

**PROGRAM STORE COPY, COMPARE,  
AND DUMP PROCEDURES AND CALL STORE DUMP PROCEDURE  
2-WIRE NO. 1 ELECTRONIC SWITCHING SYSTEM**

CONTENTS	PAGE
1. GENERAL . . . . .	1
2. APPARATUS AND DOCUMENTATION . . . . .	2
3. PRELIMINARY PROCEDURE . . . . .	2
4. PROGRAM STORE BUS CONFIGURATION PROCEDURE . . . . .	2
5. PROGRAM STORE COPY (CPY) . . . . .	3
A. CPY Program Description . . . . .	3
B. CPY Operating Instructions . . . . .	4
6. PROGRAM STORE COMPARE (CMP) . . . . .	9
A. CMP Program Description . . . . .	9
B. CMP Operating Instructions . . . . .	11
7. PROGRAM STORE DUMP (PSDMP) . . . . .	12
A. PSDMP Program Description . . . . .	12
B. PSDMP Operating Instructions . . . . .	13
8. CALL STORE DUMP (CSDMP) (APT-03, ISSUE 16 AND LATER ISSUES) . . . . .	14
A. CSDMP Program Description . . . . .	14
B. CSDMP Operating Instructions . . . . .	14
9. XCMP PROGRAM ABORT MESSAGES . . . . .	14
10. FINAL PROCEDURE . . . . .	16
11. ABBREVIATIONS . . . . .	16

**1. GENERAL**

**1.01** This section gives procedures for using the program store copy, compare, and dump program (XCMP). The XCMP program is covered in PR-1A533 and is included in the MOD 5 auxiliary test program APT-02 (Issue 20 and later) and APT-03 (Issue 19 and earlier) for the 2-wire No. 1 Electronic Switching System (ESS).

*Note:* Issue 19 was the last issue of APT-03 which contained XCMP. Beginning with Issue 20, XCMP was made a part of APT-02.

**1.02** This section has been reissued for the following reasons:

- (a) To include APT-02 references.
- (b) To remove all information pertaining to APT-03, Issue 15 and earlier.
- (c) To make minor sequence changes and minor editorial changes.

Since this is a general revision, arrows ordinarily used to indicate changes have been omitted.

**1.03** Abbreviations used in this section are listed in Part 11.

**1.04** The program store copy, compare, and dump program (XCMP) is used primarily to relocate a block of program store (PS) data via the recent change (RC) area of call store. The need for such an activity often arises during retrofit and growth activities. Other features are included that make XCMP an alternative to the use of the T-PATTERN input message sometimes used for verification of the contents of PS. XCMP consists of the following four programs.

**NOTICE**

Not for use or disclosure outside the  
Bell System except under written agreement

(1) **Copy Program (CPY)**—This program can copy either the right or left half of each word of a PS block (of variable length) into a new PS location via the RC area. The new PS cards are written using facilities normally used to update RCs. (Refer to Section 231-104-301.) The CPY program also has the ability to move 23 bit translators from left half to right half and vice versa.

**Caution:** *The relocation of memory should be done with care. Therefore, a full understanding of the consequence of using XCMP must be recognized prior to its use. It can be valuable in situations such as retrofit and growth when a block of translation data must grow beyond the space allocated to it. An obvious complication is the fact that any memory locations containing pointers into the relocated block must be updated to reflect the new addresses. Thus, this is a powerful capability which may eliminate the need for a costly and time-consuming translation repack, but which must be used cautiously.*

(2) **Compare Program (CMP)**—This program compares the contents of two blocks of PS data. The user may select right half, left half, or full word comparison. The words that do not match can be listed and a failure-to-match count is kept and printed. This program may be used to verify a relocation performed by the copy program.

(3) **Program Store Dump Program (PSDMP)**—The PSDMP program will print out program store words on the TTY. An unlimited number of words (for practical purposes) can be printed with one set of input messages. The PSDMP program also can dump a 23-bit left-half translator in right-half format.

**Note:** The above program will operate only on PS locations. Additional restrictions are given in the program operating instructions.

(4) **Call Store Dump Program (CSDMP)**—The CSDMP program will print up to 8192 words (octal 20000) of call store data.

## 2. APPARATUS AND DOCUMENTATION

2.01 One J1A064AA, List 2 (APT-02) or List 3 (APT-03), auxiliary program package is required to run the routine described in this section. To select an APT-02 or APT-03 issue that is compatible with a specific generic program, refer to Section 8 of PA-1A500.

2.02 The following documentation is required.

- Section 231-147-301 or PA-1A500.
- PR-1A533 should be furnished with the auxiliary program package.

## 3. PRELIMINARY PROCEDURE

3.01 PS module 05 or 15 must be loaded with the proper auxiliary program package; therefore, before running XCMP, perform the preliminary procedures in PA-1A500 or Section 231-147-301 for inserting an auxiliary test module, configuring auxiliary test programs into service, and using the LIB-EDIT- message.

## 4. PROGRAM STORE BUS CONFIGURATION PROCEDURE

4.01 The CPY, CMP, and PSDMP programs require a forced PS bus configuration while each program is running. After the programs have run, all buses must be configured back into service. The procedures for configuring are given in the following paragraphs and referred to in each test as needed.

### Forcing a PS Bus Configuration

4.02 Select a PS bus (either 0 or 1) to be used and force a configuration as follows:

- (1) Put the other bus (the bus *not* to be used) in the request inhibit mode by placing the BUS ISOLATION CONTROL in the appropriate REQUEST position. For example, if BUS 0 is to be used, place the BUS ISOLATION CONTROL in the REQUEST position for BUS 1. System response should be

MA10 OK TO RMV PWR ááaa bbb.

ááaa = Unit type placed in request inhibit mode

bbb = Member number of unit type.

**Note:** If the desired bus cannot be removed from service because the other bus is in trouble, the trouble must be cleared before running this program. The procedures given in Section 231-148-301 may be helpful in clearing the problem.

- (2) Configure all PSs in the office onto the desired bus by typing

PS-CØNFIG-BaaSbbbbbbbbbbbb.

aa = 01 if BUS 0 is to be used, 10 if BUS 1 is to be used.

b...b = 0 or 1. For each PS in the office, type a 1 in the associated bit position. (The rightmost b is for PS 0, and the leftmost b is for PS 12.) For each PS not in the office, type a 0 in the associated bit position.

**Caution:** *The PS-CONFIG message must be used with care because the system sets all indicators in accordance with the requested configuration but does not execute the total request if it results in insanity. An E-level interrupt will be generated if an attempt is made to configure into service a unit that has had power removed manually. In case of an erroneous input, type in PS-CONFIG-B 11 S 111111111111. Then type in the desired message again.*

System response should be MA07 PS SW ØK. Verify at the master control center (MCC) that all PSs are in service and that the desired bus is in service.

**Configuring All Buses and Program Stores Back Into Service**

- 4.03 To configure all buses and program stores back into service:

- (1) Type in

PS-CØNFIG-B11Sbbbbbbbbbbbb.

b...b = 0 or 1. For each PS in the office, type a 1 in the associated bit position. (The rightmost b is for PS 0, and the leftmost b is for PS 12.) For each PS not in the office, type a 0 in the associated bit position.

**Observe caution in paragraph 4.02.** System response should be MA07 PS SW ØK. Verify at the MCC that all buses and all stores are in service.

- (2) At the MCC alarm, display, and control panel, rotate the BUS ISOLATION CONTROL for PS buses back to the NORMAL position.

**5. PROGRAM STORE COPY (CPY)**

**A. CPY Program Description**

5.01 The CPY program can be used to move blocks of program store information from one address range to another as follows.

- (1) Left-half data can be moved to the left half or right half of a new PS block.
- (2) Right-half data can be moved to the right half or left half of a new PS block.
- (3) Twenty-three bit translator data can be moved to the left half or right half of a new PS block, or vice versa.

The program reads either the left or right half of a word in a PS block and moves the data into a primary recent change register in call store. The address (TAG) of the RC register will be the address in the PS area where the data is to be placed. After the CPY program ends, the data is moved from the primary RC area to the new PS block by card writing as is normally done to update recent changes (see Section 231-104-301).

5.02 The data in the first word of the old PS block goes into the first word of the new block, and so on, as the program continues. If the new block length specified is greater than the old block length, the extra words in the new block will be filled by the data word specified in the input message.

5.03 Moving data with CPY is illustrated in the first three examples in Fig. 1. Associated

## SECTION 231-153-301

input messages for each example are explained briefly. There are eight CPY input messages. All except the second, which is optional, must be used to run the program. The messages are explained fully in paragraph 5.09.

**5.04** If an error is found in a PS word, the error will be corrected to agree with the Hamming code, the corrected data will be moved to the new block in recent change area, and the program will print a message (paragraph 5.11) giving the address of that word.

**5.05** An error check is built in so that data cannot be moved to or from the system test words. If an attempt is made to copy over test words, an abort will occur before any copy is performed.

**5.06** Two special options are provided with the CPY program.

(1) Auxiliary block option: If the size of an auxiliary block changes during a move, the user can change the "WRDN" of the auxiliary block to the new size by setting the auxiliary block option on. Setting this option on means that

(a) If the auxiliary block is less than 32 words, the first word upper 7 bits is changed to the new size.

(b) If the auxiliary block is 32 or more words, the first word lower bits are changed to the new size. See Fig. 1D.

(2) Left-half 23-bit translator option:

(a) If a 23-bit translator is moved from left to right or right to left, the option must be set by using the LIB-SET-1,20. message.

(b) If the move is for 14 bits from right to left or left to right, the option should **not** be set. See Fig. 1E.

### B. CPY Operating Instructions

**5.07** Prior to running CPY, it will be necessary to clear the primary RC area of all permanent RCs since it is desirable to have as much vacant primary RC area as possible. This is done by updating PS memory using the memory card writer as described in Section 231-104-301. Once cards

have been written, be sure that the RC area has been updated and consolidated using the RC-UPDATE message, also described in Section 231-104-301.

**5.08** Force a PS bus configuration using the procedure in paragraph 4.02.

**5.09** To run CPY, the LIB-START and six LIB-OCT messages must be typed in as instructed below. ***If any are omitted or typed incorrectly, the program may abort or not begin.*** The LIB-SET message is used only if a 23-bit translator is to be moved from left half to right half, or vice versa.

(1) To run CPY, first type

LIB-START-15,01,b,06.

b = 1 if auxiliary block option is to be used

= 0 if auxiliary block option is **not** to be used.

**Note:** For explanation of auxiliary block option, see paragraph 5.06.

The system response to the previous message and all of the following input messages should be "OK".

(2) If a 23-bit translator is to be moved from left half to right half, or vice versa, type the following message; otherwise, go directly to (3).

LIB-SET-1,20.

(3) Then type

LIB-OCT-axxxxxxx.

a = 0 if old block to be copied is in H-half of PS

= 1 if old block to be copied is in G-half of PS

xxxxxxx = Start address of old PS block.

(4) Then type

LIB-OCT-ssss,bbbb.

**1A Example of Move**

	OLD PS BLOCK	NEW PS BLOCK	INPUT MESSAGES	EXPLANATION
aaaaaaa- H-half block	5 words data	5 words data  2 words fill	-bbbbbbb LIB-START-15,01,0,06.  G-half block LIB-OCT-0aaaaaaa. LIB-OCT-0000,0005. LIB-OCT-1bbbbbbb. LIB-OCT-0002,0005. LIB-OCT-0001,0001. LIB-OCT-ffffff.	Program start message.  Start address of old PS block (0 = H-half). 5 words in old PS block to be copied. Start address of new PS block (1 = G-half). 5 words copied into new PS block, fill 2 words. 1 old and 1 new block of data (Note 2). Data word used for fill. May be 00000000.

**1B Example of Move**

	OLD PS BLOCK	NEW PS BLOCK	INPUT MESSAGES	EXPLANATION
aaaaaaa- G-half block	12 words data  12 words data  12 words data	12 words data  1 wd fill  12 words data  1 wd fill  12 words data  1 wd fill  13 words fill  13 words fill	-bbbbbbb LIB-START-15,01,0,06.  G-half block LIB-OCT-1aaaaaaa. LIB-OCT-0000,0012. LIB-OCT-1bbbbbbb. LIB-OCT-0001,0012. LIB-OCT-0003,0005. LIB-OCT-ffffff.	Program start message.  Start address of old PS block (1 = G-half). 12 words in old PS block to be copied. Start address of new PS block (1 = G-half). 12 words copied into new PS block, fill 1 word. 3 old and 5 new blocks of data (Note 3). Data word used for fill. May be 00000000.

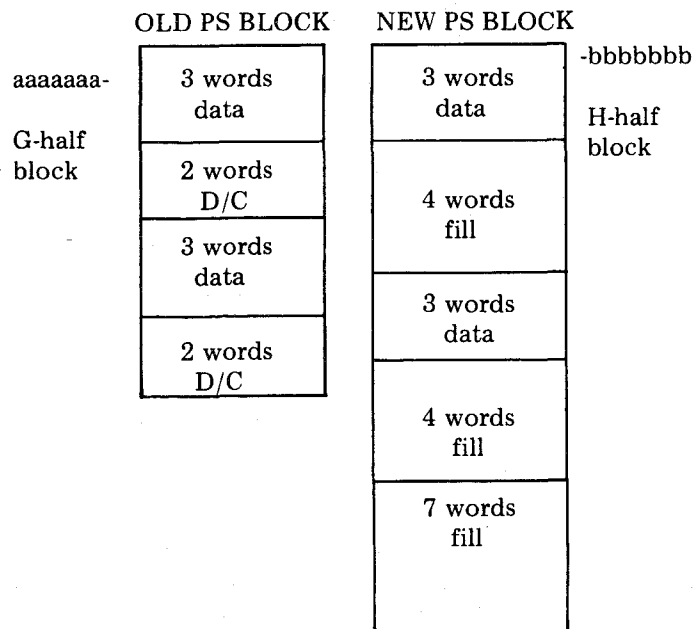
*Note 1:* Numbers used in LIB-OCT messages are octal.

*Note 2:* The explanation for this message applies to APT-03, Issues 18 and 19 and APT-02, Issue 20 and later. For APT-03 Issues 16 and 17, the variable typed first is for the new block and the variable typed last is for the old block.

*Note 3:* The LIB-OCT message shown is for APT-03, Issues 18 and 19 and APT-02, Issue 20 and later. For APT-03 Issues 16 and 17, the meanings of the variables are reversed so that the message for 3 old and 5 new blocks of data would be LIB-OCT-0005, 0003.

**Fig. 1—Examples of CPY Data Moves (Sheet 1)**

1C Example of Move



INPUT MESSAGES

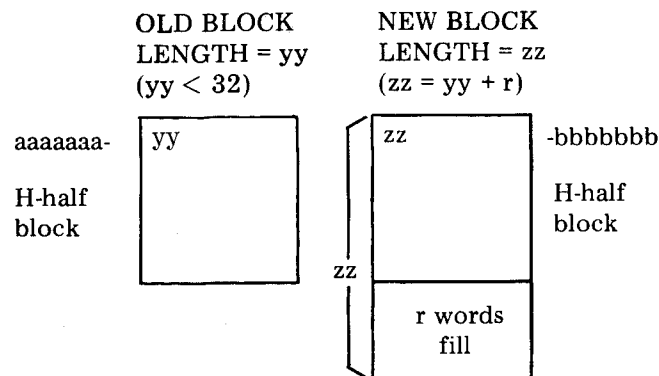
LIB-START-15,01,0,06.  
 LIB-OCT-1aaaaaaa.  
 LIB-OCT-0002,0003.  
 LIB-OCT-0bbbbbbb.  
 LIB-OCT-0004,0003.  
 LIB-OCT-0002, 0003.  
 LIB-OCT-ffffff.

EXPLANATION

Program start message.  
 Start address of old PS block (1 = G-half).  
 Copy first 3 words in old block but not last 2.  
 Start address of new PS block (0 = H-half).  
 3 words copied into new PS block, fill 4 words.  
 2 old and 3 new blocks of data (Note 4).  
 Data word used for fill. May be 00000000.

*Note 4:* The LIB-OCT message shown is for APT-03, Issues 18 and 19 and APT-02, Issue 20 and later. For APT-03 Issues 16 and 17, the variables are reversed so that the message for 2 old and 3 new blocks of data would be LIB-OCT-0003, 0002.

1D Example of Moving Auxiliary Block and Changing Word Number Word



INPUT MESSAGES

LIB-START-15,01,1,06.  
 LIB-OCT-0aaaaaaa.  
 LIB-OCT-0000,00yy.  
 LIB-OCT-0bbbbbbb.  
 LIB-OCT-000r,00yy.  
 LIB-OCT-0001,0001.  
 LIB-OCT-ffffff.

EXPLANATION

Program start message.  
 Start address of old PS block (0 = H-half).  
 yy words in old block to be copied.  
 Start address of new PS block (0 = H-half).  
 yy words copied into new block, fill r words.  
 1 old and 1 new block of data (Note 2).  
 Data word used for fill. May be 00000000.

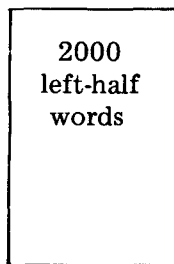
Fig. 1—Examples of CPY Data Moves (Sheet 2)

1E Example of Moving Translator Left Half to Right Half

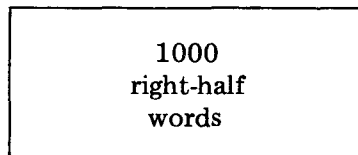
OLD SUBTRANSLATOR  
23 BIT LEFT HALF

NEW SUBTRANSLATOR  
23 BIT RIGHT HALF

aaaaaaa-



-bbbbbbb



INPUT MESSAGES

EXPLANATIONS

LIB-START-15,01,0,06.  
LIB-SET-1, 20.  
LIB-OCT-0aaaaaaa.  
LIB-OCT-0000,0002.  
LIB-OCT-0bbbbbbb.  
LIB-OCT-0000,0001.  
LIB-OCT-1000,1000.  
LIB-OCT-00000000.

Program start message.  
Move 23 bit translator from left half to right half.  
Start address of old subtranslator.  
2 left-half words per block in old subtranslator to be copied (Note 5).  
Start address of new subtranslator.  
1 right-half word per block in new subtranslator (Note 6).  
1000 old and 1000 new blocks of data (Note 2).  
Don't care.

*Note 5:* Use LIB-OCT-0000,0001. for right-half to left-half move.

*Note 6:* Use LIB-OCT-0000,0002. for right-half to left-half move.

Fig. 1—Examples of CPY Data Moves (Sheet 3)

ssss = Octal number of words not to be copied from the bottom of each old PS block (maximum = octal 3777 words)

bbbb = Octal number of words to be copied from the top of each old PS block (maximum = octal 7777 words).

(5) Then type

LIB-OCT-axxxxxxx.

a = 0 if new block is in H-half of PS

= 1 if new block is in G-half of PS

xxxxxxx = Start address of new PS block.

(6) Then type

LIB-OCT-ffff,cccc.

ffff = Octal number of words to be loaded with fill at the bottom of each new PS block (maximum = octal 3777)

cccc = Octal number of words into which data will be copied at the top of each new PS block (maximum = octal 7777).

(7) Then type

LIB-OCT-nnnnyyyy.

For APT-02, Issue 20 and later, and for APT-03, Issue 18 and 19.

nnnn = Number of old blocks of data (maximum = octal 3777)

yyyy = Number of new blocks of data (maximum = octal 7777)

For Issues 16 and 17 of APT-03,

nnnn = Number of new blocks of data (maximum = octal 7777)

yyyy = Number of old blocks of data (maximum = octal 3777)

(8) Then type

LIB-OCT-xxxxxxx.

xxxxxxx = Data word to fill remaining words in new block if length of new block is greater than length of old block (may be 00000000).

PROGRAM DISPLAY lamp 23 should light and remain lighted while the program is running.

5.10 Before the program begins to copy, a TTY message will be printed to verify the input data. Its format is as follows:

LIB15	CPY	VER
OLD ADDR		Axxxxxxx
LENGTH		xxxxx
NEW ADDR		Axxxxxxx
LENGTH		xxxxx
DAT FILL		xxxxxxxxxxxxxxxxxxxxxxxxxxxx

(a) If the data in the printout above is **not** correct, type in

LIB-SET-1,22.

Begin again with paragraph 5.09

(b) If the data **is** correct, type in the following message to start the program copying PS data.

LIB-SET-1,10.

**Note:** The program will not begin to copy data until the LIB-SET message is typed in.



5.11 It is possible that the following message **may** be printed out while the PS CPY is running:

```
LIB14 PS ERRØR ** ADDR: Axxxxxxx
```

xxxxxxx = Address of PS word in which the program found an error.

This is a normal CPY message. It indicates the program has found an error in a PS word. The old data is corrected to agree with Hamming code and copied into the new address anyway. The data should be checked as an error correction occurs. See paragraph 5.05 for restrictions on PS test words.

5.12 The program will now begin to copy the PS information. Eventually, a message will be printed out with a format the same as the LIB15 CPY VER message explained in paragraph 5.10 except that the variable aaa in the LIB15 CPY aaa heading will be one of the following:

aaa = RCA—If the heading gives this message, it indicates that another program has reserved space in the RC area during the running of CPY. CPY had ended at the point indicated by the data in the message and, after card writing the data into recent change, the program should be started again (paragraph 5.09) with the same input data as before.

= RCF—If the heading gives this message, it means that either the RC primary or the RC auxiliary area has been filled before the entire block of data has been copied. Again, CPY has ended at the point indicated by the output data of this message. The user should now update the RC area by card writing as instructed in Section 231-104-301. After this has been done, the program should be started again (paragraph 5.09) using the same input data as before.

= END—If the heading gives this message, it indicates that the program has completed moving the entire data block into the RC area. Both the new length and old length should

equal 00000. After updating the RC area as instructed in Section 231-104-301, the copying of the data will be complete.

5.13 After transferring the copied PS data from the primary RC area to the new PS block via card writing (Section 231-104-301), if it is not immediately desirable to run the compare routine (Part 6), the PS buses must be taken out of the forced configuration imposed in paragraph 4.02. This is done by following the procedure in paragraph 4.03.

5.14 If it is desired to run the compare routine or the dump routine immediately after card writing the copied PS data, leave the buses in the configuration forced in paragraph 4.02. The desired program may then be run without again forcing a bus configuration.

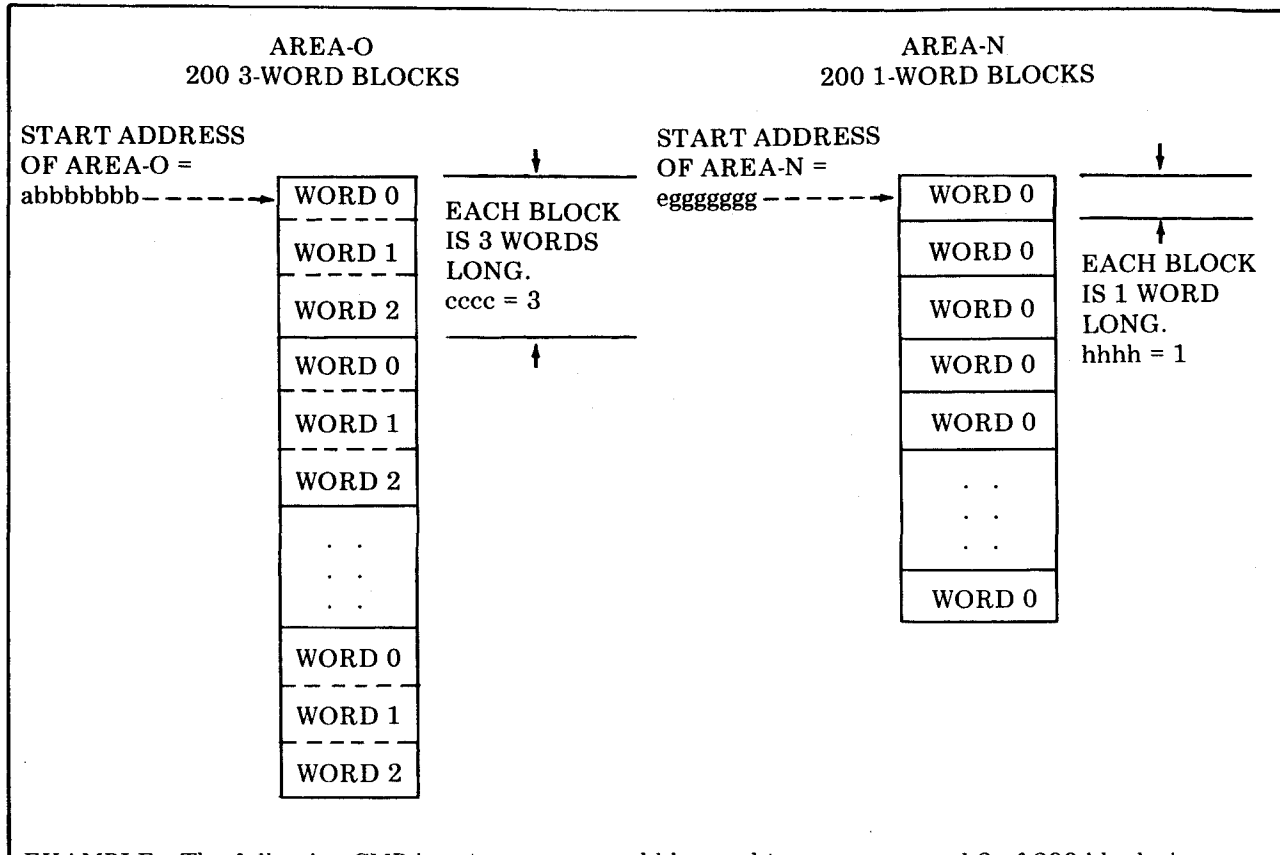
## 6. PROGRAM STORE COMPARE (CMP)

### A. CMP Program Description

6.01 The CMP program can compare the data for a selected number of words at the beginnings of consecutive blocks in program store with the same number of selected words in the beginnings of another set of consecutive blocks in program store. See Fig. 2. For example, assume two sets of consecutive blocks, ABC and GHL. If the user selects the first two words for comparison, the program compares data for the first two words of A with the first two words of G, the first two words of B with the first two words of H, and the first two words of C with the first two words of L. Addresses and LIB-SET messages can be selected to compare either the entire program store word or to compare the 23-bit left half of the PS word with another left or right half.

6.02 If requested, the program will print the addresses and the contents of the words whose data fails to match.

6.03 If either of the two words being compared contains an error, the program will print the address of the error (one or both) and count this as a mismatch (even if the data in the two are the same). There will not be a printout of the addresses and data even if requested.



EXAMPLE: The following CMP input messages could be used to compare word 2 of 200 blocks in area-O with word 0 of 200 blocks in area-N. (dddd = 2, iii = 0, nnnn = 200)

CMP GENERAL MESSAGES	CMP EXAMPLE MESSAGES	EXPLANATION
LIB-START-15,02,0,06.	LIB-START-15,02,0,06.	Start program.
LIB-OCT-abbbbbbb.	LIB-OCT-abbbbbbb.	Selects start address in area-O. (a = 0 for H and a = 1 for G half of PS)
LIB-OCT-ccccddd.	LIB-OCT-00030002.	0003 words in each area-O block. Use word 0002 for comparison.
LIB-OCT-egggggg.	LIB-OCT-egggggg.	Selects start address in area-N. (e = 0 for H and e = 1 for G half of PS)
LIB-OCT-hhhhiiii.	LIB-OCT-00010000.	0001 word in each area-N block. Use word 0000 for comparison.
LIB-OCT-000nnnn.	LIB-OCT-00000200.	200 (octal) words to be compared. (Same as number of blocks)
LIB-OCT-00000a0b.	LIB-OCT-00000a0b.	a = 0 to print count of mismatches. a = 1 to print each mismatch. b = 0 to compare right half of words. b = 1 to compare left half of words. b = 2 to compare entire PS word.

Fig. 2—CMP Example

**6.04** Upon completion, the routine will print a message giving the total number of pairs of words whose data mismatched. This printout will be given even if the optional printout which can be requested in paragraph 6.06(1) is requested.

**Note:** If the approximate number of mismatches in a large program store block is not known, run CMP first without the individual mismatch printout to get the total number of mismatches. If the total number is large, the individual printout may require excessive TTY time and be undesirable.

### B. CMP Operating Instructions

**6.05** Select a PS bus and force a configuration as explained in paragraph 4.02 unless a bus configuration has been forced prior to running CMP (as in paragraph 4.02) and has not been normalized (as in paragraph 4.03).

**6.06** To run CMP, the LIB-START and five LIB-OCT messages must be typed in as instructed below. If any are omitted or incorrectly typed, the program may abort or not begin. The LIB-SET messages are optional.

(1) Type in

LIB-START-15,02,b,05.

b = 1 to print each individual mismatch  
= 0 to print only total number of mismatches.

The system response to this and the following input messages should be OK.

(2) If the entire PS word (both right and left halves) is to be compared, type in

LIB-SET-1,09.

**Note:** If LIB-SET-1,09. is not typed in, only right- or left-half words will be compared when right- or left-half addresses are used in the first and third LIB-OCT messages.

(3) If a 23-bit left-half translator (14- and 9-bit format) is to be compared with a right-half translator, type in

LIB-SET-1,20.

(4) Type in

LIB-OCT-abbabbbb.

a = 0 if area-O is in H-half of PS

= 1 if area-O is in G-half of PS.

bbabbbb = Start address of one of the areas to be compared. To be called area-O.

(5) Type in

LIB-OCT-ccccdddd.

cccc = Number of words to skip at end of each block in area-O (maximum = octal 3777 words)

dddd = Number of words to compare at start of each block in area-O (maximum = octal 7777 words).

(6) Type in

LIB-OCT-eggggggg.

e = 0 if area-N is in H-half of PS

= 1 if area-N is in G-half of PS

ggggggg = Start address of the other area to be compared. To be called area-N.

(7) Type in

LIB-OCT-hhhhiiii.

hhhh = Number of words to skip at end of each block in area-N (maximum = octal 3777 words)

iiii = Number of words to compare at start of each block in area-N (maximum = octal 7777 words).

(8) Then type

LIB-OCT-000nnnnn.

**SECTION 231-153-301**

nnnnn = Octal number of blocks (words) to be compared (maximum = octal 7777 words).

PROGRAM DISPLAY lamp 23 should light and remain lighted while the program is running.

**6.07** When the program begins to run, the following message will be printed on the TTY:

LIB15	CMP	FAIL		
	ADDR-O	DATA	ADDR-N	DATA

If the user has requested a printout of each mismatch, the printout will be as follows:

LIB15	CMP	FAIL		
	ADDR-O	DATA	ADDR-N	DATA
LIB15	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx
LIB15	xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx

where each horizontal line is a mismatch printout. The ADDR-O column gives the octal addresses of each failing word in one of the blocks to be compared, and the ADDR-N column gives the octal addresses of each failing word in the other block to be compared to block O. The adjacent DATA columns give the octal data contained in the addresses that mismatch. The first digit of the ADDR will indicate if the word is a left-half or right-half word (ie, 1, 2, 3 = right-half word; 5, 6, 7 = left-half word). Also, if the whole word is to be compared and both halves mismatch, two lines will result.

**6.08** It is possible that the following message *may* be printed out while the CMP is running:

LIB14 PS ERRØR \*\* ADDR: Axxxxxxx

This is a normal message for CMP and indicates the program has found an error in a PS word whose address is xxxxxxx.

**6.09** At the end of the routine, a message is printed out as follows:

```
LIB15 CMP
TØTAL NØ. ØF CØMPARES = xxxxx
TØTAL NØ. ØF FAILS = nnnnn
```

nnnnn = The octal number of words compared. (Should equal nnnnn typed in.)

xxxxx = The octal number of mismatches found. (One for each half word mismatch.)

**6.10** Upon completion, the program will terminate and print

LIB15 UN15 END

**6.11** If it is not immediately desirable to run the dump routine ((Part 7) after running the CMP, the PS buses must be taken out of the forced configuration. This is done using procedures given in paragraph 4.03.

**6.12** If it is desired to run the dump routine immediately upon completion of the CMP, leave the buses in the forced configuration. The dump routine may then be run without having to force a configuration.

**7. PROGRAM STORE DUMP (PSDMP)**

**A. PSDMP Program Description**

**7.01** The PSDMP program can print out program store words on the TTY. For each word, both the right and left half are printed with the address. The program will start printing with the word at the start address indicated and continue until the specified length is reached. The maximum length that may be specified with one set of input messages is an unlimited number of words (for practical purposes) that exists after a start address can be printed.

**7.02** Left-half 23-bit translators may be dumped in right-half format.

**7.03** If any word contains an error, it will still be dumped and a message will be printed giving its address.

**7.04** If a large number of words is to be dumped requiring lengthy TTY time, the user may prefer not to use the local maintenance TTY. Another TTY channel can be selected for PSDMP input and output messages by a LIB-TCHAN message. This message can be typed on any TTY defined in the system. The call store word used to store the TTY channel selected by the LIB-TCHAN message is not zeroed by any generic or auxiliary program. The word will contain the previously selected channel number until changed by another LIB-TCHAN message. If the TTY channel selected in the last LIB-TCHAN message is not known, the message should be typed to avoid having the PSDMP output on the wrong TTY channel.

#### B. PSDMP Operating Instructions

**7.05** Select a PS bus and force a configuration as explained in paragraph 4.02 unless a bus configuration has been forced prior to running CMP (as in paragraph 4.02) and has not been normalized (as in paragraph 4.03).

**7.06** If the TTY channel selected in the last LIB-TCHAN message is unknown or not to be used for this dump, select a TTY channel to be used for PSDMP input and output messages by typing

LIB-TCHAN-aa.

aa = 00 for local maintenance

= 01 for supplementary dial service  
(second traffic)

= 02 for administrative dial service  
(first traffic)

= 03 for ALIT and LTD

= 04 for service order

= 05 for remote maintenance

= 06 for calling line identification

= 07 for network management

= 12 for fourth trunk maintenance

= 13 for first trunk maintenance

= 14 for second trunk maintenance

= 15 for third trunk maintenance.

**Note:** "aa" is the TTY channel number as given in the TTY-MLTCHAN input message in IM-1A001.

If the LIB-TCHAN message is not typed, PSDMP output will go to the same TTY channel selected in the last LIB-TCHAN message. For 1E3 generics, the TTY channel receiving the PSDMP output will be identified by a LIB00 OUTPUT ON TTY CHANNEL aa output message on the local maintenance TTY.

**7.07** To run PSDMP, the following LIB-START and two LIB-OCT messages must be typed in as instructed below. If any are omitted or typed incorrectly, the program may abort or not begin. The LIB-SET message is optional.

(1) Type in

LIB-START-15,03,0,02.

The system response to this and the following input messages should be ØK.

(2) If left-half 23-bit translators are to be dumped in right-half format, type

LIB-SET-1,20.

(3) Type in

LIB-ØCT-axxxxxxx.

a = 0 if block to be dumped is in  
H-half of PS

= 1 if block to be dumped is in  
G-half of PS

xxxxxxx = Octal start address of block  
to be dumped.

(4) Type the following LIB-OCT message.

LIB-OCT-00xxxxxx.

xxxxxx = Octal length of block to be dumped. May be any number of PS words up to octal 777777 (decimal 262,143) that exists after the start address.

PROGRAM DISPLAY lamp 23 should light and remain lighted while program is running.

**7.08** When the program begins to run, a LIB15 PSDMP message will be printed out followed by a dump of PS words in blocks of 32 octal words as requested. The format of the message and dump is shown in Fig. 3.

**7.09** It is possible that the following message may be printed out while the PSDMP is running:

LIB14 PS ERRØR \*\* ADDR: Axxxxxxx.

This is a normal message for PSDMP and indicates the program has found an error in a PS word whose address is xxxxxxxx. The message will precede the line in which the word is dumped.

**7.10** Upon completion, the program will terminate and print

LIB15 UN15 END

**7.11** If this is the last routine to be run using program stores, the PS buses must be taken out of the forced configuration. This is done using procedures given in paragraph 4.03.

## **8. CALL STORE DUMP (CSDMP)**

### **A. CSDMP Program Description**

**8.01** The CSDMP program can be used to print up to 8192 (octal 20000) call store words on a TTY.

### **B. CSDMP Operating Instructions**

**8.02** If a large number of words is to be dumped, the LIB-TCHAN message may be used. See paragraphs 7.04 and 7.06 or PR-1A503 (XLCP) for format.

**8.03** To run the CSDMP program, all of the following input messages must be typed in

as instructed below. If any are left out or incorrectly typed, the program may abort or not begin.

(1) Type in

LIB-START-15,00,0,02.

The system response to this and the following input messages is OK.

(2) Type in

LIB-OCT-00xxxxxx.

xxxxxx = Start address of dump in call store.

(3) Type in

LIB-OCT-000xxxxxx.

xxxxxx = Octal number of call store locations to be dumped. (Maximum number of words that can be dumped is octal 20000.)

This program will start to print on or before the address specified and continue until the specified length is reached. The output message format for CSDMP is shown in Fig. 4.

PROGRAM DISPLAY lamp 23 should light and remain lighted while the program is running.

## **9. XCMP PROGRAM ABORT MESSAGES**

**9.01** If it is necessary to abort the program for any reason, type

LIB-SET-1,22.

The program will abort and print

LIB 15 UN15 ABTD

**9.02** The program also may be stopped by typing

LIB-START-0000000.

System response should be AB.

**9.03** The system may print

LIB XCMP RELO ABT ADDR aaaaa

The output message format for PSDMP is as follows:

For APT-02, Issue 20 and later and for APT-03, Issues 18 and 19:

**LIB15 PS DUMP**

```

xxxxxxx yyyyy zzzzzzzz yyyyy zzzzzzzz yyyyy zzzzzzzz yyyyy zzzzzzzz
xxxxxxx yyyyy zzzzzzzz yyyyy zzzzzzzz yyyyy zzzzzzzz yyyyy zzzzzzzz
(etc.)

```

For APT-03, Issues 16 and 17

```

      LIB15          DMP
ADDR   LIB15      LHW      RHW
xxxxxxx LIB15      yyyyy   zzzzzzzz   yyyyy   zzzzzzzz   yyyyy   zzzzzzzz   yyyyy   zzzzzzzz
      LIB15      yyyyy   zzzzzzzz   yyyyy   zzzzzzzz   yyyyy   zzzzzzzz   yyyyy   zzzzzzzz
(etc.)

```

*Note 1:* Each line of the dump is broken down as follows:

Address of first word on the line	First PS Word		Second PS Word		Third PS Word		Fourth PS Word	
	Left-Half Data	Right-Half Data	Left-Half Data	Right-Half Data	Left-Half Data	Right-Half Data	Left-Half Data	Right-Half Data
xxxxxxx	yyyyyy	zzzzzzzz	yyyyyy	zzzzzzzz	yyyyyy	zzzzzzzz	yyyyyy	zzzzzzzz

*Note 2:* The contents of four PS words are dumped per line. The address on the left is the octal address of the first word on the line.

yyyyyy = Left-Half PS Data (octal)  
zzzzzzzz = Right-Half PS Data (octal)

*Note 3:* Because of the message format, a few more words may be dumped than requested.

*Note 4:* If a left-half 23-bit translator is dumped in right-half format (LIB-SET-1,20. option is used), the contents of 8 left-half words are dumped per line, and the address on the left is the even address of the first left-half word.

yyyyyy = 00000 (don't care)  
zzzzzzzz = Combined even and odd left-half words.

**Fig. 3—Program Store Dump Printout**

The output message format for CSDMP is as follows:

For APT-02, Issue 20 and later and for APT-03, Issues 18 and 19:

```

LIB15 CS DUMP xxxxxxxx
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy
yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy yyyyyyyy

```

(See Note.)

For APT-03, Issues 16 and 17:

```

LIB15 CS DUMP xxxxxxxx
yyyyyyyy
yyyyyyyy
yyyyyyyy
yyyyyyyy
yyyyyyyy
yyyyyyyy
yyyyyyyy
yyyyyyyy

```

(See Note.)

xxxxxxx = Start address of dump  
yyyyyyy = Word of call store data

*Note:* Only one block is shown. Additional blocks will be printed until the number of words requested has been dumped.

Fig. 4—Call Store Dump Printout

This message means that XCMP has aborted. An explanation of the reason for the abort is given by the number aaaa in the RELOCATABLE ABORT MESSAGES section of PR-1A533.

**10. FINAL PROCEDURE**

**10.01** After all routines in XCMP desired to be run have been finished, configure the MOD 5 containing XCMP out of service and replace it with the original PS module 05 using procedures covered in PA-1A500 or Section 231-147-301.

**11. ABBREVIATIONS**

**11.01** The following abbreviations are used in this section:

ALIT	Automatic Line Insulation Test
APT	Auxiliary Program—Telephone Company
CMP	Program Store Compare Program



CSDMP	Call Store Dump Program	RC	Recent Change
CPY	Program Store Copy Program	TAG	Address of Primary Translation Word in Subtranslator
DMP	Program Store Dump Program	TTY	Teletypewriter
ESS	Electronic Switching System		
LTD	Local Test Desk	WRDN	Word Number (Size of Auxiliary Block)
MCC	Master Control Center		
PS	Program Store	XCMP	Program Store Copy, Compare, and Dump Procedures and Call Store Dump Procedure
PSDMP	Program Store Dump Program		