

IDENTIFICATION

Product Code: DEC-12-ZR8A-D
Product Name: DIAL-MS File Commands
Program Description
Date Created: July 1, 1970
Maintainer: Software Services

LAP6-DIAL is an editor, filing system and assembler for use with the PDP-12 computer. The editor and filing portions are derived from the basic LINC program LAP6¹ by Mary Allen Wilkes of Washington University. The assembly portion is derived from several programs used for the PDP-8 computer including PAL-D².

The Digital Equipment Corporation wishes to express to the author, Mary Allen Wilkes (Clark), and the Computer Research Laboratory of Washington University, St. Louis, Missouri, its appreciation for the development set forth in LAP6 as well as its thanks for permission to use parts of the LAP6 program.

¹M. A. Wilkes, LAP6 Handbook, Computer Research Laboratory Tech. Rep. No. 2, Washington University, St. Louis, May 1, 1967.

²PAL-D Assembler Programmer's Reference Manual DEC-D8-ASAA-D.

1.0 PROGRAM OVERVIEW

File Commands, referred to as FILECOMS, is comprised of a series of file handling routines, along with Save Program, Add Binary, and Save Binary in its entirety. FILECOMS is called by the Editor whenever a Print Source, Add Binary, Save Program, or Save Binary is requested, or whenever an assembly is requested of a source file. For any of the above cases, FILECOMS uses the MC parameter table and the index to set up the required arguments. When handling files and setting up arguments, FILECOMS checks to see if the request is possible and/or legitimate. If a command cannot or will not be executed, FILECOMS displays a "NO". Once FILECOMS sets up arguments, control is turned over to the appropriate program.

File Commands is on TBLKs 350, 351, 352, and 353 and in MBLKs 0, 1, 2, and 3 of segment 2 when in core. The Editor communicates with FILECOMS relative to the tag FCSA (FILECOMS Starting Address). To execute a Save Program, for example, the Editor reads in the first block of FILECOMS and executes a JMP FCSA+3. The contents of FCSA+3 are JMP SAVPRG. The FCSA table, which must not be moved without the Editor's knowledge, has the following form:

(4020)	FCSA, JMP SAVBIN	/SAVE BINARY
	JMP ADDBIN	
	JMP .	
	JMP SAVPRG	/SAVE PROGRAM
	JMP LIORAS	/ASSEMBLER
	JMP PRNPRG	/PRINT SOURCE

At location RPLSTR is the "REPLACE?" table which contains the DIAL codes for the phrase "REPLACE?". The "REPLACE?" table is used in conjunction with the Editor's grid pattern display table (A6) to display the phrase "REPLACE?". Because FILECOMS uses the file area extensively, an explanation of index structure is crucial to understanding the secrets of FILECOMS.

The index is located at tape blocks 346 and 347; the LAP6-DIAL system expects it to be two blocks long. The index must be comprised of contiguous tape blocks and the first tape block must be of the form XX6. An index tape block of 352, for example, would not be compatible with a number of LINCtape group instructions that are scattered throughout the system. The left half of the first word on tape block 346 must be the DIAL code for a slash (57), in order to identify the block as an index. The first 10 words of tape block 346 contain 5757 (//) and are never used for file name storage by the DIAL system; the remaining words on both tape blocks are used to store index information. Each file in the index takes up 10 words, so that a DIAL tape can hold a maximum of 77 octal file names. Unfilled portions of the index contain 5757 in all 10 words. For any given 10 word portion, the first four words are reserved for the file name. The DIAL code for the first character of the name is stored in the

left half of the first word; the second character of the name goes in the right half of the first word, etc. The maximum number of characters in a name is eight. Unused half-words in a name = 77 (?). The name FILENAME would be stored in the following manner:

Ø611	FI
14Ø5	LE
16Ø1	NA
15Ø5	ME

The name DIAL would be saved as:

Ø411	DI
Ø114	AL
7777	(Not
7777	Used)

Question marks within a name are treated as ordinary characters. Leading question marks are illegal and may cause unpredictable results. Trailing question marks are ignored.

If a file name refers to a source, the fifth word of the ten word sector holds the first TBLK of the source and the sixth word holds the length. If there is no source, then the fifth and sixth words are both equal to 5757. If a name refers to a binary file, then the seventh word contains the first TBLK of the binary and the eighth word contains its length. If there is no binary, then the seventh and eighth words are both equal to 5757. In words 5, 6, 7, and 10, the low order 9 bits are reserved for tape block and length; bit zero is used to indicate whether source or binary information is present; bits 1 and 2 are not used.

2.0 SAVE PROGRAM AND FILECOMS

When a user requests a Save Program, the Editor exits, reads four tape blocks of FILECOMS (35Ø) into locations 4ØØØ-5777, loads the last tape block used in the Working Area into the AC, and jumps to FCSA+3. Save Program first stores away the complement of the source length at location F1, then checks for line number arguments, displaying "NO" if they have been requested. Next, Save Program reads the index (TBLKs 346, 347) into locations 3ØØØ-3777. At this point, Save Program checks the left half of the first word in the index for a 57; if there is no 57, Save Program fills locations 3ØØØ-3777 with 5757. This move is called making an index. Save Program then calls a subroutine called LOOKUP to check for a name match between the requested name - to be found in the MC parameter table at E6 + 2 to E6 + 5 (2372-2376) - and the index. If there is no name match, Save Program searches the index for a blank entry (a 57 in the left half of the first word). If no blank entry is found, the index is full and Save

INDEX STRUCTURE

TBLK 346

57	57	
57	57	
57	57	
57	57	
57	57	
57	57	
57	57	
57	57	
X ₁	X ₂	INDEX IDENTIFIER (10 WDS)
X ₃	X ₄	
X ₅	X ₆	
X ₇	X ₈	
Y ₁	Y ₂	SOURCE TBLK (1 WD)
Y ₃	Y ₄	SOURCE LENGTH (1 WD)
Z ₁	Z ₂	BINARY TBLK (1 WD)
Z ₃	Z ₄	BINARY LENGTH (1 WD)

TBLK 347

Z ₁	Z ₂	UNUSED HALF WORDS IN A NAME = 77 (?)
Z ₃	Z ₄	
Z ₅	Z ₆	
Z ₇	Z ₈	
Y ₁	Y ₂	FOR NO SOURCE Y ₁ =Y ₂ =Y ₃ =Y ₄ =57 (/)
Y ₃	Y ₄	
Z ₁	Z ₂	FOR NO BINARY Z ₁ =Z ₂ =Z ₃ =Z ₄ =57 (/)
Z ₃	Z ₄	

UNFILLED PORTION OF THE INDEX
5757 IN ALL WORDS

Program displays "NO". If a blank entry is found, locations E6 + 2 to E6 + 5 are placed into its first four words and the gap search commences. If a name match is found, Save Program must determine whether or not the matched entry has an accompanying source (the match could be binary only). To determine source or binary match, Save Program checks bit 0 of the fifth word in the matched entry. If the match is binary only, the gap search commences. If a requested name matches a source, Save Program displays the phrase "REPLACE?" until a key is struck on the Teletype. If the key struck on the Teletype is not an "R", Save Program returns to the Editor; if an R is typed, 5757 is put into the fifth and sixth words of the matched entry to erase the starting TBLK and length information of the old source. Save Program then calls the gap search subroutines.

Once the index has been properly tended to, Save Program must find room in the file area to store the source. There are 600 TBLKs on a DIAL tape that are available for storage. These 600 TBLKs are divided into two sections:

1. The upper file area (TBLKs 470-777)
2. The lower file area (TBLKs 0-267)

Sources are saved in the file area as close to the index as possible.

The gap search subroutines first scan all upper file entries in the index to determine where the gaps are in the upper file. A gap is any unused portion of the file area. If a file named ONE runs from TBLKs 470-517 and another file named TWO runs from TBLKs 550-600, then a 30 TBLK gap exists from TBLKs 520-547. If there are no files stored in the upper file area, then there exists a 310 TBLK gap from TBLKs 470-777. The gap search subroutines check words 5-10 of all index entries to compute the file gaps. The best TBLK in the upper file area is 470 because it is closest to the index. The gap search routines initially set the best TBLK at 470; then they start searching for gaps. If a gap is found that is smaller than the length of the requested entry, the best TBLK is reset to one greater than the last TBLK of the current entry. For example, if the best TBLK is set to 470 and the length of the requested entry is 20 TBLKs and the gap search routines find a source entry (or binary, for that matter) starting in TBLK 500 (word 5) that is fifty TBLKs long (word 6); i.e., the source in question resides in TBLKs 500-547 inclusive, then the gap is from TBLKs 470-477 (10 TBLKs). This is too small to accommodate an entry of 20 TBLKs, so the best TBLK is reset to 550, which is one more than the last TBLK (547) of the current entry. The gap search subroutines scan the index, resetting the best TBLK every time a gap that is too small is encountered.

Once the upper file has been searched, the entire sequence is repeated for the lower file. When searching the lower file, the best TBLK is set to one less than the start of the current entry when an insufficiently long gap is

encountered. Searching the upper and lower files in the manner described above really finds all those TBLKs that will not work. For example, suppose that the best TBLK in the lower file turned out to be 40; this would mean that there are no gaps between TBLKs 41-267 that are large enough to file the requested source.

Once the best TBLKs for upper and lower file have been established by the gap search routines, Save Program must determine whether or not the requested source will fit in the file area. If the length of the requested source is greater than the difference between the lower bound of the lower file (TBLK 0) and the best TBLK of the lower file, and if it is also greater than the difference between the upper bound of the upper file (TBLK 777) and the best TBLK of the upper file, then the requested source is bigger than all gaps in the file area and it will not fit. For this case, Save Program displays "NO". If the requested source will fit in either the upper or lower file area, Save Program determines which of the two best TBLKs is closest to the index before initiating the actual save.

After file area has been allocated by Save Program, the starting tape block of the requested source is stored in the fifth word of the entry and the source length is stored in the sixth word. Save Program then checks the keyboard. If a key has been struck, the request is inhibited and control is turned over to the Editor; if no key has been struck, the Save Program part of Save Program commences.

Save Program transfers 16 (octal) TBLKs at a time from the working area to the file area until the entire source has been saved. The last transfer may be from 1 to 16 TBLKs, depending on the length of the source being saved.

3.0 PRINT SOURCE AND FILECOMS

Whenever a Print Source is requested, FILECOMS must be called ahead of Print Source to set up arguments for Print Source. The Editor executes this call by reading TBLKs 350-353 of FILECOMS into locations 4000-5777 and doing a JMP FCSA+5. FILECOMS first checks the left half of E6 + 2 (2373) in the MC parameter table for a 77 (?). A 77 indicates that the Working Area is to be printed; for this case FILECOMS reads the TBLKs containing Print Source (363) and TTY (364) into locations 5000-5777, moves the source working area unit to E6 + 6, and executes a JMP 1000. If the Print Source is by file name, FILECOMS checks to see if the requested unit has an index, displaying "NO" if it does not. If the requested unit has an index, FILECOMS calls the routine LOOKUP to check for a name match. If there is no name match, FILECOMS displays "NO". If there is a name match, FILECOMS further checks to see if the name refers to binary only, and if so displays a "NO". If the name match is legal, FILECOMS reads Print Source and TTY into locations 5000-5777, retrieves the first TBLK of the source

from the index and puts it into AC. FILECOMS then calls Print Source by executing a JMP 1000.

4.0 THE ASSEMBLER AND FILECOMS

Whenever an assembly is requested of a source in the file area, FILECOMS must be called to set up arguments for the Assembler. The Editor executes this call by reading TBLKs 350-353 into locations 4000-5777 and doing a JMP FCSA + 4. The index routines are next called by FILECOMS and executed in exactly the same manner as if a Print Source had been requested. FILECOMS places the unit number at location UNITNO (4777), places the QL or LI word at location UNITNO-3 (4774), and puts the starting TBLK of the requested source in the AC. FILECOMS then recalls the Editor to set up further arguments for the Assembler, read it in and start it up.

5.0 SAVE BINARY AND FILECOMS

When a Save Binary is requested, the Editor exits, loads four blocks of FILECOMS into 4000-5777, and jumps to FCSA, File Commands starting address. Save Binary reads the binary header block (57 in the binary working area) into 5400-5777.

E6, the MC parameter table, is examined to determine the starting mode and address. If none were specified (+SB FILENAME,U), E6 contains zero. A LINC-mode halt is stored in the header, and Save Binary jumps to SB020 to find space for the file. If 8-mode was specified, E6 contains 4400 plus the PDP-8 field in which the program is to start. E6+1 contains the 12-bit address within that field. The following instruction sequence is stored in the header:

PDP	
CIF X	/WHERE X IS 0 OR 10, DEPENDING ON /THE FIELD SPECIFIED
JMP I 377	/START THE PROGRAM
START	/STARTING ADDRESS FROM E6+1

If LINCmode was specified, E6 contains 0400 plus the starting PDP-8 field, and E6+1 contains the 12-bit address within the field. The following instruction sequence is built in the header:

LIF X	/X IS 0-7, DEPENDING ON THE SEGMENT /SPECIFIED.
JMP START	/START IS THE 10 LOW-ORDER BITS /FROM E6+1.

The default starting addresses are field 0, location 200 for PDP-8 mode and segment 2, location 20 for LINCmode. These are set up by the Editor before

calling FILECOMS, if a starting address was not explicitly requested.

After building these instructions in the header, the starting address is checked for validity. It must be less than 20000 (8K), and in a block which contains assembled code.

Save Binary next obtains the number of binary blocks (location 337 within the header) from the header. One is added to this number, to allow for the header, and the result complemented and stored at F1. This is used as the length of the desired file during the search for file space. The two's complement of the number of binary blocks is calculated and stored at MBCNT, as a loop control during copying. Save Binary then follows the same sequence as Save Program to read the index, find space for the file, and write the updated index. Next, the I/O control blocks are initialized, an auto-index register is set up for the block map in the header, and copying begins.

Each word of the block map is tested for zero. If it is zero, the corresponding block in the working area is not used, so the input block number is incremented and the loop restarted. If the word is not zero, the corresponding block is read, and MBCNT is tested to see whether it is the last binary block. If it is the last, Save Binary computes the number of blocks currently in the buffer, and goes to SB13# to write them. If not the last block, Save Binary increments the input buffer address, and checks to see whether the buffers are full. If not, the input block number is incremented and the loop re-entered. If the buffers are full, or the last block has been read, HDRSW is tested to determine whether the header block has been written. If not, it is written at this time, and the switch is set. Then the buffers are written, and MBCNT is tested to check for completion. If there is more to go, the I/O control blocks are reinitialized and the loop is re-entered. Otherwise, the Editor is called by jumping to 7777 in field 1.

6.0 ADD BINARY AND FILECOMS

When Add Binary is requested, the Editor exits and loads blocks 35#-353 into locations 4000-5777. It then jumps to FCSA+1. Add Binary reads the index of the requested file, and finds the binary requested, displaying "NO" if unsuccessful.

The header and up to 15 blocks of the binary file are read, as well as the header of the binary working area.

The header from the file is scanned by subroutine SCANHD to determine the memory address for which the first block was assembled. If a relocation (other than zero) was specified, it is stored as the new memory address for the program.

The binary data from the file is scanned, and non-zero words moved to the buffer for the binary working area.

When the end of an input block is reached, a running block count (MBLKS) is incremented and checked for end of file. If the end has been reached, the final binary working area block is written, the binary header is written, and control returns to the Editor.

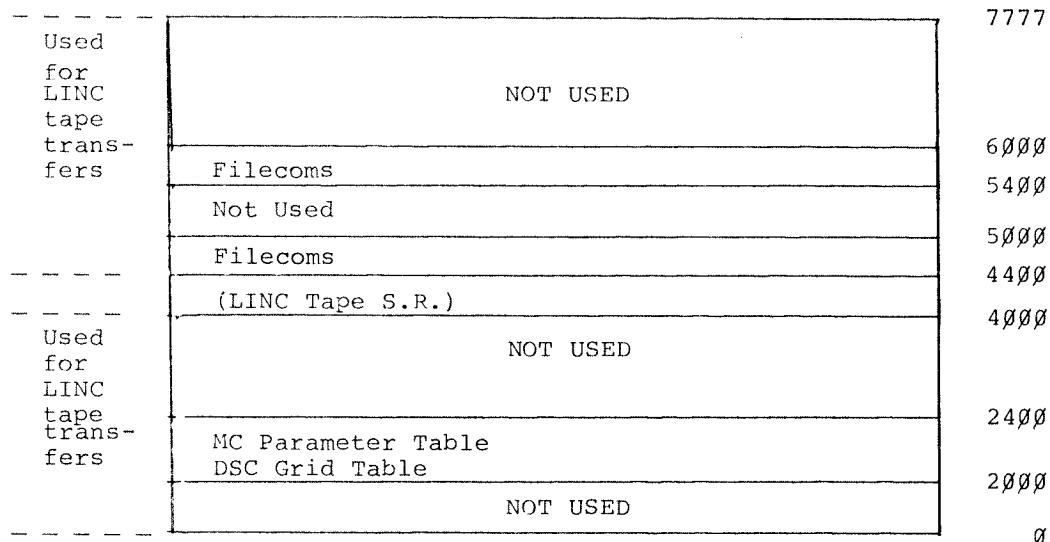
If there remains more binary to add, the header is scanned so that the relocation can be adjusted for skipped blocks and, if the end of the buffer has been reached, the buffer is refilled with up to 15 blocks from the file.

When a non-zero word is found which goes into another block of the binary working area (other than the block currently in core), the old block is written, the header updated, and, if the header indicates that the new block is used, the new block is read in. If there is nothing in the new block, the buffer is cleared.

7.0 FLOW DIAGRAMS (Attached)

8.0 PROGRAM LISTING (Attached)

9.0 MEMORY MAP



FILECOMS SYMBOL TABLE

ADVIND	-	The first core location of the subroutine that advances the index pointer (BETA 2) by 10 words at a time and checks for end of index.
AUTO	-	8-mode auto-index register (10) used by SB.
A6	-	The first core location of the Editor's DSC grid table.
BFAD	-	The location containing buffer address, shifted right 8.
BFLN	-	The location containing the length of buffer, in blocks.
BUFFER	-	The buffer address, shifted right 8.
BUFLEN	-	The length of the buffer, in blocks.
BUMP	-	The location containing the increment for index pointer to address desired (source or binary) pointers.
BWA	-	The pseudo-unit for binary working area.
CHKIND	-	The first core location of the subroutine that reads in and checks for the validity of an index.
COMPBN	-	The first core location of the subroutine that compares TBLKs in the index with the prospective best TBLK and sets the best TBLK during the gap search for Save Program.
CRL	-	The symbolic code for carriage return; it is equal to 4300 and is used in conjunction with the SHD I instruction.
DATSEG	-	The data segment used throughout FILECOMS. It contains A6, E6, and Index.
DIALU	-	The pseudo-unit that contains the resident DIAL system (100).
DRSTRT	-	The address of DIAL bootstrap routine.
EDRTN	-	The start of routine to call DIAL bootstrap.
ENTER	-	The first core location of the subroutine that searches for a blank slot in the index during a Save Program.
E6	-	The first core location of the MC parameter table.
FCSA	-	The first core location of the FILECOMS starting address table.

FILE - The first TBLK in the upper file area above the index that is available for file storage.

FINDSP - The first core location of the subroutine that sets up arguments for upper and lower file during a gap search.

FREE - The first TBLK+1 in the lower file area below the index that is available for file storage (also the first TBLK of the free area on the DIAL tape).

F1 - The core location that holds the one's complement of the length of a source during a Save Program, or one's complement of the length of the binary (including header) during a Save Binary.

GAPSR - The first core location of the main gap search subroutine which is called during a Save Program and Save Binary.

GETBN - The first core location in the subroutine that sets up the starting TBLK and source length during a gap search.

GETPS - The starting point of code which reads in and calls Print Source.

G1 - The first core location of the subroutine that determines the nearest of the two best TBLKs in the upper and lower file area; it is called during a Save Program.

HDRBLK - The TBLK of binary header, relative to binary working area.

HDRI0 - The control block for reading and writing header.

HDRSW - The switch is 7777 if the header has not been written; Ø after writing.

HEADER - The core location in the current segment which contains the header.

HI8FLD - The field containing I/O routines and bootstrap.

H1 - The first core location of the subroutine that is called during a Save Program to determine if the best TBLKs will accommodate the length of the requested source.

INDEX - The core location of the index (3000).

INFIL - The control block used in reading files to be copied.

KBDOPR - The core location in the Editor relative to which FILECOMS effects its return to the Editor after an assembly by file name has been requested.

KBDSEG - The segment containing KBDOPR.

K2 - A tag in the subroutine REPLAC; tests for response from user.

LIORAS - The first core location of the main subroutine that is called by FILECOMS during an assembly by file name.

LOOKUP - The first core location of the subroutine that searches the index for a name match.

LO8FLD - The field in which FILECOMS resides.

MAKIND - The first core location of the subroutine that makes an index.

MAP - The core location in the current segment containing the block map from header.

MBCNT - The location containing minus block count: two's complement of number of blocks to read from working area.

MBFAD - The location containing the minus buffer address, shifted right 8.

MBFEND - The location containing minus (buffer end + 1), shifted right 8.

OUTFIL - The control block for output file during copies.

PDPMOD - The location of the routine for setting up PDP-8 mode start in Save Binary.

PRNPRG - The first core location of the main subroutine that is called by FILECOMS during a Print Source.

PSBLK - The first TBLK of Print Source (relative to DIAL).

PSENT - The entry point of Print Source main code.

PSIN - The control block for reading Print Source.

PSWA - The routine which sets up for printing the source working area.

P1 - Constant $\emptyset\emptyset\emptyset 1$.

P2 - Constant $\emptyset\emptyset\emptyset 2$.

READ - The page \emptyset location pointing to Read routine.

REPLAC - The first core location of the subroutine that displays the phrase "REPLACE?" on the scope.

RPLSTR - The first core location of the replace table.

RSTRX - The first core location of the subroutine that writes out the index.

SAVBIN - The first location of Save Binary routine.

SAVPRG - The first location of Save Program routine.

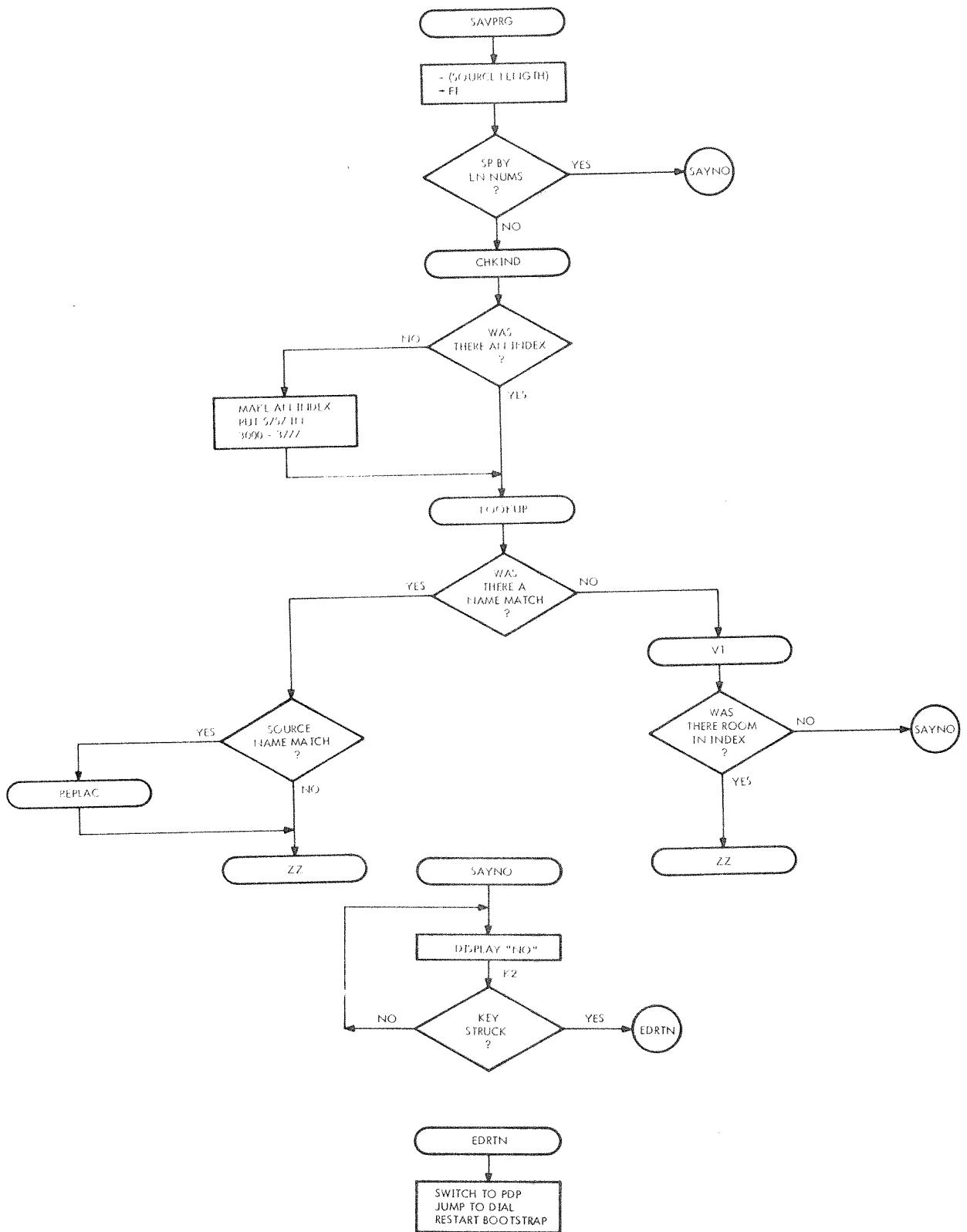
SAYNO - The first core location of the subroutine that displays the word "NO" on the scope.

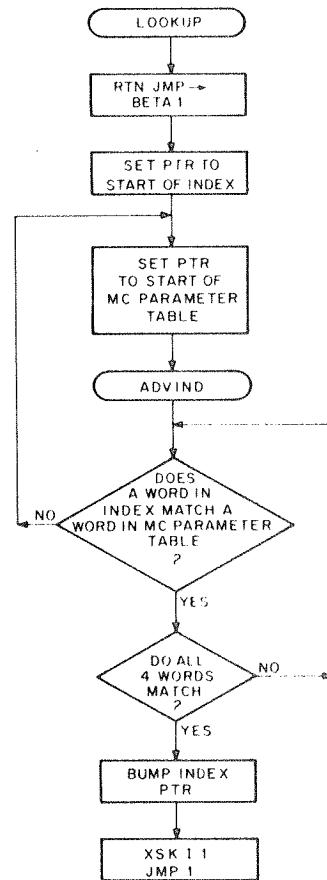
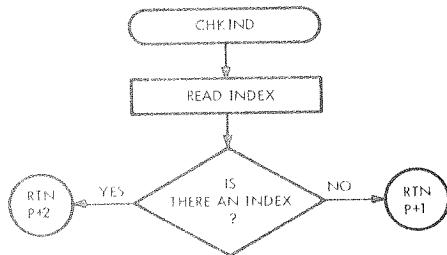
SB \emptyset 1 \emptyset	-
SB \emptyset 2 \emptyset	-
SB \emptyset 3 \emptyset	-
SB \emptyset 4 \emptyset	-
SBL \emptyset \emptyset	-
	Tags in Save Binary
SB11 \emptyset	-
SB12 \emptyset	-
SB13 \emptyset	-
SB14 \emptyset	-
SB16 \emptyset	-

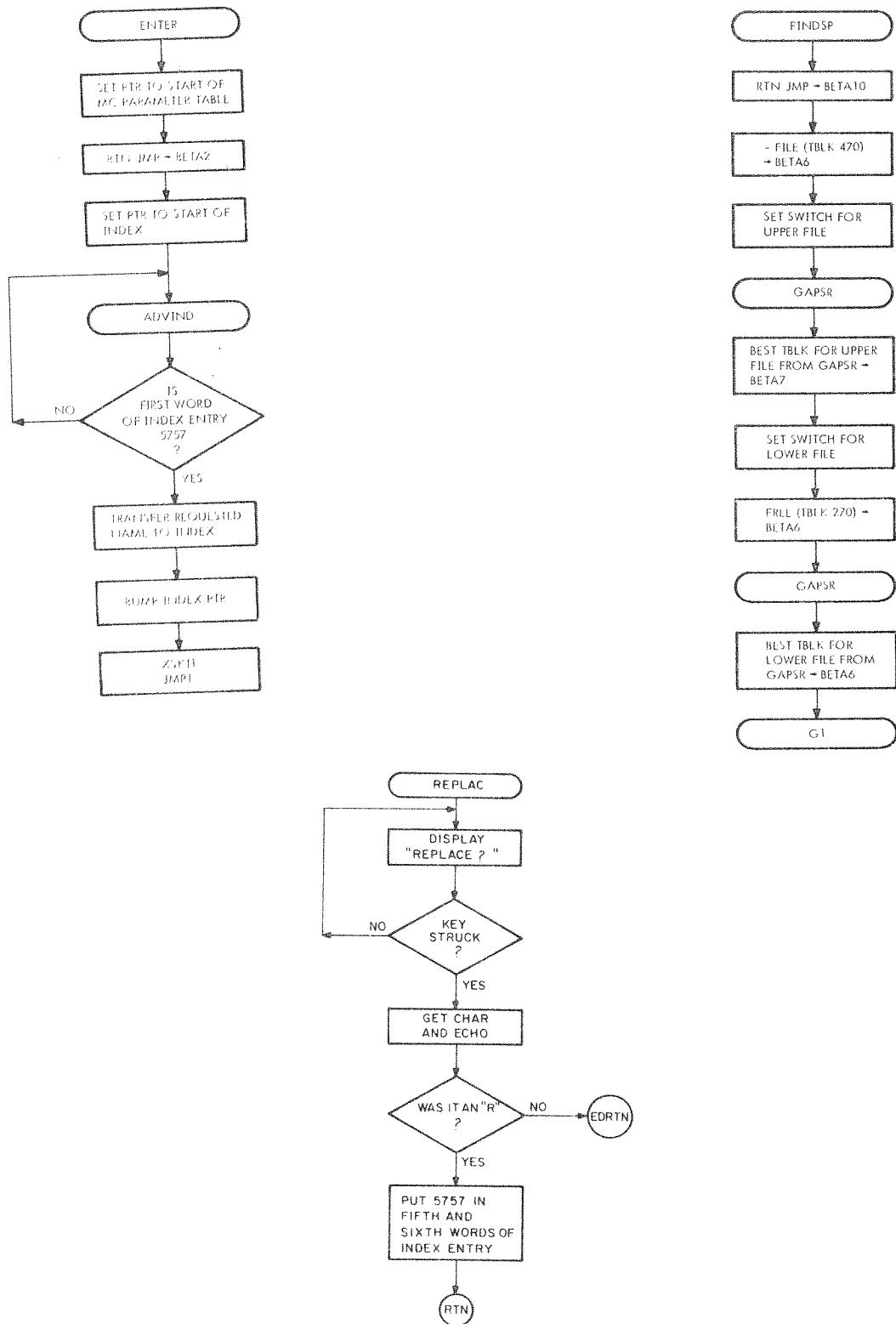
SETFLD - The location of LIF instruction inserted into the header.

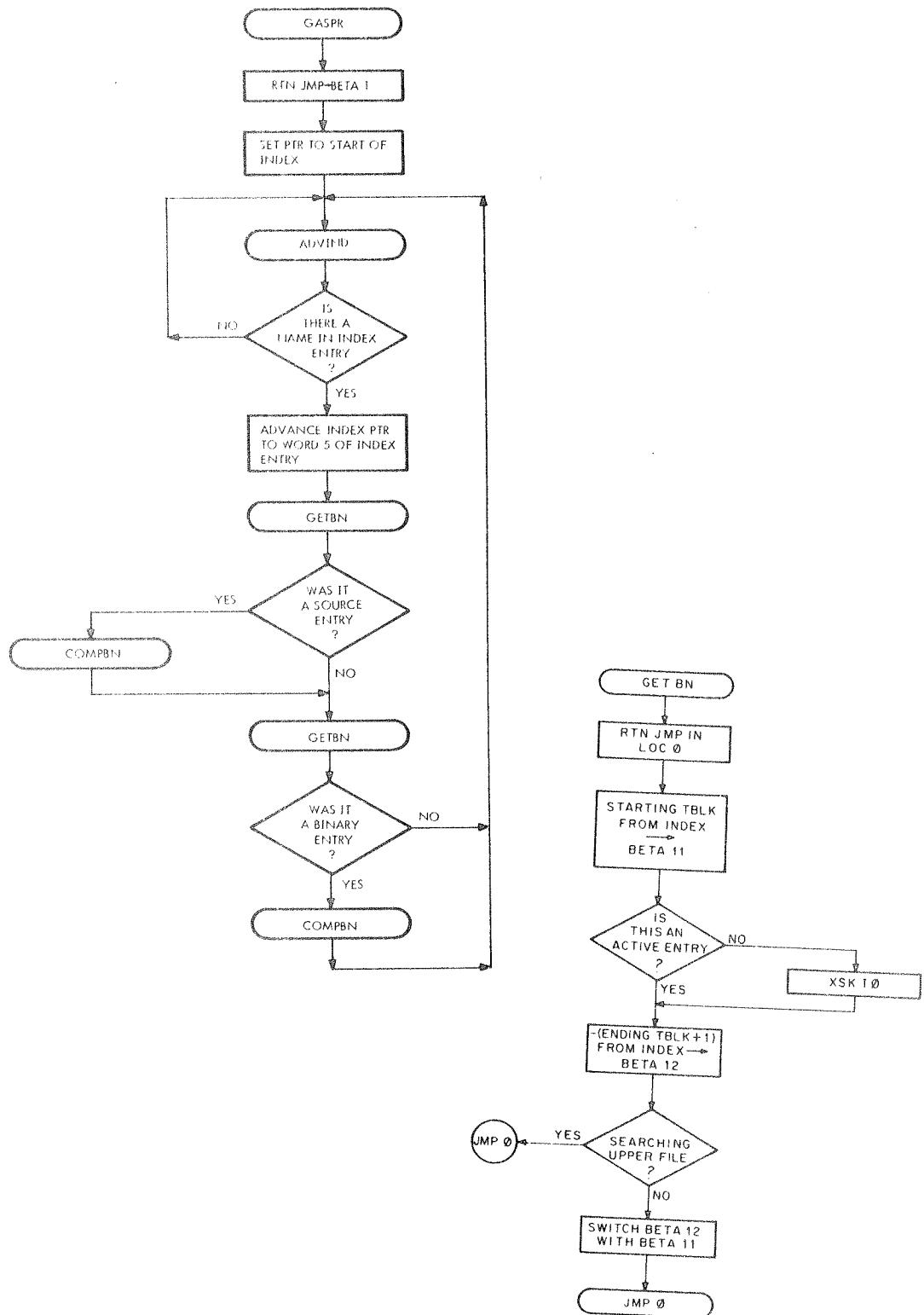
SP \emptyset 2 \emptyset	-
SP \emptyset 3 \emptyset	-
SP1 \emptyset \emptyset	-
	Tags in Save Program
SP11 \emptyset	-
SP1 \emptyset	-

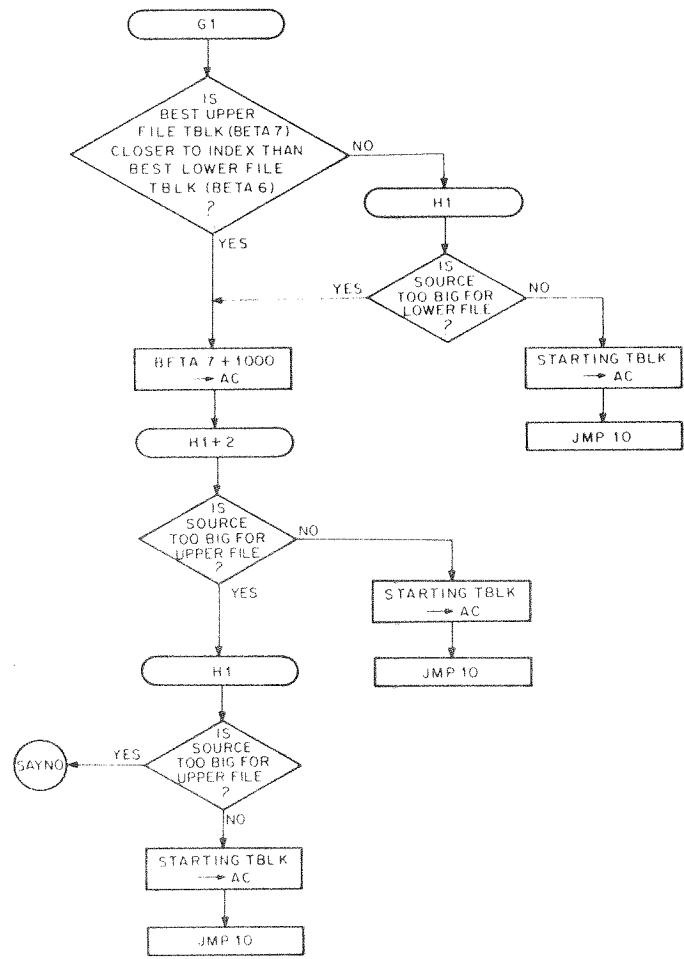
SRORBN	-	The first core location of the subroutine that determines whether a name match is source or binary.
SWA	-	The pseudo-unit that holds the source working area.
SWITCH	-	A core location that is used by the gap search subroutines in conjunction with the upper and lower file area.
UNITNO	-	The core location that holds the unit number for assembly by name.
USENO	-	The core location containing the count of binary blocks used.
WA	-	The first TBLK of the Working Area.
WRITE	-	The page \emptyset location pointing to Write routine.
XBLK	-	The TBLK containing the index.
XIO	-	The control block for reading and writing index.
ZERO	-	Constant zero.

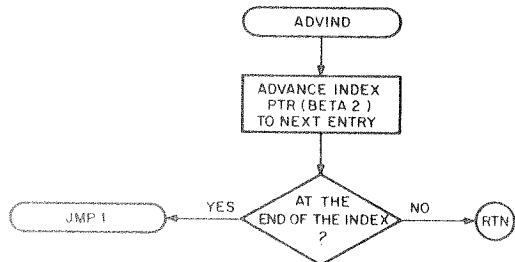
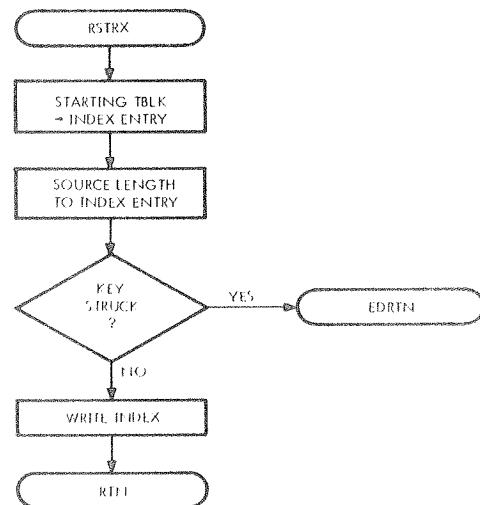
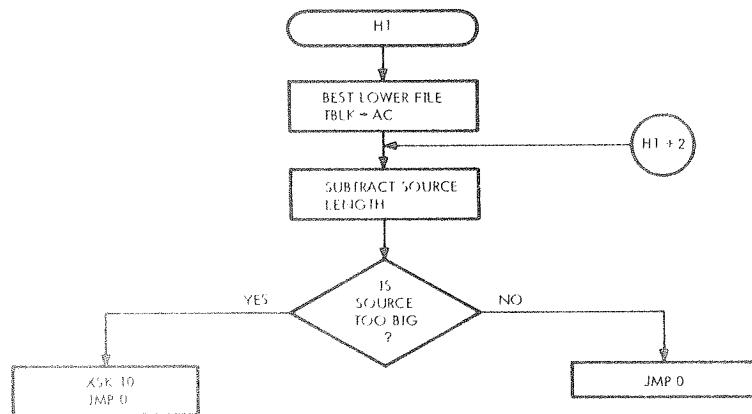


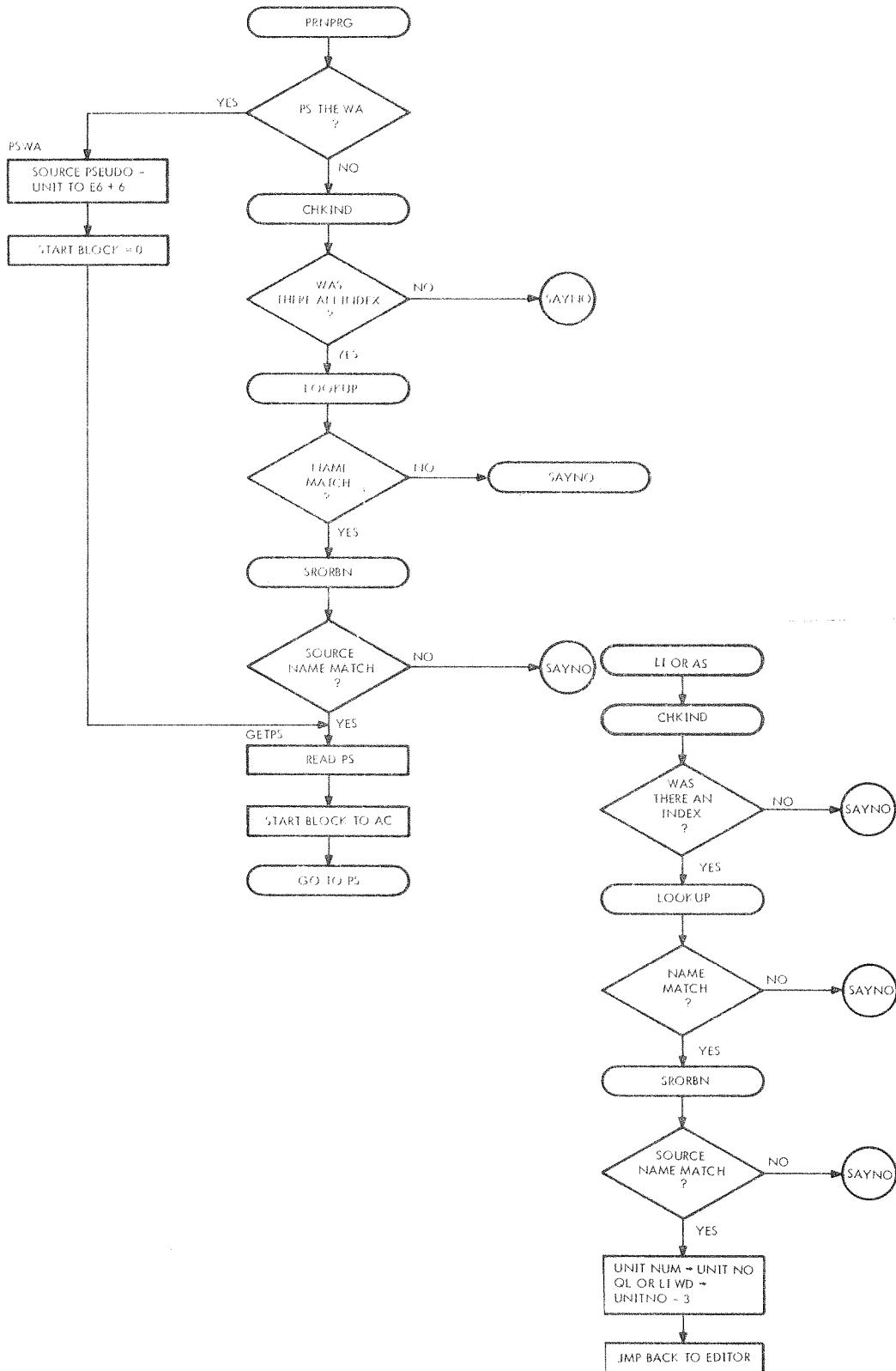


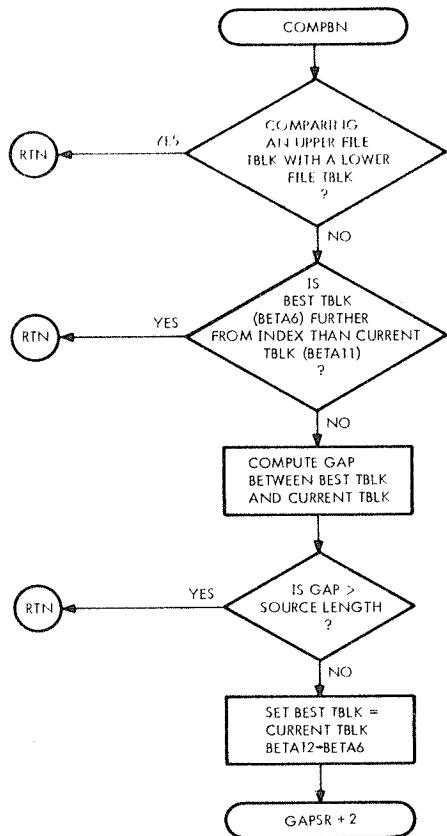


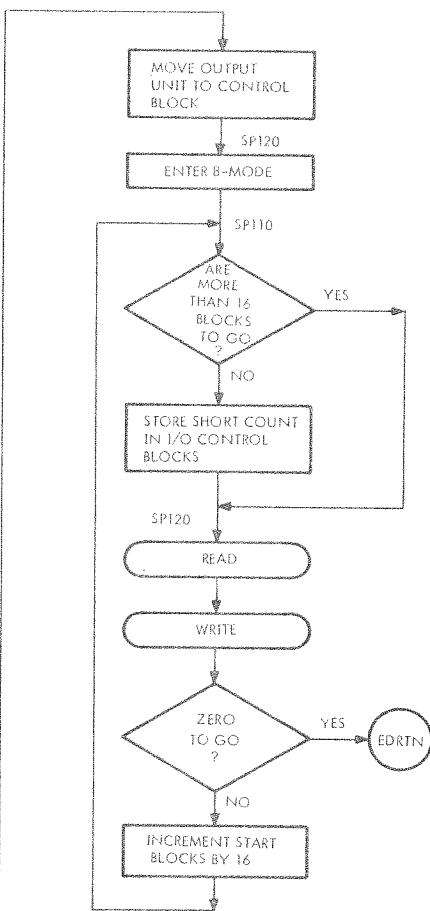
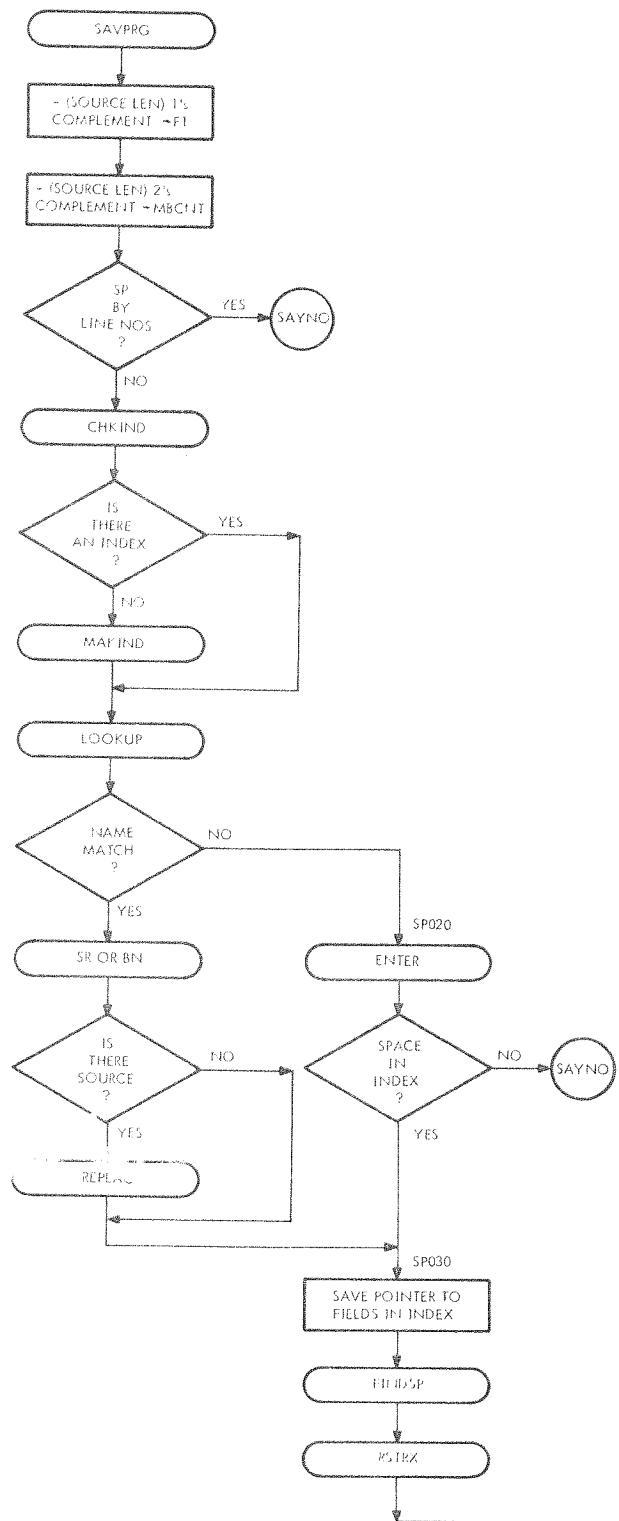


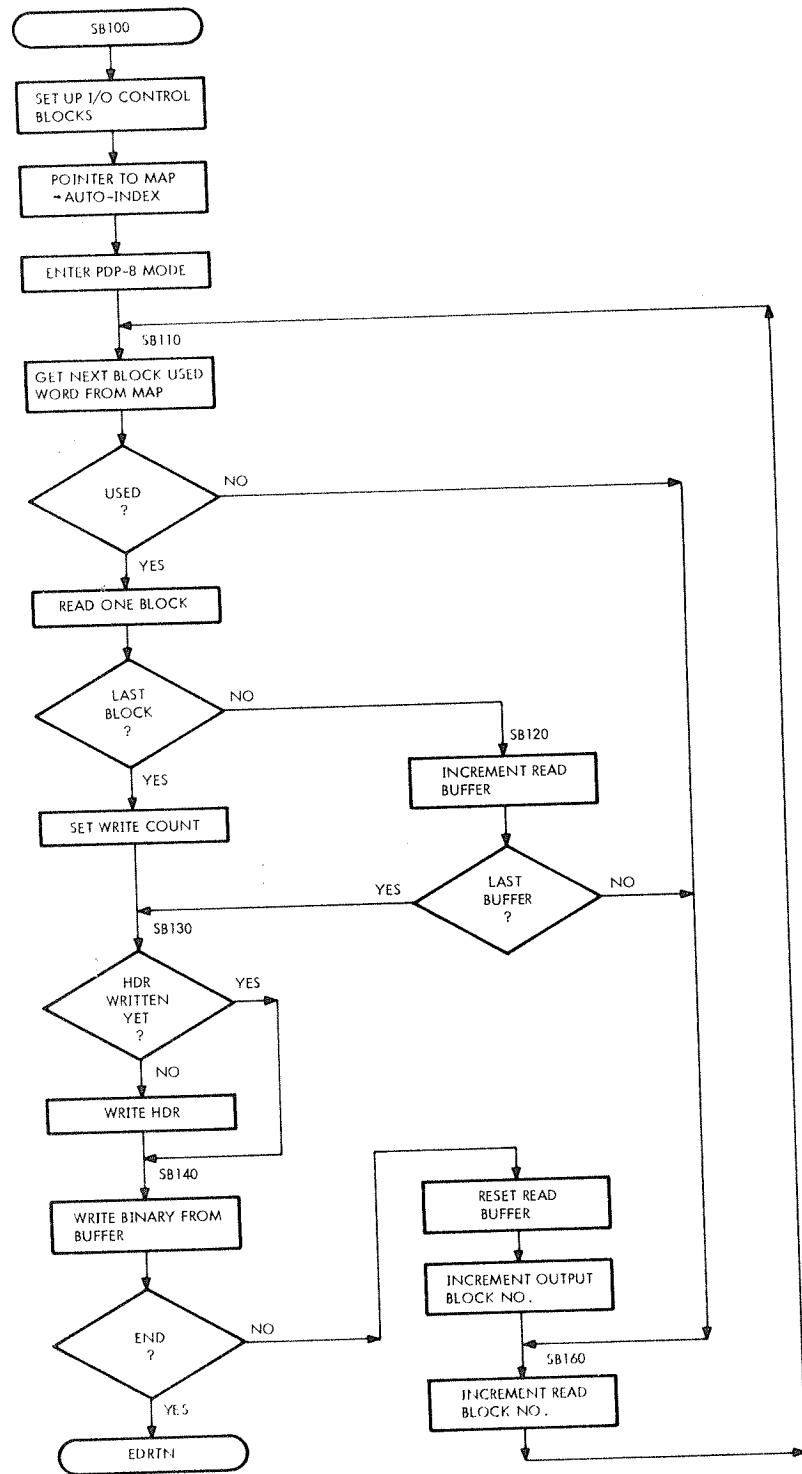


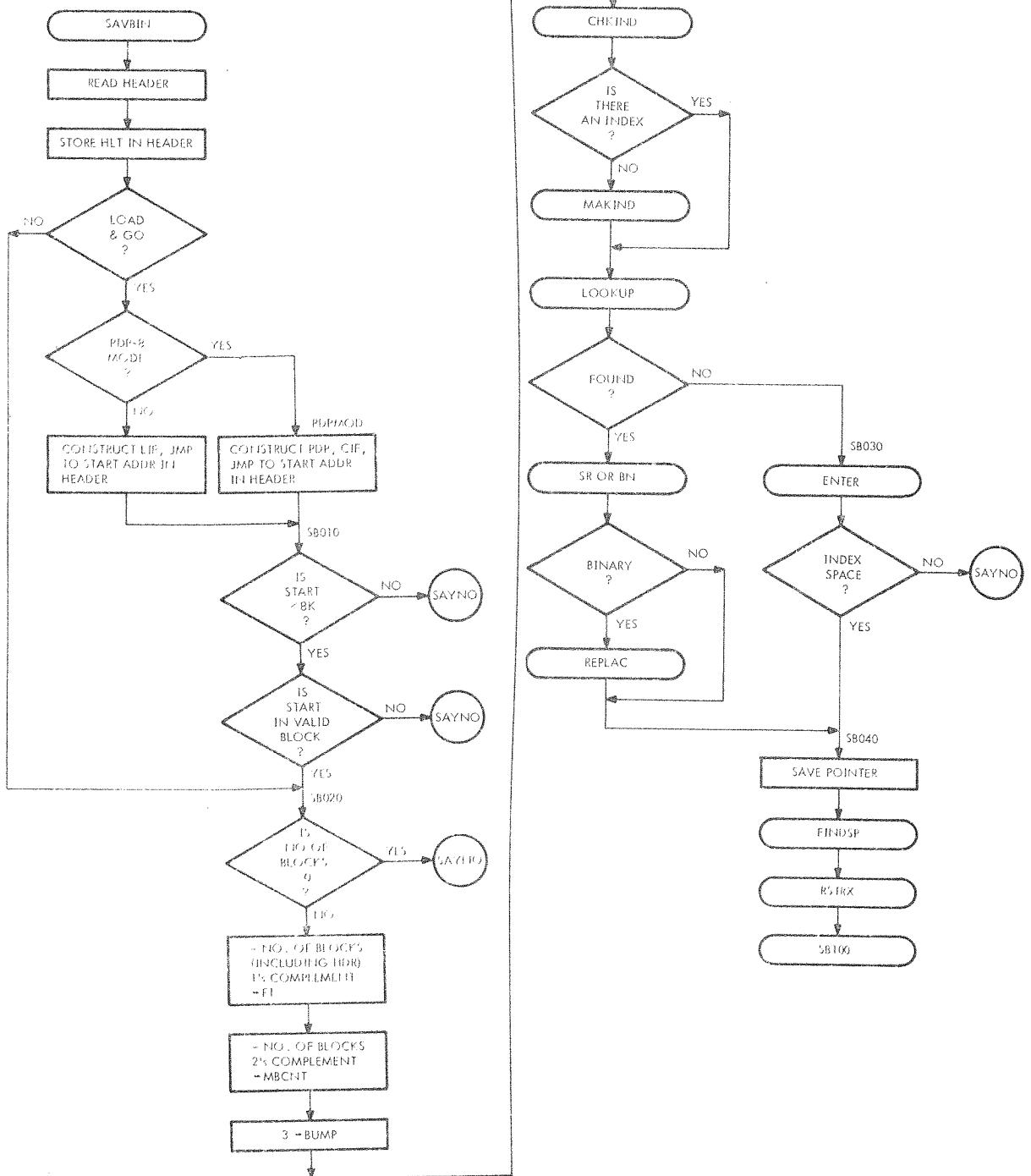












LAPE-FILE COMMANDS . . . SB,SP,AB,LIL,GLAS,PS
MAR 31, 1976

THIS ROUTINE IS A MODIFICATION OF FUNCTIONALLY EQUIVALENT CODE FROM LAPE. COMMENTS HAVE BEEN ADDED, TAGS CHANGED FROM LETTER-DIGIT TO MNEMONIC NAMES, ABSOLUTE MEMORY ADDRESS REFERENCES REMOVED WHERE POSSIBLE, I/O DISCIPLINE AND BUFFERING IMPOSED, AND THE WHOLE CONDENSED FROM 7 BLOCKS TO 4, WE ARE DEEPLY INDEBTED.

RESIDES IN DIAL BLOCKS 50-53
OCCUPIES MEMORY LOCATIONS 4000-5777

46 MBLKS (OCTAL) IN UPPER FIELD ARE USED FOR BUFFERS: 1 2222 - 1 6777 LOCATIONS 17000 TO 17777 MUST REMAIN INTACT.
THESE LOCATIONS CONTAIN DIAL I/O ROUTINES AND BOOTSTRAP,

THE FOLLOWING AREAS IN THE LOWER RDP-6 FIELD ARE USED:

AUTO-INDEX REGISTER 10 IS USED BY SB AND AB,
PAGE-ZERO LOCATIONS 21 AND 22 ARE ASSUMED TO CONTAIN POINTERS
TO THE DIAL I/O ROUTINES IN THE UPPER FIELD,
20000 - 22000 ARE ASSUMED TO CONTAIN THE DIAL CHARACTER DISPLAY TABLE,
2371 - 2377 CONTAIN THE MC PARAMETER TABLE, E6,
36000 - 3777 ARE USED TO READ AND WRITE THE DIAL INDEX,
AND FOR HEADERS DURING AB,
4400 - 5777 ARE USED FOR THE BINARY HEADER BLOCK DURING SB OPERATIONS,
50000 - 5777 ARE USED BY THE MAIN CODE OF PS (PRINT SOURCE),
50000 - 5777 ARE USED FOR THE BINARY HEADER BLOCK DURING SB OPERATIONS.

ODE IS NOT REUSABLE

MP SAVBIN /EDITOR
MP ADDBIN /CALLS
MP ' /ALL
MP SAVPRG /FCCMS
MP LIORAS /HERE
MP PRNPRG /PROJECT

```

    /
    /
    / SAVE BINARY ENTERS HERE
    /
    /
    0052
    0053
    0054
    0055
    0056
    0057 0026 0002 SAVBIN, PDP PMODE
    0060 4027 6201 CDF L08FLD
    0061 4030 6212 CIF H18FLD
    0062 4031 4421 JMS I READ
    0063 4032 4323 HDRIO
    0064 4033 6141 LINC
    0065
    0066
    0067 0034 0641 LMODE
    0070 0035 0011 LDF DATSEG
    0071 0036 5400 CLR
    0072 0037 1000 STC HEADER
    0073 0040 2371 LDA
    0074 / SIGN E6
    0075 0041 0470 EQ 0 IF LINC; EQ 0 IF NOT L&G
    0076 0042 6134 AZE I
    0077 0043 0451 JMP SB020 /NOT L&G
    0078
    0100 0044 6070 APO
    0101
    0102
    0103
    0104 0045 0242 ROL 2
    0105 0046 1560 BCL 1
    0106 0047 7773 7773
    0107 0050 1340 STH
    0110 0051 4060 SETFLD:4000
    0111 0052 1000 LOA
    0112 0053 2372 E6*1
    0113
    0114 0054 0242 ROL 2
    0115 0055 1560 BCL 1
    0116 0056 7774 7774
    0117 0057 1620 BSE 1
    0118 0060 0600 SETFLD, LIF
    0120 0061 5400 STC HEADER
    0121 0062 1000 LDA
    0122 0063 2372 E6*1
    0123 0064 1620 BSE 1
    0124 0065 6000 JMP
    0125 0066 5401 STC HEADER+1
    0126 0067 6111 JMP SB010 /ALL SET
    0127
    0130

```

```

        / COME HERE IF PROGRAM IS TO SAT IN 8MODE
        / PDPMOD, LDA I           /SET UP FOR 8MOD START
        PDP
        STC HEADER
        LDA I
        /JMP I 377
        STC HEADER+2
        LDA /CHK FIELD
        E6
        ROL 3
        BCL 1
        /8K FOR THIS VERSION
        7767
        BSE I
        /MAKE A CIF
        6202
        STC HEADER+1
        LDA
        E6+1
        STC HEADER+3
        /STARTING ADDR

        / CHECK STARTING ADDR FOR VALIDITY
        / (LT 8K AND IN A VALID BLOCK)
        SB010, LDA /HIGH ORDER ADDRESS DIGIT
        E6
        ROR I 1
        BCL I
        /MOVE LOWER/UPPER SEGMENT BIT TO LINK
        /CLEAR NON-ADDRESS BITS
        7774
        AZE
        /SKIP IF START ADD LT 8KK
        JMP SAYNO
        /OTHERWISE REFUSE
        LDA
        /GET LOW ORDER ADDRESS
        E6+1
        BCL I
        /TRUNCATE TO START OF MBLK
        0377
        ROL I 5
        BSE I
        /MOVE RELATIVE BLOCK NO (0-37) TO LOW ORDER
        /INDEX THE ADD INSTRUCTION
        ADD MAP
        STC +1
        /STORE AND EXECUTE IT
        0
        /FILLED BY (ADD MAP+BLK) INSTRUCTION
        SAE I
        /TEST FOR VALID BLOCK
        7777
        JMP SAYNO
        EJECT
        /INVALID -- REFUSE
        -

```

```

/
/
/
/ HEADER IS NOW PROPERLY SET UP
/ LOOK FOR SPACE FOR THE FILE
0205
0206
0207
0210
0211
0212
0213
0214
0215
0216
0217
0218
0219
0220
0221
0222
0223
0224
0225
0226
0227
0228
0229
0230
0231
0232
0233
0234
0235
0236
0237
0238
0239
0240
0241
0242
0243
0244
0245
0246
0247
0248
0249
0250
0251
0252
0253
0254
0255
0256
0257
0258
0259
0260
0261
0262
0263
0264
3737 SB020, ADD USENO /NO OF BLKS USED (FROM HEADER)
0134 0470 JMP SAYNO /BOMB IF ZERO
0135 7071 ADD P1 /PLUS ONE FOR HEADER
0136 2347 COM /FINDSP WANTS IT NEGATIVE
0137 0140 STA /HOLD FOR SPACE SCAN
0138 0141 F1 ADD P2 /TWO'S COMP OF NO OF BLOCKS (WITHOUT HEADER)
0139 0142 ADD MBCNT /HOLD FOR COPY
0140 0143 2161 STC P3 /REPLACE BUMP CONSTANT IN LOOKUP
0141 0144 4327 STC BUMP /READ IN THE INDEX BLK
0142 0145 3145 JMP CHKIND /RTN FOR NO INDEX-MAKE ONE
0143 0146 4536 JMP MAKIND /SEARCH INDEX FOR NAME IN E6+2
0144 0147 6505 JMP LOOKUP /NOT FOUND
0145 0148 6505 JMP SB030 /CHK IF SYMBOLIC ONLY
0146 0149 6517 JMP SRORBN /NO-DISPLAY REPLACE FOR VERIFY
0147 0150 6517 JMP REPLAC /BIN ENTRY EMPTY, OR REPLACE REQUESTED
0148 0151 6517 JMP SB040 /PUT NAME IN INDEX
0149 0152 6156 JMP ENTER /NO SPACE IN THE INDEX
0150 0153 6436 JMP SAYNO /SAVE XR PTR
0151 0154 6543 SET 4
0152 0155 6160 JMP SB030, /FIND BEST SPACE, RETURN TBLK IN AC
0153 0156 6622 JMP ENTER /UPDATE & WRT XR
0154 0157 7071 JMP SAYNO /BINARY WORK UNIT...
0155 0160 0044 SB040, 2 JMP FINDSP /FIND OUTFILE+281777 /STARTING BLOCK
0156 0161 0002 P2, 2 JMP RSTRX /STC INFIL /., TO CONTROL INPUT
0157 0162 6647 LDA I ADD OUTFILE+281777 /., IS HEADER TBLK...
0158 0163 6775 BWA STA HDR10+2 /., FOR WRITING HEADER...
0159 0164 1020 SB100, 0111 STC INFIL ADD P1 /., PLUS 1
0160 0165 0111 LDA I ADD OUTFILE+281777 /., IS START OF ACTUAL BINARY
0161 0166 4313 BWA STA HDR10+2 /CONSTANT 1...
0162 0167 2321 STC INFIL+3 /FOR SINGLE-BLOCK INPUT
0163 0168 1040 ADD X10 /OUTPUT UNIT
0164 0169 1040 STA HDR10 /., FOR WRITING HEADER...
0165 0170 4325 STC OUTFILE /., AND BINARY
0166 0171 2347 LDA I /GET ADDRESS OF BLOCK MAP
0167 0172 4321 ADD P1 /MAP=1:4000
0168 0173 2547 STC INFIL+2 /MAP IN LINE FIELD 2
0169 0174 2547 ADD PDP /ENTER PMODE FOR COPY
0170 0175 4316 PMODE
0171 0176 3017 DCA AUTO /SETUP AUTO-INDEX REG
0172 0177 1040 EJECT
0173 0178 4523
0174 0179 0201 4317
0175 0180 1020 5737
0176 0181 0202 0002
0177 0182 0203 0204
0178 0183 0204 0002
0179 0184 4202 3010
0180 0185 0205 0264

```

3265

3267

3270

3271

3272

3273

3274

3275

3276

3277

3278

3279

3280

3281

3282

3283

3284

3285

3286

3287

3288

3289

3290

3291

3292

3293

3294

3295

3296

3297

3298

3299

3300

3301

3302

3303

3304

3305

3306

3307

3308

3309

3310

3311

3312

3313

3314

3315

3316

3317

3318

3319

3320

3321

3322

3323

3324

3325

3326

3327

3328

3329

3330

3331

3332

3333

3334

3335

3336

3337

3338

3339

3340

3341

3342

3343

3344

3345

3346

3347

3348

3349

3350

3351

3352

3353

3354

3355

3356

3357

3358

3359

3360

3361

3362

3363

3364

3365

3366

3367

3368

3369

3370

3371

3372

3373

3374

3375

3376

3377

3378

3379

3380

3381

3382

3383

3384

3385

3386

3387

3388

3389

3390

3391

3392

3393

3394

3395

3396

3397

3398

3399

3300

3301

3302

3303

3304

3305

3306

3307

3308

3309

3310

3311

3312

3313

3314

3315

3316

3317

3318

3319

3320

3321

3322

3323

3324

3325

3326

3327

3328

3329

3330

3331

3332

3333

3334

3335

3336

3337

3338

3339

3340

3341

3342

3343

3344

3345

3346

3347

3348

3349

3350

3351

3352

3353

3354

3355

3356

3357

3358

3359

3360

3361

3362

3363

3364

3365

3366

3367

3368

3369

3370

3371

3372

3373

3374

3375

3376

3377

3378

3379

3380

3381

3382

3383

3384

3385

3386

3387

3388

3389

3390

3391

3392

3393

3394

3395

3396

3397

3398

3399

3300

3301

3302

3303

3304

3305

3306

3307

3308

3309

3310

3311

3312

3313

3314

3315

3316

3317

3318

3319

3320

3321

3322

3323

3324

3325

3326

3327

3328

3329

3330

3331

3332

3333

3334

3335

3336

3337

3338

3339

3340

3341

3342

3343

3344

3345

3346

3347

3348

3349

3350

3351

3352

3353

3354

3355

3356

3357

3358

3359

3360

3361

3362

RECORDED AND INDEXED
MAY 1964

0530
0531
0532
0533
0534
0535
0536
0537
0538
0539
0540
0541
0542
0543
0544
0545
0546
0547
0548
0549
0550
0551
0552
0553
0554
0555
0556
0557
0560
0561
0562
0563
0564
0565
0566
0567
0570
0571
0572
0573
0574
0575
0576
0577
0600
0601
0602
0603
0604
0605
0606
0607
0610
0611

PRINT SOURCE

```
 / / / / PRNPRG, LDF DATSEG /CHK MC PARAM
 / / / / LDH /TABLE /FILE ENTRY
 0641 1300 SHD 1 /OR WA?
 0376 0377 E6*2 /PRINT THE WA
 0400 2373 0401 1420 /NAMED FILE -- READ AND CHECK THE INDEX
 0402 7000 0402 7700 /NO INDEX
 0403 6413 0403 6413 /INDEX OK -- FIND THE NAME
 0404 6464 0404 6464 /NO NAME MATCH
 0405 7071 0405 7071 /NO THERE SOURCE?
 0406 6517 0406 6517 /YES -- LOAD PS
 0407 7071 0407 7071 /BINARY ONLY, CANT DO IT
 0410 6436 0410 6421
 0411 6421 0411 6421
 0412 7071 0412 7071
 0552 // COME HERE IF SWA TO BE PRINTED
 0553 // SOURCE WA PSEUDO-UNIT
 0413 1020 0413 LDA 1
 0414 0110 0414 SWA
 0415 1040 0415 STA //,,TO MC PARAM LIST
 0416 2377 0416 E6*6 //,,AS I/O UNIT
 0417 0062 0417 SET 1 2 //STARTING BLOCK IS ZERO
 0420 0435
 0564 // READ PS MAIN CODE -- BLOCKS 63, 4 ON DIAL
 0565 0421 0002 0421 GETPS, POP //PHODE FOR READ
 0566 // PMODE
 0567 4422 6212 0567 CIF H18FLD
 0570 4423 4421 0570 JMS 1 READ /READ PS AND TELETYPE INTO 5002
 0571 4424 4431 0571 PSIN
 0572 4425 6141 0572 LINC
 0573 // LMODE
 0574 0426 0641 0574 LDF DATSEG /SET DATA FIELD FOR PS
 0575 0427 1002 0575 LDA 2 /PICK UP STARTING BLOCK NO OF DESIRED SOURCE
 0576 0430 7000 0576 JMP PSENT /GO TO PS MAIN PROCESSING
 0600 // PARAMS FOR READING PS AND TTY
 0601 // DIALU /DIAL RESIDENCE UNIT
 0602 PSIN, 0602 0100 DIALU
 0603 0431 0012 0603 12 /MEM ADDR = 5000
 0604 0432 0012 0604 PSBLK
 0605 0433 0063 0605 2 /DIAL BLOCKS 63,64 CONTAIN PS, TTY
 0606 0434 0002 0606 /GET EM BOTH
 0607 0435 0000 0607 ZERO, 0607 EJECT
 0610 //CONSTANT ZERO TO USE AS STARTING BLOCK IF SWA PRINTED
 0611 ~
```

SUBROUTINES

```

    /
    /
    /      MAKE AN EMPTY INDEX
0707
0710
0711
0712
0713      0041  MAKIND, SET 1
0714      0000  0      0
0715      0062  SET 1 2
0716      0510  2777  /POINTER TO INDEX
0717      0511  1020  INDEX-1
0718      0512  5757  LDA 1
0719      0513  1062  /FILL WITH
0720      0514  0202  5757 (//)
0721      0515  6513  STA 1 2
0722      0516  6001  XSK 2
0723      0517  6515  /DONE ?
0724      0518  6001  /NO
0725      0519  6516  JMP 1-2
0726      0520  6001  JMP 1
0727      0521  0000  /
0728      0522  0062  /SEARCH FOR A NAME IN INDEX
0729      0523  3000  NAME TO FIND IS AT E6*2
0730      0524  0063  /
0731      0525  7773  /START OF INDEX
0732      0526  0064  INDEX
0733      0527  6444  SET 1 3
0734      0528  6444  /LENGTH OF NAME
0735      0529  1022  *4
0736      0530  1022  SET 1 4
0737      0531  1464  /START OF
0738      0532  6523  E6*1
0739      0533  0223  /REQUESTED NAME
0740      0534  6530  JMP ADVIND
0741      0535  1020  LDA 1 2
0742      0536  0001  /CHK INDEX NAME
0743      0537  1140  SAE 1 4
0744      0538  0002  /WITH REQ NAME
0745      0539  0003  JMP LOOKUP+4
0746      0540  0004  XSK 1 3
0747      0541  0005  /NO MATCH -- TRY NEXT NAME
0748      0542  0006  /MATCHED ALL ?
0749      0543  0007  JMP 1-4
0750      0544  0008  /NO TRY NXT PAIR
0751      0545  0009  LDA 1
0752      0546  0010  /INCR POINTER TO ADDR STARTING TBLK
0753      0547  0011  BUMP,
0754      0548  0012  1
0755      0549  0013  ADM
0756      0550  0014  2
0757      0551  0015  XSK 1 1
0758      0552  0016  JMP 1
0759      0553  0017  EJECT
0760      0554  0018  /RTN
0761      0555  0019  -

```



```

1046          / INDEX SLOT SEARCH
1047          / ENTER,      SET I 4      /PTR TO
1048          0064      E6*1      /REQ NAME
1049          0622      2372      /RTN JMP
1050          0623      0041      /PTR TO INDEX
1051          0624      0000      INDEX
1052          0625      0000      /PTR TO INDEX
1053          0626      0000      INDEX
1054          0627      3000      /PTR TO INDEX
1055          0628      6444      INDEX
1056          0629      0045      /PUT REQ NAME
1057          0630      0045      ADVIND
1058          0631      0045      /PUT REQ NAME
1059          0632      0002      SET 5
1060          0633      0002      2
1061          0634      1022      LDA I 2      /FOUND A
1062          0635      1460      SAE I      /BLANK SLOT ?
1063          0636      5757      5757      /NO TRY NXT SLOT
1064          0637      6630      JMP ENTER*6
1065          0638      1024      LDA I 4      /PUT REQ NAME
1066          0639      1065      STA I 5      /IN INDEX
1067          0640      1520      SRO I      /SKP OUT AFT
1068          0641      1020      3567      /4 PASSES
1069          0642      5567      JMP *4      /TO END OF
1070          0643      6637      SET 2      /NAME
1071          0644      0042      5      /BUMP PTR AND EXIT
1072          0645      0005      JMP BUMP*1
1073          0646      6535      /
1074          0647      0000      /
1075          0648      0000      /
1076          0649      0000      / FIND THE BEST SPACE FOR A FILE
1077          0650      0000      /B6 IS
1078          0651      0000      /DYNAMICALLY
1079          0652      0000      /RESET TO
1080          0653      0000      /NEAREST TBLK
1081          0654      0000      -FILE
1082          0655      0000      LDA I
1083          0656      0000      /SWITCH OFF
1084          0657      0000      7777      /LOWER FILE
1085          0658      0000      STC SWITCH
1086          0659      0000      JMP GAPS
1087          0660      0000      /RTN WITH B6 SET
1088          0661      0000      /TO -CLOSEST BLK
1089          0662      0000      SET 7
1090          0663      0000      6
1091          0664      0000      CLR      /SWITCH ON
1092          0665      0000      STC SWITCH
1093          0666      0000      SET 1 6
1094          0667      0000      FREE
1095          0668      0000      JMP GAPS
1096          0669      0000      EJECT
1097          0670      0000
1098          0671      0000
1099          0672      0000
1100          0673      0000
1101          0674      0000
1102          0675      0000
1103          0676      0000
1104          0677      0000
1105          0678      0000
1106          0679      0000
1107          0680      0000
1108          0681      0000
1109          0682      0000
1110          0683      0000
1111          0684      0000
1112          0685      0000
1113          0686      0000
1114          0687      0000
1115          0688      0000
1116          0689      0000
1117          0690      0000
1118          0691      0000
1119          0692      0000
1120          0693      0000
1121          0694      0000

```

```

        /
        /
        / FIND NEAREST SPACE OF TWO
1122
1123
1124
1125
1126      1000    G1   LDA    7    /UPPER FILE
1127      0007    P7,   COM    6    /LOWER FILE
1128      0017    0017   ADD   1    ADA 1
1129      2006    0017   ADD   6    /LOWER FILE
1130      0071    2006   ADD   1    ADA 1
1131      0071    2006   ADD   6    /LOWER FILE
1132      0072    1120   ADD   1    ADA 1
1133      0073    1120   -XBLK-XBLK
1134      0074    0451   APO    *+3  /WHICH IS BEST ?
1135      0075    6700   JMP    H1   /CHK UPPER FILE
1136      0076    6454   JMP    H1   /CHK LOWER FILE
1137      0077    6010   JMP    H1   /OK 1ST TBLK IN AC
1138      0078    1020   LDA    1
1139      0079    1000   LDA    1
1140      007A    1000   1000
1141      007B    2007   ADD   7    /UPPER FILE
1142      007C    2007   JMP    H1+2  /OK
1143      007D    6456   JMP    H1+4  /OK
1144      007E    6710   JMP    H1   /CHK LOWER
1145      007F    6454   JMP    H1   /OK
1146      0080    6010   JMP    H1   /FILE AREA FULL
1147      0081    6010   JMP    SAYNO
1148      0082    6010   LDA    7
1149      0083    6010   JMP    10
1150      0084    6010   LDA    7
1151      0085    6010   JMP    10
1152      0086    6010   LDA    7
1153      0087    6010   COM    10
1154      0088    6010   JMP    10
1155      0089    0041   GAPSR, SET 1
1156      0090    0000   0
1157      0091    0000   SET 1, 2
1158      0092    0000   INDEX:4000
1159      0093    0000   JMP ADVIND
1160      0094    0000   LDH    1, 2
1161      0095    0000   SHD    1
1162      0096    0000   5700
1163      0097    0000   /IS THERE
1164      0098    0000   /AN ENTRY ?
1165      0099    0000   /NO DO SOME MORE
1166      0100    0000   /ADV ENTRY TO
1167      0101    0000   /TBLK INFO
1168      0102    0000   /SECTOR
1169      0103    0000   /RTN P+2 IF
1170      0104    0000   /INVALID
1171      0105    0000   JMP COMPBN
1172      0106    0000   JMP GETBN
1173      0107    0000   JMP COMPBN
1174      0108    0000   JMP GETBN
1175      0109    0757   JMP COMPBN
1176      0110    0754   JMP GAPSR+4
1177      0111    6720   EJECT

```



```

1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284

// RE-WRITE THE INDEX
// ENTERED WITH STARTING TBLK IN AC
0775 0053 RSTRX, SET 13
0776 0000 0 STA 4 / STORE STARTING BLOCK IN INDEX
0777 1044 1560 BCL I
1000 1000 7000 7000 /START BLOCK TO PARAMETER LIST
1002 4321 STC OUTFILE+2 /MINUS FILE LEN
1003 2345 ADD F1
1004 0017 COM
1005 1064 STA 1 4 /STORE LEN IN INDEX
1006 0435 KST I /LAST CHANCE TO
1007 6335 JMP EDRTN /INHIBIT COMMAND
1010 0002 PDP /CALL WRITE IN 8-MODE
PMODE
1011 6201 CDF LO8FLD
1012 6212 CIF HIBFLD
2013 4422 JMS I WRITE /RE-WRITE THE INDEX
5014 5017 X10
5015 6141 LINC
1272 1016 6013 LMODE
JMP 13
1275 /
1276 /
1277 1017 0000 X10, 0 /UNIT NUMBER
1020 0006 6 /MEM ADDR = 3000
1021 0346 XBLK /INDEX TBLK = 346
1022 0002 2 /LENGTH IS 2 BLOCKS
1304 EJECT
-
```

```

// LI, QL, OR AS (BY NAME) ENTER HERE
// LIORAS, CLR DATSEG SET 17 /FORCE DATA FIELD TO ACCESS E6
0011 0641
1305 1023 0024 0057
1306 1311 1025 1026 2371
1307 1312 1027 1440 SAE
1308 1313 1028 2372 E6+1
1309 1314 1029 1030 2373
1310 1315 1031 1032 1440 SAE
1311 1316 1033 1034 2374
1312 1317 1035 1036 2375 /SKIP IF LN2 IS ZERO
1313 1320 1037 1038 2376 /BOMB IF LN1 IS ZERO
1314 1321 1039 1040 2377 /PICK UP LN1
1315 1322 1041 1042 1200
1316 1323 1043 1044 2378 /LN2 = LN1
1317 1324 1045 1046 1201
1318 1325 1047 1048 2379 /LN2 TO 17
1319 1326 1049 1050 1202
1320 1327 1051 1052 2380 /NO INDEX
1321 1330 1053 1054 1203 /SEARCH FOR NAME MATCH
1322 1331 1055 1056 2381 /NO NAME MATCH
1323 1332 1057 1058 1204 /SOURCE OR BINARY ?
1324 1333 1059 1060 2382 /SOURCE IS THERE, CONTINUE
1325 1334 1061 1062 1205 /BINRAY ONLY
1326 1335 1063 1064 2383 /BINRAY ONLY
1327 1336 1065 1066 1206
1328 1337 1067 1068 2384 /QL OR LI WD
1329 1338 1069 1070 1207
1330 1339 1071 1072 2385 /START BLOCK TO AC
1331 1340 1073 1074 1208
1332 1341 1075 1076 2386 /LIF KBDSEG
1333 1342 1077 1078 1209
1334 1343 1079 1080 2387 /JMP KBDOPR-1
1335 1344 1081 1082 1210
1336 1345 1083 1084 2388 /RTN TO EDITOR
1337 1346 1085 1086 1211
1338 1347 1087 4777 STC UNITNO
1339 1348 1088 1089 LDA
1340 1349 1090 2000
1341 1350 1091 2001
1342 1351 1092 2002
1343 1352 1093 2003
1344 1353 1094 2004
1345 1354 1095 2005
1346 1355 1096 2006
1347 1356 1097 2007
1348 1357 1098 2008
1349 1358 1099 2009
1350 1359 1100 2010
1351 1360 1101 2011
1352 1361 1102 2012
1353 1362 1103 2013
1354 1363 1104 2014
1355 1364 1105 2015
1356 1365 1106 2016
1357 1366 1107 2017
1358 1367 1108 2018
1359 1368 1109 2019
1360 1369 1110 2020
1361 1370 1111 2021
1362 1371 1112 2022
1363 1372 1113 2023
1364 1373 1114 2024
1365 1374 1115 2025
1366 1375 1116 2026
1367 1376 1117 2027
1368 1377 1118 2028
1369 1378 1119 2029
1370 1379 1120 2030
1371 1380 1121 2031
1372 1381 1122 2032
1373 1382 1123 2033
1374 1383 1124 2034
1375 1384 1125 2035
1376 1385 1126 2036
1377 1386 1127 2037
1378 1387 1128 2038
1379 1388 1129 2039
1380 1389 1130 2040
1381 1390 1131 2041
1382 1391 1132 2042
1383 1392 1133 2043
1384 1393 1134 2044
1385 1394 1135 2045
1386 1395 1136 2046
1387 1396 1137 2047
1388 1397 1138 2048
1389 1398 1139 2049
1390 1399 1140 2050
1391 1390 1141 2051
1392 1391 1142 2052
1393 1392 1143 2053
1394 1393 1144 2054
1395 1394 1145 2055
1396 1395 1146 2056
1397 1396 1147 2057
1398 1397 1148 2058
1399 1398 1149 2059
1400 1399 1150 2060
1401 1400 1151 2061
1402 1401 1152 2062
1403 1402 1153 2063
1404 1403 1154 2064
1405 1404 1155 2065
1406 1405 1156 2066
1407 1406 1157 2067
1408 1407 1158 2068
1409 1408 1159 2069
1410 1409 1160 2070
1411 1410 1161 2071
1412 1411 1162 2072
1413 1412 1163 2073
1414 1413 1164 2074
1415 1414 1165 2075
1416 1415 1166 2076
1417 1416 1167 2077
1418 1417 1168 2078
1419 1418 1169 2079
1420 1419 1170 2080
1421 1420 1171 2081
1422 1421 1172 2082
1423 1422 1173 2083
1424 1423 1174 2084
1425 1424 1175 2085
1426 1425 1176 2086
1427 1426 1177 2087
1428 1427 1178 2088
1429 1428 1179 2089
1430 1429 1180 2090
1431 1430 1181 2091
1432 1431 1182 2092
1433 1432 1183 2093
1434 1433 1184 2094
1435 1434 1185 2095
1436 1435 1186 2096
1437 1436 1187 2097
1438 1437 1188 2098
1439 1438 1189 2099
1440 1439 1190 2100
1441 1440 1191 2101
1442 1441 1192 2102
1443 1442 1193 2103
1444 1443 1194 2104
1445 1444 1195 2105
1446 1445 1196 2106
1447 1446 1197 2107
1448 1447 1198 2108
1449 1448 1199 2109
1450 1449 1200 2110
1451 1450 1201 2111
1452 1451 1202 2112
1453 1452 1203 2113
1454 1453 1204 2114
1455 1454 1205 2115
1456 1455 1206 2116
1457 1456 1207 2117
1458 1457 1208 2118
1459 1458 1209 2119
1460 1459 1210 2120
1461 1460 1211 2121
1462 1461 1212 2122
1463 1462 1213 2123
1464 1463 1214 2124
1465 1464 1215 2125
1466 1465 1216 2126
1467 1466 1217 2127
1468 1467 1218 2128
1469 1468 1219 2129
1470 1469 1220 2130
1471 1470 1221 2131
1472 1471 1222 2132
1473 1472 1223 2133
1474 1473 1224 2134
1475 1474 1225 2135
1476 1475 1226 2136
1477 1476 1227 2137
1478 1477 1228 2138
1479 1478 1229 2139
1480 1479 1230 2140
1481 1480 1231 2141
1482 1481 1232 2142
1483 1482 1233 2143
1484 1483 1234 2144
1485 1484 1235 2145
1486 1485 1236 2146
1487 1486 1237 2147
1488 1487 1238 2148
1489 1488 1239 2149
1490 1489 1240 2150
1491 1490 1241 2151
1492 1491 1242 2152
1493 1492 1243 2153
1494 1493 1244 2154
1495 1494 1245 2155
1496 1495 1246 2156
1497 1496 1247 2157
1498 1497 1248 2158
1499 1498 1249 2159
1500 1499 1250 2160
1501 1500 1251 2161
1502 1501 1252 2162
1503 1502 1253 2163
1504 1503 1254 2164
1505 1504 1255 2165
1506 1505 1256 2166
1507 1506 1257 2167
1508 1507 1258 2168
1509 1508 1259 2169
1510 1509 1260 2170
1511 1510 1261 2171
1512 1511 1262 2172
1513 1512 1263 2173
1514 1513 1264 2174
1515 1514 1265 2175
1516 1515 1266 2176
1517 1516 1267 2177
1518 1517 1268 2178
1519 1518 1269 2179
1520 1519 1270 2180
1521 1520 1271 2181
1522 1521 1272 2182
1523 1522 1273 2183
1524 1523 1274 2184
1525 1524 1275 2185
1526 1525 1276 2186
1527 1526 1277 2187
1528 1527 1278 2188
1529 1528 1279 2189
1530 1529 1280 2190
1531 1530 1281 2191
1532 1531 1282 2192
1533 1532 1283 2193
1534 1533 1284 2194
1535 1534 1285 2195
1536 1535 1286 2196
1537 1536 1287 2197
1538 1537 1288 2198
1539 1538 1289 2199
1540 1539 1290 2200
1541 1540 1291 2201
1542 1541 1292 2202
1543 1542 1293 2203
1544 1543 1294 2204
1545 1544 1295 2205
1546 1545 1296 2206
1547 1546 1297 2207
1548 1547 1298 2208
1549 1548 1299 2209
1550 1549 1300 2210
1551 1550 1301 2211
1552 1551 1302 2212
1553 1552 1303 2213
1554 1553 1304 2214
1555 1554 1305 2215
1556 1555 1306 2216
1557 1556 1307 2217
1558 1557 1308 2218
1559 1558 1309 2219
1560 1559 1310 2220
1561 1560 1311 2221
1562 1561 1312 2222
1563 1562 1313 2223
1564 1563 1314 2224
1565 1564 1315 2225
1566 1565 1316 2226
1567 1566 1317 2227
1568 1567 1318 2228
1569 1568 1319 2229
1570 1569 1320 2230
1571 1570 1321 2231
1572 1571 1322 2232
1573 1572 1323 2233
1574 1573 1324 2234
1575 1574 1325 2235
1576 1575 1326 2236
1577 1576 1327 2237
1578 1577 1328 2238
1579 1578 1329 2239
1580 1579 1330 2240
1581 1580 1331 2241
1582 1581 1332 2242
1583 1582 1333 2243
1584 1583 1334 2244
1585 1584 1335 2245
1586 1585 1336 2246
1587 1586 1337 2247
1588 1587 1338 2248
1589 1588 1339 2249
1590 1589 1340 2250
1591 1590 1341 2251
1592 1591 1342 2252
1593 1592 1343 2253
1594 1593 1344 2254
1595 1594 1345 2255
1596 1595 1346 2256
1597 1596 1347 2257
1598 1597 1348 2258
1599 1598 1349 2259
1600 1599 1350 2260
1601 1600 1351 2261
1602 1601 1352 2262
1603 1602 1353 2263
1604 1603 1354 2264
1605 1604 1355 2265
1606 1605 1356 2266
1607 1606 1357 2267
1608 1607 1358 2268
1609 1608 1359 2269
1610 1609 1360 2270
1611 1610 1361 2271
1612 1611 1362 2272
1613 1612 1363 2273
1614 1613 1364 2274
1615 1614 1365 2275
1616 1615 1366 2276
1617 1616 1367 2277
1618 1617 1368 2278
1619 1618 1369 2279
1620 1619 1370 2280
1621 1620 1371 2281
1622 1621 1372 2282
1623 1622 1373 2283
1624 1623 1374 2284
1625 1624 1375 2285
1626 1625 1376 2286
1627 1626 1377 2287
1628 1627 1378 2288
1629 1628 1379 2289
1630 1629 1380 2290
1631 1630 1381 2291
1632 1631 1382 2292
1633 1632 1383 2293
1634 1633 1384 2294
1635 1634 1385 2295
1636 1635 1386 2296
1637 1636 1387 2297
1638 1637 1388 2298
1639 1638 1389 2299
1640 1639 1390 2300
1641 1640 1391 2301
1642 1641 1392 2302
1643 1642 1393 2303
1644 1643 1394 2304
1645 1644 1395 2305
1646 1645 1396 2306
1647 1646 1397 2307
1648 1647 1398 2308
1649 1648 1399 2309
1650 1649 1400 2310
1651 1650 1401 2311
1652 1651 1402 2312
1653 1652 1403 2313
1654 1653 1404 2314
1655 1654 1405 2315
1656 1655 1406 2316
1657 1656 1407 2317
1658 1657 1408 2318
1659 1658 1409 2319
1660 1659 1410 2320
1661 1660 1411 2321
1662 1661 1412 2322
1663 1662 1413 2323
1664 1663 1414 2324
1665 1664 1415 2325
1666 1665 1416 2326
1667 1666 1417 2327
1668 1667 1418 2328
1669 1668 1419 2329
1670 1669 1420 2330
1671 1670 1421 2331
1672 1671 1422 2332
1673 1672 1423 2333
1674 1673 1424 2334
1675 1674 1425 2335
1676 1675 1426 2336
1677 1676 1427 2337
1678 1677 1428 2338
1679 1678 1429 2339
1680 1679 1430 2340
1681 1680 1431 2341
1682 1681 1432 2342
1683 1682 1433 2343
1684 1683 1434 2344
1685 1684 1435 2345
1686 1685 1436 2346
1687 1686 1437 2347
1688 1687 1438 2348
1689 1688 1439 2349
1690 1689 1440 2350
1691 1690 1441 2351
1692 1691 1442 2352
1693 1692 1443 2353
1694 1693 1444 2354
1695 1694 1445 2355
1696 1695 1446 2356
1697 1696 1447 2357
1698 1697 1448 2358
1699 1698 1449 2359
1700 1699 1450 2360
1701 1700 1451 2361
1702 1701 1452 2362
1703 1702 1453 2363
1704 1703 1454 2364
1705 1704 1455 2365
1706 1705 1456 2366
1707 1706 1457 2367
1708 1707 1458 2368
1709 1708 1459 2369
1710 1709 1460 2370
1711 1710 1461 2371
1712 1711 1462 2372
1713 1712 1463 2373
1714 1713 1464 2374
1715 1714 1465 2375
1716 1715 1466 2376
1717 1716 1467 2377
1718 1717 1468 2378
1719 1718 1469 2379
1720 1719 1470 2380
1721 1720 1471 2381
1722 1721 1472 2382
1723 1722 1473 2383
1724 1723 1474 2384
1725 1724 1475 2385
1726 1725 1476 2386
1727 1726 1477 2387
1728 1727 1478 2388
1729 1728 1479 2389
1730 1729 1480 2390
1731 1730 1481 2391
1732 1731 1482 2392
1733 1732 1483 2393
1734 1733 1484 2394
1735 1734 1485 2395
1736 1735 1486 2396
1737 1736 1487 2397
1738 1737 1488 2398
1739 1738 1489 2399
1740 1739 1490 2400
1741 1740 1491 2401
1742 1741 1492 2402
1743 1742 1493 2403
1744 1743 1494 2404
1745 1744 1495 2405
1746 1745 1496 2406
1747 1746 1497 2407
1748 1747 1498 2408
1749 1748 1499 2409
1750 1749 1500 2410
1751 1750 1501 2411
1752 1751 1502 2412
1753 1752 1503 2413
1754 1753 1504 2414
1755 1754 1505 2415
1756 1755 1506 2416
1757 1756 1507 2417
1758 1757 1508 2418
1759 1758 1509 2419
1760 1759 1510 2420
1761 1760 1511 2421
1762 1761 1512 2422
1763 1762 1513 2423
1764 1763 1514 2424
1765 1764 1515 2425
1766 1765 1516 2426
1767 1766 1517 2427
1768 1767 1518 2428
1769 1768 1519 2429
1770 1769 1520 2430
1771 1770 1521 2431
1772 1771 1522 2432
1773 1772 1523 2433
1774 1773 1524 2434
1775 1774 1525 2435
1776 1775 1526 2436
1777 1776 1527 2437
1778 1777 1528 2438
1779 1778 1529 2439
1780 1779 1530 2440
1781 1780 1531 2441
1782 1781 1532 2442
1783 1782 1533 2443
1784 1783 1534 2444
1785 1784 1535 2445
1786 1785 1536 2446
1787 1786 1537 2447
1788 1787 1538 2448
1789 1788 1539 2449
1790 1789 1540 2450
1791 1790 1541 2451
1792 1791 1542 2452
1793 1792 1543 2453
1794 1793 1544 2454
1795 1794 1545 2455
1796 1795 1546 2456
1797 1796 1547 2457
1798 1797 1548 2458
1799 1798 1549 2459
1800 1799 1550 2460
1801 1800 1551 2461
1802 1801 1552 2462
1803 1802 1553 2463
1804 1803 1554 2464
1805 1804 1555 2465
1806 1805 1556 2466
1807 1806 1557 2467
1808 1807 1558 2468
1809 1808 1559 2469
1810 1809 1560 2470
1811 1810 1561 2471
1812 1811 1562 2472
1813 1812 1563 2473
1814 1813 1564 2474
1815 1814 1565 2475
1816 1815 1566 2476
1817 1816 1567 2477
1818 1817 1568 2478
1819 1818 1569 2479
1820 1819 1570 2480
1821 1820 1571 2481
1822 1821 1572 2482
1823 1822 1573 2483
1824 1823 1574 2484
1825 1824 1575 2485
1826 1825 1576 2486
1827 1826 1577 2487
1828 1827 1578 2488
1829 1828 1579 2489
1830 1829 1580 2490
1831 1830 1581 2491
1832 1831 1582 2492
1833 1832 1583 2493
1834 1833 1584 2494
1835 1834 1585 2495
1836 1835 1586 2496
1837 1836 1587 2497
1838 1837 1588 2498
1839 1838 1589 2499
1840 1839 1590 2500
1841 1840 1591 2501
1842 1841 1592 2502
1843 1842 1593 2503
1844 1843 1594 2504
1845 1844 1595 2505
1846 1845 1596 2506
1847 1846 1597 2507
1848 1847 1598 2508
1849 1848 1599 2509
1850 1849 1600 2510
1851 1850 1601 2511
1852 1851 1602 2512
1853 1852 1603 2513
1854 1853 1604 2514
1855 1854 1605 2515
1856 1855 1606 2516
1857 1856 1607 251
```

```

1360
1361
1362
1363    / / COME HERE TO DISPLAY "NO"
1364    1066 0000 NOPE, 0           /PMODE ENTRY POINT
1365    1067 6141             /ENTER LINE MODE
1366    1068 7071             JMP .+1
1367    1069 0050 SAYNO, SET 10  /SAVE RETURN FOR DEBUG
1368    1070 0000             2
1369    1071 0071             LOC A,1
1370    1072 0000             /GET FULL SIZE
1371    1073 1020             2200
1372    1074 0200             ESF
1373    1075 0004             CLR
1374    1076 0011             /Y-COORDINATE ZERO (MIDDLE SCREEN)
1375    1077 0061             SET I,1
1376    1100 0344             /INITIAL X-COORDINATE
1377    1101 1760             344
1378    1102 3077             DSC I
1379    1103 1760             /DISPLAY N
1380    1104 7706             DSC I
1381    1105 0061             7706
1382    1106 0374             SET I,1
1383    1107 1760             /ADJUST X-COORDINATE TO NEXT CHAR
1384    1108 4177             374
1385    1109 4177             DSC I
1386    1110 4177             /DISPLAY O
1387    1111 1760             4177
1388    1112 7741             DSC I
1389    1113 0415             7741
1390    1114 7077             KST
1391    1115 0011             JMP SAYNO+6
1392    1116 0004             CLR
1393    1117 6335             ESF
1394    1118 0004             /RESET TO HALF SIZE
1395    1119 6335             JMP EDRTN
1396    1120 EJECT             /THEN GO TO EDITOR

```

```

1416      / ADD BINARY ENTERS HERE
1420      / ADDBIN, LDF DATSEG   /FORCE DATA FIELD TO GET E6
1421      LDH DATSEG          /FIRST CHAR OF NAME
1422      0641 1300           E6+2
1423      1121 2373           SHD 1
1424      1122 1420           /WAS NAME SPECIFIED?
1425      1123 7720           7720
1426      1124 7025           JMP SAYNO
1427      1125 7071           /NO - CANT DO IT
1428      1126 1020           LDA I
1429      1127 0003           /SET BUMP CONST
1430      P3,
1431      1130 4536           STC BUMP
1432      1131 6464           /... IN LOOKUP
1433      1132 7071           JMP CHKIND
1434      1133 6517           /GET INDEX, CHECK FOR VALIDITY
1435      1134 7071           JMP SAYNO
1436      1135 6436           JMP LOOKUP
1437      1136 0456           /NO INDEX - TOO BAD
1438      1137 7071           JMP SRORBN
1439      1138 0000           SKP
1440      1139 0000           /IS THERE A BINARY?
1441      1140 0000           JMP SAYNO
1442      1141 0000           /NO - TOO BAD
1443      // WHEN WE GET HERE WEVE FOUND THE FILE TO ADD
1444      1142 1040           LDA 2
1445      1143 1166           STA
1446      1144 2347           FHDRIN+2
1447      1145 2347           ADD P1
1448      1146 5507           /SECOND BLOCK
1449      1147 1022           STC BINIPT+2
1450      1148 1022           /...IS START OF BINARY
1451      1149 3163           LDA 12
1452      1150 1040           ADD M1
1453      1151 5504           STA
1454      1152 0017           /...MINUS ONE
1455      1153 2347           /...IS BINARY BLOCK COUNT
1456      1154 3017           STA
1457      1155 1040           FHDRIN
1458      1156 1164           STC BINIPT
1459      1157 5525           ADD P7
1460      1158 2667           /3400 SHIFTED 8
1461      1159 1040           STC HDRIO+1
1462      1160 1164           ADD P7
1463      1161 4324           /READ HEADER AT 3400
1464      1162 7200           JMP AB0000
1465      1163 7776           /GO TO NEXT PAGE
1466      1164 0000           1
1467      1165 0000           -1
1468      1166 0000           /UNIT FOR FILE
1469      1167 0001           2
1470      1168 0000           /HEADER TO 3000
1471      1169 0000           6
1472      1170 0000           2
1473      1171 0000           1
1474      1172 0000           EJECT
1475      1173 0000           -

```

1476
 1477
 1500 1200 0002 AB0000,
 *1200 POP PMODE
 CIF H18FLD
 JMS I READ FHDRIN
 /READ BINARY FILE HEADER
 JMS I PFILBF
 /... AND UP TO 15 BLOCKS OF FILE
 JMS I H18FLD
 JMS I READ HDRIO
 /SCAN HEADER FOR STANDARD RELOCATION
 JMS SCANHD
 TAD I PE6A
 CLL RAR
 /MOVE TO LINK
 SZA
 /SKIP IF FIELD 0 OR 1
 JMS I PNOPE
 /OTHERWISE ERROR
 TAD I PE6
 /GET RELOCATION ADDR
 SZA
 /IF RELOC ADDR...
 JMP ,*3
 /...AND FIELD ARE ZERO...
 JMP AB010
 /...DO NOT RELOCATE
 AND P377
 /ADDR WITHIN BLOCK
 DCA RELADR
 TAD I PE6
 AND P7400
 /BLOCK NO - HIGH ORDER BIT IN LINC
 RTL
 RTL
 / MOVE
 / TO
 RAL
 / LOW ORDER
 DCA RELBLK
 /RELOCATED BLOCK ADDR
 EJECT
 1525 5225 0350
 1526 5226 7006
 1527 5227 7006
 1530 5230 7004
 1531 5231 3356
 1532

```

1533 // NOW BEGINS THE REAL STUFF
1534 // SEARCH FOR NON-ZERO IN INPUT
1535 //
1536 //
1537 5232 AB010, CDF HIBFLD /GET ONE BINARY WORD
1538 5233 1751 TAD I PINDAT /DO WE MOVE IT?
1539 5234 7640 SZA CLA /YUP
1540 5235 4301 JMS MOVED /INCR POINTER
1541 5236 2351 ISZ PINDAT /PICK IT UP
1542 5237 1351 TAD PINDAT /TEST FOR EOB
1543 5238 0347 AND P377 /SKIP IF END OF BLOCK
1544 5240 7640 SZA CLA /ELSE LOOP
1545 5241 5233 JMP AB020
1546 //
1547 //
1548 // END OF INPUT BLOCK
1549 //
1550 //
1551 //
1552 //
1553 5243 6201 CDF LO8FLD /INDIRECTS TO THIS FIELD
1554 5244 2277 ISZ MBLKS /ANY MORE INPUT?
1555 5245 5254 JMP NXTBLC /YES - GO GET EM
1556 5246 4745 JMS I PWRBLK /THATS ALL - WRITE OUT CURRENT BUFFER
1557 5247 6212 CIF H18FLD /...AND THE HEADER
1558 5250 4422 JMS I WRITE
1559 5251 4323 HDR10
1560 5252 2653 JMP I .+1 /GO BACK TO EDIT
1561 5253 4336 EDRTN+1
1562 //
1563 //
1564 //
1565 //
1566 5254 7300 NXTBLC, CLA CLL /ALL CLEAR
1567 5255 1351 TAD PINDAT /INPUT DATA ADDR
1568 5256 1276 TAD BUFEND /SET LINK IF END OF BUFFER
1569 5257 7630 S2L CLA /SKIP IF MORE IN CORE
1570 5258 4700 JMS I PFILBF /ELSE REFILL BUFFERS
1571 5259 4263 JMS SCANHD /ADJUST RELOCATION POINTERS
1572 5260 5232 JMP AB010 /CONTINUE
1573 //
1574 //
1575 //
1576 //
1577 //
1578 //
1579 //
1580 //
1581 5263 0000 SCANHD, 2 /SCAN INPUT HEADER TO FIND NEXT USED BLOCK
1582 5264 2356 ISZ RELBLK /INCR RELOCATION BLOCK NO
1583 5265 2275 ISZ PINHDR /INCR BLOCK MAP POINTER
1584 5266 1675 TAD I PINHDR /GET A WORD OF MAP
1585 5267 7650 SNA CLA /IS BLOCK USED?
1586 5268 4264 JMP SCANHD+1 /NO - TRY NEXT
1587 5269 5232 JMP I SCANHD /YES - RETURN
1588 //
1589 //
1590 //
1591 5272 5066 PNOPE, NOPE /PMODE SAYNO
1592 5273 2371 PEO, E6
1593 5274 2372 PEO, E6+1
1594 5275 3340 PINHDR, 3340 /POINTER TO INPUT FILE HEADER CONTROL WORD
1595 5276 10000 BUFFEND, -7000 /CAUTION: THIS IS SNEAKY
1596 5277 00000 MBLKS, 0
1597 5278 0456 PFILBF, FILBUF EJECT

```

1621	1623	5301	0000
1624	1625	5302	135
1626	1627	5303	034
1628	1629	5304	135
1630	1631	5305	335
1632	1633	5306	135
1634	1635	5311	531
1636	1637	5312	135
1638	1639	5313	034
1640	1641	5314	335
1642	1643	5315	765
1644	1645	5321	700
1646	1647	5322	135
1648	1649	5323	135
1650	1651	5324	534
1652	1653	5325	620
1654	1655	5326	135
1656	1657	5327	765
1658	1659	5328	135
1660	1661	5329	135
1662	1663	5330	770
1664	1665	5331	135
1666	1667	5332	135
1668	1669	5333	135
1670	1671	5334	524
1672	1673	5335	524
1674	1675	5336	524
1676	1677	5337	524
1678	1679	5338	524
1680	1681	5339	524
1682	1683	5340	524
1684	1685	5341	524
1686	1687	5342	524
1688	1689	5343	524
1690	1691	5350	524
1692	1693	5351	524
1694	1695	5352	524
1696	1697	5353	524
1698	1699	5354	524
1700	1701	5355	524
1702	1703	5356	524
1704	1705	5357	524
1706	1707	5358	524
1708	1709	5359	524
1710	1711	5360	524
1712	1713	5361	524
1714	1715	5362	524
1716	1717	5363	524
1718	1719	5364	524
1720	1721	5365	524
1722	1723	5366	524

WEVE GOT A NON-ZERO WORD

*5400

```

1710
1711
1712
1713
1714      54000 00000          GETBLK, 0
1715      54001 3254           DCA BWA10+2 /BLOCK WANTED
1716      54002 1254           TAD BWA10+2
1717      54003 1251           TAD POUTHD /ADDRESS CTL WD
1720      54004 3246           DCA TMP2 /GET CONTROL WD
1721      54005 1646           TAD 1 TMP2 /SKIP IF BULK USE
1722      54006 7650           CLA SNA /ELSE CLEAR BUFF
1723      54007 5216           JMP CLRBUF

1724      / READ THE APPROPRIATE BLOCK OF BWA INTO WORKING BUFFER
1725
1726
1727      5410 6212           CIF H18FLD
1730      5411 4421           JMS I READ /GET IT
1731      5412 5452           BWA10
1732      5413 6211           CDF H18FLD /SET UPPER FIELD FOR MOVE
1733      5414 7200           CLA
1734      5415 5600           JMP I GETBLK /RETURN

1735      / THIS BLOCK OF BWA IS NOT USED
1736      / CLEAR THE BUFFER
1737
1740      5416 1247           CLRBUF, TAD M400 /SET COUNT
1742      5417 3246           DCA TMP2 /-1 TO AC
1743      5420 7040           CMA AUTO /SET AUTO-INDEX
1744      5421 3010           DCA AUTO
1745      5422 6211           CDF H18FLD
1746      5423 3410           DCA 1 AUTO
1747      5424 2246           ISZ TMP2 /LOOP TILL CLEAR
1750      5425 5223           JMP ;-2 /THEN RETURN
1751      5426 5600           JMP I GETBLK

1752
1753
1754      5427 0000           WRTBLK, 0
1755      5430 6212           CIF H18FLD
1756      5431 4422           JMS I WRITE /SEND IT OUT
1760      5432 5452           BWA10
1761      5433 7200           CLA
1762      5434 1254           TAD BWA10+2 /BLOCK NO.**
1763      5435 1251           TAD POUTHD /...PLUS HEADER ADDRESS.**
1764      5436 5246           DCA TMP2 /...IS ADDR OF CONTROL WORD
1765      5437 1646           TAD 1 TMP2 /GET IT
1766      5440 7640           SZA CLA /TEST FOR ALREADY USED
1767      5441 5627           JMP I WRTBLK /YES - RETURN
1770      5442 7240           CMA /NO - UPDATE HDR
1771      5443 3646           DCA I TMP2 /SET USED INDICATOR
1772      5444 2650           ISZ I PBUSE /INCR BLOCKS USED COUNT
1773      5445 5627           JMP I WRTBLK /NOW RETURN

1774
1775      5446 0000           1 TMP2, 0
1776      5447 7400           M400, -400
1777      5450 3737           PBUSE, USENO:2000
1780      5451 3740           POUTHD, MAP:2000
1781      5452 0111           BWA10, 111
2001      5453 0020           BUFFER, 0
2002      5454 0000           /BWA UNIT /ADDR = 1 0000
2003      5455 0001           /BLOCK NO /ONE AT A TI,
2004
2005

```

2006
2007
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2030
2031
2032
2033
2034
2035
2036
2037
2040
2041
2042
2043
2044
2045

/ FILL BUFFERS FROM THE BINARY FILE
FILBUF, 0 TAD BLOCKS /GET REMAINING COUNT
TAD M15 /SUBTRACT BUFFER LENGTH
SMA /SKIP IF NOT ENOUGH TO FILL BUFFER
JMP FB010 /ELSE READ 15
CLA
TAD BLOCKS /REMAINING BLKS
DCA BINIPT+3 /TO I/O LIST
DCA BLOCKS /STORE NEW REMAINING COUNT
CIF HIBFLD
JMS I READ /READ UP TO 15
BINIPT
CLA /ADJUST STRT BLK
TAD BINIPT+2 /BY LENGTH
TAD BINIPT+3
DCA BINIPT+2
DCA BINIPT+2 /START OF FIRST BLOCK READ
TAD INSTRT /RESET POINTER
DCA I PPINDT /RETURN
JMP I FILBUF
/
M15, -15
INSTRT, 400
PPINDT, PINDAT
BLOCKS, 0
BINIPT, 0 /UNIT WITH FILE
BUFFER+1 /ADDR = 10400
0 /START BLOCK
15 /NO TO READ
EJECT

/ / / / / SPECIAL SYMBOLS

SPECIAL SYMBOLS

NO EIS

SYMBOL	VALUE	DEF	REFERENCES
AB000	5200	1477	1466
AB010	5232	1537	1521 1575
AB020	5233	1540	1547
AB030	5247	1557	1657
ADDIN	5120	1421	0044
ADVIN	4444	0634	0742 1056 1161
AUTO	0010	2077	0263 0271 1744 1746
A6	2001	2062	1003
BFAD	4332	0447	0350
BFLN	4331	0446	0352
BINPT	5505	2041	1450 1463 2020 2024 2026 2027 2030
BLOCKS	5504	2040	1454 2012 2017 2021
BUFEND	5276	1615	1571
BUFFER	0020	2102	0426 0433 0447 0450 0451 2002 2042
BUFLEN	0016	2103	0430 0435 0446 0451
BUMP	4536	0751	0223 0742 1431
BWA	0111	2075	0242 0437
BWA10	5452	2001	1715 1716 1731 1760 1762
CHKIND	4464	0663	0224 0507 0543 1334 1432
CLRBUF	5416	1741	1723
COMPBN	4757	1224	1173 1175
CRI	4300	2071	0777
CURBLK	5354	1704	1645 1651 1661
DATSEG	0001	2106	0067 0502 0535 0575 0677 1312 1421
DIALU	0100	2072	0603
DRSTRT	7777	2070	0461
EDRTN	4335	0456	0343 0412 1051 1263 1414 1563
ENTER	4622	1050	0233 0516 1064
E6	2371	2055	0073 0112 0123 0143 0153 0162 0171 0504 0537 0560 0666 0741 1051 1314 1316 1321 1324 1327 1344 1423 1612 1
FB010	5466	2021	2015
FCSA	4020	0043	
FHDRIN	5164	1471	1446 1462 1503
FILBUF	5456	2011	1617 2033
FILE	0470	2054	1104
FINDSP	4647	1101	0237 0522
FREE	0270	2061	1117
F1	4345	0476	0217 0651 1233 1256
GAPSR	4714	1155	1110 1120 1176 1241
GETBLK	5400	1714	1674 1734 1751
GETBN	4735	1201	1171 1174
GETPS	4421	0566	0550
G1	4666	1126	
HDRBLK	0057	2076	0441
HDR10	4523	0437	0064 0246 0255 0334 1465 1507 1561
HDRSW	4350	0445	0326 0531
HEADER	1400	2063	0071 0121 0126 0136 0141 0151 0154 2064 2065
H18FLD	0010	2104	0062 0274 0332 0335 0401 0404 0457 0570 0672 1267 1501 1505 1537 1557 1727 1732 1745 1756 2022
H1	4454	0647	1136 1143 1145
INDEX	3000	2060	0701 0716 0735 1055 1160
INF1	4513	0425	0243 0252 0276 0306 0314 0315 2317 0351 0355 0375 0376 0403 0414 0415
INSTRT	5502	2056	2051
KBDOPR	1400	2066	1356
KBDSTG	0003	2107	1355
K2	4573	1017	1000
L10RAS	5023	1511	0247
LOOKUP	4517	0752	0226 0511 0545 0745 1536 1434
LOBFL	0000	2105	0061 1266 1553 1650
MAKIN	4505	0715	0225 0510
MAP	1740	2065	0176 0260 2000
MBCNT	4327	0244	0221 0341 0367 0373 0400 0410 0501

SYMBOL VALUE DEF REFERENCES

MFFEND	4334	0451	0316	
MBLK5	5277	1616	1457	1554
MOVEIT	5541	1667	1647	
MOVEWD	5301	1624	1542	1671
MW010	5316	1641	1634	
M1	5163	1470	1452	
M15	5501	2035	2015	
M40	5346	1676	1655	
M400	5447	1776	1741	
NOPE	5066	1363	1611	
NXTBLK	5254	1567	1555	
OUTBLK	5353	1703	1642	1654
OUTFIL	4317	0432	0244	0250
PBUSE	5450	1777	1772	0256
PDPMOD	4070	0134	0100	0307
PE6	5273	1612	1515	1524
PE6A	5274	1613	1611	
PFILBF	5300	1617	1504	1573
PGETBK	5344	1674	1663	
PINDAT	5351	1701	1540	1543
PINHDR	5275	1614	1603	1604
PNOPE	5272	1611	1514	
POUTDT	5352	1702	1630	1631
POUTHD	2451	2000	1717	1763
PPINDT	5503	2037	2032	
PRNPRG	4376	0555	0050	
PSBLK	0063	2073	0605	
PSENT	1000	2067	0577	
PSIN	4431	0603	0572	
PSWA	4413	0555	0542	
PWRBLK	5345	1675	1556	1653
P1	4347	0500	0214	0247
P2	4161	0236	0220	
P3	5127	1430	0222	
P377	5347	1677	1522	1545
P7	4667	1127	1464	
P7400	5350	1700	1525	1632
READ	0021	2100	0063	0275
RELAADR	5355	1705	1523	1627
RELBLK	5356	1706	1531	1602
REPLAC	4543	0764	0231	0514
RPLSTR	4615	1042	0774	
RSTRX	4775	1250	0240	0523
SAVBIN	4026	0057	0043	
SAVPRG	4541	0472	0046	
SAYNO	2071	1366	0167	0203
SB010	4111	0161	0127	0234
SB020	4134	0211	0076	
SB110	4206	0271	0356	
SB120	4223	0314	0301	
SB132	4231	0326	0310	
SB142	4240	0355	0330	
SB160	4254	0355	0273	0321
SCANHD	5263	1601	1510	1574
SETFLD	4060	0120	0110	0100
SP000	4566	0515	0515	0515

SYMBOL	VALUE	DEF	REFERENCES
SP120	4271	0400	0371
SRORB_N	4436	0621	0230 0513 0547 1340 1436
SWA	0110	2074	0425 0556
SWITCH	4751	1215	1107 1115
TMP2	5446	1775	1720 1721 1742 1747 1764 1765 1771
UNITNO	0777	2056	1350 1355
USENO	1737	2064	0211 1777
WA	0370	2053	0473
WRITE	0022	2101	0333 0336 0405 1270 1560 1757
WRTBLK	5427	1755	1675 1767 1773
XBLK	0346	2057	1133 1302
X10	5017	1300	0253 0524 0667 0674 1271 1460
ZERO	4435	0610	0562