILE
2DE8 DE20 1A39 3589          OC    R2,SEEK          SEEK
2DEC 41D0 2B4C 3590          BAL   R13,DISKWAIT
2DF0 2415      3591          LIS   R1,5
2DF2 4010 1A56 3592          STH   R1,RETRY         SET RETRY COUNTER
2DF6 0815      3593  SECT1   LDAR  R1,R5            START ADDRESS
```

LEVEL 14 SUBROUTINES

```
  2DF8  CA10 00FF   3594        AHI   R1,255        NEW END ADDRESS(ONE SECTOR)
  2DFC  DE40 1A30   3595        OC    R4,STOPS      SELCH STOP
  2E00  9845        3596        WHR   R4,R5         SEND START ADDRESS
  2E02  9841        3597        WHR   R4,R1         SEND END ADDRESS
  2E04  4870 1A68   3598        LH    R7,DISKTYPE   GET DISK TYPE FLAG
  2E08  C570 0002   3599        CLHI  R7,2          2.5 OR 10 MB
* 2E0C  233A        3600        BE    SECT2         BRANCH IF YES
  2E0E  0819        3601        LDAR  R1,R9         'HEAD'
  2E10  911A        3602        SLLS  R1,10
  2E12  0618        3603        OAR   R1,R8         'CYLINDER'
  2E14  C410 7FFF   3604        NHI   R1,X'7FFF'    CLEAR UNUSED BITS
  2E18  9A3A        3605        WDR   R3,R10        WRITE SECTOR NUMBER TO CONTROLLER
  2E1A  9831        3606        WHR   R3,R1         WRITE HEAD.CYLINDER INFO
  2E1C  9829        3607        WHR   R2,R9         SEND HEAD NO. TO DRIVE
  2E1E  2307        3608        BS    SECT3
  2E20  0819        3609 SECT2  LDAR  R1,R9         HEAD
  2E22  9115        3610        SLLS  R1,5
  2E24  061A        3611        OAR   R1,R10        SECTOR & HEAD
  2E26  C410 003F   3612        NHI   R1,X'3F'      CLEAR UNUSED BITS
  2E2A  9A31        3613        WDR   R3,R1         SEND TO CONTROLLER
  2E2C  4870 1A52   3614 SECT3  LH    R7,DRWFLAG
* 2E30  2136        3615        BNZ   WSECT         WRITE SECTOR
  2E32  DE30 1A31   3616        OC    R3,DREAD      READ SECTOR
  2E36  DE40 1A3C   3617        OC    R4,SREAD      SELCH READ
* 2E3A  230A        3618        B     SSTA
  2E3C  9021        3619 WSECT  SSR   R2,R1
  2E3E  C310 0080   3620        THI   R1,X'80'
  2E42  4230 2A9A   3621        BNZ   WPROT         WRITE PROTECTED!
  2E46  DE30 1A32   3622        OC    R3,DWRITE     DISK WRITE
  2E4A  DE40 1A33   3623        OC    R4,SWRITE     SELCH WRITE
  2E4E  9041        3624 SSTA   SSR   R4,R1
  2E50  2081        3625        BTBS  8,1           SELCH BUSY
  2E52  DE40 1A30   3626        OC    R4,STOPS
  2E56  9947        3627        RHR   R4,R7         GET FINAL ADDRESS
  2E58  C775 00FF   3628        XHI   R7,255(R5)    COMPARE TO START PLUS 255
  2E5C  4070 1A54   3629        STH   R7,SELERR     LOGICAL COMPARE TO CHECK LATER
  2E60  0803        3630        LDAR  R0,R3         CONTROLLER ADDRESS
  2E62  9031        3631 SSTA1  SSR   R3,R1
  2E64  C310 0080   3632        THI   R1,X'80'      WRITE PROTECT OR OVERRUN
  2E68  4230 2E8A   3633        BNZ   AGAIN
  2E6C  C310 0002   3634        THI   R1,2
  2E70  2237        3635        BZS   SSTA1         LOOP UNTIL IDLE
  2E72  C310 0061   3636        THI   R1,X'61'      HD/ADR FAIL, DEF SEC, DEF TRK?
* 2E76  213A        3637        BNZ   AGAIN         TRY AGAIN IF ERROR
  2E78  0804        3638        LDAR  R0,R4         SELCH ADDRESS
  2E7A  4810 1A54   3639        LH    R1,SELERR     RESULT OF FINAL ADDRESS CHECK
* 2E7E  2136        3640        BNZ   AGAIN         TRY AGAIN IF SELCH XFER ERROR
  2E80  41D0 2B4C   3641        BAL   R13,DISKWAIT  GOOD TRANSFER, WAIT FOR IDLE
  2E84  DE40 1A30   3642        OC    R4,STOPS
  2E88  030E        3643        BR    R14           RETURN
  2E8A  4870 1A56   3644 AGAIN  LH    R7,RETRY
  2E8E  2771        3645        SIS   R7,1
  2E90  4070 1A56   3646        STH   R7,RETRY
```

LEVEL 14 SUBROUTINES

```
2E94  4310 2DF6          3647          BNM   SECT1
2E98  4300 2AA2          3648          B     MTERXY


                         3650  * S U B R O U T I N E  C O M P A R E
                         3651  *
2E9C  C860 0100          3652  COMPAREX LHI   R6,256
2EA0  C850 3C5C          3653  COMPAREY LDAI  R5,CHKBUF
2EA4  2410               3654  COMPARE  LIS   R1,0              INDEX
2EA6  4871 385C          3655  COMPAREL LH    R7,INBUF(R1)      DATA READ
2EAA  4575 0000          3656           CLH   R7,0(R5)          COMPARED TO EXPECTED
2EAE  023E               3657           BNER  R14               RETURN ON NON MATCH
2EB0  2652               3658           AIS   R5,2              PATTERN ADDRERSS
2EB2  2612               3659           AIS   R1,2              BUMP INDEX
2EB4  0516               3660           CLAR  R1,R6             DONE?
*2EB6 2088               3661           BL    COMPAREL          LOOP
2EB8  2410               3662           LIS   R1,0              CLEAR CONDITION CODE
2EBA  030E               3663           BR    R14               RETURN/COMPLETE MATCH
```

LEVEL 15 SUBROUTINES

```
                                3665  * S U B R O U T I N E   N O M O T I O N
                                3666  *
2EBC  0802                      3667  NOMOTION LDAR  R0,R2
2EBE  9D21                      3668           SSR   R2,R1
2EC0  4210 2A84                 3669           BTC   1,DU              DU ERROR
2EC4  C310 0010                 3670           THI   R1,X'10'          NO MOTION SET?
2EC8  2236                      3671           BZS   NOMOTION          WAIT FOR IT
2ECA  030F                      3672           BR    R15               RETURN


                                3674  * S U B R O U T I N E   D I R S E R C H
                                3675  *
2ECC  0380 1A14                 3676  DIRSERCH LB    R8,DIRSTART        DIRECTORY CYLINDER
2ED0  2490                      3677           LIS   R9,0              HEAD#
2ED2  24A0                      3678           LIS   R10,0             SECTOR
2ED4  41C0 2AC6                 3679           BAL   R12,SULH56.2      SET START & END ADDRESSES
2ED8  41E0 2DD6                 3680           BAL   R14,READSECT
2EDC  48E0 385C                 3681           LH    R14,INBUF
2EE0  C5E0 FEEE                 3682           CLHI  R14,X'EEEE'       GOOD DIRECTORY?
2EE4  4230 2A70                 3683           BNE   RDIRERR1          ERROR IF NO
2EE8  2450                      3684  DIRSRCH1 LIS   R5,0              INDEX
2EEA  0365 385E                 3685  DIRSRCH2 LB    R6,INBUF+2(R5)     LOOK AT LS SEQUENCE
2EEE  0866                      3686           LDAR  R6,R6             ZERO?
2EF0  033F                      3687           BZR   R15               EXIT IF YES
2EF2  0460 375E                 3688           CLB   R6,PDB+2          MATCH CURRENT SEQUENCE?
* 2EF6  2139                    3689           BNE   DIRSRCH3          SKIP IF NO
2EF8  4865 385C                 3690           LH    R6,INBUF(R5)      MS SEQUENCE
2EFC  4560 375C                 3691           CLH   R6,PDB            MATCH?
* 2F00  2134                    3692           BNE   DIRSRCH3          SKIP IF NO
2F02  4860 1AFC                 3693           LH    R6,SEQNUM         SET R6 = HEX SEQUENCE
2F06  030F                      3694           BR    R15               AND RETURN
2F08  2658                      3695  DIRSRCH3 AIS   R5,8              NEXT ENTRY IN DIRECTORY
2F0A  C550 0100                 3696           CLHI  R5,256            OUT OF SECTOR?
2F0E  4280 2EEA                 3697           BL    DIRSRCH2          NO, KEEP LOOKING
2F12  26A1                      3698           AIS   R10,1             YES, INCREMENT SECTOR NUMBER
2F14  4870 1A68                 3699           LH    R7,DISKTYPE
2F18  45A7 1A8C                 3700           CLH   R10,SECTAB(R7)    LIMIT CHECK
* 2F1C  2188                    3701           BL    DIRSRCH4
2F1E  24A0                      3702           LIS   R10,0             BACK TO SECTOR ZERO
2F20  2691                      3703           AIS   R9,1              ON THE NEXT HEAD
2F22  4597 1A9A                 3704           CLH   R9,HDTAB(R7)      LAST HEAD?
2F26  2183                      3705           BLS   DIRSRCH4          SKIP IF NO
2F28  2561                      3706           LCS   R6,1              R6 GETS -1
2F2A  030F                      3707           BR    R15               RETURN, DIRECTORY FULL
2F2C  41C0 2AC6                 3708  DIRSRCH4 BAL   R12,SULH56.2
2F30  41E0 2DD6                 3709           BAL   R14,READSECT      READ NEXT DIRECTORY SECTOR
2F34  4300 2EE8                 3710           B     DIRSRCH1          GO LOOK AT IT


                                3712  * S U B R O U T I N E   R N X T S E C T
                                3713  *
```

LEVEL 15 SUBROUTINES

```
2F38  2470          3714  RNXTSECT LIS   R7,0
2F3A  2302          3715           BS    RWNEXT
                    3716  *  S U B R O U T I N E   W N X T S E C T
                    3717  *
2F3C  2471          3718  WNXTSECT LIS   R7,1
2F3E  4070 1A52     3719  RWNEXT   STH   R7,DRWFLAG
2F42  41D0 2B20     3720  NEXTSEC1 BAL   R13,FILFSET
2F46  DE20 1A39     3721           OC    R2,SEEK
2F4A  41D0 2B4C     3722           BAL   R13,DISKWAIT
2F4E  4870 1A68     3723  NEXTSEC2 LH    R7,DISKTYPE
2F52  45A7 1A8C     3724           CLH   R10,SECTAB(R7)      BEYOND MAX SECTOR?
*2F56  218A          3725           BL    NEXTSEC4
2F58  24A0          3726           LIS   R10,0               RESET TO SECTOR 0
2F5A  2691          3727           AIS   R9,1                NEXT HEAD
2F5C  4597 1A9A     3728           CLH   R9,HDTAB(R7)        BEYOND MAX HEAD
2F60  2185          3729           BLS   NEXTSEC4            BRANCH IF NO
2F62  2490          3730  NEXTSEC3 LIS   R9,0               SWITCH BACK TO HEAD ZERO
2F64  2681          3731           AIS   R8,1                NEXT CYLINDER
2F66  4300 2F42     3732           B     NEXTSEC1            GO SEEK IT
2F6A  41E0 2C70     3733  NEXTSEC4 BAL   R14,READCHCK       READ CHECK THIS SECTOR
2F6E  C310 0020     3734           THI   R1,X'20'            TEST FOR DEFECTIVE SECTOR
2F72  2334          3735           BZS   NEXTSEC5            GO READ OR WRITE IF OK
2F74  26A1          3736           AIS   R10,1
2F76  4300 2F4E     3737           B     NEXTSEC2            DEFECTIVE SECTOR
2F7A  4870 1A52     3738  NEXTSEC5 LH    R7,DRWFLAG
2F7E  2334          3739           BZS   RSECT
2F80  41E0 2DDA     3740           BAL   R14,WRITSECT
2F84  2303          3741           BS    NEXTDISK
2F86  41E0 2DD6     3742  RSECT    BAL   R14,READSECT
2F8A  4870 1A68     3743  NEXTDISK LH    R7,DISKTYPE         POINT TO NEXT SECTOR
2F8E  26A1          3744           AIS   R10,1
2F90  45A7 1A8C     3745           CLH   R10,SECTAB(R7)
*2F94  2188          3746           BL    NEXTEXIT            CLEAR CC AND LEAVE
2F96  24A0          3747           LIS   R10,0               SECTOR ZERO
2F98  2691          3748           AIS   R9,1                NEXT HEAD
2F9A  4597 1A9A     3749           CLH   R9,HDTAB(R7)        LIMIT?
*2F9E  2183          3750           BL    NEXTEXIT            NO, RESET CC & LEAVE
2FA0  2490          3751           LIS   R9,0                RESET HEAD
2FA2  2681          3752           AIS   R8,1                NEXT CYLINDER
2FA4  2400          3753  NEXTEXIT LIS   R0,0               CLEAR CONDITION CODE
2FA6  030F          3754           BR    R15                 RETURN


                    3756  * S U B R O U T I N E   F O R M L B A
                    3757  *
2FA8  4870 1A68     3758  FORMLBA  LH    R7,DISKTYPE
2FAC  2400          3759           LIS   R0,0                R0,R1 = ACCUMULATOR
2FAE  2410          3760           LIS   R1,0
2FB0  08B8          3761           LDAR  R11,R8
2FB2  27B1          3762  FORMLBA1 SIS   R11,1               R0,R1 GETS CYLINDER NUMBER
2FB4  2186          3763           BLS   FORMLBA2            TIMES THE NUMBER OF SECTORS
2FB6  4A17 1AB6     3764           AH    R1,LBATAB(R7)       ON EACH CYLINDER, ACCORDING
```

LEVEL 15 SUBROUTINES

```
2FBA  2284          3765            BNCS  FORMLBA1          TO DISK TYPE.
2FBC  2601          3766            AIS   R0,1
2FBE  2206          3767            BS    FORMLBA1
2FC0  08B9          3768  FORMLBA2  LDAR  R11,R9            NOW ADD HEAD NUMBER TIMES
2FC2  27B1          3769  FORMLBA3  SIS   R11,1             THE NUMBER OF SECTORS ON
2FC4  2186          3770            BLS   FORMLBA4          EACH TRACK, ACCORDING TO
2FC6  4A17 1A8C     3771            AH    R1,SECTAB(R7)     DISK TYPE
2FCA  2284          3772            BNCS  FORMLBA3
2FCC  2601          3773            AIS   R0,1
2FCE  2206          3774            BS    FORMLBA3
2FD0  0A1A          3775  FORMLBA4  AAR   R1,R10            FINALLY ADD THE SECTOR NUMBER
2FD2  038F          3776            BNCR  R15
2FD4  2601          3777            AIS   R0,1
2FD6  030F          3778            BR    R15


                    3780  * S U B R O U T I N E   D I S K S E T
                    3781  *
2FD8  41D0 2B20     3782  DISKSET   BAL   R13,FILESET        SET UP FILE...R7 GETS DISKTYPE
2FDC  0E20 1A39     3783            OC    R2,SEEK
2FE0  41D0 284C     3784            BAL   R13,DISKWAIT      WAIT: CONTROLLER IDLE, DISKRSRW
2FE4  41E0 2C70     3785            BAL   R14,READCHCK      READ CHECK THE SECTOR
2FE8  C310 0020     3786            THI   R1,X'20'          DEFECTIVE SECTOR?
2FEC  4230 1C4C     3787            BNZ   MDERR001          DEFECTIVE MEDIA
2FF0  41C0 2AC6     3788            BAL   R12,SULH56,2      INPUF, INBUF+255
2FF4  4810 1A18     3789            LH    R1,VERIFLAG
* 2FF8  2135         3790            BNZ   DISKSET1          SKIP IF VERIFY
2FFA  41E0 2DDA     3791            BAL   R14,WRITSECT      WRITE THIS SECTOR
2FFE  4300 2F8A     3792            B     NEXTDISK          BUMP POINTERS
                    3793  *
3002  C850 3C5C     3794  DISKSET1  LDAI  R5,CHKBUF
3006  C865 00FF     3795            LHI   R6,255(R5)
300A  41E0 2DD6     3796            BAL   R14,READSECT      READ THE VOLUME DESCRIPTOR
300E  41E0 2E9C     3797            BAL   R14,COMPAREX      COMPARE INBUF & CHKBUF
3012  023F          3798            BNER  R15               RETURN IF BAD COMPARE
3014  4300 2F8A     3799            B     NEXTDISK          BUMP POINTERS IF GOOD
```

FMD BOOT LOADER

```
      0000 3018          3801  STARTAD  EQU   *
3018  2440               3802  BOOTST   LIS   R4,0
301A  2303               3803           BS    BOOT1
301C  4000               3804           DCX   4000
301E  4010               3805           DCX   4010
3020  4040 0022          3806  BOOT1    STH   R4,X'22'            REGISTER SAVE POINTER(16BIT)
3024  C840 001C          3807           LHI   R4,28               LOAD DIRECTORY START LRN
3028  D310 0078          3808           LB    R1,X'78'            LOAD DEVICE ADDRESS
302C  D320 0079          3809           LB    R2,X'79'
3030  C420 0030          3810           NHI   R2,X'30'            REMOVE DRIVE SELECT BITS
3034  C620 00C7          3811           OHI   R2,X'C7'            FORM STOP COMMAND
3038  C850 00D0          3812           LHI   R5,X'D0'            START ADDRESS
303C  C860 01F7          3813           LHI   R6,ENDAD-STARTAD+X'D0'
                         3814  * LOAD REST OF BOOT LOADER
3040  9D13               3815  BOOT1B   SSR   R1,R3
3042  2081               3816           BTBS  8,1                 WAIT ON BUSY
3044  D915 0000          3817           RH    R1,0(R5)            READ HALFWORDS
3048  2652               3818           AIS   R5,2                BUMP INDEX
304A  0565               3819           CLAR  R6,R5
304C  2286               3820           BNLS  BOOT1B              LOOP
304E  9D13               3821           SSR   R1,R3               CHECK FINAL STATUS
3050  2152               3822           BTFS  5,REDOBL            RETRY BOOT LOAD
3052  230E               3823           BS    STOPA
3054  C850 0500          3824  REDOBL   LHI   R5,X'D500'
3058  4050 0050          3825           STH   R5,X'50'
305C  C850 00CF          3826           LHI   R5,X'CF'
3060  4050 0052          3827           STH   R5,X'52'
3064  9E12               3828           OCR   R1,R2               STOP
3066  9D13               3829           SSR   R1,R3
3068  2221               3830           BFBS  2,1                 WAIT FOR IDLE
306A  4300               3831           DCX   4300,0050           B   X'50'
306C  0050
306E  9E12               3832  STOPA    OCR   R1,R2               STOP COMMAND
3070  9D13               3833  IDLE     SSR   R1,R3
3072  2221               3834           BFBS  2,1                 LOOP ON NOT IDLE
3074  C850 01A8          3835           LDAI  R5,LDBUF            START ADDRESS
3078  C860 01C8          3836           LDAI  R6,LDBUF+32
307C  9814               3837           WHR   R1,R4               WRITE LRN TO CONTROLLER
307E  2726               3838           SIS   R2,6                FORM READ COMMAND
3080  9E12               3839           OCR   R1,R2               ISSUE READ COMMAND
                         3840  * READ DIRECTORY
3082  9D13               3841  BOOT1C   SSR   R1,R3
3084  2081               3842           BTBS  8,1
3086  D915 0000          3843           RH    R1,0(R5)
308A  2652               3844           AIS   R5,2
308C  0565               3845           CLAR  R6,R5
308E  2286               3846           BNLS  BOOT1C              LOOP
3090  9D13               3847           SSR   R1,R3
3092  2152               3848           BTFS  5,REDO              ERROR, RETRY
3094  2305               3849           BS    STOP1
3096  2626               3850  REDO     AIS   R2,6
3098  9E12               3851           OCR   R1,R2               ISSUE STOP COMMAND
309A  4300 00D8          3852           B     IDLE-STARTAD+X'80'  GO TO IDLE
```

FMD BOOT LOADER

```
309E  2626                3853 STOP1   AIS   R2,6
30A0  9E12                3854         OCR   R1,R2            ISUE STOP COMMAND
30A2  9D13                3855         SSR   R1,R3
30A4  2221                3856         BFBS  2,1             WAIT FOR IDLE
30A6  C840 4000           3857         LHI   R4,X'4000'      TEST PATTERN
30AA  0A44                3858         AAR   R4,R4
30AC  2115                3859         BMS   IS16            BRANCH, 16 BIT HOST
30AE  2470                3860         LIS   R7,0            SET 32 BIT FLAG
30B0  4840 01AE           3861         LH    R4,LDBUF+6      32 BIT PDB POINTER
30B4  2304                3862         BS    COM
30B6  4840 01AA           3863 IS16    LH    R4,LDBUF+2      16 BIT PDB POINTER
30BA  2472                3864         LIS   R7,2            16 BIT FLAG
30BC  C850 01A8           3865 COM     LDAI  R5,LDBUF        START ADDRESS
30C0  C860 0227           3866         LDAI  R6,LDBUF+127    END ADDRESS
30C4  9814                3867 COM1    WHR   R1,R4           PDB LRN TO CONTROLLER
30C6  2726                3868         SIS   R2,6
30C8  9E12                3869         OCR   R1,R2           ISSUE READ COMMAND
                          3870 * READ PDB
30CA  9D13                3871 BOOT1D  SSR   R1,R3
30CC  2081                3872         BTBS  8,1             WAIT ON BUSY
30CE  D915 0000           3873         RH    R1,0(R5)
30D2  2652                3874         AIS   R5,2
30D4  0565                3875         CLAR  R6,R5
30D6  2286                3876         BNLS  BOOT1D          LOOP
30D8  9D13                3877         SSR   R1,R3
30DA  2152                3878         BTFS  5,REDO1         ERROR RETRY
30DC  2307                3879         BS    RDLDRGEN        ELSE READ PROGRAM
30DE  2626                3880 REDO1   AIS   R2,6
30E0  9E12                3881         OCR   R1,R2           ISSUE STOP COMMAND
30E2  9D13                3882         SSR   R1,R3
30E4  2221                3883         BFBS  2,1             WAIT FOR IDLE
30E6  4300 0124           3884         B     COM-STARTAD+X'80'  GO TO COM
30EA  2626                3885 RDLDRGEN AIS  R2,6
30EC  9E12                3886         OCR   R1,R2           ISSUE STOP COMMAND
30EE  9D13                3887         SSR   R1,R3
30F0  2221                3888         BFBS  2,1             WAIT FOR IDLE
30F2  2726                3889         SIS   R2,6
30F4  2641                3890         AIS   R4,1
30F6  4850 01B0           3891 REDO3   LH    R5,LDBUF+8      LOAD LOW
30FA  4050 01A0           3892         STH   R5,BOOTEN16-STARTAD+X'80'
30FE  4050 01A6           3893         STH   R5,BOOTEN32-STARTAD+X'80'
3102  4860 01B4           3894         LH    R6,LDBUF+12     LOAD HIGH
3106  9814                3895         WHR   R1,R4           WRITE LRN TO CONTROLLER
3108  9E12                3896         OCR   R1,R2           ISSUE READ COMMAND
                          3897 * READ LOADER-GENERATOR PROGRAM
310A  9D13                3898 BOOT1E  SSR   R1,R3
310C  2081                3899         BTBS  8,1             LOOP ON BUSY
310E  D915 0000           3900         RH    R1,0(R5)
3112  2652                3901         AIS   R5,2
3114  0565                3902         CLAR  R6,R5
3116  2286                3903         BNLS  BOOT1E          LOOP
3118  9D13                3904         SSR   R1,R3
311A  2152                3905         BTFS  5,REDO2         ERROR RETRY
```

FMD BOOT LOADER

```
311C   2307                3906            BS     TURNOVER
311E   2626                3907  REDO2     AIS    R2,6
3120   9E12                3908            OCR    R1,R2              ISSUE STOP COMMAND
3122   9D13                3909            SSR    R1,R3
3124   2221                3910            BFBS   2,1                WAIT FOR IDLE
3126   4300 015E           3911            B      REDO3-STARTAD+X'80' GO TO REDO3
312A   2626                3912  TURNOVER  AIS    R2,6
312C   9E12                3913            OCR    R1,R2              ISSUE STOP COMMAND
312E   9D13                3914            SSR    R1,R3
3130   2221                3915            BFBS   2,1                WAIT FOR IDLE
3132   0877                3916            LDAR   R7,R7              TEST HOST FLAG
3134   2333                3917            BZS    GO32               BRANCH, 32 BIT
3136   4300                3918            DCX    4300               ELSE TAKE 16 BIT BRANCH
3138   6000                3919  BOOTEN16  DCX    6000
313A   4300                3920  GO32      DCX    4300
313C   4000                3921            DCX    4000               RX3 BRANCH
313E   6000                3922  BOOTEN32  DCX    6000               32 BIT START
       0000 313F           3923  ENDAD     EQU    *-1
                           3924  *
```

MAG TAPE BOOT LOADER

```
3140  0310 0078          3927  MTBOOT   LB     R1,X'78'          GET TAPE DEVICE ADDRESS
3144  2420               3928           LIS    R2,0
3146  0330 007D          3929           LB     R3,X'7D'          PICK UP SELCH ADDRESS
314A  9423               3930           EXBR   R2,R3             (R2)='SS00' OR '0000SS00'
314C  EC20 0008          3931           SRL    R2,8              IF 16 BIT, (R2,R3)='00SS,0000'
                         3932  *                                 IF 32 BIT, (R2)=(R3)='000000SS'
3150  9D19               3933  MTBOOT1  SSR    R1,R9             STATUS CHECK
3152  9198               3934           SLHLS  R9,11             NO MOTION BIT TO CARRY
                         3935  *                                 LS BYTE OF R9 = 00
3154  2282               3936           BNCS   MTBOOT1           WAIT FOR NO-MOTION
3156  C850 0100          3937           LHI    R5,X'100'         LOAD START ADDRESS
315A  2461               3938           LIS    R6,1              BXLE INCREMENT
315C  C860 0263          3939           LHI    R6,MTLOADE-MTLOADS+X'100' LOAD END ADDRESS
3160  C800 0030          3940           LHI    R0,X'30'          SELCH READ COMMAND
3164  2448               3941           LIS    R4,8              SELCH STOP COMMAND
3166  9E24               3942           OCR    R2,R4             SELCH STOP
3168  9A39               3943           WDR    R3,R9             MS BYTE OF 3 BYTE ADDRESS
                         3944  *                                 R3=SELCH ADDRESS IF 32 BIT HOST
316A  9825               3945           WHR    R2,R5             LS 2 BYTES OF START ADDRESS
316C  9A39               3946           WDR    R3,R9             MS BYTE OF 3 BYTE END ADDRESS
316E  9826               3947           WHR    R2,R6             LS 2 BYTES OF END ADDRESS
3170  DE10 0079          3948           OC     R1,X'79'          MAG TAPE WRITE
3174  9E30               3949           OCR    R3,R0             SELCH GO
3176  2145               3950           BOS    MTBOOT2           FALSE SYNC = NO SELCH
3178  9D29               3951           SSR    R2,R9             ELSE, WAIT FOR SELCH NOT BUSY
317A  2081               3952           BTBS   8,1               LOOP ON BUSY
317C  9E24               3953           OCR    R2,R4             SELCH STOP
317E  0305               3954           BR     R5                BRANCH, START CODE JUST LOADED
3180  9D19               3955  MTBOOT2  SSR    R1,R9             MAG TAPE STATUS
3182  2081               3956           BTBS   8,1               LOOP ON BUSY
3184  DB15 0000          3957           RD     R1,0(R5)          READ A BYTE
3188  C150 3180          3958           BXLE   R5,MTBOOT2        DECREMENT INDEX & LOOP
318C  4300 0100          3959           B      X'100'            GO TO LOAD START ADRS
3190  0000               3960           DCX    0000              FILLER
3192  0000               3961           DCX    0000              FILLER
```

```
                        3963  * THE FIRST PART OF THE BOOT LOADER RESIDES IN MEMORY
                        3964  * FROM LOCATION X'80' TO X'CF'.  IT IS LOADED BY THE
                        3965  * X'50' SEQUENCE.  WHEN CONTROL IS TRANSFERRED TO IT,
                        3966  * THAT CODE READS IN THIS NEXT SECTION WHICH STARTS
                        3967  * AT ADDRESS X'100'.

                        3969  * ON INPUT, (R0)=SELCH READ COMMAND, X'30'
                        3970  *           (R1)=MAG TAPE DEVICE ADDRESS
                        3971  *           (R2)=SELCH ADDRESS
                        3972  *           (R3)=0 IF 16 BIT HOST
                        3973  *           (R3)=SELCH ADDRESS IF 32 BIT HOST
                        3974  *           (R4)=SELCH STOP COMMAND, X'08'
                        3975  *           (R6)=1
                        3976  *
                        3977  *
3194  D310 0078         3978  MTLOADS  LB    R1,X'78'          GET TAPE DEVICE ADDRESS
3198  9D19              3979  MTLOADS0 SSR   R1,R9             MAG TAPE STATUS
319A  C390 0010         3980           THI   R9,X'10'          TEST FOR NO MOTION
319E  2233              3981           BZS   MTLOADS0          WAIT FOR IT
31A0  C8F0 A023         3982           LHI   R15,X'A023'       FF COMMANDS
31A4  C8E0 D8C0         3983           LHI   R14,X'D8C0'       DISARM COMMANDS
31A8  D390 0079         3984           LB    R9,X'79'          GET READ COMMAND
31AC  C590 00A1         3985           CLHI  R9,X'A1'          800/1600 BPI?                 *
31B0  2333              3986           BES   MTLOADS1          SKIP IF YES                   *
31B2  90F8              3987           SRHLS R15,8             FF COMMAND FOR 6250           *
31B4  90E8              3988           SRHLS R14,8             DISARM COMMAND FOR 6250       *
31B6  9E1F              3989  MTLOADS1 OCR   R1,R15            DISARM
31B8  9E1E              3990           OCR   R1,R14            FORWARD FILE MARK
31BA  9D19              3991  MTLOAD2  SSR   R1,R9
31BC  C390 0010         3992           THI   R9,X'10'          NO MOTION?
31C0  2233              3993           BZS   MTLOAD2           NO, WAIT
31C2  0833              3994           LDAR  R3,R3             CHECK HOST
31C4  2336              3995           BZS   MTLOAD4           SKIP IF 16 BIT HOST
31C6  9E1E              3996           OCR   R1,R14            ELSE, ANOTHER FORWARD FILE
31C8  9D19              3997  MTLOAD3  SSR   R1,R9
31CA  C390 0010         3998           THI   R9,X'10'          NO MOTION CHECK
31CE  2233              3999           BZS   MTLOAD3
                        4000  *                               TAPE IS NOW IN POSITION AT THE
                        4001  *                               BEGINNING OF THE FIRST OR SECOND
                        4002  *                               PROGRAM, DEPENDING ON THE HOST.
                        4003  *                               NEXT, READ IN PART OF THAT PROGRAM'S
                        4004  *                               PROGRAM DEFINITION BLOCK (PDB)
                        4005  *
31D0  C850 0264         4006  MTLOAD4  LHI   R5,BOOTEN-MTLOADS+X'100' START ADRS
31D4  C860 0281         4007           LHI   R6,BOOTEN-MTLOADS+X'110' END ADRS
31D8  9E24              4008           OCR   R2,R4             SELCH STOP
31DA  2490              4009           LIS   R9,0
31DC  9A39              4010           WDR   R3,R9             MS BYTE OF 3 BYTE START ADRS
31DE  9825              4011           WHR   R2,R5             LS 2 BYTES OF START ADRS
31E0  9A39              4012           WDR   R3,R9             MS BYTE OF 3 BYTE END ADRS
31E2  9826              4013           WHR   R2,R6             LS 2 BYTES OF END ADDRESS
31E4  DE10 0079         4014           OC    R1,X'79'          START TAPE
31E8  9E30              4015           OCR   R3,R0             SELCH GO
```

MAG TAPE ROOT LOADER

```
31EA  2145            4016             BOS   MTLOAD5              SKIP IF FALSE SYNC
31EC  9D29            4017             SSR   R2,R9                ELSE WAIT FOR SELCH
31EE  2081            4018             BTBS  8,1                  TO GO NON-BUSY
31F0  9E24            4019             OCR   R2,R4                THEN STOP THE SELCH
31F2  230A            4020             BS    MTLOAD6
31F4  9019            4021   MTLOAD5   SSR   R1,R9                MAG TAPE STATUS
31F6  2081            4022             BTBS  8,1                  WAIT FOR NON-BUST
31F8  DB15 0000       4023             RD    R1,0(R5)             READ
31FC  2651            4024             AIS   R5,1                 BUMP INDEX
31FE  0565            4025             CLAR  R6,R5                DONE?
3200  2286            4026             BNLS  MTLOAD5              NO, LOOP
3202  9019            4027             SSR   R1,R9                FINAL STATUS
3204  2170            4028             BTFS  7,0                  HANG ON ERROR
3206  9019            4029   MTLOAD6   SSR   R1,R9                TRANSFER COMPLETE
3208  C390 0010       4030             THI   R9,X'10'             WAIT FOR NO MOTION
320C  2233            4031             BZS   MTLOAD6              WAIT FOR NO MOTION
320E  D390 0276       4032             LB    R9,BOOTEN-MTLOADS+X'100'+18
3212  D350 0277       4033             LB    R5,BOOTEN-MTLOADS+X'100'+19
3216  9158            4034             SLHLS R5,8                 POSITION BITS 16-23
3218  D360 0278       4035             LB    R6,BOOTEN-MTLOADS+X'100'+20
321C  0656            4036             OAR   R5,R6                (R9,R5)=PROGRAM START ADDRESS
321E  0833            4037             LDAR  R3,R3                TEST HOST
3220  2334            4038             BZS   MTLOAD6A             SKIP IF 16 BIT
                      4039   *         EXHR  R7,R9                (R7)='00XX0000'
3222  3479            4040             DCX   3479
3224  0675            4041             OAR   R7,R5                (R7)='00XXYYZZ'
3226  2302            4042             BS    MTLOAD6B
3228  0875            4043   MTLOAD6A  LDAR  R7,R5                SAVE START ADDRESS
322A  0857            4044   MTLOAD6B  LDAR  R5,R7                (R5)=START ADDRESS
322C  D3A0 0279       4045             LB    R10,BOOTEN-MTLOADS+X'100'+21
3230  D360 027A       4046             LB    R6,BOOTEN-MTLOADS+X'100'+22
3234  9466            4047             EXBR  R6,R6
3236  D380 027B       4048             LB    R8,BOOTEN-MTLOADS+X'100'+23
323A  0668            4049             OAR   R6,R8                (R10,R6)=LOAD END ADDRESS
323C  0833            4050             LDAR  R3,R3                TEST HOST
323E  2230            4051             BZS   MTLOAD6C             SKIP IF 16 BIT
**  U002  **   MTLOAD6C
                      4052   *         EXHR  R11,R10              (R11)='00XX0000'
3240  348A            4053             DCX   348A
3242  066B            4054             OAR   R6,R11               (R6)='00XXYYZZ'
3244  0885            4055   MTLOAD7   LDAR  R8,R5                START ADRS
3246  CA80 00FF       4056             AHI   R8,X'FF'             PLUS 255
324A  0568            4057             CLAR  R6,R8                COMPARE TO END ADDRESS
324C  2382            4058             BNLS  MTLOAD7A             SKIP IF NOT LESS
324E  0886            4059             LDAR  R8,R6                IF YES, USE REAL END ADDRESS
3250  9E24            4060   MTLOAD7A  OCR   R2,R4                SELCH STOP
3252  9A39            4061             WDR   R3,R9                MS BYTE OF ADDRESS
3254  9825            4062             WHR   R2,R5                OUTPUT START ADRS
3256  9A3A            4063             WDR   R3,R10
3258  9826            4064             WHR   R2,R6                OUTPUT END ADRS
325A  DE10 0079       4065             OC    R1,X'79'             MAG TAPE START
325E  9E20            4066             OCR   R2,R0                SELCH START
3260  2145            4067             BOS   MTLOAD8              SKIP IF FALSE SYNC
```

PREVIOUS ERROR ON PAGE 34

MAG TAPE BOOT LOADER

```
3262  9D29              4068          SSR   R2,R9
3264  2081              4069          BTBS  8,1               WAIT ON SELCH BUSY
3266  9E24              4070          OCR   R2,R4             SELCH STOP
3268  2308              4071          BS    MTLOAD9
326A  9D19              4072  MTLOAD8 SSR   R1,R9             MAG TAPE STATUS
326C  2081              4073          BTBS  8,1               LOOP ON BUSY
326E  DB15 0000         4074          RD    R1,0(R5)          READ BYTES
3272  2651              4075          AIS   R5,1              BUMP ADDRESS
3274  0585              4076          CLAR  R8,R5             DONE CHECK
3276  2286              4077          BNLS  MTLOAD8           LOOP ON RECORD
3278  D390 0079         4078  MTLOAD9 LB    R9,X'79'          GET READ COMMAND
327C  C590 00A1         4079          CLHI  R9,X'A1'          800/1600 BPI?
3280  2334              4080          BES   MTLOAD9B          SKIP IF YES
3282  9D19              4081  MTLOAD9A SSR  R1,R9             6250 STATUS
3284  212D              4082          BTCS  2,MTLOAD10        DONE IF EOF
3286  2307              4083          BS    MTLOAD9C          ELSE KEEP READING
3288  9D19              4084  MTLOAD9B SSR  R1,R9
328A  C390 0040         4085          THI   R9,X'40'          EOF?
328E  2138              4086          BNZS  MTLOAD10          DONE IF YES
3290  9095              4087  MTLOAD9C SRHLS R9,5             NO MOTION CHECK
**  1001  **
3292  228D              4088          BNCS  MTLOAD9           WAIT FOR IT
3294  C858 0001         4089  MTLOAD9C LHI  R5,1(R8)         NEXT START ADRS
**  1001  **
3298  0556              4090          CLAR  R5,R6             COMPARE TO END ADDRESS/
329A  4280 01B0         4091          BL    MTLOAD7-MTLOADS+X'100' LOOP
329E  0857              4092  MTLOAD10 LDAR R5,R7             START ADDRESS
                        4093  *                               (R6)=END ADDRESS
32A0  24A0              4094          LIS   R10,0             CHECKSUM ACCUMULATOR
32A2  D3B7 0000         4095  MTLOAD11 LB   R11,0(R7)
32A6  07AB              4096          XAR   R10,R11           CALCULATE CHECKSUM
32A8  2671              4097          AIS   R7,1
32AA  0567              4098          CLAR  R6,R7
32AC  2285              4099          BNLS  MTLOAD11
32AE  9D19              4100  MTLOAD12 SSR  R1,R9
32B0  C390 0010         4101          THI   R9,X'10'          NO MOTION CHECK
32B4  2233              4102          BZS   MTLOAD12
32B6  C870 0038         4103          LHI   R7,X'38'
32BA  D390 0079         4104          LB    R9,X'79'          GET READ COMMAND
32BE  C590 00A1         4105          CLHI  R9,X'A1'          800/1600 BPI?
32C2  2333              4106          BES   MTLOAD13          SKIP IF YES
32C4  C870 00E0         4107          LHI   R7,X'E0'          RW COMMAND FOR 6250
32C8  9E17              4108  MTLOAD13 OCR  R1,R7             REWIND THE TAPE
32CA  9D19              4109  MTLOAD14 SSR  R1,R9
32CC  C390 0010         4110          THI   R9,X'10'          NO MOTION CHECK
32D0  2233              4111          BZS   MTLOAD14
32D2  D370 027C         4112          LB    R7,BOOTEN-MTLOADS+X'100'+24
32D6  057A              4113          CLAR  R7,R10            CHECK CHECKSUM
32D8  0335              4114          BER   R5                GO IF OK
32DA  C810 00EE         4115          LHI   R1,X'EE'
32DE  24A1              4116          LIS   R10,1
32E0  C8B0 0040         4117          LHI   R11,X'40'
32E4  9EAB              4118          OCR   R10,R11           DISPLAY IN INCREMENTAL MODE
```

PREVIOUS ERROR ON PAGE 86

MAG TAPE BOOT LOADER

```
32E6   9AA1             4119          WDR   R10,R1
32E8   24E0             4120          LIS   R14,0
32EA   9AAE             4121          WDR   R10,R14
32EC   9AAE             4122          WDR   R10,R14
32EE   9AAE             4123          WDR   R10,R14
32F0   C8B0 0080        4124          LHI   R11,X'80'
32F4   9EAB             4125          OCR   R10,R11          NORMAL MODE
32F6   2200             4126          BS    *                HANG ON CHECKSUM ERROR
       0000 32F7        4127  MTLOADE EQU   *-1
       0000 32F8        4128  BOOTEN  EQU   *
```

```
32F8  4552 524F 5220 4F4E  4130  DIRER     DB   C'ERROR ON DIRECTORY UPDATE',13,10,0
3300  2044 4952 4543 544F
3308  5259 2055 5044 4154
3310  450D 0A00
3314  4445 4645 4354 4956  4131  MEDMSG    DB   C'DEFECTIVE OUTPUT MEDIA',13,10,0
331C  4520 4F55 5450 5554
3324  204D 4544 4941 0D0A
332C  00
332D  454E 4420 4F46 2056  4132  EOVMSG2   DB   C'END OF VOLUME',13,10,0
3335  4F4C 554D 450D 0A00
333D  4449 5245 4354 4F52  4133  RDEMSG    DB   C'DIRECTORY READ ERROR',13,10,0
3345  5920 5245 4144 2045
334D  5252 4F52 0D0A 00
3354  454F 5620 4E4F 5420  4134  EOVRTNM   DB   C'EOV NOT FOUND',13,10,0
335C  464F 554E 440D 0A00
3364  4449 5348 4554 5445  4135  TRIGMSG   DB   C'DISKETTE FULL',13,10,0
336C  2046 554C 4C0D 0A00
3374  4449 5245 4354 4F52  4136  EODMSG    DB   C'DIRECTORY FULL',13,10,0
337C  5920 4655 4C4C 0D0A
3384  00
3385  4455 504C 4943 4154  4137  GOTMSG    DB   C'DUPLICATE SEQUENCE NUMBER',13,10,0
338D  4520 5345 5155 454E
3395  4345 204E 554D 4245
339D  520D 0A00
33A1  424F 4F54 204C 4F41  4138  VERRMSG1  DB   C'BOOT LOADER '
33A9  4445 5220
33AD  5645 5249 4659 2045  4139  CHKERRM   DB   C'VERIFY ERROR',13,10,0
33B5  5252 4F52 0D0A 00
33BC  494E 5641 4C49 4420  4140  SPECERR   DB   C'INVALID OUTDEV SPECIFICATION',13,10,0
33C4  4F55 5444 4556 2053
33CC  5045 4349 4649 4341
33D4  5449 4F4E 0D0A 00
33D8  494E 5641 4C49 4420  4141  ERRMSG1   DB   C'INVALID DIRECTORY ON OUTPUT DISK',13,10,0
33E3  4449 5245 4354 4F52
33EB  5920 4F4E 204F 5554
33F3  5055 5420 4449 5348
33FB  0D0A 00
33FE  5044 4220 5645 5249  4142  VERRMSG2  DB   C'PDB VERIFY ERROR',13,10,0
3406  4659 2045 5252 4F52
340E  0D0A 00
3411  4F2E 532E 2049 4D41  4143  OSERRM    DB   C'O.S. IMAGE VERIFY ERROR',13,10,0
3419  4745 2056 4552 4946
3421  5920 4552 524F 520D
3429  0A00
342B  4249 5420 4D41 5020  4144  OSERRM2   DB   C'BIT MAP VERIFY ERROR',13,10,0
3433  5645 5249 4659 2045
343B  5252 4F52 0D0A 00
3442  564F 4C55 4D45 2044  4145  OSERRM3   DB   C'VOLUME DESCRIPTOR VERIFY ERROR',13,10,0
344A  4553 4352 4950 544F
3452  5220 5645 5249 4659
345A  2045 5252 4F52 0D0A
3462  00
3463  494E 4445 562F 4F55  4146  SPECERR1  DB   C'INDEV/OUTDEV CONFLICT',13,10,0
3468  5444 4556 2043 4F4E
3473  464C 4943 540D 0A00
```

```
347B 494E 5641 4C49 4420  4147  SPECERR2 DB    C'INVALID INDEV SPECIFICATION',13,10,0
3483 494E 4445 5620 5350
348B 4543 4946 4943 4154
3493 494F 4E0D 0A00
3499 4445 5649 4345 2057  4148  WPROTMSG DB    C'DEVICE WRITE PROTECTED',13,10,0
34A1 5249 5445 2050 524F
34A9 5445 4354 4544 0D0A
34B1 00
34B2 4445 5649 4345 2055  4149  DEVUNA   DB    C'DEVICE UNAVAILABLE',13,10,0
34BA 4E41 5641 494C 4142
34C2 4C45 0D0A 00
34C7 554E 5245 434F 5645  4150  UNRECOV  DB    C'UNRECOVERABLE ERROR',13,10,0
34CF 5241 424C 4520 4552
34D7 524F 520D 0A00
34DD 4E4F 2053 5543 4820  4151  ABSENTM  DB    C'NO SUCH SEQUENCE ON OUTPUT MEDIA',13,10,0
34E5 5345 5155 454E 4345
34ED 204F 4E20 4F55 5450
34F5 5554 204D 4544 4941
34FD 0D0A 00
3500 0C                   4152  HEADER   DB    X'0C'               FORM FEED
3501 5345 5120 20         4153           DB    C'SEQ  '
3506 4F42 4A45 4354 204E  4154           DB    C'OBJECT NUMBER  '
350E 554D 4245 5220 20
3515 504B 4720            4155           DB    C'PKG '
3519 5052 4F47 5241 4020  4156           DB    C'PROGRAM TITLE'
3521 5449 544C 45
3526 2020 2020 2020 2020  4157           DB    C'          '
352E 2020
3530 2020 2020 2020 2020  4158           DB    C'          '
3538 20
3539 4C4F 5720 2020 2048  4159           DB    C'LOW    HIGH   FLAG'
3541 4947 4820 2020 464C
3549 4147
3548 0D0A 0D0A 00         4160           DB    X'0D',X'0A',X'0D',X'0A',0
     0000 354F            4161  LNZB     EQU   *-1
3550                      4162  PRINTOUT DS    80
35A0                      4163           ALIGN 8
35A0                      4164  $TBRKSV  DS    8
35A8                      4165  $R15SAV  DS    4
35AC                      4166  $R14SAV  DS    8
35B4                      4167  R5SAV    DS    4
35B8                      4168  R6SAVE   DS    4
35BC                      4169  $OUTBUF  DS    $BUFLEN
360C                      4170  $INBUF   DS    $BUFLEN
365C                      4171           ALIGN 4
365C                      4172  IDIRBLK  DS    256
375C                      4173  PDB      DS    256
385C                      4174  INBUF    DS    1024
3C5C                      4175  CHKBUF   DS    1024
405C                      4176  REGISTER DS    64
     0000 409C            4177  PSWSAVE  EQU   *
409C                      4178  RSAVE    DS    64
40DC                      4179  INTSAV   DS    64
411C                      4180           END
```

ASSEMBLED BY CAL 03-066R08-00 (32-BIT)

START OPTIONS:. T=16,ERLST

9 CAL ERRORS PREVIOUS ERROR ON PAGE 87
NO CAL WARNINGS
6 PASSES

TABLE SPACE USED : 21K

```
DIRERR1   0000 2A68      3223*
DIRSERCH  0000 2ECC      2871   3108   3676*
DIRSRCH1  0000 2EE8      3684*  3710
DIRSRCH2  0000 2EEA      3685*  3697
DIRSRCH3  0000 2F08      3689   3692   3695*
DIRSRCH4  0000 2F2C      3701   3705   3708*
DIRSTART  0000 1A14      1924*  2336   2412   3194   3676
DISABLE   0000 1A2C      1940*  1941
DISK      0000 1FB8      2392   2404*
DISKCHK   0000 18DE      1717   1739   1775*
DISKCHK1  0000 18E2      1777*
DISKCHK2  0000 18EA      1778   1781*  1784   1789
DISKCHK3  0000 18EE      1780   1783*
DISKCHK4  0000 1922      1802*  1833   1856
DISKCHK5  0000 1962      1828   1834*
DISKCHK6  0000 1992      1851   1853*
DISKERR1  0000 1E9C      2317   2320*
DISKERR2  0000 1EA6      2270   2280   2290   2324*
DISKETT1  0000 265E      2913   2921*
DISKETT2  0000 266A      2920   2924*
DISKETTE  0000 263A      2735   2911*
DISKFIN   0000 28C0      3082   3108*
DISKOUT   0000 27BE      2981   3025*
DISKSET   0000 2FC8      2269   2279   2289   2316   3207   3782*
DISKSET1  0000 3002      3790   3794*
DISKTYP1  0000 1A6A      1724   1979*  2406   2956
DISKTYP2  0000 1A6C      1747   1980*  2084   2779   2834   3030
DISKTYPE  0000 1A68      1978*  2085   2241   2255   2375   2407   2780   2957   3031   3304   3598   3699   3723
                         3743   3758
DISKWAIT  0000 2B4C      1814   1837   2341   2352   3324*  3456   3590   3641   3722   3784
DISPLAY   0000 0D36       476*
DREAD     0000 1A31      1945*  3616
DRESET    0000 1A35      1785   1853   1949*  2236   3305
DRWFLAG   0000 1A52      1967*  3341   3347   3362   3382   3586   3614   3719   3738
DSCRPT    0000 16BC      1642*  2300   2302
DSIZE     0000 0030      1910*  1913   1914   1914
DTAB      0000 1AE0      1723   1746   2046*
DU        0000 2A84      3232*  3328   3669
DUPEPROG  0000 2670      2764   2774   2852   2893   2927*  3062
DWRITE    0000 1A32      1946*  1947   3622
ENDAD     0000 313F      1919   2168   2178   2181   3813   3923*
ENDDIR    0000 1A12      1922*  2162   3097
ENDOVOL   0000 2A3E      2464   3212*
ENDOVOL1  0000 2A42      2430   3213*  3414
EOD       0000 2A7C      3229*  3571
EOD1      0000 2DB8      3556   3570*
EODMSG    0000 3374      3229   4136*
EOTMSG    0000 1620      1616*
EOV1      0000 2DA8      3557   3564*
EOVEX     0000 2DAC      3566*  3569
EOVMSG2   0000 332D      3213   4132*
EOVRTN    0000 2420      2710   2753*  2877
EOVRTNM   0000 3354      2753   4134*
ERR       0000 0D5C       495*  1521
ERR1      0000 0E00       539    563*
```

```
  MMSW        0000 1544    1403   1406   1411   1413   1529   1532   1550*
  MMSWMSG     0000 15DA    1541   1608*  1609
  MOD32       0000 1548     188    317    606    623   1221   1247   1263   1363   1401   1442   1501   1530   1552*
  MREADC      0000 0A44     153*
  MSMCLFLT    0000 1A3D    1957*  2244
  MTBOOT      0000 3140    2204   2218   3927*
  MTBOOT1     0000 3150    3933*  3936
  MTBOOT2     0000 3180    3950   3955*  3958
  MTCHCK      0000 18C2    1715   1735   1764*
  MTERR       0000 2C1C    3375   3392   3409*  3411
  MTERR1      0000 2C2E    3416*
  MTERR2      0000 2C42    3422*  3424
  MTERXY      0000 2AA2    3241*  3420   3648
  MTESTNO     0000 1590     532   1595*
  MTLOAD10    0000 329E    4082   4086   4092*
  MTLOAD11    0000 32A2    4095*  4099
  MTLOAD12    0000 32AE    4100*  4102
  MTLOAD13    0000 32C8    4106   4108*
  MTLOAD14    0000 32CA    4109*  4111
  MTLOAD2     0000 31BA    3991*  3993
  MTLOAD3     0000 31C8    3997*  3999
  MTLOAD4     0000 31D0    3995   4006*
  MTLOAD5     0000 31F4    4016   4021*  4026
  MTLOAD6     0000 3206    4020   4029*  4031
  MTLOAD6A    0000 3228    4038   4043*
  MTLOAD6B    0000 322A    4042   4044*
U MTLOAD6C    0000 0000    4051   4051
  MTLOAD7     0000 3244    4055*  4091
  MTLOAD7A    0000 3250    4058   4060*
  MTLOAD8     0000 326A    4067   4072*  4077
  MTLOAD9     0000 3278    4071   4078*  4088
  MTLOAD9A    0000 3282    4081*
  MTLOAD9B    0000 3288    4080   4084*
M MTLOAD9C    0000 3294    4083   4087*  4089*
  MTLOADE     0000 32F7    2208   2223   2226   3939   4127*
  MTLOADS     0000 3194    2207   2223   2225   2226   3939   3978*  4006   4007   4032   4033   4035   4045   4046
                           4048   4091   4112
  MTLOADS0    0000 3198    3979*  3981
  MTLOADS1    0000 31B6    3986   3989*
  MTREAD      0000 1A2E    1942*  1943   3364   3384
  MTWRITE     0000 1A41    1961*  3396   3405
  NAME        0000 1728    1653*
  NAMELOOP    0000 2166    2532*  2537   2540   2542
  NAMELP01    0000 2180    2538   2541*
  NEWPSW      0000 1538     437    438   1548*
  NEWSTART    0000 23DA    2728   2730*
  NEXTCYL     0000 1A1A    1931*  2093   2310   2788   2843   2878   2890   3027   3041   3050
  NEXTDISK    0000 2F8A    2309   2329   2334   2437   2889   3741   3743*  3792   3799
  NEXTEXIT    0000 2FA4    3746   3750   3753*
  NEXTHEAD    0000 1A1C    1932*  2094   2311   2789   2844   2879   2891   3028   3042   3051
  NEXTLRN     0000 1A28    1938*  2171   2730   2911   2925   2982   2991   3001
  NEXTSEC1    0000 2F42    3720*  3732
  NEXTSEC2    0000 2F4E    3723*  3737
  NEXTSEC3    0000 2F62    3730*
  NEXTSEC4    0000 2F6A    3725   3729   3733*
```

			879	923	924	926	929	931	934	935	936	938	941	942	943
			945	947	948	950	953	955	957	1027	1030	1032	1036	1047	1048
			1049	1050	1052	1054	1056	1058	1059	1138	1139	1142	1143	1146	1147
			1148	1152	1153	1155	1157	1159	1161	1161	1202	1203	1204	1205	1206
			1207	1208	1214	1215	1216	1217	1225	1226	1227	1255	1257	1258	1263
			1338	1382	1383	1432	1433	1439	1441	1515	1516	1529	1685	1688	1820
			1821	1822	1823	1843	1844	1845	1846	2092	2093	2113	2115	2118	2125
			2126	2237	2238	2260	2264	2265	2266	2275	2276	2277	2308	2355	2358
			2583	2587	2590	2591	2594	2596	2609	2622	2626	2627	2629	2629	2632
			2644	2648	2652	2656	2796	3126	3133	3193	3199	3203	3242	3290	3295
			3296	3297	3298	3306	3312	3316	3324	3327	3334	3357	3358	3365	3371
			3372	3374	3387	3390	3391	3393	3397	3409	3410	3412	3413	3422	3429
			3449	3450	3451	3452	3458	3459	3460	3461	3469	3470	3472	3505	3506
			3508	3537	3538	3540	3591	3592	3593	3594	3597	3601	3602	3603	3604
			3606	3609	3610	3611	3612	3613	3619	3620	3624	3631	3632	3634	3636
			3639	3654	3655	3659	3660	3662	3668	3670	3734	3760	3764	3771	3775
			3786	3789	3808	3815	3817	3821	3828	3829	3832	3833	3837	3839	3841
			3843	3847	3851	3854	3855	3867	3869	3871	3873	3877	3881	3882	3886
			3887	3895	3896	3898	3900	3904	3908	3909	3913	3914	3927	3933	3948
			3955	3957	3978	3979	3989	3990	3991	3996	3997	4014	4021	4023	4027
			4029	4065	4072	4074	4081	4084	4100	4108	4109	4115	4119		
R10	0000	000A	90*	1331	1335	1335	1380	1380	1381	1813	1819	1836	1842	2249	2250
			2259	2297	2312	2315	2338	2353	2376	2377	2414	2416	2424	2425	2426
			2427	2433	2440	2716	2718	2747	2790	2845	2880	2883	2892	2962	2969
			3029	3043	3052	3088	3197	3206	3448	3460	3532	3541	3542	3576	3577
			3605	3611	3678	3698	3700	3702	3724	3726	3736	3744	3745	3747	3775
			4045	4063	4094	4096	4113	4116	4118	4119	4121	4122	4123	4125	
R11	0000	000B	91*	2261	2264	2274	2275	2408	2419	2420	2421	2422	2428	2429	2431
			2432	2433	2448	2456	2457	2458	2462	2463	2465	3761	3762	3768	3769
			4054	4095	4096	4117	4118	4124	4125						
R12	0000	000C	92*	229	234	267	273	284	287	292	302	308	315	359	361
			420	537	540	666	2116	2150	2151	2155	2158	2163	2281	2282	2298
			2299	2342	2343	2357	2714	2720	2723	2740	2743	2786	2787	2853	2854
			2895	2914	2935	2944	2952	2970	3094	3099	3102	3121	3136	3137	3253
			3259	3270	3280	3286	3535	3561	3679	3708	3788				
R13	0000	000D	93*	538	541	819	822	982	983	984	1012	1015	1020	1398	1398
			1411	1414	1814	1815	1837	1838	2111	2128	2132	2149	2206	2209	2217
			2224	2251	2339	2341	2350	2352	2401	2479	2480	2483	2485	2488	2489
			2490	2491	2492	2493	2494	2495	2496	2497	2498	2501	2502	2503	2507
			2508	2514	2518	2544	2545	2546	2548	2549	2550	2451	2553	2555	2556
			2558	2763	2769	2821	2830	2943	3017	3022	3212	3215	3225	3291	3300
			3318	3320	3330	3336	3351	3378	3441	3447	3454	3456	3457	3468	3485
			3487	3504	3521	3534	3566	3570	3575	3587	3588	3590	3641	3720	3722
			3782	3784											
R14	0000	000E	94*	174	176	177	190	268	269	271	293	294	295	296	300
			306	317	357	432	437	516	539	540	541	563	570	573	579
			582	588	595	601	602	603	617	619	630	631	635	636	637
			639	640	642	643	655	660	676	1072	1084	1092	1095	1101	1105
			1114	1116	1122	1125	1127	1131	1212	1230	1233	1235	1237	1240	1242
			1244	1249	1252	1254	1270	1273	1288	1291	1294	1296	1330	1337	1348
			1399	1410	1505	1507	1517	1524	1684	1685	1687	1713	1715	1717	1735
			1737	1739	1767	1773	1781	1781	1782	1803	1859	1863	1874	2153	2156
			2159	2164	2169	2179	2182	2220	2227	2354	2360	2369	2415	2416	2417
			2429	2436	2453	2471	2559	2563	2701	2721	2724	2772	2804	2805	2806
			2807	2808	2809	2810	2811	2812	2813	2814	2815	2816	2817	2822	2829

		2859	2888	2896	2899	2915	2918	2923	2951	2987	2997	3004	3083	3095
		3100	3103	3122	3223	3236	3246	3455	3486	3497	3522	3567	3643	3657
		3663	3680	3681	3682	3709	3733	3740	3742	3785	3791	3796	3797	3983
		3988	3990	3996	4120	4121	4122	4123						
R15	0000 000F	95*	175	191	196	197	199	206	207	214	216	218	223	271
		283	288	290	298	310	311	314	324	365	386	393	398	399
		409	422	424	425	429	433	434	435	436	436	441	453	455
		463	466	469	478	482	487	495	496	499	500	504	505	509
		510	515	517	521	522	535	536	537	538	551	554	555	592
		612	620	623	628	629	638	641	661	662	664	668	722	729
		821	832	837	839	862	863	864	865	876	894	901	908	910
		913	916	922	927	959	960	1046	1076	1077	1085	1086	1087	1090
		1091	1093	1096	1097	1099	1101	1103	1105	1106	1106	1110	1111	1112
		1115	1117	1118	1122	1123	1125	1127	1128	1129	1131	1132	1163	1169
		1175	1231	1238	1245	1269	1272	1279	1280	1287	1290	1293	1297	1338
		1384	1390	1510	1518	1520	1521	1539	1540	1687	1703	1704	1705	1706
		1707	1708	1709	1721	1722	1723	1731	1743	1745	1746	1757	1759	1771
		1776	1804	1852	1872	1878	2081	2138	2139	2142	2144	2183	2185	2196
		2198	2202	2211	2221	2228	2269	2270	2279	2280	2289	2290	2306	2309
		2316	2317	2320	2324	2330	2332	2333	2397	2437	2460	2461	2476	2477
		2560	2561	2562	2563	2564	2604	2612	2625	2646	2650	2654	2658	2673
		2677	2684	2686	2687	2688	2707	2757	2773	2791	2823	2824	2825	2826
		2827	2828	2842	2858	2860	2863	2865	2867	2871	2874	2889	2900	2903
		2908	2919	2963	3003	3009	3037	3046	3108	3125	3191	3198	3207	3209
		3221	3222	3233	3234	3243	3244	3494	3495	3672	3687	3694	3707	3754
		3776	3778	3798	3982	3987	3989							
R15SAVE	0000 1A6E	290	310	1703	1757	1981*								
R2	0000 0002	82*	208	209	210	363	365	370	371	374	385	392	461	465
		468	591	610	627	637	691	765	766	811	814	817	818	819
		820	822	823	828	829	830	929	934	1028	1033	1035	1037	1038
		1042	1043	1044	1054	1055	1056	1057	1064	1064	1065	1067	1147	1148
		1329	1346	1349	1472	1479	1494	1505	1538	1710	1732	1744	1758	1764
		1768	1775	1777	1794	1796	1816	1829	1839	1853	1860	1865	1866	1869
		1870	2079	2080	2082	2083	2117	2118	2119	2121	2125	2195	2197	2201
		2203	2210	2212	2237	2244	2252	2340	2351	2387	2398	2500	2504	2505
		2513	2517	2562	2570	2571	2572	2573	2602	2610	2623	2645	2649	2653
		2657	2691	2692	2693	2762	2766	2831	2938	2976	3216	3290	3297	3298
		3305	3310	3311	3314	3315	3319	3326	3427	3357	3364	3365	3367	3371
		3384	3396	3397	3399	3405	3409	3412	3416	3421	3422	3439	3471	3472
		3473	3476	3483	3493	3507	3508	3509	3512	3519	3539	3540	3544	3546
		3550	3589	3607	3619	3667	3668	3721	3783	3809	3810	3811	3828	3832
		3838	3839	3850	3851	3853	3854	3868	3869	3880	3881	3885	3886	3889
		3896	3907	3908	3912	3913	3928	3930	3931	3942	3945	3947	3951	3953
		4008	4011	4013	4017	4019	4060	4062	4064	4066	4068	4070		
R3	0000 0003	83*	232	236	238	244	246	247	252	366	370	383	389	400
		401	653	654	656	673	674	675	692	758	759	760	762	767
		806	808	812	994	1029	1030	1033	1050	1051	1052	1053	1065	1066
		1067	1070	1329	1347	1711	1733	1772	1772	1777	1779	1783	1817	1819
		1823	1824	1825	1831	1840	1842	1846	1847	1848	1854	1858	1858	2084
		2085	2096	2097	2099	2106	2107	2108	2109	2110	2119	2147	2199	2233
		2236	2386	2388	2390	2404	2443	2444	2445	2446	2447	2499	2512	2516
		2530	2532	2534	2539	2564	2567	2568	2569	2569	2571	2575	2576	2694
		2695	2697	2699	2700	2705	2709	2709	2777	2832	2939	2940	2946	2948
		2949	2956	2957	2958	2977	2978	2980	2983	2984	2985	3025	3306	3312
		3316	3324	3334	3417	3418	3419	3431	3438	3448	3452	3453	3461	3475

```
                                3509  3544  3550  3555  3558  3565  3605  3606  3613  3616  3622  3630  3631
                                3815  3821  3829  3833  3841  3847  3855  3871  3877  3882  3887  3898  3904
                                3909  3914  3929  3930  3943  3946  3949  3994  3994  4010  4012  4015  4037
                                4037  4050  4050  4061  4063
R4          0000 0004            84*   215   217   220   236   241   246   248   250   266   272   297   303
                                 355   367   368   377   378   410   411   413   419   457   458   459   653
                                 657   659   662   761   762   763   764   764   765   816   817   827   828
                                 893   895   895   909   911   952   985   986   988   996   998   999  1001
                                1003  1004  1006  1008  1010  1015  1016  1059  1060  1061  1062  1062  1068
                                1068  1186  1261  1268  1282  1283  1284  1682  1712  1734  1779  1788  2076
                                2077  2090  2091  2112  2114  2117  2129  2133  2152  2154  2157  2160  2161
                                2162  2166  2170  2171  2172  2176  2193  2194  2234  2235  2394  2396  2405
                                2450  2454  2455  2463  2465  2466  2467  2468  2510  2511  2515  2519  2520
                                2521  2523  2524  2525  2526  2527  2528  2529  2531  2533  2535  2536  2541
                                2605  2606  2607  2608  2613  2614  2616  2617  2618  2619  2620  2621  2630
                                2631  2637  2638  2659  2660  2661  2661  2662  2663  2664  2665  2666  2675
                                2676  2713  2718  2719  2722  2726  2756  2778  2833  2911  2924  2925  2942
                                2950  2959  2982  2990  2991  3000  3001  3014  3026  3088  3097  3101  3218
                                3360  3360  3379  3380  3381  3385  3387  3389  3390  3406  3430  3436  3457
                                3471  3507  3533  3539  3564  3568  3576  3595  3596  3597  3617  3623  3624
                                3626  3627  3638  3642  3802  3806  3807  3837  3857  3858  3858  3861  3863
                                3867  3890  3895  3941  3942  3953  4008  4019  4060  4070
R5          0000 0005            85*   233   237   239   241   564   606   611   618   640   806   808   859
                                 864   869   893   897   898   900   914  1018  1719  1741  1760  1793  1795
                                1796  1797  1817  1825  1827  1831  1840  1848  1850  1854  2121  2122  2123
                                2136  2167  2177  2176  2180  2186  2204  2205  2207  2215  2218  2222  2225
                                2225  2321  2327  2331  2367  2399  2400  2406  2407  2410  2411  2434  2435
                                2451  2452  2469  2470  2579  2580  2581  2633  2635  2636  2637  2638  2639
                                2640  2711  2711  2713  2725  2726  2727  2729  2730  2731  2732  2733  2734
                                2737  2745  2746  2747  2750  2753  2760  2761  2767  2768  2770  2856  2857
                                2864  2866  2884  2885  2897  2904  2907  2912  2916  2921  2922  2928  2929
                                2932  2934  2937  2964  2965  2974  2975  2994  2995  2996  3008  3010  3011
                                3019  3020  3021  3034  3035  3036  3038  3039  3047  3048  3055  3060  3061
                                3085  3087  3089  3092  3093  3104  3112  3114  3116  3118  3120  3213  3217
                                3220  3226  3229  3239  3251  3252  3257  3258  3264  3265  3266  3267  3269
                                3350  3353  3367  3368  3369  3376  3380  3393  3399  3400  3401  3425  3467
                                3476  3477  3480  3491  3503  3512  3513  3517  3526  3546  3547  3548  3552
                                3555  3560  3568  3593  3596  3628  3653  3656  3658  3684  3685  3690  3695
                                3696  3794  3795  3812  3817  3818  3819  3824  3825  3826  3827  3835  3843
                                3844  3845  3865  3873  3874  3875  3891  3892  3893  3900  3901  3902  3937
                                3945  3954  3957  3958  4006  4011  4023  4024  4025  4035  4034  4036  4041
                                4043  4044  4055  4062  4074  4075  4076  4089  4090  4092  4114
R5SAV       0000 35B4           4167*
R5SAVE      0000 1884           2067*  3264  3269  3350  3425  3467  3491  3503  3526
R6          0000 0006            86*   235   237   243   274   282   286   291   299   301   305   307   313
                                 319   322   358   360   360   362   649   667   668   675  1038  1039  1040
                                1040  1372  1377  1378  1379  1386  1387  1389  1450  1455  1460  1465  1470
                                1484  1492  1503  1508  1512  1513  1514  1522  1523  1525  1688  1689  1795
                                1797  1799  2123  2168  2178  2181  2205  2208  2216  2219  2223  2226  2400
                                2411  2435  2452  2470  2761  2768  2771  2857  2872  2876  2876  2885  2898
                                2917  2922  2927  2931  2933  2934  2936  2937  2965  2972  2973  2975  2988
                                2989  2990  2994  2996  2998  2999  3000  3002  3019  3021  3034  3036  3039
                                3048  3056  3057  3057  3059  3060  3064  3065  3070  3072  3073  3074  3076
                                3078  3079  3081  3084  3085  3086  3087  3091  3092  3093  3109  3109  3111
                                3112  3113  3114  3115  3116  3117  3118  3119  3120  3138  3140  3143  3145
```

			3147	3149	3151	3153	3158	3159	3160	3161	3162	3163	3165	3175	3177
			3179	3181	3183	3185	3187	3189	3252	3258	3267	3274	3354	3356	3376
			3480	3517	3548	3553	3652	3660	3685	3686	3688	3690	3691	3693	
			3706	3795	3813	3819	3836	3845	3866	3875	3894	3902	3938	3939	3947
			4007	4013	4025	4035	4036	4046	4047	4047	4049	4054	4057	4059	4064
			4090	4098											
R6SAVE	0000	35B8	2933	2936	4168*										
R7	0000	0007	87*	319	323	651	805	829	835	2241	2242	2245	2255	2256	2283
			2286	2300	2301	2302	2303	2304	2305	2344	2345	2346	2347	2348	2375
			2377	2380	2758	2779	2780	2781	2792	2793	2794	2794	2795	2796	2797
			2798	2801	2802	2819	2820	2830	2834	2835	2835	2836	2837	2838	2840
			2846	2847	2848	2849	2851	2886	2953	2954	3006	3015	3030	3031	3032
			3129	3130	3131	3131	3132	3133	3134	3134	3139	3141	3142	3144	3146
			3148	3150	3152	3154	3155	3156	3157	3164	3166	3167	3168	3169	3170
			3171	3172	3173	3174	3176	3178	3180	3182	3184	3186	3188	3190	3274
			3275	3276	3277	3278	3279	3281	3282	3283	3284	3285	3304	3308	3340
			3341	3346	3347	3362	3382	3445	3475	3478	3511	3514	3554	3582	3585
			3586	3598	3599	3614	3627	3628	3629	3644	3645	3646	3655	3656	3699
			3700	3704	3714	3718	3719	3723	3724	3728	3738	3743	3745	3749	3758
			3764	3771	3860	3864	3916	3916	4041	4043	4044	4092	4095	4097	4098
			4103	4107	4108	4112	4113								
R8	0000	0008	88*	1811	1822	1834	1845	2247	2257	2295	2310	2313	2336	2382	2383
			2412	2431	2438	2745	2788	2843	2878	2881	2890	2960	2967	3027	3041
			3050	3194	3204	3314	3319	3451	3603	3676	3731	3752	3761	4048	4049
			4055	4056	4057	4059	4076	4089							
R9	0000	0009	89*	989	990	992	993	994	996	1182	1812	1820	1835	1843	2248
			2258	2296	2311	2314	2337	2349	2379	2380	2413	2417	2432	2439	2746
			2789	2844	2879	2882	2891	2961	2968	3028	3042	3051	3195	3205	3310
			3449	3458	3536	3558	3601	3607	3609	3677	3703	3704	3727	3728	3730
			3748	3749	3751	3768	3933	3934	3943	3946	3951	3955	3979	3980	3984
			3985	3991	3992	3997	3998	4009	4010	4012	4017	4021	4027	4029	4030
			4032	4061	4068	4072	4078	4079	4081	4084	4085	4087	4100	4101	4104
			4105	4109	4110										
RCHECK	0000	1A34	1824	1847	1948*	1949	3453								
RCNT	0000	0022	1904*												
RDEMSG	0000	333D	3226	4133*											
RDHL	0000	2D76	3546*	3549											
RDIRECT	0000	2D48	2701	3083	3531*										
RDIRECT1	0000	2D8E	3555*	3560											
RDIRECT2	0000	2D68	3541*	3562											
RDIRECT3	0000	2DB2	3559	3568*											
RDIRERR	0000	2DC0	3543	3551	3572*										
RDIRERR1	0000	2A70	3225*	3573	3683										
RDLORGEN	0000	30EA	3879	3885*											
RDLOOP	0000	2D52	3534*	3578											
RDLRN	0000	2CA0	2179	2453	2471	2721	2724	2915	2951	2997	3103	3466*			
RDLRN.01	0000	2CA6	3468*	3492											
RDLRN.02	0000	2CB4	3473*	3481											
RDLRN.03	0000	2CBC	3476*	3479											
RDLRNE	0000	2CD6	3484	3487*											
RDLRNE1	0000	2CEA	3489	3493*											
READCHCK	0000	2C70	2354	3445*	3733	3785									
READCHK1	0000	2C8E	3446	3456*											
READMAG	0000	2B6A	2217	2224	2401	2769	2943	3022	3346*						
READMAG1	0000	2BA4	3365*	3370											

```
TSTMSG     0000 158A      1594*  1595
TURNOVER   0000 312A      3906   3912*
UNARY      0000 0EEE       673*
UNRECOV    0000 34C7      3220   3245   3496   4150*
USEDBITS   0000 1AD2      2037*  2256   2283
UTILITY    0000 1B8A      2070*  2250   2327   2330   2333   2692   2864   2867   3008   3011
VD.ATRB    0000 0004      1884*
VD.FDB     0000 0008      1885*  2307   2308
VD.OSP     0000 000C      1886*
VD.OSS     0000 0010      1887*
VD.VOL     0000 0000      1883*  2301   2303
VERI.OS    0000 2584      2841   2856*
VERIF256   0000 27EA      3037*  3040
VERIFLAG   0000 1A18      1681   1930*  2147   2199   2358   2707   2740   2743   2758   2840   2874   2886   2912
                          2985   3015   3032   3076   3789
VERIFMD2   0000 275A      2986   2994*
VERIFY     0000 16F8       257   1649*
VERIFY01   0000 2772      3002*  3023   3044
VERIMAG2   0000 27AE      3016   3019*
VERIPDB    0000 2608      2887   2895*
VERR.001   0000 1CE8      2183   2185*  2221   2228
VERR.002   0000 2624      2773   2900   2903*  2919
VERRMSG1   0000 33A1      2186   4138*
VERRMSG2   0000 33FE      2904   4142*
VMAGBOOT   0000 1D4C      2200   2215*
VMAGPDB    0000 244A      2759   2766*
VOLUME     0000 16A4      1639*  1689   2485
WBERR      0000 2D36      3520   3523*
WBERR1     0000 2A4A      3215*  3524
WCNT       0000 0020      1903*
WFILEMK    0000 1A40      1960*  1961   2201   2210   2212
WNXTSECT   0000 2F3C      2842   3046   3718*
WPROT      0000 2A9A      3239*  3621
WPROTMSG   0000 3499      3239   4148*
WRITE      0000 1764      1658*
WRITE256   0000 2806      3033   3046*  3049
WRITEIT    0000 2714      2945   2955   2972*
WRITEMAG   0000 2B62      2206   2209   2763   3017   3340*
WRITMAG1   0000 28FE      3397*  3402
WRITSECT   0000 2DDA      2369   2888   3122   3585*  3740   3791
WRLRN      0000 2D40      2153   2156   2159   2164   2169   2923   2987   3095   3100   3502*
WRLRN.01   0000 2D06      3504*  3527
WRLRN.02   0000 2D14      3509*  3518
WRLRN.03   0000 2D1C      3512*  3515
U WRT.CMD  0000 0000      1658   1658
WSECT      0000 2E3C      3615   3619*
XIERR1     0000 1406      1373   1378*
XIERR2     0000 1434      1388   1392*
ZEROBUF1   0000 2AD6      3265*  3268
ZEROBUFF   0000 2AD0      2151   2282   2299   2343   2787   2854   2935   3137   3263*
ZERONE     0000 0B76       282*
```

ERROR & WARNING SUMMARY :

U002 @ LINE 1652
U002 @ LINE 1653
U002 @ LINE 1656
U002 @ LINE 1657
U002 @ LINE 1658
U002 @ LINE 1829
U002 @ LINE 4051
M001 @ LINE 4087
M001 @ LINE 4089