GH20-0858-1

Type III Class A Program

IBM

Control Program-67/Cambridge Monitor System (CP-67/CMS) Version 3.2
Program Number 360D-05.2.005
Hardware Maintainability Guide

The CP-67/CMS Hardware Maintainability Guide contains the following information:

- Conventions, instructions, and restrictions for running Customer Engineer Diagnostics under the system
- CP-67 Error-Recording Facility, including error-recording mechanism, conditions and devices involved in recording, and instructions for retrieval of I/O error, channel check, and machine check information
- Sample terminal sessions for running DME and FRIEND

Note: This document must be read in its entirety by customer engineers before they use the serviceability features of CP-67.

NOTICE

This Type III Program performs functions which may be fundamental to the operation and maintenance of a system. It has not been subjected to formal test by IBM.

Until program reclassification, IBM will provide

- Central Programming Service, including design error correction and automatic distribution of corrections
- FE Programming Service, including:
  (1) Design error verification
  (2) APAR documentation and submission
  (3) Application of Program Temporary Fixes or development of an emergency bypass when required.

IBM does not guarantee service results or represent or warrant that all errors will be corrected.

The user is expected to make the final evaluation as to the usefulness of this program in his own environment.

THE FOREGOING IS IN LIEU OF ALL WARRANTIES EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Second Edition (October 1971)

This edition, together with Technical Newsletters GN20-2497 dated January 24, 1972 and GN20-2620 dated May 15, 1973, applies to Version 3, Modification Level 2, of the Control Program-67/Cambridge Monitor System (360D-05.2.005) and to all subsequent versions and modifications until otherwise indicated in new editions or Technical Newsletters.

Information in this publication is subject to change. Therefore, be sure you have the latest edition and any pertinent Technical Newsletters.

Changes to text and illustrations are indicated by a vertical line to the left of the change.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form has been provided at the back of this publication for readers' comments. If this form has been removed, address comments to: IBM Corporation, VM/370 Publications, 24 New England Executive Park, Burlington, Massachusetts 01803.

PREFACE

This publication is a guide for the maintenance of the hardware using CP-67/CMS. The following publications are referenced in this manual:

CP-67/CMS User's Guide, GH20-0859
CP-67 Program Logic Manual, GY20-0590

Additional references are:

CP-67/CMS System Description Manual, GH20-0802
CP-67 Operator's Guide, GH20-0856
CP-67/CMS Installation Guide, GH20-0857
CMS Program Logic Manual, GY20-0591
CP-67: Operating Systems
in a Virtual Machine, GH20-1029

CONTENTS

FIGURES

RUNNING CE PROGRAMS UNDER CP-67

Control Program-67 is a multiprogramming, time-sharing system that runs on an IBM System/360 Model 67. Its function is to create virtual machines in which users can run the operating system of their choice concurrently with other users. The virtual machine created by CP looks to the user like a typical System/360 in the range from a Model 30 to a Model 67. Operating systems that run on the real 360 can run on the virtual 360 as long as they do not have timing dependencies or certain types of dynamically modified I/O CCW chains.

In a time-sharing system such as CP-67, a facility is needed for running diagnostic programs in order to allow testing of I/O units concurrently with the operation of other system tasks. This assists diagnosis, testing, and checkout of malfunctioning I/O units, while maintaining system operation.

The capability to run unit tests under control of CP-67 provides the flexibility necessary to determine the impact of malfunctioning devices. In most cases, it is possible to determine the problem at the device level and thereby take immediate corrective action.

Certain restrictions are imposed in order to run diagnostic programs under control of CP-67. These restrictions are as follows:

- Timing dependencies in a program cannot be guaranteed (for example, tape motion test).
- Certain dynamically modified CCW sequences cannot be translated by CP-67.
- The DIAGNOSE instruction cannot be executed.

Certain CE diagnostics contain these restrictions and therefore cause erroneous diagnostic messages. The DIAGNOSE restriction eliminates running almost all CPU, storage, and channel diagnostics, and some control unit tests. Although it is now possible to ipl DME from tape this does not ensure that every self-modifying ipl sequence will necessarily work correctly.

DEFINING A CUSTOMER ENGINEER'S VIRTUAL MACHINE

The CE's virtual machine must be defined in the System Directory. This can be accomplished by contacting the operations personnel at the system location.

The following are the necessary directory entries that the CE must have in order to create and save cataloged procedures (called EXEC files) and Diagnostic Monitors, and to run the error-recording edit and print programs.

```
Note 1   userid    USER password, accounta,C,30
                   CORE 256K
Note 2             UNIT 009,1052
Note 2             UNIT 00E,1403
Note 2             UNIT 00C,2540R
Note 2             UNIT 00D,2540P
Note 3             UNIT 0FF,TIMR
Note 4             UNIT 190,231X,CMS190,000,053,RDONLY
Note 5             UNIT 191,231X,CPDSK5,160,166
Note 6             UNIT 230,231X,CPDSK1,004,004,RDONLY
Note 7             *EOU*
```

Note 1. The CE should have privilege class C, as this prevents error recordings from being generated when diagnostics are being run, and authorizes him to clear the recording area.

Note 2. Standard CMS devices.

Note 3. Standard CMS pseudo-timer device.

Note 4. The CMS system disk. The X is 1 or 4 depending upon the type of disk drive on which the system resides. This entry may be different, depending on installation practices.

Note 5. Private disk storage area. The X is 1 or 4. This example gives 7 cylinders. In order to utilize the serviceability facilities in CP-67, it is necessary to allocate at least seven cylinders for a 2314, or twenty five cylinders for a 2311.

Note 6. The error-recording cylinder on the CP SYSRES device. The location of the error-recording cylinder is specified at CP SYSGEN time. Cylinder four (004) is the default value. The X is 1 or 4. CPDSK1 is the volid of the CP SYSRES device. Check with the installation to ensure the volid has not been changed.

Note 7. A tape drive has not been assigned since the operator will normally "attach" one to the user (if DME is to be run from tape.)

Note that after CMS has been IPL'ed during the customer engineer's initial terminal session, a FORMAT P ALL should be issued in order to have his P-disk area formatted. This should only have to be performed once unless the P-disk area is destroyed.

CONVENTIONS FOR RUNNING DIAGNOSTICS UNDER CP-67 CONTROL

When diagnostics are being run under CP-67, the 1052 request key has to be simulated to cause the diagnostic program to issue a read to the console. To simulate the 1052 request key, hit the ATTN key once, let the keyboard unlock, and then hit ATTN a second time. The second ATTN is reflected to the virtual machine, whose operating system then reads the console.

To simulate the cancel key on the 1052 console, the user may employ the CP Console Function SET LINEDIT ON. Once having issued this command, all input lines are edited by CP before being passed to the virtual machine. Both character-delete (@) and line delete (¢) symbols may be employed.

To stop the virtual machine at any time, hit ATTN once. To start it running again from the location at which it was stopped, type in BEGIN or B, followed by a carriage return.

To simulate the interrupt button on the System/360 console, there is a CP console function called EXTERNAL or EXT. To generate an external interrupt while a diagnostic is running, hit ATTN once and type in EXT, followed by a carriage return. The external interrupt is reflected to the virtual machine, and control is returned to the virtual machine's external interrupt processor.

To have a device that CP is using for paging or spooling attached to your machine, it is necessary that the device be offline at system IPL time. After CP is IPL'ed, the device may be switched online and attached to the CE's machine by the operator's ATTACH console function.

Note: Terminals which are equivalent to those explicitly supported may also function satisfactorily. The customer is responsible for establishing equivalency. IBM assumes no responsibility for the impact that any changes to the IBM-supplied products or programs may have on such terminals.

INSTRUCTIONS FOR RUNNING DIAGNOSTICS UNDER CP-67 CONTROL

RUNNING DME FROM TAPE

Previous versions of CP-67 were not able to properly execute the self-modifying CCW sequences encountered in the tape loader of the Diagnostic Monitors. The restriction no longer applies.

To ipl from tape it is necessary to have the system operator "attach" a tape drive to the virtual machine. This physical tape drive on which the Diagnostic Tape is mounted has a real hardware address. CP allows the operator to attach it to the configuration of the virtual machine at any unused address, referred to as the virtual address. Usually the real and virtual addresses will be the same. Should they differ, however, (for example, the tape is mounted on real address 0C4 and the message DEV 181 ATTACHED appears at the terminal indicating the virtual machine will operate as though a tape drive existed at address 181) the customer engineer must use the virtual address in all DME commands.

Communication with CP-67 is established by the user and is indicated with the CP-67 online message. (Instructions for operating each type of terminal supported are contained in the CP-67/CMS User's Guide.)

ATTN is depressed once causing the keyboard to unlock. This allows the user to log in with his userid. After the message ENTER PASSWORD is typed, the keyboard is again unlocked. The user then enters his password. Terminals having the print inhibit feature prevent printing of the password on the terminal. For TTY 33/35, three lines of characters are overprinted before the user enters his password. This keeps the security of the password.

The operator's log message (if any) is then printed. The keyboard unlocks and the user may ipl the tape drive (once the system operator has attached it to the virtual machine) by issuing the command IPL ccu, where ccu is the virtual address of the tape drive.

If an attempt is made to ipl before the drive is attached, the system action by CP will be equivalent to attempting to ipl a nonexistent address on the real hardware (the message will be typed.)

The copy of DME used should have UDT entries which at least include the devices in the virtual configuration noted above. Note the address of the console (which is usually 009) may differ. Secondly, the console and not the printer should be configured as the primary output device.

A sample terminal session showing the sequence of commands is shown in Figure 1.

```
 R|   cp-67 online    xd.65 qsyosu
 C|   login ce
 R|   ENTER PASSWORD
 C|
 R|   CP-67 WILL RUN TILL 17:00 AND FROM 18:00 to 24:00
 R|   READY AT 14.50.12 ON 12/03/69
 R|   CP
 |
 R|   DEV 181 ATTACHED <-Diagnostic tape mounted on this drive
 R|   DEV 236 ATTACHED <-Device to be tested
 R|   CP
 |
 C|   ipl 181
 |
M2|   WTE IPL DME REV LEVEL 9
 |
 R|   CP            <---ATTN key hit
 |              <---ATTN key hit
 I|   1/b
M2|   MODEL 65  256K          03FFFF
 R|   CP           <---ATTN key hit
 C|   ext
 |        .          .
 |        .          .
 |        .          .
M2|    T
M2|    WTE DME 9
 R|   CP       <---ATTN key hit
 C|   detach 236
 R|   DEV  236 DETACHED

 C|   detach 181
 R|   DEV 181 DETACHED

 C|   logout
 R|   CONNECT= 00.12.17 VIRTCPU= 003.11.66 TOTCPU= 005.23.51
 R|   LOGOUT AT 15.45.58 ON 12/03/69
```

C = Command issued by user
I = Instruction for diagnostic monitor issued by user
M2 = Message from diagnostic monitor
R = Response from CP-67

Figure 1. Sample Terminal Session for Running DME from Tape

BUILDING AN EXEC FILE FOR LOADING DME

Although diagnostic routines are usually tape resident they can be put on disk. This enables diagnostics to be run 1) without tying up a tape drive and 2) from remote locations without operator intervention.

The necessary EXEC files to assist in running diagnostics are included in the CE serviceability files. For an example of how these were created see Figure 1 and the explanation below; otherwise, skip to "Sample Command Stream to Run a Diagnostic".

Improved EXEC procedures, invoked by DIAGS EXEC, are now included in the serviceability files. However, since these are somewhat more complex, the description of how to build and use an EXEC procedure deals with a very simple case so as not to obscure the process.

DIAGS EXEC is described in a self-contained appendix, "CP-67 Online Testing User's Guide" and should be consulted for information on its use.

Communication with CP-67 is established by the user and is indicated with the CP-67 online message. (Instructions for operating each type of terminal supported are contained in the CP-67/CMS User's Guide.)

ATTN is depressed once causing the keyboard to unlock. This allows the user to log in with his userid. After the message ENTER PASSWORD is typed, the keyboard is again unlocked. The user then enters his password. Terminals having the print inhibit feature prevent printing of the password on the terminal. For TTY 33/35, three lines of characters are overprinted before the user enters his password. This keeps the security of the password.

The operator's log message (if any) is then printed. The keyboard unlocks and the user may enter the command IPL CMS. The next message is documentary in nature. Note that if the installation does not have a "saved" copy of CMS, substitute IPL 190 for IPL CMS.

The next twelve lines cause a file of the name of DME EXEC to be built. This is the series of commands that will be executed when the command DME is typed under CMS.

The command EDIT is used to create a file as indicated by the return message NEW FILE. INPUT signifies that the input environment has been entered, and the next six lines are input data to the user's file DME EXEC. When the return key is depressed without entering any characters on that line, the edit environment is entered. This allows

additions and changes to be made to the file. When FILE is entered, the data lines are written out on disk as file DME EXEC.

The user is now ready to read in a deck of cards containing DME. These cards are used to build a file with the filename of DME, but a filetype of IPL. First, however, two control cards must be put in front of the card deck. The first card must be the normal CP-67 ID card (ID must be punched in columns 1 and 2, and the userid starting in column 10). The second control card must contain the following information starting in column 1: OFFLINE READ DME IPL. The DME deck with these two control cards can then be placed in the system card reader and read in by CP-67. CP-67 puts the deck on the spooling disk (which is the "virtual card reader").

When the user types in the command OFFLINE READ *, the deck is read from the virtual card reader by CMS. This causes the file DME IPL to be built on the P-disk and the Diagnostic Monitor cards to be written into it.

The last command issued is LOGOUT, which terminates the user's operation under CP-67.

When the command DME is entered while operating under CMS, a deck of cards representing the Diagnostic Monitor DME is transferred into the user's virtual card reader, from which DME is automatically IPL'ed into the virtual machine.

The EXEC file need be created only once, and the OFFLINE READ * function need be performed only once, since the files are permanently saved on the P-disk.

```
R|  cp-67 online    xd.65 qsyosu
 |            <----Hit ATTN Key
C|  login ce      <----Log in with your userid
R|  ENTER PASSWORD:
C|            <----Type protected password
R|  CP67 WILL RUN UNTIL 24:00  <----Message of the day
R|  READY AT 09.41.29 ON 12/09/69
R|  CP
C|  ipl cms
V|  CMS...VERSION 3.0 -- 11/04/69
```

(See FORMAT note at the end of "Defining a Customer Engineer's Virtual Machine")

```
C|  edit dme exec
E|  NEW FILE.
E|  INPUT:
D|  &typeout off
D|  cp close c
D|  cp purge rdr
D|  cp xfer d to * <--* means the userid issuing the command
D|  offline punch dme ipl
D|  cp ipl c
D|        <---Hit RETURN for null line
E|  EDIT:
C|  file
V|  R; T=0.05/0.35 09.42.39
V|  **  CARDS HAVE BEEN READ ** <--deck entered by operator


C|  offline read * <---Reads deck in virtual card reader
V|  OFFLINE READ DME IPL <---This is the information contained
V|  R; T=0.08/0.20 09.45.00      in offline read control card.


C|  cp logout
R|  CONNECT= 00.04.42 VIRTCPU= 000.00.17 TOTCPU= 000.00.83
```

C = Command for CP-67/CMS issued by user
D = Data for CP-67/CMS issued by user
R = Response from CP-67
V = Response from CMS
E = Response from CMS EDIT command

Figure 2. Sample terminal session to build DME EXEC file

SAMPLE COMMAND STREAM TO RUN A DIAGNOSTIC

A sample command stream for running diagnostics is shown in Figure 3:

Note: For consistency, the sample describes the running of the DME EXEC file built in the previous section. The DIAGS EXEC procedures are more inclusive and easier to run and are described in Appendix A.

Initially, the communications between the user at the terminal and CP-67 must be established.

The user logs in with his userid and the correct password.

Following the log message, two messages are written indicating that devices 181 and 236 are attached to the user. This is a result of having asked the operator to ATTACH these devices, either verbally or via the message function.

After these messages, the user can enter the command IPL CMS. This causes the Cambridge Monitor System to be loaded into the user's virtual machine. Following the declaratory message for CMS, the command DME is entered. This causes the command sequence explained in the section "Building an EXEC File for Loading DME" to be executed. The Diagnostic Monitor DME has now been IPL'ed and the user is in the environment of DME. This means that the Diagnostic Monitor has overlaid CMS in the user's virtual machine. The terminal now appears to the user as the console typewriter. The only difference is that the user must communicate with CP-67 for certain preparatory functions to be performed in the diagnostic program. One of these functions is the simulation of the external interrupt key. One depression of the ATTN key at any time during this period puts the user in communication with CP.

This is illustrated by the sequence beginning after the message WTE IPL DME REV LEVEL 9. ATTN causes CP to be given control. A second ATTN causes control to be passed to DME. The Diagnostic Monitor (DME) instruction "l/b" is entered, and DME is given control. Depressing the ATTN key the next time allows the user to type in EXT, which tells CP to signal an external interrupt to DME. The next two ATTN requests put DME in control, so the user can type in the message describing the DM Loader Device and what DME is to run for diagnostics.

The Diagnostic Monitor writes an appropriate message on the terminal after running the requested diagnostics. The user should then hit ATTN. This puts CP in control, and the

user can issue DETACH commands to release the devices ATTACH'ed by the console operator at the start of the run. The user then issues the command LOGOUT to release the terminal. Figure 3 follows.

```
 R|  cp-67 online    xd.65 qsyosu
 C|  login ce
 R|  ENTER PASSWORD:
 C|
 R|  CP-67 WILL RUN TILL 17:00 AND FROM 18:00 TO 24:00
 R|  READY AT 14.50.12 ON 12/03/69
 R|  CP

 R|  DEV  181 ATTACHED <-Diagnostic tape mounted on this drive
 R|  DEV  236 ATTACHED <-Device to be tested
 R|  CP

 C|  ipl cms
 V|  CMS...VERSION 3.0 -- 11/04/69

 C|  dme
 V|  NO FILES PURGED
 V|  ** CARDS  XFERED BY CE    **
M2|   WTE IPL  DME REV LEVEL 9

 R|  CP          <---ATTN key hit
           <---ATTN key hit
 I|  l/b
M2|   MODEL 65  256K         03EFFF
 R|  CP          <---ATTN key hit
 C|  ext

M2|   WTE DME 9
 R|  CP          <---ATTN key hit
       <---ATTN key hit
 I|  a50,0181/a6a,0236/i10181/16a0,6bd,0236/b <-initialize DME
M2|   S    F6A03                                 loader and specify
M2|   T                                          tests.
M2|   S    F6A13
M2|   T
M2|   S    F6A23
M2|   T
M2|   S    F6A33
M2|   T
M2|   S    F6A43
M2|   T
M2|   S    F6A53
M2|  *SD0  F6A53 01  0035B2  0236
M2|  SHLD GET CC1 ON SEEK INDICATING CUB BUT DID NOT CSW IS 000035E80C000000
M2|   T
M2|   S    F6A63
M2|  *SD0  F6A63 01  0033B4   0236
M2|  ERR CSW 0000363002000011 CSW SHLD BE 0000363072000011
M2|  SNS 801000400100 SNS SHLD BE 80100040XXXX
M2|   T
M2|   S    F6A73
M2|  *SD0  F6A73 01  0033B4   0236
M2|  ERR CSW 000035002000009 CSW SHLD BE 0000350072000009
```

```
M2|  SNS 801000400100 SNS SHLD BE 80100D40XXXX
M2|  *SD0  F6A73 01  0033B4   0236
M2|  ERR CSW 0000356802000009  CSW SHLD BE 0000356872000009
M2|  SNS 801000400100 SNS SHLD  BE 80100040XXXX
M2|  *SD0  F6A73 01  0033B4   0236
M2|  ERR CSW 000037200200000E  CSW SHLD BE 000037207200000E
M2|  SNS 801000400100 SNS SHLD  BE 80100040XXXX
M2|   T
M2|   S    F6A83
M2|   T
M2|   S    F6A93
M2|   T
M2|   S    F6AA4
M2|  *SD0  F6AA4 01  003350   0236
M2|  ERR-E R A S E  CAW 003458
M2|  EXP CSW 000034787C000000 ACT CSW 000034780C000000
M2|  EXP SNS 00000040XXXX ACT SNS 000000400100
M2|  EXP READ 00C3000003 ACT READ 00C3000003
M2|  *SD0  F6AA4 02  00358A   0235
M2|  ERR-SPACE COUNT  CAW 0037F0
M2|  EXP CSW 0000381872000003 ACT CSW 0000381802000003
M2|  EXP SNS 00000040XXXX ACT SNS 080000400100
M2|  EXP RD DATA 2000000002
M2|  ACT RD DATA 2000000002
M2|   T
M2|   S    F6AB3
M2|   T
M2|   S    F6AC3
M2|   T
M2|   S    F6AE3
M2|   T
M2|   S    F6AF3
M2|   T
M2|   S    F6B03
M2|   T
M2|   S    F6B13
M2|   T
M2|   S    F6B23

M2|  *SD0  F6B23 01  003394   0236
M2|  CSW 000036C000200000 CSW SHLD BE 000036C00C000001 SNS 000000400100
M2|  SNS SHLD BE 000000400X00 CAW 00003698
M2|  ERR WHEN SCAN BQ KEY DATA REMAINDER OF CYL, AFTER DEF TRK VIA 6TH SYS BY
M2|  TO LOOP ERR TURN ON SSW0
M2|   T
M2|   S    F6B33
M2|   T
M2|   S    F6BA3
M2|   T
M2|   S    F6BB3
M2|   T
M2|   S    F6BC3
M2|   T
```

```
M2|   S    F6BD4
M2|   T
M2|   WTE DME 9
 R|  CP     <---ATTN key hit
 C|  detach 236
 R|  DEV  236 DETACHED

 C|  detach 181
 R|  DEV  181 DETACHED

 C|  logout
 R|  CONNECT= 00.12.17 VIRTCPU= 003.11.66 TOTCPU= 005.23.51
 R|  LOGOUT AT 15.45.58 ON 12/03/69
```

C = Command issued by user
I = Instruction for diagnostic monitor issued by user
M2 = Message from diagnostic monitor
R = Response from CP-67
V = Response from CMS

Figure 3. Sample Terminal Session for Running 2314 Diagnostics

OLTS TEST PROCEDURE FOR A 3420

The Process of initiating an OLTS Test on a 3420 is similar to DME and 240X diagnostics. To initiate an OLTS test, the following procedure is followed:

- Mount the 3420 with a blank reel of tape with the file protect ring installed. 'READY' the device.
- Mount the OLTSEP diagnostic tape on another tape drive. 'READY' the device.
- System operator attaches (ATTACH) the above two devices to the CE's virtual machine.
- CE logs in (LOGIN) and IPLs the OLTSEP diagnostic tape.
- The response on the terminal indicates OLTS operation.
- CE selects test and options described in OLTS and OLTSEP literature.

Figure 3.1 illustrates the virtual machine printout of the operation just described.

```
CP-67 online   xd.65 qsyosu

l ce
ENTER PASSWORD:

194 NOT AVAILABLE
SYSTEM WILL RUN FROM 08:30 TO 17:00
AND FROM 18:00 TO 24:00
READY AT 11.35.33 ON 10/30/71
CP

m operator please attach 181 and 390 to me as 181 and 390

DEV 181 ATTACHED
DEV 390 ATTACHED
CP

ipl 181
04 SEP188D ENTER DATE IN FOLLOWING FORMAT 'MM/DD/YY'
r 04, '10/30/71'
04 SEP330D ENTER TIME IN THE FORMAT 'HH.MM.SS'
r 04, '11.38.00'
04 SEP102I OLTS RUNNING
04 SEP107I OPTIONS ARE NTL,NEL,NPP, FE,NMI, EP, CP, PR, SI
01 SEP1050 ENTER DEV/TEST/OPT/
r 01, '390/2313a-f/nfe,pp(3)/'
04 SEP158I  S T2313A  UNIT 0390
04 SEP158I  T T2313A  UNIT 0390
```

The start terminate messages will continue down through test 2313F at which time OLTSEP will request the next device, test, and options to be run.

Figure 3.1. Sample Printout of OLTSEP Test from a Virtual Terminal

INSTRUCTIONS FOR RUNNING FRIEND UNDER CP-67

In addition to the standard diagnostics under CP, FRIEND has been used extensively with no known problems. It should be noted that in the CP-67 error-recording mechanism, the failing CCW string on I/O errors is recorded with the error information. This makes FRIEND much more valuable as a trouble-shooting aid under CP.

The contents of FRIEND EXEC are shown in Figure 4.

```
print friend exec
&TYPEOUT OFF
CP PURGE RDR
CP XFER D TO *
OFFLINE PUNCH FRIEND IPL
CP IPL C
R; T=0.03/0.10  09.45.00
```

Figure 4. Sample Terminal Session To Display FRIEND EXEC File

When the command FRIEND is entered while operating under CMS, a deck of cards representing FRIEND is written into the user's virtual card reader, from which it is IPL'ed into the virtual machine.

A sample terminal session for running FRIEND is shown in Figure 5.

FRIEND can be loaded by an alternate means if necessary if it does not already exist on a CMS disk. This alternate procedure is as follows: An ID card can be placed in front of the FRIEND deck and put in the system card reader. CP-67 reads the deck in, and puts it on the spooling disk (the virtual card reader). Then, when CP control has been established, type in IPL C and wait a couple of seconds. The keyboard unlocks. Signal an EOB, and the FRIEND heading types out. If you wish to re-IPL FRIEND a number of times, issue the SET CARDSAVE ON command to CP. When you have finished using FRIEND, issue to CP the command SET CARDSAVE OFF.

To clear FRIEND from the virtual card reader, issue the CP command PURGE RDR.

```
 R|  cp-67 online    xd.65
  |                <---Hit ATTN
 C|  login ce
 R|  ENTER PASSWORD:
 C|                  <---Password entered
 R|  CP-67 WILL RUN TILL 24:00
 R|  READY AT 13.33.43 ON 12/08/69
 R|  CP
 C|  ipl cms
 R|  DEV 101 ATTACHED
 V|  CMS...VERSION 3.0 -- 11/04/69
 C|  friend
 V|  NO FILES PURGED
 V|  ** CARDS XFERED BY CE
      <---keyboard should unlock - hit RETURN
 R|  FRIEND-3-DES"CP  <---Friend now running in users
 C|  reset                virtual machine - hit ATTN
M2|  DEV = DEVICE ADDRESS XXX = 101
M2|  ENTER CCW LIST IN ENGLISH
 I|  sk
 I|   cyl=00
 I|   hd=190
 I|  read ha
 I|  loop
M2|  NUMBER OF TIME = 100
 I|  go
M2|  LOOP IS FINISHED ON UNIT 101
 R|  CP          <---ATTN key was hit
 C|  log
 R|  CONNECT=.   .02.27  VIRTCPU= .2.23.62
 R|  LOGOUT AT 11.16.23 ON 01/08/69
```

```
C  = Command for CP-67/CMS--issued by user
M2 = Message from FRIEND
I  = Instructions to FRIEND--issued by user
R  = Message from CP-67
V  = Message from CMS
```

Figure 5. Sample terminal session for running FRIEND

ERROR-RECORDING TECHNIQUES

This section is written for the customer engineer to more fully acquaint himself with the error-recording mechanism used by CP-67.

The Logrec (SYSERR) area used by CP-67 is located on the CP system residence device (normally the label is CPDSK1), which can be a 2311 Disk Storage Drive or 2314 Direct Access Storage Facility. It is one cylinder long, and its location is specified in the CP SYSGEN macro SYSRES SYSERR=xxx, where xxx is the cylinder address desired. If the SYSERR option was not specified, the default cylinder address would be set to cylinder 4.

Before a CP-67 sysgen, the CP system residence volume must be initialized by the CP Format utility program, which writes dummy records without key fields for the entire disk (as shown in Figure 6).

When the newly created CP-67 control program is IPL'ed the first time, it further formats the Logrec cylinder by locating its address (passed via the SYSGEN macro), and inserts eight bytes of control information into the beginning of each physical data record contained on that cylinder (as shown in Figure 7).

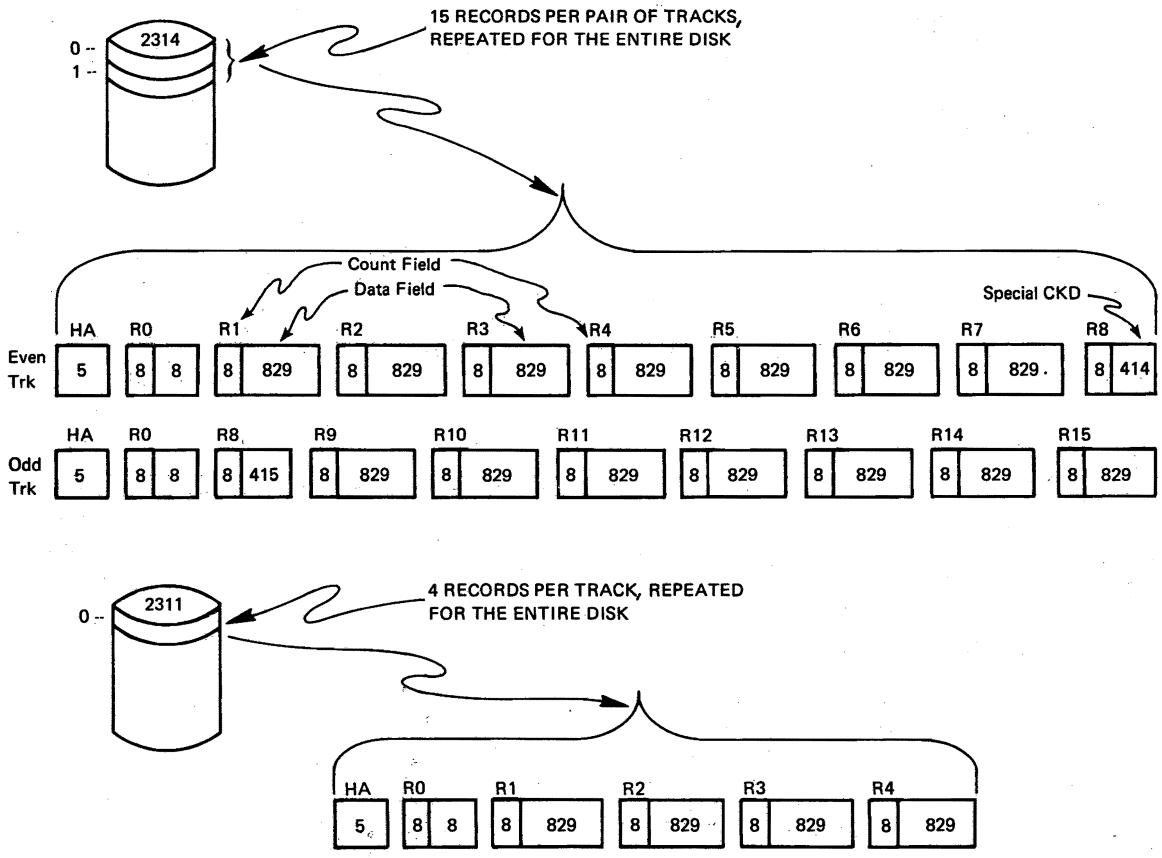
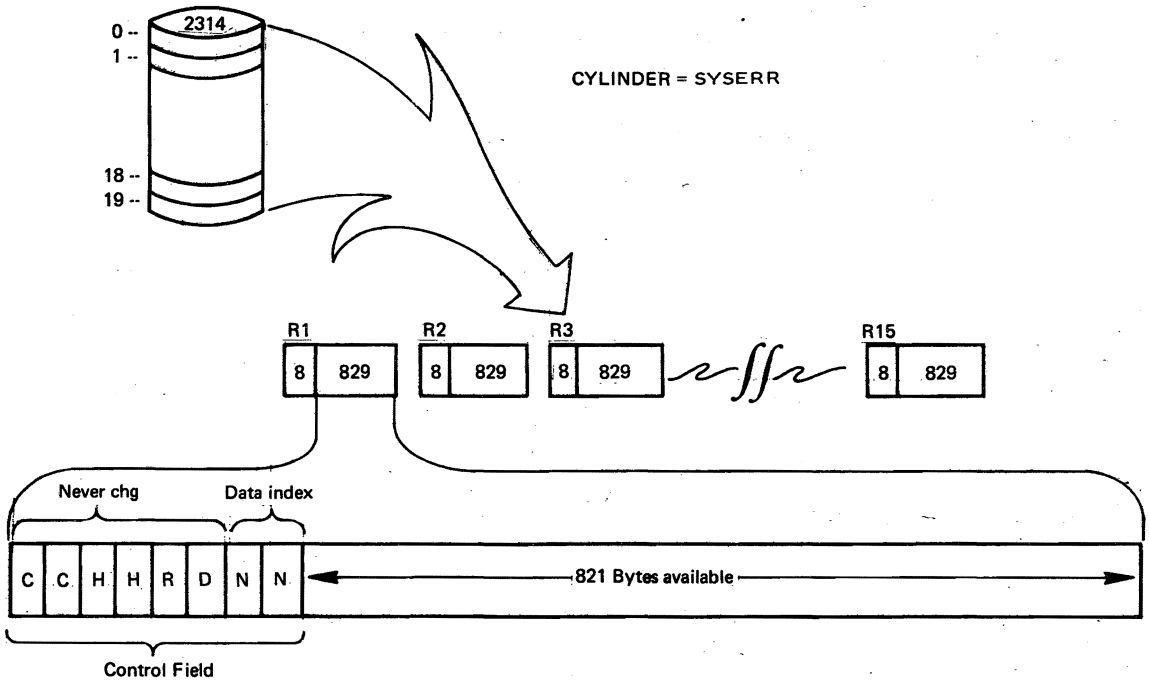


Figure 6. CP-67 Format for 2311 and 2314



CC = SYSERR
HH = Head this physical record on
R = Same as physical record in count field
D = 'FF' - Flag - Always FF
N = } Data Index - Initialized to zero. As error records are added,
N = } it contains the number of bytes used in the record.

Note: 2311 would be formatted exactly the same, except it would have four (4) physical records per track.

Figure 7. Initializing Logrec Records

The Logrec cylinder is split into two areas by CP. Tracks 0 and 1 are used to record machine-check errors; tracks 2-19 are used to record I/O outboard and channel errors (tracks 2-9 for 2311).

To locate the first or next available disk location for an error record to be written, the Control Program maintains two tables in core. The first table (label is IOERR) contains the disk address for the next available I/O error record slot, and the second table (label is MCERR) contains the disk address for the next available machine-check record slot. The tables, called the in-core record pointers, are initialized during the first IPL of CP to reflect the beginning of the I/O and machine-check recording slots (as shown in Figure 8).

```
IOERR       DS    OF
IOERRBB     DC    X'0000'     Bin--Always zero
IOERRCC     DC    X'XXXX'     SYSERR addr. See note.
IOERRHH     DC    X'0002'     Starting head for I/O errors
IOERRR      DC    X'01'       Record 1
IOERRD      DC    X'FF'       Flag
IOINDEX     DC    X'0000'     Data index

MCERR       EQU   *
MCERRBB     DC    X'0000'     Bin--Always zero
MCERRCC     DC    X'XXXX'     SYSERR addr. See note.
MCERRHH     DC    X'0000'     Starting head for Mch. Ck. errors
MCERRR      DC    X'01'       Record 1
MCERRD      DC    X'FF'       Flag
MCEINDEX    DC    X'0000'     Data index
SYSERR       DC    H'04'       Set at IPL--Default to '04'
```

Note: At IPL time CP initializes to whatever cylinder address is in the SYSERR location, as specified at SYSGEN.

Figure 8. Logrec in-core pointers, located in module IOERROR

Before the Control Program attempts to write an error record on Logrec, CP gets the next available physical record pointer (I/O or MCH) from the "in-core record pointers", and reads that physical data record into core. CP also updates the in-core pointers for the next error-recording requester, then uses the data index bytes (in the control field of the physical data record just read) to insert this error into the correct physical data record offset. CP adds this error record length to the data index bytes and writes the physical data record back to the same location it was read from. Figures 9 and 10 show the format of two physical

records, and the format of the machine-check and I/O outboard records.

Count Field

C	C	H	H	R	K	D	L
00	04	00	01	0A	00	03	3D

Data Field

C	C	H	H	R	D	N	N			53
00	04	00	01	0A	FF	01	80	Mach Ck Rec	Empty Slot	Not Used
								384	384	

Machine Check Record

Byte	Field Name	Field Description, Contents, Meaning
0-5	LOGMDATE	Mo/Day/Yr/Hr/Min/Sec
6-7	LOGMCODE	Extended Mck Int Code
8-183	LOGMCPU	CPU Logout Data
184-223	LOGMPSW	Old PSW's at time of Machine Check
224-287	LOGMGRS	General registers at time of Machine Check
288-351	LOGMCRS	Extended Control Registers
352-383	LOGMFPRS	Floating Point Registers

Note: The next available in-core record pointers would have CC=04, HH=01, R=0A, D=FF, INDEX=0180

Figure 9. Physical and machine-check record formats

Count Field

C	C	H	H	R	K	D	L
00	04	00	07	0F	00	03	3D

Data Field

C	C	H	H	R	D	N	N	112	112	112	112	112	112	112	37
00	04	00	07	0F	FF	03	10	I/O Rec *	I/O Rec *	I/O Rec *	I/O Rec *	I/O Rec *	I/O Rec *	I/O Rec *	Not Used

* These may be either I/O Error Records or Channel Check Records

I/O Error Record

Byte	Field Name		Field Description, Contents, Meaning
0-5	LOGSNSE		Sense Bytes from Device (up to 6, if device provides less than 6, the remainder contain 00). If the device type is 3420 these 6 sense bytes are not used. The 3420 provides 24 sense bytes and they are in the LOGCCWS area.
6	LOGCODE	CODE	Type of I/O Error
		Fx	First encountered error of this type
		Cx	Counter overflow for this error type
		x1	Bus Out
		x2	Equipment Check
		x3	Data Check
		x4	Seek Check or Data Convert Check
7	LOGTYPE		CP-67 Device Type Code (See Sec. 3, CP PLM)
8-13	LOGVOLID		Volume Serial Number of device upon which the error occurred (if known to CP)
14-15	LOGADDR		Channel/Unit Address of erring device
16-21	LOGDATE		Month/Day/Year/Hour/Minute/Second
22-27	LOGCSW		CSW at the time of the error
30-31			Reserved for future use
32-103	LOGCCWS		Failing CCW string (last 9 CCW's or less) If the device type is 3420, the first 3 double words contain the 24 sense bytes and only the last 6 CCW's, or less, of the failing CCW string are retained.
104-111	LOGSKLOC		Last seek address prior to the failure

Channel Check Record

Byte	Field Name		Field Description, Contents, Meaning
0-5	LOGSNSE		Up to 6 Sense Bytes
6	LOGCODE	CODE	Type of Channel Error
		Fx	All Channel Errors are flagged as first encountered errors for consistency with I/O error records
		xA	Channel Data Check
		xB	Channel Control Check
		xC	Interface Control Check
		xD	Chaining Check
7	LOGTYPE		CP-67 Device Type Code (See Sec. 3, CP PLM)
8-13	LOGVOLID		Volume Serial Number of device performing I/O when Channel Check occurred (if known to CP)
14-15	LOGADDR		Channel/Unit Address of device
16-21	LOGDATE		Month/Day/Year/Hour/Minute/Second
22-29	LOGCSW		CSW at time of error
30-31			Reserved for future use
32-39	LOGIOPSW		Old I/O PSW stored at time of error
40-63	LOGCHLOG		Channel Logout Data
64-67	LOGCAW		CAW at time of error
68-111			Unused

Note: The next available in-core record pointers would have CC=04, HH=08, R=01, D=FF, INDEX=0000.

Figure 10. Physical and I/O error record formats

If the attempt to read or write an error record fails, it is retried eight times. Upon continued failure, the error message ** IOERROR RECORDING FAILURE ON DEV XXX ** is sent to the operator, and that error record will be lost. If there is no more room in the machine-check or I/O outboard areas, the message ** LOGREC FULL; (I/O or M/C) ERRORS NOT RECORDED ** is sent to the CP-67 operator.

If CP crashes, it must, on the next IPL, reestablish the Logrec in-core record pointers. CP does this by reading in the first physical data record on the Logrec cylinder (machine-check area), and checking the control field. If the data index bytes reflect the maximum number of bytes that physical record can hold, CP reads in the next sequential physical record, repeating this until a record is found with some space left in it. CP then stores that physical data record control field into the machine-check Logrec in-core record pointers. CP repeats the above procedure for locating the next available physical record for I/O error records, starting at head 2, and storing its control field into the I/O error Logrec in-core record pointers. Error recording can now resume where it left off.

If CP found either the machine check or I/O error areas full, the message ** LOGREC FULL; (I/O or M/C) ERRORS NOT RECORDED ** is sent to the CP-67 operator.

A summary of the recording and recovery actions of the system for both machine checks and I/O and channel checks are shown in Figure 11.

	Problem State	Supervisor State
Recording	CPU Logout on CE Cyl (Date Time etc)	CPU Logout on CE Cyl
Recovery	M.C. Reflected to V.M. Message to operator Message to User	System Dump & Auto RE-IPL Message to operator

I/O & CHANNEL

		SELECTOR		MPX	
		CP I/O	V.M.I/O	V.M. I/O	CP I/O
RECORDING	IFCC CCC	Channel logout on CE Cyl. CAW, CSW, DEV. Addr VOLID, Date Time Msg. to operator		Not currently implemented	
RECORDING	Data UC Chain	Sense bytes & up to last 9 CCW's Time, Date, Dev. Type, Dev. Addr., Seek Addr CSW Msg. to Operator at Threshold			No recording currently Msg. to operator
RECOVERY	Data UC Chain	Retry 64 times	Reflect to V.M.		Retry up to 9999 *Times* Then log user out
RECOVERY	IFCC CCC	Retry 64 times Addr Known(?)	Reflect to V.M.		If MPX Chain Reset - System *must be re-IPLed*

Figure 11: Summary of CP-67 Error Recording and Recovery

CONDITIONS AND DEVICES INVOLVED IN RECORDING

CP-67 records various types of input/output and machine-check failures so that the customer engineer can review the specific types of hardware malfunction which occur during normal system operation. The customer engineer can use this information, in addition to CE diagnostic test information, in monitoring and maintaining system hardware.

I/O Errors

Four types of I/O errors are recorded by CP-67 when a unit check occurs as the result of an I/O operation. These errors are recorded according to the bit position in sense byte 0.

Error Type Indicator	Sense Bit	Type of Error
1	2	Bus Out
2	3	Equipment Check
3	4	Data Check
4	7	Seek Check (DASD) or Data Convert Check (24XX)

Counters for each of the above types of errors are maintained by CP for each I/O device represented by a selector channel device block (RDEVBLOK). This includes all I/O devices except shared unit record equipment and nondedicated terminals. If the virtual machine is defined with privilege class C, no errors associated with I/O tasks originated by that virtual machine will be recorded. This prevents the recording of intentional errors caused by CE diagnostic programs. The first error of a given type encountered after system IPL is recorded for each device. Subsequently, every eighth error of this type causes the counter to overflow and a counter overflow record to be recorded.

In cases where two of the five error conditions exist, the error is counted as belonging to the counter corresponding to the leftmost bit within sense byte 0.

In recording errors, no distinction is made between permanent I/O failures and temporary, or intermittent, failures, since the I/O operation is generally under control of the virtual machine and such information is not known to CP. For example, the eighth retry of a given I/O operation to a given device may cause a counter overflow record to be written. However, the eighth intermittent error of a given type to any location on a given device also causes a counter

overflow record to be written. Furthermore, a counter overflow record can be initiated by repetition of the same channel program, or by the intermittent failures of eight different channel programs. Therefore, successive entries under a specific counter for a given device may reflect the failure of unrelated channel programs.

Note that, since CP appears as an extension of the hardware to the virtual machine, there is no mechanism within CP to detect repeated failures of the same virtual machine I/O sequence. Rather, error conditions are reflected back to the virtual machine as they occur for each event. Accordingly, only the virtual machine's control program can distinguish between temporary and permanent I/O failures and take appropriate action. Therefore, the CP error-recording mechanism cannot distinguish between permanent and temporary virtual machine I/O errors.

In addition to I/O outboard errors, four types of channel errors are recorded. These are the following:

Error Type Indicator	CSW Bit	Type of Error
A	44	Channel data check
B	45	Channel control check
C	46	Interface control check
D	47	Chaining check

Every occurrence of a channel check is recorded by CP-67. Indicators are kept in the Real Channel Block (RCHBLOK) for each type of channel check. Channel errors which do not log out (channel data check and chaining check) are retrieved by the I/O Error Edit and Print program, CPIOEREP. Channel errors for which logout data is provided (channel control check and interface control check) are retrieved by CPMCEREP, and the logout data is formatted by SEREP1 (System/360 Model 67-1) or SEREP2 (System/360 Model 67-2).

The utility program CPIOEREP is provided to retrieve, edit, and print these I/O error records. The following information is provided about each recorded I/O error, channel data check, or chaining check:

Type of error
First encounter or counter overflow
Date and time of error
Device type and device address
Volume serial of DASD if known to CP
Sense and channel status word
The failing CCW string (up to the last nine CCW's)
Which CCW failed (identified by *)

For direct access devices, the last seek address

A sample I/O Error Record printout is shown in Figure 12.

```
I/O ERROR      COUNTER  NO. D    CHAINING CHECK      TYPE   FIRST ENCOUNTERED
DATE  08/18/70          TIME  09.16.59
DEVICE TYPE 2314        DEVICE ADDR    235          MODULE   A
      VOLID  APLSY3     SENSE  00 00 00 40 00 00    CSW 00076220 0C0109CC
      FAILING CCW STRING
            OP CODE     DATA ADDR    FLAGS    BYTE COUNT       CONTROL DATA
                                                               B  B  C  C  H  H
              07          076230       40        0006          00 00 00 16 00 11
              31          05BF3C       40        0005
              08          0761F8       60        0005
              06          04CC78       80        0388
              06          045000       80        00A8
            * 06          042634       80        09CC
```

Figure 12. Sample Printout of I/O Error Record

A total error count for each device since system IPL is also maintained by CP-67. If this error count reaches 8, a message is printed on the operator's console to alert him. This message identifies the device on which the errors occurred, the error count, sense information, and status information. The message will be repeated after a further 256 (or multiple of 256) errors occur. Thus an error message is printed when the error count on a device reaches 8, 264, 520, etc., in addition to the error recording on SYSERR. These messages are printed even if error recording is not in effect because the SYSERR is full.

All nonrecursive machine checks, whether in supervisor or problem state, are recorded by CP-67. If a machine check occurs in the problem state, the virtual machine user is placed in console function mode, and receives the message ** MACHINE CHECK ** - CP ENTERED, REQUEST, PLEASE. The system operator receives the message ** MACHINE CHECK PROBLEM MODE ** USERID =XXXXXXXX, and system operation continues.

If a machine check occurs in the supervisor state, the system operator receives the message ** MACHINE CHECK CP-67 SUPERVISOR MODE ** and the system dumps and automatically re-IPLs. If multiple machine checks occur while in the supervisor state, the system operator receives the message ** MULTIPLE MACHINE CHECKS ** RUN SEREP AND RE-IPL ** and the system goes into the wait state. Multiple machine checks are not recorded by CP-67.

Each machine-check error record contains the following information:

Date and time of error

Machine check code

The five old PSW's (external, SVC, program, machine check, and I/O)

Standard SEREP output without storage key printout. This includes the CPU logout data, general registers, extended control registers, and floating-point registers.

A program called CPMCEREP is used to retrieve the machine-check records from Logrec, and then to invoke the SEREP program to format the data. Instructions for running the CPMCEREP program as a part of the LOGPRINT CMS command may be found in the following section, "Instructions for Recording and Retrieval".

A sample Machine Check Record is shown in Figure 13.

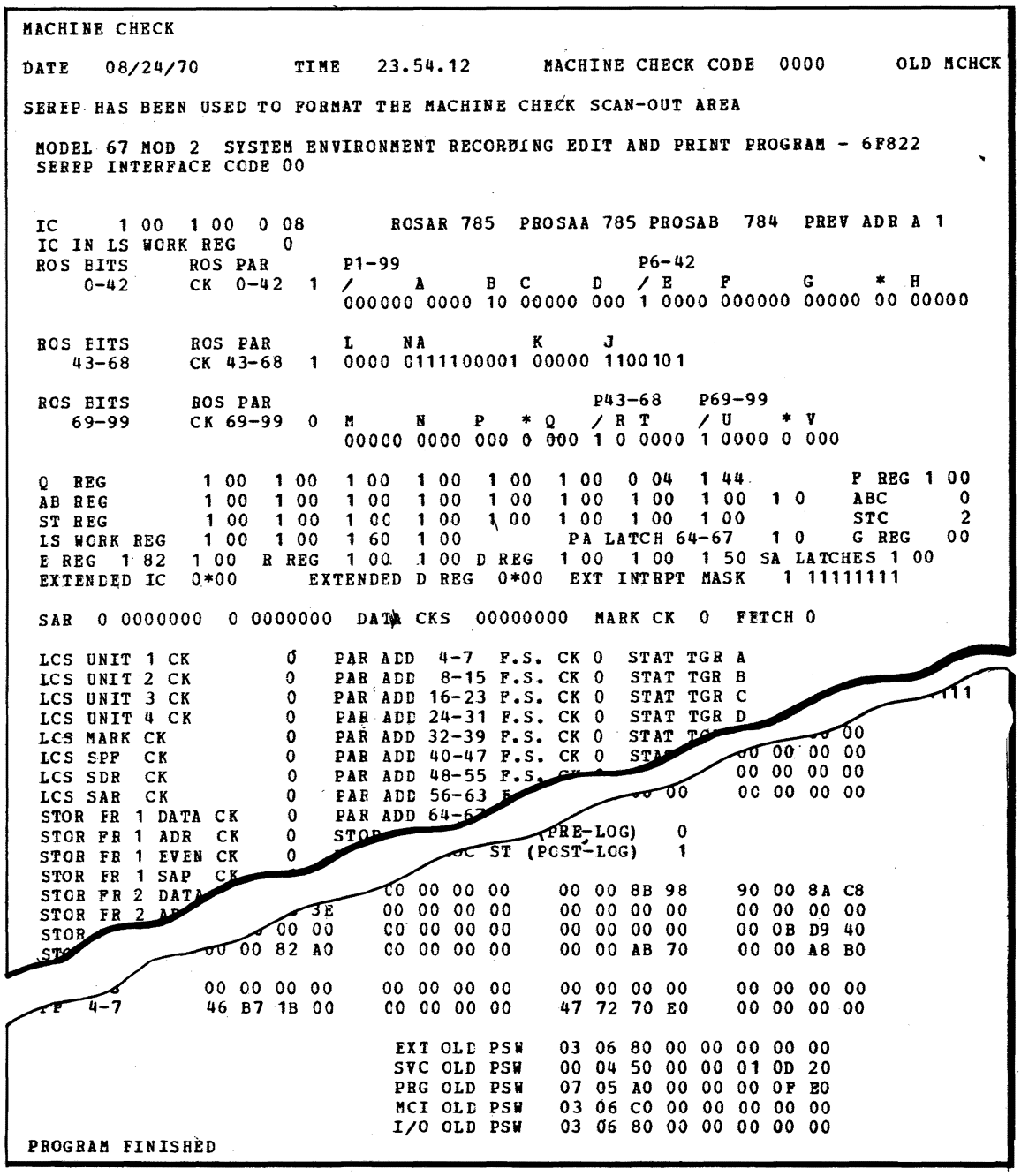


Figure 13. Sample Printout of Machine Check Record

Channel control checks and interface control checks are also retrieved by the CPMCEREP program. The following information is provided about each error:

Sense information

Error type code

Volume serial and channel/unit address of DASD, to or from which data was being transferred when channel error occurred

Date and time of error

Channel status word

I/O old PSW

Channel logout data

Channel address word

This data is also formatted by the SEREP program.

A sample Channel Error Record is shown in Figure 14.

```
CHANNEL CHECK                          INTERFACE CONTROL CHECK

     DATE  08/17/70          TIME  20.32.41

     DEVICE TYPE 2314        DEVICE ADDR    330          MODULE   F

          VOLID  MEYER1      SENSE  00 00 00 40 05 00     CSW F007B0D8 0C020000

     SEREP HAS BEEN USED TC FORMAT THE CHANNEL LOGOUT AREA

MODEL 67 MOD 2  SYSTEM ENVIRONMENT RECORDING EDIT AND PRINT PROGRAM - 6F822
SEREP INTERFACE CODE 0F


CHAN/UNIT ADR        330

I/O OLD PSW          03060330 00000000

CAW                  F007B0C8


CSW FIELDS

KEY                   F
COM ADR         07 B0 D8
UNIT STATUS
 ATTENTION             0    STAT MODIFIER          0  CTRL UNIT END          0
 CHAN END              1    DEVICE END             1  UNIT CHECK             0
 BUSY                  0    UNIT EXCEPTION         0
CHAN STATUS
 PCI                   0    INCORR LENGTH          0  PROG CK                0
 CHAN DATA CK          0    CHAN CTRL CK           0  I/F CTRL CK            1
 PROTECT CK            0    CHAINING CK            0
COUNT              00 00

                                                      MAND                   0
                                                      FLAG                   0
                                                      SIM I.F.               0

                       0
 BIT 2 BC REG          0
 BIT 1 BC REG          0
 SEQ TGRS REG         09
PROGRAM FINISHED
```

Figure 14: Sample Printout of Channel Error Record

For a detailed description of I/O error recording by CP-67, see CP-67 Program Logic Manual.

Print Logrec

A set of CMS files, called CE serviceability files, are provided with the basic program material for CP-67. These programs have been included on the CP-67 basic program material tape as file 5. (See CP-67 Application Directory for details.)

In order to load these serviceability files into a virtual machine, the customer engineer should execute the following procedure after the CP-67 system source tape has been attached to his virtual machine as 181 by the system operator:

1. IPL CMS.

2. Issue the TAPE REWIND command.

3. Issue the TAPE SKIP 4 command to position the tape to file 5.

4. Once the tape has been positioned, issue the TAPE LOAD command to load the files into the virtual machine.

5. Issue the MODEL 1 or MODEL 2 command according to whether the Model 67 CPU is a simplex (Model 1) or half-duplex (Model 2). This enables the customer engineer to obtain the appropriate version of SEREP for analysis of the machine-check and channel error recordings.

Figure 15 is a sample terminal session to load and initialize the CE serviceability files.

```
R|  cp-67 online    xd.65 qsyosu
C|  login ce
R|  ENTER PASSWORD:
C|            <---Type protected password
R|  CP-67 WILL RUN 24 HRS. A DAY, 7 DAYS A WEEK. <-message of the day
R|  READY AT 13.15.10 ON 04/10/70
R|  CP
R|  DEV 181 ATTACHED
C|  ipl cms
V|  CMS..VERSION 3.0 - 04/09/70
V|  R; T=0.05/0.27 13.17.05
```

(See FORMAT note at end of "Defining a Customer Engineer's Virtual Machine").

```
C|  tape rewind
V|  R; T=0.03/0.17 13.17.27
C|  tape skip 4
V|  R; T=0.02/0.30 13.18.05
C|  tape load
T|  LOADING . . .
|       .
|       .
|       .
|    Files loaded are listed here
|       .
|       .
|       .
V|  R; T=0.10/0.53 13.19.24

C|  model 2
X|  13.19.31 EDIT LOGPRINT EXEC
E|  EDIT:
E|  OFFLINE PUNCH SEREPX IPL
E|  OFFLINE PUNCH SEREP2 IPL
V|  R; T=0.16/0.61 13.19.34

C = Command CP-67/CMS - issued by user
R = Message from CP-67
V = Message from CMS
T = Response from CMS TAPE command
X = Response from CMS EXEC command
E = Response from CMS EDIT command
```

Figure 15. Sample terminal session to load CE Serviceability Files

Note that each virtual machine defined for a customer engineer of an installation must have the privilege class of C in order to clear the LOGREC area.

In order to obtain a printout of I/O, channel, and machine checks which have been recorded, this procedure should be followed:

IPL 190 (do not IPL CMS).
Issue the LOGPRINT command.

If only the I/O error printout is desired, the command LOGPRINT IO should be issued. If only the machine-check error printout is desired, the command LOGPRINT MC should be issued. A sample terminal session, in which LOGPRINT is invoked, is shown in Figure 16.

```
R|  cp-67 online    xd.65 qsyosu
C|  login ce
R|  ENTER PASSWORD:
C|         <---Type protected password
R|  CP-67 WILL RUN 24 HRS. A DAY, 7 DAYS A WEEK.
R|  THE "TN" CHAIN IS ON THE PRINTER.
R|  READY AT 13.20.44 on 04/10/70
R|  CP

C|  ipl 190      ** Note - do not IPL CMS **
V|  CMS..VERSION 3.0 - 04/09/70

C|  logprint
X|  13.30.55  CLOSIO PRINT OFF
X|  CPIOEREP - I/O ERROR EDIT AND PRINT
P|  ALL RECORDS PROCESSED
X|  13.31.03 CP XFER D TO *
X|  13.31.05 CP CLOSE 00C
X|  13.31.07 CP PURGE RDR
R|  NO FILES PURGED
X|  13.31.09 OFFLINE PUNCH SEREP2 IPL
R|  ** CARDS  XFERED BY CE       **
X|  13.31.16 CP SET CARD ON
X|  13.31.20 CP XFER D OFF
Q|  MACHINE CHECK ERROR EDIT AND PRINT
R|  01 FILES PURGED
V|  CMS..VERSION 3.0 - 04/09/70
```

C = Command CP-67/CMS, issued by user
R = Message from CP-67
V = Message from CMS
X = Message from CMS EXEC command
P = Message from CPIOEREP program
Q = Message from CPMCEREP program

Figure 16. Sample terminal session for running LOGPRINT

CPIOEREP and CPMCEREP are the CMS programs which print the I/O error log and machine-check log, respectively. These commands are invoked by LOGPRINT, which is a CMS EXEC procedure. The source for these programs is included on the CE serviceability files as CPIOEREP SYSIN and CPMCEREP SYSIN. It is necessary to reassemble these files if the address of the CE cylinder is other than 004, or if a 2311 is used as the CP SYSRES. The procedure for modifying the conditional assembly options and reassembling is explained in the module's prose. If these files need to be reassembled, the macro libraries MACUP, CPMACS, and SYSLIB, must be included at assembly time. If there is no need to reassemble these programs, CPIOEREP SYSIN and CPMCEREP SYSIN may be erased from the P-Disk in order to conserve space. In addition, the version of SEREP which is not applicable to the installation (SEREP1 IPL or SEREP2 IPL) may also be erased. Standard CE diagnostics, such as DME, and FRIEND, may not be loaded into the CE's virtual machine by the procedure described in previous sections (i.e., by using the CMS tape load command). To load these diagnostics monitors or their diagnostic sections, they must be punched (either physically or virtually) and read by the CMS "OFFLINE READ filename filetype" command.

Create 3420 History Tape

A special CMS EXEC procedure and program are provided for installations with IBM 3420 tape drives. The EXEC procedure (M3420 EXEC) invokes the HIST3420 program to produce a history tape. The program scans the I/O error records on the CE cylinder and selects those error records for the 3420 tape drives. The program creates a history record for each 3420 record and writes the record on a tape. The tape is written with no label and unblocked 122-byte records and is suitable for processing by the Error Recording Analysis Program (ERAP) run with OLTSEP.

The M3420 EXEC procedure should be used before the CE cylinder is cleared but may be run either before or after the LOGPRINT function since the data on the CE cylinder is not modified.

The following steps should be followed to create a history tape for 3420 error records:

1. Login as the 'CE' virtual machine. This is the same ID used to perform the LOGPRINT function.

2. IPL CMS as either CMS or 190 or whatever the installation provides.

3. Request the CP-67 operator to attach the tape (to become the history tape) to the 'CE' machine as address 181 with the ring in.

4. Issue the CMS command M3420.

The following messages will be issued by M3420 and HIST3420 procedure:

```
EXECUTION BEGINS . . .

3420 HISTORY TAPE BEING PRODUCED

HIST3420 - 3420 HISTORY TAPE MAKER

ALL RECORDS PROCESSED

R;
```

Once the procedure has finished, the 'CE' may 'detach 181' or 'logout' of CP-67. The tape should be retained for input to the ERAP program with OLTSEP. This function may also be performed in a virtual machine environment subject to the usual CP-67 timing and operation restrictions.

Figure 16.1 shows the terminal printout and user input when running the M3420 EXEC procedure. It shows the output produced if the history tape is in a not-ready status and then is made ready for normal completion, and the results if the tape is not attached.

Figures 16.2 and 16.3 show the structure and content of the history tape records produced for the 3420.

Figure 16.4 is a hex printout of four typical history tape records.

```
m3420
TAP2 NOT READY YET
(OK - READY NOW)
EXECUTION BEGINS . . .
3420 HISTORY TAPE BEING PRODUCED
HIST3420 - 3420 HISTORY TAPE MAKER
ALL RECORDS PROCESSED
R;
cp detach 181
DEV  181 DETACHED
R;
m3420
TAP2 NOT ATTACHED
TAPE DRIVE NOT READY
R;
```

Figure 16.1 Terminal printout and user input under M3420 EXEC procedure

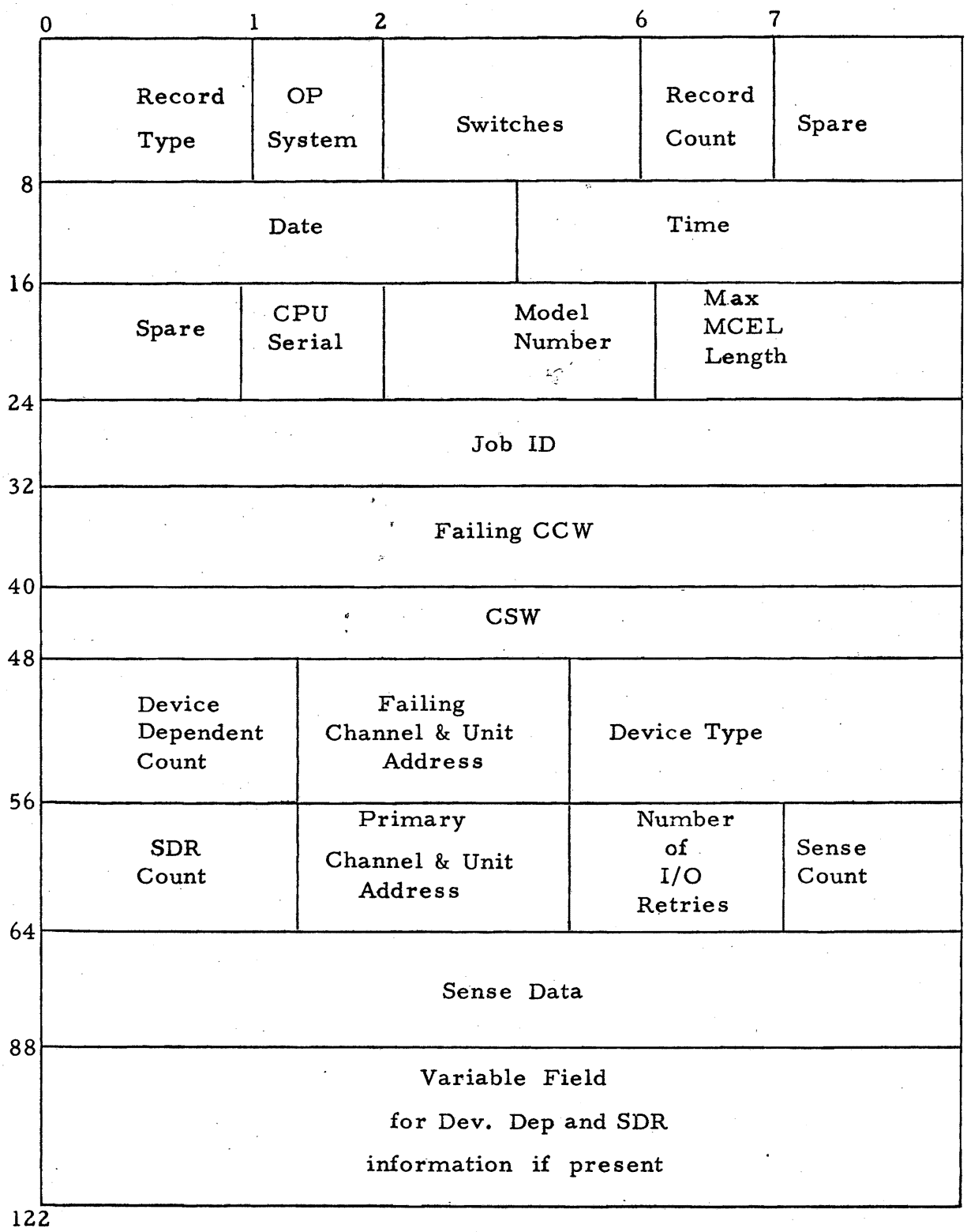


Figure 16.2 Structure of 3420 history tape.

Record Type	- Set to x'30' = OBR unit check
OP System	- Set to x'63' = CP67 Release 3.1
Switches	- Set to x'03000000' = Relocate - EC Mode
Record Count	- Set to x'11' = Indicate fixed record
Date	- Actual DATE ERROR was RECORDED - from LOGDATE
Time	- Time FAILURE was RECORDED - from LOGDATE
CPU Serial	- N/A 360
Model Number	- Set to x'0067' = Model 67
Max MCEL Length	- Set to x'0024' = 36-byte variable field
Job ID	- Set to "CP67-3.1" = User ID not available
Failing CCW	- Actual CCW that had unit check = taken from LOGCCWS
Failing CSW	- Actual CSW that showed error status = from LOGCSW
Device Dependant Ct.	- Always set to zero
Failing Address	- Address of failing unit - taken from LOGADDR
Device type	- Set to '00008003' - 3420 device type
SDR Count	- Always set to zero
Primary Address	- Address of failing unit - taken from LOGADDR
Number of retries	- Set to zero
Sense count	- Set to x'0018' - 24 sense bytes
Sense data	- 24 bytes of sense information - from LOGCCWS
Variable field	- To be used in future for device dependent and SDR data.

Figure 16.3 Contents of 3420 History Tape

```
1
122  306303000000110010217115220 4F00300000000 6724C3D7F6F760F348F10102C9CB205C0325F003  *.............0.......CP67.3.1..I.....0.*
     80080F000000000003900000800300000390000000180844100C00400400000000000008143028A11  *.............................. .............*
     0000001A000800000000000000000000000000000000000000000000000000000000000000000000  *.....................................*
     0000                                                                               *..                                 *

2
122  306303000000110010217115220 5F00300000000 6724C3D7F6F760F348F10102C9CB205C0325F003  *.............0.......CP67.3.1..I.....0.*
     8D480F000000000003900000800300000390000000180844100C00400400000000000008143028A11  *.............................. .............*
     0000001A000800000000000000000000000000000000000000000000000000000000000000000000  *.....................................*
     0000                                                                               *..                                 *

3
122  306303000000110010227118395 9F00300000000 6724C3D7F6F760F348F10203764C245C7FFF8003  *.............0.......CP67.3.1.........0.*
     76480E007C8400000390000080030000039000000 1808C0FFD4084004000000000000008143028A11  *.....d.....................M. .............*
     0000001A01CF000000000000000000000000000000000000000000000000000000000000000000000  *.....................................*
     0000                                                                               *..                                 *

4
122  306303000000110010227118400 0F00300000000 6724C3D7F6F760F348F10203ABCC245C7FFFF003  *.............. .0.......CP67.3.1.........0.*
     ABC80E007C840000039000008003000003900000001808C0FFD4084004000000000000008143028A11  *.H.........................M. .............*
     0000001A01CE000000000000000000000000000000000000000000000000000000000000000000000  *.....................................*
     0000                                                                               *..                                 *
```

Figure 16.4. Hex Printout of Four History Tape Records.

Clear Logrec

In order to clear the CE cylinder, the programs CLEARIO and CLEARMC may be invoked. These programs execute a special 'diagnose' instruction (code X'1C') to invoke a CP-67 routine that clears the relevant section of the CE cylinder SYSERR and starts error recording from the beginning. CLEARIO and CLEARMC may be executed at the discretion of the customer engineer, but should always be executed when the message ** SYSERR FULL; (I/O or MC) ERRORS NOT RECORDED appears at the operator's terminal. Note that the LOGPRINT should be run before clearing Logrec.

Change Sense Counter Limit

The customer engineer may wish to change the value at which the error counter overflows for one or more types of I/O error. This location (the entry point CTRLIM in the CP Load Map) contains a one-byte counter limit for each of the counters types 0-4. Counter 0 is currently set to zero; consequently no command reject errors are recorded. Counters 1-4 are each set to 08 so that every eighth error of a given type is recorded. To display the current value of the counters, the system programmer can display the address indicated for CTRLIM in the load map. He may also dynamically modify these counter limits. If a count is reset to 01, every error of this type is recorded, if the counter is reset to 02, every second error is recorded, etc.

APPENDIX A: STATUS OF DIAGNOSTICS UNDER CP-67

The following notes apply to the tests in this appendix:

Note 1. Time dependency routine

Note 2. Dynamically modified CCW sequence

Note 3. Diagnose instruction used

Note 4. Cause of failure has not been determined

Note 5. The core addresses in the SD0 messages should be adjusted to reflect the appropriate Diagnostic Monitor being used.

The decision whether to include test sections depends on answers to the following questions:

- What devices are available in system configuration of test system?
- What test sections are available on diagnostic tape?
- Would the test section cause a system crash? (It should not.)
- Is the test too cumbersome to run under CP-67?

Due to varying system load conditions, it is possible that not all of the documented errors will occur; it is also possible that some other errors could occur.

DASD diagnostics may not function properly on virtual minidisks. In order to ensure correct operation, the DASD device that is to be exercised by the diagnostic test sections should be dedicated in its entirety to the virtual machine by the use of the ATTACH command.

2820-2301 Test

Test Section	Reason for Exception	Exception Message Number
691-3	None	(Might not work on two-drum system)
692-3	None	
693-3	None	
694-4	None	
695-3	None	
696-3	None	
697-3	None	
69A-4	None	
69B-3	Note 1	Msg. 1, 2

Msg. 1
```
*SD0  F69B3 02  007D0E  0101
 LISTING ADDRESS 006102
CHECK THAT BUSY CONDITIONS ARE RETURNED CORRECTLY.
CSW COMMAND ADDRESS - 006AF8, CSW COUNT - 0000
CONDITION CODE ERROR. RCD -00-  EXP -01-
STATUS 0C 00 RCVD
       00 00 EXPCTD
```

Msg. 2
```
*SD0  F69B3 02  007F2E  0101
 LISTING ADDRESS 0072A4
THIS ROUTINE HAS FAILED
DEPENDENT ROUTINES WILL NOT BE RUN
```

2841-2311 Test

Test Section	Reason for Exception	Exception Message Number
603-6	None	
604-6	None	
605-6	Note 1	Msg. 1, 2
606-1	Note 1	Msg. 3
607-1	Note 1	Msg. 4, 5, 6
608-1	None	
609-1	None	
60A-6	Note 1	Msg. 7
	Note 4	Msg. 8, 9
610-2	None	
612-2	None	
613-3	None	
616-0	None	
617-0	None	
618-1	None	
61B-1	None	

```
          S     F6056
Msg. 1 *SD0   F6056 01   0069F6   0391
       SHLD GET CC1 ON SEEK INDICATING CUB BUT DID NOT CSW IS 00006A300C000000

Msg. 2 *SD0 F6056 03 006DF6   0391
       ERR INV OP 01 CSW SHLD BE 0000682002000001 CSW IS 000068280C000001
       SNS DATA SHLD BE 800000C8 SNS IS 000000C8
          T
          S     F6061
Msg. 3 *SD0   F6061 01   0063A4   0391
       ERR CSW 0000662002000011 CSW SHLD BE 0000662072000011
       SNS 801000C8 SNS SHLD BE 801000C8
          T
          S     F6071
Msg. 4 *SD0   F6071 01   0063A4   0391
       ERR CSW 000064F002000009 CSW SHLD BE 000064F072000009
       SNS 801000C8 SNS SHLD BE 801000C8
Msg. 5 *SD0   F6071 01   0063A4   0391
       ERR CSW 0000655802000009 CSW SHLD BE 0000655872000009
       SNS 801000C8 SNS SHLD BE 801000C8
Msg. 6 *SD0   F6071 01   0063A4   0391
       ERR CSW 000067100200000E CSW SHLD BE 000067107200000E
       SNS 801000C8 SNS SHLD BE 801000C8
          T
          S     F60A6
Msg. 7 *SD0   F60A6 01   00691E   0391
       ERR-SPACE COUNT  CAW 006A60
       EXP CSW 00006B7872000003 ACT CSW 00006B7802000003
       EXP SNS 801000C8 ACT SNS 801000C8
Msg. 8 *SD0   F60A6 02   00691E   0391
       ERR-SPACE COUNT  CAW 006A60
       EXP CSW 00006BD00C000000 ACT CSW 00006BD00E400000
       EXP SNS 000000C8 ACT SNS 080000C8
       EXP RD DATA 2222222222
       ACT RD DATA 2222222222
Msg. 9 *SD0   F60A6 02   00691E   0391
       ERR-SPACE COUNT  CAW 00D416
       EXP CSW 00006BF00E000000 ACT CSW 00006BF00E400000
       EXP SNS 080000C8 ACT SNS 080000C8
       EXP RD DATA 11112222222222FFDDCCFFFFFFFFFF
       ACT RD DATA 1111EEEECCFFFFFFFFFFFFFFFF00
          T
```

2314 Test

Test Section	Reason for Exception	Exception Message Number
6A0-3	None	
6A1-3	None	
6A2-3	None	
6A3-3	None	
6A4-3	None	
6A5-3	Note 1	Msg. 1
6A6-3	Note 1	Msg. 2
6A7-3	Note 1	Msg. 3, 4, 5
6A8-3	None	
6A9-3	None	
6AA-4	Note 1	Msg. 6, 7
	Note 4	Msg. 8, 9
6AB-3	None	
6AC-3	Note 1	Msg. 10, 11
6AE-3	None	
6AF-3	None	
6B0-3	None	
6B1-3	None	
6B2-3	Note 1	Msg. 12
6B3-3	None	
6BA-3	None	
6BB-3	None	
6BC-3	None	
6BD-4	None	

```
sg. 1 *SD0  F6A53 01  0065B2  0331
      SHLD GET CC1 ON SEEK INDICATING CUB BUT DID NOT CSW IS 000065E80C000000
       T
       S    F6A63
Msg. 2 *SD0  F6A63 01  0063B4  0331
      ERR CSW 0000663002000011 CSW SHLD BE 0000663072000011
      SNS 801000400000 SNS SHLD BE 80100040XXXX
       T
       S    F6A73
Msg. 3 *SD0  F6A73 01  0063B4  0331
      ERR CSW 0000650002000000 CSW SHLD BE 0000650072000009
      SNS 801000400000 SNS SHLD BE 80100040XXXX
Msg. 4 *SD0  F6A73 01  0063B4  0331
      ERR CSW 0000656802000000 CSW SHLD BE 0000656872000009
      SNS 801000400000 SNS SHLD BE 80100040XXXX
Msg. 5 *SD0  F6A73 01  0063B4  0331
      ERR CSW 000067200200000E CSW SHLD BE 000067207200000E
      SNS 801000400000 SNS SHLD BE 80100040XXXX
Msg. 6 *SD0  F6AA4 01  00635C  0331
      ERR-E R A S E  CAW 006458
      EXP CSW 000064787C000000 ACT CSW 000064780C000000
      EXP SNS 00000040XXXX ACT SNS 000000400000
      EXP READ 00C3000003 ACT READ 00C3000003
Msg. 7 *SD0  F6AA4 02  00658A  0331
      ERR-SPACE COUNT  CAW 0067F0
      EXP CSW 0000681872000003 ACT CSW 0000681802000003
      EXP SNS 80100040XXXX ACT SNS 801000400000
Msg. 8 *SD0  F6AA4 02  00658A  0331
      ERR-SPACE COUNT  CAW 006840
      EXP CSW 000068600C000000 ACT CSW 000068600F400000
      EXP SNS 00000040XXXX ACT SNS 080000400000
      EXP RD DATA 2000000002
      ACT RD DATA 2000000002
Msg. 9 *SD0  F6AA4 02  00658A  0331
      ERR-SPACE COUNT  CAW 006860
      EXP CSW 000068800E000000 ACT CSW 000068800E400018
      EXP SNS 08000040XXXX ACT SNS 080000400000
      EXP RD DATA E7E7E7E7E7E7E7E7E7E7E7E7E7E7E7E7E7E7E7E7E7FFFFXX7DFF
      ACT RD DATA E700000000000000000000000000000000000C300030000
Msg.10 *SD0  F6AC3 02  0064AE  0031
      ERR "IO EXPECT CC = 2 ACTUAL CC = 0
Msg.11 *SD0  F6AC3 02  006504  0331
      ERR-"IO FAILED TO STOP CHAIN ON WRITE-DID NOT PAD WITH ZEROS
Msg.12 *SD0  F6B23 01  005394  0331
      CSW 0056C00000200001 CSW SHLD BE 000056C00C000001 SNS 000000400000
      SNS SHLD BE 000000400X00 CAW 00005698
      ERR WHEN SCAN FO KEY DATA REMAINDER OF CYL, AFTER DEF TRK VIA 6TH SNS BYTE
      TO LOOP ERR TURN ON SSW0
```

2402-2403 Test

Test Section	Reason for Exception	Exception Message Number
501-8	None	
502-8	None	
503-8	None	
504-8	Note 1	Msg. 1
505-8	Note 1	Msg. 2, 3, 4, 5, 6
506-6	Note 4	Msg. 7
507-7	Note 1	Msg. 8, 9, 10
508-9	None	
50A-1	None	
50B-1	None	
50C-1	None	
50D-0	None	
50E-1	None	
50F-3	None	
510-1	None	
51B-5	None	

3420 OLTS Test

MACHINE	MODEL	TEST	VERSION	LEVEL	OPTIONS	ERRORS	COMMENTS
3420	A	0	0	NFE,PP(3)	None		
	B				None		
	C				None		
	D				None		
	E				RTN1	HANG	
	F				None		
	G				RTN2		
	H				None		
	I				RTN2		
	J				None		
	K				None		
	L				None		
	M				None		
	N				None		
	O				None		
	P				None		
	Q				None		
	R				None		
	S				None		
	T				None		

3420 OLTS failures

3420 Test E Routine 1-timing dependency failure
HANGS SYSTEM DO NOT RUN

3420 Test G Routine 2-timing dependency failure

3420 OLTS Test

RTN1
None
RTN2
None
None
None
None

```
Msg. 1 *SD0  F5048 29  0064F4  0181
        MSG F COND CD ERR  NRZI
        SIO 006672  CAW 006CF0
        CCW 04006E18 20000006
        ACT CC 1  EXP CC 0
        ACT INT STS 0400-EXP INT STS 0000
        CSW
        EXP FIN STS  0000
       SNS LINK ADR  006BB2

Msg. 2 *SD0  F5055 32  00695C  0181
        ACT INT STS  5000-EXP INT STS 2400
       MSG 45

Msg. 3 *SD0  F5055 32  0064F4  0181
        MSG F COND CD ERR  NRZI
        SIO 06672  CAW 006DE8
        CCW 04006E30 20000006
        ACT CC 1  EXP CC 0
        ACT INT STS  0400-EXP INT STS 0000
        CSW
        EXP FIN STS  0000
       SNS LINK ADR  006966
       SNS 00  00  00  00  00  00

Msg. 4 *SD0  F5055 34  006AC8  0181
        ACT INT STS  3000-EXP INT STS 3400
       MSG 51

Msg. 5 *SD0  F5055 34  0064F4  0181
        MSG F COND CD ERR  NRZI
        SIO 006672  CAW 006DF8
        CCW 04006E30 20000008
        ACT CC 1  EXP CC 0
        ACT INT STS  0400-EXP INT STS 0000
        CSW
        EXP FIN STS  0000
       SNS LINK ADR  006AD2
       SNS 00  00  00  00  00  00

Msg. 6 *SD0  F5055 35  006B70  0181
        ACT INT STS  0400-EXP INT STS 1400
       MSG 53
       SNS 00  44  03  00  00  00

Msg. 7 *SD0  F5066 44  0064F4  0180
        MSG E STATUS ERROR NRZI
        SIO 006A2A  CAW 006D28
        CCW 0C000000 20000004
        ACT CC 0  EXP CC 0
```

```
        ACT INT STS      -EXP INT STS 0000
        CSW 00006D30 0C010003
        EXP FIN STS  0C20


Msg. 8    S    F5077
          *SD0  F5077 51  006A54  0182
          MSG 74
          SNS 00  44  03  00  00  00

Msg. 9    *SD0  F5077 52 0064F4  0182
          MSG E STATUS ERROR NRZI
          SIO 006B50  CAW 006CE8
          CCW 01000000 20001FF0
          ACT CC 0  EXP CC 0
          ACT INT STS      -EXP INT STS 1000
          EXP FIN STS  0000

Msg. 10   *SD0  F5077 52  006B6C  0182
          MSG 76
          SNS 00 44 03 00 00 00

          UNITS RUN 0182  XXXX  XXXX  XXXX  XXXX  XXXX  XXXX  XXXXNRZI
```

2703 Test

Test Section	Reason for Exception	Exception Message Number
C21	Note 1	Msg. 1, 2
C2B	None	

Msg. 1 *SD0 EC211 01 005BAC 0030
PRELIMINARY I/F CHECK WITH TIO, NOP, SAD.
TIO DID NOT STORE CSW
LINK 005354
CC ACT 0 CC EXP 1

Msg. 2 *SD0 EC211 06 005BAC 0030
TEST INITIAL STATUS CONDITIONS
SENSE COMMAND ACCEPTED, EXPECTING CONTROL UNIT BUSY
LINK 005906 CAW 00 0065D0
CC ACT 0 CC EXP 1

CCW SENSE ADDR 006414 FLAGS=SLI BYTE CT=01

2540 Test

Test Section	Reason for Exception	Exception Message Number
804-1	None	
810-3	Note 1	Msg. 1, 2
811-2	None	
819-0	None	
820-3	Note 1	Msg. 3, 4
821-3	None	
822-0	None	

```
Msg. 1 *SD0  F8103 09  0063B2  000C
        TST 0250
        ADR 006AB4 LINK
        ADR 0061AC SIO
        CAW 006D08
        CCW 02006740 00000050
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 10000000 EXP
        CSW 00000000 04000000 ACT
        CSW 00000000 04000000 EXP
        SNS 00 EXP
        SNS 00 ACT
        SET SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP

Msg. 2 *SD0  F8103 10  0063B2  000C
        TST 0290
        ADR 006B08 LINK
        ADR 0061AC SIO
        CAW 006D10
        CCW 0C006740 00000001
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 14000000 EXP
        CSW 00000000 04000000 ACT
        SNS 00 EXP
        SNS 00 ACT
        SET SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP

       *SD0  F8103 11  0063B2  000C
        TST 0350
        ADR 006B64 LINK
        ADR 0061AC SIO
```

2703 Test

Test Section	Reason for Exception	Exception Message Number
C21	Note 1	Msg. 1, 2
C2B	None	

Msg. 1 *SD0 EC211 01 005BAC 0030
PRELIMINARY I/F CHECK WITH TIO, NOP, SAD.
TIO DID NOT STORE CSW
LINK 005354
CC ACT 0 CC EXP 1

Msg. 2 *SD0 EC211 06 005BAC 0030
TEST INITIAL STATUS CONDITIONS
SENSE COMMAND ACCEPTED, EXPECTING CONTROL UNIT BUSY
LINK 005906 CAW 00 0065D0
CC ACT 0 CC EXP 1

CCW SENSE ADDR 006414 FLAGS=SLI BYTE CT=01

2540 Test

Test Section	Reason for Exception	Exception Message Number
804-1	None	
810-3	Note 1	Msg. 1, 2
811-2	None	
819-0	None	
820-3	Note 1	Msg. 3, 4
821-3	None	
822-0	None	

```
Msg. 1 *SD0  F8103 09  0063B2  000C
        TST 0250
        ADR 006AB4 LINK
        ADR 0061AC SIO
        CAW 006D08
        CCW 02006740 00000050
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 10000000 EXP
        CSW 00000000 04000000 ACT
        CSW 00000000 04000000 EXP
        SNS 00 EXP
        SNS 00 ACT
        SET SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP

Msg. 2 *SD0  F8103 10  0063B2  000C
        TST 0290
        ADR 006B08 LINK
        ADR 0061AC SIO
        CAW 006D10
        CCW 0C006740 00000001
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 14000000 EXP
        CSW 00000000 04000000 ACT
        SNS 00 EXP
        SNS 00 ACT
        SET SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP

       *SD0  F8103 11  0063B2  000C
        TST 0350
        ADR 006B64 LINK
        ADR 0061AC SIO
```

```
        CAW 006D08
        CCW 02006740 00000050
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 14000000 EXP
        CSW 00000000 04000000 ACT
        SNS 00 EXP
        SNS 00 ACT
        SET SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP

Msg. 3 *SD0  F8203 06  006388  00DD
        TST 0150
        ADR 006878 LINK
        ADR 0061AE SIO
        CAW 006F58
        CCW F50066D8 20000001
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 14000000 EXP
        CSW 00000000 04000000 ACT
        SNS 00 EXP
        SNS 00 ACT
        SET SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP

Msg. 4 *SD0  F8203 07  006388  000D
        TST 0210
        ADR 0068DA LINK
        ADR 0061AE SIO
        CAW 006F58
        CCW 410066D8 20000001
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 14000000 EXP
        CSW 00000000 04000000 ACT
        SNS 00 EXP
        SNS 00 ACT
        SET SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP
```

1403 Test

Test Section	Reason for Exception	Exception Message Number
F830	Note 1	Msg. 1, 2
F832	None	

```
Msg. 1 *SD0  F8304 06  006378  0010
        TST 0240
        ADR 006ADC LINK
        ADR 00619E SIO
        CAW 006EC0
        CCW 05006881 00000084
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 14000000 EXP
        CSW 00000000 14000000 ACT
        SNS 00 EXP
        SNS 00 ACT
        SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP

Msg. 2 *SD0  F8304 12  006378  0010
        TST 0480
        ADR 006D26 LINK
        ADR 00619E SIO
        CAW 006EC0
        CCW 0C006881 00000084
        CC  1 EXP
        CC  1 ACT
        CSW 00000000 14000000 EXP
        CSW 00000000 04000000 ACT
        SNS 00 EXP
        SNS 00 ACT
        SS 0 ON FOR LOOP ON SIO, SS 1 ON FOR TIO SIO LOOP
```

APPENDIX B: CP-67 ON-LINE TESTING USER'S GUIDE

PURPOSE

The purpose of this series of EXEC procedures is to provide the ability to run DM-controlled diagnostics or stand-alone programs, including I/O exercisers, in a CP-67 virtual machine, to permit on-line testing and exercising of I/O devices. This document describes only the operating procedures for the on-line testing system. Generation and maintenance of the system is described in Appendix B.

The programs to be run reside on disk within the system, and are called for execution using only a few simple commands. A command is provided to accomplish the attaching of I/O devices to the virtual machine in which the diagnostics are to be run. This permits conflicts between real and virtual unit addresses to be automatically avoided.

The objective has been to make on-line testing available to all CE's, regardless of their level of acquaintance with System/360 Model 67.

REQUIREMENTS

Use of this system requires that the CE have access to a terminal (2741 or equivalent) to control testing. The on-line testing system must have been previously generated under CP-67 (per the Generation and Maintenance Guide, Appendix C), and the CE must know the USERID and PASSWORD under which it was generated.

OPERATING PROCEDURES

Concepts

CP-67 is an operating system designed to run on a System/360 model 67. It is a time-sharing system; that is, a system which supports many users, one at a time, but services all users so quickly that each seems to have complete control of the system to himself. Each user communicates with the system from a terminal, such as a 2741. For each user in the system, CP-67 creates a virtual system, in other words, each user in the system thinks that he has a System/360 system which is entirely his own. The user can do anything with this virtual system that he could do with a real System/360, except for a very few restrictions. From his terminal, the user can type CP-67 commands to perform all of the console

functions of a real machine: store, display, ipl, system reset, psw restart, etc. He can also start and stop his virtual machine at will. CP-67 also makes the user's terminal appear to be the console 1052 of his virtual system.

The CE also has his own virtual System/360 for testing purposes. It is time-shared with all the other users of the system, and can run I/O diagnostics under DME, or run I/O exercisers such as Friend-3. The ability to use I/O exercisers on line will be found especially helpful in repairing I/O problems.

Just as a real System/360 has much more I/O than a 1052, a virtual system also has various associated virtual I/O devices. The actual card reader, card punch, printer and disk drives have their counterparts in the virtual 360. If a program in the virtual machine performs START I/Os and writes on device 00E, CP-67 intercepts the START I/O and places the data on disk to be printed later when a real printer in the computer room becomes available. In the same way, a deck of cards placed in the real reader in the computer room with a USERID card in front of it will instruct CP-67 to put the cards on disk and "feed" them to the virtual reader (address 00C) when a START I/O is issued to it in the virtual system. The virtual card punch operates in the same way; data punched on device 00D from a virtual machine is held by CP-67 on disk, and is punched on the real card punch when it is available. The two virtual disk drives are actually parts of disk packs maintained by CP-67; they appear to be "mini-disks" with only a few cylinders to the virtual system. Diagnostic programs and I/O exercisers are stored on one of them. The other one is an operating system called CMS. CMS will be used to prepare a set of diagnostics to be run. More about CMS later.

Thus far we have discussed a virtual system and its virtual I/O, which allow the user to run any program that he could run on a real System/360. There is one more thing that is needed to test a failing device. The system operator can issue a command to ATTACH a real I/O device to a CE's virtual machine (or any other virtual machine). When a device is attached, it becomes part of one virtual machine only. This is the way we will gain access to any I/O device that we wish to test.

RESTRICTIONS

There are a few things that cannot be done in a CP-67 virtual machine.

1. The diagnose instruction is not supported. This means that all CPU, channel, and core diagnostics will not function.

2. Self-modifying channel programs will not always run properly.

3. Time-dependent programs may not run properly. This is because the real system is being shared with other users. It will show in some diagnostics as unexpected condition codes on I/O instructions or failure to get control unit busy status when the diagnostic expects it. The section, "Messages and Printouts", in this guide shows some examples of the type of error printout that may occur from this cause. The CE should learn to recognize this type of message and not treat it as a hardware failure. Since it depends on the other activity in the system, it is difficult to predict exactly where this type of printout will occur.

4. DASD Diagnostics may not function properly when run on minidisks. To ensure correct operation, the DASD device to be exercised by the diagnostic test section should be dedicated in its entirety to the virtual machine by use of the ATTACH command.

LOGIN

The following is a sample LOGIN procedure. The lines that the CE must type are indicated by asterisks at the left. The first message is printed by CP-67 after communication is established with the system (by dialing the telephone number assigned and going to data mode if a dial-up terminal is used).

```
          cp-67 online    xdh65 qsyosu
*         login xxxxxx    (HIT THE ATTN KEY FIRST)
                (xxxxxx IS YOUR USERID)
                (USE ..login..OR..l..)
          ENTER PASSWORD:
*         yyyyyy      (YOUR UNIQUE PASSWORD)
          READY AT 11.01.52 ON 05/22/70
```

At this point the virtual System/360 is ready to use. CP-67 may now be asked to perform any console function on the system. To run any of the diagnostic programs or exercisers already stored on disk, a program called CMS must be used first. The CP-67 IPL command is used to ipl CMS.

CMS IPL

The virtual disk drive 190 has a copy of CMS on it. After logging in to CP-67, type...

ipl 190 (OR ..i 190..)

A message will be received when CMS is loaded...

CMS..VERSION x LEVEL y (date)

The commands can now be given to start a test run. The UNIT command is used to ask the system operator to attach the I/O device to be tested. The DIAGS command may then be used to select those diagnostics or stand-alone programs that are to be run and ipl them. These commands are described in the following paragraphs.

UNIT COMMAND

The unit command issues messages to the system operator to request him to ATTACH an I/O device to the virtual system so that the CE can test it. When the message confirming that the device has been attached is printed, hit the ATTN key to continue. To attach I/O device 230, type...

unit 230

The unit command will cause messages to be printed if a device is requested whose real address is the same as one of the virtual devices. For example....

If it is desired to attach the real printer 00E, by typing ... unit 00e ... CP-67 cannot attach this device with that address, because the virtual or spooling printer is 00E. In this case, 00E (the real printer) will be attached to the virtual system as 004, and can be addressed as 004. Thus, the CE need only give the real address of the device to be tested. The unit command will prevent any possible conflicts.

The attaching of multiple devices may also be requested with one UNIT command. If it is desired to test devices 380, 381, and 233, type....

unit 380 381 233

Unit addresses should always be typed as 3 digits.

DIAGS COMMAND

Execution of the DIAGS command results in the programs chosen being placed in the virtual card reader of the virtual system, and the reader being IPL'ed. For example, to load FRIEND, type...

```
diags friend
```

If you wish to run a DM-controlled diagnostic, for example section 504, type...

```
diags 504
```

DME will automatically be placed in front of section 504. The same results will occur if the following is typed...

```
diags dme 504
```

Multiple sections may be requested....

```
diags 501 502 503 509
```

or....

```
diags dme 501 502 503 509
```

IPL programs and DM-controlled diagnostics may be intermixed....

```
diags 601 603 friend 606
```

The preceding will generate the following virtual card deck....

```
DME,601,603,FRIEND,DME,606
```

Note that every group of sections requested has DME placed in front of it.

Pre-defined groups of diagnostic sections may also be requested. For example....

```
diags 2311
```

will load all of the 2311 disk drive diagnostics, with DME again automatically inserted in front.

Some of the groups of sections which may have been defined in your system are....

```
2311..... 2311 DISK DRIVE
2841..... 2841 CONTROL UNIT
2313..... 2314 DISK DRIVES
2314..... 2314 CONTROL UNIT
2400..... 24xx TAPE DRIVES
```

The naming of section groups, and the diagnostic sections and IPL programs present in the system are under the control of the CE responsible for maintenance of the on-line testing

system. He should be consulted for specific information.

PROGRAM CONTROL

The successful completion of the DIAGS command results in IPL'ing the virtual card reader of the system, overlaying CMS with the first program requested. Control of the virtual machine then passes to that program. The message formats from this point on are those of the program being run. If an unrecoverable software problem causes loss of control of the program,the entire card deck can be re-positioned to the start and re-IPL'ed by entering CP-67 (ATTN key) and typing....

```
c c
i c
```

If it is not desired to have the cards in the virtual reader saved in this way, the word NOSAV should be given as a parameter to the DIAGS command, e.g. ....

```
diags 601 602 nosav
```

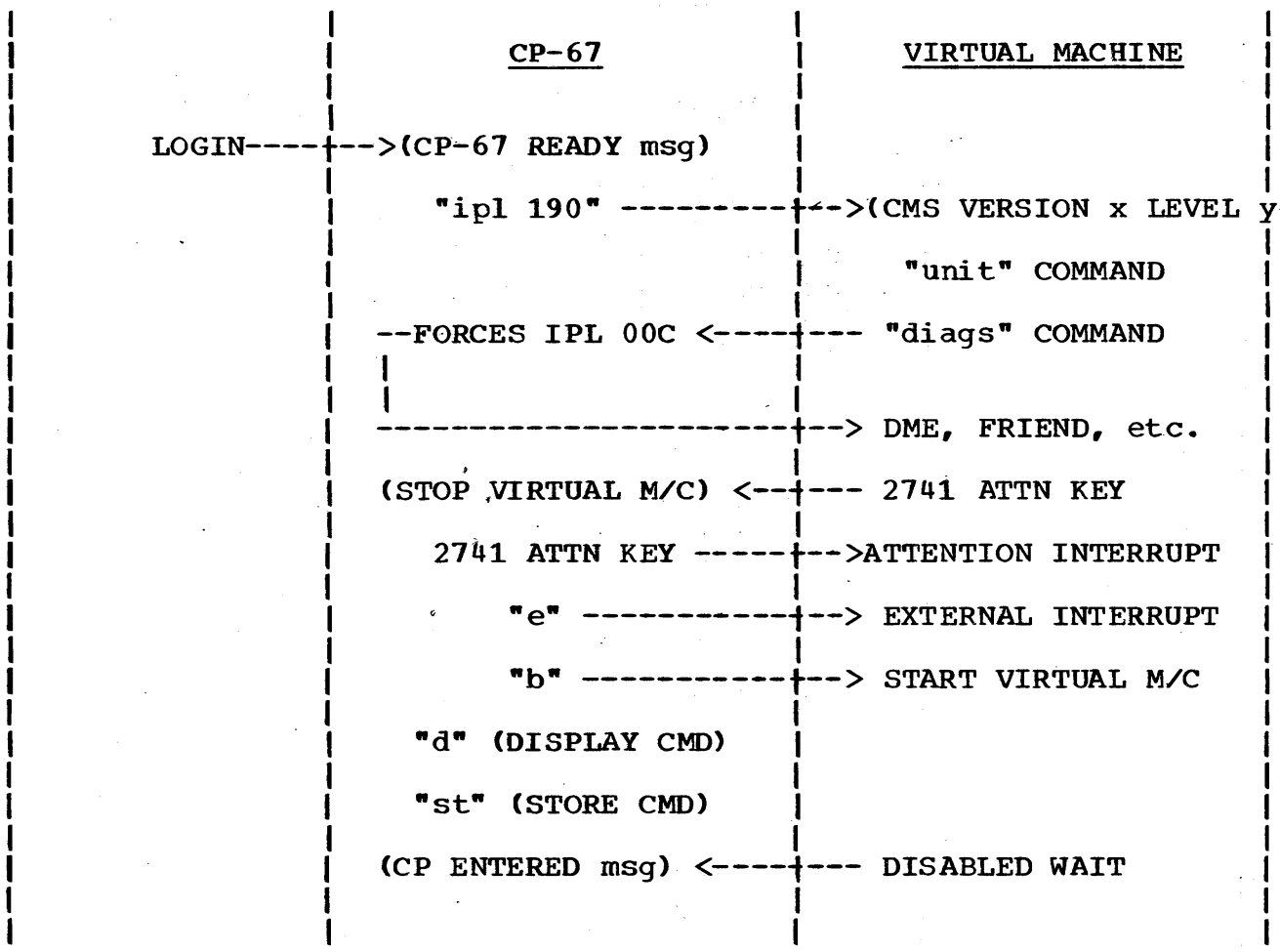
CONSOLE FUNCTIONS

A virtual machine can be started, stopped, and controlled in the same way as a real machine. The 2741 ATTN key is used to alternate control between CP-67 and the virtual machine's simulated 1052. For example, to generate a 1052 request key operation in the virtual machine....

1. Press the ATTN key once. This stops the virtual machine and passes control to CP-67.

2. Press the ATTN key again. This restarts the virtual machine and simulates a 1052 attention interrupt.

The following chart shows the various ways in which control is passed by the user between CP-67 and the program(s) in a virtual machine.....



The following chart shows the use of the CP-67 store and display commands.....

TO DISPLAY	TYPE
Main Storage addr xxxx (1 word)	d xxxx
Main Storage addresses xxxx to yyyy (nearest fullword) (xxxx and yyyy are hex addresses, 1 to 6 digits long)	d xxxx-yyyy
GPR n (n is gpr no.. 1 to 15)	d gn
GPRs n to m (n and m are 1 to 15)	d gn-m
FPR n	d yn
FPRs n to m	d yn-m
CURRENT PSW	d psw

TO STORE IN	TYPE
Main Storage addr xxxx Data hhhhhhhh (hhhhhhhh is hex data one or more fullwords separated by blanks)	st xxxx hhhhhhhh
GPRs starting at GPR n (each word into next higher GPR)	st gn hhhhhhhh hh..etc.
FPRs starting at FPR n	st yn hhhhhhhh hh..etc.
CURRENT PSW	st psw hhhhhhhh h..etc.

MESSAGES AND PRINTOUTS

As previously discussed, most printouts from this system are self-explanatory, or are documented in the literature provided for the programs which are being run in the virtual machine. An exception to this are the diagnostic printouts which result from time-dependencies within the code of CE diagnostics. In most cases these time-dependencies are deliberately created to test some function of the device under test. A couple of examples follow.....

```
 S F6096
*SD0  F6096 01  0056F6  0394
ERR HIO EXPECT CC = 2 ACTUAL CC = 0
*SD0  F6096 01  00574A  0394
ERR-HIO FAILED TO STOP CHAIN
*SD0  F6096 02  0058F8  0394
ERR HIO EXPECT CC = 2 ACTUAL CC = 0
*SD0  F6096 02  005950  0394
ERR-HIO FAILED TO STOP CHAIN ON WRITE-DID NOT PAD
  WITH ZEROS
T
```

The previous printout shows an attempt by the diagnostic to issue a HIO instruction while a channel program is running on device 0394. The expected condition code of 2 (burst operation stopped) was not returned. This occurred because of the way in which CP-67 schedules I/O operations. The channel program which the diagnostic was attempting to stop with a HIO had probably already terminated by the time the HIO instruction was executed. Other printouts may be seen in which expected control unit busy status fails to be reported to the virtual machine.

Ending interrupts may also occur at times other than those expected on a real machine, resulting in printouts like the following.....

```
*SD0  F5055 34  006AC8  0181
 ACT INT STS  3000-EXP INT STS 3400
MSG 51
```

Notice that the expected device end status failed to occur.

These are only two examples of the general type of printout to expect. Only a small percentage of diagnostic sections generate these printouts.

APPENDIX C:

CP-67 ON-LINE TESTING GENERATION AND MAINTENANCE GUIDE

PURPOSE

The purpose of this series of EXEC procedures is to provide the ability to run DM-controlled CE diagnostics or stand-alone programs, including I/O exercisers, in a CP-67 virtual machine, to permit on-line testing and exercising of I/O devices. This document describes the generation and maintenance procedures for the on-line test system. The necessary commands and operating procedures for actual device testing are described in a separate user's guide.

The objective of this package is to make on-line testing available to all CEs regardless of their level of acquaintance with CP-67.

Ease of operation has been provided by making all programs to be run resident on disk within the system, and by providing a straightforward command language to prepare diagnostics for execution. Commands are also provided to facilitate initial generation and subsequent maintenance of the on-line testing system.

REQUIREMENTS

To make use of this on-line testing system, a virtual machine must be defined in which to run it. A typical directory entry for this virtual machine follows. Some details of this entry are dependent on installation practices.

```
userid USER password,accounta,C,80
  CORE 256K
  UNIT 009,1052
  UNIT 00E,1403
  UNIT 00C,2540R
  UNIT 00D,2540P
  UNIT 190,2314,CMS190,000,053,RDONLY
  UNIT 191,2314,CPDSK5,108,117
  *EOU*
```

The P-disk space (unit 191) should be approximately 10 2314 cylinders.

OPERATING PROCEDURES

INITIAL GENERATION

After the preceding virtual machine definition has been entered into the user directory, the following procedure should be used to load the P-disk files of the on-line testing system....

1. LOGIN on the testing system. (See the on-line testing user's guide, Appendix B, if more information on LOGIN is needed.)

2. Issue the CP-67 command IPL 190 to load CMS.

3. If this is a new virtual machine with no previous P-disk files, issue the CMS command FORMAT P ALL.

4. Have a tape unit ATTACHed by the operator to the virtual machine as address 181, and mount the tape containing the on-line test system.

5. Issue the CMS command TAPE LOAD. At this point the on-line test system files have been loaded into the P-disk of the virtual machine.

DUMPIT COMMAND

The DUMPIT command, given to CMS in the on-line test system, will cause a report to be generated showing the status of the entire on-line test system at that time. The report is printed on the high speed printer of the real system, and consists of 4 sections....

1. A listing of filename and filetype for each file on the CMS P-disk. Certain filetypes have a special meaning to CMS. The filetype EXEC defines a file containing CMS command statements. Most of the functions of the on-line test system are provided by using the EXEC file capability of CMS. In addition, some filetypes have been given special meanings within the on-line test system. The filetype SECT is assumed to indicate that the file in question is a diagnostic section which must be run under control of a diagnostic monitor program. The filetype IPL must be applied to any program which can be ipl'ed from the virtual machine's card reader. This classification includes DME and I/O exercisers. DME itself must have the filename and filetype DME IPL in order for it to be automatically accessed by the testing system. To create a copy of DME IPL or FRIEND IPL punch a card deck of

the program and follow the directions in the NEWFILES command.

2. The second part of the DUMPIT report is a listing of the configuration cards of DME. This printout will be used to assist in changing the DME configuration using the SYS command, which will be described later.

3. The third report from DUMPIT lists the first (ESD) card of each SECT file on the P-disk. This provides a reference to the revision level, part number, and EC level of all diagnostic sections.

4. The final part of the DUMPIT report is a printout of the contents of all the EXEC files on the P-disk.

A current copy of the DUMPIT report should be kept on hand to indicate the status of the on-line test system, and to aid in modifying the system.

SYS COMMAND

The command SYS is used to initiate modifications to DME in the on-line test system. It starts a CMS EDIT of DME IPL, locates the DMIO card, and then unlocks the keyboard to allow EDIT commands to be entered. N.B. the user should verify that DME IPL contains only one DMIO card for this command to function correctly. The edit commands which should be used are....

1. LOCATE Command

The LOCATE command is used to scan the file being edited until a specified character string is found. The string can be any combination of characters and spaces. It may be positioned anywhere on the line. The LOCATE command has the following format....

```
L $string$
```

where string is the sequence of characters to be located. The string must be enclosed in two unique characters which do not appear in the data being searched (dollar signs were used here). When the string is located, the entire line containing the string will be printed.

2. CHANGE Command

The CHANGE command is used to alter the data on the current line of the file being edited. It has the

following format....

```
C $string1$string2$
```

The character string, string1, will be replaced by string2. The changed line will be printed.

3. PRINT Command

To print the current line, type p. To print x lines starting with the current one, type p x.

4. UP Command

To move the pointer up x lines in the file, type u x.

5. NEXT Command

To move the pointer down x lines in the file, type n x.

6. FILE Command

When all editing is completed, type file.

For further information on the above commands, consult the Control Program-67/Cambridge Monitor System User's Guide. Note that while other commands are available in EDIT to make changes to a data record, only the CHANGE command should be used under SYS. This will protect the data in columns 1 to 12 and 73 to 80 of the configuration cards.

A sample printout of the SYS operation follows....

```
sys
USE EDIT COMMANDS "C","L", AND "FILE" TO MODIFY DME TABLES.
DEFAULT TABS SET.
EDIT:
TXT  -      DMIO      001F001F
1 $230$
TXT        UDT4 6A,,0230-0237/6A,,0340-0347/
c $237$234$
TXT        UDT4 6A,,0230-0234/6A,,0340-0347/
1 $udtd$
TXT        UDTD
c $udtd$udtd 50,28,0380-0385/$
TXT        UDTD 50,28,0380-0385/
file
END DME MODIFY.
R; T=0.78 15.59.26
```

An inspection of the DUMPIT printout of the DME tables in the on-line test system will show four unusual unit

addresses. The unit addresses 0001,0002,0003, and 0004 should be used when describing the real I/O devices 0009 (or 001F), 000C, 000D, and 000E respectively. The UNIT command will cause them to be attached with these special unit addresses to prevent conflict with the virtual devices at these addresses.

In addition, UDTs should be provided for the virtual devices 0009 (or 001F), 000C, and 000E to permit DME to use these virtual devices as loader, input, and output devices.

The DMIO card should also reflect the address of the virtual 1052, to force the initial messages from DME to print at the terminal.

The SRT card should describe a 256K model 65. If it should be decided to replace DME with a new deck using the NEWFILES command (described below), the following should be checked....

1. Some DME decks have two groups of configuration cards at the end. One of these decks must be removed in this case. The DUMPIT printout will show the presence of the second deck if this is not done, and the SYS command will not function properly.

2. All of the SRT, SDT, and UDT cards should have a sequence number in column 16, i.e. UDT8, UDT9, UDTA, etc. This will facilitate editing with the LOCATE command under SYS.

NEWFILES COMMAND

To add new files to the system, or replace existing ones, issue the command NEWFILES. This command will automatically purge any files which may have been left in the virtual reader by the DIAGS command, and will request the files to be added to be placed in the real card reader. The deck should be preceded by a USERID card (columns 1-2 punched ID, and columns 10-17 punched with your USERID). Each file within the deck to be read should be preceded by an OFFLINE READ card (punched OFFLINE READ filename filetype starting in column 1). For example, to add or replace the diagnostic 50a, a card punched OFFLINE READ 50A SECT should be used. When the message *CARDS HAVE BEEN READ* has been printed at the terminal, hitting the ATTN key on the 2741 will cause the files to be read into the P-disk, replacing any files of the same name which are already there.

DEFINING GROUPS OF SECTIONS

Groups of diagnostic sections to be called by a single name in the DIAGS command must have an EXEC created defining the group. Models of this type of EXEC can be found in the DUMPIT printout. If the user is proficient with EDIT, groups can be changed or added using it. Otherwise, it is suggested that the group EXEC be punched into cards with an OFFLINE READ groupname EXEC card in front of it, and the NEWFILES command be used to insert the EXEC. Group EXECs which may be found in the initial system are 2311, 2841, 2313 and 2314.

TERMEX TERMINAL EXERCISERS

TERMEX is a collection of exercisers designed for a 2741 terminal. It is essentially self-documenting. To obtain a user's guide for TERMEX, type....

```
TERMEX
```

When a pattern name and count is requested, give any reply. TERMEX will respond with instructions on obtaining a list of available tests.

PRINTOUTS

Most of the printouts have been discussed. They are largely self-explanatory. It should be stressed that a DUMPIT printout should be on hand at all times for reference purposes.

INDEX

GH20-0858-1

Control Program-67/Cambridge Monitor System (CP-67/CMS) V. 3.1 Printed in U.S.A. GH20-0858-1

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

IBM

Technical Newsletter

Base Publ. No. GH20-0858-1

This Newsletter No. GN20-2497

Date January 24, 1972

Previous Newsletter Nos. None

Control Program–67/Cambridge Monitor System
(CP–67/CMS) Version 3.1
Hardware Maintainability Guide
Program Number 360D-05.2.005

This Technical Newsletter, a part of Version 3, Modification Level 1, of Control Program–67/Cambridge Monitor System, provides replacement pages for the subject manual. These replacement pages remain in effect for subsequent versions and modifications unless specifically altered. Pages to be inserted and/or removed are listed below.

Pages

Contents/Figures
13, 14
13.1 (added)
21, 22
27, 28
35
35.1–35.4 (added)
36
41, 42

Minor additions and changes have been made to promote program support information on IBM 3420 Magnetic Tape Unit as well as information on OLTSEP diagnostics for the 3420.

Vertical rules in the left margin indicate changes.

Please file this cover letter at the back of the manual to provide a record of changes.

IBM Corporation, Technical Publications Dept., 1133 Westchester Avenue, White Plains, N. Y. 10604

Printed in U.S.A.

IBM Technical Newsletter

This Newsletter No.	GN20-2620
Date	May 15, 1973
Base Publication No.	GH20-0858-1
File No.	(CP-67/CMS Version 3.2)
Previous Newsletters	GN20-2497

Control Program-67/Cambridge Monitor System
(CP-67/CMS) Version 3.2
Program Number 360D-05.2.005
Hardware Maintainability Guide

This Technical Newsletter, a part of Version 3, Modification Level 2 of Control Program-67/Cambridge Monitor System (CP-67/CMS), provides replacement pages for your publication. These replacement pages remain in effect for subsequent CP-67/CMS releases unless specifically altered. Pages to be removed and/or inserted are listed below.

Title page, 2	1-4
Preface	13,14-14.1
Contents	35.4-36
Figures	51,52-52.1

Changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

SUMMARY OF AMENDMENTS

This TNL reflects:

- The possibility of encountering errors when running diagnostic test sections on a virtual machine's minidisk.
- Other minor technical corrections.

Note: Please file this cover letter at the back of your publication to provide a record of changes.

IBM Corporation, VM/370 Publications, 24 New England Executive Park, Burlington, Massachusetts 01803

Printed in U.S.A.

Please comment on the usefulness and readability of this publication, suggest additions and deletions, and list specific errors and omissions (give page numbers). All comments and suggestions become the property of IBM. If you wish a reply, be sure to include your name and address.

COMMENTS

fold fold

fold fold

- Thank you for your cooperation. No postage necessary if mailed in the U.S.A. FOLD ON TWO LINES, STAPLE AND MAIL.

YOUR COMMENTS PLEASE...

Your comments on the other side of this form will help us improve future editions of this publication. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material.

fold fold

FIRST CLASS
PERMIT NO. 1359
WHITE PLAINS, N. Y.

BUSINESS REPLY MAIL
NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY...

IBM Corporation
1133 Westchester Avenue
White Plains, N.Y. 10604

Attention: Technical Publications

fold fold

Control Program-67/Cambridge Monitor System (CP-67/CMS) V. 3.1 Printed in U.S.A. GH20-0858-1

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
[U.S.A. only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]