

KMS-BD

KMS-BD DCLT
CZKMSAO

AH-S978A-MC
FICHE 1 OF 2

MAY 1983
COPYRIGHT © 1983
MADE IN USA



A microfiche card containing a grid of approximately 100 small frames. Each frame contains a different view or data point related to the project identified in the header. The frames are arranged in a regular grid pattern across the card.

KMS-BD

KMS-BD DCLT
CZKMSAO

AH-S978A-MC
FICHE 2 OF 2

MAY 1983
COPYRIGHT © 1983
MADE IN USA



Microfiche grid containing multiple frames of data, including text and barcodes.

1
2
3

.TITLE CZKMSAO KMS11-BD/BE DCLT

.REM 8

IDENTIFICATION

PRODUCT CODE: AC-S977A-MC
PRODUCT NAME: CZKMSAO KMS-BD DCLT
PRODUCT DATE: 01-APR-83
MAINTAINER: MERRIMACK DIAGNOSTIC ENGINEERING
AUTHOR: GLORIA MEREDITH

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1983 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL	PDP	UNIBUS	MASSBUS
DEC	DECUS	DECTAPE	

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 3

REVISION HISTORY:

REV ---	DATE ----	AUTHOR -----	REASON -----
A	01-APR-83	G. MEREDITH	ORIGINAL ISSUE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 4

.SBTTL PROGRAM DOCUMENT

TABLE OF CONTENTS

- 1.0 GENERAL INFORMATION
 - 1.1 PROGRAM ABSTRACT
 - 1.2 SYSTEM REQUIREMENTS
 - 1.3 RELATED DOCUMENTS AND STANDARDS
 - 1.4 DIAGNOSTIC HIERARCHY PREREQUISITES
 - 1.5 ASSUMPTIONS - RESTRICTIONS
- 2.0 OPERATING INSTRUCTIONS
 - 2.1 COMMANDS
 - 2.2 SWITCHES
 - 2.3 FLAGS
 - 2.4 HARDWARE QUESTIONS
 - 2.5 DATA COMM. LINK TEST COMMANDS
 - 2.5.1 MESSAGE COMMANDS
 - 2.5.2 LINE SELECTION COMMANDS
 - 2.5.3 OPERATOR ASSISTANCE COMMANDS
 - 2.5.4 RUN COMMANDS
 - 2.5.5 PRINT COMMANDS
 - 2.5.6 DEFAULTS
 - 2.6 QUICK STARTUP PROCEDURE
- 3.0 ERROR INFORMATION
 - 3.1 TYPES OF ERROR MESSAGES
 - 3.2 SPECIFIC ERROR MESSAGES
 - 3.2.1 COMMAND LINE INTERPRETER ERRORS
 - 3.2.2 DCLT ERROR MESSAGES
 - 3.2.3 DEVICE ERROR MESSAGES
- 4.0 PERFORMANCE AND PROGRESS REPORTS
 - 4.1 PRINTING EVENT LOG
 - 4.2 OPERATOR STATUS MESSAGES
 - 4.3 PRINTING LINE STATUS
- 5.0 DEVICE INFORMATION TABLES

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 5

- 6.0 MODE AND MESSAGE DESCRIPTIONS
 - 6.1 MODE DESCRIPTIONS
 - 6.1.1 TRANSMIT MODE
 - 6.1.2 RECEIVE MODE
 - 6.1.3 PASSIVE MODE
 - 6.1.4 ACTIVE MODE
 - 6.1.5 DOWN LINE LOAD
 - 6.1.6 TALK AND LISTEN
 - 6.1.6.1 TALK MODE
 - 6.1.6.2 LISTEN MODE
 - 6.1.7 MAINTENANCE "LOOP" MODES
 - 6.1.8 MODE SUMMARY TABLE
 - 6.2 MESSAGE DESCRIPTIONS
- 7.0 OTHER INFORMATION
 - 7.1 INTERFACING TO AN "ITEP" NODE
 - 7.2 TROUBLESHOOTING HINTS
 - 7.3 EXAMPLES OF COMMANDS
 - 7.4 THINGS TO WATCH OUT FOR

1.0 GENERAL INFORMATION

1.1 PROGRAM ABSTRACT

THIS DCLT (DATA COMMUNICATION LINK TEST) PROGRAM IS MEANT TO PROVIDE FIELD SERVICE WITH A TOOL TO MAINTAIN KMS11-BD/BE TO DDCMP POINT-TO-POINT COMMUNICATION LINKS. THIS DCLT PROGRAM WILL PROVIDE THE COVERAGE NECESSARY TO DETECT FAILURES IN THE COMPUTER EQUIPMENT, THE COMMUNICATION LINK, OR THE MODEM.

THIS DIAGNOSTIC HAS BEEN WRITTEN FOR USE WITH THE DIAGNOSTIC RUNTIME SERVICES SOFTWARE (SUPERVISOR). THESE SERVICES PROVIDE THE INTERFACE TO THE OPERATOR AND TO THE SOFTWARE ENVIRONMENT. THIS PROGRAM CAN BE USED WITH XXDP+, ACT, SLIDE AND PAPER TAPE. FOR A COMPLETE DESCRIPTION OF THE RUNTIME SERVICES, REFER TO THE XXDP+ USER'S MANUAL (CHOUS?.SEQ WHERE ? IS REV. LEVEL OF THE MANUAL). THERE IS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES IN SECTION 2 OF THIS DOCUMENT.

1.2 SYSTEM REQUIREMENTS

IN ORDER TO RUN THE KMS11-BD/BE DCLT PROGRAM, THE FOLLOWING MINIMUM HARDWARE IS REQUIRED:

- A PDP-11 CPU
 - MINIMUM OF 24K WORDS OF MEMORY
 - A WORKING, LINE OR REAL-TIME CLOCK
 - A CONSOLE TERMINAL
 - ANY XXDP+ SUPPORTED LOAD MEDIA
 - A KMS11-BD/BE WITH H3256 TURNAROUND CONNECTOR
- THE KMS11-BD/BE COMPONENT PARTS ARE:
- 1 KMC11-B MICROPROCESSOR
 - 1 DMS11-DA LINE UNIT
 - 1 H317-M DISTRIBUTION PANEL
 - 1 DM11-BA MODEM CONTROL MULTIPLEXER

1.3 RELATED DOCUMENTS AND STANDARDS

- KMS11-BD/BE OPTION DESCRIPTION YM-C174C-00
- KBC11 MICROPROCESSOR USERS MANUAL EK-KMC11-0P
- KMC11-B PROGRAMMERS MANUAL YM-P093C-00
- KMC11-B UNIBUS MICROPROCESSOR
OPTION DESCRIPTION YM-C093C-00
- DM11-BA MODEM CONTROL MULTIPLEXER YM-C138C-00
- DMS11-D/DA SYNCHRONOUS LINE UNIT
OPTION DESCRIPTION YM-C186C-00
- XXDP+ USER'S MANUAL (CHOUS?.SEQ WHERE ? IS THE REV. LEVEL OF
THE MANUAL - "C" IS THE CURRENT REV.).

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 7

1.4 DIAGNOSTIC HIERARCHY PREREQUISITES

THE GOAL OF THE DATA COMM. LINK TEST PROGRAM IS TO TEST THE COMMUNICATION LINK AND THEREFORE ASSUMES THAT THE CPU'S, CLOCKS, THE KMS11-BD/BE USED AT ONE END OF THE LINK AND THE DDCMP SUPPORTING COMMUNICATIONS DEVICE USED AT THE OTHER END OF THE LINK HAVE ALREADY BEEN TESTED.

IF NO LINE OR REAL-TIME CLOCK IS FOUND, THE PROGRAM WILL CONTINUE BUT WILL NOT EXIT FROM ROUTINES USED TO TIME WAITS FOR COMPLETION OF A PROCESS AND WILL NOT REPORT ELAPSED TIME IN SUCH TIMING LOOPS. IN ADDITION, THE EVENT LOG WILL CONTAIN A ZERO EVENT TIME FOR ALL EVENTS LOGGED.

IT IS NOT THE INTENTION OF A DATA COMM. LINK TEST PROGRAM TO TEST THE KMS11-BD/BE BUT TO TEST THE COMMUNICATION LINK TO WHICH IT IS CONNECTED.

SOME OF THE DIAGNOSTICS THAT COULD BE RUN IF THE KMS11-BD/BE LOOKS BAD:

CZKMB - KMC11 TEST - PART 1	YM-2093D-A (MICROPROCESSOR)
CZKMC - KMC11 TEST - PART 2	YM-2093D-B (MICROPROCESSOR)
CZKMD - DMS11-DA STATIC TEST	YM-2136D-1 (LINE UNIT)
CZKME - DMS11-DA DYNAMIC TEST	YM-2136D-2 (LINE UNIT)
CZKMF - DMS-11- DM11 INTERACTION TEST	(LINE UNIT-MODEM CONTROL INTERACTION)
CZKMG - DM11-BA MODEM CONTROL TEST	YM-2138D-A (MODEM CONTROL)

1.5 ASSUMPTIONS - RESTRICTIONS

IT IS ASSUMED THAT THE KMS11-BD/BE HAS BEEN TESTED USING THE PREREQUISITE DIAGNOSTICS. THE OPERATOR SHOULD HAVE READ THE USER DOCUMENTATION PORTION OF THE LISTING TO FAMILIARIZE HIMSELF WITH THE COMMANDS AND CAPABILITIES AVAILABLE UNDER THE DIAGNOSTIC SUPERVISOR AND DCLT.

2.0 OPERATING INSTRUCTIONS

THIS SECTION CONTAINS A BRIEF DESCRIPTION OF THE RUNTIME SERVICES.
FOR DETAILED INFORMATION, REFER TO THE XXDP+ USER'S MANUAL (CHOUS).

2.1 COMMANDS

THERE ARE ELEVEN LEGAL COMMANDS FOR THE DIAGNOSTIC RUNTIME SERVICES
(SUPERVISOR). THIS SECTION LISTS THE COMMANDS AND GIVES A VERY
BRIEF DESCRIPTION OF THEM. THE XXDP+ USER'S MANUAL HAS MORE DETAILS.

COMMAND	EFFECT
START	START THE DIAGNOSTIC FROM AN INITIAL STATE
RESTART	START THE DIAGNOSTIC WITHOUT INITIALIZING
CONTINUE	CONTINUE AT TEST THAT WAS INTERRUPTED (AFTER ^C)
PROCEED	CONTINUE FROM AN ERROR HALT
EXIT	RETURN TO XXDP+ MONITOR (XXDP+ OPERATION ONLY!)
ADD	ACTIVATE A UNIT FOR TESTING (ALL UNITS ARE CONSIDERED TO BE ACTIVE AT START TIME)
DROP	DEACTIVATE A UNIT
PRINT	PRINT STATISTICAL INFORMATION (IF IMPLEMENTED BY THE DIAGNOSTIC - SECTION 4.0)
DISPLAY	TYPE A LIST OF ALL DEVICE INFORMATION
FLAGS	TYPE THE STATE OF ALL FLAGS (SEE SECTION 2.3)
ZFLAGS	CLEAR ALL FLAGS (SEE SECTION 2.3)

A COMMAND CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. SO
YOU MAY, FOR EXAMPLE, TYPE "STA" INSTEAD OF "START".

2.2 SWITCHES

THERE ARE SEVERAL SWITCHES WHICH ARE USED TO MODIFY SUPERVISOR OPERATION. THESE SWITCHES ARE APPENDED TO THE LEGAL COMMANDS. ALL OF THE LEGAL SWITCHES ARE TABULATED BELOW WITH A BRIEF DESCRIPTION OF EACH. IN THE DESCRIPTIONS BELOW, A DECIMAL NUMBER IS DESIGNATED BY "DDDDD".

SWITCH	EFFECT
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. THE KMS11-BD/BE DCLT PROGRAM CONTAINS ONLY ONE TEST.
/PASS:DDDDD	EXECUTE DDDDD PASSES (DDDDD = 1 TO 64000)
/FLAGS:FLGS	SET SPECIFIED FLAGS. FLAGS ARE DESCRIBED IN SECTION 2.3.
/EOP:DDDDD	REPORT END OF PASS MESSAGE AFTER EVERY DDDDD PASSES ONLY. (DDDDD = 1 TO 64000)
/UNITS:LIST	TEST/ADD/DROP ONLY THOSE UNITS SPECIFIED IN THE LIST. THE KMS11-BD/BE DCLT PROGRAM WILL TEST ONLY ONE UNIT

EXAMPLE OF SWITCH USAGE:

START /PASS:1000/EOP:100

THE EFFECT OF THIS COMMAND WILL BE: 1) TESTS 1 THROUGH 5 WILL BE EXECUTED, 2) ALL UNITS WILL TESTED 1000 TIMES AND 3) THE END OF PASS MESSAGES WILL BE PRINTED AFTER EACH 100 PASSES ONLY. A SWITCH CAN BE RECOGNIZED BY THE FIRST THREE CHARACTERS. YOU MAY, FOR EXAMPLE, TYPE "/TES:1-5" INSTEAD OF "/TESTS:1-5".

BELOW IS A TABLE THAT SPECIFIES WHICH SWITCHES CAN BE USED BY EACH COMMAND.

	TESTS	PASS	FLAGS	EOP	UNITS
START	X	X	X	X	X
RESTART	X	X	X	X	X
CONTINUE		X	X	X	
PROCEED			X		
DROP					X
ADD					X
PRINT					
DISPLAY					X
FLAGS					
ZFLAGS					
EXIT					

2.3 FLAGS

FLAGS ARE USED TO SET UP CERTAIN OPERATIONAL PARAMETERS SUCH AS LOOPING ON ERROR. ALL FLAGS ARE CLEARED AT STARTUP AND REMAIN CLEARED UNTIL EXPLICITLY SET USING THE FLAGS SWITCH. FLAGS ARE ALSO CLEARED AFTER A START COMMAND UNLESS SET USING THE FLAG SWITCH. THE ZFLAGS COMMAND MAY ALSO BE USED TO CLEAR ALL FLAGS. WITH THE EXCEPTION OF THE START AND ZFLAGS COMMANDS, NO COMMANDS AFFECT THE STATE OF THE FLAGS; THEY REMAIN SET OR CLEARED AS SPECIFIED BY THE LAST FLAG SWITCH.

FLAG	EFFECT
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	"BELL" ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION) THE KMS11-BD/BE DCLT REQUIRES MANUAL INTERVENTION TO SELECT A LINK TO TEST.
ISR	INHIBIT STATISTICAL REPORTS (DOES NOT APPLY TO DIAGNOSTICS WHICH DO NOT SUPPORT STATISTICAL REPORTING)
IDR	INHIBIT PROGRAM DROPPING OF UNITS
ADR	EXECUTE AUTODROP CODE
LOT	LOOP ON TEST
EVL	EXECUTE EVALUATION (ON DIAGNOSTICS WHICH HAVE EVALUATION SUPPORT)

*ERROR MESSAGES ARE DESCRIBED IN SECTION 3.1

SEE THE XXDP+ USER'S MANUAL FOR MORE DETAILS ON FLAGS. YOU MAY SPECIFY MORE THAN ONE FLAG WITH THE FLAG SWITCH. FOR EXAMPLE, TO CAUSE THE PROGRAM TO LOOP ON ERROR, INHIBIT ERROR REPORTS AND TYPE A "BELL" ON ERROR, YOU MAY USE THE FOLLOWING STRING:

/FLAGS:LOE:IER:BOE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 11

2.4 HARDWARE QUESTIONS

WHEN A DIAGNOSTIC IS STARTED, THE RUNTIME SERVICES WILL PROMPT THE USER FOR HARDWARE INFORMATION BY TYPING "CHANGE HW (L) ?" YOU MUST ANSWER "Y" AFTER A START COMMAND UNLESS THE HARDWARE INFORMATION HAS BEEN 'PRELOADED' USING THE SETUP UTILITY (SEE CHAPTER 6 OF THE XXDP+ USER'S MANUAL). WHEN YOU ANSWER THIS QUESTION WITH A "Y", THE RUNTIME SERVICES WILL ASK FOR THE NUMBER OF UNITS (IN DECIMAL).

THE KMS11-BD/BD COMM. LINK TEST PROGRAM WILL NOT USE MORE THAN ONE UNIT. FOR THE KMS11-BD/BE THE HARDWARE INFORMATION REQUESTED WILL BE:

UNITS (D) ? 1<CR>

UNIT 0

FULL DUPLEX ONLY

DEVICE CSR ADDRESS : (0) 164100 ?

INTERRUPT VECTOR ADDRESS: (0) 400 ?

INTERRUPT PRIORITY: (0) 5 ?

DM11-BA CSR ADDRESS (0) 164110 ?

AFTER THESE HARDWARE QUESTIONS ARE ANSWERED THE PROGRAM ENTERS THE DCLT COMMAND LEVEL AND PRINTS THE FOLLOWING MESSAGE. THE LAST LINE OF THE MESSAGE IS THE PROMPT FOR A DCLT> COMMAND.

THIS IS DCLT. TYPE "H" OR "?" FOR DETAILS

MODE=ACTIVE/PASS=00001

/NOSTATUS/CHECK/NOECHO/NOMODEM

DCLT> (A) ?

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 12

2.5 DATA COMM. LINK TEST COMMANDS

THE DCLT COMMAND LEVEL IS ENTERED AFTER THE HARDWARE QUESTIONS ARE ANSWERED. TYPE IN DCLT COMMANDS WHEN THE DCLT> (A) ? PROMPT IS PRINTED.

COMMANDS CAN BE TYPED WITH ONLY ENOUGH CHARACTERS TO UNIQUELY SPECIFY A COMMAND.

THE COMMAND LINE IS INTERPRETED FROM LEFT TO RIGHT. THEREFORE THE QUALIFIER FARTHEREST TO THE RIGHT TAKES PRECEDENCE SINCE IT IS INTERPRETED LAST. (I.E. IF /CHECK...../NOCHECK APPEAR ON THE SAME LINE, NOCHECK WILL APPLY TO THE COMMAND.)
PARAMETERS WORD.)

SECTION 6.0 DESCRIBES THE DIFFERENT MODES OF TEST OPERATION AND THE TYPES OF MESSAGES AVAILABLE.

2.5.1 MESSAGE COMMANDS

COMMAND -----	DESCRIPTION -----
DCLT> (A) ? CLEAR EXPECTLIST	FILLS THE EXPECTLIST WITH ZEROS AND THEN PUTS ONE ITEP MESSAGE ON THE LIST.
DCLT> (A) ? CLEAR TRANSMITLIST	FILLS THE TRANSMITLIST WITH ZEROS AND THEN PUTS ONE ITEP MESSAGE ON THE LIST.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 13

DCLT> (A) ? SET TRANSMITMSG=TYPE/QUAL DEFINE A MESSAGE TO BE PUT ON
THE TRANSMIT LIST

DCLT> (A) ? SET EXPECTMSG=TYPE/QUAL DEFINE A MESSAGE TO BE PUT ON
THE EXPECTED LIST

WHERE: =TYPE IS ONE OF:

=ONES
=ZERUES
=1ALT
=OALT
=ITEP
=CCITT
=ALPHA
='A-Z,0-9,SPACES OR TABS IN QUOTES'

AND THE OPTIONAL /QUAL IS ONE OR BOTH OF:

/SIZE=NNN MAKE THE MESSAGE 'NNN' BYTES
LONG. (DEFAULT SIZE IS
DETERMINED BY THE MESSAGE
TYPE.

/COPY=NN COPY THIS MESSAGE INTO THE
BUFFER 'NN' TIMES (DEFAULT
IS 0 = PUT THE MESSAGE IN
ONLY ONCE)

NN AND NNN ARE DECIMAL NUMBERS.

NOTE: SET COMMANDS ADD MESSAGES TO THE LIST IN THE ORDER THEY'RE
DEFINED. THE FIRST SET COMMAND OVERWRITES THE ITEP DEFAULT
MESSAGE.

DCLT> (A) ? SET EXPECT=TRANSMIT COPY THE TRANSMIT LIST INTO
THE EXPECT LIST.

DCLT> (A) ? SHOW EXPECTLIST PRINT THE SIZE AND TYPE OF THE
MESSAGES IN THE EXPECT LIST.

DCLT> (A) ? SHOW TRANSMITLIST PRINT THE SIZE AND TYPE OF THE
MESSAGES IN THE TRANSMIT LIST

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 14

2.5.2 LINE SELECTION COMMANDS

THE KMS11-BD/BE DCLT COMMUNICATIONS LINK TEST PROGRAM TESTS A DDCMP COMMUNICATIONS LINK ON ANY OF THE EIGHT LINES SUPPORTED BY A KMS11-BD/BE. ONLY ONE LINK CAN BE TESTED AT A TIME. THE TEST OPERATOR MUST ESTABLISH A SPECIFIC LINE TO TEST BEFORE A LINK CAN BE TESTED IN RESPONSE TO A DCLT> RUN COMMAND.

COMMAND	DESCRIPTION
DCLT> (A) ? LINE ESTABLISH=N WHERE N = 0..7	ESTABLISH N AS THE CURRENT LINE NUMBER. SUBSEQUENT RUN COMMANDS APPLY TO LINE N
DCLT> (A) ? LINE KILL	DELETE THE CURRENT LINE NUMBER
DCLT> (A) ? LINE SHOW	PRINT THE CURRENT LINE NUMBER

2.5.3 OPERATOR ASSISTANCE COMMANDS

HELP	PRINT HELP INFORMATION
?	PRINT HELP INFORMATION
DUMP SSSSSS-EEEEEE/B WHERE SSSSSS AND EEEEE ARE OCTAL ADDRESSES AND /B IS OPTIONAL. THE DEFAULT IS PRINT WORDS. PRINT BYTES IF /B.	PRINT THE CONTENTS OF MEMORY LOCATIONS SSSSSS TO EEEEE. IF -EEEEEE IS NOT SPECIFIED THEN PRINT THE WORD AT LOCATION SSSSSS.

NOTE: THE DUMP COMMAND IS USEFUL FOR EXAMINING MESSAGE DATA. STARTING ADDRESSES CAN BE FOUND IN THE EVENT LOG.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 15

2.5.4 RUN COMMANDS

RUN MODES ARE DESCRIBED IN SECTION 6

COMMAND -----	DESCRIPTION -----
RUN MODE=MTYPE/QUAL	STARTS DCLT EXECUTING IN THE MODE SPECIFIED

NOTE: THERE IS NO DEFAULT MODE. MODE=MTYPE MUST BE TYPED
IN EACH RUN COMMAND.

WHERE =MTYPE IS ANY ONE OF THE FOLLOWING:

=ACTIVE	(FORCES /NOECHO ,NO LOOPING)
=PASSIVE	(FORCES NO LOOPING)
=RECEIVE	(FORCES /NOECHO ,NO LOOPING)
=LISTEN	(FORCES /NOECHO ,NO LOOPING, /NOCHECK)
=TRANSMIT	(FORCES /NOECHO ,NO LOOPING, /NOCHECK)
=TALK	(FORCES /NOECHO ,NO LOOPING, /NOCHECK)

THE KMS11-BD/BE DOES NOT SUPPORT DOWN-LINE-LOAD,
A MODE AVAILABLE IN SOME DCLT PROGRAMS.

AND OPTIONAL /QUAL IS ANY COMBINATION OF THE FOLLOWING:

/CHECK/NOCHECK	ENABLES/DISABLES CHECKING OF RECEIVED DATA AGAINST THE EXPECTED DATA
----------------	---

NOTE: IF BOTH NODES IN A DCLT TEST RUN IN ACTIVE MODE
AND /NOCHECK IS USED THEN END-OF-PASS IS DEFINED AS
OCCURRING WHEN THE NUMBER OF MESSAGES RECEIVED EQUALS
THE NUMBER OF MESSAGES IN THE TRANSMIT LIST. WITHOUT
DATA CHECKING, THERE IS NO WAY FOR DCLT TO KNOW HOW
MANY MESSAGES ARE EXPECTED TO BE RECEIVED.

/STATUS/NOSTATUS	ENABLES/DISABLES PRINTING PROGRAM STATUS MESSAGES.
------------------	---

/ECHO/NOECHO	ENABLES/DISABLES RETRANSMISSION OF THE DATA RECEIVED IN PASSIVE MODE.
--------------	--

NOTE: VALID ONLY FOR PASSIVE MODE. IF THIS SWITCH
IS USED THE TRANSMIT LIST MUST BE REBUILT.

CZKMSAO KMS11-BD/BE DCLT
 CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 16

/LOOP=LTYPE

ENABLES LOOPBACK TEST IN ACTIVE MODE TEST RUNS. THE ACTIVE MODE DEFAULT IS NO LOOPBACK. /LOOP IS IGNORED IN OTHER MODES.

WHERE =LTYPE IS ONE OF THE FOLLOWING:

=INTERNAL TTL ENABLES LINE UNIT INTERNAL LOOPBACK FOR THE ESTABLISHED LINE.

=CABLE PROVIDES A USEFUL RECORD FOR A TEST THAT IS RUN WITH A LOOPBACK CONNECTOR INSTALLED. THERE IS NO EFFECT ON THE HARDWARE OR IN THE DCLT PROGRAM.

THE FOLLOWING LOOP TYPES ARE NOT SUPPORTED BY THE KMS11-BD/BE. THEY HAVE NO EFFECT IN THIS DCLT PROGRAM.

=LOCALMODEM (NOT SUPPORTED BY KMS11-BD/BE)

=REMOTEMODEM (NOT SUPPORTED BY KMS11-BD/BE)

NOTE: THE KMS11-BD/BE DOES NOT REPORT MODEM STATUS.

/PASS=NN

SPECIFIES EXECUTION OF NN ITERATIONS OF THE TEST BEFORE END-OF-PASS. DEFAULT IS 1. IF NN = -1 THE TEST WILL BE REPEATED UNTIL CNTRL-C IS TYPED.

NOTE: SEE SECTION 6.1 FOR A DESCRIPTION OF THE RUN MODES.

CZKMSAO KMS11-BD/BE DCLT
 CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 17

EXIT COMMAND

THE EXIT COMMAND RETURNS THE USER TO THE SUPERVISOR DR>
 PROMPT AFTER PRINTING A SUPERVISOR END OF PASS.

DCLT> (A) ? EXIT

2.5.5 PRINT COMMAND

THE PRINT COMMAND INVOKES A REPORT COMMAND INTERPRETER.
 REPORT COMMANDS ARE USED TO PRINT THE EVENT LOG AND THE
 STATUS OF THE ESTABLISHED LINE.

RPT> IS THE PROMPT FOR REPORT COMMANDS

<u>COMMAND</u>	<u>DESCRIPTION</u>
RPT> HELP	PRINTS HELP INFORMATION FOR RPT>
RPT> ?	PRINTS HELP INFORMATION FOR RPT>
RP1> STATUSONLINE	PRINTS STATUS INFORMATION FOR THE ESTABLISHED LINE
RPT> LOG	DUMPS THE EVENT LOG
RPT> EXIT	RETURNS TO THE COMMAND PROMPT WHERE THE PRINT COMMAND WAS TYPED. DCLT> OR DR>

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 18

2.5.6 DEFAULTS

IF NO DCLT> SET COMMANDS HAVE BEEN USED THEN DEFAULTS EQUIVALENT TO THE FOLLOWING COMMANDS APPLY.

DCLT> (A) ? SET TRANSMITMSG=ITEP/SIZE=58/COPY=0
DCLT> (A) ? SET EXPECTMSG=ITEP/SIZE=58/COPY=0

THE DEFAULT COPY AND SIZE FOR EACH MESSAGE TYPE IS:

ONES - /SIZE=64/COPY=0
ZERONES - /SIZE=64/COPY=0
OALT - /SIZE=64/COPY=0
TALT - /SIZE=64/COPY=0
CCITT - /SIZE=64/COPY=0
ALPHA - /SIZE=65/COPY=0
ITEP - /SIZE=58/COPY=0
OPER. SPEC'D - /SIZE=LENGTH-OF-TEXT-TYPED-BETWEEN-QUOTES/COPY=0

DEFAULTS FOR THE DCLT> RUN COMMAND

MODE:
THERE IS NO DEFAULT MODE FOR THE DCLT> RUN COMMAND.
A MODE=MTYPE MUST BE TYPED EACH TIME A RUN IS TYPED.

DEFAULT QUALIFIERS FOR THE DCLT> RUN COMMAND ARE:

/NOSTATUS/CHECK/NOECHO/NOMODEM/PASS=1

HENCE:

DCLT> (A) ? RUN MODE=ACTIVE

IS EQUIVALENT TO:

DCLT> (A) ? RUN MODE=ACTIVE/NOSTATUS/CHECK/NOECHO/NOMODEM/PASS=1

THE KMS11-BD/BE DCLT PROGRAM HAS NO DEFAULT ESTABLISHED LINE.

OTHER NOTES:

^C ALWAYS RETURNS YOU TO "DR>" (THE SUPERVISOR)
<CR> IS SEEN AS A COMMAND TERMINATOR
"RUBOUT" DELETE LAST CHAR. TYPED IN COMMAND STRING

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 19

2.6 QUICK START-UP PROCEDURE (XXDP+)

TO START-UP THIS PROGRAM:

1. BOOT XXDP+
2. GIVE THE DATE AND ANSWER THE LSI AND 50HZ (IF THERE IS A CLOCK) QUESTIONS
3. TYPE "R CZKMSA", WHERE NAME IS THE NAME OF THE BIN OR BIC FILE FOR THIS PROGRAM
4. TYPE "START"
5. ANSWER THE "CHANGE HW" QUESTION WITH "Y"
6. ANSWER ALL THE HARDWARE QUESTIONS. THE NUMBER OF UNITS THAT CAN DCLT CAN USE IS ALWAYS "1".

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING ONLY THE DEFAULTS FOR FLAGS. THESE DEFAULTS ARE DESCRIBED IN SECTION 2.3.

7. AT THE "DCLT> (A) ?" PROMPT, TYPE

DCLT> (A) ? LINE ESTABLISH=N<CR>

WHERE N IS THE NUMBER IN THE RANGE
0 TO 7 OF THE LINE TO BE TESTED

DCLT> (A) ? RUN MODE=ACTIVE<CR>

A SHORT FORM OF THESE COMMANDS CAN BE USED.

DCLT> (A) ? L E=N<CR>

DCLT> (A) ? R M=A<CR>

WHEN YOU FOLLOW THIS PROCEDURE YOU WILL BE USING THE DEFAULT TRANSMIT AND EXPECTED MESSAGES. THE DEFAULT PASS COUNT AND "RUN" QUALIFIERS ARE ALSO BEING USED. THESE DEFAULTS ARE DESCRIBED IN SECTION 2.5.3.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 20

3.0 ERROR INFORMATION

3.1 TYPES OF ERROR MESSAGES

THERE ARE THREE LEVELS OF ERROR MESSAGES THAT MAY BE ISSUED BY A DIAGNOSTIC: GENERAL, BASIC AND EXTENDED. GENERAL ERROR MESSAGES ARE ALWAYS PRINTED UNLESS THE "IER" FLAG IS SET (SECTION 2.3). THE GENERAL ERROR MESSAGE IS OF THE FORM:

```
NAME TYPE NUMBER ON UNIT NUMBER TST NUMBER PC:XXXXXX
ERROR MESSAGE
```

WHERE: NAME = DIAGNOSTIC NAME
TYPE = ERROR TYPE (SYS FATAL, DEV FATAL, HARD OR SOFT)
NUMBER = ERROR NUMBER
UNIT NUMBER = 0 - N (N IS LAST UNIT IN PTABLE)
TST NUMBER = TEST AND SUBTEST WHERE ERROR OCCURRED
PC:XXXXXX = ADDRESS OF ERROR MESSAGE CALL

BASIC ERROR MESSAGES ARE MESSAGES THAT CONTAIN SOME ADDITIONAL INFORMATION ABOUT THE ERROR. THESE ARE ALWAYS PRINTED UNLESS THE "IER" OR "IBE" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL MESSAGE.

EXTENDED ERROR MESSAGES CONTAIN SUPPLEMENTARY ERROR INFORMATION SUCH AS REGISTER CONTENTS OR GOOD/BAD DATA. THESE ARE ALWAYS PRINTED UNLESS THE "IER", "IBE" OR "IXE" FLAGS ARE SET (SECTION 2.3). THESE MESSAGES ARE PRINTED AFTER THE ASSOCIATED GENERAL ERROR MESSAGE AND ANY ASSOCIATED BASIC ERROR MESSAGES.

3.2 SPECIFIC ERROR MESSAGES

3.2.1 COMMAND LINE INTERPRETER ERRORS:

ERROR MESSAGE:	MEANING
-----	-----
?ILL CMD-BAD SYNTAX?	A COMMAND WITH AN ILLEGAL CHAR WAS TYPED - RETYPE THE COMMAND. THE VALID COMMANDS AND THEIR SYNTAX ARE SHOWN IN SECTION 2.5.
?INCMPLTE CMD?	A REQUIRED PART OF A COMMAND WAS LEFT OUT.
?NUM TOO BIG?	THE VALUE OF A NUMERIC STRING IN THE COMMAND LINE WAS LARGER THAN 65535 OR 177777 OCTAL. (> 16 BITS).
?BAD RADIX?	A "8" OR "9" WAS TYPED WHEN AN OCTAL STRING WAS EXPECTED. PROBABLY OCCURRED WHEN TYPING A "DUMP" COMMAND WHERE OCTAL ADDRESSES ARE EXPECTED.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 21

- ? "LOOP" VALID ONLY IN ACTIVE? THE "/LOOP=.." SWITCH WAS TYPED IN A RUN COMMAND BUT THE MODE WAS NOT SET TO ACTIVE. MAINTENANCE LOOP IS ONLY POSSIBLE IF THE MODE OF OPERATION IS ACTIVE.
- ? "ECHO" VALID ONLY IN PASSIVE? THE "/ECHO" SWITCH WAS TYPED IN A RUN COMMAND BUT THE MODE WAS NOT SET TO PASSIVE. ECHOING OF RECEIVED DATA IS ONLY POSSIBLE IF THE MODE OF OPERATION IS PASSIVE.
- ? ILL CHR- "A-Z,0-9,SP,TAB" ONLY? A CHARACTER TYPED WITHIN QUOTES WHEN TRYING TO DEFINE THE CONTENTS OF A TRANSMIT OR EXPECT MESSAGE WAS NOT A "A-Z,0-9,SPACE OR TAB". RETYPE THE COMMAND WITH ONLY THESE CHARACTERS BETWEEN QUOTES.
- ? "SIZE=0" NOT VALID? A MESSAGE ZERO BYTES LONG CAN NOT BE BUILT. RETYPE THE COMMAND WITH A "/SIZE=NNN". IF NO "/SIZE=" IS TYPED A DEFAULT SIZE WILL BE USED.
- ? TRANSMIT AND EXPECT LIST MUST BE IDENTICAL FOR LOOP?
IF RUN COMMAND WITH "/LOOP/CH" IS TYPED TRANSMIT AND EXPECT LISTS MUST BE EQUAL. USE "SE E=T" COMMAND.
- ? A LINE MUST BE ESTABLISHED TO EXECUTE?
A DCLT> RUN COMMAND WAS TYPED WITH NO LINE NUMBER ESTABLISHED.
A DCLT> LINE ESTABLISH=N
COMMAND MUST BE TYPED IN BEFORE A DCLT> RUN COMMAND CAN EXECUTE.
- ? LINE NUMBER= XXX INVALID (NUMBERS 0-7 ARE VALID)?
A DCLT> LINE ESTABLISH=XXX WAS TYPED IN WITH XXX NOT IN THE RANGE 0..7.
- DOWN LINE LOAD NOT AVAILABLE FOR THE KMS11-BD/BE
A DCLT> RUN MODE=DOWN WAS TYPED.
THE KMS11-BD/BE DOES NOT SUPPORT DOWN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 22

LINE LOAD.

3.2.2 DCLT ERROR MESSAGES:

CLOCK NOT FOUND

THIS MEANS THAT NO CLOCK WAS FOUND ON THE SYSTEM THE DIAGNOSTIC WILL STILL RUN BUT NONE OF THE TIME OUT CONDITIONS WILL OCCUR.

BAD CLOCK - PROGRAM WILL HANG ON "TIMEOUT"!!

THIS MEANS THAT EITHER NO CLOCK WAS ON THE SYSTEM OR THE ONE THAT WAS FOUND DID NOT INTERRUPT WHEN ASKED TO DO A "TICK".
THE PROGRAM WILL STILL RUN, BUT ANY OF THE PROGRAM THAT TIMES THE DEVICE WILL HANG IF THE DEVICE TIMES OUT. ALSO, THE EVENT LOG WILL CONTAIN A ZERO EVENT TIME FOR ALL EVENTS LOGGED.

MAX. CHAR. MSG COUNT EXCEEDED - MSG. NOT BUILT !!

THIS MEANS THAT THE TRANSMIT OR EXPECT BUFFER IS FULL. NO MORE MESSAGES CAN BE ADDED TO THAT BUFFER.

BUFFER FULL - MSG. NOT BUILT !!

THIS MEANS THAT THE LAST MESSAGE YOU TRIED TO ADD TO EITHER THE TRANSMIT OR EXPECT BUFFER CAUSED THE TOTAL NUMBER OF MESSAGES TO BE EXCEEDED. NO MORE MESSAGES CAN BE ADDED TO THAT BUFFER. THE LIMIT IS DETERMINED BY THE SIZE OF THE MESSAGE POINTER TABLE. THE LIMIT IS CURRENTLY 15.

CHAR. COUNT EXCEEDS BUFF LIMIT - MSG TRUNCATED

THIS MEANS THAT THE LAST MESSAGE YOU TRIED TO ADD TO THE TRANSMIT OR EXPECT BUFFER CAUSED THE TOTAL CHAR. COUNT FOR THAT BUFFER TO EXCEED THE LIMIT. THE LIMIT IS 512. BYTES.
THE MESSAGE WAS TRUNCATED TO COMPLETELY FILL THE BUFFER. NO MORE MESSAGES CAN BE ADDED TO THAT BUFFER.

NO LINE NUMBER IS ESTABLISHED

RESPONSE TO DCLT> LINE SHOW COMMAND WHEN NO LINE NUMBER IS ESTABLISHED.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 23

3.2.3 DEVICE ERROR MESSAGES:

DATA COMPARISON DATA ERROR
BYTE # IN MSG=XXX EXPTD=YYY

RECVD=ZZZ
XXX= OFFSET OF THAT BYTE FROM THE START
OF THE COMPARE OR EXPECT MESSAGE.
YYY= THE CONTENTS OF THAT BYTE IN THE
EXPECTED MESSAGE
ZZZ= THE CONTENTS OF THAT BYTE IN THE
RECEIVED MESSAGE

UP TO FIVE OF THESE ERRORS WILL BE
PRINTED PER MESSAGE COMPARED. ONLY
THE FIRST FIVE MISMATCHES WILL BE
INDIVIDUALLY REPORTED, BUT TOTAL
NUMBER OF MISMATCHES IS REPORTED
BY ANOTHER ERROR.

PRINTING THE EVENT LOG AND USING THE
DCLT "DUMP" COMMAND WILL ALLOW YOU TO
FIND THE ADDRESS OF THE MESSAGE AND
EXAMINE IT.

DATA COMPARISON DATA ERROR
TOTAL MISMATCHES IN MSG = NNN

THIS MEANS THAT WHEN THE MESSAGE
RECEIVED WAS COMPARED AGAINST THE
MESSAGE THAT WAS EXPECTED, SOME OF
THE CHARS. WERE NOT THE SAME.

DATA COMPARISON LENGTH ERROR
COMPARE COUNT= XXX RECEIVE COUNT= ZZZ

XXX= NUMBER OF BYTES IN THE COMPARE
MESSAGE
ZZZ= NUMBER OF BYTES IN THE RECEIVED
MESSAGE
THIS MEANS THAT THE MESSAGE RECEIVED
WAS A DIFFERENT LENGTH THEN THE MESSAGE
THAT WAS EXPECTED.

* NOTE * - IN THE FOLLOWING ERROR DESCRIPTIONS XXXXX
***** REFERS TO THE OCTAL CONTENTS OF THE DEVICE REGISTERS
SPECIFIED.

KMS FAILED TO START
SELO = XXXXXX SEL2 = XXXXXX

: THIS ERROR INDICATES THAT THE
: DEVICE DID NOT CLEAR BSEL2
: WITHIN 1000 (OCTAL) CLOCK TICKS
: AFTER THE RUN BIT WAS SET.
: MAY INDICATE A MICROPROCESSOR
: FAILURE.

:NOTE: NO CONNECTION HAS BEEN ESTABLISHED
: IF THIS ERROR OCCURS

KMS DID NOT RESPOND TO CSR COMMAND
SELO = XXXXXX SEL2 = XXXXXX

:THIS ERROR INDICATES THAT THE
:KMS FAILED TO RETURN AN EXPECTED
:RESPONSE TO A COMMAND ISSUED
:BY THE PDP-11 HOST.

WAITING TO COMPLETE DEVICE INITIALIZATION
SELO = XXXXXX SEL2 = XXXXXX

:THIS ERROR INDICATES THAT THE
:KMS DID NOT RETURN AN INITIALIZATION
:COMPLETED RESPONSE AFTER THE
:PDP-11 ISSUED A COMMAND TO
:INITIALIZE THE DEVICE. THIS
:MAY INDICATE A MICROPROCESSOR
:FAILURE.

:NOTE: NO CONNECTION HAS BEEN ESTABLISHED
: IF THIS ERROR OCCURS

WAITING FOR START DDCMP REQUEST TO COMPLETE
SELO = XXXXXX SEL2 = XXXXXX

:THIS ERROR INDICATES THAT THE
:KMS HAS NOT RETURNED A RESPONSE
:CONFIRMING THAT THE SYSTEM AT
:THE OTHER END OF THE LINK HAS
:HAS STARTED (OR RESTARTED)
:DDCMP AFTER THE PDP11 ISSUED
:A REQUEST TO THE KMS TO ENTER
:DDCMP START STATE.

:NOTE: IT IS POSSIBLE THAT NO CONNECTION
: HAS BEEN ESTABLISHED IF THIS
: ERROR OCCURS

TIME OUT WAITING FOR TX OR RX TO COMPLETE
SELO = XXXXXX SEL2 = XXXXXX

:THIS ERROR IS THE MOST POPULAR
:IT INDICATES THAT THE 60 SEC
:TIMER EXPIRED WHEN THE DEVICE
:WAS EXPECTING TO GET A RX OR
:TRANSMIT COMPLETE. AFTER THIS
:ERROR OCCURS THE PROGRAM WILL
:RESET THE TIMER AND LOOP AGAIN

KMS DID NOT RESPOND TO RQI
SELO = XXXXXX SEL2 = XXXXXX

:THIS ERROR INDICATES THAT THE
:DEVICE DID NOT RETURN RQI IN
:RESPONSE TO AN RQI BEFORE THE
:TIMER EXPIRED. THE TIMER IS
: 2000 TICKS.

ILLEGAL TRANSMIT COMPLETE
LINE NO BDL NO
XXX XXX
:INDICATES DEVICE GOT A TX
:COMPLETE WHEN IT WAS NOT
:EXPECTING IT.

ILLEGAL RECEIVE COMPLETE
LINE NO BDL NO
XXX XXX
:INDICATES DEVICE GOT A RX
:COMPLETE WHEN IT WAS NOT
:EXPECTING IT.

DDCMP RESTART RECEIVED
LINE NO = XXX
:THIS ERROR INDICATES THAT
:A DDCMP RESTART MESSAGE WAS
:RECEIVED WHILE THE KMS WAS
:IN THE RUN STATE.

KMS TIMEOUT WAITING FOR RESPONSE TO TX
STATUS = XXX FLAGS = XXX
TEMP DATA = XXX LAST TX MSG = XXX
ACKS RCVD = XXX MSGS RCVD = XXX
NAKS SENT = XXX NAKS RCVD = XXX
:THIS ERROR INDICATES THAT THE
:KMS HAS TRANSMITTED A MESSAGE
:AND IS WAITING FOR A RESPONSE

KMS TIMEOUT. NO RESPONSE TO START MESSAGE
STATUS = XXX FLAGS = XXX
TEMP DATA = XXX LAST TX MSG = XXX
ACKS RCVD = XXX MSGS RCVD = XXX
NAKS SENT = XXX NAKS RCVD = XXX
:THIS ERROR INDICATES THAT
:THE KMS HAS TRANSMITTED A
:DDCMP START MESSAGE AND IS
:WAITING FOR A RESPONSE

:NOTE: THE TWO MESSAGES ABOVE INDICATE THAT THE KMS
: IS UNABLE TO CONTINUE TRANSMISSION ON A LINE.
: IT IS POSSIBLE THAT THE LINE IS NOT CONNECTED TO THE MODEM.

TX THRESHOLD REACHED
LINE NO NAK REASON
XXX TEXT STATING REASON
:THIS ERROR INDICATES THAT THE
:KMS HAS RECEIVED EIGHT SUCCUESSIVE
:NAKS FROM THE OTHER END OF THE LINK

TX REJECTED BEFORE COMPLETION
LINE NO BDL NO
XXX XXX
:THIS ERROR INDICATES THAT THE
:KMS HAS NOT COMPLETED A REQUESTED
:MESSAGE TRANSMISSION AND WILL
:NOT ATTEMPT TO RETRANSMIT

KMS TIMEOUT. WAITING FOR CSR INPUT
SEL0 = XXXXXX SEL2 = XXXXXX
:THIS ERROR OCCURS AFTER THE
:KMS HAS RESPONDED TO AN RQI
:ISSUED BY THE HOST PDP-11 WITH
:A RDYI AND IS WAITING FOR THE
:PDP-11 TO CLEAR RDYI. THE PDP-11
:CLEARS RDYI TO INDICATE THAT
:A COMMAND IS SPECIFIED IN THE
:CSR REGISTERS.

KMS TIMEOUT. CSR OUTPUT NOT ACCEPTED
SEL0 = XXXXXX SEL2 = XXXXXX
:THIS ERROR OCCURS AFTER THE
:KMS HAS SET RDYO TO INDICATE
:THAT THE CSR REGISTERS CONTAIN
:AN OUTPUT CONTROL MESSAGE FOR
:FOR THE HOST PDP-11.

4.0 PERFORMANCE AND PROGRESS REPORTS

DCLT USES IT'S OWN METHOD FOR DETERMINING AN "END OF PASS" WHICH IS CALLED A "DCLT END OF PASS". THE NUMBER OF "DCLT PASSES" TO BE RUN IS SPECIFIED BY THE "/PASS=XXX" SWITCH ON THE DCLT RUN COMMAND. THE TOTAL NUMBER OF "DCLT ERRORS" ARE LOGGED IN IN THE EVENT LOG WHEN EACH "DCLT PASS" IS COMPLETED.

4.1 PRINTING OF EVENT LOG

SIGNIFICANT EVENTS OR CHECK-POINTS WILL BE LOGGED IN A "CIRCULAR QUEUE" STORAGE AREA CALLED THE EVENT LOG. THE LAST 45 EVENTS ARE KEPT LOGGED AND CAN BE LISTED ON THE OPERATORS CONSOLE BY GIVING A "PRINT" COMMAND AT THE "DR>" (DIAGNOSTIC SUPERVISOR) OR "DCLT>" (DCLT) LEVEL. THE PRINT COMMAND MUST BE FOLLOWED BY A LOG COMMAND. THE EVENTS ARE PRINTED IN A "LAST-IN FIRST-OUT" ORDER.

EVENT TIME IS TYPED OUT AS MMM:SS:TT (LIKE 254:36:07) WHERE MMM,SS,TT REPRESENT THE NUMBER OF MINUTES, SECONDS, CLOCK TICKS SINCE THE LAST START OR RESTART. IT SHOULD BE NOTED THAT THE TIMES ARE RELATIVE SINCE WHILE THE PROCESSOR IS RUNNING AT PRIORITY 7 THE CLOCK CAN'T INTERRUPT TO KEEP TIME. THIS IS THE CASE WHILE THE PROGRAM IS FETCHING DCLT COMMANDS FROM THE OPERATOR. IT SHOULD ALSO BE NOTED THAT THERE ARE ONLY 8 BITS AVAILABLE TO STORE RELATIVE MINUTES SO "TIME" WILL WRAP TO 000:00:00 AFTER 256:59:59.

A START OR RESTART COMMAND AT THE "DR>" LEVEL INITIALIZES THE EVENT LOG. THEREFORE IT IS WISE TO DO A "PRINT" "LOG" AT THE "DR>" LEVEL BEFORE GIVING A "START" OR "RESTART".

THE TYPES OF EVENTS KEPT IN THE EVENT LOG ARE:

TRANSMIT MESSAGE QUEUED:

EVENT TIME, ADDRESS OF TRIBUTARY TO/FROM
ADDRESS OF 1ST BYTE OF MESSAGE,
TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

TRANSMIT MESSAGE COMPLETED:

EVENT TIME, ADDRESS OF TRIBUTARY TO/FROM
ADDRESS OF 1ST BYTE OF MESSAGE,
TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

RECEIVE SPACE QUEUED:

EVENT TIME, ADDRESS OF TRIBUTARY TO/FROM
ADDRESS OF 1ST BYTE OF MESSAGE,
TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

RECEIVE MESSAGE COMPLETED:

EVENT TIME, ADDRESS OF TRIBUTARY TO/FROM
ADDRESS OF 1ST BYTE OF MESSAGE,
TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

DATA COMPARISON STARTED:

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 27

EVENT TIME, ADDRESS OF TRIBUTARY TO/FROM
ADDRESS OF 1ST BYTE OF RECEIVED MSG.,
TOTAL NO. OF BYTES IN RCV. MSG., TOTAL NO. OF BYTES
IN EXPECT MSG.

DATA COMPARISON DATA ERROR:
EVENT TIME, ADDRESS OF TRIBUTARY TO/FROM
ADDRESS OF 1ST BYTE OF RECEIVED MSG.,
TOTAL NO. OF BYTES IN RCV. MSG., TOTAL NO. OF
COMPARISON FAILURES

DATA COMPARISON LENGTH ERROR:
EVENT TIME, ADDRESS OF TRIBUTARY TO/FROM
ADDRESS OF 1ST BYTE OF RECEIVED MSG.,
TOTAL NO. OF BYTES IN RCV. MSG., TOTAL NO. OF BYTES
IN EXPECT MSG.

DEVICE INIT AND SETUP:
EVENT TIME, MODE OF OPERATION, TYPE OF MAINTENANCE
LOOP, "DCLT" PASS COUNT, "RUN" PARAMETERS

DEVICE ERROR:
EVENT TIME, DEVICE ERROR MESSAGE, CONTENTS OF TWO
REGISTERS RELATING TO THE ERROR.

END OF PASS:
EVENT TIME, "DCLT" PASS COUNT, "DCLT" ERROR COUNT,
#OF RX THRESHOLD ERRORS, # OF TX THRESHOLD ERRORS

NOTE - RX THRESHOLDS AND TX THRESHOLDS OCCUR IF
ONE STATION IS STARTED BEFORE THE OTHER
OR IF LINKS ARE RUN AT HIGH SPEED

CZKMSAO KMS11-BD/BE DCLT
 CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 28

4.2 OPERATOR STATUS MESSAGES

THE "/STATUS, /NOSTATUS" QUALIFIERS FOR THE DCLT "RUN" COMMAND ENABLES/DISABLES THE PRINTING OF PROGRAM STATUS MESSAGES TO THE OPERATOR. THESE MESSAGES ARE INTENDED TO TELL THE OPERATOR WHAT THE DCLT PROGRAM IS CURRENTLY DOING. BELOW ARE THE MESSAGES THAT MIGHT BE PRINTED AND THEIR MEANING:

MESSAGE	MEANING
-----	-----
TXQ	DEVICE IS ABOUT START TRANSMITTING A MESSAGE
TXC	TRANSMISSION OF MESSAGE COMPLETED
RXQ	DEVICE HAS QUEUED SPACE TO RECEIVE/ COMPLETED RECEIVE
ERR	DEVICE ERROR HAS OCCURRED
INI	DEVICE ABOUT TO BE INITIALIZED
CMP	ABOUT TO DO DATA CHECKING OF RECVD VS. EXPTD DATA
CML	LENGTH ERROR OCCURRED DURING DATA COMPARISON
CMD	DATA ERROR OCCURRED DURING DATA COMPARISON
EOP	END OF PASS

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 29

4.3 PRINTING LINE STATUS

THE KMS11-BD/BE DCLT PROGRAM READS THE LINE STATUS MAINTAINED IN THE DEVICE WHEN:

1. A DCLT> RUN COMMAND COMPLETES EXECUTION
2. A DCLT> EXIT COMMAND IS PERFORMED
3. A CTRL/C EXIT FROM THE DCLT> COMMAND LEVEL OCCURS
4. A MESSAGE THAT HAS BEEN SUBMITTED FOR TRANSMISSION IS REJECTED BY THE DEVICE BEFORE THE TRANSMISSION HAS COMPLETED.

THE RPT> STATUSONLINE COMMAND PRINTS THE LINE STATUS OF THE ESTABLISHED LINE. IF THE DEVICE MICROPROCESSOR IS RUNNING WHEN A RPT> STATUSONLINE COMMAND IS EXECUTED LINE STATUS IS READ AND PRINTED. IF THE DEVICE MICROPROCESSOR IS NOT RUNNING THE LAST RECORDED LINE STATUS IS PRINTED.

A LINE STATUS REPORT INCLUDES:

LINE STATUS BYTE:

- BIT0 = 1 LINE ENABLED
= 0 LINE DISABLED
- BIT1 = 1 MOP MODE ***NOT SUPPORTED IN KMS11-BD/BE DCLT
= 0 DDCMP PROTOCOL ENABLED
- BIT2 = 1 TRANSMIT ACTIVE
- BIT3 = 1 RECEIVE ACTIVE
- BIT4 = 1 START PENDING
- BIT5 = 1 REP PENDING
- BIT6 = 0 FULL DUPLEX
= 1 HALF DUPLEX **NOT SUPPORTED IN KMS11-BD/BE
- BIT7 = 1 TRANSMIT SELECTED

LINE FLAGS BYTE:

- BIT0 = 1 START RECEIVED
- BIT1 = 1 NAK RECEIVED
- BIT2 = 1 REP RECEIVED
- BIT3 = 1 CONTROL MESSAGE PENDING
- BIT4 = 1 ACK PENDING
- BIT5 = 1 STACK RECEIVED
- BIT6 = 1 TX TIMEOUT
- BIT7 = 1 START PENDING

TEMP DATA BYTE:

- NUMBER OF LAST MESSAGE TRANSMITTED (1 BYTE)
- NUMBER OF ACKS RECEIVED (1 BYTE)
- NUMBER OF MESSAGES RECEIVED (1 BYTE)
- NUMBER OF NAKS TRANSMITTED (1 BYTE)
- NUMBER OF NAKS RECEIVED (1 BYTE)

CZKMSAO KMS11-BD/BE DCLT
 CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 30

5.0 DEVICE INFORMATION TABLES

THIS IS THE DEFAULT HARDWARE P-TABLE. THE VALUES AND SIZE ARE USED AS A "TEMPLATE" FOR CREATING ACTUAL P-TABLE ENTRIES AND THE DEFAULT VALUES PROVIDED FOR THE OPERATOR. SEE SECTION 2.4 FOR AN EXAMPLE OF THE HARDWARE QUESTIONS.

THE NUMBERS IN BRACKETS (I.E. [10]) INDICATES THE OFFSET OF THE WORD INTO THE HARDWARE P-TABLE. THE OFFSETS MUST MATCH THE P-TABLE OFFSETS USED IN THE HARDWARE PARAMETER CODING SECTION WHERE THE "GET PARAMETER" CALLS ARE USED TO FILL THE P-TABLE.

.WORD	164100	:[0] CSR ADDRESS
.WORD	400	:[2] INTERRUPT VECTOR
.WORD	240	:[4] INTERRUPT PRIORITY (5)
.WORD	164110	:[6] MODEM CONTROL CSR ADDRESS
		:

6.0 MODE AND MESSAGE DESCRIPTIONS

6.1 MODE DESCRIPTIONS

6.1.1 TRANSMIT MODE

THE TRANSMIT LIST OF MESSAGES IS TRANSMITTED WITHOUT EXPECTING ANY DATA TO BE RECEIVED.

6.1.2 RECEIVE MODE

SPACE IS QUEUED FOR THE DEVICE TO RECEIVE MESSAGES. AFTER RECEIVING AN "EXPECTED" NUMBER OF MESSAGES, THE DATA RECEIVED CAN BE COMPARED AGAINST A LIST OF "EXPECT TO RECEIVE" MESSAGES IF DATA-CHECKING IS ENABLED.

6.1.3 PASSIVE MODE

EVERY TIME A MESSAGE IS RECEIVED, A MESSAGE IS TRANSMITTED. DATA CHECKING CAN BE DONE ON THE RECEIVED DATA. THE "/ECHO, /NOECHO" ENABLES/DISABLES THE RETRANSMISSION OF THE DATA RECEIVED.

6.1.4 ACTIVE MODE

A LIST OF MESSAGES IS TRANSMITTED AND MESSAGES ARE RECEIVED. AFTER RECEIVING AN "EXPECTED" NUMBER OF MESSAGES, THE DATA RECEIVED CAN BE COMPARED AGAINST A LIST OF "EXPECT TO RECEIVE" MESSAGES IF DATA-CHECKING IS ENABLED.

6.1.5 DOWN-LINE-LOAD

THE KMS11-BD/BE DOES NOT SUPPORT DOWN LINE LOAD

6.1.6 TALK AND LISTEN MODE
-----6.1.6.1 TALK MODE

THE "TALK" END OF THE LINK TRANSMITS OPERATOR-TYPED MESSAGES UNTIL A "EXIT" MESSAGE IS TYPED. AT THAT POINT, THE NODE GOES INTO "LISTEN" MODE. AN "EXIT MESSAGE" IS A MESSAGE WHOSE FIRST FOUR CHARACTERS ARE "EXIT". SINCE ONLY THE FIRST FOUR CHARACTERS NEED TO BE "EXIT", MORE CHARACTERS CAN BE ADDED SO THAT A MESSAGE MAY BE SENT AND THE MODE SWITCHED ALL AT ONCE. FOR EXAMPLE:

TLK> EXIT ALL OF THIS LINE IS SENT THEN MODE SWITCHED

6.1.6.2 LISTEN MODE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 32

THE "LISTEN" END OF THE LINK PRINTS ALL OF THE MESSAGES
RECEIVED BY THE DEVICE ON THE OPERATOR'S CONSOLE. IF THE MESSAGE
RECEIVED IS AN "EXIT" MESSAGE, THEN THE NODE ENTERS "TALK" MODE.
AN "EXIT MESSAGE" IS A MESSAGE WHOSE FIRST FOUR CHARACTERS ARE "EXIT".

6.1.7 MAINTENANCE "LOOP" MODES

REMEMBER THAT THE WHENEVER A "RUN" COMMAND IS TYPED, THE DEFAULT IS NO LOOPBACK AND THAT A LOOP MODE MUST BE SPECIFIED BY A "/LOOP=.." IF A LOOP MODE IS DESIRED.
LOOP MODES ARE ONLY VALID IF THE MODE TO RUN IS ACTIVE

INTERNALTTL LOOPS DATA INTERNALLY IN THE LINE UNIT

THE FOLLOWING LOOP MODES HAVE NO EFFECT. IN THE KMS11-BD/BE DCLT PROGRAM THEY ARE EQUIVALENT TO NO LOOPBACK.

CABLE HAS NO EFFECT BUT MAKES A NICE BOOKKEEPING FEATURE WHEN TESTING WITH A TURNAROUND CONNECTOR.

LOCALMODEM NOT SUPPORTED BY KMS11-BD/BE
 (ALSO CALLED ANALOG- LOOPBACK)

REMOTEMODEM NOT SUPPORTED BY KMS11-BD/BE
 (ALSO CALLED DIGITAL-LOOPBACK)

NOTE THAT THE ACTIVE MODE OF THE DCLT> RUN COMMAND CAN BE USED WITH AN H3256 TURNAROUND CONNECTOR.

THE TURNAROUND CONNECTOR CAN BE PLACED:

1. ON THE DISTRIBUTION PANEL
2. ON THE FAR END OF A CABLE CONNECTED TO THE DISTRIBUTION PANEL
3. ON THE FAR END OF A CABLE CONNECTED TO A MODEM OR MODEM ELIMINATOR.

THE COMMAND:

DCLT> (A) ? R MODE=ACTIVE/L=CABLE

WILL THEN RECORD THAT LOOPING VIA A TURNAROUND CONNECTOR IS BEING ATTEMPTED.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 34

6.1.8 MODE SUMMARY TABLE

THE FOLLOWING TABLE SUMMARIZES THE MODES THAT CAN BE RUN TOGETHER WHEN THE DCLT PROGRAM IS RUNNING ON TWO PROCESSORS (ONE AT EACH END OF THE LINK):

STATION A "HOST" NODE	STATION A "/LOOP" ALLOWED?	STATION B "REMOTE" NODE
TALK	NO	LISTEN*, RECEIVE
LISTEN	NO	TALK*, TRANSMIT
TRANSMIT	NO	RECEIVE*, LISTEN
RECEIVE	NO	TRANSMIT*, TALK
PASSIVE	NO	ACTIVE*
ACTIVE	YES	ACTIVE*
ACTIVE	YES	PASSIVE*

*= MOST LIKELY TO BE IN THAT MODE

6.2 MESSAGE DESCRIPTIONS

NAME	DESCRIPTION
ZEROES	MESSAGE OF ALL 0'S (00000000,00000000,00000000,...)
ONES	MESSAGE OF ALL 1'S (11111111,11111111,11111111,...)
1ALT	MESSAGE OF ALTERNATING 1'S (10101010,10101010,...)
0ALT	MESSAGE OF ALTERNATING 0'S (01010101,01010101,...)
CCITT	"CCITT" 512-BIT (VS. 511 BITS) TEST PATTERN
ITEP	"INTERPROCESSOR TEST PROGRAM'S (ITEP)" MESSAGE 1(DP1:) (<177><177>/SA THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG.<15><12><001><177><177><177><177>)
ALPHA	ALPHA-NUMERICS (OR FUTURE COMM TURNAROUND MSG) (#\$!' (AMPERSAND)'()*+,-.0123456789:;<=>?@ABCDEFGHIJK LMNOPQRSTUVWXYZ/[\] ^ _ %)
OPERATOR-SPECIFIED	"A-Z,0-9,SPACES,TABS" THESE ARE THE CHARACTERS THAT CAN BE TYPED BETWEEN QUOTATION MARKS ("..") TO SPECIFY A UNIQUE MESSAGE.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 35

7.0 OTHER INFORMATION

7.1 INTERFACING TO AN "ITEP" NODE

THIS DCLT WILL INTERFACE ONLY TO THE ITEP FOR DMC.
AND THE FOLLOWING TABLE APPLIES TO THE ITEP NODE:

ITEP NODE	DCLT NODE
ONE-WAY-OUT	RECEIVE OR LISTEN
ONE-WAY-IN	TRANSMIT OR TALK
INTERNAL LOOP	ACTIVE
EXTERNAL LOOP	ACTIVE OR PASSIVE

NOTE: WHEN INTERFACING TO ITEP THE RX BUFFER ON THE
ITEP SIDE IS ONLY 10 BYTES LARGER THAN THE TX BUFFER YOU
HAVE SELECTED, SO BE SURE TO SET THE TX BUFFER ON THE DCLT
NODE ACCORDINGLY.

WHEN ITEP IS IN A MODE THAT IT IS EXPECTING TO BE TRANSMITTED
TO, A SOFT ERROR "BASE TABLE ERR COUNTS NON-ZERO" WILL OCCUR.
THIS IS DUE TO THE SPEED DIFFERENCES IN THE SOFTWARE.

WHEN DCLT IS IN LISTEN MODE THE RX BUFFER IS ONLY
82 BYTES LONG THEREFORE DO NOT SEND THE DCLT NODE
ITEP MSG. 3 FROM THE ITEP NODE OR A "LOST DATA" ERROR WILL
OCCUR

BE SURE ITEP NODE HAS INCORPORATED PATCH FROM DEPO# MD-11-DZDMO-A1

ITEP NODE SHOULD ALWAYS BE RUN WITH SW 4 = TO 0

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 36

7.2 TROUBLESHOOTING HINTS

LISTED BELOW ARE SOME SETUPS THAT COULD BE USED FOR ISOLATING FAULTS. THESE ARE BY NO MEANS THE ONLY WAYS DCLT CAN BE USED !!!!!!! DCLT IS MEANT TO BE A VERY FLEXIBLE TOOL! THIS SECTION IS MEANT TO GIVE SOMEONE NOT TOO FAMILIAR WITH DCLT A PLACE TO START.

REMEMBER THAT THE PRINTING OF STATUS MESSAGES AND PRINTING OF THE EVENT LOG CAN PROVIDE A LOT OF INFORMATION ABOUT THE SEQUENCE OF EVENTS AND HOW THE DEVICE AND LINK ARE BEHAVING.

NOTE: IF BOTH NODES IN ACTIVE AND "/NOCHECK" IS USED, ----- END-OF-PASS IS DEFINED AS RECEIVING 1 MESSAGE AND COMPLETING THE TRANSMIT LIST. WITH NO DATA CHECKING, THERE IS NO WAY FOR DCLT TO KNOW HOW MANY MESSAGES IT SHOULD EXPECT TO RECEIVE.

7.2.1 A SAMPLE TEST SEQUENCE AS SEEN FROM THE KMS11-BD/BE END OF A LINK.

7.2.1.1 INVOKE AND START THE DCLT TEST PROGRAM

BOOT XXDP+

.R CZKMSA
DR>START
CHANGE HW (L) ? Y<CR>

UNITS (D) ? 1<CR>

UNIT 0

FULL DUPLEX ONLY

DEVICE CSR ADDRESS : (0) 164100 ?<CR>
INTERRUPT VECTOR ADDRESS: (0) 400 ?<CR>
INTERRUPT PRIORITY: (0) 5 ?<CR>
DM11-BA CSR ADDRESS (0) 164110 ?<CR> VALUES
THIS IS DCLT. TYPE "H" OR "?" FOR DETAILS
MODE=ACTIVE/PASS=00001

/NOSTATUS/CHECK/NOECHO/NODEM
DCLT> (A) ?

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 37

7.2.1.2 RUN A TEST PASS IN ACTIVE MODE WITH INTERNAL LOOPBACK
WITHOUT CHECKING THE RECEIVED MESSAGE.
PRINT STATUS AS THE PASS PROGRESSES.

DCLT> (A) ? L E=0
DCLT> (A) ? R M=A/L=I/STA/NOCHECK

THIS COMMAND SEQUENCE MEANS:
THE COMMAND L E=0 ESTABLISHES LINE 0 AS THE LINE TO BE USED.
KMS11-BD/BE DCLT CAN TEST ONLY ONE LINE IN A DCLT> RUN COMMAND.
THE PROGRAM CAN BE USED TO TEST ALL EIGHT LINES ONE AT A TIME.
THE LINE TO BE TESTED MUST BE ESTABLISHED BY A DCLT> L E=X
(WHERE X = 0..7) PRIOR TO TYPING THE DCLT> RUN COMMAND.
LINE 0 IS USED HERE AS AN EXAMPLE. REPLACE 0 WITH THE NUMBER
IN THE RANGE 0..7 FOR THE LINE YOU ARE TESTING.

THE COMMAND R M=A/L=I/STA/NOCHECK SETS THE RUN MODE TO ACTIVE, THE
LOOP TYPE TO INTERNAL, REQUESTS STATUS MESSAGE PRINTING AS THE RUN
PROGRESSES AND DISABLES CHECKING OF THE RECEIVED MESSAGE.
THE DEFAULT ITEMP MESSAGE (THE QUICK BROWN...) WILL BE TRANSMITTED.
A RECEIVED MESSAGE WILL BE ACCEPTED BUT NOT CHECKED.

IF THE RUN IS SUCCESSFUL YOU SHOULD SEE:

```
INI  RXQ  TXQ  TXC  EOP
      MODE=ACTIVE/LOOP=INTERNAL/PASS=00000
      /STATUS/NOCHECK/NOECHO/NOMODEM
DCLT> (A) ?
```

A SUCCESSFUL RUN INDICATES THAT THE KMS11-BD/BE CAN TRANSMIT
AND RECEIVE. ANY ERRORS REPORTED WILL PROBABLY BE DUE TO
INCORRECT DEVICE ADDRESSES BEING USED OR A FAULTY DEVICE.
TO CHECK FOR THESE RETURN TO SUPERVISOR COMMAND LEVEL
BY TYPING:

DCLT (A) ? EXIT

OR:

CNTRL/C

CHECK ADDRESSES BY TYPING:

DR> DISPLAY<CR>

RUN KMS11-BD/BE DIAGNOSTICS TO CHECK THE DEVICE.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 38

7.2.1.3 RUN SEVERAL PASSES WITH LOOP TYPE = INTERNAL AND
CHECKING OF THE RECEIVED MESSAGE ENABLED.

DCLT> (A) ? R M=A/L=1/STA/CHECK/PASS=3

THIS COMMAND MEANS:

R M=A

THE DEFAULT (ITEP) MESSAGE WILL BE TRANSMITTED ONCE PER PASS.
ONE MESSAGE WILL BE RECEIVED PER PASS.

/L=1

MESSAGES TRANSMITTED WILL BE LOOPED BACK INTERNALLY WITHIN THE
LINE UNIT AND ACCEPTED AS RECEIVED MESSAGES.

/STA

STATUS MESSAGES WILL BE PRINTED AS THE RUN PROGRESSES.

/CHECK

THE MESSAGE RECEIVED WILL BE CHECKED AGAINST THE DEFAULT (ITEP)
MESSAGE.

/PASS=3

THIS SEQUENCE WILL BE EXECUTED THREE TIMES.

IF SUCCESSFUL YOU SHOULD SEE:

```

INI  RXQ  TXQ  TXC  CMP  EOP  RXQ  TXQ
TXC  CMP  EOP  RXQ  TXQ  TXC  CMP  EOP
MODE=ACTIVE/LOOP=INTERNAL/PASS=00000
/STATUS/CHECK/NOECHO/NOMODEM

```

DCLT> (A) ?

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 39

7.2.1.4 RUN IN ACTIVE MODE WITH A TURNAROUND CONNECTOR

PUT AN H3256 TURNAROUND CONNECTOR ON THE DISTRIBUTION PANEL PLUG FOR THE ESTABLISHED LINE (THE LEFTMOST PLUG AS YOU FACE THE PANEL IS THE PLUG FOR LINE 0).

DCLT> (A) ? R M=A/L=C/PASS=3

THIS IS SIMILAR TO THE INTERNAL LOOPBACK TEST. THE DEFAULT MESSAGE IS TRANSMITTED FROM THE TRANSMIT BUFFER, TURNED AROUND AT THE CONNECTOR, RECEIVED AND CHECKED AGAINST THE EXPECTED DEFAULT MESSAGE. THE COMMAND WILL INITIATE TRANSMISSION ONLY IF A LINE HAS PREVIOUSLY BEEN ESTABLISHED WITH A DCLT> L E=X (WHERE X = 0..7) COMMAND. L=C IN THE COMMAND STANDS FOR LOOP TYPE = CABLE. IT IS INCLUDED AS A RECORD KEEPING NICETY. IN KMS11-BD/BE DCLT DCLT> (A) ? R M=A/L=C IS EQUIVALENT TO DCLT> (A) ? R M=A STATUS AND CHECKING ARE IS STILL ENABLED. THE MODE MUST BE EXPLICITLY SPECIFIED IN EVERY DCLT> RUN COMMAND. LOOPTYPE AND THE STATUS, CHECK AND ECHO QUALIFIERS RETAIN THEIR SETTINGS UNTIL EXPLICITLY CHANGED IN A DCLT> RUN COMMAND OR UNTIL A DR> START COMMAND RESETS ALL DEFAULTS.

IF THIS TEST IS SUCCESSFUL YOU WILL SEE:

```

INI  RXQ  TXQ  TXC  CMP  EOP  RXQ  TXQ
TXC  CMP  EOP  RXQ  TXQ  TXC  CMP  EOP
MODE=ACTIVE/LOOP=CABLE/PASS=00000
      /STATUS/CHECK/NOECHO/NOMODEM
DCLT> (A) ?

```

THE SAME TEST CAN BE REPEATED WITH THE TURNAROUND CONNECTOR AT THE END OF A CABLE PLUGGED TO THE DISTRIBUTION PANEL.

IF THESE TESTS RUN SUCCESSFULLY YOU CAN ASSUME THAT THE HOST COMPUTER ON WHICH KMS11-BD/BE DCLT IS RUNNING AND ITS KMS11-BD/BE ARE TRANSMITTING AND RECEIVING SUCCESSFULLY.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 40

7.2.2 TRANSMIT ON ONE NODE RECEIVE ON THE OTHER

NOW TRY TRANSMITTING FROM ONE END AND RECEIVING ON THE OTHER. MAYBE WITH NO DATA CHECKING AT FIRST TO ESTABLISH IF THE LINK IS WORKING. POSSIBLE COMMAND SEQUENCES ARE:

NODE A	NODE B
-----	-----
DCLT> (A) ? CL E	DCLT> (A) ? CL E
DCLT> (A) ? CL T	DCLT> (A) ? CL T
DCLT> (A) ? SE T=1ALT/S=250	
* DCLT> (A) ? L E=1	
DCLT> (A) ? R M=TR/PAS=3	DCLT> (A) ? R M=R/NOCH/PAS=3

* NOTE: THIS COMMAND IS SPECIFIC TO KMS11-BD/BE DCLT

WHAT THIS SEQUENCE MEANS:

CL E INITIALIZE THE EXPECT LIST
 CL T INITIALIZE THE TRANSMIT LIST
 SE T=1ALT/S=250
 SETS THE TRANSMIT LIST ON NODE A TO BE 1 MESSAGE
 WITH A LENGTH OF 250 BYTES AND DATA OF ALTERNATING
 ONES AND ZEROS.
 L E=1 ESTABLISHES LINE NUMBER 1 AS THE LINE TO TEST.
 R M=TR/PAS=3 SETS THE RUN MODE OF NODE A TO TRANSMIT
 AND THE PASS COUNT TO 3.
 R M=R/NOCH/PAS=3 SETS THE RUN MODE OF NODE B
 TO BE RECEIVE, NO DATA CHECKING IS TO BE DONE, AND
 THE PASS COUNT IS SET TO THREE.

WHAT YOU SHOULD SEE AFTER ENTERING THE RUN COMMAND
 IF THINGS ARE RUNNING CORRECTLY :

FOR NODE A:

```
INI TXQ TXC EOP TXQ TXC EOP TXQ
TXC EOP
MODE=TRANSMIT/PASS=00000
/STATUS/NOCHECK/NOECHO/NOMODEM
DCLT> (A) ?
```

FOR NODE B:

```
INI RXQ EOP RXQ EOP RXQ EOP
MODE=RECEIVE/PASS=00000
/STATUS/NOCHECK/NOECHO/NOMODEM
DCLT> (A) ?
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 41

NOW TRY DOING DATA CHECKING ON THE MESSAGE(S) BEING
TRANSMITTED. POSSIBLE COMMAND SEQUENCES ARE:

R M=TR/PAS=3

SE E=1ALT/S=250
R M=R/CH/PAS=3

WHAT THIS SEQUENCE MEANS:

THE "SE E=1ALT/S=250" LINE MUST BE ADDED HERE
TO SET UP THE "EXPECT" LIST ON THE RECEIVE NODE
SO IT WILL KNOW WHAT TO COMPARE AGAINST.
THE CHANGE IN THE RUN COMMAND IS FROM "NOCH" TO "CH"
THE "CH" ENABLES DATA CHECKING

WHAT YOU SHOULD SEE AFTER ENTERING THE RUN COMMAND
IF THINGS ARE RUNNING CORRECTLY :

NODE A: IS THE SAME AS ABOVE.

NODE B:

```
INI RXQ CMP EOP RXQ CMP EOP RXQ
CMP EOP
MODE=RECEIVE/PASS=00000
/STATUS/NOCHECK/NOECHO/NODEM
DCLT> (A) ?
```

NOW RUN THRU THE SEQUENCE AGAIN WITH NODE A RECEIVING
AND NODE B TRANSMITTING TO CHECK OUT THE OPPOSITE DIRECTION
OF DATA FLOW.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 42

7.2.3

ONE NODE ACTIVE THE OTHER NODE PASSIVE

NOW TRY RUNNING ONE NODE IN ACTIVE MODE WHILE THE OTHER
END RUNS IN PASSIVE. DATA CHECKING SHOULD BE TURNED OFF
IF THE MESSAGE LISTS ARE NOT THE SAME.
POSSIBLE COMMAND SEQUENCES ARE:

NODE A	NODE B
-----	-----
DCLT> (A) ? CL E	DCLT> (A) ? CL E
DCLT> (A) ? CL T	DCLT> (A) ? CL T
DCLT> (A) ? SE T=CCITT/S=10/C=2	DCLT> (A) ? SE T=1ALT/S=20/C=2
* DCLT> (A) ? L E=1	
DCLT> (A) ? R M=ACT/NOCH/PAS=3	DCLT> (A) ? R M=P/NOCH/PAS=3

* NOTE: THIS COMMAND IS SPECIFIC TO THE KMS11-BD/BE

WHAT THIS SEQUENCE MEANS:

THE EXECUTION OF THIS SEQUENCE CAUSES THE FOLLOWING
THINGS TO HAPPEN ON NODE A. THE TRANSMIT AND EXPECT
LISTS ARE INITIALIZED THEN THE TRANSMIT LIST IS SET
TO 3 MESSAGES OF 10 BYTES EACH. THE DATA USED IN THE
TRANSMIT MESSAGES IS THE CCITT PATTERN. LINE 1 IS
ESTABLISHED AS THE LINE TO TEST.
THEN NODE A IS RUN IN ACTIVE MODE
WITH DATA CHECKING DISABLED AND THE PASS COUNT
SET TO THREE. NOTE STATUS WOULD STILL BE
PRINTED IF THE PREVIOUS SEQUENCES HAD BEEN RUN,
IF YOU ARE RUNNING FROM LOAD TIME YOU WOULD HAVE
TO ADD A "/STA TO THE RUN COMMAND LINE.
NODE B: THE TRANSMIT AND EXPECT LISTS ARE INITIALIZED
THEN THE TRANSMIT LIST IS SET TO 3 MESSAGES OF
20 BYTES EACH. THE DATA FOR EACH MESSAGE IS ALTERNATING
1'S AND 0'S.
RUN IN PASSIVE MODE WITH DATA CHECKING DISABLED
AND THE PASS COUNT SET TO 3.

WHAT YOU SHOULD SEE AFTER ENTERING THE RUN COMMAND
IF THINGS ARE RUNNING CORRECTLY :

FOR NODE A:

```
INI RXQ TXQ TXC TXQ RXQ TXC TXQ
RXQ TXC EOP RXQ TXQ RXC TXC TXQ
RXQ TXC TXQ RXQ TXC EOP RXQ TXQ
RXQ TXC TXQ RXQ TXC TXQ RXQ TXC
EOP
MODE=ACTIVE/PASS=00000
/STATUS/NOCHECK/NOECHO/NODEM
DCLT> (A) ?
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 43

FOR NODE B:

```
INI RXQ TXQ TXC RXQ TXQ TXC RXQ
TXQ TXC EOP
MODE=PASSIVE/PASS=00000
/STATUS/NOCHECK/NOECHO/NOMODEM
DCLT> (A) ?
```

NOW USE DATA CHECKING WITH THE "EXPECT MESSAGE LISTS" SET UP APPROPRIATELY. ANOTHER VARIATION IS TO HAVE LARGE SIZE MESSAGES ON ONE SIDE WITH SMALL MESSAGES ON THE OTHER.

THEN REVERSE THE SETUP SO THAT THE NODE RUNNING IN ACTIVE IS RUNNING IN PASSIVE AND VICE VERSA.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 44

7.2.4 BOTH NODES ACTIVE

NOW BOTH NODES CAN BE RUN IN ACTIVE WITH DATA CHECKING ON.
STATUS PRINTING COULD BE TURNED OFF IF YOU'RE NOT INTERESTED
IN THEM.

NODE A -----	NODE B -----
DCLT> (A) ? CL E	DCLT> (A) ? CL E
DCLT> (A) ? CL T	DCLT> (A) ? CL T
DCLT> (A) ? SE T=OALT/S=10	DCLT> (A) ? SE E=OALT/S=10
DCLT> (A) ? SE T=CCITT/S=20	DCLT> (A) ? SE E=CCITT/S=20
DCLT> (A) ? SE T=ALPHA/S=30	DCLT> (A) ? SE E=ALPHA/S=30
DCLT> (A) ? SE E=ZERO/S=11	DCLT> (A) ? SE T=ZERO/S=11
DCLT> (A) ? SE E=ONES/S=21	DCLT> (A) ? SE T=ONES/S=21
DCLT> (A) ? SE E=ITEP/S=31	DCLT> (A) ? SE T=ITEP/S=31
DCLT> (A) ? R M=A/CH/NOST/PAS=3	DCLT> (A) ? R M=A/CH/NOST/PAS=3

WHAT THIS SEQUENCE MEANS:

NODE A SETS UP ITS TRANSMIT LIST TO BE
3 MESSAGES. MESSAGE 1 IS 10 BYTES LONG AND
CONTAINS DATA OF ALTERNATING 0'S AND 1'S
MESSAGE 2 IS 20 BYTES LONG AND CONTAINS
DATA OF THE CCITT PATTERN. MESSAGE THREE
IS 30 BYTES LONG AND CONTAINS ALPHANUMERIC
FOR DATA. THE EXPECT LIST ALSO CONTAINS
3 MESSAGES. MESSAGE 1 IS 11 BYTES LONG AND
CONTAINS 0'S FOR DATA. MESSAGE TWO IS 21
BYTES LONG AND CONTAINS 1'S FOR DATA. MESSAGE
3 IS 31 BYTES LONG AND CONTAINS THE ITEP DATA.
NODE B HAS THE SAME MESSAGES EXCEPT THAT THE
TRANSMIT MESSAGE LIST IS THE EXPECT MESSAGE LIST
AND VICE VERSA.
BOTH NODES ARE RUN IN THE ACTIVE MODE WITH NO
DATA CHECKING AND PASS COUNT EQUAL TO THREE.

WHAT YOU SHOULD SEE AFTER ENTERING THE RUN COMMAND
IF THINGS ARE RUNNING CORRECTLY :

ON BOTH NODES A AND B:

```
MODE=ACTIVE/PASS=00000
/NOSTATUS/CHECK/NOECHO/NODEM
```

DCLT> (A) ?

A VARIATION THAT CAN BE USED IS FOR ONE END TO SEND A LOT OF
SMALL MESSAGES AND THE OTHER TO SEND A FEW LARGE MESSAGES.
THE "END-OF-PASS" POINT WILL BE OUT OF SYNC BUT THIS IS NOT
A PROBLEM.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 45

7.2.5 TALK AND LISTEN MODES FOR COMMUNICATING

TALK AND LISTEN MODES ARE USEFUL IF THE OPERATORS WISH TO COMMUNICATE WITH EACH OTHER. JUST SETUP A TIME THAT EACH WILL GO TO THEIR MODE, TALK OR LISTEN, AND SEND MESSAGES OVER THE LINK. POSSIBLE COMMAND SEQUENCES ARE. WHEN USING TALK AND LISTEN MODES ON MULTIPOINT LINKS REMEMBER THAT YOU CAN ONLY USE THESE MODES FROM THE CONTROL STATION TO THE FIRST TRIBUTARY IN THE TRIB LIST.

DCLT> (A) ? R M=LIS/NOST
LIS>

DCLT> (A) ? R M=TA/NOST
TLK>

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 46

7.3 EXAMPLES OF COMMANDS

THIS SECTION WILL SHOW A SAMPLING OF COMMANDS AND EXACTLY WHAT TO EXPECT FROM THEM.

7.3.1 EXAMPLES OF MESSAGE COMMANDS

THE CLEAR COMMANDS .

```
DCLT> (A) ? CL E
DCLT> (A) ? CL T
THIS WILL INITIALIZE THE TRANSMIT AND EXPECT LIST
TO 1 MESSAGE OF 58 BYTES. THE DATA OF THE MESSAGE WILL
BE THE ITEP MESSAGE.
```

IF THESE COMMANDS ARE FOLLOWED BY A SHOW COMMAND

```
DCLT> (A) ? SH E
SUCH AS THE SHOW EXPECT LIST, WHAT YOU WOULD SEE IS
```

```
MSG: TYPE=ITEP/SIZE=58
MODE=ACTIVE/PASS=00001
/NOSTATUS/CHECK/NOECHO/NOMODEM
```

NOW IF YOU DID A SET EXPECT LIST COMMAND SUCH AS:

```
DCLT> (A) ? SE E=A/S=35/C=3
AND FOLLOWED IT WITH A SHOW EXPECT LIST COMMAND
```

```
DCLT> (A) ? SH E
WHAT YOU WOULD SEE IS
MSG: TYPE=ALPHA/SIZE=35
MSG: TYPE=ALPHA/SIZE=35
MSG: TYPE=ALPHA/SIZE=35
MSG: TYPE=ALPHA/SIZE=35
MODE=ACTIVE/PASS=00001
/NOSTATUS/CHECK/NOECHO/NOMODEM
DCLT> (A) ?
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 47

7.3.2 EXAMPLES OF LINE SELECTION COMMANDS

WHEN THE DCLT> COMMAND LEVEL IS ENTERED FROM DR>START
NO LINE TO TEST IS ESTABLISHED.

IF A LINE SHOW COMMAND IS TYPED: DCLT> (A) ? L SH
YOU WOULD SEE: NO LINE NUMBER IS ESTABLISHED

DCLT> (A) ?

TO ESTABLISH A LINE TO TEST TYPE:

DCLT> (A) ? L E=N (WHERE N IS IN THE RANGE 0..7)

FOR EXAMPLE AFTER: DCLT> (A) ? L E=5
A LINE SHOW COMMAND DCLT> (A) ? L SH
WILL PRINT LINE NUMBER: 5

ANY TESTS INVOKED BY SUBSEQUENT RUN COMMANDS WILL APPLY TO
LINE 5. A LINE NUMBER REMAINS ESTABLISHED UNTIL IT IS CHANGED
BY A LINE ESTABLISH OR A LINE KILL COMMAND OR UNTIL THE DCLT>
COMMAND LEVEL IS INVOKED BY A SUPERVISOR DR>START COMMAND

THE LINE KILL COMMAND

DCLT> (A) ? L K

LEAVES THE KMS11-BD/BE DCLT WITH NO ESTABLISHED LINE.

THE COMMAND SEQUENCE:

DCLT> (A) ? L K

DCLT> (A) ? L SH

PRINTS:

NO LINE NUMBER IS ESTABLISHED

IT IS NOT NECESSARY TO KILL AN ESTABLISHED LINE BEFORE
ESTABLISHING A NEW LINE NUMBER.

DCLT> (A) ? L SH
LINE NUMBER: 5
DCLT> (A) ? L E=3
DCLT> (A) ? L SH
LINE NUMBER: 3

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 48

7.3.3 EXAMPLES OF OPERATOR ASSISTANCE COMMANDS

IF YOU TYPE A HELP COMMAND:

EITHER DCLT> (A) ? HELP OR DCLT> (A) ? ?

YOU WILL SEE:

DCLT CMDS:

CLEAR OR SHOW EXPECTLIST OR TRANSMITLIST

PRINT OR EXIT

DUMP START-END/B

LINE SHOW, LINE ESTABLISH=N (N=0-7)OR LINE KILL

SET EXPECTMSG OR TRANSMITMSG=TYPE/SIZE=N OR /COPY=N

TYPE=ONES,ZEROES,1ALT,0ALT,ITEP,CCITT,ALPHA

OR "OPR SPCD=A-Z,SP,TAB,0-9 IN QUOTES"

SET EXPECT=TRANSMIT

RUN MODE=MTYP/LOOP=LTP/CHECK,STATUS,ECHO,MODEM,PASS=N

MTYP=TRAN,REC,ACT,PAS,TAL,LIS,DOWN

LTP=INT,CAB,LOC,REM/

DCLT> (A) ?

THE DUMP COMMAND:

DCLT> (A) ? DUMP 41260-41300

WILL PRINT THE CONTENTS OF PDP-11 MEMORY ADDRESSES
41260 THROUGH 41300 IN THE FOLLOWING MANNER:

```
41260 104423 000167 177772 021122 012112 006312 006312 006312
41300 006312
```

WITH THE /B SWITCH:

DCLT> (A) ? DUM 41260-41300/B

THE DUMP COMMAND PRINTS:

```
41260 023 211 167 000 372 377 122 024
41270 112 024 312 014 312 014 312 014
41300 312
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 49

7.3.4 EXAMPLES OF RUN COMMMANDS

YOU CAN FIND SEVERAL EXPAMLES OF THE RUN COMMMAND IN THE TROUBLE SHOOTING HINTS SECTION BUT HERE ARE SOME OTHERS.

IF YOU TYPE IN THE RUN COMMAND

```
DCLT> (A) ? R M=TR/NOST/CH/PAS=4
```

THE PROGRAM WILL EXECUTE 4 PASSES AND THEN PRINT

```
MODE=TRANSMIT/PASS=00000  
/NOSTATUS/CHECK/NOECHO/NOMODEM
```

```
DCLT> (A) ?
```

IF YOU EXECUTE THE RUN COMMAND

```
DCLT> (A) ? R M=A/LO=I/ST/CH/PAS=3
```

WITH DEFAULT TRANSMIT AND EXPECT MESSAGES YOU SHOULD SEE:

```
INI RXQ TXQ TXC CMP EOP RXQ TXQ  
TXC CMP EOP RXQ TXQ TXC CMP EOP  
MODE=ACTIVE/LOOP=INTERNAL/PASS=0000  
/STATUS/CHECK/NOECHO/NOMODEM
```

```
DCLT> (A) ?
```

IN RESPONSE TO THE EXIT COMMAND

```
DCLT> (A) ? EXIT
```

YOU WILL SEE :

```
CZKMSA EOP  
0 CUMLATIVE ERRORS
```

```
DR>
```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 51

THE STATUSONLINE COMMAND PRINTS THE STATUS OF THE ESTABLISHED
LINE. IF THERE IS AN ESTABLISHED LINE THEN

RPT> (A) ? STATUSONLINE

PRINTS:

```
STATUS REPORT
LINE NUMBER = XXX
STATUS      = XXX   FLAGS = XXX
TEMP DATA = XXX   LAST TX MSG = XXX
ACKS RCVD  = XXX   MSGS RCVD = XXX
NAKS SENT  = XXX   NAKS RCVD = XXX
```

THE EXIT COMMAND RETURNS TO THE COMMAND LEVEL FROM WHICH
REPORT PRINTING WAS INVOKED.

IF REPORT PRINTING WAS INVOKED FROM THE SUPERVISOR
VIA:

DR> PRINT

THEN:

RPT> (A) ? EXIT

IS FOLLOWED BY:

DR>

IF REPORT PRINTING WAS INVOKED FROM DCLT
VIA:

DCLT> (A) ?

THEN:

RPT> (A) ? EXIT

IS FOLLOWED BY:

DCLT> (A) ?

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 52

7.4 THINGS TO WATCH OUT FOR

IF YOU ARE RUNNING DCLT ON SYSTEMS THAT HAVE CONSOLES WITH DIFFERENT SPEEDS YOU WILL BE UNABLE TO USE THE PRINT STATUS FEATURE IN CERTAIN MODES. THE RULE IS IF IT DOESNT WORK WITH STATUS PRINTING RUN THE MODE WITH NOSTATUS.

IF YOU ARE USING PASSVIE MODE WITH THE ECHO SWITCH THEN YOU WILL PROBABLY HAVE TO RE ENTER THE TRANSMIT LIST ON THE SIDE WITH THE ECHO SWITCH. THE REASON IS THAT THE TRANSMIT LIST GETS OVER WRITTEN WITH THE RECEIVE LIST WHEN USING THE ECHO SWITCH

&

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 53

1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048

002000

002000

002000

002000

002000

002001

002002

002003

002004

002005

002006

002007

002010

002011

002011

002012

002012

002014

002014

002016

002016

002020

002020

002022

002022

002024

002024

002026

002026

002030

002030

002032

002032

002034

002034

002036

103
132
113
115
123
000
000
000

101
060

000000

003410

067230

000000

002130

000000

067542

000000

000000

000000

.SBTTL PROGRAM HEADER

BGNMOD

::++
: THE PROGRAM HEADER IS THE INTERFACE BETWEEN
: THE DIAGNOSTIC PROGRAM AND THE SUPERVISOR.
:--

POINTER ERR TBL ,BGNRPT ,BGNAU ,BGNDU

HEADER CZKMS ,A ,0 ,1800 . ,G ,PRI07

LSNAME ::
 .ASCII /C/
 .ASCII /Z/
 .ASCII /K/
 .ASCII /M/
 .ASCII /S/
 .BYTE 0
 .BYTE 0
 .BYTE 0
LSREV ::
 .ASCII /A/
LSDEPO ::
 .ASCII /O/
LSUNIT ::
 .WORD 0
LSTIML ::
 .WORD 1800.
LSHPCP ::
 .WORD LSHARD
LSSPCP ::
 .WORD 0
LSHPTP ::
 .WORD LSHW
LSSPTP ::
 .WORD 0
LSLADP ::
 .WORD LSLAST
LSSTA ::
 .WORD 0
LSCO ::
 .WORD 0
LSDTYP ::
 .WORD 0
LSAPT ::

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.F11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 54
PROGRAM HEADER

2049	002036	000000
2050	002040	
2051	002040	002124
2052	002042	
2053	002042	000340
2054	002044	
2055	002044	000000
2056	002046	
2057	002046	000000
2058	002050	
2059	002050	003
2060	002051	003
2061	002052	
2062	002052	000000
2063	002054	000000
2064	002056	
2065	002056	000000
2066	002060	
2067	002060	013714
2068	002062	
2069	002062	032576
2070	002064	
2071	002064	000000
2072	002066	
2073	002066	000000
2074	002070	
2075	002070	054156
2076	002072	
2077	002072	054150
2078	002074	
2079	002074	000000
2080	002076	
2081	002076	013730
2082	002100	
2083	002100	104035
2084	002102	
2085	002102	013704
2086	002104	
2087	002104	032612
2088	002106	
2089	002106	054022
2090	002110	
2091	002110	054020
2092	002112	
2093	002112	032604
2094	002114	
2095	002114	000000
2096	002116	
2097	002116	000000
2098	002120	
2099	002120	000000
2100		

LSDTP::	.WORD	0
LSPRIO::	.WORD	LSDISPATCH
LSENV1::	.WORD	PRI07
LSEXP1::	.WORD	0
LSMREV::	.WORD	0
	.BYTE	CSREVISION
	.BYTE	CSREDIT
LSEF::	.WORD	0
	.WORD	0
LSSPC::	.WORD	0
L\$DEVP::	.WORD	0
LSREPP::	.WORD	LSDVTYP
LSEXP4::	.WORD	LSRPT
LSEXP5::	.WORD	0
LSAUT::	.WORD	0
LSDUT::	.WORD	LSAU
LSLUN::	.WORD	LSDU
L\$DESP::	.WORD	0
L\$LOAD::	.WORD	L\$DESC
L\$ETP::	EMT	E\$LOAD
LSICP::	.WORD	LSERRTBL
L\$CCP::	.WORD	L\$INIT
L\$ACP::	.WORD	L\$CLEAN
L\$PRT::	.WORD	L\$AUTO
L\$TEST::	.WORD	L\$PROT
L\$DLY::	.WORD	0
L\$HIME::	.WORD	0
	.WORD	0

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 55
DISPATCH TABLE

2101
2102
2103
2104
2105
2106
2107
2108
2109
2110

002122
002122 000001
002124
002124 054164

::++
: THE DISPATCH TABLE CONTAINS THE STARTING ADDRESS OF EACH TEST.
: IT IS USED BY THE SUPERVISOR TO DISPATCH TO EACH TEST.
:--

DISPATCH 1

.WORD 1
L\$DISPATCH::
.WORD 11

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 56
DEFAULT HARDWARE P-TABLE

```

2111
2112
2113
2114
2115
2116
2117
2118 002126
2119 002126 000006
2120 002130
2121 002130
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136 002130 164100
2137 002132 000400
2138 002134 000240
2139 002136 164110
2140 002140 000000
2141 002142 000000
2142
2143
2144 002144
2145 002144

```

```

:++
: THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
: THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
: IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
: AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
:--

```

```

          BGNHW  DFPTBL
                                   .WORD  L10000-L$HW/2
                                   DFPTBL::

```

```

:INDEPENDENT SECTION
: THE NUMBERS IN BRACKETS ARE THE OFFSET VALUES USED IN THE PARAMETER
: CODING SECTION.

```

```

:DEVICE DEPENDENT SECTION
: ADDING OR REMOVING WORDS FROM THIS TABLE EFFECTS THE "GET" CALLS IN
: THE HARDWARE PARAMTER CODING SECTION BY CHANGING "OFFSETS"

```

```

          .WORD  164100      :[0] CSR ADDRESS (KMC11-B)
          .WORD   400       :[2] INTERRUPT VECTOR (KMC11-B)
          .WORD   240       :[4] INTERRUPT PRIORITY (BR5) (KMC11-B)
          .WORD  164110     :[6] CSR ADDRESS (DM11-BA MODEM CONTROL (CSR)
          .WORD    0        :[10] SPARE
          .WORD    0        :[12] SPARE

```

```

          ENDPHW
                                   L10000:

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 57
DEFAULT HARDWARE P-TABLE

2146
2147
2148

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 58
GLOBAL EQUATES SECTION

2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204

002144

100000
040000
020000
010000
004000
002000
001000
000400
000200
000100
000040
000020
000010
000004
000002
000001

001000
000400
000200
000100
000040
000020
000010
000004
000002
000001

000040
000037
000036
000035
000034

000340
000300
000240

..++
: THE GLOBAL EQUATES SECTION CONTAINS PROGRAM EQUATES THAT
: ARE USED IN MORE THAN ONE TEST.
:--

EQUALS

: BIT DEFINITIONS

BIT15== 100000
BIT14== 40000
BIT13== 20000
BIT12== 10000
BIT11== 4000
BIT10== 2000
BIT09== 1000
BIT08== 400
BIT07== 200
BIT06== 100
BIT05== 40
BIT04== 20
BIT03== 10
BIT02== 4
BIT01== 2
BIT00== 1

BIT9== BIT09
BIT8== BIT08
BIT7== BIT07
BIT6== BIT06
BIT5== BIT05
BIT4== BIT04
BIT3== BIT03
BIT2== BIT02
BIT1== BIT01
BIT0== BIT00

: EVENT FLAG DEFINITIONS
: EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION

EF.START== 32.
EF.RESTART== 31.
EF.CONTINUE== 30.
EF.NEW== 29.
EF.PWR== 28.

: START COMMAND WAS ISSUED
: RESTART COMMAND WAS ISSUED
: CONTINUE COMMAND WAS ISSUED
: A NEW PASS HAS BEEN STARTED
: A POWER-FAIL/POWER-UP OCCURRED

: PRIORITY LEVEL DEFINITIONS

PRI07== 340
PRI06== 300
PRI05== 240

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 59
GLOBAL EQUATES SECTION

2205	000200	PRI04==	200
2206	000140	PRI03==	140
2207	000100	PRI02==	100
2208	000040	PRI01==	40
2209	000000	PRI00==	0
2210		.	
2211		.;OPERATOR FLAG BITS	
2212		.	
2213	000004	EVL==	4
2214	000010	LOT==	10
2215	000020	ADR==	20
2216	000040	IDU==	40
2217	000100	ISR==	100
2218	000200	UAM==	200
2219	000400	BOE==	400
2220	001000	PNT==	1000
2221	002000	PRI==	2000
2222	004000	IXE==	4000
2223	010000	IBE==	10000
2224	020000	IER==	20000
2225	040000	LOE==	40000
2226	100000	HOE==	100000
2227			

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 60
GLOBAL EQUATES SECTION

```

2228           ; INDEPENDENT EQUATES
2229
2230           000000           ENUM = 0           ; INITIAL VALUE FOR ERROR NUMBER
2231
2232           001000           BUFLIM=512.        ; MAX BUFFER SIZE IN BYTES
2233           ; APPLIES TO TX,RX AND CMP BUFFS
2234           000017           MSGLIM=15.         ; MAX NO. OF MESSAGES PER BUFFER
2235           ; (FOR EACH INCREMENT (+1) TO MSGLIM,
2236           ; ADD 6 WORDS TO THE POINTER TABLE
2237           ; (PTRTAB:) SINCE THIS MEANS 2 MORE
2238           ; "POINTER" WORDS PER BUFFER.
2239
2240
2241           ;MODE OF OPERATION EQUATES
2242           000000           REC=0               ; RECEIVE MODE
2243           000001           TRA=1               ; TRANSMIT MODE
2244           000002           PAS=2               ; PASSIVE MODE
2245           000003           ACT=3               ; ACTIVE MODE
2246           ;KMS11-BD,BE DCLT DOES NOT SUPPORT DOWN-LINE-LOAD MODE
2247           000004           DOW=4               ; DOWN-LINE-LOAD MODE
2248           000005           TAL=5               ; TALK MODE
2249           000006           LIS=6               ; LISTEN MODE
2250
2251           ;MAINT LOOP TYPE EQUATES
2252
2253           000000           NONE= 0              ; NO LOOP
2254           000001           TTL= 1              ; INTERNAL TTL
2255           000002           CABLE= 2            ; CABLE LOOP
2256
2257           ;KMS11-BD,BE DCLT DOES NOT SUUPORT MODEM LOCAL,MODEM REMOTE OR MOP
2258           000003           MODLOC= 3           ; MODMEM LOCAL
2259           000004           MODREM= 4           ; MODEM REMOTE
2260           000005           MOP= 5              ; MOP
2261
2262
2263           ;CLOCK ENABLE VALUES TO BE LOADED IN CLK'S CSR
2264
2265           000100           LCLKEN= 100         ; L-CLOCK CSR VALUE TO ENABLE THE CLOCK
2266           000111           PCLKEN= 111         ; P-CLOCK CSR VALUE TO ENABLE THE CLOCK
2267           001600           PCLKCT= 1600        ; P-CLOCK COUNT SET REGISTER FOR COUNTER
2268
2269           ;PARAM WORD EQUATES
2270
2271           000001           STATB= BIT0          ; OPERATOR AWAKE ASKED FOR
2272           000002           DATCKB= BIT1         ; DATA CHECK BIT
2273           000004           ECHOB= BIT2          ; ECHO BIT
2274           000010           MOCHK= BIT3         ; MODEM CHECK/NO CHECK
2275           000020           CRCB= BIT4          ; CRC CALCULATE ASKED FOR
2276           000040           PROTOB= BIT5        ; PROTOCOL PROCESSING ASKED FOR
2277

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 61
GLOBAL EQUATES SECTION

2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326

000000
000002
000004
000006
000010
000012
000014
000016
000020
000022
000024
000026

000001
000002
000004
000010
000100
000200
000400
001000
002000
004000

000000
000001
000002
000003
000004
000005
000006
000007
000010
000011
000012

:OPTION TYPE EQUATES

:EVENT LOG MESSAGE TYPES (USED TO LOCATE EVENT DESCRIPTION IN EVENT TABLE
: AND DISPATCHING TO SEPARATE SECTIONS OF THE EVENT REPORTING SECTION)

TXQ= 0 :TRANSMIT MESSAGE QUEUED
TXC= 2 :TRANSMIT COMPLETE
RXQ= 4 :RECEIVE BUFFER QUEUED
RXC= 6 :RECEIVE COMPLETE
DER= 10 :DEVICE INFORMATION
DVI= 12 :DEVICE ABOUT TO INIT
DCK= 14 :DATA COMPARISON RESULTS
MSC= 16 :MODEM STATUS CHANGE
DLE= 20 :DATA COMPARISON LENGTH ERROR
DDE= 22 :DATA COMPARISON DATA ERROR
EOP= 24 :END OF PASS
ABO= 26

:EQUATES FOR FLAG WORD

ININT = BIT0 :INPUT INT. REC.
OTINT = BIT1 :OUTPUT INT REC
QRX = BIT2 :RX QUED /COMPL
QTX = BIT3 :TX QUED/COMPL
ERX = BIT6 :EXPECT TO GET A RX COMPLETED
ETX = BIT7 :EXPECT TO GET A TX COMPLETED
INITC = BIT8 :EXPECT INIT SYSTEM COMPLETED
SETC = BIT9 :EXPECT SET LINE COMPLETED
STAT = BIT10 :STATUS REPORT EXPECTED
TXTO = BIT11 :TRANSMIT TIMEOUT OCCURRED

: SPECIAL CLI CODES FOR "CHAR" ARGUMENT IN CLI CALLS
:(COMMAND LINE INTERPRETER DEFINITIONS)

CLIERR= 0
CLIEXI= 1
CLIBR= 2
CLIBIF= 3
CLISPA= 4
CLINUM= 5
CLIALP= 6
CLIALN= 7
CLIOCT= 8.
CLIDEC= 9.
CLISTR= 10.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 62
GLOBAL EQUATES SECTION

: DEFS FOR COMMAND LINE INTERPRETATION ACTION VALUES

2327		
2328	000000	NULL=0
2329	000001	CLEAR=1
2330	000002	SHOW=2
2331	000003	CHECK=3
2332	000004	RUN=4
2333	000005	HLP=5
2334	000006	CSHEXP=6
2335	000007	CSHTRN=7
2336	000010	SETEXP=10
2337	000011	SETTRN=11
2338	000012	SIZE=12
2339	000013	QCOPY=13
2340	000014	NUM=14
2341	000015	OPRMSG=15
2342	000016	STATUS=16
2343	000017	ENDQO=17
2344	000020	CMSG0=20
2345	000021	CMSG1=21
2346	000022	CMSG2=22
2347	000023	CMSG3=23
2348	000024	CMSG4=24
2349	000025	CMSG5=25
2350	000026	CMSG6=26
2351	000027	ATVMOD=27
2352	000030	PASMOD=30
2353	000031	RECMOD=31
2354	000032	LISMOD=32
2355	000033	DLLMOD=33
2356	000034	TRAMOD=34
2357	000035	TALMOD=35
2358	000036	NO=36
2359	000037	ECHO=37
2360	000040	CRC=40
2361	000041	PROTO=41
2362	000042	PASC=42
2363	000043	MOP=43
2364	000044	TTLLOP=44
2365	000045	CBLLOP=45
2366	000046	LMOLOP=46
2367	000047	RMDLOP=47
2368	000050	NOTNUF=50
2369	000051	BADCHR=51
2370	000052	DMPS=52
2371	000053	DMPE=53
2372	000054	DMPQ=54
2373	000055	PRNT=55
2374	000056	MOSC=56
2375	000057	EXIT=57
2376	000060	SETET=60
2377	000061	SLST=61
2378	000062	ETRB=62
2379	000063	KTRB=63
2380	000064	KALL=64
2381	000065	EKTB=65
2382		

2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393

000001
000002
000003
000004
000005
000006
000007
000010

:FOLLOWING EQUATES USED IN REPORT CLI

RPHLP=1
RPEXT=2
RPLOG=3
RPSOL=4
RNOTNF=5
RDMP5=6
RDMPQ=7
RDMPE=10

2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443

000000
000000
000000
000000
000000
000000
000000
000000
100000
040000
010000
004000
002000
001000
000400
000200
000020
000001
000200
000020
007726

```
: DEVICE DEPENDENT EQUATES
: MODEM SIGNAL BIT DEFINITIONS
:   IF SIGNAL AVAILABLE IN DEVICE, EQUATE NAME TO BIT POSITION,
:   ELSE EQUATE IT TO = 0
: (KMS11-C FIRMWARE DOES NOT MAKE MODEM SIGNALS AVAILABLE)
      CTS= 0           ;CLEAR TO SEND (CIRCUIT CB)
      DSR= 0           ;DATA SET READY (CIRCUIT CC)
      DCD= 0           ;DATA CARRIER DETECT (CIRCUIT CF)
      RTS= 0           ;REQUEST TO SEND (CIRCUIT CA)
      RI= 0            ;RING INDICATOR (CIRCUIT CE)
      SQD= 0           ;SIGNAL QUALITY DETECT (CIRCUIT CG)
      TM= 0            ;MODEM IN TEST MODE (RS 449 ONLY CIRCUIT TM)

: DEVICE BIT DEFINITIONS
:KMS11-BD/BE REGISTER BIT DEFINITIONS
:CSR0:
      KRUN= BIT15      ;RUN BIT
      MCLR= BIT14      ;MASTER CLEAR
              ;BIT13   ;UNDEFINED
      EXT1= BIT12      ;STEP LINE UNIT(OPTIONAL. USED WITH SOME LINE UNITS)
      EXT0= BIT11      ;LINE UNIT LOOP (OPTIONAL. USED WITH SOME LINE UNITS)
      RAM0= BIT10      ;LOAD VERIFY RAM
      RAM1= BIT9       ;MAINTENANCE INSTRUCTION REGISTER CLEARS PC
              ;AND LOADS MAINT.INSTRUCTION INTO CRAM LOCATION ZERO
      STEP= BIT8       ;SINGLE STEP MICROPROCESSOR
      RQI= BIT7        ;REQUEST PERMISSION TO TRANSFER COMMAND TO DEVICE
      IE0= BIT4        ;OUTPUT INTERRUPT ENABLE
      IE1= BIT0        ;INPUT INTERRUPT ENABLE

:CSR2: USED FOR COMMUNICATION BETWEEN COMM DEVICE AND ITS LOCAL HOST PROCESSOR
      RDY0= BIT7       ;READY OUT BIT.
              ;SET BY DEVICE TO REQUEST HOST TO ACCEPT MESSAGE
              ;CLEARED BY LOCAL HOST WHEN IT HAS ACCEPTED MESSAGE
      RDY1= BIT4       ;READY IN BIT
              ;SET BY DEVICE TO GRANT LOCAL HOST PERMISSION
              ; TO TRANSFER COMMAND TO DEVICE
              ;CLEARED BY LOCAL HOST WHEN COMMAND HAS BEEN
              ; TRANSFERRED TO DEVICE CSRS

:KMLTYP - ADDRESS IN KMC DATA RAM OF INTERNAL LOOPBACK MODE SPECIFIER
      KMLTYP= 7726    ;***THIS VALUE IS A KMC DATA ADDRESS
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 65
GLOBAL EQUATES SECTION

2444
2445
2446
2447
2448
2449
2450
2451
2452
2453

000000
000000
000001
000002
000003

: MISCELLANEOUS EQUATES FOR ERROR CALLS

ENUM = 0
T.ESF = 0
T.EDF = 1
T.EHRD = 2
T.ESFT = 3

:INITIALIZE ERROR NUMBER
:SYSTEM FATAL ERROR TYPE
:DEVICE FATAL ERROR TYPE
:UNRECOVERABLE ERROR TYPE
:SOFT ERROR TYPE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 66
GLOBAL DATA SECTION

2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478

:++
: THE GLOBAL DATA SECTION CONTAINS DATA THAT ARE USED
: IN MORE THAN ONE TEST.
:--

.SBTTL DEFAULT MESSAGE DEFINITIONS AND TABLES
:MESSAGE BYTE COUNT TABLE

DMSGCT:			
MSG0C:	.WORD	EMSG0-MSG0	:BYTE COUNT OF MESSAGE #0
MSG1C:	.WORD	EMSG1-MSG1	:BYTE COUNT OF MESSAGE #1
MSG2C:	.WORD	EMSG2-MSG2	:BYTE COUNT OF MESSAGE #2
MSG3C:	.WORD	EMSG3-MSG3	:BYTE COUNT OF MESSAGE #3
MSG4C:	.WORD	EMSG4-MSG4	:BYTE COUNT OF MESSAGE #4
MSG5C:	.WORD	EMSG5-MSG5	:BYTE COUNT OF MESSAGE #5
MSG6C:	.WORD	EMSG6-MSG6	:BYTE COUNT OF MESSAGE #6
OPCNT:	.WORD	0	:BYTE COUNT FOR OPERATOR SPEC'D MSG.
MSG8C:	.WORD	EMSG8-MSG8	:BYTE COUNT OF RECEIVE BUFFER FILL PATTERN
DLLM1C:	.WORD	DLLM1E-DLLM1	:DLL MSG 1 COUNT
DLLM2C:	.WORD	DLLM2E-DLLM2	:DLL MSG 2 COUNT

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 67
DEFAULT MESSAGE DEFINITIONS AND TABLES

;MESSAGE ADDRESS TABLE

2479				
2480				
2481	002172			
2482	002172	002214		
2483	002174	002215		
2484	002176	002216		
2485	002200	002217		
2486	002202	002220		
2487	002204	002320		
2488	002206	002412		
2489	002210	002520		
2490	002212	002542		
2491				
2492	002214	000		
2493	002215			
2494	002215	377		
2495	002216			
2496	002216	252		
2497	002217			
2498	002217	125		
2499	002220			
2500	002220			
2501	002220	177603	157427	031011
2502	002226	047321	163715	105221
2503	002234	143325	142304	
2504	002240	040041	014116	052606
2505	002246	172334	105025	123754
2506	002254	111337	111523	
2507	002260	030030	145064	137642
2508	002266	143531	063617	135075
2509	002274	066730	026575	
2510	002300	052012	053627	070071
2511	002306	151172	165044	031605
2512	002314	166632	016741	
2513	002320			
2514	002320			
2515				
2516	002320	077577	040444	052040
2517	002326	042510	050440	044525
2518	002334	045503	041040	047522
2519	002342	047127	043040	054117
2520	002350	045040	046525	042520
2521	002356	020104	053117	051105
2522	002364	052040	042510	046040
2523	002372	055101	020131	047504
2524	002400	027107		
2525	002402	005015	077401	077577
2526	002410	000177		
2527	002412			
2528	002412			
2529	002412	022043	021041	023040
2530	002420	024047	025051	026053
2531	002426	027055	030460	031462
2532	002434	032464	033466	034470
2533	002442	035472	036474	037476
2534	002450	040500	041502	042504

DMSGAD:

MSG0		: ADDRESS OF MESSAGE #0
MSG1		: ADDRESS OF MESSAGE #1
MSG2		: ADDRESS OF MESSAGE #2
MSG3		: ADDRESS OF MESSAGE #3
MSG4		: ADDRESS OF MESSAGE #4
MSG5		: ADDRESS OF MESSAGE #5
MSG6		: ADDRESS OF MESSAGE #6
OPBUF		: ADDRESS OF OPERATOR SPEC'D MSG.
MSG8		: ADDRESS OF RECEIVE BUFFER FILL PATTERN

MSG0:	.BYTE	000	: MESSAGE OF ALL 0'S
EMSG0:			
MSG1:	.BYTE	377	: MESSAGE OF ALL 1'S
EMSG1:			
MSG2:	.BYTE	252	: MESSAGE OF ALTERNATING 1'S
EMSG2:			
MSG3:	.BYTE	125	: MESSAGE OF ALTERNATING 0'S
EMSG3:			
MSG4:			: "CCITT" 512-BIT (VS. 511 BITS) TEST PATTERN

.WORD	177603,157427,031011,047321,163715,105221,143325,142304
.WORD	040041,014116,052606,172334,105025,123754,111337,111523
.WORD	030030,145064,137642,143531,063617,135075,066730,026575
.WORD	052012,053627,070071,151172,165044,031605,166632,016741

EMSG4:
MSG5:

.ASCII	<177><177>/SA THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG./
--------	--

EMSG5:
MSG6:

.ASCII	/# \$! " & ' () * + , - . 0 1 2 3 4 5 6 7 8 9 : ; < = > ? @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z /
--------	---

CZKMSA0 KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 68
DEFAULT MESSAGE DEFINITIONS AND TABLES

2535	002456	043506	044510	045512
2536	002464	046514	047516	050520
2537	002472	051522	052524	053526
2538	002500	054530	132	
2539	002503	057	056133	057135
2540	002510	022537	000	
2541	002513			
2542		002514		

.ASCIZ ?/[\\]^_#?

EMSG6:

.EVEN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 69
DEFAULT MESSAGE DEFINITIONS AND TABLES

2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555

002514 047045 040445
002520 000122
002642
002642 033
002643

; THESE THREE STORAGE AREAS MUST NOT BE SEPARATED !!!!
OPBFPT: .ASCII /%NZA/
OPBUF: .BLKB 82. ;BUFFER FOR OPERATOR SPEC'D MESSAGES
OPEND:
; THE ABOVE THREE LINES MUST BE KEPT TOGETHER

MSG8: .BYTE 33 ;RECEIVE BUFFER FILL PATTERN
EMSG8:

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 70
DEFAULT MESSAGE DEFINITIONS AND TABLES

*****KMS11-BD/BE DOES NOT SUPPORT DOWN LINE LOAD*****
: DOWN-LINE-LOAD MESSAGE DEFINITIONS

2556
2557
2558
2559
2560
2561 002643 006
2562 002644 000
2563 002645 000
2564 002646 000
2565 002647 000
2566 002650
2567
2568 002650 000
2569 002651 000
2570 002652 006
2571 002653 000
2572 002654 000
2573 002655 000
2574
2575
2576
2577
2578 002656 000240
2579 002660 005037 000006
2580 002664 012706 001000
2581 002670 012701 177560
2582 002674 010700
2583 002676 062700 000034
2584 002702 105761 000004
2585 002706 100375
2586 002710 112061 000006
2587 002714 001372
2588 002716 012737 000026 000024
2589 002724 005037 000026
2590 002730 000777
2591 002732 006412 047502 052117
2592 002740 046440 051505 040523
2593 002746 042507 053440 051501
2594 002754 051040 041505 105
2595 002761 111 042526 020104
2596 002766 052523 041503 051505
2597 002774 043123 046125 054514
2598 003002 026440 105
2599 003005 116 020104 043117
2600 003012 052040 051505 020524
2601 003020 041
2602 003021 012 027015 027056
2603 003026 051056 046105 040517
2604 003034 020104 051120 043517
2605 003042 040522 027115 027056
2606 003050 000056

:::::ENTER MOP MODE MESSAGE FORMAT
:::::THE NODE WILL ENTER MAINTENANCE MODE ONLY IF THE PASSWORD MATCHES.
DLLM1: .BYTE 6 :BINARY CODE FOR MAINTENANCE MODE
PASS1: .BYTE 0 :PASSWORD BYTE #1 LEGAL VALUE 0 - 255
PASS2: .BYTE 0 :VALUE IN BYTE 1 IS DUPLICATED HERE
PASS3: .BYTE 0 :AND HERE
PASS4: .BYTE 0 :AND HERE
DLLM1E: :END ENTER MOP MODE MESSAGE FORMAT
:::::MEMORY LOAD WITH TRANSFER ADDRESS MESSAGE FORMAT
DLLM2: .BYTE 0 :CODE
.BYTE 0 :LOAD NUMBER
.BYTE 6 :LOAD ADDRESS LSB
.BYTE 0
.BYTE 0
.BYTE 0 :LOAD ADDRESS

:'LOADER' IMAGE DATA :LOCATIONS (WORDS)
:ADDR CONTENTS
:-----
NOP :'BYTE COUNT' :006 000240
CLR @#6 : (240(8)=160.) :010 005037 000006
MOV #1000,SP : :014 012706 001000
MOV #177560,R1 :SET UP TTY :020 012706 177560
MOV PC,R0 :MAKE ADDR.PIC :024 010700
ADD #<MSG-.>,R0 :ADDRESS MSG. :026 062700 000034
1\$: TSTB 4(R1) :TTY READY? :032 105761 000004
BPL 1\$:WAIT TIL YES :036 100375
MOVB (R0)+,6(R1) :TYPE A CHAR :040 112061 000006
BNE 1\$:KEEP GOING :044 001372
MOV #26,@#24 :SETUP PWR FAIL :046 012737 000026 000024
CLR @#26 :ENSURE TBIT CLR :054 005037 000026
BR :JMP ON YOURSELF :060 000777
MSG: .ASCII <12><15>/BOOT MESSAGE WAS RECE/ :062 006412 047502

.ASCII /IVED SUCCESSFULLY -E/ ;
.ASCII /ND OF TEST!!/ ;
.ASCIIZ <12><15>/....RELOAD PROGRAM..../;

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 71
DEFAULT MESSAGE DEFINITIONS AND TABLES

2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626

003052 177603 157427 031011
003060 047321 163715 105221
003066 143325 142304 040041
003074 014116 052606 172334
003102 105025 123754 111337
003110 111523 030030 145064

003116 126361

003120 006
003121 000
003122 000
003123 000
003124

:PADDING TO OBTAIN 240(8) BYTE LOADER
:(USE PART OF CCITT "WORST CASE BIT PATTERN"
.WORD 177603,157427,031011 :202 177603 157427 031011
.WORD 047321,163715,105221 :210 047321 163715 105221
.WORD 143325,142304,040041 :216 143325 142304 040041
.WORD 014116,052606,172334 :224 014116 052606 172334
.WORD 105025,123754,111337 :232 105025 123754 111337
.WORD 111523,030030,145064 :240 111523 030030 145064

.WORD 126361 :'"CHECKSUM WORD":246 126361

.BYTE 6
.BYTE 0
.BYTE 0
.BYTE 0

DLLM2E :

.EVEN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 72
COMMAND LINE INTERPRETER BUFFER, LOCATIONS AND MESSAGES

```

2627
2628
2629 003124 000122          CMDBUF: .BLKB 82.          :BUFFER FOR OPERATOR COMMANDS
2630 003246 000000          KEYWD1: .WORD 0           :THIS LOC WILL =1 IF CLEAR TYPED, 2 FOR SHOW,
2631                                     : A 4 IF RUN WAS TYPED, 5 IF HELP WAS TYPED
2632 003250 000000          QUALFG: .WORD 0           :THIS LOC HOLDS QUALIFIER VALUE (SIZE OR COPY)
2633 003252 000000          QUALVL: .WORD 0
2634 003254 014623          HLPTAB: .WORD HLP1
2635 003256 014636          .WORD HLP2
2636 003260 014754          .WORD HLP2B
2637 003262 015036          .WORD HLP3
2638 003264 015123          .WORD HLP3A
2639 003266 015150          .WORD HLP4
2640 003270 015227          .WORD HLP4A
2641 003272 015305          .WORD HLP5
2642 003274 015375          .WORD HLP6
2643 003276
2644 003276 015607          HLPEND:
2645 003300 015631          RHLPTB: .WORD RHL1
2646 003302 015646          .WORD RHL2
2647 003304          .WORD RHL3
2648
2649
2650
2651 003304 015720 015727 015734 SHTYTB: .WORD SHTYP0,SHTYP1,SHTYP2,SHTYP3,SHTYP4,SHTYP5,SHTYP6,SHTYP7
2652 003312 015741 015746 015754
2653 003320 015761 015767
2654
2655          : THE LIST OF BYTES BELOW ARE THE FIRST BYTES OF THE PREDEFINED MESSAGES
2656          : USED TO "SHOW" THE TRANSMIT AND COMPARE BUFFER CONTENTS.
2657
2658 003324 000 377 252 SHTAB: .BYTE 0,377,252,125,203,177,043
2659 003327 125 203 177
2660 003332 043
2661 003333
2662          SHTEND:
2663          .EVEN
2664 003334 016326          MODES: .WORD M00          :ADDRESSES OF MODE TYPES IN ASCII
2665 003336 016336          .WORD M01
2666 003340 016347          .WORD M02
2667 003342 016357          .WORD M03
2668 003344 016366          .WORD M04
2669 003346 016403          .WORD M05
2670 003350 016410          .WORD M06
2671
2672 003352 016417          LOOPS: .WORD LP0          :ADDRESSES OF LOOP TYPES IN ASCII
2673 003354 016427          .WORD LP1
2674 003356 016440          .WORD LP2
2675 003360 016446          .WORD LP3
2676 003362 016461          .WORD LP4

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 73
COMMAND LINE INTERPRETER BUFFER, LOCATIONS AND MESSAGES

2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690

003364 000000
003366 000000
003370 000000
003372 000000
003374 000000
003376 000000
003400 000
003401 000
003402 000
003404 003404
000000

:COMMAND LINE TRAVERSE LOCATIONS (USED BY "P\$TRV")

PSBUFA: .WORD 0
P\$TREE: .WORD 0
P\$ACT: .WORD 0
P\$CNT: .WORD 0
P\$NUM: .WORD 0
P\$RADX: .WORD 0
P\$NNUF: .BYTE 0
P\$GDBD: .BYTE 0
WRFLG: .BYTE 0
VALTRB: .WORD 0

:LOC. TO HOLD ADDR. OF CMD LINE BUFFER
:LOC. TO HOLD ADDR. OF PARSING TREE
:LOC. TO HOLD ADDR. OF ACTION ROUTINE
:LOC. TO BE A COUNTER LOCATION
:LOC. TO HOLD NUMERIC VALUE FROM PARSE
:LOC. TO HOLD RADIX USED(LO) AND +/- (HI BYTE)
:RETURN =0 IF ENOUGH OF COMMAND FOUND
:RETURN CODE 0 IF NO ERROR FOUND
:WRITE FLAG
:VALID LINE FLAG..IF SET -1 THEN VALID

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 74
MESSAGE BUFFERS AND POINTER TABLES

2691					
2692	003406	001000	TXBUF: .BLKB	BUFLIM	: TRANSMITTER BUFFERS
2693	004406	001000	CMPBUF: .BLKB	BUFLIM	: COMPARISON BUFFERS
2694	005406	001000	RXBUF: .BLKB	BUFLIM	: RECEIVER BUFFERS
2695					
2696					
2697	006406	000036	PTRTAB: .BLKW	MSGLIM*2	: TABLE FOR MESSAGE ADDRS. & BYTE COUNTS
2698	006502	000036	PTR13: .BLKW	MSGLIM*2	: COMPARE MESSAGE POINTERS
2699	006576	000036	PTR23: .BLKW	MSGLIM*2	: RECEIVE MESSAGE POINTERS
2700	006672	000322		MSGLIM*2*7.	: TABLE FOR MULTILINE POINTERS
2701					
2702	007536		PTREND:		: END OF MSG. PTR. TABLE
2703					
2704	007536	000002		2	: FILLER FOR OVERFLOW OF RX POINTER TABLE
2705	007542	000010	CPTCLS: .BLKW	8.	: TABLE FOR MULTILINE RX POINTERS
2706	007562	000010	CPTTLS: .BLKW	8.	: TABLE FOR MULTILINE TX POINTERS
2707	007602	000010	DVRCLS: .BLKB	8.	: TABLE (BYTES) FOR REC COUNTS
2708	007612	000010	DVTCLS: .BLKB	8.	: TABLE (BYTES) FOR TX COUNTS
2709	007622	000010	TRIBLS: .BLKB	8.	: TABLE (BYTES) OF LINE NUMBERS
2710	007632	125252		125252	: (END OF LINE NUMBER LIST)
2711	007634	000000	TRBTOT: .WORD	0	: TOTAL NUMBER OF LINES IN LIST
2712	007636	000000	TRIBN: .WORD	0	: CURRENT LINE NUMBER
2713	007640	000000	INDW: .WORD	0	: WORD INDEX
2714	007642	000000	INDEX: .WORD	0	: BYTE INDEX FOR LINES
2715	007644	000000	CTX: .WORD	0	: COUNTER FOR TX BUFFER COMPLETE INTERRUPTS
2716	007646	000000	CRX: .WORD	0	: COUNTER FOR RX BUFFER COMPLETE INTERRUPTS
2717	007650	000000	RSPTRS: .WORD	0	: STACK POINTER FOR RX INTERPUTS ON STACK
2718	007652	000000	RSPTRE: .WORD	0	: STACK POINTER FOR RX INTERPUTS OFF STACK
2719	007654	000000	TSP1R: .WORD	0	: STACK POINTER FOR TX INTERRUPTS
2720	007656	000006	TXSTAK: .BLKW	6.	
2721	007672	000066	RXSTAK: .BLKW	54.	: TX AND RX INT STACKS
2722	010046		RXSKEN:		
2723	010046	000000	MPLY: .WORD	0	: MULTIPLIER
2724	010050	000000	RXPTR: .WORD	0	: RECEIVER MESSAGE POINTER
2725	010052	000000	TXPTR: .WORD	0	: TRANSMITTER BUFFER POINTER
2726	010054	000000	CMPPTR: .WORD	0	: COMPARISON BUFFER POINTER
2727	010056	000000	CMPTOT: .WORD	0	: CMP MSG TOTAL
2728	010060	000000	CTOTCC: .WORD	0	: COMPARE BUFFER CHAR. COUNT
2729	010062	000000	CCURAD: .WORD	0	: CURRENT ADDR OF CMP BUFF TO ADD AT
2730	010064	000000	DVTXA: .WORD	0	: DEVICE TX ADDR
2731	010066	000000	DVTCC: .WORD	0	: DEVICE TX CHAR COUNT
2732	010070	000000	DVTTB: .WORD	0	: DEVICE LINE NUMBER (TRIBN)
2733	010072	000000	DVTCT: .WORD	0	: DEVICE TX MESSAGE COUNT
2734	010074	000000	TXMTOT: .WORD	0	: TX MSG TOTAL
2735	010076	000000	TTOTCC: .WORD	0	: TX BUFFER CHAR. COUNT
2736	010100	000000	TCURAD: .WORD	0	: CURRENT ADDR. OF TX BUFF TO ADD AT
2737	010102	000000	DVRTB: .WORD	0	: RECEIVE LINE (TRIBN)
2738	010104	000000	DVRXA: .WORD	0	: DEVICE RX ADDR
2739	010106	000000	DVRCC: .WORD	0	: DEVICE RX CHAR COUNT
2740	010110	000000	DVRCT: .WORD	0	: DEVICE RX MESSAGE COUNT
2741	010112	000000	RXMTOT: .WORD	0	: TOTAL NUMBER OF RX MSGS PER LINE
2742					
2743	010114	000000	LCNNT: .WORD	0	: NUMBER OF OPERATOR AWAKE MSGS
2744	010116	000000	OPVAR: .WORD	0	: HOLDER FOR OPTIONAL VARIABLE (1)
2745	010120	000000	OPVAR1: .WORD	0	: HOLDER FOR OPTION VARIABLE (2)
2746	010122	000000	PSCNT: .WORD	0	: PASS COUNTER

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 75
MESSAGE BUFFERS AND POINTER TABLES

2747	010124	000000
2748	010126	000000
2749	010130	000000
2750	010132	000000
2751	010134	000000
2752		
2753		
2754	010136	000000
2755	010140	000000
2756	010142	000000
2757	010144	000000
2758		

ERRCNT:	.WORD	0
STADD:	.WORD	0
ENADD:	.WORD	0
BYTBIT:	.WORD	0
CLNSET:	.WORD	0
RQIFLG:	.WORD	0
FTLFLG:	.WORD	0
TSSFLG:	.WORD	0
OVRCNT:	.WORD	0

:ERROR COUNTER
:START ADDR.
:END ADDR. FOR DUMP
:BYTE BIT FOR DUMP ROUTINE
:CLEANSET FLAG SET AND CLEARED IN CLEAN UP
:INDICATES TO OUTPUT HANDELER THAN NO OUTPUTS SHOULD
:BE PRINTED
:RQI FLAG
:USED AS FATEL ERROR FLAG
:USED AS TSS FLAG
:USED FOR QUE OVERFLOW FLAG

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 76
MESSAGE BUFFERS AND POINTER TABLES

2759
 2760
 2761
 2762 010146 000000
 2763
 2764 010150 000000
 2765 010152 000000
 2766 010154 000000
 2767 010156 000000
 2768 010160 000000
 2769 010162 000000
 2770 010164 000000
 2771 010166 000000
 2772 010170 000000
 2773 010172 000000
 2774 010174 000000
 2775 010176 000000
 2776 010200 000000
 2777 010202 000000
 2778 010204 000
 2779 010205 000
 2780

:OTHER MESSAGE RELATED STORAGE LOCATIONS

MSGTYP: .WORD 0 ;TYPE OF DATA 0=0'S,1=1'S,2=10'S,3=01'S
 ;4=CCITT,5=QUICK FOX,6=ALPHA/NUM,7=OPER
 CURCC: .WORD 0 ;TX/RX/CMP CHAR COUNT
 CPTRR: .WORD 0 ;CURRENT RX POINTER
 CPTR: .WORD 0 ;CURRENT POINTER
 CURADD: .WORD 0 ;CURRENT TX/RX/CMP START ADDD
 TOTCC: .WORD 0 ;TOTAL CHAR COUNT NOT MORE THEN 'BUFLIM'
 OFFSET: .WORD 0 ;OFFSET COUNT
 TEMP: .WORD 0 ;TEMPORARY LOCATIONS (USED A LOT)
 TEMP1: .WORD 0
 TEMP2: .WORD 0
 TEMP3: .WORD 0
 TEMP4: .WORD 0
 TEMP5: .WORD 0
 SAVSP: .WORD 0 ;STACK POINTER SAVE AREA
 CONOTM: .WORD 0 ;CONTROL OUT ERROR MSG. ADDRESS AND TSS AND GSS MSGS.
 GOOD: .BYTE 0 ;BYTE TO HOLD EXPECTED MESSAGE DATA BYTE FOR ERR REPORT
 BAD: .BYTE 0 ;BYTE TO HOLD RECEIVED MESSAGE DATA BYTE FOR ERR REPORT

```

2781
2782
2783      ;MORE INDEPENDENT CODE STORAGE LOCATIONS
2784 010206 000000 LOGUNT: .WORD 0      ;LOC. TO HOLD LOGICAL UNIT NUMBER
2785 010210 000000 PCADD:  .WORD 0      ;LOC. HOLD PC OF CALLING ROUTINE
2786 010212 000000 DCLFLG: .WORD 0      ;LOC. TO HOLD DO CLEAN FLAG 1 IF DOCLEAN INIT 0 IF NOT.
2787 010214 000000 RESFLG: .WORD 0      ;LOC TO HOLD FLAG (-1) THAT A RESTART WAS GIVEN
2788 010216 000000 MODTYP: .WORD 0      ;DCLT MODE OF OPERATION TYPE
2789                                     ; (0=REC-ONLY, 1=TX-ONLY, 2=PASSIVE-LOOPBK,
2790                                     ; 3=ACTIVE-LOOPBK, 4=DOWN L.L., 5=TALK, 6=LISTEN)
2791 010220 000000 MLTYP:  .WORD 0      ;MAINTENANCE LOOP TYPE (0=NONE, 1=INTERNAL TTL,
2792                                     ; 2=CABLE, 3=MODEM-ANALOG LOOPBK (LOCAL),
2793                                     ; 4=MODEM-DIGITAL LOOPBK (REMOTE), 5=MOP)
2794 010222 000000 FHDPLX: .WORD 0      ;FULL OR HALF DUPLEX FLAG (1=FULL FROM P-TABLE)
2795 010224 000002 PARAM:  .WORD 2      ;PROGRAM PARAMETERS
2796                                     ; BIT0= STATUS MSGS TO OPR PRINTED (1=YES)
2797                                     ; BIT1= DATA CHECKING DONE ON RCVD MSGS (1=YES)
2798                                     ; BIT2= ECHO (TRANSMIT) RCV'D MSG.(PASSIVE)(1=YES)
2799                                     ; BIT3= MODEM STATUS CHECK (1=YES)
2800                                     ; BIT4= CRC CALC./CHECK DONE (1=YES)
2801                                     ; BIT5= PROTOCOL EMULATION (1=YES)
2802                                     ; BIT6= SPARE
2803 010226 000000 RPASS:  .WORD 0      ;PASS NUMBER FROM RUN COMMAND
2804 010230 000000 FLAG:   .WORD 0      ;DEVICE FLAG WORD
2805                                     ;BIT0 = INPUT INT. REC.
2806                                     ;BIT1 = OUTPUT INT REC
2807                                     ;BIT2 = RX QUED /COMPL
2808                                     ;BIT3 = TX QUED/COMPL
2809                                     ;BIT6 = EXPECT TO GET A RX COMPLETED
2810                                     ;BIT7 = EXPECT TO GET A TX COMPLETED
2811                                     ;BIT8 = EXPECT INIT SYSTEM COMPLETED
2812                                     ;BIT9 = EXPECT SET LINE COMPLETED
2813                                     ;BIT10 = STATUS REPORT EXPECTED
2814                                     ;BIT11 = TRANSMIT TIMEOUT OCCURRED
2815
2816
2817
2818      ;MODE DISPATCH TABLE
2819
2820 010232 060736 MODE:   .WORD RXONLY ;RX ONLY DISPATCH
2821 010234 060764      .WORD TXONLY ;TX ONLY DISPATCH
2822 010236 061022      .WORD PLCK  ;PASSIVE LOOP BACK DISP
2823 010240 061050      .WORD ALCK  ;ACTIVE LOOP BACK DISP
2824 010242 062324      .WORD DLL   ;DOWN LINE LOAD DISP
2825 010244 062350      .WORD TALCK ;TALK MODE DISPATCH
2826 010246 062602      .WORD LISCK ;LISTEN MODE DISPATCH
2827
2828
2829 010250 000000 CLKCSR: .WORD 0      ;CLOCK CSR ADDRESS
2830 010252 000000 CLKBR:  .WORD 0      ;CLOCK INTERRUPT LEVEL
2831 010254 000000 CLKVEC: .WORD 0      ;CLOCK INTERRUPT VECTOR
2832 010256 000074 CLKHZ:  .WORD 60.     ;CLOCK'S HERTZ RATE
2833 010260 000000 CLKEN:  .WORD 0      ;CLOCK'S CSR VALUE TO INTRPT. ENABLE IT
2834
2835 010262 000000 TIMMIN: .WORD 0      ;PLACE TO KEEP TIME-SINCE-START
2836 010264 000000 TIMSEC: .WORD 0

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 78
CLOCK TABLES, EVENT LOG AND POINTERS

2837 010266 000000
2838
2839 010270 000000
2840 010272 000000
2841 010274 000000
2842

TIMTCK: .WORD 0 ;PLACE TO KEEP # OF TICKS/SEC
TIMER1: .WORD 0 ;EVENT TIMER #1 (TICKS)
TIMER2: .WORD 0 ;EVENT TIMER #2 (TICKS)
TIMERS: .WORD 0 ;EVENT TIMER #3 (SECONDS)

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 79
CLOCK TABLES, EVENT LOG AND POINTERS

2843
2844
2845
2846
2847
2848
2849
2850
2851
2852

010276 010300
010300 177777
010302 000415
011334 000001

:EVENT LOG TABLE AND ITS NEXT ENTRY POINTER

EVTPTN: .WORD EVTLOG : POINTER TO NEXT FREE SPACE IN EVENT LOG
EVTLOG: .WORD #-1 : SPECIFY EVENT LOG IS EMPTY
 .BLKW 269. : EVENT LOG BUFFER
EVTEND: .BLKW 1. : APPROXIMATE END OF EVENT TABLE (ALLOWS CIRCULAR QUE)

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 80
MODEM DATA SECTION

2853
 2854
 2855 011336 000000
 2856
 2857
 2858
 2859 011340 000000
 2860 011342 000000
 2861 011344 000000
 2862 011346 000000
 2863 011350 000000
 2864 011352 000000
 2865 011354 000000
 2866 011356
 2867
 2868
 2869
 2870 011356 021613
 2871 011360 021617
 2872 011362 021623
 2873 011364 021627
 2874 011366 021633
 2875 011370 021637
 2876 011372 021643
 2877

MODS: .WORD 0 ;MODEM STATUS

;TABLE OF MODEM SIGNAL BIT DEFINITIONS

MOBITS: .WORD CTS ;CLEAR TO SEND (CIRCUIT CB)
 .WORD DSR ;DATA SET READY (CIRCUIT CC)
 .WORD DCD ;DATA CARRIER DETECT (CIRCUIT CF)
 .WORD RTS ;REQUEST TO SEND (CIRCUIT CA)
 .WORD RI ;RING INDICATOR (CIRCUIT CE)
 .WORD SQD ;SIGNAL QUALITY DETECT (CIRCUIT CG)
 .WORD TM ;MODEM IN TEST MODE (RS 449 ONLY CIRCUIT TM)

MOBITE:

;TABLE OF ADDRESSES OF MODEM SIGNAL MESSAGE POSITIONS

MOMSGS: .WORD EVMCTS ;CLEAR TO SEND (CIRCUIT CB)
 .WORD EVMSDR ;DATA SET READY (CIRCUIT CC)
 .WORD EVMDCD ;DATA CARRIER DETECT (CIRCUIT CF)
 .WORD EVMRTS ;REQUEST TO SEND (CIRCUIT CA)
 .WORD EVMRI ;RING INDICATOR (CIRCUIT CE)
 .WORD EVMSQD ;SIGNAL QUALITY DETECT (CIRCUIT CG)
 .WORD EVMTM ;MODEM IN TEST MODE (RS 449 ONLY CIRCUIT TM)

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 81
EVENT DESCRIPTION MESSAGES LOCATIONS AND TABLES

2878
2879
2880
2881
2882
2883 011374 017440
2884 011376 017464
2885 011400 017513
2886 011402 017540
2887 011404 017566
2888 011406 017633
2889 011410 017603
2890 011412 016417
2891 011414 017661
2892 011416 017716
2893 011420 017751
2894 011422 020022
2895
2896
2897
2898 011424 000000
2899 011426 000000
2900 011430 000000
2901 011432 000000
2902 011434 000000
2903 011436 000000
2904
2905
2906
2907 011440 025614
2908 011442 025614
2909 011444 025614
2910 011446 025614
2911 011450 025666
2912 011452 026316
2913 011454 026526
2914 011456 000000
2915 011460 026526
2916 011462 026446
2917 011464 026366
2918 011466 026366
2919
2920
2921 011470 000000
2922 011472 000000
2923 011474 000000
2924 011476 000000
2925
2926 011500 000000
2927 011502 000000
2928 011504 000000
2929 011506 000000
2930 011510 000000
2931 011512 000000
2932
2933 011514 000000

:TABLE OF ADDRESSES OF EVENT DESCRIPTION MESSAGES
: ORDER CORRESPONDS TO MESSAGE TYPE VALUES

EVTLST: .WORD EDTXQ ;TRANSMIT MESSAGE QUEUED
.WORD EDTXC ;TRANSMIT OF MESSAGE COMPLETE
.WORD EDRXQ ;RECEIVE MESSAGE SPACE QUEUED
.WORD EDRXC ;MESSAGE RECEIVED - RECEIVE COMPLETE
.WORD EDDER ;DEVICE INFORMATION
.WORD EDDVI ;DEVICE INITIALIZE STARTED
.WORD EDDCK ;DATA COMPARISON DONE
.WORD LPO ;NULL STRING
.WORD EDDLE ;DATA COMPARE LENGTH ERROR
.WORD EDDDE ;DATA COMPARE DATA ERROR
.WORD EDEOP ;END OF PASS
.WORD EDABO ;^C ABORT

:LOCATIONS USED DURING EVENT REPORTING

EVTSEC: .WORD 0 ;TEMPORARY LOCS TO KEEP EVENT TIME WHILE REPORTING
EVTMIN: .WORD 0
EVTICK: .WORD 0
EVTADD: .WORD 0 ;TEMP. LOC. TO HOLD ADDRESS DURING EVENT REPORTING
EVTBCT: .WORD 0 ; " " BYTE COUNT " " "
EVTIMP: .WORD 0 ; " " OTHER DATA " " "

:REPORT CODING DISPATCH TABLE

RPTDSP: .WORD RPTTXQ ;TRANSMIT QUEUED ENTRY DECODING
.WORD RPTTXQ ;TRANSMIT COMPLETE ENTRY DECODING
.WORD RPTTXQ ;RECEIVER QUEUED ENTRY DECODING
.WORD RPTTXQ ;RECEIVER COMPLETE ENTRY DECODING
.WORD RPTDER ;DEVICE ERROR ENTRY DECODING
.WORD RPTDVI ;DEVICE INIT ENTRY DECODING
.WORD RPTDCK ;DATA COMPARISON ENTRY DECODING
.WORD 0 ;PLACE HOLDER
.WORD RPTDLE ;DATA COMPARISON LENGTH ERROR
.WORD RPTDDE ;DATA COMPARISON DATA ERROR
.WORD RPTTEOP ;END OF PASS
.WORD RPTABO ;^C ABORT

DEV1: .WORD 0 ;TEMP LOCS TO HOLD DATA FOR EVENT REPORTING
DEV2: .WORD 0 ; AND SHOW MODE,... SUBROUTINE
DEV3: .WORD 0
DEV4: .WORD 0

DEVS1: .WORD 0 ;TEMP DATA FOR STATUS REPORTS
DEVS2: .WORD 0
DEVS3: .WORD 0
DEVS4: .WORD 0

LINEST: .WORD 0 ;LINE NUMBER OF STATUS DATA
ENRDS1: .WORD 0 ;= 1 READ STATUS ENABLED. SET = 1 IN KSTART
;= 0 READ STATUS DISABLED SET = 0 IN BGNCLN
VSTAT: .WORD 0 ;= 1 STATUS DATA IS VALID. SET = 1 IN LNSTAT

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 82
EVENT DESCRIPTION MESSAGES LOCATIONS AND TABLES

2934
2935 011516 000000
2936

CNTCLN: .WORD 0

::= 0 STATUS DATA NOT VALID. SET = 0 IN KSTART
:COUNT OF EXCURSIONS THRU BGNCLN CODE
: SINCE LOAD OF DCLT PROGRAM

C7KMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 83
COMMAND LINE ACTION TREE

2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975
2976

011520

011520
011524
011530
011532
011546
011550
011564
011566
011602
011604
011616
011622
011636
011642
011656
011662
011676
011702
011706
011720
011724
011736
011742

:SAMPLE CLI TREE NODE (ALWAYS AT LEAST 1 WORD)

! ACTION ! CHAR CODE !	
! MISS DISPLACEMENT !	ONLY IF 'MISS' ARGUMENT DEFINED
! NEXT NODE DISPLMNT !	ONLY IF 'ASCII' ARGUMENT DEFINED
! ASCIIZ MATCH STRING ! (.EVEN) !	ONLY IF 'ASCII' ARGUMENT DEFINED

CLITRE:

:FIRST KEYWORD

```

N10$: CLI CLISPA,0,N10$
      CLI <'?'>,HLP,N42$
N42$: CLI CLIEXI,0
      CLI CLISTR,HLP,N43$,<'HELP'>
N43$: CLI CLIEXI,0
      CLI CLISTR,PRNT,N44$,<'PRINT'>
N44$: CLI CLIEXI,0
      CLI CLISTR,EXIT,N45$,<'EXIT'>
N45$: CLI CLIEXI,0
      CLI CLISTR,RUN,N46$,<'RUN'>
N46$: CLI CLIBR,0,N80$
      CLI CLISTR,NOTNUF,N40$,<'DUMP'>
N40$: CLI CLIBR,0,N50$
      CLI CLISTR,CLEAR,N47$,<'CLEAR'>
N47$: CLI CLIBR,NOTNUF,N100$
      CLI CLISTR,NOTNUF,N20$,<'LINE'>
N20$: CLI CLIBR,0,N105$
      CLI <'S'>,NOTNUF,N30$
      CLI CLISTR,SHOW,N25$,<'HOW'>
N25$: CLI CLIBR,0,N100$
      CLI CLISTR,0,N30$,<'ET'>
N30$: CLI CLIBR,0,N110$
      CLI CLIERR,0

```

```

:SKIP ANY LEADING SPACES
:IS THE FIRST NON-SP CHAR A '?'
: IF YES DO 'HLP' AND EXIT
:ELSE, IS FIRST WORD A 'HELP'
: IF YES DO 'HLP' AND EXIT
:ELSE, IS FIRST WORD A 'PRINT'
: IF YES DO 'PRINT' AND EXIT
:ELSE, IS FIRST WORD A 'EXIT'
: IF YES DO 'EXIT' AND EXIT
:ELSE, IS FIRST WORD A 'RUN'
: IF YES DO 'RUN' & GOTO N80$
:ELSE, IS FIRST WORD A 'DUMP'
: IF YES GOTO N80$
:ELSE, IS FIRST WORD A 'CLEAR'
: IF YES DO 'CLR' & GOTO N100$
:ELSE IS FIRST WORD LINE

:ELSE, IS FIRST CHAR. A 'S'
: IF YES IS REST OF WORD 'HOW'
: IF YES, DO 'SHOW',BR N100$
: ELSE, IS REST OF WORD 'ET'
: IF YES, DO 'SET', BR N110$
:OTHERWISE 'ILL CMD' - EXIT

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 84
COMMAND LINE ACTION TREE

```

2977      ;SECOND KEYWORD (MODE=) FOR RUN COMMAND
2978
2979 011744 N80$: CLI CLISPA,0,N30$ ;SKIP LEADING SPS, IF NONE-ERR
2980 011750 N81$: CLI CLISTR,NOTNUF,N30$,<'MODE'> ;IS NEXT WORD 'MODE='
2981 011764      CLI <'='>,0,N30$ ; IF NO, IT'S WRONG -ERR -EXIT
2982 011770      CLI CLISTR,ATVMOD,N82$,<'ACTIVE'> ;IS NEXT WORD 'ACTIVE'
2983 012006      CLI CLIBR,0,N115$ ; IF YES, DO 'ACTIVE',BR N115$
2984 012012 N82$: CLI CLISTR,PASMOD,N83$,<'PASSIVE'> ;IS NEXT WORD 'PASSIVE'
2985 012030      CLI CLIBR,0,N115$ ; IF YES, DO 'PASSIVE',BR N115$
2986 012034 N83$: CLI CLISTR,RECMOD,N84$,<'RECEIVE'> ;IS NEXT WORD 'RECEIVE'
2987 012052      CLI CLIBR,0,N115$ ; IF YES, DO 'RECEIVE',BR N115$
2988 012056 N84$: CLI CLISTR,LISMOD,N85$,<'LISTEN'> ;IS NEXT WORD 'LISTEN'
2989 012074      CLI CLIBR,0,N115$ ; IF YES, DO 'LISTEN',BR N115$
2990 012100 N85$: CLI CLISTR,DLLOD,N86$,<'DOWNLINELOAD'> ;IS NEXT WORD 'DOWNLINELOAD'
2991 012124      CLI CLIBR,0,N115$ ; IF YES, DO 'DOWNLINELOAD',BR N115$
2992 012130 N86$: CLI <'T'>,0,N30$ ;IS NEXT CHAR A 'T'
2993 012134      CLI CLISTR,TRAMOD,N87$,<'TRANSMIT'> ; IS REST OF WORD 'TRANSMIT'
2994 012152      CLI CLIBR,0,N115$ ; IF YES, DO 'TRANSMIT',BR N115$
2995 012156 N87$: CLI CLISTR,TALMOD,N30$,<'ALK'> ; IS REST OF WORD 'ALK'
2996 012170      CLI CLIBR,0,N115$ ; IF YES, DO 'ALK',BR N115$
2997      ; IF NO, ERROR - EXIT
2998
2999      ;SECOND KEYWORD (FOR CLEAR OR SHOW)
3000 012174 N100$: CLI CLISPA,0,N30$ ;SKIP LEADING SPACES, NONE=ERR
3001 012200 N102$: CLI CLISTR,CSHEXP,N104$,<'EXPECTBUFF'> ;IS NEXT WORD 'EXPECTBUFF'
3002 012222      CLI CLIEXI,0 ; IF YES, DO CLR-EXP,EXIT
3003 012224 N104$: CLI CLISTR,CSHTRN,N30$,<'TRANSMITBUFF'> ;IS NEXT WORD 'TRANSMITBUFF'
3004 012250      CLI CLIEXI,0 ; IF YES, DO CLR-TRN,EXIT
3005      ; IF NO - ERROR - EXIT
3006
3007
3008      ;SECOND KEYWORD (FOR SET)
3009 012252 N110$: CLI CLISPA,0,N30$
3010 012256 N111$: CLI CLISTR,SETEXP,N112$,<'EXPECTMSG'>
3011 012276      CLI CLIBR,0,N120$
3012 012302 N112$: CLI CLISTR,SETTRN,N30$,<'TRANSMITMSG'>
3013 012324      CLI CLIBR,0,N120$
3014
3015      ;GET ADDRESSES FOR DUMP COMMAND
3016 012330 N50$: CLI CLIALP,0,N51$
3017 012334 N51$: CLI CLISPA,0,N52$
3018 012340 N52$: CLI CLIOCT,DMPS,N30$
3019 012344      CLI <'-'>,NOTNUF,N125$
3020 012350      CLI CLIOCT,DMPE,N30$
3021 012354      CLI <'/'>,NOTNUF,N125$
3022 012360      CLI <'B'>,DMPQ,N30$
3023 012364      CLI CLIBR,0,N125$

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 85
COMMAND LINE ACTION TREE

```

3024          ;QUALIFIERS FOR THE RUN COMMAND
3025 012370   N115$: CLI CLIALP,0,N114$
3026 012374   N114$: CLI <' />,NOTNUF,N125$
3027 012400   CLI CLISTR,NO,N116$,<'NO'>
3028 012412   N116$: CLI <'C'>,0,N117$
3029 012416   CLI CLISTR,CHECK,N117$,<'HECK'>
3030 012432   CLI CLIBR,0,N115$
3031
3032
3033 012436   N117$: CLI CLISTR,STATUS,N118$,<'STATUS'>
3034 012454   CLI CLIBR,0,N115$
3035 012460   N118$: CLI CLISTR,ECHO,N130$,<'ECHO'>
3036 012474   CLI CLIBR,0,N115$
3037
3038 012500   N130$: CLI CLISTR,0,N132$,<'PASS'>
3039 012514   CLI CLIBR,0,N150$
3040
3041 012520   N132$: CLI CLISTR,MOSC,N131$,<'MODEM'>
3042 012534   CLI CLIBR,0,N115$
3043
3044 012540   N131$: CLI CLISTR,0,N30$,<'LOOP'>
3045 012554   CLI CLIBR,0,N140$

```

```
3046      :GET MESSAGE TYPE FOR SET MESSAGE COMMANDS
3047 012560 N120$: CLI <'=>,0,N30$
3048
3049      : LOOK FOR DEFAULT MESSAGE NAME
3050 012564 N60$: CLI CLISTR,CMMSG1,N61$,<'ONES'>
3051 012600      CLI CLIBR,0,N121$
3052 012604 N61$: CLI CLISTR,CMMSG0,N62$,<'ZEROES'>
3053 012622      CLI CLIBR,0,N121$
3054 012626 N62$: CLI CLISTR,CMMSG2,N63$,<'1ALT'>
3055 012642      CLI CLIBR,0,N121$
3056 012646 N63$: CLI CLISTR,CMMSG3,N64$,<'0ALT'>
3057 012662      CLI CLIBR,0,N121$
3058 012666 N64$: CLI CLISTR,CMMSG5,N65$,<'ITEP'>
3059 012702      CLI CLIBR,0,N121$
3060 012706 N65$: CLI CLISTR,CMMSG4,N66$,<'CCITT'>
3061 012722      CLI CLIBR,0,N121$
3062 012726 N66$: CLI CLISTR,CMMSG6,N67$,<'ALPHA'>
3063 012742      CLI CLIBR,0,N121$
3064 012746 N67$: CLI CLISTR,SETET,N68$,<'TRANSMIT'>
3065 012766      CLI CLIBR,0,N125$
3066
3067      : LOOK FOR QUOTED MESSAGE
3068 012772 N68$: CLI <'>,OPRMSG,N30$
3069 012776 N70$: CLI <'>,ENDQO,N71$
3070 013002      CLI CLIBR,0,N121$
3071 013006 N71$: CLI CLISPA,0,N72$
3072 013012 N72$: CLI CLIALN,0,N73$ ;ONLY A-Z,SP,TAB, OR 0-9 BETWEEN ''S
3073 013016      CLI CLIBR,0,N70$
3074 013022 N73$: CLI CLIERR,BADCHR ;PRINT ERROR IF NONE LEGAL CHAR FOR ''S
3075
3076      :GET QUALIFIERS (SIZE OR COPY) FOR SET MESSAGE COMMANDS
3077 013024 N121$: CLI CLIALP,0,N123$
3078 013030 N123$: CLI <'/'>,NOTNUF,N125$
3079 013034      CLI CLISTR,SIZE,N122$,<'SIZE'>
3080 013050      CLI CLIBR,0,N126$
3081 013054 N122$: CLI CLISTR,QCOPY,N30$,<'COPY'>
3082 013070      CLI CLIBR,0,N126$
3083
3084      :NUMER FOR SIZE OR COPY
3085 013074 N126$: CLI <'=>,0,N30$
3086 013100      CLI CLIDEC,NUM,N30$
3087 013104      CLI CLIBR,0,N121$
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 87
COMMAND LINE ACTION TREE

3088
3089 013110
3090
3091
3092 013114
3093 013136
3094 013142
3095 013156
3096 013162
3097 013204
3098 013210
3099 013232
3100
3101
3102 013236
3103 013242
3104 013246
3105
3106
3107 013252
3108 013256
3109 013272
3110 013274
3111 013314
3112 013320
3113 013334
3114
3115 013336
3116 013342
3117 013354
3118 013360
3119 013364
3120
3121
3122 013370
3123

:GET MAINTENANCE LOOP TYPE FOR RUN "LOOP" QUALIFIER
N140\$: CLI <'=>,0,N30\$

N141\$: CLI CLISTR,TTLLOP,N142\$,<'INTERNAL TTL'>
CLI CLIBR,0,N115\$
N142\$: CLI CLISTR,CBLLOP,N143\$,<'CABLE'>
CLI CLIBR,0,N115\$
N143\$: CLI CLISTR,LMDLOP,N144\$,<'LOCAL MODEM'>
CLI CLIBR,0,N115\$
N144\$: CLI CLISTR,RMDLOP,N30\$,<'REMOTEMODEM'>
CLI CLIBR,0,N115\$

:GET NUMBER FOR "PASS" RUN QUALIFIER
N150\$: CLI <'=>,0,N30\$
CLI CLIDEC,PASC,N30\$
CLI CLIBR,0,N115\$

:GET LINE SHOW OR LINE FOR KILL OR ESTABLISH
N105\$: CLI CLISPA,NOTNUF,N106\$
N106\$: CLI CLISTR,SLST,N107\$,<'SHOW'>
CLI CLIEXI,0
N107\$: CLI CLISTR,ETRB,N108\$,<'ESTABLISH'>
CLI CLIBR,0,N160\$
N108\$: CLI CLISTR,KTRB,N30\$,<'KILL'>
CLI CLIEXI,0

N160\$: CLI <'=>,0,N30\$
N161\$: CLI CLISTR,KALL,N162\$,<'ALL'>
N162\$: CLI CLIDEC,EKTB,N30\$
N163\$: CLI 54,NOTNUF,N125\$
CLI CLIBR,0,N161\$

;LOOKING FOR "",

:END-OF-LINE
N125\$: CLI CLIEXI,0

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 88
DEVICE DEPENDENT LOCATIONS

3124
3125
3126
3127
3128
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150

013372 000000
013372 000000
013374 000000
013376 000000
013376 000000
013400 000000
013402 000000
013402 000000
013404 000000
013406 000000
013406 000000
013410 000000

013412 000000
013414 000000
013416 000000
013420 000000
013422 000000
013424 000000
013426 000000

:DEVICE DEPENDENT STORAGE LOCATIONS FOR
: CURRENT DEVICE PARAMETERS

SELO:
BSEL0: .WORD 0
BSEL1: .WORD 0
SEL2:
BSEL2: .WORD 0
BSEL3: .WORD 0
SEL4:
BSEL4: .WORD 0
BSEL5: .WORD 0
SEL6:
BSEL6: .WORD 0
BSEL7: .WORD 0

:ADDRESSES OF REGISTERS SELO THRU BSEL7

INVEC: .WORD 0
OUTVEC: .WORD 0
INTPRI: .WORD 0
OPTYP: .WORD 0
DEVPAR: .WORD 0
STATYP: .WORD 0
MODCSR: .WORD 0

:INPUT INTERRUPT VECTOR ADDRESS
:OUTPUT INTERRUPT VECTOR ADDRESS
:INTERRUPT PRIORITY
:OPTION TYPE

:CSR OF MODEM CONTROL MUX (DM11-BA)

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 89
DEVICE DEPENDENT LOCATIONS

;(KMS11-BD BUFFER DESCRIPTOR LISTS)

```

:-----:
: !STATUS ! MEMBER 'N'
:-----:
: !BUFFER START ADDR !
:-----:
: !#OF CHARS (14BITS) !
:-----:
: ! MESSAGE # !
:-----:

```

3151
3152
3153
3154
3155
3156
3157
3158
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204
3205
3206

```

013430 000000
013432 000000
013434
013434 000004
013444 000004
013454 000004
013464 000004
013474 000004
013504 000004
013514 000004
013524 000004
013534 000004
013544 000004
013554 000004
013564 000004
013574 000004
013604 000004
013614 000004
013624 000004
013634
013634 000000
013636 000000
013640 000000
013642 000000
013644 000001 000002 000003
013652 000010 000011 000020
013660 000021 177777
013664 023075
013666 023126
013670 023155
013672 023172
013674 023215
013676 023235
013700 023256
013702 023306

```

```

BDLMNM: .WORD 0 ;LOC TO CALCULATE BDL MEMBER NUMBER
BDLMAD: .WORD 0 ;LOC TO CALCULATE BDL MEMBER ADDRESS
BDLBAS:
RXBDL: .BLKW 4 ; MEMBER #1 (RX FOR LINE 0)
        .BLKW 4 ; MEMBER #2 (RX FOR LINE 1)
        .BLKW 4 ; MEMBER #3 (RX FOR LINE 2)
        .BLKW 4 ; MEMBER #4 (RX FOR LINE 3)
        .BLKW 4 ; MEMBER #5 (RX FOR LINE 4)
        .BLKW 4 ; MEMBER #6 (RX FOR LINE 5)
        .BLKW 4 ; MEMBER #7 (RX FOR LINE 6)
        .BLKW 4 ; MEMBER #10 (8.) (RX FOR LINE 7)
TXBDL: .BLKW 4 ; MEMBER #11 (9.) (TX FOR LINE 0)
        .BLKW 4 ; MEMBER #12 (10.) (TX FOR LINE 1)
        .BLKW 4 ; MEMBER #13 (11.) (TX FOR LINE 2)
        .BLKW 4 ; MEMBER #14 (12.) (TX FOR LINE 3)
        .BLKW 4 ; MEMBER #15 (13.) (TX FOR LINE 4)
        .BLKW 4 ; MEMBER #16 (14.) (TX FOR LINE 5)
        .BLKW 4 ; MEMBER #17 (15.) (TX FOR LINE 6)
        .BLKW 4 ; MEMBER #20 (16.) (TX FOR LINE 7)
BDLEND:
;KMS RESPONSE COUNTS..
ILLRSP: .WORD ;COUNT OF ILLEGAL RESPONSES FROM KMS
STARTC: .WORD ;COUNT OF RESTART DDCMP COMPLETED
ERRTHR: .WORD ;COUNT OF KMS ERROR THRESHOLD REACHED RESPONSES
RXMIT: .WORD ;RETRANSMIT SPECIFIER
;NAK REASON CODES
NAKX: .WORD 1,2,3,8.,9.,16.,17.,-1
;POINTERS TO NAK REASON MESSAGES
NAKMSG: .WORD #NAKR1
        .WORD #NAKR2
        .WORD #NAKR3
        .WORD #NAKR8
        .WORD #NAKR9
        .WORD #NAKR16
        .WORD #NAKR17
        .WORD #NAKR00

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 90
MISCELLANEOUS STORAGE FOR ERROR REPORTS

ERRTBL

LSERRTBL::

3207	013704	
3208	013704	
3209	013704	000000
3210	013706	000000
3211	013710	000000
3212	013712	000000
3213		

ERRTYP::	.WORD	0
ERRNBR::	.WORD	0
ERRMSG::	.WORD	0
ERRBLK::	.WORD	0

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 91
GLOBAL TEXT SECTION

3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252

:+
: THE GLOBAL TEXT SECTION CONTAINS FORMAT STATEMENTS,
: MESSAGES, AND ASCII INFORMATION THAT ARE USED IN
: MORE THAN ONE TEST.
:--

.SBTTL DEVICE SUPPORTED
: NAMES OF DEVICES SUPPORTED BY PROGRAM
:

DEV TYP <KMS11-BD,BE>

L\$DVTYP::
.ASCIZ /KMS11-BD,BE/
.EVEN

.SBTTL PROGRAM IDENTIFICATION
: TEST DESCRIPTION
:

DESCRIPT <CZKMSAO KMS11-BD,BE DATA COMM. LINK TEST>

L\$DESC::
.ASCIZ /CZKMSAO KMS11-B

013714
013714
013714 046513 030523 026461
013722 042102 041054 000105

013730
013730
013730 055103 046513 040523
013736 020060 046513 030523
013744 026461 042102 041054
013752 020105 040504 040524
013760 041440 046517 027115
013766 046040 047111 020113
013774 042524 052123 000
014002

.EVEN

.EVEN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 92
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

3253

.NLIST BEX

014002	041504	052114	000076	CLISPM:	.ASCIZ	/DCLT>/
014010	047045	040445	044477	CLIERM:	.ASCIZ	/XN%?ILL CMD-BAD SYNTAX?/
014040	047045	040445	044477	CLINUF:	.ASCIZ	/XN%?INCMPLTE CMD?/
014063	045	022516	037501	CLINBG:	.ASCIZ	/XN%?NUM TOO BIG?/
014105	045	022516	037501	CLIBRX:	.ASCIZ	/XN%?BAD RADIX?/
014125	045	022516	037501	CLIBDL:	.ASCIZ	/XN%?"LOOP" VALID ONLY IN ACTIVE?/
014167	045	022516	037501	KMSLOP:	.ASCII	/XN%?"LOOP=INT" NOT SUPPORTED BY KMS11-C FIRMWARE?/<15><12>
014253	040	051124	020131		.ASCIZ	/ TRY "L=C" & TEST CONNECTOR/
014307	045	022516	037501	CLINPS:	.ASCIZ	/XN%?"ECHO" VALID ONLY IN PASSIVE?/
014352	047045	040445	044477	CLIBCR:	.ASCIZ	/XN%?ILL CHR- "A-Z,0-9,SP,TAB" ONLY?/
014417	045	022516	037501	CLISEO:	.ASCIZ	/XN%?"SIZE=0" NOT VALID?/
014450	047045	040445	052077	CLIPW:	.ASCIZ	/XN%?TRANSMIT & EXPECT LIST MUST BE IDENTICAL FOR LOOP?/
014540	047045	040445	044124	HLP0:	.ASCIZ	/XN%THIS IS DCLT. TYPE "H" OR "?" FOR DETAILS/
014616	047045	052045	000	HLPF:	.ASCIZ	/XN%T/
014623	104	046103	020124	HLP1:	.ASCIZ	/DCLT CMDS:/
014636	041440	042514	051101	HLP2:	.ASCII	/ CLEAR OR SHOW EXPECTBUFF OR TRANSMITBUFF/<15><12>
014712	050040	044522	052116		.ASCII	/ PRINT OR EXIT/<15><12>
014732	042040	046525	020120		.ASCIZ	? DUMP START-END/B?
014754	046040	047111	020105	HLP2B:	.ASCIZ	? LINE SHOW, LINE ESTABLISH=N,(N=0-7) OR LINE KILL?
015036	051440	052105	042440	HLP3:	.ASCIZ	? SET EXPECTMSG OR TRANSMITMSG=TYPE/SIZE=N OR /COPY=N?
015123	040	042523	020124	HLP3A:	.ASCIZ	/ SET EXPECT=TRANSMIT/
015150	020040	052040	050131	HLP4:	.ASCIZ	? TYPE=ONES,ZEROES,1ALT,0ALT,ITEP,CCITT,ALPHA?
015227	040	020040	020040	HLP4A:	.ASCIZ	/ OR "OPR SPCD=A-Z,SP,TAB,0-9 IN QUOTES"/
015305	040	052522	020116	HLP5:	.ASCIZ	? RUN MODE=MTYP/LOOP=LTP/CHECK,STATUS,ECHO,MODEM,PASS=N?
015375	040	020040	052115	HLP6:	.ASCII	/ MTYP=TRAN,REC,ACT,PAS,TAL,LIS/<15><12>
015437	040	020040	052114		.ASCIZ	/ LTP=INT,CAB/
015457	045	022516	042101	DLLO1:	.ASCIZ	/XN%ADOWN LINE LOAD NOT AVAILABLE FOR THE KMS11-BD,BE/
015544	050122	037124	000	CLISRP:	.ASCIZ	/RPT>/
015551	045	022516	052101	RHLP0:	.ASCIZ	/XN%ATYPE "H" OR "?" FOR HELP!/
015607	104	046103	020124	RHLP1:	.ASCIZ	/DCLT REPORT CMDS:/
015631	040	054105	052111	RHLP2:	.ASCIZ	/ EXIT OR LOG/
015646	051440	040524	052524	RHLP3:	.ASCIZ	/ STATUSONLINE/
015664	047045	040445	051515	SHMSG:	.ASCIZ	?XN%AMSG: TYPE=%T%A/SIZE=%D3?
015720	042532	047522	051505	SHTYP0:	.ASCIZ	/ZEROES/
015727	117	042516	000123	SHTYP1:	.ASCIZ	/ONES/
015734	040461	052114	000	SHTYP2:	.ASCIZ	/1ALT/
015741	060	046101	000124	SHTYP3:	.ASCIZ	/0ALT/
015746	041503	052111	000124	SHTYP4:	.ASCIZ	/CCITT/
015754	052111	050105	000	SHTYP5:	.ASCIZ	/ITEP/
015761	101	050114	040510	SHTYP6:	.ASCIZ	/ALPHA/
015767	117	051120	051440	SHTYP7:	.ASCIZ	/OPR SPEC/
016000	047045	040445	047516	SHTRE:	.ASCIZ	\XN%AND LINE NUMBER IS ESTABLISHED\
016042	047045	040445	044514	SHTRH:	.ASCIZ	\XN%ALINE NUMBER : \
016066	042045	000063		SHTAP:	.ASCIZ	\%D3\
				::*834	SHTFL:	.ASCIZ \XN%ALINE NUMBER LIST FULL - LINE= %Z3%A NOT ADDED\
				::*834	SHTUN:	.ASCIZ \XN%?LINE NUMBER= %Z3%A IS NOT UNIQUE?\
				::*834	SHTNF:	.ASCIZ \XN%?LINE NUMBER= %Z3%A NOT FOUND?\
016072	047045	040445	040477	SHTLPA:	.ASCIZ	/XN%?A LINE MUST BE ESTABLISHED TO EXECUTE?/
016146	047045	040445	046077	SHTIV:	.ASCIZ	\XN%?LINE NUMBER= %Z3%A INVALID (NUMBERS 0-7 ARE VALID)?\
016237	045	022516	051101	SHTBR:	.ASCIZ	/XN%ARX BUFFER NOT BIG ENOUGHXN%ATOO MANY LINES OR MSGS/
016326	042522	042503	053111	MO0:	.ASCIZ	/RECEIVE/
016336	051124	047101	046523	MO1:	.ASCIZ	/TRANSMIT/
016347	120	051501	044523	MO2:	.ASCIZ	/PASSIVE/
016357	101	052103	053111	MO3:	.ASCIZ	/ACTIVE/

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 93
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

016366	047504	047127	044514	MO4:	.ASCIZ	/DOWNLINELOAD/
016403	124	046101	000113	MO5:	.ASCIZ	/TALK/
016410	044514	052123	047105	MO6:	.ASCIZ	/LISTEN/
016417	000			LP0:	.ASCIZ	//
016420	046057	047517	036520	LP00:	.ASCIZ	?/LOOP=?
016427	111	052116	051105	LP1:	.ASCIZ	?INTERNAL?
016440	040503	046102	000105	LP2:	.ASCIZ	?CABLE?
016446	047514	040503	046514	LP3:	.ASCIZ	?LOCALMODEM?
016461	122	046505	052117	LP4:	.ASCIZ	?REMOTEMODEM?
016475	116	117		PNST:	.ASCII	/NO/
016477	123	040524	052524	PST:	.ASCIZ	/STATUS/
016506	047516			PNCK:	.ASCII	/NO/
016510	044103	041505	000113	PCK:	.ASCIZ	/CHECK/
016516	047516			PNEC:	.ASCII	/NO/
016520	041505	047510	000	PEC:	.ASCIZ	/ECHO/
016525	116	117		PNMS:	.ASCII	/NO/
016527	115	042117	046505	PMS:	.ASCIZ	/MODEM/
016535	045	022516	046101	LISP:	.ASCIZ	/%N%ALIS>/
016546	046124	037113	000	OPRMM:	.ASCIZ	/TLK>/
016553	124	044510	020123	L5060:	.ASCIZ	/THIS A 50. OR 60. HZ. LSI-11:/
	016612				.EVEN	

:
: FORMAT STATEMENTS USED IN PRINT CALLS
:

016612	047045	040445	052123	PRTNOS:	.ASCIZ	/N%ASTATUS NOT AVAILABLE UNTIL NEXT DCLT>RUN%/
016671	045	022516	041501	BDCLK:	.ASCIZ	/N%ACLOCK NOT FOUND/
016715	045	022516	041101	NOCLK:	.ASCIZ	/N%ABAD CLOCK - PROGRAM WILL HANG ON "TIMEOUT"!//
016776	040515	027130	041440	TABEX:	.ASCIZ	/MAX. CHAR. MSG COUNT EXCEEDED -/
017036	052502	043106	051105	BUFEX:	.ASCIZ	/BUFFER FULL -/
017054	047045	052045	040445	MSGTRN:	.ASCIZ	/N%T%A MSG. NOT BUILT !!!/
017105	045	022516	041501	MSGTRU:	.ASCIZ	/N%ACHAR. COUNT EXCEEDS BUFF LIMIT - MSG TRUNCATED/
017170	047045	051445	022465	SHFO:	.ASCIZ	?N%S5%AMODE=%T%T%T%A/PASS=%Z5?
017226	047045	051445	022465	SHF1:	.ASCIZ	?N%S5%S5%S5%A/%T%A/%T%A/%T%A/%T?
017266	051445	022465	052101	EFM2:	.ASCIZ	/S5%ATOTAL MISMATCHES IN MSG = %D5/
017331	045	022516	031523	PCPM:	.ASCIZ	/N%S3%ACALLED FROM PC=%O6/
017363	045	032523	040445	EFM11:	.ASCIZ	/S5%ACOMPARE COUNT=%D5%S3%ARECEIVE COUNT=%D5/

:EVENT DESCRIPTION MESSAGES

017440	051124	047101	046523	EDTXQ:	.ASCIZ	/TRANSMIT MSG QUEUED/
017464	051124	047101	046523	EDTXC:	.ASCIZ	/TRANSMIT MSG COMPLETED/
017513	122	041505	044505	EDRXQ:	.ASCIZ	/RECEIVE SPACE QUEUED/
017540	042522	042503	053111	EDRXC:	.ASCIZ	/RECEIVE MSG COMPLETED/
017566	042504	044526	042503	EDDER:	.ASCIZ	/DEVICE ERROR/
017603	104	052101	020101	EDDCK:	.ASCIZ	/DATA COMPARISON STARTED/
017633	104	053105	041511	EDDVI:	.ASCIZ	/DEVICE INIT AND SETUP/
017661	104	052101	020101	EDDLLE:	.ASCIZ	/DATA COMPARISON LENGTH ERROR/

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 95
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

```

:EXECUTION STATUS MESSAGES TO BE PRINTED TO KEEP OPERATOR AWAKE
021650 047045 000 CR: .ASCIIZ /%N/ ;CR FOR LINES IN A ROW
021653 045 031523 040445 STXQ: .ASCIIZ /%S3%ATXQ/ ;ABOUT TO TRANSMIT
021664 051445 022463 052101 STXC: .ASCIIZ /%S3%ATXC/ ;TX COMPLETED
021675 045 031523 040445 SRXQ: .ASCIIZ /%S3%ARXQ/ ;ABOUT TO RECEIVE
021706 051445 022463 042501 SDVE: .ASCIIZ /%S3%AERR/ ;DEVICE ERROR
021717 045 031523 040445 SCM: .ASCIIZ /%S3%ACMP/ ;ABOUT TO DO DATA CHECKING OF RECVD VS. EXPTD
021730 051445 022463 044501 SDVI: .ASCIIZ /%S3%AINI/ ;DEVICE ABOUT TO BE INITIALIZED
021741 045 031523 040445 SCML: .ASCIIZ /%S3%ACML/ ;COMPARE LENGTH ERROR
021752 051445 022463 041501 SCMD: .ASCIIZ /%S3%ACMD/ ;COMPARE DATA ERROR
021763 045 031523 040445 SEOP: .ASCIIZ /%S3%AEOP/ ;END OF PASS
021774 051445 022463 046501 SMSC: .ASCIIZ /%S3%AMSC/ ;MODEM STATUS CHANGE
022006 022006 .EVEN

```

:DEVICE ERROR MESSAGES

```

:DEVICE ERROR MESSAGES FROM ERRORS DETECTED IN PDP-11
022006 046513 020123 040506 DVEM0: .ASCIIZ /KMS FAILED TO START/
022032 046513 020123 044504 DVEM1: .ASCIIZ /KMS DID NOT RESPOND TO CSR COMMAND/
022075 127 044501 044524 DVEM1A: .ASCIIZ /WAITING TO COMPLETE DEVICE INITIALIZATION/
022147 127 044501 044524 DVEM1B: .ASCIIZ /WAITING FOR START DDCMP REQUEST TO COMPLETE/
022223 124 046511 020105 DVEM2: .ASCIIZ /TIME OUT WAITING FOR TX OR RX TO COMPLETE/
022275 113 051515 042040 DVEM3: .ASCIIZ /KMS DID NOT RESPOND TO ROI/
022330 046111 042514 040507 DVEM4: .ASCIIZ /ILLEGAL TRANSMIT COMPLETE/
022361 015 020012 044514 DVEM5: .ASCIIZ <15><12>/ LINE NO BDL NO/
022407 111 046114 043505 DVEM5: .ASCIIZ /ILLEGAL RECEIVE COMPLETE/
022437 015 020012 044514 DVEM5: .ASCIIZ <15><12>/ LINE NO BDL NO/

:DEVICE ERROR MESSAGES FOR CONDITIONS REPORTED BY KMS11
022465 104 041504 050115 DVEM11: .ASCIIZ /DDCMP RESTART RECEIVED/
022514 046513 020123 044524 DVEM14: .ASCIIZ /KMS TIMEOUT WAITING FOR RESPONSE TO TX/
022563 113 051515 052040 DVM14B: .ASCIIZ /KMS TIMEOUT. NO RESPONSE TO START MSG/
022631 124 020130 044124 DVEM15: .ASCIIZ /TX THRESHOLD REACHED/
022655 015 020012 044514 DVEM15: .ASCIIZ <15><12>/ LINE NO NAK REASON/
022705 124 020130 042522 DVEM17: .ASCIIZ /TX REJECTED BEFORE COMPLETION/
022742 005015 046040 047111 DVEM17: .ASCIIZ <15><12>/ LINE NO BDL NO/
022766 046513 020123 044524 DVEM33: .ASCIIZ /KMS TIMEOUT WAITING FOR CSR INPUT/
023030 046513 020123 044524 DVEM34: .ASCIIZ /KMS TIMEOUT. CSR OUTPUT NOT ACCEPTED/

```

:NAK REASON MESSAGES

```

023075 110 040505 042504 NAKR1: .ASCIIZ /HEADER BLOCK CHECK ERROR/
023126 040504 040524 041040 NAKR2: .ASCIIZ /DATA BLOCK CHECK ERROR/
023155 122 050105 051040 NAKR3: .ASCIIZ /REP RESPONSE/
023172 052502 043106 051105 NAKR8: .ASCIIZ /BUFFER UNAVAILABLE/
023215 122 041505 044505 NAKR9: .ASCIIZ /RECEIVE OVERRUN/
023235 115 051505 040523 NAKR16: .ASCIIZ /MESSAGE TOO LONG/
023256 051515 020107 042510 NAKR17: .ASCIIZ /MSG HEADER FORMAT ERROR/

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 96
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

023306 040516 020113 042522 NAKR00: .ASCIZ /NAK REASON NOT RECOGNIZED/

.LIST .EVEN
BEX

3254
3255
3256
3257
3258
3259
3260
3261

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 97
GLOBAL ERROR REPORT SECTION

..+.
: THE GLOBAL ERROR REPORT SECTION CONTAINS MESSAGE PRINTING AREAS
: USED BY MORE THAN TEST TO OUTPUT ADDITIONAL ERROR INFORMATION. PRINTB
: (BASIC) AND PRINTX (EXTENDED) CALLS ARE USED TO CALL PRINT SERVICES.
:--

3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312
3313
3314
3315
3316
3317

023340
023340
023340
023340 005046
023342 153716 010205
023346 005046
023350 153716 010204
023354 013746 010162
023360 012746 021057
023364 012746 000004
023370 010600
023372 104414
023374 062706 000012
023400
023400 104423
023402
023402
023402 013746 010174
023406 012746 017266
023412 012746 000002
023416 010600
023420 104414
023422 062706 000006
023426
023426 104423
023430
023430
023430 013746 010172
023434 010446
023436 012746 017363
023442 012746 000003
023446 010600
023450 104414
023452 062706 000010
023456
023456
023456 104423

BGNMSG ERR1

PRINTB #EVTF5A,OFFSET,<B,GOOD>,<B,BAD>

ENDMSG

BGNMSG ERR2

PRINTB #EFM2,TEMP4

ENDMSG

BGNMSG ERR10

PRINTB #EFM11,R4,TEMP3

ENDMSG

ERR1::
;INDIVIDUAL DATA COMPARE ERROR
CLR -(SP)
BISB BAD,(SP)
CLR -(SP)
BISB GOOD,(SP)
MOV OFFSET, -(SP)
MOV #EVTF5A, -(SP)
MOV #4, -(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #12,SP

L10001: TRAP C\$MSG

ERR2::
;TOTAL DATA COMPARE FAILS ERROR
MOV TEMP4, -(SP)
MOV #EFM2, -(SP)
MOV #2, -(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #6,SP

L10002: TRAP C\$MSG

ERR10::
;LENGTH COMPARISON ERROR
MOV TEMP3, -(SP)
MOV R4, -(SP)
MOV #EFM11, -(SP)
MOV #3, -(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #10,SP

L10003: TRAP C\$MSG

;PRINT SEL0 = XXXXXX SEL2 = XXXXXX WITH VALUES FROM TEMP4..5

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 98
GLOBAL ERROR REPORT SECTION

3318							
3319	023460			BGNMSG	ERR13		
3320	023460					ERR13::	
3321	023460			PRINTB	#EFT3C,TEMP4,TEMP5		
3322	023460	013746	010176			MOV	TEMP5,-(SP)
3323	023464	013746	010174			MOV	TEMP4,-(SP)
3324	023470	012746	020330			MOV	#EFT3C,-(SP)
3325	023474	012746	000003			MOV	#3,-(SP)
3326	023500	010600				MOV	SP,R0
3327	023502	104414				TRAP	C\$PNTB
3328	023504	062706	000010			ADD	#10,SP
3329	023510			ENDMSG			
3330	023510					L10004:	
3331	023510	104423				TRAP	C\$MSG
3332							
3333							
3334				BGNMSG	: PRINT "LINE NO = " FOLLOWED BY OCTAL BYTE IN TEMP3+1		
3335	023512			BGNMSG	ERR14	ERR14::	
3336	023512						
3337	023512			PRINTB	#EVT40,<B,TEMP3+1>		
3338	023512	005046				CLR	-(SP)
3339	023514	153716	010173			BISB	TEMP3+1,(SP)
3340	023520	012746	020446			MOV	#EVT40,-(SP)
3341	023524	012746	000002			MOV	#2,-(SP)
3342	023530	010600				MOV	SP,R0
3343	023532	104414				TRAP	C\$PNTB
3344	023534	062706	000006			ADD	#6,SP
3345	023540			ENDMSG			
3346	023540					L10005:	
3347	023540	104423				TRAP	C\$MSG
3348							
3349							
3350				BGNMSG	: PRINT TWO OCTAL BYTES FROM TEMP3+1, TEMP4		
3351	023542			BGNMSG	ERR22	ERR22::	
3352	023542						
3353	023542			PRINTB	#EVT42,<B,TEMP3+1>,<B,TEMP4>		
3354	023542	005046				CLR	-(SP)
3355	023544	153716	010174			BISB	TEMP4,(SP)
3356	023550	005046				CLR	-(SP)
3357	023552	153716	010173			BISB	TEMP3+1,(SP)
3358	023556	012746	020472			MOV	#EVT42,-(SP)
3359	023562	012746	000003			MOV	#3,-(SP)
3360	023566	010600				MOV	SP,R0
3361	023570	104414				TRAP	C\$PNTB
3362	023572	062706	000010			ADD	#10,SP
3363	023576			ENDMSG			
3364	023576					L10006:	
3365	023576	104423				TRAP	C\$MSG
3366							

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 99
GLOBAL ERROR REPORT SECTION

```

3367
3368 023600
3369 023600
3370 023600
3371 023600 005046
3372 023602 153716 010175
3373 023606 005046
3374 023610 153716 010174
3375 023614 005046
3376 023616 153716 010173
3377 023622 012746 020507
3378 023626 012746 000004
3379 023632 010600
3380 023634 104414
3381 023636 062706 000012
3382 023642
3383 023642
3384 023642 104423
3385
3386
3387 023644
3388 023644
3389 023644
3390 023644 013746 010174
3391 023650 005046
3392 023652 153716 010173
3393 023656 012746 020532
3394 023662 012746 000003
3395 023666 010600
3396 023670 104414
3397 023672 062706 000010
3398 023676
3399 023676
3400 023676 104423

```

```

: PRINT THREE OCTAL BYTES IN TEMP3+1, TEMP4, TEMP4+1
BGNMSG ERR23
PRINTB #EVT43,<B,TEMP3+1>,<B,TEMP4>,<B,TEMP4+1>

```

ERR23::

```

CLR -(SP)
BISB TEMP4+1,(SP)
CLR -(SP)
BISB TEMP4,(SP)
CLR -(SP)
BISB TEMP3+1,(SP)
MOV #EVT43,-(SP)
MOV #4,-(SP)
MOV SP,RO
TRAP C$PNTB
ADD #12,SP

```

ENDMSG

L10007:

TRAP C\$MSG

```

: PRINT 1 BYTE FOLLOWED BY MESSAGE POINTED TO IN TEMP4
BGNMSG ERR24
PRINTB #EVT44,<B,TEMP3+1>,TEMP4

```

ERR24::

```

MOV TEMP4,-(SP)
CLR -(SP)
BISB TEMP3+1,(SP)
MOV #EVT44,-(SP)
MOV #3,-(SP)
MOV SP,RO
TRAP C$PNTB
ADD #10,SP

```

ENDMSG

L10010:

TRAP C\$MSG

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 100
GLOBAL ERROR REPORT SECTION

3401
3402 023700
3403 023700
3404
3405 023700
3406 023700 005046
3407 023702 153716 010175
3408 023706 005046
3409 023710 153716 010174
3410 023714 012746 021207
3411 023720 012746 000003
3412 023724 010500
3413 023726 104414
3414 023730 062706 000010
3415 023734
3416 023734 005046
3417 023736 153716 010177
3418 023742 005046
3419 023744 153716 010176
3420 023750 012746 021264
3421 023754 012746 000003
3422 023760 010600
3423 023762 104414
3424 023764 062706 000010
3425 023770
3426 023770
3427 023770 104423
3428
3429 023772
3430 023772 000167
3431 023774 177772
3432
3433

BGNMSG ERR25

ERR25::
:PRINT 4 STATUS BYTES FROM TEMP4,TEMP5
PRINTB #EVTFS2,<B,TEMP4>,<B,TEMP4+1>

CLR -(SP)
BISB TEMP4+1,(SP)
CLR -(SP)
BISB TEMP4,(SP)
MOV #EVTFS2,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #10,SP
CLR -(SP)
BISB TEMP5+1,(SP)
CLR -(SP)
BISB TEMP5,(SP)
MOV #EVTFS3,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C\$PNTB
ADD #10,SP

PRINTB #EVTFS3,<B,TEMP5>,<B,TEMP5+1>

ENDMSG

L10011:

TRAP C\$MSG

EXIT MSG

.WORD JSJMP
.WORD L10011-2-

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 101
GLOBAL SUBROUTINES SECTION

3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478

..++
: THE GLOBAL SUBROUTINES SECTION CONTAINS THE SUBROUTINES
: THAT ARE USED IN MORE THAN ONE TEST.
:--

.SBTTL CLOCK SETUP SUBROUTINE

..++
: FUNCTIONAL DESCRIPTION:
: THIS SUBROUTINE SETS UP THE CLOCK INFORMATION TABLE FOLLOWING A "CLOCK"
: CALL EXECUTED IN THE INITIALIZATION CODE. BUT SINCE THE "CLOCK" CALL
: SAYS NOTHING ABOUT AN LSI-11'S CLOCK, THIS ROUTINE IS ONLY USED IF A
: LINE OR P-CLOCK IS FOUND.

INPUTS:
R1= POINTS TO SUPERVISOR SPACE WHERE CLOCK INFO WAS RETURNED
R2= POINTS TO "CLK" TABLE WHERE CLOCK INFO WILL BE KEPT

IMPLICIT INPUTS:
THE SUPERVISOR SPACE WHERE CLOCK INFO WAS RETURNED BY THE "CLOCK" CALL

OUTPUTS:
"CLKCSR" GETS LOADED WITH THE CLOCK'S CSR ADDRESS
"CLKBR" GETS LOADED WITH THE CLOCK'S INTERRUPT LEVEL
"CLKVEC" GETS LOADED WITH THE CLOCK'S INTERRUPT VECTOR
"CLKHZ" GETS LOADED WITH THE LINE FREQ. (HERTZ RATE) WHICH DETERMINES
THE NUMBER OF TICKS IN A SECOND

CALLING SEQUENCE:
JSR PC,CLKSET ;CALL CLOCK SETUP WITH R1 & R2 SETUP

..--
CLKSET:
MOV (R1)+,(R2)+ ;LOAD CLOCK'S CSR ADDR. INTO "CLKCSR"
MOV (R1)+,(R2) ;LOAD CLOCK'S INT. LEVEL INTO "CLKBR"
ASL (R2) ;ADJUST THE INT. LEVEL FOR LOADING INTO
; THE PSW WITH A "SETVEC" CALL
ASL (R2)
ASL (R2)
ASL (R2)
ASL (R2)+
MOV (R1)+,(R2)+ ;LOAD CLOCK'S INT. VECTOR INTO "CLKVEC"
MOV (R1)+,(R2)+ ;LOAD CLOCK'S HERTZ RATE INTO "CLKHZ"
RTS PC

023776
023776 012122
024000 012112
024002 006312
024004 006312
024006 006312
024010 006312
024012 006322
024014 012122
024016 012122
024020 000207

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 102
CLOCK INTERRUPT SERVICE ROUTINE

3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534

.SBTTL CLOCK INTERRUPT SERVICE ROUTINE

```

:++
: FUNCTIONAL DESCRIPTION:
: THIS IS THE CLOCK INTERRUPT SERVICE ROUTINE WHICH TAKES CARE OF
: KEEPING THE "TIME-SINCE-START" AND COUNTING DOWN ANY OF THE
: "EVENT" TIMERS. THE TIMERS ARE USED TO TIME COMPLETION OF DEVICE
: REQUESTS. THE "TIME-SINCE-START" IS USED TO BE LOGGED WITH EACH ENTRY
: INTO THE EVENT LOG.
:
: IMPLICIT INPUTS:
: TIMTCK: THE CURRENT NO. OF TICKS LEFT TO BE COUNTED UNTIL A SECOND
: WAS BEEN COUNTED OFF
: CLKHZ: THE NO. OF TICKS IN A SECOND, DETERMINED BY THE SYS. LINE FREQ.
: TIMMIN & TIMSEC: CURRENT VALUE OF "TIME-SINCE-START"
: IN MINUTES & SECONDS
: TIMER 1,2, & S: CURRENT VALUES OF THE "EVENT TIMERS"
:
: IMPLICIT OUTPUTS:
: NEW VALUE OF EVENT TIMER "1" DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
: NEW VALUE OF EVENT TIMER "2" DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
: NEW VALUE OF EVENT TIMER "S" DECREMENTED BY 1 SECOND IF IT WAS NON-ZERO
:
: FUNCTIONAL SIDE EFFECTS:
: THE CLOCK IS DISABLED UPON ENTRY AND REENABLED WHEN LEAVING
:
: CALLING SEQUENCE:
: THIS ROUTINE IS CALLED WHEN THE CLOCK INTERRUPTS THRU "CLKVEC".
: THE ADDRESS OF THIS ROUTINE WAS LOADED INTO THE CLOCK'S INTERRUPT
: VECTOR WITH A SUPERVISOR "SETVEC" CALL.
:--

```

```

BGNSRV CLKINT CLKINT::
CLR @CLKCSR ;DISABLE THE CLOCK FROM INTERRUPTING
DEC TIMTCK ;DECREMENT THE # OF TICKS/SEC.
BNE 1$ ;GO CHECK TIMERS (1$2-TICKS, 3-SECONDS)
MOV CLKHZ,TIMTCK ;RESET THE # OF TICKS/SEC.
INC TIMSEC ;INC # OF SECS-SINCE-START
CMP #60.,TIMSEC ;SEE IF WE'VE COUNTED 60 SECS. YET
BNE 1$ ;IF NOT, GO CHECK TIMERS
INC TIMMIN ; ELSE INC MINUTES-SINCE-START
CLR TIMSEC ; AND RESTART SECOND COUNTER

1$: TST TIMER1 ;SEE IF TIMER #1, TIMING ANYTHING
BEQ 2$ ; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
DEC TIMER1 ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
2$: TST TIMER2 ;SEE IF TIMER #2, TIMING ANYTHING
BEQ 3$ ; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
DEC TIMER2 ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
3$: TST TIMERS ;SEE IF TIMER #3, TIMING ANYTHING
BEQ 4$ ; IF=0, NOTHING BEING TIMED, LEAVE
CMP CLKHZ,TIMTCK ;SEE IF A SECOND HAS BEEN COUNTED OFF
BNE 4$ ; BR IF NO
DEC TIMERS ; ELSE DECREMENT THE TIMER VALUE (BY 1 SEC.)

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 103
CLOCK INTERRUPT SERVICE ROUTINE

3535 024134 013777 010260 164106 48:
3536 024142
3537 024142
3538 024142 000002

MOV CLKEN,@CLKCSR ;REENABLE THE CLOCK TO INTERRUPT
ENDSRV

L10012:
RTI

3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3590
3591
3592

.SBTTL EVENT LOG SUBROUTINES

```

:++
: FUNCTIONAL DESCRIPTION:
: THIS SUBROUTINE HAS A DIFFERENT ENTRY POINT
: FOR EACH EVENT TO BE LOGGED AND ALWAYS PRINTS
: THE SHORT "OPERATOR AWAKE" MESSAGE TO CONSOLE THEN LOGS THE
: EVENT TYPE, TIME, AND THE OTHER 3 WORDS OF INFO PASSED TO THE
: SUBROUTINE AT CALLING TIME
:
: INPUTS:
: TIMMIN & TIMSEC: CURRENT VALUE OF "TIME-SINCE-START"
: TEMP2: WORD #1 OF EVENT LOG INFORMATION (FOR MOST EVENT TYPES)
: TEMP3: WORD #2 OF EVENT LOG INFORMATION
: TEMP4: WORD #3 OF EVENT LOG INFORMATION
:
: OUTPUTS:
: "OPERATOR AWAKE" MESSAGE SENT TO THE CONSOLE
: NEW EVENT LOGGED IN "EVTLOG" (EVENT LOG)
: UPDATED "EVTPTN" (EVENT LOG ENTRY POINTER)
:
: FUNCTIONAL SIDE EFFECTS:
: TEMP: USED TO STORE ADDRESS OF "OPERATOR AWAKE" MESSAGE
: TEMP1: USED TO SETUP THE VALUE OF THE "EVENT TYPE" BYTE FOR LOGGING
:
: CALLING SEQUENCE:
: JSR PC,LOGTXQ ;CALL THE LOG EVENT SUBROUTINE WITH TEMP,TEMP1,
: ; TEMP2, TEMP3, AND TEMP4 SETUP
:
: JSR PC,LOGCMP
:--

```

```

LOGTXQ: MOV #STXQ,TEMP1 ;SET UP MSG. TO PRINT
        MOV #TXQ,TEMP ;SET UP EVENT TYPE
        BR LOGS1 ;GO LOG EVENT AND TIME
LOGTXC: MOV #STXC,TEMP1 ;SET UP MSG. TO PRINT
        MOV #TXC,TEMP ;SET UP EVENT TYPE
        BR LOGS1 ;GO LOG EVENT AND TIME
LOGRXQ: MOV #SRXQ,TEMP1 ;SET UP MSG. TO PRINT
        MOV #RXQ,TEMP ;SET UP EVENT TYPE
        BR LOGS1 ;GO LOG EVENT AND TIME
LOGRXC: MOV #RXC,TEMP ;SET UP EVENT TYPE
        BR LOGS1 ;GO LOG EVENT AND TIME
LGDVE: MOV #SDVE,TEMP1 ;SET UP MSG. TO PRINT
        MOV #DER,TEMP ;SET UP EVENT TYPE
        BR LOGS3 ;GO LOG EVENT AND TIME

```

```

024144
024144 012737 021653 010166
024152 012737 000000 010164
024160 000522
024162
024162 012737 021664 010166
024170 012737 000002 010164
024176 000513
024200
024200 012737 021675 010166
024206 012737 000004 010164
024214 000504
024216
024216 012737 000006 010164
024224 000500
024226
024226 012737 021706 010166
024234 012737 000010 010164
024242 000474

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 105
EVENT LOG SUBROUTINES

3593	024244				LOGDVI:						
3594	024244	012737	021730	010166		MOV	#SDVI,TEMP1		:SET UP MSG. TO PRINT		
3595	024252	012737	000012	010164		MOV	#DVI,TEMP		:SET UP EVENT TYPE		
3596	024260	113737	010216	010170		MOVB	MODTYP,TEMP2				
3597	024266	113737	010220	010171		MOVB	MLTYP,TEMP2+1				
3598	024274	013737	010226	010172		MOV	RPASS,TEMP3				
3599	024302	013737	010224	010174		MOV	PARAM,TEMP4		:SET UP EVNT ENTRIES		
3600	024310	000451				BR	LOGS3		:GO LOG EVENT AND TIME		
3601											
3602	024312				LOGCMP:						
3603	024312	012737	021717	010166		MOV	#SCM,TEMP1		:SET UP MSG. TO PRINT		
3604	024320	012737	000014	010164		MOV	#DCK,TEMP		:SET UP EVENT TYPE		
3605	024326	000415				BR	LOGS3A				
3606	024330				LOGCML:						
3607	024330	012737	021741	010166		MOV	#SCML,TEMP1				
3608	024336	012737	000020	010164		MOV	#DLE,TEMP		:SET UP MSG. AND TYPE		
3609	024344	000406				BR	LOGS3A		:GO LOG EVENT AND TIME		
3610	024346				LOGCMD:						
3611	024346	012737	021752	010166		MOV	#SCMD,TEMP1				
3612	024354	012737	000022	010164		MOV	#DDE,TEMP				
3613	024362	013737	007636	010176	LOGS3A:	MOV	TRIBN,TEMP5				
3614	024370	000421				BR	LOGS3		:GO LOG MSG TYPE AND TIME		
3615											
3616	024372				LOGEOP:						
3617	024372	012737	021763	010166		MOV	#SEOP,TEMP1				
3618	024400	012737	000024	010164		MOV	#EOP,TEMP				
3619	024406	000412				BR	LOGS3		:GO LOG MSG TYPE AND TIME		
3620											
3621	024410				LOGMSC:						
3622	024410	012737	021774	010166		MOV	#SMSC,TEMP1				
3623	024416	012737	000016	010164		MOV	#MSC,TEMP				
3624	024424	000403				BR	LOGS3				
3625											
3626	024426	013737	007636	010176	LOGS1:	MOV	TRIBN,TEMP5		:SAVE LINE NUMBER (TRIBN)		
3627											
3628	024434				LOGS3:						
3629	024434	022737	000006	010164		CMP	#RXC,TEMP				
3630	024442	001434				BEG	LOGS5		:IF RXC DON'T PRINT		
3631	024444	032737	000001	010224		BIT	#STATB,PARAM				
3632	024452	001430				BEG	LOGS5		:IF NO STATUS SELECTED		
3633									:GO TO 5		
3634											
3635	024454	022737	000010	010114		CMP	#10,LNCNT		:HAVE WE DONE 10?		
3636	024462	001012				BNE	LOGS4		:IF NOT GO TO 4		
3637	024464	005037	010114			CLR	LNCNT		:ELSE CLEAR IT		
3638											
3639	024470					PRINTF	#CR		:ELSE PRINT CR		
3640	024470	012746	021650							MOV	#CR,-(SP)
3641	024474	012746	000001							MOV	#1,-(SP)
3642	024500	010600								MOV	SP,R0
3643	024502	104417								TRAP	C\$PNTF
3644	024504	062706	000004							ADD	#4,SP

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 106
EVENT LOG SUBROUTINES

```

3645 024510
3646 024510 005237 010114
3647 024514
3648 024514 013746 010166
3649 024520 012746 000001
3650 024524 010600
3651 024526 104417
3652 024530 062706 000004
3653 024534 010346
3654 024536 013703 010276
3655 024542 113723 010164
3656 024546 013737 010256 010164
3657 024554 163737 010266 010164
3658 024562 113723 010164
3659 024566 113723 010264
3660 024572 113723 010262
3661 024576 013723 010170
3662 024602 013723 010172
3663 024606 013723 010174
3664 024612 013723 010176
3665 024616 020327 011334
3666 024622 103404
3667
3668 024624 012713 177777
3669 024630 012703 010300
3670 024634 010337 010276
3671 024640 012603
3672 024642 000207
3673
3674

```

```

LOGS4: INC LNCNT ;INC COUNTER OF # OF AWAKE MSGS
PRINTF TEMP1 ;PRINT OPERATOR AWAKE MSG.

```

```

MOV TEMP1,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #4,SP

```

```

LOGS5: MOV R3,-(SP) ;SAVE R3 ON THE STACK
MOV EVTPTR,R3
MOVB TEMP,(R3)+ ;LOG EVENT
MOV CLKHZ,TEMP
SUB TIMTCK,TEMP
MOVB TEMP,(R3)+ ;LOG TIME SINCE START
MOVB TIMSEC,(R3)+ ;TICKS,SECS AND MINS.
MOVB TIMMIN,(R3)+
MOV TEMP2,(R3)+ ;LOG EVNT ENTRY 3
MOV TEMP3,(R3)+ ;LOG EVNT ENTRY 4
MOV TEMP4,(R3)+ ;LOG EVNT ENTRY 5
MOV TEMP5,(R3)+ ;LOG EVNT ENTRY 6
CMP R3,#EVTEND
BLO LOGS2

```

```

LOGS2: MOV #-1,(R3) ;IF EVENT LOG FULL GO
MOV #EVTLOG,R3 ;CONTINUE;ELSE GO TO 2
MOV R3,EVTPTR ;LOG A TABLE END
MOV (SP)+,R3 ;PUT R3 TO START OF TABLE
RTS ;RESTORE POINTER
PC ;RESTORE R3

```

LOGEX: RTS

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 107
DUMP EVENT LOG AND BASE TABLE

```

3675
3676
3677
3678 024644 010246
3679 024646 010346
3680 024650 010446
3681
3682
3683
3684 024652
3685 024652 012746 015551
3686 024656 012746 000001
3687 024662 010600
3688 024664 104417
3689 024666 062706 000004
3690 024672 105037 003401
3691 024676 105037 003400
3692
3693
3694
3695 024702
3696 024702 104443
3697 024704 000406
3698 024706 003124
3699 024710 000142
3700 024712 015544
3701 024714 000377
3702 024716 000001
3703 024720 000110
3704 024722
3705 024722 012737 003124 003364
3706 024730 012737 027070 003366
3707 024736 012737 026650 003370
3708 024744 005037 003250
3709 024750 004737 031506
3710 024754 105737 003401
3711 024760 001412
3712 024762
3713 024762 012746 014010
3714 024766 012746 000001
3715 024772 010600
3716 024774 104417
3717 024776 062706 000004
3718 025002 000137 024672
3719
3720 025006 105737 003400
3721 025012 001411
3722 025014
3723 025014 012746 014040
3724 025020 012746 000001
3725 025024 010600
3726 025026 104417
3727 025030 062706 000004
3728 025034 000716

```

.SBTTL DUMP EVENT LOG AND BASE TABLE

```

REPORT: MOV R2,-(SP) ;SAVE R2,R3,R4 ON THE STACK
        MOV R3,-(SP)
        MOV R4,-(SP)

```

;PRINT REPORT HELP MESSAGE

PRINTF #RHLPO

```

MOV #RHLPO,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #4,SP

```

```

GETRCL: CLRB P$GDBD ;CLEAR GOOD BAD FLAG
        CLRB P$NNUF

```

;PRINT PROMPT RPT>

GMANID CLISRP,CMDBUF,A,377,1,72..NO

```

TRAP C$GMAN
BR 10000$
.WORD CMDBUF
.WORD T$CODE
.WORD CLISRP
.WORD 377
.WORD T$LOLIM
.WORD T$HILIM

```

10000\$:

```

MOV #CMDBUF,P$BUFA
MOV #CLIRT,P$TREE
MOV #CLIRAC,P$ACT
CLR QUALFG
JSR PC,P$TRV
TSTB P$GDBD
BEQ 1$
PRINTF #CLIERM

```

```

;CLEAR QUALIFIER FLAG LOCATION
;GO PARSE COMMAND LINE
;SEE IF PARSED OK OR AN ERROR
;OK. CONTINUE PROCESSING COMMAND
;ERROR. REPORT INVALID COMMAND

```

```

MOV #CLIERM,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #4,SP

```

JMP GETRCL

;GO PRINT RPT> PROMPT AGAIN

```

1$: TSTB P$NNUF
   BEQ 10$
   PRINTF #CLINUF

```

```

;WAS COMPLETE COMMAND TYPED ?
;YES. GO PERFORM REQUESTED ACTION
;NO. REPORT INCOMPLETE COMMAND TYPED

```

```

MOV #CLINUF,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #4,SP

```

BR GETRCL

;GO PRINT RPT> PROMPT AGAIN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 108
DUMP EVENT LOG AND BASE TABLE

```

3729 025036          10$:
3730 025036 023727 003246 000004  CMP   KEYWD1,#RPSOL      ;PERFORM REQUESTED REPORT ACTION
3731 025044 001003          BNE   20$              ;WAS A DEVICE STATUS REPORT REQUESTED ?
3732          004737 025112          JSR   PC,RPTSOL        ;DEVICE STATUS REPORT REQUESTED
3733 025046 000707          BR    GETRCL           ;YES. REPORT DEVICE STATUS
3734 025052          20$:          ;GO PRINT RPT> PROMPT AGAIN
3735          000006 20$:          CMP   KEYWD1,#RDMP5    ;IS DUMP REQUESTED ?
3736 025054 023727 003246 000006  BNE   30$              ;NO.
3737 025062 001003          ;DUMP DATA REQUESTED
3738          004737 027264          JSR   PC,DUMPSR        ;YES. PRINT REQUESTED DATA
3739 025064 000700          BR    GETRCL           ;GO PRINT RPT> PROMPT
3740 025070          30$:          CMP   KEYWD1,#RPEXT    ;IS EXIT REPORT SECTION REQUESTED ?
3741 025072 023727 003246 000002  BNE   GETRCL           ;NO. GO PRINT RPT> PROMPT AGAIN
3742 025100 001274          ;EXIT REPORT PROCESSOR REQUESTED
3743          ;YES. RESTORE R4,R3,R2
3744          ENDALL: MOV   (SP)+,R4
3745          MOV   (SP)+,R3
3746 025102 012604          MOV   (SP)+,R2
3747 025104 012603          RTS   PC              ;RETURN TO CALLING ROUTINE
3748 025106 012602
3749 025110 000207
3750
3751
3752

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 110
RPTSOL -GET LINE STATUS REPORT FROM DEVICE-

3801	025246		
3802	025246	005046	
3803	025250	153716	011503
3804	025254	005046	
3805	025256	153716	011502
3806	025262	012746	021264
3807	025266	012746	000003
3808	025272	010600	
3809	025274	104416	
3810	025276	062706	000010
3811	025302		
3812	025302	005046	
3813	025304	153716	011505
3814	025310	005046	
3815	025312	153716	011504
3816	025316	012746	021341
3817	025322	012746	000003
3818	025326	010600	
3819	025330	104416	
3820	025332	062706	000010
3821	025336		
3822	025336	005046	
3823	025340	153716	011507
3824	025344	005046	
3825	025346	153716	011506
3826	025352	012746	021416
3827	025356	012746	000003
3828	025362	010600	
3829	025364	104416	
3830	025366	062706	000010
3831	025372		
3832	025372	000207	
3833			
3834			
3835			

PRINTS #EVTFS3,<B,DEVS2>,<B,DEVS2+1>

```

CLR      -(SP)
BISB    DEVS2+1,(SP)
CLR      -(SP)
BISB    DEVS2,(SP)
MOV     #EVTFS3,-(SP)
MOV     #3,-(SP)
MOV     SP,RO
TRAP    C$PNTS
ADD     #10,SP

```

PRINTS #EVTFS4,<B,DEVS3>,<B,DEVS3+1>

```

CLR      -(SP)
BISB    DEVS3+1,(SP)
CLR      -(SP)
BISB    DEVS3,(SP)
MOV     #EVTFS4,-(SP)
MOV     #3,-(SP)
MOV     SP,RO
TRAP    C$PNTS
ADD     #10,SP

```

PRINTS #EVTFS5,<B,DEVS4>,<B,DEVS4+1>

```

CLR      -(SP)
BISB    DEVS4+1,(SP)
CLR      -(SP)
BISB    DEVS4,(SP)
MOV     #EVTFS5,-(SP)
MOV     #3,-(SP)
MOV     SP,RO
TRAP    C$PNTS
ADD     #10,SP

```

30\$: RTS PC ;RETURN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 111
REPROG PRINT EVENT LOG

.SBTTL REPROG PRINT EVENT LOG

```

3836
3837
3838
3839 025374 010246      REPROG: MOV    R2,-(SP)      ;SAVE R2,R3,R4 ON THE STACK
3840 025376 010346      MOV    R3,-(SP)
3841 025400 010446      MOV    R4,-(SP)
3842
3843 025402 013702 010276      MOV    EVTPT,R2      ;MAKE R2 A POINTER TO EVENT TABLE
3844 025406 023727 010300 177777      CMP    EVTLOG,#-1    ;SEE IF EVENT TABLE IS EMPTY
3845 025414 001034      BNE   RPT0           ;BR IF NO
3846 025416      PRINTS #NULEVT      ;IF EMPTY TELL OPERATOR.
3847 025416 012746 020057      MOV    #NULEVT,-(SP)
3848 025422 012746 000001      MOV    #1,-(SP)
3849 025426 010600      MOV    SP,R0
3850 025430 104416      TRAP  C$PNTS
3851 025432 062706 000004      ADD   #4,SP
3852 025436 000137 026606      JMP   ENDEVT        ;AND END
3853
3854 025442 162702 000014      RPT:  SUB   #14,R2   ;NOW POINT BACK TO TOP OF ENTRY U
3855      ;JUST PRINTED
3856
3857 025446 020227 010300      CMP    R2,#EVTLOG   ;POINTING TO TOP OF EVNT LOG QUEUE?
3858 025452 001010      BNE   RPT1          ;BR IF NO
3859 025454 012702 011334      MOV    #EVTEND,R2  ;SET R2 TO POINT TO BOTTOM OF LOG
3860 025460 026227 177776 177777      CMP    -2(R2),#-1
3861 025466 001007      BNE   RPT0          ;IF END OF LOG IS NOT EMPTY
3862 025470 000137 026606      JMP   ENDEVT        ;CONTINUE...ELSE EXIT
3863
3864 025474 020237 010276      RPT1: CMP    R2,EVTPT  ;ARE WE BACK TO POINTER?
3865 025500 001002      BNE   RPT0          ;IF NOT CONTINUE
3866 025502 000137 026606      JMP   ENDEVT        ;IF SO EXIT....
3867
3868 025506 162702 000014      RPT0: SUB   #14,R2   ;POINT R2 TO START OF ENTRY
3869 025512      RPTAA: PRINTS #EVTFO ;PRINT EVENT ENTRY HEADER
3870 025512 012746 020117      MOV    #EVTFO,-(SP)
3871 025516 012746 000001      MOV    #1,-(SP)
3872 025522 010600      MOV    SP,R0
3873 025524 104416      TRAP  C$PNTS
3874 025526 062706 000004      ADD   #4,SP
3875 025532 112203      MOV    (R2)+,R3     ;PUT EVENT TYPE INTO R3
3876 025534 112237 011430      MOV    (R2)+,EVTICK ;PUT EVENT TIME (TICKS,SECS,MINS IN TEMP LOC.S)
3877 025540 112237 011424      MOV    (R2)+,EVTSEC
3878 025544 112237 011426      MOV    (R2)+,EVTMIN
3879 025550      PRINTS #EVTf1,EVTMIN,EVTSEC,EVTICK,EVTLS(T(R3) ;PRINT EVENT TIME AND DESCRIPT.
3880 025550 016346 011374      MOV    EVTLS(T(R3),-(SP)
3881 025554 013746 011430      MOV    EVTICK,-(SP)
3882 025560 013746 011424      MOV    EVTSEC,-(SP)
3883 025564 013746 011426      MOV    EVTMIN,-(SP)
3884 025570 012746 020215      MOV    #EVTf1,-(SP)
3885 025574 012746 000005      MOV    #5,-(SP)
3886 025600 010600      MOV    SP,R0
3887 025602 104416      TRAP  C$PNTS
3888 025604 062706 000014      ADD   #14,SP
3889 025610 000173 011440      JMP   @RPTDSP(R3)  ;DISPATCH TO DECODING SECTION FOR SPECIFIC TYPE

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 112
REPLUG PRINT EVENT LOG

3890	025614	012237	011432
3891	025620	012237	011434
3892	025624	012203	
3893	025626	004737	026616
3894	025632		
3895	025632	013746	011434
3896	025636	013746	011432
3897	025642	012746	020244
3898	025646	012746	000003
3899	025652	010600	
3900	025654	104416	
3901	025656	062706	000010
3902	025662	000137	025442

```

RPTTXQ: MOV      (R2)+,EVTADD      ;STORE MESSAGE ADDRESS FOR PRINTING
          MOV      (R2)+,EVTBCT     ;STORE BYTE COUNT FOR PRINTING
          MOV      (R2)+,R3         ;STORE MODEM STATUS FOR PRINTING
          JSR      PC,PNTTRB        ;PRINT LINE NO.
          PRINTS  #EVTF2,EVTADD,EVTBCT ;PRINT ADDR,BYTE CNT

```

```

MOV      EVTBCT,-(SP)
MOV      EVTADD,-(SP)
MOV      #EVTF2,-(SP)
MOV      #3,-(SP)
MOV      SP,R0
TRAP    C$PNTS
ADD     #10,SP

```

```

JMP      RPT          ;GO BACK FOR NEXT EVENT ENTRY

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 113
REPLOG PRINT EVENT LOG

3903
3904 025666
3905 025666 012237 011436
3906 025672
3907 025672 013746 011436
3908 025676 012746 020316
3909 025702 012746 000002
3910 025706 010600
3911 025710 104416
3912 025712 062706 000006
3913 025716 012237 011470
3914 025722 012237 011472
3915 025726 012237 011474
3916 025732 010246
3917 025734 005003
3918 025736 113703 011470
3919 025742 006303
3920 025744 016303 025764
3921 025750 062703 025764
3922 025754 004713
3923 025756 012602
3924 025760 000137 025442
3925
3926 025764
3927 025764 000016
3928 025766 000050
3929 025770 000052
3930 025772 000110
3931 025774 000154
3932 025776 000204
3933 026000 000240
3934 026002
3935 026002
3936 026002 013746 011474
3937 026006 013746 011472
3938 026012 012746 020330
3939 026016 012746 000003
3940 026022 010600
3941 026024 104416
3942 026026 062706 000010
3943 026032 000207
3944
3945
3946 026034
3947 026034 000207
3948

```

RPTDER:      ;PRINT ERROR EVENT FROM LOG
MOV          (R2)+,EVTMP      ;GET ADDRESS OF DEVICE INFO MESSAGE
PRINTS      #EVTFS,EVTMP     ;PRINT MESSAGE
MOV          EVTTMP,-(SP)
MOV          #EVTFS,-(SP)
MOV          #2,-(SP)
MOV          SP,R0
TRAP        C$PNTS
ADD         #6,SP

MOV          (R2)+,DEV1      ;GET PRINT CODE AND LINE NUMBER
MOV          (R2)+,DEV2      ;COPY SAVED DEVICE REGISTER CONTENTS
MOV          (R2)+,DEV3      ;COPY 2ND SAVED REGISTER
MOV          R2,-(SP)        ;SAVE R2 ON STACK
CLR          R3              ;CLEAR INDEX REGISTER
MOVB        DEV1,R3         ;GET PRINT CODE
ASL         R3              ;SHIFT CODE LEFT TO USE AS WORD INDEX
MOV          PRTEB(R3),R3    ;GET PRINT ROUTINE OFFSET FROM TABLE BASE
ADD         #PRTEB,R3       ;ADD IN ADDRESS OF TABLE BASE
JSR         PC,(R3)         ;PRINT REGISTERS
MOV          (SP)+,R2        ;RESTORE R2
JMP         RPT            ;GO BACK FOR NEXT EVENT ENTRY

PRTEB:      ;TABLE OF OFFSETS TO REGISTER PRINT ROUTINES
.WORD      #PRTE0-PRTEB
.WORD      #PRTE1-PRTEB
.WORD      #PRTE2-PRTEB
.WORD      #PRTE3-PRTEB
.WORD      #PRTE4-PRTEB
.WORD      #PRTE5-PRTEB
.WORD      #PRTE6-PRTEB

PRTE0:      ;PRINT 2 WORDS OF DEVICE REGISTER CONTENTS
PRINTS     #EFT3C,DEV2,DEV3 ;PRINT 2 WORDS
MOV          DEV3,-(SP)
MOV          DEV2,-(SP)
MOV          #EFT3C,-(SP)
MOV          #3,-(SP)
MOV          SP,R0
TRAP        C$PNTS
ADD         #10,SP

RTS         PC              ;RETURN

PRTE1:      ;** NOT USED
RTS         PC              ;RETURN

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 114
REPLUG PRINT EVENT LOG

3949	026036		
3950	026036		
3951	026036	005046	
3952	026040	153716	011472
3953	026044	005046	
3954	026046	153716	011471
3955	026052	012746	020472
3956	026056	012746	000003
3957	026062	010600	
3958	026064	104416	
3959	026066	062706	000010
3960	026072	000207	
3961			
3962			
3963	026074		
3964	026074		
3965	026074	005046	
3966	026076	153716	011473
3967	026102	005046	
3968	026104	153716	011472
3969	026110	005046	
3970	026112	153716	011471
3971	026116	012746	020507
3972	026122	012746	000004
3973	026126	010600	
3974	026130	104416	
3975	026132	062706	000012
3976	026136	000207	
3977			
3978			
3979	026140		
3980	026140		
3981	026140	005046	
3982	026142	153716	011471
3983	026146	012746	020446
3984	026152	012746	000002
3985	026156	010600	
3986	026160	104416	
3987	026162	062706	000006
3988	026166	000207	

PRTE2: :PRINT 2 BYTES
PRINTS #EVT42,<B,DEV1+1>,<B,DEV2>

CLR	-(SP)
BISB	DEV2,(SP)
CLR	-(SP)
BISB	DEV1+1,(SP)
MOV	#EVT42,-(SP)
MOV	#3,-(SP)
MOV	SP,R0
TRAP	C\$PNTS
ADD	#10,SP

RTS PC ;RETURN

PRTE3: :PRINT 3 BYTES
PRINTS #EVT43,<B,DEV1+1>,<B,DEV2>,<B,DEV2+1>

CLR	-(SP)
BISB	DEV2+1,(SP)
CLR	-(SP)
BISB	DEV2,(SP)
CLR	-(SP)
BISB	DEV1+1,(SP)
MOV	#EVT43,-(SP)
MOV	#4,-(SP)
MOV	SP,R0
TRAP	C\$PNTS
ADD	#12,SP

RTS PC ;RETURN

PRTE4: :PRINT LINE NUMBER
PRINTS #EVT40,<B,DEV1+1>

CLR	-(SP)
BISB	DEV1+1,(SP)
MOV	#EVT40,-(SP)
MOV	#2,-(SP)
MOV	SP,R0
TRAP	C\$PNTS
ADD	#6,SP

RTS PC

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 115
REPLOG PRINT EVENT LOG

3989	026170		
3990	026170		
3991	026170	013746	011472
3992	026174	005046	
3993	026176	153716	011471
3994	026202	012746	020532
3995	026206	012746	000003
3996	026212	010600	
3997	026214	104416	
3998	026216	062706	000010
3999	026222	000207	
4000			
4001	026224		
4002			
4003	026224		
4004	026224	005046	
4005	026226	153716	011473
4006	026232	005046	
4007	026234	153716	011472
4008	026240	012746	021207
4009	026244	012746	000003
4010	026250	010600	
4011	026252	104416	
4012	026254	062706	000010
4013	026260		
4014	026260	005046	
4015	026262	153716	011475
4016	026266	005046	
4017	026270	153716	011474
4018	026274	012746	021264
4019	026300	012746	000003
4020	026304	010600	
4021	026306	104416	
4022	026310	062706	000010
4023	026314	000207	
4024			

PRTE5: :PRINT LINE NUMBER AND MESSAGE POINTED TO BY DEV2
PRINTS #EVT44,<B,DEV1+1>,DEV2

```

MOV  DEV2,-(SP)
CLR  -(SP)
BISB DEV1+1,(SP)
MOV  #EVT44,-(SP)
MOV  #3,-(SP)
MOV  SP,R0
TRAP C$PNTS
ADD  #10,SP

```

RTS PC ;RETURN

PRTE6: :PRINT 4 STATUS BYTES
PRINTS #EVTFS2,<B,DEV2>,<B,DEV2+1>

```

CLR  -(SP)
BISB DEV2+1,(SP)
CLR  -(SP)
BISB DEV2,(SP)
MOV  #EVTFS2,-(SP)
MOV  #3,-(SP)
MOV  SP,R0
TRAP C$PNTS
ADD  #10,SP

```

PRINTS #EVTFS3,<B,DEV3>,<B,DEV3+1>

```

CLR  -(SP)
BISB DEV3+1,(SP)
CLR  -(SP)
BISB DEV3,(SP)
MOV  #EVTFS3,-(SP)
MOV  #3,-(SP)
MOV  SP,R0
TRAP C$PNTS
ADD  #10,SP

```

RTS PC

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 116
REPLUG PRINT EVENT LOG

4025 026316 005037 011470
4026 026322 005037 011472
4027 026326 112237 011470
4028 026332 112237 011472
4029 026336 012237 011474
4030 026342 012237 011476
4031 026346 010246
4032 026350 004737 031204
4033 026354 012602
4034 026356 012237 010176
4035 026362 000137 025442
4036
4037
4038
4039 026366
4040 026366 012237 011432
4041 026372 012237 011434
4042 026376 012237 011436
4043 026402 012237 010176
4044
4045
4046
4047 026406
4048 026406 013746 011436
4049 026412 013746 011434
4050 026416 013746 011432
4051 026422 012746 020600
4052 026426 012746 000004
4053 026432 010600
4054 026434 104416
4055 026436 062706 000012
4056
4057 026442 000137 025442
4058
4059
4060 026446 012237 011432
4061 026452 012237 011434
4062 026456 012237 011436
4063 026462 004737 026616
4064 026466
4065 026466 013746 011436
4066 026472 013746 011434
4067 026476 013746 011432
4068 026502 012746 020657
4069 026506 012746 000004
4070 026512 010600
4071 026514 104416
4072 026516 062706 000012
4073 026522 000137 025442

RPTDVI: CLR DEV1
CLR DEV2 ;CLEAR UPPER BYTES OF DEV1 & DEV2 BEFORE USE
MOVB (R2)+,DEV1 ;STORE SETUP OPERATION PARAMETERS FOR PRINTING
MOVB (R2)+,DEV2
MOV (R2)+,DEV3
MOV (R2)+,DEV4
MOV R2,-(SP) ;SAVE R2 ON THE STACK
JSR PC,SHWOP ;GO PRINT MODE, MAINT-LOOP TYPE, PARAMTERS.
MOV (SP)+,R2 ;RESTORE R2
MOV (R2)+,TEMP5 ;DUMMY MOVE
JMP RPT ;GO BACK FOR NEXT EVENT ENTRY

;REPORT END OF PASS R ^C ABORT

RPTABO: RPTTEOP: MOV (R2)+,EVTADD
MOV (R2)+,EVTBCT
MOV (R2)+,EVTTMP
MOV (R2)+,TEMP5 ;DUMMY MOVE

;PRINT PASCOUNT ERROR COUNT RX THRES AND TX TTHRES

PRINTS #EVT4B,EVTADD,EVTBCT,EVTTMP

MOV EVTTMP,-(SP)
MOV EVTBCT,-(SP)
MOV EVTADD,-(SP)
MOV #EVT4B,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #12,SP

JMP RPT ;THEN GO GET NEXT EVENT ENTRY

RPTDDE: MOV (R2)+,EVTADD ;STORE MESSAGE ADDRESS FOR PRINTING
MOV (R2)+,EVTBCT ;STORE BYTE COUNT FOR PRINTING
MOV (R2)+,EVTTMP ;STORE TOTAL # OF CMP ERRORS
JSR PC,PNTTRB ;PRINT LINE NO.

PRINTS #EVT4,EVTADD,EVTBCT,EVTTMP ;PRINT ADDR, BYTE CNT, # CMP ERRS

MOV EVTTMP,-(SP)
MOV EVTBCT,-(SP)
MOV EVTADD,-(SP)
MOV #EVT4,-(SP)
MOV #4,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #12,SP

JMP RPT ;THEN GO GET NEXT EVENT ENTRY

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 117
REPLOG PRINT EVENT LOG

4074	026526		
4075	026526	012237	011432
4076	026532	012237	011434
4077	026536	012237	011436
4078	026542	004737	026616
4079	026546		
4080	026546	013746	011436
4081	026552	013746	011434
4082	026556	013746	011432
4083	026562	012746	020761
4084	026566	012746	000004
4085	026572	010600	
4086	026574	104416	
4087	026576	062706	000012
4088			
4089	026602	000137	025442
4090			
4091			
4092			
4093	026606		
4094	026606	012604	
4095	026610	012603	
4096	026612	012602	
4097	026614	000207	
4098			
4099			
4100			
4101	026616	012237	010176
4102	026622		
4103	026622	013746	010176
4104	026626	012746	020546
4105	026632	012746	000002
4106	026636	010600	
4107	026640	104416	
4108	026642	062706	000006
4109	026646	000207	

```

RPTDLE:
RPTDCK: MOV (R2)+,EVTADD ;STORE MSG ADDR FOR PRINT
          MOV (R2)+,EVTBCT ;STORE BYTE COUNT
          MOV (R2)+,EVTTMP ;STORE BYTE COUNT COMP
          JSR PC,PNTTRB ;PRINT LINE NO.
          PRINTS #EVT4A,EVTADD,EVTBCT,EVTTMP ;PRINT ADDR,RXBYTES,CMPBYTES.
          MOV EVTTMP,-(SP)
          MOV EVTBCT,-(SP)
          MOV EVTADD,-(SP)
          MOV #EVT4A,-(SP)
          MOV #4,-(SP)
          MOV SP,R0
          TRAP C$PNTS
          ADD #12,SP

          JMP RPT ;THEN GO GET NEXT EVENT ENTRY

ENDEVT: ;RETURN TO CALLER AFTER REG RESTORE
          MOV (SP)+,R4 ;RESTORE R4,R3,R2
          MOV (SP)+,R3
          MOV (SP)+,R2
          RTS PC ;RETURN TO CALLING ROUTINE

          ;PRINT LINE NO.

PNTTRB: MOV (R2)+,TEMP5
          PRINTS #EVT6,TEMP5 ;PRINT LINE NUMBER.
          MOV TEMP5,-(SP)
          MOV #EVT6,-(SP)
          MOV #2,-(SP)
          MOV SP,R0
          TRAP C$PNTS
          ADD #6,SP

          RTS PC ;RETURN TO EVENT

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 118
CLI FOR REPORT CODING SECTION

4110						
4111	026650			.SBTTL	CLI FOR REPORT CODING SECTION	
4112	026650	006302		CLIRAC:		
4113	026652	016202	026666	ASL	R2	
4114	026656	062702	026666	MOV	10\$(R2),R2	;FORM ADDRESS OF ACTION ROUTINE
4115	026662	004712		ADD	#10\$,R2	
4116	026664	000207		JSR	PC,(R2)	
4117				RTS	PC	
4118	026666	000030		10\$:	.WORD	ACTRNL-10\$
4119	026670	000032			.WORD	ACTRHL-10\$;RPHLP
4120	026672	000076			.WORD	ACTREX-10\$;RPEXT
4121	026674	000106			.WORD	ACTRLG-10\$;RPLOG
4122	026676	000122			.WORD	ACTRSL-10\$;RPSOL
4123	026700	000022			.WORD	ACTRNF-10\$;RNOTNF
4124	026702	000136			.WORD	ACTRDS-10\$;RDMP5
4125	026704	000160			.WORD	ACTRDQ-10\$;RDMPQ
4126	026706	000166			.WORD	ACTRDE-10\$;RDMPE
4127						
4128						

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 119
REPORT COMMAND ACTION ROUTINES

```

4129
4130 026710 112737 177777 003400 .SBTTL REPORT COMMAND ACTION ROUTINES
4131 026716 000207 ACTRNF: MOVB #-1,P$NNUF ;SET FLAG TO SAY MORE NEEDED
4132 026720 012702 003276 ACTRNL: RTS PC
4133 026724 012702 003276 ACTRHL: MOV #RHLPTB,R2 ;SETUP R2 AS A POINTER TO HELP MSG TABLE
4134 026724 012246 1$ :PRINT HELP INFORMATION MESSAGES
4135 026726 012746 014616 MOV (R2)+,-(SP)
4136 026732 012746 000002 MOV #HLPF,-(SP)
4137 026736 010600 MOV #2,-(SP)
4138 026740 104417 MOV SP,R0
4139 026742 062706 000006 TRAP C$PNTF
4140 026746 020227 003304 ADD #6,SP
4141 026752 001364 CMP R2,#RHLPEN ;SEE IF ALL INFO PRINTED YET
4142 026754 012737 000001 003246 BNE 1$ ;IF NO KEEP PRINTING
4143 026762 000207 MOV #RPHLP,KEYWD1
4144 ACTREX: MOV #RPEXT,KEYWD1 ;SET UP EXIT WORD
4145 026764 012737 000002 003246 RTS PC
4146 026772 000207 ACTRLG: JSR PC,REPLOG ;GO REPORT DCLT EVENT LOG
4147 026774 004737 025374 ACTRLG: MOV #RPLOG,KEYWD1
4148 027000 012737 000003 003246 RTS PC
4149 027006 000207 ACTRSL: MOV #RPSOL,KEYWD1 ;SET UP KEY WORD
4150 027010 012737 000004 003246 MOV P$NNUF ;CLEAR NOT ENOUGH FLAG
4151 027010 105037 003400 CLR B ;AND RETURN
4152 027016 105037 003400 RTS PC
4153 027022 000207 ACTRDS: MOV P$NUM,STADD ;SETUP STARTING ADDRESS FOR DUMP
4154 027024 013737 003374 010126 CLR BYTBIT ;SET DEFAULT OF WORD DUMP
4155 027032 005037 010132 003246 MOV #RDMPS,KEYWD1 ;FLAG THAT A DUMP WAS TYPED
4156 027036 012737 000006 003246 BR ACTRDE
4157 027044 000403 ACTRDQ: MOV #-1,BYTBIT ;SET DUMP FLAG TO "DUMP-BYTE"
4158 027046 012737 177777 010132 ACTRDE: MOV P$NUM,ENADD ;SETUP END ADDRESS FOR DUMP (=START IF NO "EEE"
4159 027054 013737 003374 010130 ACTRDX: MOV P$NNUF ;CLEAR NOT-ENOUGH FLAG, "DUMP N-N/B" IS VALID
4160 027062 105037 003400 RTS PC
4161 027066 000207
4162
4163
4164
4165
4166
4167
4168
    
```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 120
REPORT CODE COMMAND LINE PARSING TREE

```

4169          .SBTTL          REPORT CODE COMMAND LINE PARSING TREE
4170
4171 027070    CLIRT:  CLI      CLISPA,0,R10$  :SKIP ANY SPACES
4172 027074    R10$:   CLI      <'?'>,RPHLP,R11$ :IS FIRST NON-SP CHAR A "'?'"
4173 027100    CLIRT:  CLI      CLIEXI,0      :EXIT
4174 027102    R11$:   CLI      CLISTR,RPHLP,R12$,<'HELP'>
4175 027116    CLIRT:  CLI      CLIEXI,0
4176 027120    R12$:   CLI      CLISTR,RPEXT,R13$,<'EXIT'>
4177 027134    CLIRT:  CLI      CLIEXI,0
4178 027136    R13$:   CLI      CLISTR,RPSOL,R14$,<'STATUSONLINE'>
4179 027162    CLIRT:  CLI      CLIEXI,0
4180 027164    R14$:   CLI      CLISTR,RPLOG,R46$,<'LOG'>
4181 027176    CLIRT:  CLI      CLIEXI,0
4182
4183          :DUMP ADDED TO REPORT
4184 027200    R46$:   CLI      CLISTR,RNOTNF,R30$,<'DUMP'>  :ELSE, IS FIRST WORD A 'DUMP'
4185 027214    CLIRT:  CLI      CLIBR,0,R50$                : IF YES GOTO R50$
4186
4187          :GET ADDRESSES FOR DUMP COMMAND
4188
4189 027220    R50$:   CLI      CLIALP,0,R51$
4190 027224    R51$:   CLI      CLISPA,0,R52$
4191 027230    R52$:   CLI      CLIOCT,RDMP$ ,R30$
4192 027234    CLIRT:  CLI      <'-'>,RNOTNF,R125$
4193 027240    CLIRT:  CLI      CLIOCT,RDMP$,R30$
4194 027244    CLIRT:  CLI      <'/'>,RNOTNF,R125$
4195 027250    CLIRT:  CLI      <'B'>,RDMPQ,R30$
4196 027254    CLIRT:  CLI      CLIBR,0,R125$
4197
4198
4199
4200 027260    R30$:   CLI      CLIERR,0                      :OTHERWISE "ILL CMD" - EXIT
4201
4202          :END-OF-LINE
4203 027262    R125$:  CLI      CLIEXI,0
4204
4205

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 121
DUMP BYTES OR WORDS

.SBTTL DUMP BYTES OR WORDS

4206
4207
4208
4209
4210
4211
4212
4213
4214
4215
4216
4217
4218
4219
4220
4221
4222
4223
4224
4225
4226
4227
4228
4229
4230
4231
4232
4233 027264 013702 010126
4234 027270 005003
4235 027272
4236 027272 010246
4237 027274 012746 020051
4238 027300 012746 000002
4239 027304 010600
4240 027306 104417
4241 027310 062706 000006
4242 027314 005737 010132
4243 027320 001416
4244 027322 112237 010164
4245 027326
4246 027326 005046
4247 027330 153716 010164
4248 027334 012746 020033
4249 027340 012746 000002
4250 027344 010600
4251 027346 104417
4252 027350 062706 000006
4253 027354 000411
4254 027356
4255 027356 012246
4256 027360 012746 020042
4257 027364 012746 000002
4258 027370 010600
4259 027372 104417
4260 027374 062706 000006
4261 027400 020237 010130

```

:++
FUNCTIONAL DESCRIPTION:
DUMPSR - DUMP BYTES OR WORDS SUBROUTINE

THIS SUBROUTINE PRINTS THE CONTENTS OF THE LOCATIONS BETWEEN
A STARTING AND END ADDRESS IN LOCS. "STADD" AND "ENADD".
THE WORD OR BYTE CONTENTS ARE PRINTED 8 TO A LINE WITH THE
ADDRESS OF THE FIRST BYTE AS THE FIRST 6 OCTAL CHARS. FOLLOWED
BY A SEMICOLON.

INPUTS:
STADD= STARTING ADDRESS (FIRST LOC. TO PRINT)
ENADD= END ADDRESS (LAST LOCATION TO DUMP)
BYTBIT= 1 IF SUPPOSED TO PRINT "BYTES"
        0 IF SUPPOSED TO PRINT "WORDS"

OUTPUTS:
CONTENTS OF A RANGE OF LOC.S PRINTED ON THE OPERATORS CONSOLE.

CALLING SEQUENCE.
JSR PC,DUMPSR          ;CALL DUMP BYTES SUBROUTINE
--

```

```

DUMPSR: MOV     STADD,R2          ;SET R2 UP TO STARTING ADDR.
DUM4:   CLR     R3              ;CLEAR R3
        PRINTF #BASM1,R2       ;PRINT ADDRESS

        MOV     R2,-(SP)
        MOV     #BASM1,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP   C$PNTF
        ADD     #6,SP

DUM3:   TST     BYTBIT          ;IS THIS BYTE OR WORD
        BEQ     DUM1           ;BR IF WORD
        MOVB   (R2)+,TEMP      ;MOV BYTE TO TEMP
        PRINTF #BASM3,<B,TEMP> ;PRINT BYTE

        CLR     -(SP)
        BISB   TEMP,(SP)
        MOV     #BASM3,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP   C$PNTF
        ADD     #6,SP

DUM1:   BR      DUM2           ;PRINT WORD
        PRINTF #BASM2,(R2)+

        MOV     (R2)+,-(SP)
        MOV     #BASM2,-(SP)
        MOV     #2,-(SP)
        MOV     SP,R0
        TRAP   C$PNTF
        ADD     #6,SP

DUM2:   CMP     R2,ENADD       ;COMPARE FOR LAST ADD

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 122
DUMP BYTES OR WORDS

4262 027404 003005
4263 027406 005203
4264 027410 022703 000010
4265 027414 001725
4266 027416 000736
4267
4268 027420 000207
4269

BGT DUMEX
INC R3
CMP #8, R3
BEQ DUM4
BR DUM3

DUMEX: RTS PC

:IF DONE EXIT
:ELSE BUMP R3
:HAVE WE PRINTED 8 ACROSS
:IF SO GO BACK TO 4
:ELSE GO BACK AND PRINT ANOTHER
:BYTE OR WORD
:RETURN TO CALLER

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 123
UPDATE TOTAL CHAR. COUNT SUBROUTINE

4270
4271
4272
4273
4274
4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290
4291
4292
4293
4294
4295
4296
4297
4298
4299
4300
4301
4302
4303
4304
4305
4306
4307
4308
4309
4310
4311

```
.SBTTL          UPDATE TOTAL CHAR. COUNT SUBROUTINE

:++
: FUNCTIONAL DESCRIPTION:
:   UPDATES TOTAL CHAR. COUNT TOTCC BASED ON CURCC.
:   LAST MESSAGE IS TRUNCATED TO FIT INTO THE
:   BUFFER IF TOTAL CHAR. COUNT EXCEEDS 'BUFLIM' A MESSAGE
:   IS PRINTED TELLING THE OPERATOR THE TRUNCATION OCCURRED.
:
: INPUTS:
:   CURCC= CHAR. COUNT OF MESSAGE BEING ADDED
:   TOTCC= TOTAL CHAR COUNT OF BUFFER ITS BEING ADDED TO
:
: OUTPUTS:
:   MESSAGE TO OPERATOR IF MESSAGE TRUNCATED TO FIT
:
: FUNCTIONAL SIDE EFFECTS:
:   LOCATION "TEMP" USED FOR CALCULATIONS
:
: CALLING SEQUENCE:
:   JSR      PC,ADDCC          ;UPDATED TOTAL CHAR. COUNT
:--
```

```

027422 063737 010150 010160 ADDCC: ADD      CURCC,TOTCC      ;ADD CURRENT TO TOTAL
027430 022737 001000 010160      CMP      #BUFLIM,TOTCC    ; COMPARE TO 'BUFLIM'
027436 103027              BHIS     ADDC1           ;IF NOT MORE THEN 'BUFLIM' EXIT

; PRINT MESSAGE AND TRUNCATE COUNT
PRINTF #MSGTRU

027440 012746 017105              MOV      #MSGTRU,-(SP)
027444 012746 000001              MOV      #1,-(SP)
027450 010600              MOV      SP,R0
027452 104417              TRAP    C$PNTF
027454 062706 000004              ADD      #4,SP

027460 163737 010150 010160      SLB     CURCC,TOTCC      ;SUB CURRENT FROM TOTAL
027466 012737 001000 010164      MOV     #BUFLIM,TEMP    ;MOV 'BUFLIM' TO TEMP
027474 163737 010160 010164      SUB     TOTCC,TEMP      ;SUB TOTAL FROM 'BUFLIM'
027502 013737 010164 010150      MOV     TEMP,CURCC      ;AND ESTABLISH NEW CURRENT
027510 063737 010150 010160      ADD     CURCC,TOTCC     ;ADD "ADJUSTED CURRENT" TO TOTAL CHAR. CNT.
027516 000207              ADDC1: RTS     PC        ;RETURN TO CALLER
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 124
BUILD MESSAGE BUFFERS SUBROUTINE

```

4312 .SBTTL          BUILD MESSAGE BUFFERS SUBROUTINE
4313
4314
4315 :++
4316 :FUNCTIONAL DESCRIPTION:
4317 :BLDBUF-- BUILD POINTER TABLE AND BUFFERS
4318
4319 :THIS SUBROUTINE ADDS A MESSAGE TO THE TRANSMIT OR EXPECT LIST
4320 :USING THE POINTER, BYTE COUNT, AND ADDRESS PASSED TO IT.
4321
4322 :INPUTS:
4323 :CURCC= CHAR. COUNT OF MESSAGE TO BE ADDED
4324 :CURADD= ADDRESS OF MESSAGE TO BE ADDED
4325 :CPTR= ADDRESS OF POINTER TABLE WORD WHERE MESSAGE POINTERS ARE
4326 :       TO BE BUILT
4327 :MSGTYP= VALUE TO USE AS AN INDEX TO FIND SOURCE OF MESSAGE DATA
4328 :         INDEX INTO DMSGCT() AND DMSGAD().
4329
4330 :OUTPUTS:
4331 :A MESSAGE ADDED TO EITHER TXBUF OR CMPBUF
4332 :APPROPRIATE POINTERS IN PTRTAB POINTER TABLE
4333
4334 :CALLING SEQUENCE:
4335 :JSR PC,BLDBUF          :BUILD MESSAGE IN BUFFER AND ADD PTRS.
4336
4337 :BLDBUF:
4338 :MOV     R2,-(SP)      :SAVE R2 AND R3 ON THE STACK
4339 :MOV     R3,-(SP)
4340 :MOV     CPTR,R2
4341
4342 :BLDB1: MOV     CURADD,(R2)+ :PUT CURRENT ADD ON POINTER TAB
4343 :MOV     CURCC,(R2)+   :PUT CURRENT CC ON POINTER TAB
4344 :MOV     R2,CPTR      :PUT UPDATED R2 BACK TO CURRENT POINT
4345 :MOV     MSGTYP,R2    :GET MESSAGE TYPE TO USE AS INDEX
4346 :ASL     R2           :DOUBLE FOR WORD INDEX
4347 :MOV     CURADD,TEMP   :MOVE CURRENT ADD TO TEMP
4348 :ADD     CURCC,TEMP    :ADD CHAR COUNT TO IT TO GET END
4349 :MOV     CURADD,R3    :SET R3 TO CURRENT START ADD
4350 :BLDB2: MOV     DMSGCT(R2),TEMP2 :GET BYTE COUNT
4351 :MOV     DMSGAD(R2),R4 :PUT STARTING FROM ADD IN R4
4352 :ADD     R4,TEMP2     :ADD IT TO TEMP2 TO GET END OF FROM
4353 :MOVB    (R4)+,(R3)+  :MOV BYTE FROM PATTERN TO BUFFER
4354 :CMP     R3,TEMP      :ALL DONE?
4355 :BEQ     BLDBEX       :IF SO EXIT
4356 :CMP     R4,TEMP2     :IS PATTERN COUNT EXPIRED
4357 :BEQ     BLDB2        :IF SO GO START AGAIN
4358 :BR      BLDB3        :IF NOT GET ANOTHER BYTE
4359 :BLDBEX: ADD     CURCC,CURADD   :BUMP CURADD
4360 :MOV     (SP)+,R3     :RESTORE R3 AND R2
4361 :MOV     (SP)+,R2
4362 :RTS     PC           :RETURN TO CALLER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 125
FCURAD FORCE CURRENT ADDRESS TO WORD BOUNDARY

4363
4364
4365
4366
4367
4368
4369
4370
4371
4372
4373
4374
4375
4376
4377
4378
4379
4380
4381
4382
4383

.SBTTL FCURAD FORCE CURRENT ADDRESS TO WORD BOUNDARY

++
:FUNCTIONAL DESCRIPTION
:FORCE CURRENT ADDRESS: THIS ROUTINE IS USED TO ENSURE
:THAT CURADD POINTS TO A WORD BOUNDARY
:IF CURADD DOES NOT POINT TO A WORD BOUNDARY
:THEN A BYTE IS CLEARED AT CURADD
:CURADD IS INCREMENTED
:AND TOTCC IS INCREMENTED
--

027644
027644 032737 000001 010156
027652 001406
027654 105077 160276
027660 005237 010160
027664 005237 010156
027670 000207

FCURAD: ;CHECK CURRENT ADDRESS
BIT #1,CURADD ;IS CURADD ODD ?
BEQ 10\$;NO. GO EXIT
CLRB @CURADD ;YES. INSERT A FILL BYTE
INC TOTCC ;INCREMENT CHARACTER COUNT
INC CURADD ; AND ADDRESS TO FORCE TO A WORD BOUNDARY
10\$: RTS PC

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 126
CREATE FACSIMILE OF TX BUFFER AND MESSAGE LIST

4384
4385
4386
4387
4388
4389
4390
4391
4392
4393
4394
4395
4396
4397
4398
4399
4400
4401
4402
4403
4404
4405
4406
4407
4408
4409
4410
4411
4412
4413
4414
4415
4416
4417
4418
4419
4420
4421
4422
4423
4424
4425
4426
4427
4428
4429
4430
4431
4432
4433
4434
4435
4436
4437
4438
4439

.SBTTL CREATE FACSIMILE OF TX BUFFER AND MESSAGE LIST

::*

:FUNCTIONAL DESCRIPTION:

FACSIMILE: THIS ROUTINE IS USED TO CREATE A FACSIMILE OF THE
OF THE TRANSMIT LIST AND TRANSMIT BUFFER IN THE
EXPECT LIST AND EXPECT BUFFER. THE ROUTINE IS
NORMALLY CALLED WHEN USER COMMAND "SET E [XPECT]=
T [TRANSMIT] IS ENTERED.

CALLING SEQUENCE: JSR PC,FACSIMILE

DEFINITIONS CMPBUF = EXPECTED DATA BUFFER HOLDS MAX 512 BYTES
TXBUF = TRANSMIT DATA BUFFER HOLDS MAX 512 BYTES
ITOTCC = NUMBER OF BYTES IN TXBUF
PTRTAB = TOP OF MESSAGE LIST POINTER TABLE
CTOTCC = NUMBER OF BYTES IN EXPECT MESSAGE
CMPTOT = NUMBER OF EXPECTED MESSAGES
CMPPTR = EXPECTED MESSAGE LIST POINTER
TXPTR = TRANSMIT MESSAGE LIST POINTER
TXMTOT = NUMBER OF TRANSMIT MESSAGES
CCURAD = STORAGE ADDRESS OF MESSAGE IN CMPBUF
MSGLIM = MAXIMUM NUMBER OF MESSAGES THAT CAN BE STORED
BUFLIM = NUMBER OF BYTES IN BUFFER

BEGIN FACSIMILE ROUTINE

(*COPY TXBUF ==> CMPBUF*)

..SAVE R1

..INIT R1

..REPEAT

....[CMPBUF]R1=[TXBUF]R1

....R1=R1+1

..UNTIL R1 = BUFLIM

(*NOW CALCULATE EXPECT LIST MESSAGE POINTER*)

..CMPPTR = PTRTAB + (2 * MSGLIM)

(*NOW PRIME THE WHILE - DO LOOP*)

..TXPTR = PTRTAB

..CCURAD = CMPBUF

..TXPTR = TXPTR + 2

..CTOTCC = [TXPTR]

..CMPTOT = 0

..WHILE TXMTOT <> CMPTOT DO

....[CMPPTR] = CCURAD

....CMPPTR = CMPPTR + 2

....[CMPPTR] = CTOTCC

....TXPTR = TXPTR + 4

....CCURAD = CCURAD + CTOTCC

....CTOTCC = [TXPTR]

....CMPPTR = CMPPTR + 2

....CMPTOT = CMPTOT + 1

..END WHILE DO

..CTOTCC = ITOTCC

END FACSIMILE ROUTINE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 127
CREATE FACSIMILE OF TX BUFFER AND MESSAGE LIST

```

4440
4441 027672
4442 027672 010146
4443 027674 005001
4444 027676 116161 003406 004406 10$:
4445 027704 005201
4446 027706 020127 001000
4447 027712 001371
4448
4449 027714 012701 000017 20$:
4450 027720 006301
4451 027722 006301
4452 027724 012737 006406 010054
4453 027732 060137 010054
4454 027736 005001
4455
4456
4457 027740 012737 006406 010052
4458 027746 012737 004406 010062
4459 027754 062737 000002 010052
4460 027762 017737 160064 010060
4461 027770 005037 010056
4462
4463
4464 027774 023737 010074 010056 30$:
4465 030002 001436
4466 030004 013777 010062 160042
4467 030012 062737 000002 010054
4468 030020 013777 010060 160026
4469 030026 062737 000004 010052
4470 030034 063737 010060 010062
4471
4472 030042 032737 000001 010062
4473 030050 001402
4474 030052 005237 010062
4475 030056
4476 030056 017737 157770 010060 35$:
4477 030064 062737 000002 010054
4478 030072 005237 010056
4479 030076 000736
4480
4481 030100 013737 010076 010060 40$:
4482
4483
4484 030106 012601
4485 030110 000207
4486
4487

FACSIMILE:
MOV R1,-(SP) ;SAVE R1
CLR R1 ;INIT R1
MOV TXBUF(R1),CMPBUF(R1) ;COPY TX BUFFER TO EXPECTED BUFFER
INC R1 ;BUMP INDEX
CMP R1,#BUFLIM ;ALL DATA COPIED ?
BNE 10$ ;NO,BRANCH

20$:
MOV #MSGLIM,R1 ;MESSAGE LIMIT
ASL R1 ;MULTIPLY BY 2
ASL R1 ;MULTIPLY BY 2
MOV #PTRTAB,CMPPTR ;TOP OF POINTER TABLE
ADD R1,CMPPTR ;START OF EXPECTED POINTER TABLE
CLR R1 ;INIT R1

;SET UP WHILE - DO LOOP
MOV #PTRTAB, TXPTR ;TX POINTER NOW AT TOP OF TABLE
MOV #CMPBUF, CCURAD ;TRANSFER ADDRESS OF 1ST MESSAGE
ADD #2, TXPTR ;BUMP POINTER
MOV @TXPTR, CTOTCC ;BYTE COUNTER 1ST MESSAGE
CLR CMPTOT ;INIT EXPECTED MESSAGE COUNT

30$:
;WHILE TX MESSAGE TOTAL <> EXPECTED MESSAGE TOTAL DO
CMP TXMTOT, CMPTOT ;ALL MESSAGES COPIED ?
BEQ 40$ ;YES,BRANCH
MOV CCURAD, @CMPPTR ;TRANSFER ADDRESS OF MESSAGE
ADD #2, CMPPTR ;BUMP POINTER
MOV CTOTCC, @CMPPTR ;BYTE COUNT OF MESSAGE
ADD #4, TXPTR ;BUMP TX MESSAGE POINTER
ADD CTOTCC, CCURAD ;CALC. TRANSFER ADDRESS
;ADJUST ODD ADDRESS
BIT #1, CCURAD ;IS COMPUTED ADDRESS ODD ?
BEQ 35$ ;NO. OMIT ADJUSTMENT
INC CCURAD ;YES. INCREMENT ADDRESS

35$:
MOV @TXPTR, CTOTCC ;BYTE COUNT NEXT MESSAGE
ADD #2, CMPPTR ;BUMP POINTER
INC CMPTOT ;INCREMENT MESSAGE COUNT
BR 30$ ;DO IT AGAIN

40$:
;END WHILE - DO
MOV TTOTCC, CTOTCC ;COPY TOTAL CHARACTER COUNT

;END ROUTINE
MOV (SP)+, R1 ;RESTORE R1
RTS PC ;RETURN
    
```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 128
QUEUE UP ALL REC BUFFERS FOR MULTIPOINT

.SBTTL QUEUE UP ALL REC BUFFERS FOR MULTIPOINT

```

:++
: FUNCTIONAL DESCRIPTION:      RXQUAL - QUEUE ALL REC BUFFERS
:
: THIS ROUTINE QUEUES ALL REC BUFFERS FOR VALID LINES
:
: SUBORDINATE ROUTINES USED:
:   GTVIND - LOADS INDEX WITH OFFSET TO NEXT
:             VALID LINE AND LOADS TRIBN WITH
:             ADDRESS OF NEXT VALID LINE
:   ULRPLS - MOVES RXPTR FOR THIS LINE TO
:             CPTRR FROM CPTRLS.
:   LOGAQR - QUES REC BUFFER POINTED TO BY
:             CPTRR AND LOGS THIS IN EVENT LOG
:   LDRPLS - MOVES VALUE OF CPTRR TO SLOT IN
:             CPTRLS FOR THIS LINE
:
: CALLING SEQUENCE:
:   JSR      PC,RXQUAL
:--

```

4488
4489
4490
4491
4492
4493
4494
4495
4496
4497
4498
4499
4500
4501
4502
4503
4504
4505
4506
4507
4508
4509
4510
4511
4512
4513
4514
4515
4516
4517
4518
4519
4520
4521

```

030112 012737 177777 007642 RXQUAL: MOV      #-1,INDEX      ;SET INDEX TO -1
030120 004737 030402          RXQU1: JSR      PC,GTVIND    ;GET NEXT VALID INDEX
030124 022737 000010 007642          CMP      #8.,INDEX    ;IS ALL DONE
030132 001411          BEQ      RXQUEX      ;IF SO EXIT
030134 004737 030454          JSR      PC,ULRPLS    ;LOAD CPTRR FOR THIS LINE
030140 052737 000004 010230          BIS      #QRX,FLAG    ;INDICATE RX QUEUED
030146 004737 031142          JSR      PC,LOGAQR
030152 004737 030434          JSR      PC,LDRPLS
030156 000207          RXQUEX: RTS      PC      ;RELOAD RX PTR LIST
                                ;RETURN TO CALLER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 129
LOAD CPTRLS LIST INITIALLY

4522
4523
4524
4525
4526
4527
4528
4529
4530
4531
4532
4533
4534
4535
4536
4537
4538
4539
4540
4541
4542
4543
4544
4545
4546
4547
4548
4549
4550
4551
4552
4553
4554
4555
4556

```

.SBTTL          LOAD CPTRLS LIST INITIALLY
:++
: FUNCTIONAL DESCRIPTION:          LCPRLS -LOAD CPTR LIST INITIALLY
:
: THIS ROUTINE LOADS UP THE CPTRLS LIST FOR ALL
: VALID LINE NUMBERS IN THE TRIBLS IT ALSO LOADS
: THE DVRCLS LIST FOR MSG COUNTS.
:
: INPUTS:          RXMTOT - TOTAL NUMBER OF RX MSGS PER LINE
:
: OUTPUTS:         CPTRLS - LOADED WITH POINTERS TO THE RXPTR TABLE
:                   FOR EACH LINE
:                   DVRCLS - LOADED WITH RXTOT COUNT FOR EACH LINE
:
: SUBORDINATE ROUTINES USED:
:                   GTVIND - GETS NEXT VALID INDEX BY
:                           CHECKING TRIBLS FOR NON ZERO ENTRY
:                   LCPRL1 - LOADS POINTER TABLE FOR LINE AT THIS
:                           INDEX VALUE AND RXMTOT TO DVRCLS FOR
:                           THIS LINE.
:
: CALLING SEQUENCE:
:                   JSR          PC,LCPRLS
:--
LCPRLS: MOV        #-1,INDEX          ;SET UP INDEX VALUE TO -1
LCPR1:  JSR        PC,GTVIND          ;GET VALID INDEX
:                   CMP        #8.,INDEX ;IS IT 8?
:                   BEQ        LCPREX   ;BRANCH IF 8.
:                   JSR        PC,LCPRL1 ;IF NOT LOAD CPTRLS FOR THIS LINE.
LCPREX: RTS        PC                ;RETURN TO CALLER WHEN DONE WITH ALL.

```

```

030160 012737 177777 007642
030166 004737 030402
030172 022737 000010 007642
030200 001402
030202 004737 030210
030206 000207

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 130
LOAD CPTCLS AND DVRCLS FROM INDEX

4557
4558
4559
4560
4561
4562
4563
4564
4565
4566
4567
4568
4569
4570
4571
4572
4573
4574
4575
4576
4577
4578
4579
4580
4581
4582
4583
4584
4585
4586
4587
4588
4589
4590

```
.SBTTL          LOAD CPTCLS AND DVRCLS FROM INDEX
:++
: FUNCTIONAL DESCRIPTION:          LCPRL1 - LOAD CPTCLS AND DVRCLS FROM INDEX
:
: THIS ROUTINE LOADS UP THE CPTCLS LIST FOR THE
: INDEX VALUE AND THE DVRCLS IS LOADED WITH RXMTOT.
:
: INPUTS:          RXMTOT - TOTAL NUMBER OF RX MSGS PER LINE
:                  PTR23  - START OF RX POINTER TABLE
:
: OUTPUTS:         CPTCLS - LOADED WITH POINTERS TO THE RXPTR LIST
:                  DVRCLS - LOADED WITH RXMTOT COUNT
:
: SUBORDINATE ROUTINES USED:
:                   MTPLY  - MULTIPLIES VALUE IN INDEX BY VALUE IN
:                           TEMP AND THEN ADDS THAT RESULT TO VALUE
:                           IN TEMP2 AND PUTS FINAL RESULT IN TEMP2
:
: CALLING SEQUENCE:
:                   JSR    PC,LCPRL1
:--
```

```
030210 012737 006576 010170 LCPRL1: MOV    #PTR23,TEMP2 ;SET UP TEMP 2 AS BASE
030216 013737 007642 010046      MOV    INDEX,MPLY  ;SET UP MULTIPLIER
030224 012737 000074 010164      MOV    #60.,TEMP   ;SET UP MULTIPLICAN
030232 004737 030356          JSR    PC,MTPLY    ;GO MULTIPY
030236 013703 007642          LCPRL2: MOV    INDEX,R3
030242 113763 010112 007602      MOVB  RXMTOT,DVRCLS(R3) ;LOAD UP COUNT LIST
030250 006303          ASL   R3           ;MAKE R3 WORD INDEX
030252 013763 010170 007542      MOV    TEMP2,CPTCLS(R3) ;SET UP POINTER TABLE
030260 000207          RTS    PC           ;RETURN TO CALLER
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 131
CLEAR RECEIVE POINTER LIST

4591
4592
4593
4594
4595
4596
4597
4598
4599
4600
4601
4602
4603
4604
4605
4606
4607
4608
4609
4610
4611

.SBTTL CLEAR RECEIVE POINTER LIST

..++
: FUNCTIONAL DESCRIPTION: CLRPLS - CLEAR RX POINTER LIST
: THIS ROUTINE CLEARS ALL 8 SLOTS OF THE CTPRLS
: OUTPUTS: CPTRLS - IS ZEROED IN ALL SLOTS
: CALLING SEQUENCE:
: JSR PC,CLRPLS
:--

030262	012737	000010	010164	CLRPLS: MOV	#8,TEMP	
030270	012703	007542		MOV	#CPTRLS,R3	:LOAD START OF LIST TO R3
030274	005023			CLRPL1: CLR	(R3)+	:CLEAR THIS SLOT
030276	005337	010164		DEC	TEMP	
030302	001374			BNE	CLRPL1	:IF NOT DONE GO BACK
030304	000207			CLRPEX: RTS	PC	:RETURN TO CALLER WHEN DONE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 132
LOAD TX POINTER LIST INITIALLY

4612
4613
4614
4615
4616
4617
4618
4619
4620
4621
4622
4623
4624
4625
4626
4627
4628
4629
4630
4631
4632
4633
4634
4635
4636
4637
4638
4639
4640
4641
4642
4643
4644
4645
4646
4647
4648
4649

.SBTTL LOAD TX POINTER LIST INITIALLY

```

++
FUNCTIONAL DESCRIPTION: LCPTLS - LOAD TRANSMIT POINTER LIST
THIS ROUTINE LOADS CPTLS WITH TX POINTERS
FOR EACH VALID LINE.

INPUTS:
TXMTOT - TOTAL NUMBER OF TX MSGS
PTRTAB - POINTER TO TOP OF TX POINTER TABLE

OUTPUTS:
CCTLS - LOADED WITH POINTERS TO TX POINTER TABLE
FOR ALL VALID LINES
DVTCLS - TX MSG COUNT LIST LOADED WITH MSG COUNTS
FOR ALL VALID LINES

SUBORDINATE ROUTINES USED:
GTVIND - GETS NEXT VALID INDEX BY
CHECKING TRIBLS FOR NON ZERO ENTRY

LDTPLS - LOADS VALUE FROM CPTR TO CPTLS INDEXED
BY TRIBN

LDTCLS - LOADS DVTCT TO DVTCLS INDEXED BY TRIBN

CALLING SEQUENCE:
JSR PC,LCPTLS
--

```

```

030306 013737 010074 010072 LCPTLS: MOV TXMTOT,DVTCT ;LOAD UP COUNT
030314 012737 006406 010154 MOV #PTRTAB,CPTR
030322 012737 177777 007642 MOV #-1,INDEX ;LOAD INDEX WITH -1
030330 004737 030402 LCPT1: JSR PC,GTVIND ;GET VALID INDEX
030334 022737 000010 007642 CMP #8,INDEX ;IS THIS THE END
030342 001404 BEQ LCPTEX ;EXIT IF SO
030344 004737 030572 JSR PC,LDTPLS ;LOAD TX POINTER LIST
030350 004737 030632 JSR PC,LDTCLS ;LOAD TX COUNT LIST
030354 000207 LCPTEX: RTS PC ;RETURN TO CALLER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 133
MULTIPY

4650
4651
4652
4653
4654
4655
4656
4657
4658
4659
4660
4661
4662
4663
4664
4665
4666
4667
4668
4669
4670
4671
4672
4673
4674

.SBTTL MULTIPY

..**
: FUNCTIONAL DESCRIPTION: MTPLY- MULTIPY
: THIS ROUTINE MULTIPLIES THE VALUE IN MPLY BY
: THE VALUE IN TEMP AND THEN ADDS IN THE VALUE OF TEMP2
: WITH THE RELSUT GOING TO TEMP2

: INPUTS: TEMP2 - INITIALLY VALUE
: TEMP - VALUE TO MULTIPLY BY
: MPLY - NUMBER OF TIMES TO MULITPLY

: OUTPUTS: TEMP2 - RESULT OF [MPLY * TEMP]+TEMP2

: CALLING SEQUENCE:
: JSR PC,MTPLY

:--

030356 005737 010046
030362 001406
030364 063737 010164 010170
030372 005337 010046
030376 000767
030400 000207

MTPLY: TST MPLY
BEQ MTPLEX :IF MULITPLIER IS ZERO QUIT
ADD TEMP,TEMP2 :ADD THE FACTOR TO BASE
DEC MPLY :COUNT DOWN THE MULTIPLIER
BR MTPLY :GO BACK FOR MORE
MTPLEX: RTS PC :RETURN TO CALLER

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 134
GET NEXT VALID INDEX

4675
4676
4677
4678
4679
4680
4681
4682
4683
4684
4685
4686
4687
4688
4689
4690
4691
4692
4693
4694
4695
4696
4697
4698
4699

.SBTTL GET NEXT VALID INDEX

..++
FUNCTIONAL DESCRIPTION: GTVIND - GET NEXT VALID INDEX
THIS LOADS INDEX WITH INDEX VALUE OF NEXT VALID LINE. THIS ALSO
LOADS TRIBN WITH THE NUMBER.
LINE BEING THE LOCATION IN THE TRIBLS THAT HAS A NON-ZERO
ENTRY.

INPUTS: INDEX - SET TO VALUE OF LAST INDEX
OUTPUTS: INDEX - SET TO VALUE OF THIS LINE
TRIBN - NUMBER OF THIS LINE
CALLING SEQUENCE:
JSR PC,GTVIND
..--

030402 013703 007642
030406 005203
030410 116337 007622 007636
030416 123727 007636 000377
030424 001770
030426 010337 007642
030432 000207

GTVIND: MOV INDEX,R3
GTVI1: INC R3
MOVB TRIBLS(R3),TRIBN ;LOAD TRIBN
CMPB TRIBN,#377 ;SEE IF EMPTY SLOT OR NOT
BEQ GTVI1 ;IF EMPTY GO GET ANOTHER
MOV R3,INDEX ;LOAD INDEX VALUE IF NOT ZERO
RTS PC ;RETURN TO CALLER WHEN DONE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 135
LOAD REC POINTER LIST

```

4700          .SBTTL          LOAD REC POINTER LIST
4701          ++
4702          :FUNCTIONAL DESCRIPTION:      LDRPLS - LOAD RX POINTER LIST FROM CPTRR
4703          :                          THIS ROUTINE MOVES DATA FROM CPTRR TO THE SLOT IN THE
4704          :                          CPTRLS INDEXED BY INDW.
4705          :INPUTS:                      INDW - WORD INDEX INTO LIST
4706          :OUTPUTS:                     CPTRLS - CORRECT SLOT LOADED WITH DATA FROM CPTRR
4707          :SUBORDINATE ROUTINES USED:
4708          :                          GETIND - GETS INDW FOR THIS TRIBN
4709          :CALLING SEQUENCE
4710          :                          JSR      PC,LDRPLS
4711          :--
4712
4713          030434 004737 031102          LDRPLS: JSR      PC,GETIND          :GET INDW FOR THIS TRIBN
4714          030440 013703 007640          MOV      INDW,R3          :MOVE WORD INDEX TO R3
4715          030444 013763 010152 007542  MOV      CPTRR,CPTRLS(R3) :LOAD CPTRLS LIST
4716          030452 000207          RTS      PC          :RETURN TO CALLER
4717
4718          .SBTTL          UNLOAD CTPRR LIST
4719          ++
4720          :FUNCTIONAL DESCRIPTION:      ULRPLS - UNLOAD RX POINTER LIST
4721          :                          THIS ROUTINE MOVES DATA FROM CPTRLS SLOT INDEXED
4722          :                          BY INDW TO CTPRR.
4723          :IMPLICIT INPUTS:
4724          :                          TRIBN - NUMBER OF CURRENT LINE
4725          :OUTPUTS:                       CTPRR - VALUE FROM CPTRLS
4726          :SUBORDINATE ROUTINES USED:
4727          :                          GETIND - GET INDW FOR THIS TRIBN
4728          :CALLING SEQUENCE:
4729          :                          JSR      PC,ULRPLS
4730          :--
4731
4732          030454 004737 031102          ULRPLS: JSR      PC,GETIND          :GET INDEX
4733          030460 013703 007640          MOV      INDW,R3          :MOVE WORD INDEX TO R3
4734          030464 016337 007542 010152  MOV      CPTRLS(R3),CTPRR :LOAD CTPRR FROM LIST INDEX
4735          030472 000207          RTS      PC          :RETURN TO CALLER

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 136
GET REC POINTER TO CPTR

4736
4737
4738
4739
4740
4741
4742
4743
4744
4745
4746
4747
4748
4749
4750
4751
4752
4753
4754
4755
4756
4757
4758

.SBTTL GET REC POINTER TO CPTR

```

:++
: FUNCTIONAL DESCRIPTION: GRPTCP - GET RX POINTER TO CPTR
: THIS ROUTINE GETS THE RX POINTER TO CPTR FOR USE IN BUILD
: BUFFER.
:
: INPUTS: INDEX - INDEX VALUE FOR LINE
:
: OUTPUTS: CPTR - LOADED WITH ADDRESS OF RX BUFFER FOR THIS LINE
: SUBORDINATE ROUTINES USED:
: MTPLY - MULTIPLIES INDEX BY TEMP AND ADDS TEMP2 TO RESULT
: CALLING SEQUENCE:
: JSR PC,GRPTCP
:--
    
```

```

4753 030474 013737 007642 010046 GRPTCP: MOV INDEX,MPLY ;SET UP MULPILIER
4754 030502 012737 000074 010164 MOV #60.,TEMP
4755 030510 013737 010050 010170 MOV RXPTR,TEMP2
4756 030516 004737 030356 JSR PC,MTPLY ;[INDEX VALUE X 60.] + RXPTR = POINTER ADDRESS
4757 030522 013737 010170 010154 MOV TEMP2,CPTR ;SET UP POINTER ADDR.
4758 030530 000207 RTS PC
    
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 137
LOAD DVRCT LIST

```

4759          .SBTTL          LOAD DVRCT LIST
4760          ++
4761          :FUNCTIONAL DESCRIPTION:          LDRCLS - LOAD RX COUNT LIST
4762          :          THIS ROUTINE LOADS THE VALUE FROM DVRCT TO
4763          :          THE SLOT IN DVRCLS INDEXED BY TRIBN
4764          :INPUTS:          TRIBN - NUMBER OF LINE IN USE
4765          :          DRVCT - COUNT VALUE TO GO TO LIST
4766          :OUTPUTS:          DVRCLS- VALUE OF DRVCT
4767          :SUBORDINATE ROUTINES USED:
4768          :          GETIND -          GET INDEX FROM TRIBLS
4769          :CALLING SEQUENCE:
4770          :          JSR          PC,LDRCLS
4771          :--
4772
4773          030532 004737 031102          LDRCLS: JSR          PC,GETIND          :GET INDEX
4774          030536 013703 007642          MOV          INDEX,R3          :LOAD R3 WITH BYTE INDEX
4775          030542 113763 010110 007602          MOVB         DVRCT,DVRCLS(R3)      :LOAD LIST WITH COUNT
4776          030550 000207          RTS          PC          :RETURN TO CALLER
4777
4778          .SBTTL          UNLOAD DVRCT LIST
4779          ++
4780          :FUNCTIONAL DESCRIPTION:          ULRCLS - UNLOAD RX COUNT LIST
4781          :          THIS ROUTINE UNLOADS THE VALUE TO DVRCT FROM
4782          :          THE SLOT IN DVRCLS INDEXED BY TRIBN
4783          :INPUTS:          TRIBN - NUMBER OF LINE IN USE
4784          :          DVRCLS- VALUE OF DRVCT
4785          :OUTPUTS:          DRVCT - COUNT VALUE FROM LIST
4786          :SUBORDINATE ROUTINES USED:
4787          :          GETIND -          GET INDEX FROM TRIBLS
4788          :CALLING SEQUENCE:
4789          :          JSR          PC,ULRCLS
4790          :--
4791
4792
4793          030552 004737 031102          ULRCLS: JSR          PC,GETIND          :GET INDEX
4794          030556 013703 007642          MOV          INDEX,R3          :MOVE INDEX TO R3
4795          030562 116337 007602 010110          MOVB         DVRCLS(R3),DVRCT      :UNLOAD LIST
4796          030570 000207          RTS          PC          :RETURN TO CALLER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 138
LOAD CPTR LIST (TRANSMIT POINTER)

4797
4798
4799
4800
4801
4802
4803
4804
4805
4806
4807
4808
4809
4810
4811
4812
4813
4814
4815
4816
4817
4818
4819
4820
4821
4822
4823
4824
4825
4826
4827
4828
4829
4830
4831
4832
4833
4834

.SBTTL LOAD CPTR LIST (TRANSMIT POINTER)

..**
: FUNCTIONAL DESCRIPTION: LDTPLS - LOAD TX POINTER LIST
: THIS ROUTINE LOADS THE VALUE FROM CPTR TO
: THE TX POINTER LIST INDEXED BY TRIBN INDEX.
: INPUTS: TRIBN - NUMBER OF LINE IN USE
: OUTPUTS: CPTTLS - SLOT LOADED WITH CPTR DATA
: SUBORDINATE ROUTINES USED:
: GETIND - GET INDEX VALUE FROM TRIBLS
: CALLING SEQUENCE:
: JSR PC,LDTPLS
:--

030572 004737 031102
030576 013703 007640
030602 013763 010154 007562
030610 000207

LDTPLS: JSR PC,GETIND ;GET INDEX
MOV INDW,R3 ;MOVE INDEX TO R3
MOV CPTR,CPTTLS(R3) ;LOAD LIST
RTS PC ;RETURN TO CALLER

.SBTTL UNLOAD CPTR LIST (TRANSMIT POINTER)

..**
: FUNCTIONAL DESCRIPTION: ULTPLS - UNLOAD TX POINTER LIST
: THIS ROUTINE MOVES DATA FROM TX POINTER LIST
: TO CPTR.
: INPUTS: TRIBN - NUMBER OF LINE IN USE
: OUTPUTS: CPTR - VALUE FROM THE TX POINTER LIST
: SUBORDINATE ROUTINES USED:
: GETIND - GET INDEX FROM TRIBLS
: CALLING SEQUENCE:
: JSR PC,ULTPLS
:--

030612 004737 031102
030616 013703 007640
030622 016337 007562 010154
030630 000207

ULTPLS: JSR PC,GETIND ;GET INDEX
MOV INDW,R3 ;MOVE WORD INDEX TO R3
MOV CPTTLS(R3),CPTR ;GET PTR FROM LIST
RTS PC ;RETURN TO CALLER

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 139
LOAD DVTCT LIST (TRANSMIT COUNT)

4835
4836
4837
4838
4839
4840
4841
4842
4843
4844
4845
4846
4847
4848
4849
4850
4851
4852
4853
4854
4855
4856
4857
4858
4859
4860
4861
4862
4863
4864
4865
4866
4867
4868
4869
4870
4871
4872
4873
4874

.SBTTL LOAD DVTCT LIST (TRANSMIT COUNT)

..
..**
..FUNCTIONAL DESCRIPTION: LDTCLS - LOAD TX COUNT LIST
..THIS ROUTINE LOADS A VALUE FROM DVTCT TO
..THE TX COUNT LIST (DVTCLS). INDEXED BY TRIBN.
..
..INPUTS: TRIBN - NUMBER OF LINE IN USE
..DVTCT - CURRENT TX COUNT FOR LINE
..OUTPUTS: DVTCLS - SLOT LOADED WITH DVTCT
..SUBORDINATE ROUTINES USED:
..GETIND - GET INDEX FROM TRIBLS
..CALLING SEQUENCE:
..JSR PC,LDTCLS
..--

030632 004737 031102 LDTCLS: JSR PC,GETIND ;GET INDEX
030636 013703 007642 MOV INDEX,R3 ;MOVE BYTE INDEX TO R3
030642 113763 010072 007612 MOVB DVTCT,DVTCLS(R3);LOAD LIST
030650 000207 RTS PC ;RETURN TO CALLER

.SBTTL UNLOAD DVTCT LIST (TX COUNT)

..
..**
..FUNCTIONAL DESCRIPTION: ULTCLS - UNLOAD TX COUNT LIST
..THIS ROUTINE TAKES DATA FROM DVTCLS AND MOVES
..IT TO DVTCT
..INPUTS: TRIBN - NUMBER OF LINE IN USE
..OUTPUTS: DVTCT - TRANSMIT COUNT VALUE
..SUBORDINATE ROUTINES USED:
..GETIND - GET INDEX VALUE FROM TRIBLS
..CALLING SEQUENCE:
..JSR PC,ULTCLS
..--

030652 004737 031102 ULTCLS: JSR PC,GETIND ;GET INDEX
030656 013703 007642 MOV INDEX,R3 ;MOVE BYTE INDEX TO R3
030662 116337 007612 010072 MOVB DVTCLS(R3),DVTCT
030670 000207 RTS PC ;RETURN TO CALLER

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 140
GET ALL RX POINTERS FROM LIST TO CPTRR

.SBTTL GET ALL RX POINTERS FROM LIST TO CPTRR

```

:++
: FUNCTIONAL DESCRIPTION:      GARPFL - GET ALL RX POINTERS FROM LIST
: THIS ROUTINE CHECKS ALL RX POINTERS FOR VALID LINES
: IN CPTRLS AND MAKES SURE THEY ARE ALL ZERO.
:
: OUTPUTS:      CPTRR - ZERO IF ALL CPTRLS IS ZERO
:               NON ZERO IF NOT.
:
: SUBORDINATE ROUTINES USED:
:               GTVIND - GET VALID INDEX
:               ULRPLS - UNLOAD CPTRR LIST TO CPTRR
:
: CALLING SEQUENCE:
:               JSR      PC,GARPFL
:--

```

4875
4876
4877
4878
4879
4880
4881
4882
4883
4884
4885
4886
4887
4888
4889
4890
4891
4892
4893
4894
4895
4896
4897
4898
4899
4900
4901

```

030672 013737 007636 010176 GARPFL: MOV      TRIBN,TEMP5
030700 012737 177777 007642      MOV      #-1,INDEX
030706 004737 030402      GARP1: JSR      PC,GTVIND      ;GET VALID INDEX
030712 022737 000010      CMP      #8.,INDEX      ;COMPARE INDEX
030720 001405      BEQ      GARPEX      ;EXIT IF DONE
030722 004737 030454      JSR      PC,ULRPLS      ;LOAD CPTRR WITH VALUE
030726 005737 010152      TST      CPTRR      ;TEST THE VALUE
030732 001765      BEQ      GARP1      ;IF ZERO CHECK NEXT
030734 013737 010176 007636 GARPEX: MOV      TEMP5,TRIBN
030742 000207      RTS      PC      ;RETURN TO CALLER WHEN DONE

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 141
GET ALL TX COUNTS FROM LIST TO DVTCT

4902
4903
4904
4905
4906
4907
4908
4909
4910
4911
4912
4913
4914
4915
4916
4917
4918
4919
4920
4921
4922
4923
4924
4925
4926
4927
4928

030744 013737 007636 010176
030752 012737 177777 007642
030760 005037 010072
030764 004737 030402
030770 022737 000010 007642
030776 001410
031000 013703 007642
031004 105763 007612
031010 001765
031012 012737 000001 010072
031020 013737 010176 007636
031026 000207

```
.SBTTL          GET ALL TX COUNTS FROM LIST TO DVTCT
:RETURN WITH DVTCT=1 IF ANY COUNT HAS SOME IN IT
:IF ALL COUNTS ARE ZERO EXIT

:++
: FUNCTIONAL DESCRIPTION:  GATCFL - GET ALL TX COUNTS FROM LIST
: THIS ROUTINE GETS AND CHECKS ALL TX COUNTS TO BE ZERO
: OUTPUTS:                DVTCT - ZERO IF LIST IS ZERO
:                          NON ZERO IF NOT
: SUBORDINATE ROUTINES USED:
:   GTVIND -              GET NEXT VALID INDEX
: CALLING SEQUENCE:
:   JSR      PC,GATCFL
--
GATCFL: MOV      TRIBN,TEMP5
        MOV      #-1,INDEX
        CLR      DVTCT          ;CLEAR COUNT
        JSR      PC,GTVIND      ;GET VALID INDEX
        CMP      #8,INDEX      ;IS INDEX =8 ALL DONE
        BEQ      GATCEX        ;IF SO EXIT
        MOV      INDEX,R3
        TSTB    DVTCLS(R3)     ;IS THIS COUNT 0
        BEQ      GATC1         ;IF THIS ONE IS ZERO
        MOV      #01,DVTCT     ;LOAD COUNT WITH A 1
GATCEX: MOV      TEMP5,TRIBN
        RTS      PC            ;RETURN TO CALLERR
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 142
GET NEXT TX POINTER FROM LIST

4929
4930
4931
4932
4933
4934
4935
4936
4937
4938
4939
4940
4941
4942
4943
4944
4945
4946
4947
4948
4949
4950
4951
4952
4953
4954
4955
4956
4957
4958
4959
4960
4961
4962
4963
4964
4965
4966
4967
4968
4969
4970
4971
4972
4973
4974
4975

.SBTTL GET NEXT TX POINTER FROM LIST

```

:++
: FUNCTIONAL DESCRIPTION: GNTXPR - GET NEXT TX POINTER
: THIS ROUTINE GETS THE NEXT TX POINTER TO CPTR
: OUTPUTS: CPTR - POINTER FOR NEXT TRANSMIT MESG
: SUBORDINATE ROUTINES USED:
: GTVIND - GET VALID INDEX
: CALLING SEQUENCE:
: JSR PC,GNTXPR
:--

```

```

GNTXPR: CMP #8,,INDEX ;IS INDEX = DONE
        BNE GNTX1
GNTX2: MOV #-1,INDEX
GNTX1: JSR PC,GTVIND
        CMP #8,,INDEX
        BEQ GNTX2
        JSR PC,ULTCLS ;GET COUNT FROM LIST
        TST DVICT ;TEST COUNT
        BEQ GNTXPR
        JSR PC,ULTPLS ;UNLOAD POINTER
        RTS ;RETURN TO CALLER

```

.SBTTL GET INDEX BYTE AND WORD

```

:++
: FUNCTIONAL DESCRIPTION: GETIND - GET INDEX FOR WORD AND BYTE
: THIS ROUTINE GETS INDEX LOADED WITH INDEX AND INDW WITH INDEX
: FOR WORD. IF TRIBLS ENTRY IS EQUAL TO TRIBN
: OUTPUTS: INDEX - BYTE INDEX
: INDW - WORD INDEX
: CALLING SEQUENCE:
: JSR PC,GETIND
:--

```

```

GETIND: MOV #-1,R3 ;LOAD R3 WITH -1
GETI1: INC R3 ;BUMP R3
        CMP #8,,R3 ;ARE WE ALL DONE
        BEQ GETIND ;IF SO GO BACK
        CMPB TRIBLS(R3),TRIBN ;ELSE COMPARE FOR THIS LINE
        BNE GETI1 ;BRANCH IF NO MATCH
GETI2: MOV R3,INDEX ;STORE OFF BYTE INDEX
        ASL R3 ;MAKE UP WORD INDEX
        MOV R3,INDW ;STORE OFF WORD INDEX
        RTS ;RETURN TO CALLER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 143
LOG AND QUE REC BUFFERS

4976
4977
4978
4979
4980
4981
4982
4983
4984
4985
4986
4987
4988
4989
4990
4991
4992
4993
4994
4995
4996
4997

031142 013702 010152
031146 011237 010170
031152 012237 010104
031156 011237 010172
031162 011237 010106
031166 010237 010152
031172 004737 064174
031176 004737 024200
031202 000207

```

.SBTTL          LOG AND QUE REC BUFFERS
:++
: FUNCTIONAL DESCRIPTION:      LOGAQR - QUE AND LOG RX BUFFERS
:                               THIS ROUTINE QUEUES THE REC BUFFER POINTED TO BY
:                               CPTRR
: INPUTS:                      CPTRR - POINTS TO POINTER TABLE ENTRY
: IMPLICIT OUTPUTS:           BUFFER QUEUED FOR THIS ENTRY
: CALLING SEQUENCE:           JSR      PC,LOGAQR
:--

LOGAQR: MOV      CPTRR,R2          ;LOAD R2 FROM POINTER
        MOV      (R2),TEMP2       ;SET UP ADDRESS FOR LOGGING
        MOV      (R2)+,DVRXA      ;SET UP ADDRESS OF RECEIVE BUFFER FOR DEVICE
        MOV      (R2),TEMP3       ;SET UP CHAR COUNT FOR LOGGING
        MOV      (R2),DVRCC       ;SET UP COUNT FOR DEVICE
        MOV      R2,CPTRR         ;LEAVE POINTER POINTING AT CHAR COUNT
        JSR      PC,DVRXQ         ;QUEUE REC BUFFER
        JSR      PC,LOGRXQ        ;LOG RXQ
        RTS      PC              ;RETURN TO CALLER

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 144
SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS

4998
4999
5000
5001
5002
5003
5004
5005
5006
5007
5008
5009
5010
5011
5012
5013
5014
5015
5016
5017
5018
5019
5020
5021
5022
5023
5024
5025
5026
5027
5028
5029
5030
5031
5032
5033
5034
5035
5036
5037
5038
5039
5040
5041
5042
5043
5044
5045
5046
5047
5048
5049
5050
5051
5052
5053

.SBTTL SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS

..++
FUNCTIONAL DESCRIPTION:
SHWOP - SHOW MODE OF OPERATION, LOOP, QULAIFIERS
PRINTED ON THE OPERATOR'S CONSOLE.

INPUTS:
DEV1= MODE TYPE (MODTYP)
DEV2= MAINT LOOP TYPE (MLTYP)
DEV3= "RUN PASS" COUNT (RPASS) - COUNT DOWN
DEV4= PARAMTERS WORD (PARAM)

IMPLICIT INPUTS:
MODES= TABLE OF ADDRESSES OF MODE NAME STRINGS
LOOPS= TABLE OF ADDRESSES OF LOOP TYPE NAMES

CALLING SEQUENCE:
JSR PC,SHWOP
..--

```

SHWOP:  MOV    DEV1,R2          ;GET THE MODE TYPE IN R2
        ASL    R2              ;MAKE IT A WORD TABLE OFFSET
        MOV    MODES(R2),TEMP  ;GET ADDRESS OF MODE-IN-ASCII
        MOV    DEV2,R2        ;GET MAINTENANCE LOOP TYPE
        ASL    R2
        MOV    #LPO0,TEMP3     ;LOAD TEMP3 TO POINT TO "/LOOP="
        TST    R2              ;SEE IF /LOOP=XXXXX OR NONE
        BNE    10$            ;BR IF /LOOP= OF SOME KIND
        MOV    #LPO,TEMP3      ;IF NO LOOP THEN DON'T PRINT "/LOOP="
10$:    MOV    LOOPS(R2),TEMP1  ;GET ADDRESS OF LOOP-IN-ASCII
        MOV    DEV3,TEMP2     ;GET NUMBER OF PASSES
        PRINTS #SHF0,TEMP,TEMP3,TEMP1,TEMP2
                                MOV    TEMP2,-(SP)
                                MOV    TEMP1,-(SP)
                                MOV    TEMP3,-(SP)
                                MOV    TEMP,-(SP)
                                MOV    #SHF0,-(SP)
                                MOV    #5,-(SP)
                                MOV    SP,R0
                                TRAP   C$PNTS
                                ADD    #14,SP

        CLR    R2              ;NOW SET UP FOR QUALIFIERS IN ASCII
        MOV    #PST,TEMP
        BIT    #STATB,DEV4     ;SEE IF /STATUS OR /NOSTATUS
        BNE    1$              ;BR IF /STATUS
        MOV    #PNST,TEMP
        MOV    #PCK,TEMP1
        BIT    #DATCKB,DEV4    ;SEE IF /CHECK OR /NOCHECK
        BNE    2$              ;BR IF /CHECK
        MOV    #PNCK,TEMP1
        MOV    #PEC,TEMP2
        BIT    #ECHOB,DEV4     ;SEE IF /ECHO OR /NOECHO
        BNE    3$              ;BR IF /ECHO
        MOV    #PNEC,TEMP2

```

CZKMSAO KMS11-BD/BE DCLT
KMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 145
SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS

```

5054
5055
5056 031420 012737 016527 010176 3$: MOV #PMS,TEMP5
5057 031426 032737 000010 011476 BIT #MOCHK,DEV4 ;SEE IF MODEM OR /NOMODEM
5058 031434 001003 BNE 5$ ;BRANCH IF MODEM
5059 031436 012737 016525 010176 MOV #PNMS,TEMP5
5060
5061 031444 5$: PRINTS #SHF1,TEMP,TEMP1,TEMP2,TEMP5 ;,TEMP3,TEMP4 **RFU**
5062 031444 013746 010176 MOV TEMP5,-(SP)
5063 031450 013746 010170 MOV TEMP2,-(SP)
5064 031454 013746 010166 MOV TEMP1,-(SP)
5065 031460 013746 010164 MOV TEMP,-(SP)
5066 031464 012746 017226 MOV #SHF1,-(SP)
5067 031470 012746 000005 MOV #5,-(SP)
5068 031474 010600 MOV SP,R0
5069 031476 104416 TRAP C$PNTS
5070 031500 062706 000014 ADD #14,SP
5071 031504 000207 RTS PC ;RETURN
5072
5073

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 146
TRAVERSE COMMAND LINE SUBROUTINES

```

5074 .SBTTL TRVERSE COMMAND LINE SUBROUTINES
5075
5076
5077 :++
5078 P$TRV SUBROUTINE
5079 :PARSE THE COMMAND LINE SUBROUTINE
5080 :TAKE ACTIONS (VIA ACTION TREE) AS PARSING LINE
5081 :PARSING DIRECTIONS FROM "CLI PARSING NODES"
5082 : REGS USED:
5083 : PSNUM=NUMERIC CODE FROM DATA
5084 : R1,R5=SCRATCH
5085 : R2=ACTION CODE PARAMETER FROM TREE
5086 : R3=PARSE TREE POINTER
5087 : R4=INPUT STRING POINTER
5088 : CALLING SEQUENCE:
5089 : JSR PC,P$TRV
5090 :--
5091
5092 031506 P$TRV: MOV PSBUFA,R4
5093 031506 013704 003364 MOV P$TREE,R3
5094 031512 013703 003366 P$TR5: TSTB (R4)
5095 031516 105714 :SEE IF ANY CHARS LEFT IN INPUT STRING
5096 031520 001441 :BR IF NO
5097 031522 121327 000013 BEQ P$EXIT :SEE IF SPECIAL CLI CHAR CODE OR ASCII
5098 031526 003023 CMPB (R3),#11. :BR IF REGULAR ASCII CHAR.
5099 031530 111305 BGT 20$ :GET SPECIAL CHAR CODE INTO R5
5100 031532 006305 MOVB (R3),R5
5101 031534 016505 031550 ASL R5 :BUILD TRAVERSE ROUTINE ADDRESS
5102 031540 062705 031550 MOV 10$(R5),R5
5103 031544 004715 JSR PC,(R5) :JSR TO SPECIAL CLI TRAVERSE ROUTINE
5104 031546 000763 BR P$TR5 :GO SEE IF MORE OF STRING LEFT
5105
5106 031550 000114 10$: .WORD TRVERR-10$ :TRAVERSE TABLE FOR "CLI FUNCTIONS"
5107 031552 000134 .WORD TRVEXI-10$ :1
5108 031554 000152 .WORD TRVBR-10$ :2
5109 031556 000162 .WORD TRVBIF-10$ :3
5110 031560 000204 .WORD TRVSPA-10$ :4
5111 031562 000270 .WORD TRVNUM-10$ :5
5112 031564 000604 .WORD TRVALP-10$ :6
5113 031566 000650 .WORD TRVALN-10$ :7
5114 031570 000270 .WORD TRVOCT-10$ :8
5115 031572 000256 .WORD TRVDEC-10$ :9
5116 031574 000736 .WORD TRVSTR-10$ :10
5117
5118
5119 031576 121314 20$: CMPB (R3),(R4) :NOT A SPECIAL CODE
5120 031600 001403 BEQ 22$ :SEE IF FIRST CHAR OF STRING IS A MATCH
5121 031602 004737 031646 JSR PC,TRVBR :BR IF A MATCH
5122 031606 000743 BR P$TR5 :IF NOT A MATCH, GO TAKE MISS BRANCH
5123 031610 004737 031626 22$: JSR PC,TRVACT : THEN GO BACK PT'G TO MISS NODE
5124 031614 062703 000004 ADD #4,R3 :IF A MATCH, GO DO ACTION DEFINED BY
5125 : ACTION CODE IN CLI NODE, THEN
5126 031620 005204 INC R4 : ADJUST PTR TO NEXT CLI NODE
5127 031622 000735 BR P$TR5 :ADJUST BUF PTR TO NEXT CHAR IF MATCH
5128
5129 031624 000207 P$EXIT: RTS PC :RETURN FROM PARSER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 147
TRAVERSE COMMAND LINE SUBROUTINES

```

5130 ;-----
5131 :GOTO USER ACTION ROUTINE
5132 TRVACT: MOVB 1(R3),R2 ;GET ACTION CODE FROM CLI NODE
5133 031626 116302 000001 BIC #177400,R2 ;CLEAR ANY SIGN EXTENSION
5134 031632 042702 177400 MOV P$ACT,R5 ;GET ADDRESS OF CLI ACTION ROUTINE
5135 031636 013705 003370 JSR PC,(R5) ;GO DO ACTION DEFINED BY CODE
5136 031642 004715 RTS PC ;RETURN TO CALLING CODE
5137 031644 000207
5138
5139 :TAKE BRANCH IN TREE
5140 031646 016305 000002 TRVBRC: MOV 2(R3),R5 ;GET BRANCH DISPLACEMENT FROM TREE
5141 031652 060503 ADD R5,R3 ;AND POINT R3 TO THE "MISS" NODE
5142 031654 000207 RTS PC ;RETURN TO P$TRV
5143
5144 :NO BRANCH TAKEN
5145 031656 062703 000004 TRVNOB: ADD #4,R3 ;THINGS OK, UPDATE R3 TO POINT TO NEXT
5146 031662 000207 RTS PC ;NODE AND RETURN TO P$TRV
5147
5148 ;-----
5149 031664 004737 031626 TRVERR: JSR PC,TRVACT ;TAKE ERROR ACTION
5150 031670 112737 177777 003401 MOVB #-1,P$GDBD ;SET ERROR RETURN FLAG
5151 031676 005726 TST (SP)+ ;GET RID OF "JSR PUSH TO TRVERR"
5152 031700 000137 031624 JMP P$EXIT ;RETURN DIRECT TO EXIT OF P$TRV ROUTINE
5153
5154 031704 004737 031626 TRVEXI: JSR PC,TRVACT ;TAKE EXIT ACTION
5155 031710 105037 003401 CLRB P$GDBD ;SET GOOD/BAD FLAG TO "SUCCESS (0)"
5156 031714 005726 TST (SP)+ ;GET RID OF "JSR PUSH TO TRVEXI"
5157 031716 000137 031624 JMP P$EXIT ;RETURN DIRECT TO EXIT OF P$TRV ROUTINE
5158
5159 031722 004737 031626 TRVBR: JSR PC,TRVACT ;GO TAKE BRANCH ACTION
5160 031726 000137 031646 JMP TRVBRC
5161
5162 031732 004737 031626 TRVBIF: JSR PC,TRVACT
5163 031736 105737 003401 TSTB P$GDBD ;SEE IF P$GDBD SET OR CLEARED BY ACTION
5164 031742 001402 BEQ 1$ ;IF CLEAR FALL THRU TO NEXT NODE
5165 031744 000137 031646 JMP TRVBRC ;ELSE TAKE THE "MISS" BRANCH
5166 031750 000137 031656 1$: JMP TRVNOB ;JUST UPDATE TO NEXT NODE IF THINGS OK
5167
5168 031754 005005 TRVSPA: CLR R5 ;CLEAR "SPACE OR TAB FOUND" FLAG
5169 031756 121427 000011 1$: CMPB (R4),#11 ;SEE IF CHAR. IN CMD LINE= TAB
5170 031762 001003 BNE 2$ ;BR IF NO, NOT A TAB
5171 031764 005204 INC R4 ;INC INPUT STRING POINTER
5172 031766 005205 INC R5 ;INDICATE A TAB FOUND
5173 031770 000772 BR 1$ ;GO CHECK NEXT CHAR
5174
5175 031772 121427 000040 2$: CMPB (R4),#40 ;SEE IF CHAR. IN CMD LINE= SPACE
5176 031776 001003 BNE 10$ ;BR IF NO, NON-SPACE OR NON-TAB CHAR.
5177 032000 005204 INC R4 ;INC INPUT STRING POINTER
5178 032002 005205 INC R5 ;INDICATE A SPACE FOUND
5179 032004 000764 BR 1$ ;GO CHECK NEXT CHAR
5180 032006 005705 10$: TST R5 ;SEE IF ANY SPACES OR TABS FOUND
5181 032010 001404 BEQ 15$ ;BR IF NO, TAKE NO ACTION
5182 032012 004737 031626 JSR PC,TRVACT ;GO TAKE ACTION IF ANY FOUND
5183 032016 000137 031656 JMP TRVNOB ;JUST GO UPDATE R3 TO NEXT NODE IF OK
5184 032022 000137 031646 15$: JMP TRVBRC ;TAKE BRANCH (MISS) IF NONE FOUND

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 148
TRAVERSE COMMAND LINE SUBROUTINES

```

5185
5186 032026 012737 000012 003376 TRVDEC: MOV #10.,PSRADX ;USE DECIMAL AS RADIX AND ASSUME +
5187 032034 000137 032046 JMP TRVNMA
5188 032040 TRVOCT: ;(SAME AS TRVNUM SINCE DEFAULT RADIX IS OCTAL)
5189 032040 012737 000010 003376 TRVNUM: MOV #8.,PSRADX ;USE OCTAL AS RADIX AND ASSUME +
5190 032046 005005 TRVNMA: CLR R5 ;CLEAR DIGIT COUNTER
5191 032050 121427 000053 CMPB (R4),#'+' ;SEE IF THERE'S A + SIGN THERE
5192 032054 001001 BNE 10$ ; BR IF NO
5193 032056 000406 BR 11$ ; ELSE PSRADX ALREADY SAYS +, JUST BR
5194 032060 121427 000055 10$: CMPB (R4),#'-' ;SEE IF THERE'S A - SIGN THERE
5195 032064 001004 BNE 1$ ; BR IF NO
5196 032066 112737 177777 003377 MOVB #-1,PSRADX+1 ;SET "MINUS FLAG" (HI BYTE OF PSRADX)
5197 032074 005204 11$: INC R4 ;BUMP R4 TO POINT TO FIRST CHAR
5198
5199 032076 121427 000060 1$: CMPB (R4),#60 ;SEE IF CHAR. LESS THAN A "0"
5200 032102 002434 BLT 2$ ;BR IF YES (NOT NUMERIC)
5201 032104 121427 000067 CMPB (R4),#67 ;SEE IF CHAR. GREATER THAN A "7"
5202 032110 003426 BLE 13$ ; BR IF YES
5203 032112 123727 003376 000012 CMPB PSRADX,#10. ;SEE IF IN DECIMAL MODE
5204 032120 001417 BEQ 12$ ; BR IF YES (CAN USE HIGHER LIMIT)
5205 032122 121427 000071 CMPB (R4),#71 ;SEE IF DIGIT WAS A 8 OR 9
5206 032126 003022 BGT 2$ ;BR IF NON-NUMERIC
5207 032130 PRINTF #CLIBRX ;ELSE WAS A 8 OR 9 WHEN IN OCTAL RADIX
5208 032130 012746 014105 MOV #CLIBRX,-(SP)
5209 032134 012746 000001 MOV #1,-(SP)
5210 032140 010600 MOV SP,R0
5211 032142 104417 TRAP C$PNTF
5212 032144 062706 000004 ADD #4,SP
5213 032150 112737 177777 003401 MOVB #-1,P$GDBD ;SET ERROR RETURN FLAG
5214 032156 000474 BR 5$ ; PRINT ERROR AND TAKE MISS
5215
5216 032160 121427 000071 12$: CMPB (R4),#71 ;SEE IF CHAR. GREATER THAN A "9"
5217 032164 003003 BGT 2$ ;BR IF YES (NOT NUMERIC)
5218 032166 005204 13$: INC R4 ;UPDATE CMD LINE PTR TO NEXT CHAR.
5219 032170 005205 INC R5 ;INDICATE A NUMERIC FOUND
5220 032172 000741 BR 1$ ;GO LOOK AT NEXT CHAR.
5221
5222 032174 005705 2$: TST R5 ;SEE IF FOUND ANY NUMERICS
5223 032176 001464 BEQ 5$ ;BR IF NO, TAKE "MISS" BRANCH
5224 032200 010401 MOV R4,R1 ;GET POINTER TO START OF NUMERIC STRING
5225 032202 160501 SUB R5,R1
5226 032204 005037 003374 CLR PSNUM ;CLEAR LOC. WHERE VALUE WILL BE STORED
5227 032210 112102 3$: MOVB (R1)+,R2 ;GET ASCII CHAR AND CONVERT IT TO A #
5228 032212 162702 000060 SUB #60,R2
5229 032216 006337 003374 ASL PSNUM
5230 032222 103437 BCS 7$ ;SHIFT CURRENT VALUE TO MAKE ROOM
5231 032224 013737 003374 003372 MOV PSNUM,P$CNT ;ERROR IF NUMBER TOO BIG
5232 032232 006337 003374 ASL PSNUM ;SAVE FOR LATER IN CASE DECIMAL RADIX
5233 032236 103431 BCS 7$ ;ERROR IF NUMBER TOO BIG
5234 032240 006337 003374 ASL PSNUM
5235 032244 103426 BCS 7$ ;ERROR IF NUMBER TOO BIG
5236 032246 123727 003376 000012 CMPB PSRADX,#10. ;SEE IF DECIMAL RADIX
5237 032254 001004 BNE 4$ ;BR IF NOT EQUAL
5238 032256 063737 003372 003374 ADD P$CNT,PSNUM
5239 032264 103416 BCS 7$ ;ERROR IF NUMBER TOO BIG
5240 032266 060237 003374 4$: ADD R2,PSNUM

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 149
TRAVERSE COMMAND LINE SUBROUTINES

5241	032272	103413		BCS	7\$:ERROR IF NUMBER TOO BIG
5242	032274	005305		DEC	R5		
5243	032276	001344		BNE	3\$		
5244	032300	105737	003377	TSTB	PSRADX+1		:SEE IF NUM WAS PRECEDED BY A - SIGN
5245	032304	001402		BEQ	15\$: BR IF NO
5246	032306	005437	003374	NEG	PSNUM		: ELSE NEGATE THE NUMBER BEFORE LEAVING
5247	032312	004737	031626	15\$: JSR	PC,TRVACT		:SINCE NUMERIC FOUND, GO TAKE ACTION
5248	032316	000137	031656	JMP	TRVNOB		:GO POINT R3 TO NEXT NODE
5249							
5250	032322			7\$: PRINTF	#CLINBG		:PRINT NUMBER TOO BIG ERROR
5251	032322	012746	014063				MOV #CLINBG,-(SP)
5252	032326	012746	000001				MOV #1,-(SP)
5253	032332	010600					MOV SP,R0
5254	032334	104417					TRAP C\$PNTF
5255	032336	062706	000004				ADD #4,SP
5256	032342	112737	177777	003401	5\$: MOVB	#-1,P\$GDBD	:SET ERROR RETURN FLAG
5257	032350	000137	031646	JMP	TRVBRC		:TAKE "MISS" BRANCH
5258							
5259							
5260	032354	005005		TRVALP:	CLR R5		:CLEAR ALPHA FOUND FLAG
5261	032356	121427	000101	1\$: CMPB	(R4),#101		:SEE IF CHAR. LESS THAN A "A"
5262	032362	002406		BLT	2\$:BR IF YES (NOT ALPHA)
5263	032364	121427	000132	CMPB	(R4),#132		:SEE IF CHAR. GREATER THAN A "Z"
5264	032370	003003		BGT	2\$:BR IF YES (NOT ALPHA)
5265	032372	005204		INC	R4		:UPDATE CMD LINE PTR TO NEXT CHAR
5266	032374	005205		INC	R5		:INDICATE AN ALPHA WAS FOUND
5267	032376	000767		BR	1\$:GO LOOK AT NEXT CHAR.
5268	032400	005705		2\$: TST	R5		:SEE IF ANY ALPHA'S WERE FOUND
5269	032402	001404		BEQ	3\$:BR IF NO
5270	032404	004737	031626	JSR	PC,TRVACT		:IF ANY FOUND TAKE ACTION
5271	032410	000137	031656	JMP	TRVNOB		:THEN UPDATE R3 TO NEXT NODE -NO BRANCH
5272	032414	000137	031646	3\$: JMP	TRVBRC		:NONE FOUND, TAKE MISS BRANCH
5273							
5274	032420	005005		TRVALN:	CLR R5		:CLEAR ALPHANUM FOUND FLAG
5275	032422	121427	000060	10\$: CMPB	(R4),#60		:SEE IF CHAR. LESS THAN A "0"
5276	032426	002417		BLT	2\$:BR IF YES (NOT NUMERIC OR ALPHA)
5277	032430	121427	000072	CMPB	(R4),#72		:SEE IF CHAR. GREATER THAN A "9"
5278	032434	003003		BGT	1\$:BR IF YES (NOT NUMERIC)
5279	032436	005204		INC	R4		:UPDATE CMD LINE PTR TO NEXT CHAR.
5280	032440	005205		INC	R5		:INDICATE A NUMERIC FOUND
5281	032442	000767		BR	10\$:GO LOOK AT NEXT CHAR.
5282	032444	121427	000101	1\$: CMPB	(R4),#101		:SEE IF CHAR. LESS THAN A "A"
5283	032450	002406		BLT	2\$:BR IF YES (NOT ALPHA)
5284	032452	121427	000132	CMPB	(R4),#132		:SEE IF CHAR. GREATER THAN A "Z"
5285	032456	003003		BGT	2\$:BR IF YES (NOT ALPHA)
5286	032460	005204		INC	R4		:UPDATE CMD LINE PTR TO NEXT CHAR
5287	032462	005205		INC	R5		:INDICATE AN ALPHA FOUND
5288	032464	000756		BR	10\$:GO LOOK AT NEXT CHAR.
5289	032466	005705		2\$: TST	R5		:SEE IF ANY ALPHANUM'S WERE FOUND
5290	032470	001404		BEQ	3\$:BR IF NO
5291	032472	004737	031626	JSR	PC,TRVACT		:IF ANY FOUND TAKE ACTION
5292	032476	000137	031656	JMP	TRVNOB		:THEN UPDATE R3 TO NEXT NODE -NO BRANCH
5293	032502	000137	031646	3\$: JMP	TRVBRC		:NONE FOUND, TAKE MISS BRANCH
5294							
5295							
5296							

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 150
TRAVERSE COMMAND LINE SUBROUTINES

```

5297 032506 010401          TRVSTR: MOV      R4,R1          ;POINT R1 TO CMD STRING
5298 032510 010305          MOV      R3,R5
5299 032512 062705 000006   ADD      #6,R5          ;POINT R5 TO MATCH STRING FROM CLI NODE
5300 032516 005037 003372   CLR      P$CNT          ;CLEAR CHAR MATCH COUNT
5301 032522 105715          2$:  TSTB   (R5)          ;SEE IF END OF MATCH STRING YET
5302 032524 001411          BEQ      10$            ;BR IF YES
5303 032526 105711          TSTB   (R1)            ;SEE IF END OF CMD LINE YET
5304 032530 001407          BEQ      10$            ;BR IF YES
5305 032532 121115          CMPB   (R1),(R5)        ;SEE IF CHARACTERS MATCH
5306 032534 001005          BNE     10$            ;BR IF NO
5307 032536 005237 003372   INC     P$CNT          ;MATCH -INCREMENT MATCH COUNT
5308 032542 005201          INC     R1              ;UPDATE STRING POINTERS
5309 032544 005205          INC     R5
5310 032546 000765          BR      2$              ;BR TO CONTINUE CHECKING CHARS.
5311
5312 032550 005737 003372   10$:  TST   P$CNT          ;WHEN DONE SEE IF ANY MATCHES FOUND
5313 032554 001406          BEQ     15$            ;BR IF NO, GO TAKE THE MISS BRANCH
5314 032556 010104          MOV     R1,R4          ;POINT CMD POINTER TO END OF STRING &
5315 032560 004737 031626   JSR    PC,TRVACT        ;IF A MATCH FOUND, GO DO MATCH ACTION
5316 032564 066303 000004   ADD     4(R3),R3        ;UPDATE R3 TO NEXT NODE (NO BRANCH)
5317 032570 000207          RTS     PC              ; (NO RETURN THRU TRVNOB SINCE DIFFERENT
5318                                     ; DISPLACEMENT DUE TO MATCH STRING)
5319 032572 000137 031646   15$:  JMP     TRVBRC        ; GO TAKE BRANCH
5320
5321                                     ; (PARSED OK), -1 IF ILL CMD.....
5322 -----
5323

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 151
REPORT CODING SECTION

.SBTTL REPORT CODING SECTION

..+
: THE REPORT CODING SECTION CONTAINS THE
: 'PRINTS' CALLS THAT GENERATE STATISTICAL REPORTS.
:--

5324
5325
5326
5327
5328
5329
5330
5331
5332
5333
5334
5335
5336
5337
5338
5339
5340
5341
5342
5343
5344

032576
032576

032576 004737 024644

032602
032602 104425

BGNRPT

LSRPT::

JSR PC,REPORT

:CALL SUBROUTINE TO DUMP EVENT LOG
: AND BASE TABLE OR TO GET LINE STATUS

ENDRPT

L10013: TRAP CSRPT

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 152
PROTECTION TABLE

.SBTTL PROTECTION TABLE

:++
: THIS TABLE IS USED BY THE RUNTIME SERVICES
: TO PROTECT THE LOAD MEDIA.
:--

5345
5346
5347
5348
5349
5350
5351
5352 032604
5353 032604
5354
5355 032604 177777
5356 032606 177777
5357 032610 177777
5358
5359 032612
5360

BGNPROT

L\$PROT::

-1 ;OFFSET INTO P-TABLE FOR CSR ADDRESS
-1 ;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
-1 ;OFFSET INTO P-TABLE FOR DRIVE NUMBER

ENDPROT

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 153
INITIALIZE SECTION

.SBTTL INITIALIZE SECTION

: THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
: AT THE BEGINNING OF EACH PASS.
:--

5361									
5362									
5363									
5364									
5365									
5366									
5367									
5368	032612				BGNINIT				
5369	032612								L\$INIT::
5370									
5371	032612	005037	003246		CLR	KEYWD1			:CLEAR USER COMMAND SPECIFIER
5372	032616	005737	010212		TST	DCLFLG			:CLEAN UP AND EXIT REQUESTED ?
5373	032622	001403			BEQ	INIT1			:BRANCH IF NOT
5374	032624	005037	010212		CLR	DCLFLG			:CLEAR CLEANUP REQUEST SPECIFIER
5375	032630				DOCLN				:GO CLEAN UP
5376	032630	104444							TRAP CS\$DCLN
5377									
5378	032632	012737	177777	010214	INIT1:	MOV	#-1,RESFLG		:SET RESTART FLAG
5379	032640				READEF	#EF.START			:IF HERE CAUSE OF START, DO SOME INIT
5380	032640	012700	000040						MOV #EF.START,RO
5381	032644	104447							TRAP CS\$REFG
5382	032646				BCOMPLETE		START		
5383	032646	103417							BCS START
5384	032650				READEF	#EF.RESTART			:IF HERE CAUSE OF RESTART, DO SOME INIT
5385	032650	012700	000037						MOV #EF.RESTART,RO
5386	032654	104447							TRAP CS\$REFG
5387	032656				BCOMPLETE		RESTRT		
5388	032656	103514							BCS RESTRT
5389	032660				READEF	#EF.CONTINUE			:SEE IF WE'RE HERE CAUSE OF A CONTINUE
5390	032660	012700	000036						MOV #EF.CONTINUE,RO
5391	032664	104447							TRAP CS\$REFG
5392	032666				BNCOMPLETE		S1		:BR IF NOT HERE CAUSE OF CONTINUE
5393	032666	103002							BCC S1
5394	032670	000137	033646		JMP	ENDIT			:JMP IF HERE CAUSE OF A CONTINUE
5395	032674			S1:	READEF	#EF.NEW			:SEE IF THIS IS A 'NEW PASS'
5396	032674	012700	000035						MOV #EF.NEW,RO
5397	032700	104447							TRAP CS\$REFG
5398	032702				BCOMPLETE		NEW		:IF YES, BR AROUND LOGUNIT # SETUP
5399	032702	103522							BCS NEW
5400	032704	000524			BR	GETPRM			
5401									
5402	032706	005037	010214		START:	CLR	RESFLG		:CLEAR RESTART FLAG SINCE HERE ON START
5403	032712				BRESET				:INIT ENTIRE BUS
5404	032712	104433							TRAP CS\$RESET
5405	032714	005037	010254		CLR	CLKVEC			:CLEAR CLK VECTOR PTR. AS A FLAG IN
5406									: NO CLOCK IS FOUND.
5407	032720	012702	010250		MOV	#CLKCSR,R2			:SETUP R2 AS A PTR. TO CLOCK INFO BLOCK
5408	032724				CLOCK	L,R1			:LOOK FOR A LINE CLOCK
5409	032724	012700	000114						MOV #L,RO
5410	032730	104462							TRAP CS\$CLK
5411	032732	010001							MOV RO,R1
5412	032734				BNCOMPLETE		S2		: IF NONE THERE GO LOOK FOR A P-CLOCK
5413	032734	103006							BCC S2
5414	032736	004737	023776		JSR	PC,CLKSET			: GO SET UP CLOCK INFO TABLE & CLK VEC.
5415	032742	012737	000100	010260	MOV	#LCLKEN,CLKEN			:SETUP THE ENABLE LINE CLOCK DATA
5416	032750	000457			BR	RESTRT			

```

5417
5418 032752
5419 032752 012700 000120
5420 032756 104462
5421 032760 010001
5422 032762
5423 032762 103017
5424 032764 004737 023776
5425 032770 062737 000002 010250
5426 032776 012777 001600 155244
5427 033004 162737 000002 010250
5428 033012 012737 000111 010260
5429 033020 000433
5430
5431 033022
5432 033022 104407
5433 033024
5434 033024 103021
5435 033026 012737 000100 010254
5436 033034 005037 010252
5437 033040 012737 010260 010250
5438 033046
5439 033046 104443
5440 033050 000406
5441 033052 010256
5442 033054 000052
5443 033056 016553
5444 033060 000377
5445 033062 000062
5446 033064 000074
5447 033066
5448 033066 000410
5449
5450
5451 033070
5452 033070 012746 016671
5453 033074 012746 000001
5454 033100 010600
5455 033102 104417
5456 033104 062706 000004
5457 033110 005037 010262
5458 033114 005037 010264
5459 033120 013737 010256 010266
5460 033126 012702 010300
5461 033132 010237 010276
5462 033136 012722 177777
5463 033142 020227 011334
5464 033146 001373
5465
5466 033150
5467 033150 012737 177777 010206
5468
5469 033156 005237 010206
5470 033162 023737 010206 002012
5471 033170 002367
5472

S2:  CLOCK P,R1 ;LOOK FOR A P-CLOCK SINCE NO LINE CLOCK
      MOV #P,RO
      TRAP C$CLCK
      MOV RO,R1
      BNCOMPLETE S3 ; IF NONE THERE GO SEE IF THIS IS LSI
      BCC S3 ; ELSE GO SET UP CLOCK INFO & VECTOR
      JSR PC,CLKSET ;POINT CLKCSR TO P-CLK COUNT SET REG.
      ADD #2,CLKCSR ;LOAD CLK SET REG. WITH COUNT VALUE
      MOV #PCLKCT,@CLKCSR ;POINT CLKCSR BAC TO P-CLK CSR
      SUB #2,CLKCSR ;SETUP THE ENABLE THE P-CLK DATA
      MOV #PCLKEN,CLKEN
      BR RESTRT

S3:  READBUS ;READ BUS TYPE TO SEE IF ON AN LSI
      TRAP C$RDBU
      BNCOMPLETE S4 ;BR IF NOT, NO CHANCE OF A CLOCK
      BCC S4
      MOV #100,CLKVEC ;LOAD 100 AS CLK VECTOR
      CLR CLKBR ;LOAD 0 AS CLK INT. LEVEL
      MOV #CLKEN,CLKCSR ;KLUDGE UP THE CSR & ENABLE DATA LOCS
      GMANID L5060,CLKHZ,D,377,50,,60,,YES
      TRAP C$GMAN
      BR 10000$
      .WORD CLKHZ
      .WORD T$CODE
      .WORD L5060
      .WORD 377
      .WORD T$LO!IM
      .WORD T$HILIM

S4:  PRINTF #BDCLK
      MOV #BDCLK,-(SP)
      MOV #1,-(SP)
      MOV SP,RO
      TRAP C$PNTF
      ADD #4,SP
      RESTRT: CLR TIMMIN ;CLEAR TIME SINCE START LOCATIONS
              CLR TIMSEC
              MOV CLKHZ,TIMTCK ;LOAD TICKS/SEC
              MOV #EVTLOG,R2 ;INIT EVENT TABLE TO ALL 1'S AFTER EACH
              MOV R2,EVTPIR ; START OR RES AND INIT TABLE POIN'ER
1$:  MOV #-1,(R2)+
      CMP R2,#EVTEND ;SEE IF REACHED END OF TABLE
      BNE 1$ ;LOOP UNTIL DONE

NEW:  MOV #-1,LOGUNT ;INITIALIZE LOGICAL UNIT #

GETPRM: INC LOGUNT ;POINT TO NEXT LOGICAL UNIT
         CMP LOGUNT,LSUNIT ;SEE IF PAST MAX. LOG. UNIT #
         BGE NEW ;BR IF YES, AND START OVER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 155
INITIALIZE SECTION

```

5473 033172          GPHARD LOGUNT,R1          ;GET THE P-TABLE FOR THIS LOG. UNIT
5474 033172 013700 010206          MOV LOGUNT,RO
5475 033176 104442          TRAP C$GPHRD
5476 033200 010001          MOV RO,R1
5477 033202          BNCOMPLETE GETPRM          ;IF NO P-TABLE AVAIL., GO GET NEXT ONE
5478 033202 103365          BCC GETPRM
5479
5480 033204 012737 000001 010222  MOV #1,FHDPLX          ;FULL DUPLEX ONLY
5481
5482
5483          ;DEVICE DEPENDENT PART OF GETTING INFO FROM P-TABLE
5484
5485 033212 011137 013372          MOV (R1),SELO          ;STORE AWAY CSR ADDRESSES
5486 033216 011137 013374          MOV (R1),BSEL1
5487 033222 005237 013374          INC BSEL1
5488 033226 011137 013376          MOV (R1),SEL2
5489 033232 062737 000002 013376  ADD #2,SEL2
5490 033240 011137 013400          MOV (R1),BSEL3
5491 033244 062737 000003 013400  ADD #3,BSEL3
5492 033252 011137 013402          MOV (R1),SEL4
5493 033256 062737 000004 013402  ADD #4,SEL4
5494 033264 011137 013404          MOV (R1),BSEL5
5495 033270 062737 000005 013404  ADD #5,BSEL5
5496 033276 011137 013406          MOV (R1),SEL6
5497 033302 062737 000006 013406  ADD #6,SEL6
5498 033310 011137 013410          MOV (R1),BSEL7
5499 033314 062737 000007 013410  ADD #7,BSEL7
5500
5501 033322 016137 000002 013412  MOV 2(R1),INVEC          ;STORE AWAY INPUT INTERRUPT VECTOR
5502 033330 016137 000002 013414  MOV 2(R1),OUTVEC
5503 033336 062737 000004 013414  ADD #4,OUTVEC          ;BUILD OUTPUT INTERRUPT VECTOR
5504 033344 016137 000004 013416  MOV 4(R1),INTPRI          ;STORE AWAY INTERRUPT PRIORITY
5505 033352 016137 000006 013426  MOV 6(R1),MODCSR          ;STORE AWAY MODEM CSR (DM11-BA) ADDR
5506
5507          ;INITIALIZATION CODE - LOAD FIRMWARE INTO KMC-11B
5508
5509 033360 005005          LDFIRM: CLR R5          ;SPECIFY INITIAL CRAM ADDRESS
5510 033362 012704 033660          MOV #DDCMPS,R4          ;POINT TO KMS FIRMWARE IMAGE
5511 033366 012777 002000 157776  MOV #RAM0,@SELO          ;SPECIFY LOAD CRAM
5512 033374 010577 160002          2$: MOV R5,@SEL4          ;WRITE CRAM ADDRESS
5513 033400 011477 160002          MOV (R4),@SEL6          ;WRITE IMAGE WORD TO CRAM
5514 033404 017702 157776          MOV @SEL6,R2          ;READ BACK WORD JUST WRITTEN
5515 033410 022402          CMP (R4)+,R2          ;IF WORD READ BACK IS WORD WRITTEN
5516 033412 001450          BEQ 10$          ;THEN CONTINUE LOADING
5517          ;FAILURE LOADING FIRMWARE
5518 033414 010537 010170          MOV R5,TEMP2          ;SAVE CRAM ADDRESS
5519 033420 016437 177776 010172  MOV -2(R4),TEMP3          ;SAVE WORD LOADED
5520 033426 010237 010174          MOV R2,TEMP4          ;SAVE WORD READ BACK
5521 033432          PRINTF #FIRMLD          ;PRINT FAILURE MESSAGE
5522 033432 012746 053660          MOV #FIRMLD,-(SP)
5523 033436 012746 000001          MOV #1,-(SP)
5524 033442 010600          MOV SP,RO
5525 033444 104417          TRAP C$PNTF
5526 033446 062706 000004          ADD #4,SP
5527 033452          PRINTF #DATAHD          ;PRINT HEADER
5528 033452 012746 053720          MOV #DATAHD,-(SP)

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 156
INITIALIZE SECTION

```

5529 033456 012746 000001
5530 033462 010600
5531 033464 104417
5532 033466 062706 000004
5533 033472
5534 033472 013746 010174
5535 033476 013746 010172
5536 033502 013746 010170
5537 033506 012746 053771
5538 033512 012746 000004
5539 033516 010600
5540 033520 104417
5541 033522 062706 000012
5542 033526
5543 033526 104444
5544 033530
5545 033530 104422
5546 033532 000712
5547
5548
5549
5550
5551 033534 005205 10$: INC R5 ;CONTINUE LOADING FIRMWARE
5552 033536 022704 053660 CMP #DDCMPE,R4 ;UPDATE THE CRAM ADDRESS
5553 033542 001314 BNE 2$ ;TEST FOR DONE
5554
5555 033544 SETVEC CLKVEC,#CLKINT,#340 ;SETUP CLOCK VECTOR
5556 033544 012746 000340 MOV #340,-(SP)
5557 033550 012746 024022 MOV #CLKINT,-(SP)
5558 033554 013746 010254 MOV CLKVEC,-(SP)
5559 033560 012746 000003 MOV #3,-(SP)
5560 033564 104437 TRAP C$SVEC
5561 033566 062706 000010 ADD #10,SP
5562
5563 ;DEVICE DEPENDENT VECTOR SETUP
5564
5565 033572 SETVEC INVEC,#DVINS,INTPRI ;SETUP INPUT INTERRUPT VECTOR
5566 033572 013746 013416 MOV INTPRI,-(SP)
5567 033576 012746 067204 MOV #DVINS,-(SP)
5568 033602 013746 013412 MOV INVEC,-(SP)
5569 033606 012746 000003 MOV #3,-(SP)
5570 033612 104437 TRAP C$SVEC
5571 033614 062706 000010 ADD #10,SP
5572 033620 SETVEC OUTVEC,#DVOUTS,INTPRI ;SETUP OUTPUT INTERRUPT VECTOR
5573 033620 013746 013416 MOV INTPRI,-(SP)
5574 033624 012746 067214 MOV #DVOUTS,-(SP)
5575 033630 013746 013414 MOV OUTVEC,-(SP)
5576 033634 012746 000003 MOV #3,-(SP)
5577 033640 104437 TRAP C$SVEC
5578 033642 062706 000010 ADD #10,SP
5579
5580
5581 033646 ENDIT: SETPRI #PRI00 ;SET THE "RUN" PRIORITY TO 0
5582 033646
5583 033646 012700 000000 MOV #PRI00,RO
5584 033652 104441 TRAP C$SPRI

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 157
INITIALIZE SECTION

5585 033654
5586 033654 104432
5587 033656 020140
5588
5589

EXIT INIT

TRAP CSEXIT
.WORD L10015-

.NLIST BEX

033660

DDCMPS:

:KMS-11/BD DCLT MICRO CODE IMAGE

:0000 BLOCK # 1

```

033660 000434 004000 013400 .WORD ^0 434, ^0 4000, ^0 13400, ^0 402, ^0 63230, ^0 401, ^0 63231, ^0 777
033700 063232 120720 103015 .WORD ^0 63232, ^0120720, ^0103015, ^0 16400, ^0100411, ^0 10046, ^0 4017, ^0 16401
033720 016401 014000 016404 .WORD ^0 16401, ^0 14000, ^0 16404, ^0 16404, ^0 4017, ^0 10062, ^0 2777, ^0 407
033740 061233 100075 010061 .WORD ^0 61233, ^0100075, ^0 10061, ^0 4017, ^0 2777, ^0110574, ^0100534, ^0 410
033760 010050 004017 062620 .WORD ^0 10050, ^0 4017, ^0 62620, ^0110574, ^0 10041, ^0 4017, ^0 57225, ^0 62605
034000 100457 040365 101456 .WORD ^0100457, ^0 40365, ^0101456, ^0 3200, ^0 40360, ^0101057, ^0 2400, ^0 42231
034020 000404 061233 100000 .WORD ^0 404, ^0 61233, ^0100000, ^0110574, ^0100516, ^0 10042, ^0 4017, ^0 43225
034040 000400 060365 101474 .WORD ^0 400, ^0 60365, ^0101474, ^0100516, ^0 4002, ^0 10000, ^0 40371, ^0101516
034060 043621 004014 010000 .WORD ^0 43621, ^0 4014, ^0 10000, ^0 50011, ^0 40361, ^0101516, ^0120720, ^0 4014
034100 010000 052410 101110 .WORD ^0 10000, ^0 52410, ^0101110, ^0 76605, ^0 62601, ^0 65620, ^0 410, ^0 61233
034120 100000 110574 010061 .WORD ^0100000, ^0110574, ^0 10061, ^0 4017, ^0 42411, ^0 43220, ^0 530, ^0160400
034140 100436 100444 100464 .WORD ^0100436, ^0100444, ^0100464, ^0100432, ^0 10044, ^0 4017, ^0 57220, ^0 40360
034160 101542 100443 120620 .WORD ^0101542, ^0100443, ^0120620, ^0 3077, ^0103443, ^0 63677, ^0 61231, ^0 10050
034200 004017 043220 000554 .WORD ^0 4017, ^0 43220, ^0 554, ^0160400, ^0100570, ^0100645, ^0114672, ^0114672
034220 114672 100675 100737 .WORD ^0114672, ^0100675, ^0100737, ^0114672, ^0100746, ^0110502, ^0110515, ^0110570
034240 010051 004017 057220 .WORD ^0 10051, ^0 4017, ^0 57220, ^0 40360, ^0101605, ^0 3007, ^0 42407, ^0101201
034260 100602 002404 000405 .WORD ^0100602, ^0 2404, ^0 405, ^0100440, ^0 4017, ^0 10053, ^0 57220, ^0 40360
034300 101630 052410 004016 .WORD ^0101630, ^0 52410, ^0 4016, ^0 57222, ^0101604, ^0 43223, ^0 10067, ^0 4003
034320 016402 016400 076602 .WORD ^0 16402, ^0 16400, ^0 76602, ^0 16400, ^0 16400, ^0 76603, ^0 2401, ^0100437
034340 120400 103633 100443 .WORD ^0120400, ^0103633, ^0100443, ^0 3020, ^0 61202, ^0102237, ^0100643, ^0120620
034360 003200 063700 061231 .WORD ^0 3200, ^0 63700, ^0 61231, ^0 401, ^0100733, ^0120440, ^0103343, ^0 2410
034400 010067 004003 123441 .WORD ^0 10067, ^0 4003, ^0123441, ^0 3140, ^0 60660, ^0 61620, ^0 61620, ^0 61620
034420 061620 061620 076620 .WORD ^0 61620, ^0 61620, ^0 76620, ^0 3407, ^0 63261, ^0 76521, ^0122460, ^0 56667
034440 136520 136500 136560 .WORD ^0136520, ^0136500, ^0136560, ^0136540, ^0100443, ^0 10052, ^0 4017, ^0 50220
034460 004011 003016 057676 .WORD ^0 4011, ^0 3016, ^0 57676, ^0 61620, ^0 55223, ^0 55224, ^0 55225, ^0 55226
034500 055227 003003 043263 .WORD ^0 55227, ^0 3003, ^0 43263, ^0 63123, ^0 63123, ^0 63123, ^0 63123, ^0 63123
034520 060703 003200 061302 .WORD ^0 60703, ^0 3200, ^0 61302, ^0120400, ^0103326, ^0100732, ^0120620, ^0 3300
034540 063700 061231 000406 .WORD ^0 63700, ^0 61231, ^0 406, ^0 4017, ^0 10107, ^0 2410, ^0100440, ^0120440
034560 103743 002400 100443 .WORD ^0103743, ^0 2400, ^0100443, ^0 407, ^0 61233, ^0100203, ^0 2400, ^0 10067
034600 004003 057620 003003 .WORD ^0 4003, ^0 57620, ^0 3003, ^0 63263, ^0 54620, ^0 61620, ^0 3007, ^0 63267
034620 000762 160403 104403 .WORD ^0 762, ^0160403, ^0104403, ^0104572, ^0104730, ^0 417, ^0 61233, ^0100100
034640 004000 013400 003377 .WORD ^0 4000, ^0 13400, ^0 3377, ^0 16400, ^0 63077, ^0101777, ^0100773, ^0 16400

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 158
INITIALIZE SECTION

:1000 BLOCK # 2

034660	063077	105415	100777	.WORD	^0 63077,^0105415,^0100777,^0 405,^0164407,^0100770,^0104455,^0114672
034700	104504	104544	114672	.WORD	^0104504,^0104544,^0114672,^0104554,^0104563,^0 4003,^0 10100,^0 3177
034720	016777	063177	105424	.WORD	^0 16777,^0 63177,^0105424,^0104420,^0 10000,^0 4015,^0 16777,^0 63177
034740	105432	104426	010000	.WORD	^0105432,^0104426,^0 10000,^0 4007,^0 16777,^0 63177,^0105440,^0104434
034760	000600	061222	010100	.WORD	^0 600,^0 61222,^0 10100,^0 4004,^0 16777,^0 63177,^0105450,^0104444
035000	003314	016777	063174	.WORD	^0 3314,^0 16777,^0 63174,^0101723,^0104451,^0 401,^0 63236,^0 464
035020	063237	000417	061233	.WORD	^0 63237,^0 417,^0 61233,^0100230,^0 3375,^0 42675,^0 3021,^0 42701
035040	017277	042677	063127	.WORD	^0 17277,^0 42677,^0 63127,^0 63127,^0 63127,^0 63127,^0 500,^0 70007
035060	004003	016406	016700	.WORD	^0 4003,^0 16406,^0 16700,^0100443,^0 401,^0 63236,^0 513,^0 63237
035100	000417	061233	100230	.WORD	^0 417,^0 61233,^0100230,^0 43235,^0 3363,^0 56663,^0 43234,^0 63126
035120	063126	063126	000400	.WORD	^0 63126,^0 63126,^0 400,^0 73006,^0 4002,^0 60615,^0106130,^0104537
035140	060534	107537	040371	.WORD	^0 60534,^0107537,^0 40371,^0105535,^0100437,^0 16505,^0 2400,^0 500
035160	070006	002420	000412	.WORD	^0 70006,^0 2420,^0 412,^0100440,^0 4017,^0 10041,^0123140,^0 63120
035200	063120	063120	062520	.WORD	^0 63120,^0 63120,^0 62520,^0100443,^0 10055,^0 4017,^0122500,^0 56410
035220	136520	122560	100443	.WORD	^0136520,^0122560,^0100443,^0 10062,^0 4017,^0 16400,^0136500,^0136520
035240	002700	100443	000574	.WORD	^0 2700,^0100443,^0 574,^0164407,^0104622,^0104610,^0 401,^0 63236
035260	000605	063237	000417	.WORD	^0 605,^0 63237,^0 417,^0 61233,^0100230,^0 3376,^0 42676,^0100443
035300	000401	063236	000617	.WORD	^0 401,^0 63236,^0 617,^0 63237,^0 417,^0 61233,^0100230,^0 3100
035320	042700	100443	000401	.WORD	^0 42700,^0100443,^0 401,^0 63236,^0 631,^0 63237,^0 417,^0 61233
035340	100230	057233	002400	.WORD	^0100230,^0 57233,^0 2400,^0 63126,^0 63126,^0 63126,^0 63526,^0 62231
035360	000661	062232	000557	.WORD	^0 661,^0 62232,^0 557,^0 62230,^0 0,^0 620,^0 62232,^0 626
035400	062230	114705	062234	.WORD	^0 62230,^0114705,^0 62234,^0 560,^0 62232,^0 402,^0 62230,^0 10062
035420	004017	056621	101443	.WORD	^0 4017,^0 56621,^0101443,^0 57221,^0 57222,^0 43223,^0 401,^0 63236
035440	000675	063237	000414	.WORD	^0 675,^0 63237,^0 414,^0 61233,^0100000,^0 16400,^0 76607,^0 76602
035460	076601	002777	000401	.WORD	^0 76601,^0 2777,^0 401,^0 63236,^0 711,^0 63237,^0 414,^0 61233
035500	100000	016400	063533	.WORD	^0100000,^0 16400,^0 63533,^0107716,^0 16407,^0104717,^0 16403,^0 402
035520	063001	105323	104724	.WORD	^0 63001,^0105323,^0104724,^0 63062,^0 76602,^0 76601,^0 76603,^0100443
035540	000732	164407	104734	.WORD	^0 732,^0164407,^0104734,^0110425,^0 14000,^0 14000,^0 14000,^0 57233
035560	057234	003007	063667	.WORD	^0 57234,^0 3007,^0 63667,^0 63623,^0 4003,^0 10071,^0 401,^0 63236
035600	000755	063237	000417	.WORD	^0 755,^0 63237,^0 417,^0 61233,^0100230,^0 54620,^0106376,^0 401
035620	063226	000766	063227	.WORD	^0 63226,^0 766,^0 63227,^0 407,^0 61233,^0100312,^0 16416,^0 76603
035640	016400	016400	076574	.WORD	^0 16400,^0 16400,^0 76574,^0 76613,^0 2601,^0100443,^0 43236,^0 60536

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 159
INITIALIZE SECTION

:2000 BLOCK # 3

035660	107757	063127	063127	.WORD	^0107757	^0 63127	^0 63127	^0 63127	^0 400	^0 70407	^0 4002	^0 40371
035700	111412	100437	016450	.WORD	^0111412	^0100437	^0 16450	^0 76613	^0 62614	^0 4003	^0 10075	^0 40371
035720	101443	042412	010074	.WORD	^0101443	^0 42412	^0 10074	^0 42411	^0100437	^0 3007	^0 57667	^0 77220
035740	014000	057223	000402	.WORD	^0 14000	^0 57223	^0 402	^0 63236	^0 441	^0 63237	^0 413	^0 61233
035760	100075	123304	137337	.WORD	^0100075	^0123304	^0137337	^0 54220	^0111446	^0110450	^0 40220	^0111466
036000	000510	070004	111054	.WORD	^0 510	^0 70004	^0111054	^0110455	^0 64077	^0 14000	^0 54220	^0111461
036020	110463	040220	111466	.WORD	^0110463	^0 40220	^0111466	^0 646	^0 61222	^0100723	^0 402	^0 63236
036040	000475	063237	000414	.WORD	^0 475	^0 63237	^0 414	^0 61233	^0100021	^0 620	^0 76707	^0 62603
036060	000411	100440	010043	.WORD	^0 411	^0100440	^0 10043	^0 40012	^0111506	^0100443	^0 10044	^0 57220
036100	040360	111513	100443	.WORD	^0 40360	^0111513	^0100443	^0 400	^0100440	^0 4017	^0 10043	^0 40012
036120	111522	100443	010044	.WORD	^0111522	^0100443	^0 10044	^0 4017	^0 57220	^0 40360	^0111530	^0100443
036140	014000	057220	040360	.WORD	^0 14000	^0 57220	^0 40360	^0111535	^0100443	^0 10071	^0 4003	^0 43220
036160	000402	063236	000547	.WORD	^0 402	^0 63236	^0 547	^0 63237	^0 413	^0 61233	^0100075	^0123306
036200	123327	016777	016777	.WORD	^0123327	^0 16777	^0 16777	^0 16777	^0 16777	^0 510	^0 70006	^0111161
036220	110562	064067	016777	.WORD	^0110562	^0 64067	^0 16777	^0 16777	^0 16777	^0 16777	^0 413	^0100440
036240	002400	000602	061222	.WORD	^0 2400	^0 602	^0 61222	^0100723	^0 10043	^0 4017	^0 43220	^0 601
036260	170400	110626	110653	.WORD	^0170400	^0110626	^0110653	^0110664	^0110653	^0110715	^0110653	^0114446
036300	114674	110653	114535	.WORD	^0114674	^0110653	^0114535	^0110653	^0114556	^0114674	^0110653	^0114535
036320	110653	114556	110653	.WORD	^0110653	^0114556	^0110653	^0114640	^0110653	^0114655	^0 10046	^0 4017
036340	057220	040360	111643	.WORD	^0 57220	^0 40360	^0111643	^0 3005	^0 42405	^0111237	^0110640	^0 2401
036360	010043	002401	100522	.WORD	^0 10043	^0 2401	^0100522	^0 10044	^0 57220	^0 40360	^0101522	^0 42410
036400	010043	002403	100522	.WORD	^0 10043	^0 2403	^0100522	^0120600	^0102122	^0 42411	^0123620	^0112261
036420	100522	000554	061271	.WORD	^0100522	^0 554	^0 61271	^0110663	^0120620	^0103522	^0 2400	^0 10047
036440	004017	050220	004006	.WORD	^0 4017	^0 50220	^0 4006	^0 56223	^0 56222	^0 56227	^0 42226	^0 56711
036460	003300	040660	061620	.WORD	^0 3300	^0 40660	^0 61620	^0 61620	^0 61620	^0 61620	^0 61231	^0 3421
036500	114676	040220	101522	.WORD	^0114676	^0 40220	^0101522	^0 4017	^0110643	^0120620	^0103522	^0 2405
036520	010045	050220	004010	.WORD	^0 10045	^0 50220	^0 4010	^0 3007	^0 43667	^0 63220	^0 57621	^0113375
036540	060521	117430	060601	.WORD	^0 60521	^0117430	^0 60601	^0113775	^0 402	^0 63236	^0 743	^0 63237
036560	000413	061233	100146	.WORD	^0 413	^0 61233	^0100146	^0 14000	^0 0	^0120700	^0 56225	^0 42224
036600	042410	111353	110763	.WORD	^0 42410	^0111353	^0110763	^0 70220	^0 56411	^0111357	^0110763	^0 14000
036620	003100	042400	110764	.WORD	^0 3100	^0 42400	^0110764	^0 14000	^0 3300	^0 43660	^0 61620	^0 61620
036640	061620	061620	060711	.WORD	^0 61620	^0 61620	^0 60711	^0 61230	^0114702	^0 43221	^0 63161	^0 3400

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 160
INITIALIZE SECTION

:3000 BLOCK # 4

036660	063222	063121	063142	.WORD	^0	63222	^0	63121	^0	63142	^0	63121	^0	63142	^0	63121	^0	63142	^0	10043
036700	004017	002410	010055	.WORD	^0	4017	^0	2410	^0	10055	^0	57001	^0	57022	^0	0115017	^0	0114420	^0	3100
036720	057000	010000	062601	.WORD	^0	57000	^0	10000	^0	62601	^0	56224	^0	62602	^0	56225	^0	62600	^0	0110764
036740	004017	010043	002421	.WORD	^0	4017	^0	10043	^0	2421	^0	62202	^0	400	^0	62223	^0	10063	^0	56226
036760	056227	000414	061231	.WORD	^0	56227	^0	414	^0	61231	^0	421	^0	61230	^0	0114702	^0	2400	^0	10045
037000	004017	050220	004010	.WORD	^0	4017	^0	50220	^0	4010	^0	43220	^0	43221	^0	63120	^0	63120	^0	63120
037020	000407	063000	000400	.WORD	^0	407	^0	63000	^0	400	^0	70000	^0	4002	^0	43227	^0	43220	^0	42410
037040	042410	000417	063222	.WORD	^0	42410	^0	417	^0	63222	^0	42662	^0	476	^0	0174401	^0	0114506	^0	0114510
037060	114512	114514	114516	.WORD	^0	114512	^0	0114514	^0	0114516	^0	0114520	^0	0114522	^0	0114524	^0	10200	^0	0114525
037100	010220	114525	010240	.WORD	^0	10220	^0	0114525	^0	10240	^0	0114525	^0	10260	^0	0114525	^0	10300	^0	0114525
037120	010320	114525	010340	.WORD	^0	10320	^0	0114525	^0	10340	^0	0114525	^0	10360	^0	4002	^0	0120700	^0	70307
037140	016400	036400	016400	.WORD	^0	16400	^0	36400	^0	16400	^0	36420	^0	0100522	^0	42411	^0	10003	^0	36400
037160	022420	010000	056410	.WORD	^0	22420	^0	10000	^0	56410	^0	0115145	^0	0114552	^0	56411	^0	0115150	^0	0114552
037200	003100	042400	010000	.WORD	^0	3100	^0	42400	^0	10000	^0	56224	^0	56225	^0	0110764	^0	2400	^0	10003
037220	057221	057222	023003	.WORD	^0	57221	^0	57222	^0	23003	^0	23024	^0	10045	^0	50220	^0	4010	^0	3007
037240	043667	063220	054620	.WORD	^0	43667	^0	63220	^0	54620	^0	43225	^0	0117205	^0	403	^0	63236	^0	604
037260	063237	000413	061233	.WORD	^0	63237	^0	413	^0	61233	^0	0100146	^0	0114632	^0	403	^0	63236	^0	614
037300	063237	000413	061233	.WORD	^0	63237	^0	413	^0	61233	^0	0100075	^0	0137306	^0	0123327	^0	54220	^0	0115621
037320	114623	040220	115631	.WORD	^0	114623	^0	40220	^0	115631	^0	510	^0	70006	^0	0115227	^0	0114632	^0	64067
037340	114632	070206	076605	.WORD	^0	114632	^0	70206	^0	76605	^0	76602	^0	76601	^0	76604	^0	76603	^0	0100522
037360	002423	010063	054410	.WORD	^0	2423	^0	10063	^0	54410	^0	0115247	^0	62224	^0	42225	^0	0114652	^0	62224
037400	040411	062225	000415	.WORD	^0	40411	^0	62225	^0	415	^0	61230	^0	0114702	^0	2400	^0	10045	^0	50220
037420	004010	003007	043267	.WORD	^0	4010	^0	3007	^0	43267	^0	4017	^0	466	^0	60407	^0	70007	^0	16400
037440	022400	100522	000400	.WORD	^0	22400	^0	0100522	^0	400	^0	0100440	^0	2400	^0	0100522	^0	61230	^0	0120600
037460	116277	110711	120600	.WORD	^0	116277	^0	0110711	^0	120600	^0	0116302	^0	0100522	^0	10326	^0	4017	^0	421
037500	040220	115713	000621	.WORD	^0	40220	^0	0115713	^0	621	^0	0104652	^0	0	^0	0	^0	0	^0	0
037520	000000	000000	000000	.WORD	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0
037540	000000	000000	000000	.WORD	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0
037560	000000	000000	000000	.WORD	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0
037600	000000	000000	000000	.WORD	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0	^0	0
037620	001161	114761	001141	.WORD	^0	1161	^0	0114761	^0	1141	^0	0114763	^0	1121	^0	0114765	^0	1101	^0	0114767
037640	001061	114771	001041	.WORD	^0	1061	^0	0114771	^0	1041	^0	0114773	^0	1021	^0	0114775	^0	1001	^0	0114777

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 161
INITIALIZE SECTION

:4000 BLOCK # 5

037660	043625	061620	003100	.WORD	^0 43625,^0 61620,^0 3100,^0 70000,^0 4002,^0 43220,^0 410,^0160400
037700	100437	100602	100602	.WORD	^0100437,^0100602,^0100602,^0100602,^0100602,^0100602,^0100620,^0100644
037720	104443	100421	104613	.WORD	^0104443,^0100421,^0104613,^0104613,^0104750,^0110431,^0110556,^0110703
037740	114410	000000	000000	.WORD	^0114410,^0 0,^0 0,^0 0,^0 400,^0 61233,^0100063,^0123315
037760	000404	063236	000447	.WORD	^0 404,^0 63236,^0 447,^0 63237,^0 417,^0 61233,^0100222,^0 40620
040000	102052	100431	020660	.WORD	^0102052,^0100431,^0 20660,^0102055,^0100434,^0 400,^0 62232,^0123316
040020	000412	070016	016400	.WORD	^0 412,^0 70016,^0 16400,^0 2400,^0 23200,^0121273,^0123244,^0114423
040040	004002	070215	016401	.WORD	^0 4002,^0 70215,^0 16401,^0 3201,^0 62600,^0 40361,^0101505,^0 3005
040060	040365	101434	070215	.WORD	^0 40365,^0101434,^0 70215,^0 2420,^0100434,^0 507,^0160407,^0100543
040100	100540	100535	100532	.WORD	^0100540,^0100535,^0100532,^0100527,^0100524,^0100521,^0 10020,^0 4005
040120	100545	010007	004005	.WORD	^0100545,^0 10007,^0 4005,^0100545,^0 10376,^0 4004,^0100545,^0 10365
040140	004004	100545	010354	.WORD	^0 4004,^0100545,^0 10354,^0 4004,^0100545,^0 10343,^0 4004,^0100545
040160	010332	004004	100545	.WORD	^0 10332,^0 4004,^0100545,^0 10321,^0 4004,^0 54220,^0101550,^0100434
040200	040220	101553	100434	.WORD	^0 40220,^0101553,^0100434,^0 60607,^0 63221,^0 64217,^0 402,^0 70016
040220	042412	101563	100434	.WORD	^0 42412,^0101563,^0100434,^0 404,^0 63226,^0 572,^0 63227,^0 407
040240	061233	100312	016402	.WORD	^0 61233,^0100312,^0 16402,^0 76601,^0 16400,^0 16400,^0 16400,^0 16400
040260	002602	100434	020660	.WORD	^0 2602,^0100434,^0 20660,^0102205,^0100434,^0 56411,^0 400,^0 62232
040300	120700	070000	022600	.WORD	^0120700,^0 70000,^0 22600,^0 43220,^0121273,^0123244,^0114423,^0100434
040320	020660	102223	100434	.WORD	^0 20660,^0102223,^0100434,^0123314,^0 56411,^0 400,^0 62232,^0 3000
040340	121273	123244	114423	.WORD	^0121273,^0123244,^0114423,^0121273,^0123244,^0114423,^0 64217,^0 70216
040360	023200	040360	101434	.WORD	^0 23200,^0 40360,^0101434,^0100670,^0 20660,^0102247,^0100434,^0 400
040400	062232	123314	000404	.WORD	^0 62232,^0123314,^0 404,^0 63236,^0 661,^0 63237,^0 417,^0 61233
040420	100222	123316	123337	.WORD	^0100222,^0123316,^0123337,^0 413,^0 70016,^0 23200,^0 40360,^0101774
040440	004002	070214	002420	.WORD	^0 4002,^0 70214,^0 2420,^0 401,^0110743,^0 22700,^0 776,^0 63233
040460	042673	042234	003001	.WORD	^0 42673,^0 42234,^0 3001,^0 42701,^0 42234,^0 70214,^0 16400,^0 14000
040500	057224	043233	000407	.WORD	^0 57224,^0 43233,^0 407,^0 77264,^0 57222,^0 43223,^0 64217,^0 70216
040520	003200	056700	000724	.WORD	^0 3200,^0 56700,^0 724,^0160404,^0100772,^0100760,^0100746,^0100743
040540	100772	100772	100735	.WORD	^0100772,^0100772,^0100735,^0 2440,^0100736,^0 2401,^0 4013,^0 410
040560	070006	002400	100434	.WORD	^0 70006,^0 2400,^0100434,^0 16404,^0 62603,^0100434,^0 16402,^0 62602
040600	004013	000400	070006	.WORD	^0 4013,^0 400,^0 70006,^0 477,^0 62673,^0100434,^0 70220,^0100761
040620	120700	016420	076602	.WORD	^0120700,^0 16420,^0 76602,^0 14000,^0 40362,^0101767,^0100736,^0 70220
040640	002400	100434	000421	.WORD	^0 2400,^0100434,^0 421,^0110743,^0 4002,^0 70214,^0 16410,^0 405

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 162
INITIALIZE SECTION

:5000 BLOCK # 6

040660	063233	040373	101675	.WORD	^0 63233,^0 40373,^0101675,^0 3201,^0 40361,^0105411,^0 70214,^0 2420
040700	100434	014000	014000	.WORD	^0100434,^0 14000,^0 14000,^0 14000,^0 57222,^0 43223,^0 63163,^0 64217
040720	070216	040620	106437	.WORD	^0 70216,^0 40620,^0106437,^0 410,^0 63227,^0 42707,^0 54620,^0107037
040740	120700	014000	014000	.WORD	^0120700,^0 14000,^0 14000,^0 14000,^0 14000,^0 40363,^0101756,^0 4002
040760	070214	002420	100434	.WORD	^0 70214,^0 2420,^0100434,^0 20660,^0106046,^0100434,^0 2412,^0123314
041000	000405	063236	000457	.WORD	^0 405,^0 63236,^0 457,^0 63237,^0 413,^0 61233,^0100067,^0123316
041020	137337	054220	105464	.WORD	^0137337,^0 54220,^0105464,^0104471,^0 40220,^0105467,^0104471,^0 410
041040	104541	004002	000402	.WORD	^0104541,^0 4002,^0 402,^0 70014,^0 57221,^0 43222,^0 477,^0 63262
041060	064217	070216	014000	.WORD	^0 64217,^0 70216,^0 14000,^0 14000,^0 14000,^0 57226,^0 63266,^0 43224
041100	060164	105513	104515	.WORD	^0 60164,^0105513,^0104515,^0 60166,^0105545,^0 64217,^0 70216,^0 14000
041120	014000	014000	014000	.WORD	^0 14000,^0 14000,^0 14000,^0 14000,^0 0,^0123313,^0123327,^0 76601
041140	062602	070213	064207	.WORD	^0 62602,^0 70213,^0 64207,^0 56744,^0 42446,^0 40620,^0107540,^0104545
041160	000420	004002	070214	.WORD	^0 420,^0 4002,^0 70214,^0 2420,^0110743,^0 64217,^0 70076,^0 57235
041200	057233	057226	076602	.WORD	^0 57233,^0 57226,^0 76602,^0 76601,^0 16400,^0 2400,^0 405,^0 63236
041220	000565	063237	000407	.WORD	^0 565,^0 63237,^0 407,^0 61233,^0100256,^0 76613,^0 76615,^0 62606
041240	000405	063236	000577	.WORD	^0 405,^0 63236,^0 577,^0 63237,^0 417,^0 61233,^0100222,^0123316
041260	000412	070016	016400	.WORD	^0 412,^0 70016,^0 16400,^0 16400,^0 2400,^0 4002,^0 407,^0 70014
041300	062604	070214	104616	.WORD	^0 62604,^0 70214,^0104616,^0 20660,^0106216,^0100434,^0 400,^0 62232
041320	043233	123314	042731	.WORD	^0 43233,^0123314,^0 42731,^0 23200,^0121273,^0123244,^0114423,^0 60600
041340	063223	000405	063236	.WORD	^0 63223,^0 405,^0 63236,^0 640,^0 63237,^0 413,^0 61233,^0100067
041360	014000	014000	014000	.WORD	^0 14000,^0 14000,^0 14000,^0 14000,^0 57224,^0 57222,^0 0,^0137307
041400	042411	105253	104655	.WORD	^0 42411,^0105253,^0104655,^0 70207,^0 56411,^0 40362,^0105660,^0104664
041420	070207	040364	105671	.WORD	^0 70207,^0 40364,^0105671,^0 14000,^0 14000,^0 60613,^0106306,^0 62603
041440	100434	123322	070214	.WORD	^0100434,^0123322,^0 70214,^0 4002,^0 2414,^0 70207,^0 64202,^0 14000
041460	014000	060613	106306	.WORD	^0 14000,^0 60613,^0106306,^0 400,^0 62603,^0 63223,^0 43227,^0 405
041500	063236	000716	063237	.WORD	^0 63236,^0 716,^0 63237,^0 407,^0 61233,^0100256,^0120700,^0 57225
041520	057224	043226	070220	.WORD	^0 57224,^0 43226,^0 70220,^0 56410,^0105326,^0104733,^0 56411,^0105331
041540	104733	003100	042400	.WORD	^0104733,^0 3100,^0 42400,^0 405,^0 63236,^0 742,^0 63237,^0 414
041560	061233	100000	076603	.WORD	^0 61233,^0100000,^0 76603,^0 76607,^0 76604,^0 76605,^0 76606,^0100434
041600	020660	106377	100434	.WORD	^0 20660,^0106377,^0100434,^0 4002,^0 70213,^0 2420,^0 407,^0 70013
041620	004002	043233	000405	.WORD	^0 4002,^0 43233,^0 405,^0 63236,^0 771,^0 63237,^0 413,^0 61233
041640	100067	014000	014000	.WORD	^0100067,^0 14000,^0 14000,^0 14000,^0 14000,^0 62613,^0100434,^0 2415

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 163
INITIALIZE SECTION

:6000 BLOCK # 7

041660	123313	000406	063236	.WORD	*0123313,^0	406,^0	63236,^0	410,^0	63237,^0	417,^0	61233,^0	0100222
041700	040620	107353	003000	.WORD	*0 40620,^0	107353,^0	3000,^0	0121273,^0	0123244,^0	0114423,^0	0121273,^0	0123244
041720	114423	000400	062232	.WORD	*0114423,^0	400,^0	62232,^0	64217,^0	70216,^0	23203,^0	40363,^0	0101434
041740	110454	020660	112034	.WORD	*0110454,^0	20660,^0	0112034,^0	0100434,^0	400,^0	62232,^0	0123713,^0	406
041760	063236	000446	063237	.WORD	*0 63236,^0	446,^0	63237,^0	417,^0	61233,^0	0100222,^0	0123300,^0	413
042000	070000	023200	040360	.WORD	*0 70000,^0	23200,^0	40360,^0	0111460,^0	4002,^0	70213,^0	2417,^0	0100434
042020	020700	003376	060676	.WORD	*0 20700,^0	3376,^0	60676,^0	62234,^0	3001,^0	60701,^0	0,^0	0
042040	062234	000406	063236	.WORD	*0 62234,^0	406,^0	63236,^0	500,^0	63237,^0	413,^0	61233,^0	0100067
042060	054412	077221	017300	.WORD	*0 54412,^0	77221,^0	17300,^0	54660,^0	77223,^0	14000,^0	57303,^0	43227
042100	003400	063235	063121	.WORD	*0 3400,^0	63235,^0	63121,^0	63155,^0	63121,^0	63155,^0	63121,^0	63155
042120	063061	063061	004017	.WORD	*0 63061,^0	63061,^0	4017,^0	10055,^0	57001,^0	57035,^0	0111130,^0	0110531
042140	003100	043400	063222	.WORD	*0 3100,^0	43400,^0	63222,^0	406,^0	63236,^0	542,^0	63237,^0	414
042160	061233	100000	076603	.WORD	*0 61233,^0	0100000,^0	76603,^0	76607,^0	76615,^0	0,^0	0120700,^0	76601
042200	076602	070213	004002	.WORD	*0 76602,^0	70213,^0	4002,^0	16416,^0	62620,^0	0100434,^0	0137300,^0	50220
042220	004006	040620	112164	.WORD	*0 4006,^0	40620,^0	0112164,^0	0100434,^0	70200,^0	4002,^0	16400,^0	17200
042240	057660	077221	000406	.WORD	*0 57660,^0	77221,^0	406,^0	63236,^0	601,^0	63237,^0	413,^0	61233
042260	100067	123337	123316	.WORD	*0100067,^0	0123337,^0	0123316,^0	510,^0	70016,^0	0111207,^0	0110610,^0	64077
042300	043233	016777	043223	.WORD	*0 43233,^0	16777,^0	43223,^0	16777,^0	43224,^0	16777,^0	43226,^0	16777
042320	043227	016777	043220	.WORD	*0 43227,^0	16777,^0	43220,^0	2777,^0	70216,^0	64217,^0	43222,^0	76613
042340	076603	076604	076606	.WORD	*0 76603,^0	76604,^0	76606,^0	76607,^0	62600,^0	4003,^0	500,^0	70005
042360	016401	002400	000406	.WORD	*0 16401,^0	2400,^0	406,^0	63236,^0	651,^0	63237,^0	417,^0	61233
042400	100222	120700	003167	.WORD	*0100222,^0	0120700,^0	3167,^0	42667,^0	56701,^0	2410,^0	3005,^0	70005
042420	042411	043221	063606	.WORD	*0 42411,^0	43221,^0	63606,^0	63225,^0	406,^0	63226,^0	673,^0	63227
042440	000407	061233	100312	.WORD	*0 407,^0	61233,^0	0100312,^0	16400,^0	76605,^0	16400,^0	16400,^0	76601
042460	076602	002602	100434	.WORD	*0 76602,^0	2602,^0	0100434,^0	0123313,^0	16420,^0	14000,^0	17200,^0	43260
042500	000406	063236	000717	.WORD	*0 406,^0	63236,^0	717,^0	63237,^0	417,^0	61233,^0	0100222,^0	3177
042520	042677	042700	000407	.WORD	*0 42677,^0	42700,^0	407,^0	70013,^0	4002,^0	5233,^0	406,^0	63236
042540	000735	063237	000413	.WORD	*0 735,^0	63237,^0	413,^0	61233,^0	0100067,^0	14000,^0	14000,^0	14000
042560	014000	062613	000402	.WORD	*0 14000,^0	62613,^0	402,^0	63224,^0	4003,^0	500,^0	70005,^0	16402
042600	062604	000406	063236	.WORD	*0 62604,^0	406,^0	63236,^0	760,^0	63237,^0	417,^0	61233,^0	0100222
042620	137304	016410	000407	.WORD	*0137304,^0	16410,^0	407,^0	70004,^0	42411,^0	0111767,^0	0100434,^0	60606
042640	063224	000407	063226	.WORD	*0 63224,^0	407,^0	63226,^0	400,^0	63227,^0	407,^0	61233,^0	0100312

CZKMSAO KMS11-BD/BE DCIT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 164
INITIALIZE SECTION

:7000 BLOCK # 8

042660	016406	076604	016400	.WORD	^0 16406	^0 76604	^0 16400	^0 16400	^0 16400	^0 16400	^0 16400	^0 16601	^0 100434
042700	022700	003376	042676	.WORD	^0 22700	^0 3376	^0 42676	^0 42234	^0 3001	^0 0	^0 0	^0 0	^0 42701
042720	042234	002400	100434	.WORD	^0 42234	^0 2400	^0 100434	^0 60600	^0 63223	^0 60525	^0 63236	^0 400	
042740	063237	063136	063157	.WORD	^0 63237	^0 63136	^0 63157	^0 64217	^0 412	^0 73016	^0 57621	^0 43222	
042760	063226	000417	063666	.WORD	^0 63226	^0 417	^0 63666	^0 63126	^0 63006	^0 4017	^0 600	^0 70006	
043000	057721	057322	043323	.WORD	^0 57721	^0 57322	^0 43323	^0 417	^0 63121	^0 63141	^0 63141	^0 63141	
043020	063141	063261	063121	.WORD	^0 63141	^0 63261	^0 63121	^0 660	^0 70001	^0 57322	^0 43323	^0 64217	
043040	070216	076602	062603	.WORD	^0 70216	^0 76602	^0 62603	^0 63064	^0 160064	^0 660	^0 63225	^0 600	
043060	063226	000400	063224	.WORD	^0 63226	^0 400	^0 63224	^0 4017	^0 10320	^0 16400	^0 16400	^0 16400	
043100	016402	016500	016401	.WORD	^0 16402	^0 16500	^0 16401	^0 60604	^0 63221	^0 400	^0 63220	^0 63222	
043120	063223	060604	117177	.WORD	^0 63223	^0 60604	^0 117177	^0 60603	^0 60721	^0 116130	^0 10320	^0 114531	
043140	010323	061611	063221	.WORD	^0 10323	^0 61611	^0 63221	^0 54723	^0 61602	^0 63223	^0 54722	^0 41620	
043160	063222	060601	117144	.WORD	^0 63222	^0 60601	^0 117144	^0 114523	^0 63064	^0 70205	^0 76603	^0 76602	
043200	063221	123305	070206	.WORD	^0 63221	^0 123305	^0 70206	^0 60602	^0 61622	^0 61622	^0 61622	^0 61622	
043220	063222	060603	061623	.WORD	^0 63222	^0 60603	^0 61623	^0 61623	^0 61623	^0 61623	^0 63223	^0 760	
043240	076663	076662	000417	.WORD	^0 76663	^0 76662	^0 417	^0 76662	^0 400	^0 123306	^0 114513	^0 61202	
043260	000400	061233	100032	.WORD	^0 400	^0 61233	^0 100032	^0 40620	^0 4017	^0 10106	^0 56411	^0 115611	
043300	114653	042411	115614	.WORD	^0 114653	^0 42411	^0 115614	^0 114653	^0 116624	^0 123001	^0 577	^0 60661	
043320	061220	000406	063221	.WORD	^0 61220	^0 406	^0 63221	^0 114626	^0 410	^0 63221	^0 123042	^0 557	
043340	060662	061222	000407	.WORD	^0 60662	^0 61222	^0 407	^0 63226	^0 641	^0 63227	^0 407	^0 61233	
043360	100312	076601	136460	.WORD	^0 100312	^0 76601	^0 136460	^0 136500	^0 136520	^0 136540	^0 136560	^0 2603	
043400	004017	010050	002400	.WORD	^0 4017	^0 10050	^0 2400	^0 400	^0 61233	^0 100043	^0 10042	^0 4017	
043420	040620	061620	061620	.WORD	^0 40620	^0 61620	^0 61620	^0 61620	^0 63220	^0 667	^0 174400	^0 10140	
043440	114706	010143	114706	.WORD	^0 114706	^0 10143	^0 114706	^0 10146	^0 114706	^0 10151	^0 114706	^0 10154	
043460	114706	010157	114706	.WORD	^0 114706	^0 10157	^0 114706	^0 10162	^0 114706	^0 10165	^0 4005	^0 60616	
043500	061233	160217	010051	.WORD	^0 61233	^0 160217	^0 10051	^0 4017	^0 407	^0 63220	^0 42400	^0 115321	
043520	114722	002404	057220	.WORD	^0 114722	^0 2404	^0 57220	^0 40360	^0 115732	^0 70200	^0 4011	^0 60606	
043540	061233	160207	114732	.WORD	^0 61233	^0 160207	^0 114732	^0 0	^0 0	^0 0	^0 0	^0 0	
043560	000000	000000	000000	.WORD	^0 0	^0 0	^0 0	^0 0	^0 0	^0 0	^0 0	^0 0	
043600	000000	000000	000000	.WORD	^0 0	^0 0	^0 0	^0 0	^0 0	^0 0	^0 0	^0 0	
043620	001161	114761	001141	.WORD	^0 1161	^0 114761	^0 1141	^0 114763	^0 1121	^0 114765	^0 1101	^0 114767	
043640	001061	114771	001041	.WORD	^0 1061	^0 114771	^0 1041	^0 114773	^0 1021	^0 114775	^0 1001	^0 114777	

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 165
INITIALIZE SECTION

:10000 BLOCK # 9

043660	010042	004017	042231	.WORD	^0 10042,^0 4017,^0 42231,^0 43625,^0 61620,^0 70225,^0 61620,^0 61620
043700	061620	063220	004002	.WORD	^0 61620,^0 63220,^0 4002,^0 43224,^0 420,^0 60364,^0 0101021,^0 0101424
043720	100527	000504	060364	.WORD	^0 0100527,^0 504,^0 60364,^0 0101127,^0 0123301,^0 4012,^0 60520,^0 63222
043740	073122	040364	101437	.WORD	^0 73122,^0 40364,^0 0101437,^0 76604,^0 16400,^0 16400,^0 0100445,^0 14000
043760	056411	042431	042412	.WORD	^0 56411,^0 42431,^0 42412,^0 40620,^0 0103050,^0 70201,^0 4002,^0 0100527
044000	070202	002400	000410	.WORD	^0 70202,^0 2400,^0 410,^0 63236,^0 461,^0 63237,^0 417,^0 61233
044020	100222	043233	060533	.WORD	^0 0100222,^0 43233,^0 60533,^0 3173,^0 0103466,^0 3373,^0 42673,^0 57236
044040	003100	042700	057235	.WORD	^0 3100,^0 42700,^0 57235,^0 57234,^0 57233,^0 60606,^0 63222,^0 410
044060	063226	000506	063227	.WORD	^0 63226,^0 506,^0 63227,^0 407,^0 61233,^0 0100312,^0 16410,^0 76602
044100	076616	076615	076614	.WORD	^0 76616,^0 76615,^0 76614,^0 76613,^0 2601,^0 4002,^0 70201,^0 447
044120	060364	101125	016507	.WORD	^0 60364,^0 0101125,^0 16507,^0 2400,^0 0100644,^0 2504,^0 0100644,^0 531
044140	160404	100647	100727	.WORD	^0 0160404,^0 0100647,^0 0100727,^0 0104433,^0 0100534,^0 0100535,^0 0100536,^0 0104442
044160	104516	100541	104702	.WORD	^0 0104516,^0 0100541,^0 0104702,^0 0104730,^0 0110415,^0 0100545,^0 0100546,^0 0100547
044200	100550	110467	110556	.WORD	^0 0100550,^0 0110467,^0 0110556,^0 0110575,^0 0100554,^0 0110646,^0 0110661,^0 0110722
044220	100560	100561	100562	.WORD	^0 0100560,^0 0100561,^0 0100562,^0 0110767,^0 0110767,^0 0110767,^0 0110767,^0 0110767
044240	114401	100641	100641	.WORD	^0 0114401,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641
044260	100641	100641	100641	.WORD	^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641
044300	110661	110722	100612	.WORD	^0 0110661,^0 0110722,^0 0100612,^0 0100613,^0 0100614,^0 0110767,^0 0110767,^0 0110767
044320	110767	110767	114401	.WORD	^0 0110767,^0 0110767,^0 0114401,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641
044340	100641	100641	100641	.WORD	^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641,^0 0100641
044360	100641	000414	061233	.WORD	^0 0100641,^0 414,^0 61233,^0 0100037,^0 400,^0 61233,^0 0100121,^0 2401
044400	123302	000410	063236	.WORD	^0 0123302,^0 410,^0 63236,^0 660,^0 63237,^0 417,^0 61233,^0 0100222
044420	040620	102263	100644	.WORD	^0 40620,^0 0102263,^0 0100644,^0 0123301,^0 0137323,^0 40012,^0 0101644,^0 43224
044440	060524	103644	016400	.WORD	^0 60524,^0 0103644,^0 16400,^0 57225,^0 70202,^0 4002,^0 63604,^0 0102712
044460	103310	102314	061620	.WORD	^0 0103310,^0 0102314,^0 61620,^0 0102720,^0 0103316,^0 61620,^0 0102725,^0 0100644
044500	016402	100721	016406	.WORD	^0 16402,^0 0100721,^0 16406,^0 0100721,^0 16412,^0 0100721,^0 16413,^0 0100721
044520	016411	076605	076601	.WORD	^0 16411,^0 76605,^0 76601,^0 76603,^0 0100644,^0 16420,^0 0100644,^0 2400
044540	123302	000410	063236	.WORD	^0 0123302,^0 410,^0 63236,^0 740,^0 63237,^0 417,^0 61233,^0 0100222
044560	123303	123324	057220	.WORD	^0 0123303,^0 0123324,^0 57220,^0 43227,^0 60527,^0 0103644,^0 60600,^0 0103365
044600	000410	070003	054220	.WORD	^0 410,^0 70003,^0 54220,^0 0101755,^0 0100757,^0 40220,^0 0101765,^0 70003
044620	056411	042431	042412	.WORD	^0 56411,^0 42431,^0 42412,^0 40620,^0 0107005,^0 70062,^0 4002,^0 56411
044640	042431	042412	040620	.WORD	^0 42431,^0 42412,^0 40620,^0 0103375,^0 0100644,^0 60600,^0 0107014,^0 64204

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 166
INITIALIZE SECTION

:11000 BLOCK # 10

044660	000403	070003	057227	.WORD ^0	403	^0	70003	^0	57227	^0	40367	^0	0105417	^0	410	^0	64204	^0	70003
044700	016400	002400	000403	.WORD ^0	16400	^0	2400	^0	403	^0	0104420	^0	406	^0	3300	^0	0104421	^0	401
044720	003000	063227	004003	.WORD ^0	3000	^0	63227	^0	4003	^0	500	^0	70005	^0	76607	^0	62600	^0	64204
044740	070063	002410	100644	.WORD ^0	70063	^0	2410	^0	0100644	^0	16407	^0	14000	^0	14000	^0	14000	^0	16400
044760	002401	100644	123315	.WORD ^0	2401	^0	0100644	^0	0123315	^0	16407	^0	14000	^0	57226	^0	57227	^0	2400
045000	123314	064207	000406	.WORD ^0	0123314	^0	64207	^0	406	^0	70006	^0	42411	^0	0105474	^0	4013	^0	410
045020	070000	042411	003407	.WORD ^0	70000	^0	42411	^0	3407	^0	43267	^0	70074	^0	4002	^0	60367	^0	0105472
045040	002777	100644	002400	.WORD ^0	2777	^0	0100644	^0	2400	^0	0100644	^0	63600	^0	63233	^0	411	^0	63226
045060	000505	063227	000407	.WORD ^0	505	^0	63227	^0	407	^0	61233	^0	0100312	^0	16406	^0	62613	^0	57220
045100	016400	016400	016400	.WORD ^0	16400	^0	16400	^0	16400	^0	16400	^0	2601	^0	0104456	^0	0137315	^0	57222
045120	054620	057225	003004	.WORD ^0	54620	^0	57225	^0	3004	^0	43226	^0	42404	^0	57221	^0	43234	^0	3003
045140	070003	064205	043223	.WORD ^0	70003	^0	64205	^0	43223	^0	60161	^0	0105536	^0	0104564	^0	63221	^0	542
045160	164414	104544	104544	.WORD ^0	0164414	^0	0104544	^0	0104544	^0	0104545	^0	62602	^0	57224	^0	62602	^0	3377
045200	040364	105553	063077	.WORD ^0	40364	^0	0105553	^0	63077	^0	410	^0	70001	^0	76617	^0	62617	^0	4002
045220	070215	016400	016777	.WORD ^0	70215	^0	16400	^0	16777	^0	2417	^0	4007	^0	60600	^0	0106170	^0	4015
045240	070206	054360	105574	.WORD ^0	70206	^0	54360	^0	0105574	^0	0100644	^0	74602	^0	43235	^0	60343	^0	0105206
045260	040342	105203	104613	.WORD ^0	40342	^0	0105203	^0	0104613	^0	40343	^0	0105213	^0	0104620	^0	40342	^0	0105220
045300	040343	105213	104620	.WORD ^0	40343	^0	0105213	^0	0104620	^0	616	^0	0164414	^0	0104626	^0	0104626	^0	0100644
045320	000401	063234	000410	.WORD ^0	401	^0	63234	^0	410	^0	63224	^0	600	^0	63221	^0	14000	^0	43223
045340	070206	043233	002777	.WORD ^0	70206	^0	43233	^0	2777	^0	636	^0	0164414	^0	0104666	^0	0104640	^0	0104647
045360	004013	000400	070000	.WORD ^0	4013	^0	400	^0	70000	^0	43235	^0	412	^0	63224	^0	3201	^0	411
045400	063226	000656	063227	.WORD ^0	63226	^0	656	^0	63227	^0	407	^0	61233	^0	0100312	^0	76604	^0	76613
045420	016400	016400	076615	.WORD ^0	16400	^0	16400	^0	76615	^0	76603	^0	62601	^0	0100644	^0	10053	^0	4017
045440	042410	057221	040361	.WORD ^0	42410	^0	57221	^0	40361	^0	0105701	^0	70201	^0	4016	^0	76600	^0	62603
045460	100644	104701	016420	.WORD ^0	0100644	^0	0104701	^0	16420	^0	57224	^0	57222	^0	43223	^0	405	^0	70002
045500	064203	040364	105717	.WORD ^0	64203	^0	40364	^0	0105717	^0	403	^0	63220	^0	402	^0	0104721	^0	3000
045520	000401	063224	004003	.WORD ^0	401	^0	63224	^0	4003	^0	500	^0	70005	^0	76604	^0	62600	^0	0100644
045540	123314	060600	063224	.WORD ^0	0123314	^0	60600	^0	63224	^0	16420	^0	14000	^0	57223	^0	43222	^0	63120
045560	063120	063120	063120	.WORD ^0	63120	^0	63120	^0	63120	^0	4003	^0	500	^0	70000	^0	16407	^0	2700
045600	070203	064202	003357	.WORD ^0	70203	^0	64202	^0	3357	^0	56677	^0	16400	^0	16400	^0	40620	^0	16400
045620	043236	016400	016400	.WORD ^0	43236	^0	16400	^0	16400	^0	16400	^0	16400	^0	16400	^0	16777	^0	60376
045640	105776	070214	004002	.WORD ^0	0105776	^0	70214	^0	4002	^0	16505	^0	2400	^0	0100644	^0	412	^0	63226

:12000 BLOCK # 11

045660	000405	063227	000407	.WORD	^0	405.^0	63227.^0	407.^0	61233.^0	0100312.^0	16402.^0	76604.^0	16400
045700	016400	016400	016400	.WORD	^0	16400.^0	16400.^0	16400.^0	16600.^0	0100644.^0	60600.^0	63224.^0	16507
045720	016400	057222	043223	.WORD	^0	16400.^0	57222.^0	43223.^0	70202.^0	64203.^0	40620.^0	0113030.^0	0100644
045740	003357	056677	016410	.WORD	^0	3357.^0	56677.^0	16410.^0	16400.^0	16400.^0	16400.^0	16400.^0	16400
045760	016400	016777	016777	.WORD	^0	16400.^0	16777.^0	16777.^0	4003.^0	500.^0	70005.^0	16401.^0	2400
046000	000412	063226	000457	.WORD	^0	412.^0	63226.^0	457.^0	63227.^0	407.^0	61233.^0	0100312.^0	16402
046020	076604	016400	016400	.WORD	^0	76604.^0	16400.^0	16400.^0	16400.^0	16400.^0	2600.^0	0100644.^0	56411
046040	002405	004017	010062	.WORD	^0	2405.^0	4017.^0	10062.^0	54220.^0	0101644.^0	60600.^0	63227.^0	57221
046060	057222	057223	000412	.WORD	^0	57222.^0	57223.^0	412.^0	63236.^0	511.^0	63237.^0	414.^0	61233
046100	100000	016400	076607	.WORD	^0	100000.^0	16400.^0	76607.^0	76602.^0	76601.^0	2777.^0	412.^0	63236
046120	000525	063237	000414	.WORD	^0	525.^0	63237.^0	414.^0	61233.^0	0100000.^0	16400.^0	16407.^0	402
046140	063001	111133	110534	.WORD	^0	63001.^0	0111133.^0	01110534.^0	63062.^0	76602.^0	76601.^0	76603.^0	4017
046160	000466	060407	070007	.WORD	^0	466.^0	60407.^0	70007.^0	2777.^0	412.^0	63236.^0	553.^0	63237
046200	000414	061233	100021	.WORD	^0	414.^0	61233.^0	0100021.^0	2700.^0	42707.^0	0100644.^0	2422.^0	0123314
046220	000500	004003	070005	.WORD	^0	500.^0	4003.^0	70005.^0	43221.^0	16777.^0	43222.^0	4002.^0	402
046240	070014	076601	002600	.WORD	^0	70014.^0	76601.^0	2600.^0	42702.^0	0100644.^0	2424.^0	0123314.^0	412
046260	063236	000606	063237	.WORD	^0	63236.^0	606.^0	63237.^0	417.^0	61233.^0	0100222.^0	3003.^0	0120700
046300	070003	057223	014000	.WORD	^0	70003.^0	57223.^0	14000.^0	43225.^0	4002.^0	70214.^0	14000.^0	14000
046320	043227	000404	070014	.WORD	^0	43227.^0	404.^0	70014.^0	401.^0	60367.^0	0111640.^0	402.^0	60367
046340	111640	000403	060367	.WORD	^0	0111640.^0	403.^0	60367.^0	0111643.^0	16400.^0	16400.^0	2401.^0	0100644
046360	076605	016400	110636	.WORD	^0	76605.^0	16400.^0	0110636.^0	16400.^0	76603.^0	0110636.^0	42411.^0	412
046400	063236	000656	063237	.WORD	^0	63236.^0	656.^0	63237.^0	417.^0	61233.^0	0100222.^0	3004.^0	42704
046420	100644	123301	123322	.WORD	^0	0100644.^0	0123301.^0	0123322.^0	4017.^0	10062.^0	40220.^0	0111702.^0	466
046440	060400	070000	054220	.WORD	^0	60400.^0	70000.^0	54220.^0	0101644.^0	40620.^0	61620.^0	0113302.^0	60600
046460	063227	110540	070201	.WORD	^0	63227.^0	01110540.^0	70201.^0	64202.^0	42411.^0	560.^0	62232.^0	401
046500	062230	000000	000520	.WORD	^0	62230.^0	0.^0	520.^0	62232.^0	626.^0	62230.^0	406.^0	70001
046520	002406	100644	020660	.WORD	^0	2406.^0	0100644.^0	20660.^0	0113325.^0	0100644.^0	0123316.^0	63120.^0	70120
046540	004012	002400	000520	.WORD	^0	4012.^0	2400.^0	520.^0	62232.^0	626.^0	62230.^0	406.^0	70016
046560	004002	042412	111744	.WORD	^0	4002.^0	42412.^0	0111744.^0	0100644.^0	560.^0	62232.^0	400.^0	62230
046600	072616	003004	042404	.WORD	^0	72616.^0	3004.^0	42404.^0	400.^0	63237.^0	60525.^0	63236.^0	63136
046620	067157	000415	070016	.WORD	^0	67157.^0	415.^0	70016.^0	16777.^0	16400.^0	2400.^0	0100644.^0	20660
046640	113372	100644	042411	.WORD	^0	0113372.^0	0100644.^0	42411.^0	0123316.^0	406.^0	70016.^0	52411.^0	43220

:13000 BLOCK # 12

046660	114413	020660	117004	.WORD	^0114413,^0	20660,^0	0117004,^0	0100644,^0	42411,^0	0123316,^0	406,^0	70016
046700	002400	000401	063220	.WORD	^0 2400,^0	401,^0	63220,^0	520,^0	62232,^0	62210,^0	60600,^0	63223
046720	060525	063236	000400	.WORD	^0 60525,^0	63236,^0	400,^0	63237,^0	63136,^0	63157,^0	64217,^0	416
046740	073016	057621	043222	.WORD	^0 73016,^0	57621,^0	43222,^0	63226,^0	417,^0	63666,^0	63126,^0	63006
046760	004017	000600	070006	.WORD	^0 4017,^0	600,^0	70006,^0	57721,^0	57322,^0	43323,^0	417,^0	63121
047000	063141	063141	063141	.WORD	^0 63141,^0	63141,^0	63141,^0	63141,^0	63261,^0	63121,^0	660,^0	70001
047020	057322	043323	064217	.WORD	^0 57322,^0	43323,^0	64217,^0	70216,^0	76602,^0	62603,^0	0100644,^0	60605
047040	061620	061620	061620	.WORD	^0 61620,^0	61620,^0	61620,^0	61620,^0	63220,^0	477,^0	0174400,^0	0114533
047060	114530	114525	114522	.WORD	^0114530,^0	0114525,^0	0114522,^0	0114517,^0	0114514,^0	0114511,^0	10017,^0	4005
047100	114535	010006	004005	.WORD	^0114535,^0	10006,^0	4005,^0	0114535,^0	10375,^0	4004,^0	0114535,^0	10364
047120	004004	114535	010353	.WORD	^0 4004,^0	0114535,^0	10353,^0	4004,^0	0114535,^0	10342,^0	4004,^0	0114535
047140	010331	004004	114535	.WORD	^0 10331,^0	4004,^0	0114535,^0	10320,^0	4004,^0	60616,^0	61233,^0	0160217
047160	060605	061620	061620	.WORD	^0 60605,^0	61620,^0	61620,^0	61620,^0	61620,^0	63220,^0	4004,^0	551
047200	174400	114576	114574	.WORD	^0174400,^0	0114576,^0	0114574,^0	0114572,^0	0114570,^0	0114566,^0	0114564,^0	0114562
047220	010177	114577	010166	.WORD	^0 10177,^0	0114577,^0	10166,^0	0114577,^0	10155,^0	0114577,^0	10144,^0	0114577
047240	010133	114577	010122	.WORD	^0 10133,^0	0114577,^0	10122,^0	0114577,^0	10111,^0	0114577,^0	10100,^0	60616
047260	061233	160217	000000	.WORD	^0 61233,^0	0160217,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047300	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047320	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047340	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047360	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047400	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047420	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047440	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047460	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047500	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047520	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047540	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047560	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047600	000000	000000	000000	.WORD	^0 0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0,^0	0
047620	001161	114761	001141	.WORD	^0 1161,^0	0114761,^0	1141,^0	0114763,^0	1121,^0	0114765,^0	1101,^0	0114767
047640	001061	114771	001041	.WORD	^0 1061,^0	0114771,^0	1041,^0	0114773,^0	1021,^0	0114775,^0	1001,^0	0114777

:14000 BLOCK # 13

047660	010046	004017	000405	.WORD	^0	10046	^0	4017	^0	405	^0	63220	^0	42400	^0	0101007	^0	0100410	^0	2401
047700	057220	040360	101420	.WORD	^0	57220	^0	40360	^0	0101420	^0	70200	^0	4006	^0	60616	^0	61233	^0	0160217
047720	100420	004017	010044	.WORD	^0	0100420	^0	4017	^0	10044	^0	42410	^0	57226	^0	40366	^0	0101434	^0	70206
047740	004010	060616	061233	.WORD	^0	4010	^0	60616	^0	61233	^0	0160217	^0	0100434	^0	0	^0	0	^0	0140620
047760	100513	100564	100600	.WORD	^0	0100513	^0	0100564	^0	0100600	^0	0100443	^0	0100444	^0	0100445	^0	0100446	^0	0100447
050000	100671	100747	104417	.WORD	^0	0100671	^0	0100747	^0	0104417	^0	0104443	^0	0104500	^0	0104545	^0	0104606	^0	0100457
050020	100460	100461	100462	.WORD	^0	0100460	^0	0100461	^0	0100462	^0	0100463	^0	0100464	^0	0100465	^0	0100466	^0	0100467
050040	100470	100471	100513	.WORD	^0	0100470	^0	0100471	^0	0100513	^0	0104762	^0	0104772	^0	0110402	^0	0110412	^0	0110422
050060	110432	110442	110452	.WORD	^0	0110432	^0	0110442	^0	0110452	^0	0100513	^0	0110462	^0	0110672	^0	0110740	^0	0110672
050100	000400	061233	100121	.WORD	^0	400	^0	61233	^0	0100121	^0	20660	^0	0103116	^0	0100510	^0	0123313	^0	63120
050120	070120	004012	002400	.WORD	^0	70120	^0	4012	^0	2400	^0	400	^0	63237	^0	60525	^0	63236	^0	63136
050140	067157	000415	070016	.WORD	^0	67157	^0	415	^0	70016	^0	40220	^0	0101545	^0	16777	^0	16400	^0	43220
050160	002400	070213	004002	.WORD	^0	2400	^0	70213	^0	4002	^0	56411	^0	0100560	^0	2400	^0	3000	^0	0121273
050200	123244	114426	121273	.WORD	^0	0123244	^0	0114426	^0	0121273	^0	0123244	^0	0114426	^0	64217	^0	70216	^0	43220
050220	000520	062232	062210	.WORD	^0	520	^0	62232	^0	62210	^0	0100510	^0	20660	^0	0103167	^0	0100510	^0	560
050240	062232	000402	062230	.WORD	^0	62232	^0	402	^0	62230	^0	0	^0	520	^0	62232	^0	777	^0	62230
050260	016400	016400	002400	.WORD	^0	16400	^0	16400	^0	2400	^0	414	^0	63236	^0	612	^0	63237	^0	417
050300	061233	100222	003173	.WORD	^0	61233	^0	0100222	^0	3173	^0	42673	^0	43233	^0	63533	^0	0103622	^0	3200
050320	042700	100510	003177	.WORD	^0	42700	^0	0100510	^0	3177	^0	42677	^0	10062	^0	4017	^0	54220	^0	0101510
050340	057221	057222	057223	.WORD	^0	57221	^0	57222	^0	57223	^0	414	^0	63236	^0	642	^0	63237	^0	414
050360	061233	100000	016400	.WORD	^0	61233	^0	0100000	^0	16400	^0	76606	^0	76602	^0	76601	^0	2777	^0	414
050400	063236	000656	063237	.WORD	^0	63236	^0	656	^0	63237	^0	414	^0	61233	^0	0100000	^0	16400	^0	16403
050420	000402	063001	101264	.WORD	^0	402	^0	63001	^0	0101264	^0	0100665	^0	63062	^0	76602	^0	76601	^0	62603
050440	100510	060600	063222	.WORD	^0	0100510	^0	60600	^0	63222	^0	0123314	^0	16451	^0	57223	^0	43224	^0	414
050460	063236	000706	063237	.WORD	^0	63236	^0	706	^0	63237	^0	417	^0	61233	^0	0100222	^0	40620	^0	0103343
050500	000414	063236	000717	.WORD	^0	414	^0	63236	^0	717	^0	63237	^0	413	^0	61233	^0	0100146	^0	16400
050520	016400	016400	016400	.WORD	^0	16400	^0	16400	^0	16400	^0	16400	^0	16400	^0	16400	^0	76603	^0	76604
050540	000414	063236	000737	.WORD	^0	414	^0	63236	^0	737	^0	63237	^0	414	^0	61233	^0	0100021	^0	600
050560	076702	062603	100510	.WORD	^0	76702	^0	62603	^0	0100510	^0	4002	^0	70214	^0	2400	^0	0110714	^0	2452
050600	004017	010062	054220	.WORD	^0	4017	^0	10062	^0	54220	^0	0101510	^0	57221	^0	57226	^0	43223	^0	60600
050620	063222	000414	063236	.WORD	^0	63222	^0	414	^0	63236	^0	770	^0	63237	^0	414	^0	61233	^0	0100000
050640	016400	076602	076606	.WORD	^0	16400	^0	76602	^0	76606	^0	76601	^0	2777	^0	415	^0	63236	^0	404

:15000 BLOCK # 14

050660	063237	000414	061233	.WORD	^0	63237	^0	414	^0	61233	^0	100000	^0	16400	^0	16407	^0	402	^0	63001
050700	105012	104413	063066	.WORD	^0	105012	^0	104413	^0	63066	^0	76606	^0	76601	^0	76603	^0	100510	^0	2453
050720	004002	000600	070005	.WORD	^0	4002	^0	600	^0	70005	^0	16777	^0	14000	^0	16777	^0	14000	^0	16777
050740	014000	016777	014000	.WORD	^0	14000	^0	16777	^0	14000	^0	16777	^0	14000	^0	16777	^0	14000	^0	16777
050760	014000	002777	100510	.WORD	^0	14000	^0	2777	^0	100510	^0	123314	^0	415	^0	63236	^0	453	^0	63237
051000	000413	061233	100140	.WORD	^0	413	^0	61233	^0	100140	^0	120700	^0	3003	^0	70003	^0	3077	^0	57277
051020	060177	105463	104465	.WORD	^0	60177	^0	105463	^0	104465	^0	40012	^0	101510	^0	70214	^0	4002	^0	16454
051040	002601	063120	004005	.WORD	^0	2601	^0	63120	^0	4005	^0	570	^0	70000	^0	16406	^0	2400	^0	100510
051060	002455	123314	000415	.WORD	^0	2455	^0	123314	^0	415	^0	63236	^0	511	^0	63237	^0	413	^0	61233
051100	100140	014000	014000	.WORD	^0	100140	^0	14000	^0	14000	^0	17077	^0	57277	^0	120700	^0	43236	^0	40012
051120	105522	104524	070220	.WORD	^0	105522	^0	104524	^0	70220	^0	56412	^0	14000	^0	16400	^0	16400	^0	14000
051140	040371	105534	000500	.WORD	^0	40371	^0	105534	^0	500	^0	104535	^0	600	^0	63622	^0	402	^0	70014
051160	004002	076616	062617	.WORD	^0	4002	^0	76616	^0	62617	^0	42702	^0	100510	^0	16456	^0	123314	^0	415
051200	063236	000556	063237	.WORD	^0	63236	^0	556	^0	63237	^0	417	^0	61233	^0	100222	^0	3004	^0	42704
051220	003040	042700	057220	.WORD	^0	3040	^0	42700	^0	57220	^0	14000	^0	14000	^0	42411	^0	57221	^0	14000
051240	043222	000403	070014	.WORD	^0	43222	^0	403	^0	70014	^0	4002	^0	63520	^0	107600	^0	3200	^0	42700
051260	014000	076602	076601	.WORD	^0	14000	^0	76602	^0	76601	^0	16401	^0	2400	^0	100510	^0	42411	^0	415
051300	063236	000616	063237	.WORD	^0	63236	^0	616	^0	63237	^0	414	^0	61233	^0	100021	^0	62600	^0	415
051320	063236	000626	063237	.WORD	^0	63236	^0	626	^0	63237	^0	414	^0	61233	^0	100021	^0	62600	^0	415
051340	063236	000636	063237	.WORD	^0	63236	^0	636	^0	63237	^0	414	^0	61233	^0	100021	^0	62600	^0	415
051360	063236	000646	063237	.WORD	^0	63236	^0	646	^0	63237	^0	414	^0	61233	^0	100021	^0	62600	^0	10062
051400	004017	040220	101510	.WORD	^0	4017	^0	40220	^0	101510	^0	412	^0	61233	^0	100277	^0	17077	^0	57226
051420	043277	060520	063223	.WORD	^0	43277	^0	60520	^0	63223	^0	4005	^0	570	^0	70003	^0	42410	^0	105275
051440	054366	105301	040377	.WORD	^0	54366	^0	105301	^0	40377	^0	105301	^0	104711	^0	14000	^0	42411	^0	70003
051460	104670	000415	063236	.WORD	^0	104670	^0	415	^0	63236	^0	710	^0	63237	^0	414	^0	61233	^0	100021
051500	062600	004002	000520	.WORD	^0	62600	^0	4002	^0	520	^0	62232	^0	600	^0	60405	^0	70001	^0	43220
051520	105726	016777	043220	.WORD	^0	105726	^0	16777	^0	43220	^0	121273	^0	123244	^0	114426	^0	62210	^0	415
051540	063236	000736	063237	.WORD	^0	63236	^0	736	^0	63237	^0	413	^0	61233	^0	100140	^0	120700	^0	3004
051560	070004	042412	040012	.WORD	^0	70004	^0	42412	^0	40012	^0	105745	^0	100510	^0	3003	^0	70003	^0	477
051600	063227	043667	060012	.WORD	^0	63227	^0	43667	^0	60012	^0	105756	^0	42412	^0	100510	^0	70214	^0	4002
051620	002503	100510	020660	.WORD	^0	2503	^0	100510	^0	20660	^0	107365	^0	100510	^0	2474	^0	123314	^0	400
051640	063221	104711	020660	.WORD	^0	63221	^0	104711	^0	20660	^0	107375	^0	100510	^0	16475	^0	123314	^0	402

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 171
INITIALIZE SECTION

:16000 BLOCK # 15

051660	063221	104656	020660	.WORD	^0 63221	^0104656	^0 20660	^0113005	^0100510	^0 2476	^0123314	^0 404
051700	063221	104711	020660	.WORD	^0 63221	^0104711	^0 20660	^0113015	^0100510	^0 16477	^0123314	^0 406
051720	063221	104656	020660	.WORD	^0 63221	^0104656	^0 20660	^0113025	^0100510	^0 2500	^0123314	^0 410
051740	063221	104711	020660	.WORD	^0 63221	^0104711	^0 20660	^0113035	^0100510	^0 16501	^0123314	^0 412
051760	063221	104656	020660	.WORD	^0 63221	^0104656	^0 20660	^0113045	^0100510	^0 2502	^0123314	^0 414
052000	063221	104711	020660	.WORD	^0 63221	^0104711	^0 20660	^0113055	^0100510	^0 16473	^0123314	^0 416
052020	063221	104656	123314	.WORD	^0 63221	^0104656	^0123314	^0 416	^0 63236	^0 472	^0 63237	^0 417
052040	061233	100222	137313	.WORD	^0 61233	^0100222	^0137313	^0 43235	^0 63535	^0113567	^0 20660	^0113101
052060	100510	000560	062232	.WORD	^0100510	^0 560	^0 62232	^0 402	^0 62230	^0 70213	^0 520	^0 62232
052100	000777	062230	003173	.WORD	^0 777	^0 62230	^0 3173	^0 42673	^0 43233	^0 63533	^0113523	^0 600
052120	063233	042713	110567	.WORD	^0 63233	^0 42713	^0110567	^0 10062	^0 4017	^0 54220	^0111567	^0 57221
052140	057222	057223	000416	.WORD	^0 57222	^0 57223	^0 416	^0 63236	^0 541	^0 63237	^0 414	^0 61233
052160	100000	016400	076606	.WORD	^0100000	^0 16400	^0 76606	^0 76602	^0 76601	^0 2777	^0 416	^0 63236
052200	000555	063237	000414	.WORD	^0 555	^0 63237	^0 414	^0 61233	^0100000	^0 16400	^0 16403	^0 402
052220	063001	111163	110564	.WORD	^0 63001	^0111163	^0110564	^0 63062	^0 76602	^0 76601	^0 76603	^0 70214
052240	004002	016401	014000	.WORD	^0 4002	^0 16401	^0 14000	^0 14000	^0 57221	^0 14000	^0 57222	^0 60606
052260	063220	000416	063236	.WORD	^0 63220	^0 416	^0 63236	^0 610	^0 63237	^0 413	^0 61233	^0100146
052300	120700	003007	070007	.WORD	^0120700	^0 3007	^0 70007	^0 43225	^0 3004	^0 10000	^0 4007	^0 63600
052320	112222	004015	040220	.WORD	^0112222	^0 4015	^0 40220	^0111631	^0120700	^0 70004	^0111230	^0110622
052340	110630	076600	016400	.WORD	^0110630	^0 76600	^0 16400	^0 76602	^0 62605	^0 4002	^0 70214	^0 60615
052360	113645	063521	113650	.WORD	^0113645	^0 63521	^0113650	^0 2442	^0110650	^0 16507	^0 2400	^0100510
052400	010042	004017	043225	.WORD	^0 10042	^0 4017	^0 43225	^0 416	^0 63236	^0 662	^0 63237	^0 417
052420	061233	100222	123313	.WORD	^0 61233	^0100222	^0123313	^0 410	^0 70013	^0 56411	^0111270	^0100510
052440	042411	100510	123307	.WORD	^0 42411	^0100510	^0123307	^0 17004	^0 43222	^0 42404	^0111333	^0 4007
052460	060600	112303	004015	.WORD	^0 60600	^0112303	^0 4015	^0 70202	^0 40360	^0111707	^0100510	^0 43222
052500	016777	014000	014000	.WORD	^0 16777	^0 14000	^0 14000	^0 43223	^0 416	^0 63226	^0 723	^0 63227
052520	000407	061233	100312	.WORD	^0 407	^0 61233	^0100312	^0 16416	^0 76602	^0 16400	^0 16400	^0 16400
052540	076603	016601	100510	.WORD	^0 76603	^0 16601	^0100510	^0 70207	^0 43620	^0 16506	^0 62600	^0100510
052560	123313	060600	063224	.WORD	^0123313	^0 60600	^0 63224	^0 4017	^0 10053	^0 57221	^0 40361	^0115403
052600	040620	060410	073222	.WORD	^0 40620	^0 60410	^0 73222	^0 4016	^0 40364	^0111757	^0110777	^0 16777
052620	043223	000416	063226	.WORD	^0 43223	^0 416	^0 63226	^0 770	^0 63227	^0 407	^0 61233	^0100312
052640	016416	076604	016400	.WORD	^0 16416	^0 76604	^0 16400	^0 16400	^0 16400	^0 76603	^0 2601	^0 60602

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 3GA(1052) 23-DEC-82 14:01 PAGE 172
INITIALIZE SECTION

:17000 BLOCK # 16

052660	060361	115403	110751	.WORD	^0	60361,^0115403,^0110751,^0	4002,^0	70073,^0	40620,^0116421,^0	417
052700	063236	000416	063237	.WORD	^0	63236,^0 416,^0 63237,^0	417,^0	61233,^0100222,^0	3020,^0	42700
052720	004002	070213	016400	.WORD	^0	4002,^0 70213,^0 16400,^0	16777,^0	2417,^0100510,^0	60600,^0	63223
052740	060525	063236	000400	.WORD	^0	60525,^0 63236,^0 400,^0	63237,^0	63136,^0 63157,^0	64217,^0	416
052760	073016	057621	043222	.WORD	^0	73016,^0 57621,^0 43222,^0	63226,^0	417,^0 63666,^0	63126,^0	63006
053000	004017	000600	070006	.WORD	^0	4017,^0 600,^0 70006,^0	57721,^0	57322,^0 43323,^0	417,^0	63121
053020	063141	063141	063141	.WORD	^0	63141,^0 63141,^0 63141,^0	63141,^0	63261,^0 63121,^0	660,^0	70001
053040	057322	043323	064217	.WORD	^0	57322,^0 43323,^0 64217,^0	70216,^0	76602,^0 62603,^0	63064,^01	60064
053060	000502	174407	114505	.WORD	^0	502,^0174407,^0114505,^0	114564,^0114611,^0	43234,^0	417,^0	63236
053100	000515	063237	000417	.WORD	^0	515,^0 63237,^0 417,^0	61233,^0100230,^0	57221,^0	57222,^0	57223
053120	057224	057225	057235	.WORD	^0	57224,^0 57225,^0 57235,^0	57236,^0	57233,^0 417,^0	63226,^0	534
053140	063227	000407	061233	.WORD	^0	63227,^0 407,^0 61233,^0	100312,^0 16400,^0	76614,^0	76601,^0	76602
053160	076603	076604	002603	.WORD	^0	76603,^0 76604,^0 2603,^0	417,^0	63226,^0 552,^0	63,^07,^0	407
053200	061233	100312	016402	.WORD	^0	61233,^0100312,^0 16402,^0	76614,^0	76605,^0 76615,^0	76616,^0	76613
053220	002603	000400	061233	.WORD	^0	2603,^0 400,^0 61233,^0	100043,^0 14000,^0	57221,^0 43222,^0	64201,^0	64201
053240	070202	043223	000417	.WORD	^0	70202,^0 43223,^0 417,^0	63226,^0	601,^0 63227,^0	407,^0	61233
053260	100312	016404	016400	.WORD	^0	100312,^0 16404,^0 16400,^0	76602,^0	76601,^0 76603,^0	16400,^0	2603
053300	114561	014000	057221	.WORD	^0	114561,^0 14000,^0 57221,^0	57222,^0	14000,^0 43223,^0	64201,^0	70202
053320	062603	114567	060605	.WORD	^0	62603,^0114567,^0 60605,^0	61620,^0	61620,^0 61620,^0	61620,^01	14631
053340	040620	003007	063667	.WORD	^0	40620,^0 3007,^0 63667,^0	63226,^0	636,^0174406,^0114672,^0	114667,^0	114667
053360	114664	114661	114656	.WORD	^0	114664,^0114661,^0114656,^0	114653,^0114650,^0	10300,^0 4001,^0	114674,^0	4001
053400	010200	004001	114674	.WORD	^0	10200,^0 4001,^0114674,^0	10100,^0	4001,^0114674,^0	10000,^0	4001
053420	114674	010300	004000	.WORD	^0	114674,^0 10300,^0 4000,^0	114674,^0	10200,^0 4000,^0	114674,^0	10100
053440	004000	114674	010000	.WORD	^0	4000,^0114674,^0 10000,^0	4000,^0	60616,^0 61233,^0	160217,^0	0
053460	000000	000000	000000	.WORD	^0	0,^0 0,^0 0,^0	0,^0	0,^0 0,^0	0,^0	0
053500	000000	000000	000000	.WORD	^0	0,^0 0,^0 0,^0	0,^0	0,^0 0,^0	0,^0	0
053520	000000	000000	000000	.WORD	^0	0,^0 0,^0 0,^0	0,^0	0,^0 0,^0	0,^0	0
053540	000000	000000	000000	.WORD	^0	0,^0 0,^0 0,^0	0,^0	0,^0 0,^0	0,^0	0
053560	000000	000000	000000	.WORD	^0	0,^0 0,^0 0,^0	0,^0	0,^0 0,^0	0,^0	0
053600	000000	000000	000000	.WORD	^0	0,^0 0,^0 0,^0	0,^0	0,^0 0,^0	0,^0	0
053620	001161	114761	001141	.WORD	^0	1161,^0114761,^0 1141,^0	114763,^0	1121,^0114765,^0	1101,^0114767,^0	1101
053640	001061	114771	001041	.WORD	^0	1061,^0114771,^0 1041,^0	114773,^0	1021,^0114775,^0	1001,^0114777,^0	1001

053660

DDCMPE:

:END OF DCLT FIRMWARE IMAGE

020000

MCSIZ = .-DDCMPS

:SIZE OF FIRMWARE IMAGE

053660 047045 051445 022463
 053720 047045 022462 031523
 053771 045 022516 031523

FIRMLD: .ASCIZ /%N%\$3%AFailure Loading Firmware/
 DATAHD: .ASCIZ /%N2%\$3%ADDRESS%\$3%LOADED%\$3%AREAD BACK/
 DATAID: .ASCIZ /%N%\$3%06%\$4%06%\$3%06/

.LIST BEX
.EVEN

ENDINIT

L10015:

TRAP CSINIT

5590
 5591
 5592 054016
 5593 054016
 5594 054016 104411

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 173
AUTODROP SECTION

.SBTTL AUTODROP SECTION

:+
: THIS CODE IS EXECUTED IMMEDIATELY AFTER THE INITIALIZE CODE IF
: THE "ADR" FLAG WAS SET. THE UNIT(S) UNDER TEST ARE CHECKED TO
: SEE IF THEY WILL RESPOND. THOSE THAT DON'T ARE IMMEDIATELY
: DROPPED FROM TESTING.
:--

5595
5596
5597
5598
5599
5600
5601
5602
5603
5604
5605
5606
5607
5608
5609
5610

054020
054020

054020
054020
054020 104461

BGNAUTO

ENDAUTO

LSAUTO::

L10016: TRAP CSAUTO

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 174
CLEANUP CODING SECTION

.SBTTL CLEANUP CODING SECTION

: THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED
: AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.
:--

5611
5612
5613
5614
5615
5616
5617
5618
5619
5620
5621
5622
5623
5624
5625
5626
5627
5628
5629
5630
5631
5632
5633
5634
5635
5636
5637
5638
5639
5640
5641
5642
5643
5644
5645
5646
5647
5648
5649
5650
5651
5652
5653
5654
5655
5656
5657

054022
054022

054022 005737 011512
054026 001402
054030 004737 063704
054034 005237 011516
054040 005037 011512
054044 012737 177777 010134
054052 004737 064020
054056 005037 010134
054062 005077 134162
054066
054066 104433
054070
054070 012700 000340
054074 104441

054076 022737 000057 003246
054104 001416

054106 012737 000026 010164
054114 013737 010116 010174
054122 013737 010122 010170
054130 013737 010124 010172
054136 004737 024534

054142
054142
054142 104432
054144 000002

054146
054146
054146 104412

BGNCLN

L\$CLEAN::

10\$: TST ENRDST ; IS READ STATUS ENABLED ?
BEQ 10\$; NO. OMIT READING STATUS
JSR PC, LNSTAT ; READ LINE STATUS
INC CNTCLN ; INCREMENT CLEANUP COUNT
CLR ENRDST ; NEGATE READ STATUS ENABLED
MOV #-1, CLNSET ; SET THE CLEANUP FLAG
JSR PC, HLTRB ; HALT LINE
CLR CLNSET ; CLEAR CLEANUP FLAG
CLR @CLKCSR ; DISABLE CLOCK
BRESET

SETPRI #PRI07

; SET PROCESSOR PRIORITY BACK TO 7 TRAP C\$RESET
MOV #PRI07, RO TRAP
TRAP C\$SPRI

CMP #EXIT, KEYWD1 ; "EXIT" COMMAND?
BEQ EXITCLN ; YES.

: LOG ^C ABORT IN EVENT LOG

MOV #ABO, TEMP ; LOAD EVENT TYPE
MOV OPVAR, TEMP4 ; NUMBER OF NO-BUFFS
MOV PSCNT, TEMP2 ; NUMBER OF PASSES
MOV ERRCNT, TEMP3 ; ERRORS
JSR PC, LOG\$S ; GO LOG IT

EXITCLN: EXIT CLN

TRAP C\$EXIT
.WORD L10017-

.EVEN

ENDCLN

L10017: TRAP C\$CLEAN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 175
DROP UNIT SECTION

.SBTTL DROP UNIT SECTION

::++
: THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
: TO NO LONGER BE TESTED.
:--

5658
5659
5660
5661
5662
5663
5664
5665 054150
5666 054150
5667
5668
5669 054150
5670 054150 000167
5671 054152 000000
5672
5673
5674
5675
5676 054154
5677 054154
5678 054154 104453

BGNDU

L\$DU::

EXIT DU

.WORD JSJMP
.WORD L10020-2-

.EVEN

ENDDU

L10020: TRAP C\$DU

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 176
ADD UNIT SECTION

.SBTTL ADD UNIT SECTION

:+
: THE ADD-UNIT SECTION CONTAINS ANY CODE THE PROGRAMMER WISHES
: TO BE EXECUTED IN CONJUNCTION WITH THE ADDING OF A UNIT BACK
: TO THE TEST CYCLE.
:--

5679
5680
5681
5682
5683
5684
5685
5686
5687
5688
5689
5690
5691
5692
5693
5694
5695
5696
5697
5698
5699
5700
5701
5702

054156
054156

054156 000167
054156 000000

054162
054162
054162 104452

BGNAU

EXIT AU

.EVEN

ENDAU

L\$AU::

.WORD JSJMP
.WORD L10021-2-

L10021: TRAP CSAU

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 177
TEST 1: SETUP AND MODES OF OPERATION

.SBTTL TEST 1: SETUP AND MODES OF OPERATION

;++
: TEST TO DETECT FAULTS IN THE DATA COMMUNICATION LINK. THIS TEST WILL
: THE PROVIDE COVERAGE NECESSARY TO DETECT FAILURES IN THE COMPUTER
: EQUIPMENT, THE COMMUNICATION LINK, OR THE MODEM.
:--

5703
5704
5705
5706
5707
5708
5709
5710
5711
5712
5713
5714
5715
5716
5717
5718
5719
5720
5721
5722
5723
5724
5725
5726
5727
5728
5729
5730
5731
5732
5733
5734
5735
5736
5737
5738
5739
5740
5741
5742
5743
5744
5745
5746
5747
5748
5749
5750
5751
5752
5753
5754
5755
5756
5757
5758

054164
054164

BGNTST

T1::

.SBTTL PROGRAM SETUP SECTION

MOV CLKEN,@CLKCSR ;ENABLE THE CLOCK

GTXRXB:
GTRA2:

1\$: CLR R1 ;SET TIMER TO COUNT 1 TICK
MOV #1,TIMER1 ;CHECK FOR IT TO BE COUNTED OFF
TST TIMER1 ;BRANCH IF CLOCK EXISTS (COUNTED A TICK)
BEQ GTRA3

DEC R1 ;KEEP CHECKING UNTIL R1 DOES FULL COUNTDOWN
BNE 1\$;PRINT BAD CLK MSG AND WARN OF HANG IF TIMEOUT
PRINTF #NOCLK
MOV #NOCLK,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C\$PNTF
ADD #4,SP

GTRA3: TST RESFLG ;SEE IF HERE AFTER A RESTART
BNE GTRA5 ;BR IF HERE BECAUSE OF A RESTART

; CLEAR COUNTS AND SET UP DEFAULTS

GTRA4: CLR TOTCC ;CLEAR TOTAL CHAR. COUNT TEMP. LOC
CLR TTOTCC ; CLEAR TOTAL CHAR. COUNT FOR TX BUFF
CLR CTOTCC ; CLEAR TOTAL CHAR. COUNT FOR CMP BUFF
MOV #PTRTAB, TXPTR ;INIT TRANSMIT MESSAGE POINTER

CLR RXPTR ; ZERO RX POINTER
MOV #PTR13, CMPPTR ;INIT COMP POINTER

MOV #5,MSGTYP ;SET UP DEFAULT MSG TYPE (QUICK FOX - ITEMP MSG)
MOV MSG5C,CURCC ;SET UP DEFAULT CHAR COUNT
MOV #TXBUF,TCURAD ;SET UP CURRENT ADD TO START OF TX BUFFER
MOV #CMPBUF,CCURAD ;SET UP CURRENT ADD TO START OF CMP BUFFER

MOV TCURAD,CURADD ;SETUP CURRENT ADDR TO START OF TXBUF
MOV TXPTR,CPTR ;SETUP CURRENT POINTER TABLE POINTER FOR TXBUF
JSR PC,BLDBUF ; GO BUILD POINTER TABLE AND BUFFER
MOV #1,TXMTOT ;BUMP TOTAL MESSAGE COUNT

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 178
PROGRAM SETUP SECTION

```

5759
5760 054354 013737 010054 010154      MOV      CMPPTR,CPTR      ;SET UP START OF COMPARE POINTER TABLE
5761 054362 013737 010062 010156      MOV      CCURAD,CURADD   ;SET UP CURRENT ADDR. TO START OF CMPBUF
5762 054370 012737 000005 010146      MOV      #5,MSGTYP
5763 054376 013737 002156 010150      MOV      MSGSC,CURCC
5764 054404 004737 027520                JSR      PC,BLDBUF       ;PUT DEFAULT MESSAGE INTO CMPBUF
5765 054410 012737 000001 010056      MOV      #1,CMPTOT      ;BUMP THE COMP MESSG COUNT
5766 054416 012737 000003 010216      MOV      #ACT,MODTYP    ;SET DEFAULT MODE= ACTIVE
5767 054424 005037 010220                CLR      MLTYP          ;SET DEFAULT MAINTENANCE LOOP MODE =NONE
5768 054430 012737 000001 010226      MOV      #1,RPASS      ;SET UP DEFAULT 'RUN PASS' COUNT TO 1
5769 054436 012737 000002 010224      MOV      #2,PARAM      ;SET UP PROG. PARAMETERS - DATA CHECKING ENABLD
5770
5771 054444 012737 000063 003246      MOV      #KTRB,KEYWD1   ;OPERATOR STATUS MSGS. PRINT OFF
5772 054452 004737 057566                JSR      PC,ACTKAL      ;SET UP KEYWRD.
5773
5774 054456 012702 013434                1$:      MOV      #BDLBAS,R2     ;INIT MSG BUFFER DESCRIPT LIST
5775 054462 005022                CLR      (R2)+
5776 054464 020227 013634                CMP      R2,#BDLEND
5777 054470 001374                BNE      1$
5778
5779 054472                PRINTF  #HLP0
5780 054472 012746 014540                MOV      #HLP0,-(SP)
5781 054476 012746 000001                MOV      #1,-(SP)
5782 054502 010600                MOV      SP,R0
5783 054504 104417                TRAP    C$PNTF
5784 054506 062706 000004                ADD     #4,SP
5785 054512
5786 054512 013737 010216 011470      GTRAS:  MOV      MODTYP,DEV1
5787 054520 013737 010220 011472      MOV      MLTYP,DEV2
5788 054526 013737 010226 011474      MOV      RPASS,DEV3
5789 054534 013737 010224 011476      MOV      PARAM,DEV4
5790 054542 004737 031204                JSR      PC,SHWOP      ;PRINT TO OPERATOR THE CURRENT MODE.....
5791
5792 054546                MANUAL                ;SEE IF MANUAL INTERVENTION ALLOWED
5793 054546 104450                TRAP    C$MANI
5794 054550                BCOMPLETE  GETCL      ; BR IF YES (UAM=0 AND NOT CHAINED)
5795 054550 103412                BCS     GETCL
5796 054552 005737 010226                TST     RPASS          ;SEE IF THIS IS FIRST 'DCLT PASS'
5797 054556 001002                BNE     1$            ; BR IF NOT COMPLETED 1 PASS
5798 054560                EXIT  TST              ; IF DONE 1 PASS IN UNATTENDED MODE - EXIT
5799 054560 104432                TRAP    C$EXIT
5800 054562 012442                .WORD  L10022-.
5801 054564 012737 000001 010220 1$:      MOV      #TTL,MLTYP    ;SET UP DEFAULT FOR UNATTENDED MODE
5802 054572 000137 060174                JMP     GTR9           ; 'R M=ACT/LO=1/PAS=1/NOST/CH' AND RUN
5803

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 179
COMMAND LINE FETCH & INTERPRETATION SECTION

```

5804          .SBTTL          COMMAND LINE FETCH & INTERPRETATION SECTION
5805
5806 054576 105037 003401      GETCL: CLRB   P$GDBD          ;CLEAR CMD LINE PARSING ERROR FLAGS
5807 054602 105037 003400      CLRB   P$NNUF
5808 054606          GMANID  CLISPM,CMDBUF,A,377,1,72.,NO ;GET A COMMAND LINE FROM OPR.
5809 054606 104443          TRAP   C$GMAN
5810 054610 000406          BR     10000$
5811 054612 003124          .WORD  CMDBUF
5812 054614 000142          .WORD  T$CODE
5813 054616 014002          .WORD  CLISPM
5814 054520 000377          .WORD  377
5815 054622 000001          .WORD  T$LOLIM
5816 054624 000110          .WORD  T$HILIM
5817 054626          10000$:
5818 054626 012737 003124 003364      MOV   #CMDBUF,P$BUFA
5819 054634 012737 011520 003366      MOV   #CLITRE,P$TREE
5820 054642 012737 055540 003370      MOV   #CLIACT,P$ACT
5821 054650 005037 003250          CLR   QUALFG          ;CLEAR QUALIFIER FLAG LOCATION
5822 054654 004737 031506          JSR   PC,P$TRV        ;GO PARSE COMMAND LINE
5823 054660 105737 003401          TSTB  P$GDBD          ;SEE IF PARSED OK OR AN ERROR
5824 054664 001412          BEQ   1$
5825 054666          PRINTF #CLIERM
5826 054666 012746 014010          MOV   #CLIERM,-(SP)
5827 054672 012746 000001          MOV   #1,-(SP)
5828 054676 010600          MOV   SP,RO
5829 054700 104417          TRAP  C$PNTF
5830 054702 062706 000004          ADD   #4,SP
5831 054706 000137 054576          JMP   GETCL
5832 054712 105737 003400      1$:  TSTB  P$NNUF          ;SEE IF INCOMPLETE COMMAND TYPED
5833 054716 001412          BEQ   10$
5834 054720          PRINTF #CLINUF
5835 054720 012746 014040          MOV   #CLINUF,-(SP)
5836 054724 012746 000001          MOV   #1,-(SP)
5837 054730 010600          MOV   SP,RO
5838 054732 104417          TRAP  C$PNTF
5839 054734 062706 000004          ADD   #4,SP
5840 054740 000137 054576          JMP   GETCL
5841
5842 054744 023727 003246 000060 10$:  CMP   KEYWD1,#SETET    ;WAS "SET E=T" ENTERED ?
5843 054752 001711          BEQ   GETCL           ;YES,BRANCH
5844 054754 023727 003246 000004      CMP   KEYWD1,#RUN     ;SEE IF RUN WAS TYPED
5845 054762 001002          BNE   11$            ; BR IF NO
5846 054764 000137 060174          JMP   GTR9           ; START EXEC. IF YES
5847 054770 023727 003246 000052 11$:  CMP   KEYWD1,#DMPS
5848 054776 001004          BNE   14$            ; IS IT DUMP
5849 055000 004737 027264          JSR   PC,DUMPSR     ;GO TO DUMPSR
5850 055004 000137 054576          JMP   GETCL         ;AND GO BACK
5851 055010 023727 003246 000057 14$:  CMP   KEYWD1,#EXIT
5852 055016 001005          BNE   40$            ;IS IT EXIT
5853 055020 012737 177777 010212      MOV   #-1,DCLFLG    ;BRANCH IF NOT
5854 055026          EXIT   TST          ;SET DO CLEAN FLAG
5855 055026 104432          TRAP  C$EXIT
5856 055030 012174          .WORD  L10022-
5857 055032 023727 003246 000010 40$:  CMP   KEYWD1,#SETEXP
5858 055040 001001          BNE   4$
5859 055042 000522          BR    2$

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 180
COMMAND LINE FETCH & INTERPRETATION SECTION

```

5860 055044 023727 003246 000011 4$:  CMP      KEYWD1,#SETTRN      ;SEE IF SET TX
5861 055052 001402                BEQ      5$                ; BR IF YES
5862 055054 000137 054576                JMP      GETCL
5863
5864 055060 013737 010076 010160 5$:  MOV      TTOTCC,TOTCC
5865 055066 023727 010160 001000                CMP      TOTCC,#BUFLIM      ;SEE IF BUFFER ALREADY FULL
5866 055074 002414                BLT      15$                ; BR IF NOT FULL (BUFLIM # OF CHARS.)
5867 055076                PRINTF  #MSGTRN,#BUFEX      ; ELSE TELL OPR. AND DON'T BUILD MSG.
5868 055076 012746 017036                MOV      #BUFEX,-(SP)
5869 055102 012746 017054                MOV      #MSGTRN,-(SP)
5870 055106 012746 000002                MOV      #2,-(SP)
5871 055112 010600                MOV      SP,RO
5872 055114 104417                TRAP    C$PNTF
5873 055116 062706 000006                ADD      #6,SP
5874 055122 000137 054576                JMP      GETCL              ; THEN GO GET A NEW COMMAND
5875 055126 005737 010076                15$:  TST      TTOTCC              ;IF FIRST "SET" THEN GET RID OF DEFAULT
5876 055132 001002                BNE     6$
5877 055134 005037 010074                CLR      TXMTOT
5878 055140 012737 006406 010052 6$:  MOV      #PTRTAB, TXPTR      ;GET POSITION OF END OF TX LIST
5879 055146 013701 010074                MOV      TXMTOT,R1
5880 055152 020127 000017                CMP      R1,#MSGLIM
5881 055156 002414                BLT      17$
5882 055160                PRINTF  #MSGTRN,#TABEX      ;SEE IF MSG COUNT EXCEEDED.
5883 055160 012746 016776                MOV      #TABEX,-(SP)
5884 055164 012746 017054                MOV      #MSGTRN,-(SP)
5885 055170 012746 000002                MOV      #2,-(SP)
5886 055174 010600                MOV      SP,RO
5887 055176 104417                TRAP    C$PNTF
5888 055200 062706 000006                ADD      #6,SP
5889 055204 000137 054576                JMP      GETCL              ; THEN GO GET A NEW COMMAND.
5890 055210 006301                17$:  ASL      R1                  ;# OF MSGS *4 = NEXT FREE PTR BLOCK
5891 055212 006301                ASL      R1
5892 055214 060137 010052                ADD      R1, TXPTR
5893 055220 013737 010052 010154                MOV      TXPTR, CPTR
5894 055226 013737 010100 010156                MOV      TCURAD, CURADD
5895 055234 004737 027422                JSR      PC, ADDCC
5896 055240 004737 027520                JSR      PC, BLDBUF
5897 055244 004737 027644                JSR      PC, FCURAD
5898 055250 013737 010154 010052                MOV      CPTR, TXPTR
5899 055256 013737 010160 010076                MOV      TOTCC, TTOTCC
5900 055264 013737 010156 010100                MOV      CURADD, TCURAD
5901 055272 005237 010074                INC      TXMTOT
5902 055276 005337 003252                DEC      QUALVL
5903 055302 001266                BNE     5$
5904 055304 000137 054576                JMP      GETCL
5905
5906 055310 013737 010060 010160 2$:  MOV      CTOTCC, TOTCC
5907 055316 023727 010160 001000                CMP      TOTCC,#BUFLIM      ;SETUP CHAR. COUNT, CURR. ADDR. & PTR
5908 055324 002414                BLT      16$                ;SEE IF BUFFER ALREADY FULL
5909 055326                PRINTF  #MSGTRN,#BUFEX      ; BR IF NOT FULL (BUFLIM # OF CHARS.)
5910 055326 012746 017036                MOV      #BUFEX,-(SP)
5911 055332 012746 017054                MOV      #MSGTRN,-(SP)
5912 055336 012746 000002                MOV      #2,-(SP)
5913 055342 010600                MOV      SP,RO
5914 055344 104417                TRAP    C$PNTF
5915 055346 062706 000006                ADD      #6,SP

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 181
COMMAND LINE FETCH & INTERPRETATION SECTION

```

5916 055352 000137 054576
5917 055356 005737 010060
5918 055362 001002
5919 055364 005037 010056
5920 055370
5921 055370 012737 006502 010054
5922 055376 013701 010056
5923
5924 055402 020127 000017
5925 055406 002414
5926 055410
5927 055410 012746 016776
5928 055414 012746 017054
5929 055420 012746 000002
5930 055424 010600
5931 055426 104417
5932 055430 062706 000006
5933 055434 000137 054576
5934 055440 006301
5935 055442 006301
5936 055444 060137 010054
5937 055450 013737 010054 010154
5938 055456 013737 010062 010156
5939 055464 004737 027422
5940 055470 004737 027520
5941 055474 004737 027644
5942 055500 013737 010154 010054
5943 055506 005237 010056
5944 055512 013737 010156 010062
5945 055520 013737 010160 010060
5946 055526 005337 003252
5947 055532 001266
5948 055534 000137 054576
5949
5950
5951
5952
5953

16$: JMP GETCL ; THEN GO GET A NEW COMMAND
TST CTOTCC ; IF FIRST "SET" THEN GET RID OF DEFAULT
BNE 7$
CLR CMPTOT

7$: MOV #PTR13,CMPPTR ; INIT COMPARE MESSAGE POINTER
MOV CMPTOT,R1

CMP R1,#MSG LIM ; SEE IF MSG COUNT EXCEEDED.
BLT 18$ ; BR IF NO
PRINTF #MSGTRN,#TABEX ; ELSE TELL OPR. AND DON'T BUILD MSG.
MOV #TABEX,-(SP)
MOV #MSGTRN,-(SP)
MOV #2,-(SP)
MOV SP,R0
TRAP C$PNTF
ADD #6,SP

18$: JMP GETCL ; THEN GO GET A NEW COMMAND.
ASL R1 ; # OF MSGS *4 = NEXT FREE PTR BLOCK
ASL R1
ADD R1,CMPPTR
MOV CMPPTR,CPTR
MOV CCURAD,CURADD
JSR PC,ADDCC ; ADD IN XHAR. COUNT AND CHECK TOTAL
JSR PC,BLDBUF ; MAKE SURE ADDRESS IS WORD BOUNDARY
JSR PC,FCURAD
MOV CPTR,CMPPTR
INC CMPTOT
MOV CURADD,CCURAD ; UPDATE CHAR. COUNT, CURR ADDR. & PTR
MOV TOTCC,CTOTCC
DEC QUALVL ; IF COPY WAS GIVEN, PUT MSG IN BUFF
BNE 2$ ; AGAIN
JMP GETCL ; GO BACK UNTIL GET A "RUN"

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 182
COMMAND LINE FETCH & INTERPRETATION SECTION

```

5954
5955
5956
5957
5958 055540
5959 055540 006302
5960 055542 016202 055556
5961 055546 062702 055556
5962 055552 004712
5963 055554 000207
5964
5965
5966 055556 000162
5967 055560 000164
5968 055562 000174
5969 055564 001562
5970 055566 000274
5971 055570 000204
5972 055572 000320
5973 055574 000412
5974 055576 000734
5975 055600 000744
5976 055602 000762
5977 055604 000772
5978 055606 001002
5979 055610 001074
5980 055612 001570
5981 055614 001114
5982 055616 001174
5983 055620 001202
5984 055622 001212
5985 055624 001222
5986 055626 001232
5987 055630 001242
5988 055632 001260
5989 055634 001346
5990 055636 001356
5991 055640 001376
5992 055642 001404
5993 055644 001414
5994 055646 001424
5995 055650 001434
5996 055652 001462
5997 055654 001472
5998 055656 001576
5999 055660 001612
6000 055662 001644
6001 055664 001654
6002 055666 001664
6003 055670 001674
6004 055672 001704
6005 055674 001714
6006 055676 000154
6007 055700 001152
6008 055702 000670
6009 055704 000720

```

.SBTTL ACTION TABLE AND ROUTINES
: USER MUST CLEAR/SET P\$GDBD IF USE "CLIBIF" IN CONNECTION WITH ACTION
: R2 WILL HOLD ACTION CODE FROM PARSING (CLI) NODE

```

:CLIACT:
ASL R2 :MULTIPLY ACTION CODE BY 2
MOV 10$(R2),R2 :OFFSET VALUE
ADD #10$,R2 :ADD BASE VALUE
JSR PC,(R2) :GO DO ACTION
RTS PC :RETURN TO TRVACT:

```

10\$:

```

.WORD ACTNUL-10$ :BRIEF DESCRIPTION OF ACTIONS TAKEN
.WORD ACTCLR-10$ :NULL
.WORD ACTSHO-10$ :CLEAR
.WORD ACTCHK-10$ :SHOW
.WORD ACTRUN-10$ :CHECK
.WORD ACTHLP-10$ :RUN
.WORD ACTCSE-10$ :HELP
.WORD ACTCST-10$ :CLEAR OR SHOW EXPECTED
.WORD ACTSTE-10$ :CLEAR OR SHOW TRANSMIT
.WORD ACTSTT-10$ :SET EXPECTED
.WORD ACTSIZE-10$ :SET TRANSMIT
.WORD ACTCOP-10$ :SIZE
.WORD ACTNUM-10$ :COPY
.WORD ACTOPM-10$ :NUMERIC VALUE FOR SIZE OR COPY
.WORD ACTSTS-10$ :QUOTED MESSAGE FROM USER
.WORD ACTEQO-10$ :STATUS
.WORD ACTMSO-10$ :END OF QUOTED MESSAGE FROM USER
.WORD ACTMS1-10$ :ONES DATA
.WORD ACTMS2-10$ :ZEROS DATA
.WORD ACTMS3-10$ :1ALT
.WORD ACTMS4-10$ :OACT
.WORD ACTMS5-10$ :ITEP
.WORD ACTMS6-10$ :CCITT
.WORD ACTATV-10$ :ALPHA
.WORD ACTPAS-10$ :ACTIVE MODE
.WORD ACTREC-10$ :PASSIVE MODE
.WORD ACTLIS-10$ :RECEIVE MODE
.WORD ACTDLL-10$ :LISTEN MODE
.WORD ACTTRA-10$ :DOWNLINE LOAD
.WORD ACTTAL-10$ :TRANSMIT MODE
.WORD ACTNO-10$ :TALK MODE
.WORD ACTECH-10$ :/NO
.WORD ACTCRC-10$ :ECHO
.WORD ACTPRO-10$ :SET CRC BIT
.WORD ACTRPS-10$ :SET PROTOCOL BIT
.WORD ACTMOP-10$ :STATUS
.WORD ACTTLP-10$ :REMOTE STATION IN MAINTENACE LOOP MODE
.WORD ACTCLP-10$ :INTERNAL TTL
.WORD ACTLLP-10$ :CABLE LOOP
.WORD ACTRLP-10$ :LOCAL MODEM LOOP
.WORD ACTNUF-10$ :REMOTE MODEM LOOP
.WORD ACTBCR-10$ :MORE COMMAND LINE NEEDED
.WORD ACTDMS-10$ :BAD CHARACTER IN OPERATOR MESSAGE
.WORD ACTDME-10$ :DUMP MEMORY START ADDRESS

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 183
ACTION TABLE AND ROUTINES

6010	055706	000712
6011	055710	000260
6012	055712	001604
6013	055714	000250
6014	055716	001304
6015	055720	002070
6016	055722	001772
6017	055724	002002
6018	055726	002010
6019	055730	002266
6020		

.WORD	ACTDMQ-10\$:DUMP WORD
.WORD	ACTPRT-10\$:PRINT
.WORD	ACTMOS-10\$:MODEM STATUS CHANGE
.WORD	ACTEXT-10\$:EXIT
.WORD	ACTSEX-10\$:SET E=T COMMAND
.WORD	ACTSLS-10\$:SHOW LINE LIST
.WORD	ACTETB-10\$:ESTABLISH LINE
.WORD	ACTKTB-10\$:KILL LINE
.WORD	ACTKAL-10\$:KILL ALL
.WORD	ACTEKT-10\$:FLAG LINE KILLED

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 184
ACTION TABLE AND ROUTINES

6021										
6022	055732	112737	177777	003400	ACTNUF:	MOVB	#-1,P\$NNUF			:SET FLAG TO SAY NEED MORE OF COMMAND
6023	055740	000207			ACTNUL:	RTS	PC			:RETURN TO PARSER
6024										
6025	055742	012737	000001	003246	ACTCLR:	MOV	#CLEAR,KEYWD1			:SET LOC TO SAY A CLEAR WAS TYPED
6026	055750	000207				RTS	PC			
6027										
6028	055752	012737	000002	003246	ACTSHO:	MOV	#SHOW,KEYWD1			:SET LOC. TO SAY A SHOW WAS TYPED
6029	055760	000207				RTS	PC			
6030										
6031	055762	012702	003254		ACTHLP:	MOV	#HLPTAB,R2			:SETUP R2 AS A POINTER TO HELP MSG TABLE
6032	055766				1\$:	PRINTF	#HLPF,(R2)+			:PRINT HELP INFORMATION MESSAGES
6033	055766	012246								MOV (R2)+,-(SP)
6034	055770	012746	014616							MOV #HLPF,-(SP)
6035	055774	012746	000002							MOV #2,-(SP)
6036	056000	010600								MOV SP,R0
6037	056002	104417								TRAP C\$PNTF
6038	056004	062706	000006							ADD #6,SP
6039	056010	020227	003276			CMP	R2,#HLPEND			:SEE IF ALL INFO PRINTED YET
6040	056014	001364				BNE	1\$:IF NO KEEP PRINTING
6041	056016	012737	000005	003246		MOV	#HLP,KEYWD1			:SET LOC. TO SAY A HELP WAS TYPED
6042	056024	000207				RTS	PC			
6043	056026	012737	000057	003246	ACTEXT:	MOV	#EXIT,KEYWD1			
6044	056034	000207				RTS	PC			:SET UP KEYWORD AND SCOOT OUT OF HERE
6045	056036	012737	000055	003246	ACTPRT:	MOV	#PRNT,KEYWD1			:SET LOC. TO SAY A HELP WAS TYPED
6046	056044	004737	024644			JSR	PC,REPORT			:CALL ROUTINE TO PRINT EVENT LOG AND BASE TABLE
6047	056050	000207				RTS	PC			
6048										
6049	056052	012737	000004	003246	ACTRUN:	MOV	#RUN,KEYWD1			:SET RUN FLAG
6050	056060	112737	177777	003400		MOVB	#-1,P\$NNUF			:SET FLAG TO SAY NEED MORE OF COMMAND
6051	056066	012737	000001	010226		MOV	#1,RPASS			:SET DEFAULT RUN 'PASS' TO 1
6052	056074	000207				RTS	PC			
6053										
6054	056076	012737	006502	010054	ACTCSE:	MOV	#PTR13,CMPPTR			:INIT COMPARE MESSAGE POINTER
6055	056104	013701	010054			MOV	CMPPTR,R1			
6056										
6057	056110	013702	010056			MOV	CMPTOT,R2			
6058	056114	105037	003400			CLRB	P\$NNUF			:FLAG THAT HAVE VALID COMMAND AT THIS PT.
6059	056120	023727	003246	000002		CMP	KEYWD1,#SHOW			:SEE IF A CLEAR OR SHOW WAS TYPED
6060	056126	001471				BEQ	ACTSHW			:BR IF A SHOW WAS TYPED
6061	056130	012737	000001	010056		MOV	#1,CMPTOT			:CLEAR COMPARE MESSAGE COUNT, CHAR. COUNT
6062	056136	005037	010060			CLR	CTOTCC			: AND RESET POINTER
6063										
6064	056142	012737	006502	010054		MOV	#PTR13,CMPPTR			:INIT COMPARE MESSAGE POINTER
6065	056150	013737	010054	010154		MOV	CMPPTR,CPTR			:SET UP TO FILL IN DEFAULT MESSAGE
6066	056156	012701	004406			MOV	#CMPBUF,R1			
6067	056162	010137	010062			MOV	R1,CCURAD			
6068	056166	000431				BR	ACTCLB			
6069										
6070	056170	012701	006406		ACTCST:	MOV	#PTRTAB,R1			
6071	056174	013702	010074			MOV	TXMTOT,R2			
6072	056200	105037	003400			CLRB	P\$NNUF			:FLAG THAT HAVE VALID COMMAND AT THIS PT.
6073	056204	023727	003246	000002		CMP	KEYWD1,#SHOW			:SEE IF A CLEAR OR SHOW WAS TYPED
6074	056212	001437				BEQ	ACTSHW			:BR IF A SHOW WAS TYPED
6075	056214	012737	000001	010074		MOV	#1,TXMTOT			:CLEAR TRANSMIT MESSAGE COUNT, CHAR. COUNT
6076	056222	005037	010076			CLR	TTOTCC			: AND RESET POINTER

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 185
ACTION TABLE AND ROUTINES

6077	056226	012737	006406	010052		MOV	#PTRTAB, TXPTR			
6078	056234	013737	010052	010154		MOV	TXPTR, CPTR			
6079	056242	012701	003406			MOV	#TXBUF, R1			
6080	056246	010137	010100			MOV	R1, TCRAD			
6081										
6082	056252	012702	001000		ACTCLB:	MOV	#BUFLIM, R2			
6083	056256	010137	010156			MOV	R1, CURADD		;SET UP TO PUT DEFAULT MSG IN LIST AFTER 033'S	
6084	056262	012737	000005	010146		MOV	#5, MSGTYP			
6085	056270	013737	002156	010150		MOV	MSG5C, CURCC			
6086	056276	105021			1\$:	CLRB	(R1)+		;FILL EXPT OR TRAN BUFFER WITH 0'S IF A CLEAR	
6087	056300	005302				DEC	R2		;DO 'BUFLIM' NUMBER OF BYTE LOCATIONS	
6088	056302	001375				BNE	1\$			
6089	056304	004737	027520			JSR	PC, BLDBUF		;'CLEAR' REALLY MEANS TO PUT DEFAULT MSG IN	
6090	056310	000207				RTS	PC		;WHEN DONE, RETURN TO PARSER	
6091										
6092										
6093	056312	012705	003324		ACTSHW:	MOV	#SHTAB, R5			
6094	056316	122571	000000		5\$:	CMPB	(R5)+, @ (R1)		;LOOK AT FIRST BYTE OF MSG TO DECIPHER TYPE	
6095	056322	001404				BEQ	6\$			
6096	056324	020527	003333			CMP	R5, #SHTEND		;SEE IF LOOKED AT ALL OF DEFAULTS YET	
6097	056330	001372				BNE	5\$			
6098	056332	005205				INC	R5		;MUST BE OPR. SPEC'D THEN	
6099	056334	162705	003325		6\$:	SUB	#SHTAB+1, R5			
6100	056340	006305				ASL	R5			
6101	056342	016137	000002	010164		MOV	2(R1), TEMP			
6102	056350					PRINTF	#SHMSG, SHTYTB(R5), TEMP		;PRINT MSG SIZE & TYPE	
6103	056350	013746	010164						MOV	TEMP, -(SP)
6104	056354	016546	003304						MOV	SHTYTB(R5), -(SP)
6105	056360	012746	015664						MOV	#SHMSG, -(SP)
6106	056364	012746	000003						MOV	#3, -(SP)
6107	056370	010600							MOV	SP, R0
6108	056372	104417							TRAP	C\$PNTF
6109	056374	062706	000010						ADD	#10, SP
6110	056400	062701	000004			ADD	#4, R1			;BUMP R1 TO NEXT SET OF POINTERS
6111	056404	005302				DEC	R2			
6112	056406	001341				BNE	ACTSHW			
6113	056410	013737	010216	011470		MOV	MODTYP, DEV1			
6114	056416	013737	010220	011472		MOV	MLTYP, DEV2			
6115	056424	013737	010226	011474		MOV	RPASS, DEV3			
6116	056432	013737	010224	011476		MOV	PARAM, DEV4			
6117	056440	004737	031204			JSR	PC, SHWOP			;SHOW THE OPERATOR THE CURRENT MODE..... ALSO
6118	056444	000207				RTS	PC			
6119										
6120	056446	013737	003374	010126	ACTDMS:	MOV	P\$NUM, STADD			;SETUP STARTING ADDRESS FOR DUMP
6121	056454	005037	010132			CLR	BYTBIT			;SET DEFAULT OF WORD DUMP
6122	056460	012737	000052	003246		MOV	#DMPS, KEYWD1			;FLAG THAT A DUMP WAS TYPED
6123	056466	000403				BR	ACTDME			
6124										
6125	056470	012737	177777	010132	ACTDMQ:	MOV	#-1, BYTBIT			;SET DUMP FLAG TO 'DUMP-WORD'
6126	056476	013737	003374	010130	ACTDME:	MOV	P\$NUM, ENADD			;SETUP END ADDRESS FOR DUMP (=START IF NO 'EEE')
6127	056504	105037	003400		ACTDMX:	CLRB	P\$NUF			;CLEAR NOT-ENOUGH FLAG, 'DUMP N-N/B' IS VALID
6128	056510	000207				RTS	PC			
6129										
6130										
6131										
6132	056512	012737	000010	003246	ACTSTE:	MOV	#SETEXP, KEYWD1			

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 186
ACTION TABLE AND ROUTINES

6133	056520	000403			BR	ACTSTX			
6134									
6135	056522	012737	000011	003246	ACTSTT:	MOV	#SETTRN,KEYWD1		
6136	056530	012737	000001	003252	ACTSTX:	MOV	#1,QUALVL	;	SET UP DEFAULT COPY TO 1 (/COPY=0)
6137	056536	000207				RTS	PC		
6138									
6139	056540	012737	000012	003250	ACTSIZE:	MOV	#SIZE,QUALFG		
6140	056546	000207				RTS	PC		
6141									
6142	056550	012737	000013	003250	ACTCOP:	MOV	#QCOPY,QUALFG		
6143	056556	000207				RTS	PC		
6144									
6145	056560	023727	003250	000012	ACTNUM:	CMP	QUALFG,#SIZE	;	SEE IF A SIZE OR COPY TYPED
6146	056566	001023				BNE	1\$;	BR IF IT WAS A COPY
6147	056570	005737	003374			TST	PSNUM	;	CHECK TO BE SURE DIDN'T TRY SIZE=0
6148	056574	001014				BNE	3\$;	BR IF NO
6149	056576					PRINTF	#CLISEO		
6150	056576	012746	014417					MOV	#CLISEO,-(SP)
6151	056602	012746	000001					MOV	#1,-(SP)
6152	056606	010600						MOV	SP,RO
6153	056610	104417						TRAP	C\$PNTF
6154	056612	062706	000004					ADD	#4,SP
6155	056616	112737	177777	003401		MOVB	#-1,PSGDES	;	SET ERROR-IN-CMD FLAG
6156	056624	000411				BR	2\$		
6157	056626	013737	003374	010150	3\$:	MOV	PSNUM,CURCC	;	IF A SIZE LOAD CURCC WITH BYTE COUNT
6158	056634	000405				BR	2\$		
6159	056636	013737	003374	003252	1\$:	MOV	PSNUM,QUALVL	;	IF A COPY, LOAD COPY COUNT
6160	056644	005237	003252			INC	QUALVL	;	INCREMENT SO FIRST DEC MAKES IT REAL #
6161	056650	000522			2\$:	BR	ACTMEX		
6162									
6163	056652	012737	000007	010146	ACTOPM:	MOV	#7,MSGTYP		
6164	056660	010437	010164			MOV	R4,TEMP	;	KEEP TRACK OF START OF QUOTED TEXT
6165	056664	005237	010164			INC	TEMP	;	SO CAN CALC OPCNT AT END OF QUOTES
6166	056670	000207				RTS	PC		
6167									
6168	056672	010402			ACTEQO:	MOV	R4,R2		
6169	056674	163702	010164			SUB	TEMP,R2		
6170	056700	010237	010150			MOV	R2,CURCC	;	CALC BYTE COUNT FOR QUOTED TEXT
6171	056704	010237	002162			MOV	R2,OPCNT		
6172	056710	013701	010164			MOV	TEMP,R1		
6173	056714	012705	002520			MOV	#OPBUF,R5		
6174	056720	112125			1\$:	MOVB	(R1)+,(R5)+	;	COPY QUOTED TEXT TO OPBUF
6175	056722	005302				DEC	R2		
6176	056724	001375				BNE	1\$		
6177	056726	000473				BR	ACTMEX		
6178									
6179	056730				ACTBCR:	PRINTF	#CLIBCR	;	BAD CHAR. IN OPR. QUOTED STRING
6180	056730	012746	014352					MOV	#CLIBCR,-(SP)
6181	056734	012746	000001					MOV	#1,-(SP)
6182	056740	010600						MOV	SP,RO
6183	056742	104417						TRAP	C\$PNTF
6184	056744	062706	000004					ADD	#4,SP
6185	056750	000207				RTS	PC		
6186									
6187	056752	005037	010146		ACTMSO:	CLR	MSGTYP		
6188	056756	000435				BR	ACTME1		

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 187
ACTION TABLE AND ROUTINES

6189	056760	012737	000001	010146	ACTMS1: MOV	#1,MSGTYP	
6190	056766	000431			BR	ACTME1	
6191	056770	012737	000002	010146	ACTMS2: MOV	#2,MSGTYP	
6192	056776	000425			BR	ACTME1	
6193	057000	012737	000003	010146	ACTMS3: MOV	#3,MSGTYP	
6194	057006	000421			BR	ACTME1	
6195	057010	012737	000004	010146	ACTMS4: MOV	#4,MSGTYP	
6196	057016	000415			BR	ACTME1	
6197	057020	012737	000005	010146	ACTMS5: MOV	#5,MSGTYP	
6198	057026	013737	002156	010150	MOV	MSG5C,CURCC	;SETUP DEFAULT SIZE FOR THIS TYPE
6199	057034	000430			BR	ACTMEX	
6200	057036	012737	000006	010146	ACTMS6: MOV	#6,MSGTYP	
6201	057044	013737	002160	010150	MOV	MSG6C,CURCC	;SETUP DEFAULT SIZE FOR THIS TYPE
6202							
6203	057052	012737	000100	010150	ACTME1: MOV	#64,CURCC	;SETUP DEFAULT SIZE FOR MSG0-4
6204	057060	000416			BR	ACTMEX	;EXIT
6205							
6206							
6207	057062	022737	000010	003246	ACTSEX: CMP	#SETEXP,KEYWD1	;DID WE GET HERE FROM "SET E =" COMMAND?
6208	057070	001404			BEQ	10\$;YES,BRANCH
6209	057072	112737	177777	003401	MOVB	#-1,P\$GDBD	;SET ERROR FLAG
6210	057100	000406			BR	ACTMEX	;GO TO EXIT
6211	057102	004737	027672		JSR	PC,FACSIMILE	;GO COPY TRANMIT BUFFER TO EXPECT BUFFER
6212	057106	012737	000060	003246	MOV	#SETET,KEYWD1	;SET FLAG TO BE USED IN T1::
6213	057114	000400			BR	ACTMEX	;GO TO EXIT
6214							
6215							
6216							
6217	057116	105037	003400		ACTMEX: CLRB	P\$NNUF	;CLEAR NOT-ENOUGH FLAG
6218	057122	000207			RTS	PC	
6219							

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 188
ACTION TABLE AND ROUTINES

6220	057124	012737	000003	010216	ACTATV: MOV	#ACT,MODTYP	
6221	057132	000432			BR	ACTM2X	
6222							
6223	057134	012737	000002	010216	ACTPAS: MOV	#PAS,MODTYP	
6224	057142	105037	003400		CLRB	PSNNUF	:CLEAR NOT-ENOUGH FLAG
6225	057146	005037	010220		CLR	MLTYP	:CLEAR MAINT LOOP TYPE
6226	057152	000207			RTS	PC	
6227							
6228	057154	005037	010216		ACTREC: CLR	MODTYP	
6229	057160	000417			BR	ACTM2X	
6230							
6231	057162	012737	000006	010216	ACTLIS: MOV	#LIS,MODTYP	
6232	057170	000413			BR	ACTM2X	
6233							
6234	057172	012737	000004	010216	ACTDLL: MOV	#DOW,MODTYP	
6235	057200	000407			BR	ACTM2X	
6236							
6237	057202	012737	000001	010216	ACTTRA: MOV	#TRA,MODTYP	
6238	057210	000403			BR	ACTM2X	
6239							
6240	057212	012737	000005	010216	ACTTAL: MOV	#TAL,MODTYP	
6241							
6242	057220	042737	000004	010224	ACTM2X: BIC	#ECHOB,PARAM	:DISABLE /ECHO (ALL BUT PASSIVE MODE)
6243	057226	105037	003400		CLRB	PSNNUF	:CLEAR NOT-ENOUGH FLAG
6244	057232	005037	010220		CLR	MLTYP	:CLEAR MAINT LOOP TYPE
6245	057236	000207			RTS	PC	
6246							

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 189
ACTION TABLE AND ROUTINES

6247	057240	012737	000036	003250	ACTNO:	MOV	#NO,QUALFG	
6248	057246	000207				RTS	PC	
6249								
6250	057250	022737	000036	003250	ACTECH:	CMP	#NO,QUALFG	
6251	057256	001422				BEQ	1\$	
6252	057260	052737	000004	010224		BIS	#ECHOB,PARAM	
6253	057266	022737	000002	010216		CMP	#PAS,MODTYP	
6254	057274	001416				BEQ	2\$	
6255	057276					PRINTF	#CLINPS	
6256	057276	012746	014307					
6257	057302	012746	000001					
6258	057306	010500						
6259	057310	104417						
6260	057312	062706	000004					
6261	057316	112737	177777	003401		MOVB	#-1,P\$GDBD	
6262	057324	042737	000004	010224	1\$:	BIC	#ECHOB,PARAM	
6263	057332	005037	003250		2\$:	CLR	QUALFG	
6264	057336	000501				BR	ACTLXX	
6265								
6266	057340	012701	000002		ACTCHK:	MOV	#DATCKB,R1	
6267	057344	000413				BR	ACTQFG	
6268								
6269	057346	012701	000001		ACTSTS:	MOV	#STATB,R1	
6270	057352	000410				BR	ACTQFG	
6271								
6272	057354	012701	000020		ACTCRC:	MOV	#CRCB,R1	
6273	057360	000405				BR	ACTQFG	
6274								
6275	057362	012701	000010		ACTMOS:	MOV	#MOCHK,R1	
6276	057366	000402				BR	ACTQFG	
6277								
6278	057370	012701	000040		ACTPRO:	MOV	#PROTOB,R1	
6279								
6280	057374	050137	010224		ACTQFG:	BIS	R1,PARAM	
6281	057400	022737	000036	003250		CMP	#NO,QUALFG	
6282	057406	001002				BNE	1\$	
6283	057410	040137	010224			BIC	R1,PARAM	
6284	057414	005037	003250		1\$:	CLR	QUALFG	
6285	057420	000450				BR	ACTLXX	
6286								
6287	057422	013737	003374	010226	ACTRPS:	MOV	P\$NUM,RPASS	
6288	057430	000444				BR	ACTLXX	
6289								
6290	057432	012737	000005	010220	ACTMOP:	MOV	#5,MLTYP	
6291	057440	000417				BR	ACTLPX	
6292	057442	012737	000001	010220	ACTTLP:	MOV	#1,MLTYP	
6293	057450	000413				BR	ACTLPX	
6294	057452	012737	000002	010220	ACTCLP:	MOV	#2,MLTYP	
6295	057460	000407				BR	ACTLPX	
6296	057462	012737	000003	010220	ACTLLP:	MOV	#3,MLTYP	
6297	057470	000403				BR	ACTLPX	
6298	057472	012737	000004	010220	ACTRLP:	MOV	#4,MLTYP	
6299								
6300	057500	022737	000003	010216	ACTLPX:	CMP	#ACT,MODTYP	
6301	057506	001415				BEQ	ACTLXX	
6302	057510	112737	177777	003401		MOVB	#-1,P\$GDBD	

MOV #CLINPS,-(SP)
 MOV #1,-(SP)
 MOV SP,R0
 TRAP C\$PNTF
 ADD #4,SP

;BE SURE IN PASSIVE MODE IF
 ;IF TRYING TO SET /ECHO

;CLEAR 'NO' OUT OF QUALIFIER FLAG

;SET DATA CHECK BIT

;SET THE STATUS BIT

;SET THE CRC BIT

;SET THE MODEM BIT

;SET THE PROTOCOL BIT

;CLEAR 'NO' OUT OF QUALIFIER FLAG

;GET NUMBER OF 'RUN PASSES'

;SPECIFY MOP LOOPBACK

;SPECIFY INTERNAL LOOPBACK

;SPECIFY CABLE LOOPBACK

;SPECIFY LOCAL MODEM LOOPBACK

;SPECIFY REMOTE MODEM LOOPBACK

;BE SURE IN ACTIVE IF TRYING TO SET LOOP
 ; BR IF IN ACTIVE

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 190
ACTION TABLE AND ROUTINES

6303	057516	005037	010220
6304	057522		
6305	057522	012746	014125
6306	057526	012746	000001
6307	057532	010600	
6308	057534	104417	
6309	057536	062706	000004
6310	057542	105037	003400
6311	057546	000207	
6312			

CLR MLTYP ;CLEAR ANY LOOP TYPE THAT MAY HAVE GOT SET
PRINTF #CLIBDL

MOV #CLIBDL,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C\$PNTF
ADD #4,SP

ACTLXX: CLRB P\$NNUF ;CLEAR NOT-ENOUGH FLAG
RTS PC

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 191
ACTION TABLE AND ROUTINES

6313									
6314	057550	012737	000062	003246	ACTETB: MOV	#ETRB,KEYWD1		: RECORD THAT ESTABLISH TYPED	
6315	057556	000207			RTS	PC		: RETURN TO CALL	
6316									
6317	057560	012737	000063	003246	ACTKTB: MOV	#KTRB,KEYWD1		: RECORD THAT KILL LINE TYPED	
6318									
6319	057566	105037	003400		ACTKAL: CLR	PSNNUF		: CLEAR INCOMPLETE INFO FLAG	
6320	057572	022737	000063	003246	CMP	#KTRB,KEYWD1		: BE SURE "ALL" IS AFTER A "KILL"	
6321	057600	001403			BEQ	11\$: BR IF YES	
6322	057602	112737	177777	003401	MOV	#-1,PSGDBD		: ELSE ERROR IN CMD	
6323	057610	105737	003401		TST	PSGDBD		: SEE IF WAS AN ERROR FROM ..KTB	
6324	057614	001401			BEQ	10\$: BR IF NO	
6325	057616	000412			BR	2\$: ELSE EXIT	
6326	057620	005037	007634		10\$: CLR	TRBTOT		: ZERO TOTAL # OF LINE NUMBERS	
6327	057624	012702	007622		MOV	#TRIBLS,R2		: PT R2 TO LINE NUMBER TABLE	
6328	057630	012705	000004		MOV	#4.,R5		: SETUP R5 AS COUNTER	
6329	057634	012722	177777		1\$: MOV	#-1,(R2)+		: CLEAR 8 BYTES OF TABLE	
6330	057640	005305			DEC	R5			
6331	057642	001374			BNE	1\$			
6332	057644	000207			2\$: RTS	PC		: RETURN TO CALL	
6333									
6334	057646	105037	003400		ACTSLS: CLR	PSNNUF		: CLEAR THE INCOMPLETE CMD FLAG	
6335	057652	012737	000002	003246	MOV	#SHOW,KEYWD1		: SET UP TO LOOK LIKE A SHOW CMD	
6336	057660	105737	003401		TST	PSGDBD		: SEE IF WAS AN ERROR FROM ..KTB	
6337	057664	001401			BEQ	10\$: BR IF NO	
6338	057666	000465			BR	5\$: ELSE EXIT	
6339	057670	005037	010114		10\$: CLR	LNCNT		: INIT ADDR/LINE COUNTER	
6340	057674	005737	007634		TST	TRBTOT		: SEE IF LIST EMPTY	
6341	057700	001011			BNE	1\$: BR IF NO	
6342	057702				PRINTS	#SHTRE		: PRINT THE LINE LIST IS EMPTY	
6343	057702	012746	016000					MOV	#SHTRE,-(SP)
6344	057706	012746	000001					MOV	#1,-(SP)
6345	057712	010600						MOV	SP,R0
6346	057714	104416						TRAP	C\$PNTS
6347	057716	062706	000004					ADD	#4,SP
6348	057722	000437							
6349	057724	012702	007622		1\$: BR	4\$: POINT R2 TO THE LINE NUMBER LIST	
6350	057730	012705	000010		MOV	#TRIBLS,R2		: SETUP R5 AS A COUNTER	
6351	057734				MOV	#8.,R5		: PRINT LINE LIST HEADER	
6352	057734	012746	016042		PRINTS	#SHTRH			
6353	057740	012746	000001					MOV	#SHTRH,-(SP)
6354	057744	010600						MOV	#1,-(SP)
6355	057746	104416						MOV	SP,R0
6356	057750	062706	000004					TRAP	C\$PNTS
6357	057754	122712	000377					ADD	#4,SP
6358	057760	001415			2\$: CMPB	#377,(R2)		: SEE IF A NULL ENTRY	
6359	057762	111237	010164		BEQ	3\$: BR IF YES	
6360	057766				MOVB	(R2),TEMP			
6361	057766	005046			PRINTS	#SHTAP,<B,TEMP>			
6362	057770	153716	010164					CLR	-(SP)
6363	057774	012746	016066					BISB	TEMP,(SP)
6364	060000	012746	000002					MOV	#SHTAP,-(SP)
6365	060004	010600						MOV	#2,-(SP)
6366	060006	104416						MOV	SP,R0
6367	060010	062706	000006					TRAP	C\$PNTS
6368	060014	005202			3\$: INC	R2		: INCREMENT TABLE ADDRESS	

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 192
ACTION TABLE AND ROUTINES

```

6369 060016 005305
6370 060020 001355
6371 060022
6372 060022 012746 021650
6373 060026 012746 000001
6374 060032 010600
6375 060034 104416
6376 060036 062706 000004
6377 060042 000207
6378
6379 060044
6380 060044 005037 003404
6381 060050 105037 003400
6382 060054 105737 003401
6383 060060 001401
6384 060062 000443
6385
6386 060064 013701 003374
6387 060070 022701 000007
6388 060074 103012
6389 060076
6390 060076 010146
6391 060100 012746 016146
6392 060104 012746 000002
6393 060110 010600
6394 060112 104416
6395 060114 062706 000006
6396 060120 000424
6397
6398 060122 022737 000062 003246 2$:
6399 060130 001401
6400 060132 000615
6401
6402 060134
6403
6404 060134 012702 007622
6405 060140 110112
6406 060142 012737 000001 007634
6407 060150 162702 007622
6408
6409 060154 012737 177777 003404
6410 060162 010237 007642
6411 060166 010137 007640
6412
6413 060172
6414 060172 000207
6415
6416
6417

4$: DEC R5 ; SEE IF CHECKED ALL OF LIST
    BNE 2$ ; BR BACK IF NO
    PRINTS #CR ; ELSE PRINT A PARTING CR
    MOV #CR,-(SP)
    MOV #1,-(SP)
    MOV SP,R0
    TRAP C$PNTS
    ADD #4,SP

5$: RTS PC ;RETURN TO CALL

ACTEKT:
    CLR VALTRB ;INIT. VALID LINE FLAG
    CLR PSNUF ;CLEAR NOT ENOUGH INFO FLAG
    TSTB PSGDBD ; SEE IF WAS AN ERROR FROM ..KTB
    BEQ 10$ ; BR IF NO
    BR ACTEXX ; ELSE EXIT

10$: MOV PSNUM,R1
    CMP #7,R1 ;SEE THAT ITS A VALID LINE NUM (0-7)
    BHIS 2$
    PRINTS #SHTIV,R1

1$: MOV R1,-(SP)
    MOV #SHTIV,-(SP)
    MOV #2,-(SP)
    MOV SP,R0
    TRAP C$PNTS
    ADD #6,SP

BR ACTEXX

2$: CMP #ETRB,KEYWD1 ; SEE IF KILL OR ESTABLISH
    BEQ ACTEKE ; BR IF WAS AN ESTABLISH
    BR ACTKAL ; KILL LINE **827 ONLY ONE POSSIBLE

ACTEKE:
    MOV #TRIBLS,R2 ; ONCE CHECKED LIST
    MOV R1,(R2) ; LOAD LINE NUMBER IN 1ST SLOT DCLT
    MOV #1,TRBTOT ; SPECIFY TOTAL # LINES = 1 DCLT
    SUB #TRIBLS,R2 ;SUBTRACT START OF LIST FROM POINT TO
    GET INDEX
    SET VALID LINE FLAG
    MOV R2,INDEX ;MOVE R2 TO INDEX
    MOV R1,INDW ;MOVE LINE NUMBER TO INDW

ACTEXX: RTS PC ;RETURN TO CALL

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 193
SETUP FOR RECEIVER ALLOCATIONS

```

6418          .SBTTL          SETUP FOR RECEIVER ALLOCATIONS
6419
6420          ;START EXECUTION OF RUN COMMAND
6421 060174    GTR9:         ; RX ALLOCATE CODE
6422
6423
6424 060174 032737 000002 010224    BIT    #DATCKB,PARAM    ;IS THIS DATA CHECK
6425 060202 001421                BEQ    44$              ;BRANCH IF NO
6426 060204 005737 010220    TST    MLTYP
6427 060210 001416                BEQ    44$              ;BRANCH IF NOT LOOP
6428 060212 023737 010056 010074    CMP    CMPTOT, TXMTOT    ;ARE TX AND EX EQUAL
6429 060220 001412                BEQ    44$              ;BRANCH IF YES
6430 060222                PRINTF #CLIPW          ;PRINT MESSAGE TO OPERATOR
6431 060222 012746 014450                MOV    #CLIPW, -(SP)
6432 060226 012746 000001                MOV    #1, -(SP)
6433 060232 010600                MOV    SP, R0
6434 060234 104417                TRAP  C$PNTF
6435 060236 062706 000004                ADD    #4, SP
6436 060242 000137 054576    JMP    GETCL            ;GO GET NEXT COMMAND
6437
6438 060246    44$:         ;CHECK FOR AT LEAST ONE LINE ENABLED
6439 060246 005737 007634    TST    TRBTOT          ;IS LINE TOTAL
6440 060252 001013                BNE    4$              ;ZERO?..BR IF NOT
6441 060254                PRINTF #SHTLPA        ;PRINT ERROR MUST ESTABLISH LINE
6442 060254 012746 016072                MOV    #SHTLPA, -(SP)
6443 060260 012746 000001                MOV    #1, -(SP)
6444 060264 010600                MOV    SP, R0
6445 060266 104417                TRAP  C$PNTF
6446 060270 062706 000004                ADD    #4, SP
6447 060274 112737 177777 003401    MOVB   #-1, P$GDBD     ;SET ERROR FLAG
6448 060302    4$:         ;CHECK ERROR FLAG
6449 060302 105737 003401    TSTB   P$GDBD         ;TEST ERROR FLAG
6450 060306 001043                BNE    12$            ;BRANCH IF ERROR
6451
6452 060310 013737 010060 010150    MOV    CTOTCC, CURCC   ;CHECK IF DEFAULT RUN COMMAND
6453 060316 005737 010150    TST    CURCC           ;MAKE CURRENT COUNT= COMPARE COUNT
6454 060322 001003                BNE    1$              ;TEST TOTAL COMPARE COUNT
6455 060324 012737 000072 010150    MOV    #58., CURCC    ;BRANCH IF NON DEFAULT
6456 060332    1$:         ;SET UP DEFAULT
6457 060332 032737 000002 010224    BIT    #DATCKB,PARAM
6458 060340 001430                BEQ    2$              ;BRANCH IF NOT CHECKING
6459 060342 013737 007634 010046    MOV    TRBTOT, MPLY
6460 060350 013737 010150 010164    MOV    CURCC, TEMP
6461 060356 005037 010170    CLR    TEMP2
6462 060362 004737 030356    JSR    PC, MPLY        ;MULTIPY TRBTOT BY CURCC
6463                                ;RESULT IN TEMP2
6464 060366 022737 001000 010170    CMP    #BUFLIM, TEMP2  ;IS IT MUCH TO MUCH
6465 060374 002012                BGE    2$              ;NO EVERTHING IS HUNKY DORY
6466 060376                PRINTF #SHTBR        ;ERROR
6467 060376 012746 016237                MOV    #SHTBR, -(SP)
6468 060402 012746 000001                MOV    #1, -(SP)
6469 060406 010600                MOV    SP, R0
6470 060410 104417                TRAP  C$PNTF
6471 060412 062706 000004                ADD    #4, SP
6472 060416 000137 054576    12$:   JMP    GETCL            ;GO BACK TO GET NEW COMMAND
6473

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 194
SETUP FOR RECEIVER ALLOCATIONS

```

6474 060422
6475 060422 012737 006406 010052
6476 060430 012737 006502 010054
6477 060436 012737 006576 010050
6478 060444 013737 010056 010112
6479 060452 032737 000002 010224
6480 060460 001003
6481 060462 012737 000001 010112
6482 060470 005037 010134
6483 060474 005037 007644
6484 060500 005037 007646
6485 060504 005037 013634
6486 060510 005037 013636
6487 060514 005037 013640
6488 060520 005037 013642
6489 060524 012737 007672 007650
6490 060532 012737 007672 007652
6491 060540 012737 007656 007654
6492 060546 005037 010230
6493 060552 005037 010116
6494 060556 005037 010120
6495 060562 005037 010122
6496 060566 005037 010124
6497 060572 005037 010114
6498
6499 060576 004737 024244
6500 060602 004737 063024
6501
6502 060606 012737 177777 007642
6503 060614 013737 010150 010164
6504 060622 032737 000002 010224
6505 060630 001005
6506 060632 012737 001000 010150
6507 060640 005037 010164
6508 060644 004737 030402
6509 060650 022737 000010 007642
6510 060656 001422
6511
6512
6513
6514 060660 012737 005406 010170
6515 060666 013737 007642 010046
6516 060674 004737 030356
6517 060700 013737 010170 010156
6518
6519
6520
6521 060706 004737 030474
6522
6523
6524
6525 060712 012737 000010 010146
6526 060720 004737 027520
6527 060724 013702 010216
6528 060730 006302
6529 060732 000172 010232

```

```

2$:
MOV #PTRTAB, TXPTR ; INITIALIZE POINTERS, COUNTERS AND FLAGS
MOV #PTR13, CMPPTR ; INIT TRANSMIT MESSAGE POINTER
MOV #PTR23, RXPTR ; INIT COMPARE MESSAGE POINTER
MOV CMPTOT, RXMTOT ; INIT RECEIVE MESSAGE POINTER
BIT #DATCKB, PARAM ; MAKE COMPARE AND RX MESSAGE COUNTS EQUAL
BNE GTREX ; IS IT DATA CHECK
GTREX:
MOV #1, RXMTOT ; BRANCH IF CHECKING
CLR CLNSET ; IF NOCHK MAKE RXCOUNT =1
CLR CTX ; CLEAR CLEANUP IN PROGRESS SPECIFIER
CLR CRX ; CLEAR TX COMPLETE COUNT
CLR ILLRSP ; CLEAR RX COMPLETE COUNT
CLR STARTC ; CLEAR ILLEGAL RESPONSE COUNT
CLR ERRTHR ; CLEAR DDCMP START COUNT
CLR RXMIT ; CLEAR ERROR THRESHHOLD RESPONSE COUNT
MOV #RXSTAK, RSPTRS ; CLEAR RETRANSMIT COUNT
MOV #RXSTAK, RSPTRE ; POINT TO START OF RX SAVE AREA
MOV #TXSTAK, TSPTRE ; AGAIN. POINT TO START OF RX SAVE AREA
CLR FLAG ; POINT TO START OF TX SAVE AREA
CLR OPVAR ; CLEAR FLAG
CLR OPVAR1 ; CLEAR NO BUFFER COUNTER
CLR PSCNT ; CLEAR OPVAR1
CLR ERRCNT ; CLEAR PASS COUNT
CLR LNCNT ; CLEAR ERROR COUNT
CLR ; CLEAR COUNTER THAT IS USED FOR STATUS
; INITIALIZE COMMUNICATIONS DEVICE
JSR PC, LOGDVI ; LOG ABOUT TO INIT DEVICE
JSR PC, DVINIT ; INIT DEVICE

GTRX2:
MOV #-1, INDEX ; MAKE INDEX =-1
GTRX2C:
MOV CURCC, TEMP
BIT #DATCKB, PARAM ; IS THERE DATA CHECKING
BNE GTRX2A ; BRANCH IF CHECKING
MOV #BUFLIM, CURCC ; SET UP CHAR COUNT TO 'BUFLIM'
CLR TEMP
GTRX2A:
JSR PC, GTVIND ; GET VALID INDEX
CMP #8, INDEX ; IS IT 8
BEQ GTRX2B ; YES.. ALL DONE GO EXECUTE MODE

; GET RXBUF PTR FIGURE
MOV #RXBUF, TEMP2 ; TEMP = 0 FOR PTP OR MTP/W NO CHK
MOV INDEX, MPLY ; INDEX X TEMP+ RXBUF ADDR =
JSR PC, MTPLY ; NEW RXBUF ADDR.
MOV TEMP2, CURADD ; SET UP RX BUFFER ADDRESS

; GET CURRENT POINTER FIGURE
JSR PC, GRPTCP

; GO LOAD '33' TO BUFFER
MOV #10, MSGTYP ; SET UP FOR 33 TO FILL RX BUFFERS
JSR PC, BLDBUF ; CLEAR RX BUFFER
GTRX2B:
MOV MODTYP, R2
ASL R2
JMP @MODE(R2) ; MODE DISPATCH

```

M 15

SEQ 194

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 195
SETUP FOR RECEIVER ALLOCATIONS

6530

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 196
RECEIVE MODE SECTION

6531
6532
6533
6534
6535
6536
6537
6538
6539
6540
6541
6542
6543
6544
6545
6546
6547
6548
6549
6550
6551
6552

```

.SBTTL          RECEIVE MODE SECTION
:++
:FUNCTIONAL DESCRIPTION:
:RECEIVE-ONLY (OR ONE-WAY-IN) ROUTINE
:IN THIS MODE OF TESTING THE DEVICE'S RECEIVER IS ENABLED IN EXPECTATION
:OF RECEIVING A MESSAGE. AFTER RECEIVING AN "EXPECTED" NUMBER OF
:MESSAGES, THE DATA RECEIVED CAN BE COMPARED AGAINST A LIST OF "EXPECT
:TO RECEIVE" MESSAGES IF DATA-CHECKING IS ENABLED.
:
:SUBORDINATE ROUTINES USED:
:      "ALLTR"
:
:CALLING SEQUENCE:
:      JMP      @MODE(R2)      ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
:--
:RXONLY: BIS      #QRX+ERX,FLAG ;SET UP RX QUE: RX QUEUED, RX EXPECTED
:          JSR      PC,LCPRLS   ;LOAD CPTRLS (RX PTRS)
:          JSR      PC,RXQUAL   ;GO QUE ALL VALID RX'S
:RXON3:  CLR      CPTR
:          JMP      ALLTR      ;GO RX.

```

```

060736 052737 000104 010230
060744 004737 030160
060750 004737 030112
060754 005037 010154
060760 000137 061110

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 197
TRANSMIT MODE SECTION

6553
6554
6555
6556
6557
6558
6559
6560
6561
6562
6563
6564
6565
6566
6567
6568
6569
6570
6571
6572
6573
6574
6575

.SBTTL TRANSMIT MODE SECTION

..**
: FUNCTIONAL DESCRIPTION:
: TRANSMIT-ONLY (OR ONE-WAY-OUT) ROUTINE
: IN THIS MODE OF TESTING A LIST OF MESSAGES IS TRANSMITTED WITHOUT
: EXPECTING ANY DATA TO BE RECEIVED. A REPETITION COUNT CAN BE
: SPECIFIED TO REPETITIVELY TRANSMIT THE LIST.

: SUBORDINATE ROUTINES USED:
: "ALLTR"

: CALLING SEQUENCE:
: JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2

:--
TXONLY: BIC #DATCKB,PARAM ;SET NOCHECK
TXON2: JSR PC,LCPTLS ;LOAD TX POINTERS AND TX COUNTS
BIS #QTX+ETX,FLAG ;TX QUEUED, TX COMPLETE EXPECTED
JSR PC,CLRPLS ;CLEAR RXPRT LIST
MOV #8,INDEX
JMP ALLTR ;GO TX.

060764 042737 000002 010224
060772 004737 030306
060776 052737 000210 010230
061004 004737 030262
061010 012737 000010 007642
061016 000137 061110

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 198
PASSIVE MODE SECTION

6576
6577
6578
6579
6580
6581
6582
6583
6584
6585
6586
6587
6588
6589
6590
6591
6592
6593
6594
6595
6596
6597
6598
6599

.SBTTL PASSIVE MODE SECTION

++
FUNCTIONAL DESCRIPTION:
PASSIVE MODE SECTION
IN THIS MODE OF TESTING, THE DEVICE'S RECEIVER IS ENABLED IN
EXPECTATION OF RECEIVING A MESSAGE. THEN EVERY TIME A MESSAGE IS
RECEIVED, A MESSAGE IS TRANSMITTED. DATA CHECKING CAN BE DONE ON THE
RECEIVED DATA.

SUBORDINATE ROUTINES USED:

"ALLTR"

CALLING SEQUENCE:

JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2

--

061022 004737 030306
061026 004737 030160
061032 052737 000104 010230
061040 004737 030112
061044 000137 061110

PLCK: JSR PC,LCPTLS ;LOAD TX POINTERS AND TX COUNTS
JSR PC,LCPRLS ;SET UP CPTRR TO REC POINTERS
BIS #QRX+ERX,FLAG ;RX QUEUED AND EXPECT RX
JSR PC,RXQUAL ;QUE ALL
JMP ALLTR ;AND GO RX FIRST MSG.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 199
ACTIVE MODE SECTION

6600
6601
6602
6603
6604
6605
6606
6607
6608
6609
6610
6611
6612
6613
6614
6615
6616
6617
6618
6619
6620
6621
6622
6623
6624
6625
6626
6627
6628

.SBTTL ACTIVE MODE SECTION

..++
FUNCTIONAL DESCRIPTION:
ACTIVE MODE SECTION
IN THIS MODE OF TESTING, A LIST OF MESSAGES IS TRANSMITTED AND THOSE
MESSAGES ARE EXPECTED TO BE RECEIVED. RECEIVED DATA CAN BE COMPARED
AGAINST "EXPECTED" DATA IF DATA-CHECKING IS ENABLED.
NOTE: IF BOTH ENDS OF THE LINK ARE IN ACTIVE MODE, THEN THE
LINK MUST BE A FULL DUPLEX LINK!

..SUBORDINATE ROUTINES USED:

"ALLTR"

..CALLING SEQUENCE:

JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2

..--

ALCK:

BIT #DATCKB,PARAM ;IS IT DATA CHECK
BNE 1\$;BRANCH IF CHECK
MOV TXMTOT,RXMTOT ;IF NOCH MAKE RX=TX
1\$: JSR PC,LCPTLS ;LOAD TX POINTERS AND COUNTS
JSR PC,LCPRLS ;LOAD RX POINTERS
BIS #QRX+QTX+ETX+ERX,FLAG ;RX QUEUED, TRANSMIT QUEUED
;EXPECT RECEIVE, EXPECT TRANSMIT
JSR PC,RXQUAL ;QUE UP 1 RX BUFFER FOR ALL VALID LINES

061050
061050 032737 000002 010224
061056 001003
061060 013737 010074 010112
061066 004737 030306
061072 004737 030160
061076 052737 000314 010230
061104 004737 030112

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 200
TRANSMIT - RECEIVE FOR ALL STANDARD MODES

.SBTTL TRANSMIT - RECEIVE FOR ALL STANDARD MODES

6629
6630
6631
6632
6633
6634
6635
6636
6637
6638
6639
6640
6641
6642
6643
6644
6645
6646
6647
6648
6649
6650
6651
6652
6653
6654
6655
6656
6657
6658
6659
6660
6661
6662
6663
6664
6665
6666
6667
6668
6669
6670
6671
6672
6673
6674
6675
6676
6677
6678
6679
6680
6681
6682
6683
6684

..**
FUNCTIONAL DESCRIPTION:
THIS CODE PERFORMS THE FOLLOWING FUNCTIONS:

- 1.) IF RX BUFFERS ARE TO BE QUEUED, TELL DEVICE CODE TO QUE THEM AND LOG 'RECEIVE QUED'.
- 2.) IF TX BUFFERS ARE TO BE QUED, TELL DEVICE CODE TO QUE THEM AND LOG 'TRANSMIT QUED'.
- 3.) WAIT FOR EITHER RECIVE BUFFER OR TRANSMIT BUFFER OR BOTH TO COMPLETE THE SETUP PROCESS.
- 4.) IF RECEIVE COMPLETE, LOG IT. THEN, IF DATA CHECKING IS ENABLED, UPDATE THE RX TABLE.
- 5.) IF TRANSMIT COMPLETE, LOG IT.
- 6.) WHEN BOTH TRANSMIT AND RECEIVE LISTS ARE DONE, GO TO THE COMPARE BUFFER CODE.

SUBORDINATE ROUTINES USED:

```

"DVRXQ" -QUE RECEIVE BUFFER SPACE TO DEVICE
"LOGRXQ"-LOG RECEIVE BUFFER SPACE TO EVENT LOG
"LOGTXQ"-LOG TRANSMIT BUFFER QUED TO EVENT LOG
"DVTXRX"-QUE TRANSMIT BUFFER AND WAIT FOR RX
OR TX TO COMPLETE
"LOGRXC"-LOG RECEIVE BUFFER COMPLETED TO EVENT LOG
"LOGTXC"-LOG TRANSMIT BUFFER COMPLETED TO EVENT LOG

```

USE OF FLAG BITS:

```

QRX - SET ON ENTRY TO ALLTR IF RECEIVE IS TO BE QUED TO
      DEVICE. CLEARED BY DVRXQ AND THEN SET BY DVTXRX
      WHEN RX IS COMPLETED.
QTX - SET ON INPUT TO ALLTR IF TRANSMIT IS TO BE QUED TO
      DEVICE. CLEARED ON ENTRY TO DVTXRX AND SET BY DVTXRX
      WHEN TX BUFFER IS COMPLETED.
ETX - USED BY DVTXRX TO DETERMINE IF TX BUFFER COMPLETED IS
      EXPECTED.
ERX - USED BY DVTXRX TO DETERMINE IF RX BUFFER COMPLETED IS
      EXPECTED.

```

CALLING SEQUENCE:

JMP ALLTR

;GO TO TRANSMIT-RECEIVE FOR ALL STANDARD MODES

--

```

061110 ALLTR:
061110 032737 000004 010230 ALCK5: BIT #QRX,FLAG ;IS A RECEIVE QUEUED?
061116 001406 BEQ ALCK1 ;IF NO RX QUEUE PENDING, SEE IF TX PENDING
061120 004737 030454 ALCK5B: JSR PC,ULRPLS ;GET RX INDEX
061124 004737 031142 JSR PC,LOGAQR ;LOG AND QUE REC.
061130 004737 030434 JSR PC,LDRPLS ;RESTORE RX PTR TO LIST
061134 032737 000010 010230 ALCK1: BIT #QTX,FLAG ;TRANSMIT QUEUED
061142 001422 BEQ ALCK2 ;IF NO TX'S GO TO 2
061144 004737 031030 JSR PC,GNTXPR ;TRANSMIT COMPLETE FLAG SET: GET NEXT XMIT PTR.

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 201
TRANSMIT - RECEIVE FOR ALL STANDARD MODES

6685	061150	013702	010154		MOV	CPTR,R2	: POINTER TO NEXT ADDRESS, BYTE COUNT PAIR
6686	061154	011237	010170		MOV	(R2),TEMP2	: ADDRESS OF THE MESSAGE
6687	061160	012237	010064		MOV	(R2)+,DVTXA	: SAME ADDRESS
6688	061164	011237	010172		MOV	(R2),TEMP3	: BYTE COUNT OF THE MESSAGE
6689	061170	012237	010066		MOV	(R2)+,DVTCC	: SAME BYTE COUNT
6690	061174	010237	010154		MOV	R2,CPTR	: NOW, POINTING TO FOLLOWING MESG ADDR, BYTE CT
6691	061200	004737	030572		JSR	PC,LDTPLS	: RELOAD LIST
6692	061204	004737	024144		JSR	PC,LOGTXQ	: LOG TRANSMIT QUEUED
6693							
6694	061210	004737	064340		ALCK2: JSR	PC,DVTXRX	: GO TO TX AND RX SUBROUTINE
6695							
6696	061214	032737	000004	010230	BIT	#QRX,FLAG	: CHECK FOR REC. QUEUED
6697	061222	001542			BEQ	ALCK3	: IF MESSAGE NOT RECEIVED, CHECK FOR TX
6698	061224	013737	010104	010170	MOV	DVRXA,TEMP2	: MESSAGE RECEIVED, LOG IT
6699	061232	013737	010106	010172	MOV	DVRCC,TEMP3	: LOG EVENT
6700	061240	013737	010102	007636	MOV	DVRTB,TRIBN	: LOG MESSAGE RECEIVED
6701	061246	004737	024216		JSR	PC,LOGRXC	: LOG REC COMPLETE
6702							
6703	061252	032737	000004	010224	UPTABL: BIT	#ECHOB,PARAM	: IS THIS ECHO MODE(PASSIVE)
6704	061260	001410			BEQ	UPTA4	: IF NOT GO TO 4
6705	061262	004737	030612		JSR	PC,ULTPLS	
6706	061266	013702	010154		MOV	CPTR,R2	: ELSE SET R2 TO PRESENT TX TABLE
6707	061272	013722	010170		MOV	TEMP2,(R2)+	: STORE OFF RX ADD
6708	061276	013712	010172		MOV	TEMP3,(R2)	: AND CC
6709	061302	032737	000002	010224	UPTA4: BIT	#DATCKB,PARAM	: IS DATA CHECKING ASKED FOR
6710	061310	001012			BNE	UPTA1	: IF SO GO TO UPTA1
6711	061312	004737	031102		JSR	PC,GETIND	: GET INDEX
6712	061316	004737	030210		JSR	PC,LCPRL1	: RESTORE POINTER
6713	061322	013737	010170	010152	MOV	TEMP2,CPTRR	: RESTORE POINTER
6714	061330	004737	030434		JSR	PC,LDRPLS	: LOAD COUNT AND LIST
6715	061334	000440			BR	UPTEX	
6716							
6717	061336	004737	030454		UPTA1: JSR	PC,ULRPLS	: GET PTR FROM LIST
6718	061342	013702	010152		MOV	CPTRR,R2	
6719	061346	013737	010172	010160	MOV	TEMP3,TOTCC	: COPY RCVD MESSAGE COUNT
6720	061354	013737	010170	010156	MOV	TEMP2,CURADD	: COPY RCVD MESSAGE ADDRESS
6721	061362	063737	010160	010156	ADD	TOTCC,CURADD	: POINT TO END OF RCVD MESSAGE
6722	061370	004737	027644		JSR	PC,FCURAD	: MAKE SURE ADDRESS IS ON WORD BOUNDARY
6723	061374	011237	010164		MOV	(R2),TEMP	: LOAD TEMP WITH PREV. RX BUFF LENGTH
6724	061400	163737	010160	010164	SUB	TOTCC,TEMP	: SUBTRACT CURRENT COUNT
6725	061406	013722	010172		MOV	TEMP3,(R2)+	: COPY ACTUAL COUNT RECEIVED
6726	061412	013722	010156		MOV	CURADD,(R2)+	: STORE OF NEW RXBUFF ADDRESS
6727	061416	013712	010164		MOV	TEMP,(R2)	: AND NEW BUFF LENGTH
6728	061422	162702	000002		SUB	#2,R2	: PUT POINTER BACK TO ADDR.
6729	061426	010237	010152		MOV	R2,CPTRR	: AND RESTORE IT.
6730	061432	004737	030434		JSR	PC,LDRPLS	
6731	061436				UPTEX:		
6732	061436	022737	000002	010216	CMP	#PAS,MODTYP	
6733	061444	001011			BNE	ALCK2A	: IF NOT PASSIVE LOOP THEN GO TO 2A
6734	061446	005337	007642		DEC	INDEX	: IF PASSIVE NEXT TXQ WILL BE FOR THIS LINE
6735	061452	042737	000004	010230	BIC	#QRX,FLAG	: CLEAR BOTH EXPECTED AND COMPLETED FLAGS
6736	061460	052737	000210	010230	BIS	#QTX+ETX,FLAG	: TX QUEUED, EXPECT TRANSMIT COMPLETE
6737	061466	000622			BR	ALCK1	
6738							
6739	061470	004737	030552		ALCK2A: JSR	PC,ULRCLS	: GET COUNT
6740	061474	005337	010110		DEC	DVRCT	: DEC REC COUNT

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 202
TRANSMIT - RECEIVE FOR ALL STANDARD MODES

6741	061500	004737	030532		JSR	PC,LDRCLS	:RESTORE COUNT
6742	061504	005737	010110		TST	DVRCT	:IS IT ALL DONE
6743	061510	001007			BNE	ALCK3	:NO. GO CHECK TX
6744	061512	042737	000004	010230	BIC	#QRX,FLAG	:CLEAR THE RX QUEUED FLAG
6745	061520	005037	010152		CLR	CPTRR	:YES. CLEAR POINTER
6746	061524	004737	030434		JSR	PC,LDRPLS	:AND RELOAD LIST
6747							
6748							
6749	061530	032737	000010	010230	:CHECK FOR MESSAGE TRANSMITTED		
6750	061536	001467			ALCK3:	BIT #QTX,FLAG	:IS A TX QUEUED?
6751	061540	013737	010064	010170	BEQ	ALCK4	:IF NOT TX THEN GO BACK (NEITHER RX NOR TX)
6752	061546	013737	010066	010172	MOV	DVTXA,TEMP2	:TRANSMIT COMPLETED- LOG IT
6753	061554	013737	010070	007636	MOV	DVTCC,TEMP3	:LOG TX COMPLETED
6754	061562	004737	024162		MOV	DVTTB,TRIBN	:LOG EVENT-
6755					JSR	PC,LOGTXC	:TRANSMIT COMPLETE
6756	061566	004737	030652		JSR	PC,ULTCLS	:GET CURRENT TRANSMIT COUNT TO DVTCT
6757	061572	005337	010072		DEC	DVTCT	:DEC TX COUNT TO RECORD 1 MESSAGE TX COMPLETED
6758	061576	004737	030632		JSR	PC,LDTCLS	:AND RELOAD LIST
6759							
6760	061602	022737	000002	010216	CMP	#PAS,MODTYP	:ARE WE IN PASSIVE MODE?
6761	061610	001020			BNE	ALCK3A	:IF NOT PASSIVE MODE GO TO 3A
6762	061612	042737	000010	010230	BIC	#QTX,FLAG	:TRANSMIT COMPLETE, NO LONGER QUEUED.
6763	061620	005737	010072		TST	DVTCT	
6764	061624	001403			BEQ	ALCK3D	:IF NO MORE MESSG TO RX FOR THIS LINE
6765							:EXIT WITHOUT RESETTING QRX (RX STILL QUEUED)
6766	061626	052737	000104	010230	BIS	#QRX+ERX,FLAG	:AND SET THE RX FLAGS
6767	061634	004737	030744		ALCK3D:	JSR PC,GATCFL	
6768	061640	005737	010072		TST	DVTCT	
6769	061644	001007			BNE	ALCK3C	:IF MORE TX'S TO IT
6770	061646	000137	061742		JMP	CMPSR	: ELSE COMPARE
6771							
6772							
6773	061652	004737	030744		:NOT PASSIVE MODE		
6774	061656	005737	010072		ALCK3A:	JSR PC,GATCFL	:GET ALL TX COUNTS FROM LIST
6775	061662	001404			TST	DVTCT	:IS IT ALL DONE? ALL TRANSMISSIONS COMPLETE?
6776	061664	004737	031102		BEQ	ALCK3B	:IF NOT GO BACK TO 5
6777	061670	000137	061110		ALCK3C:	JSR PC,GETIND	:ALL TRANSMISSIONS COMPLETE FOR CURRENT LINE
6778	061674	005037	010154		JMP	ALCK5	:NOW GET ANOTHER VALID LINE.
6779	061700	042737	000010	010230	ALCK3B:	CLR CPTR	:CLEAR POINTER
6780	061706	032737	000002	010224	BIC	#QTX,FLAG	:TRANSMIT COMPLETE, NO LONGER QUEUED.
6781	061714	001405			BIT	#DATCKB,PARAM	:IS IT DAT CHECK
6782	061716	004737	030672		BEQ	ALCK4A	:IF NOT THEN END WORKING RX.
6783	061722	005737	010152		ALCK4:	JSR PC,GARPFL	
6784	061726	001356			TST	CPTRR	
6785	061730	005737	010154		BNE	ALCK3C	:IF SOME RX'S LEFT GO BACK
6786	061734	001402			ALCK4A:	TST CPTR	
6787	061736	000137	061210		BEQ	CMPSR	:BRANCH IF ANY TX'S LEFT
6788					JMP	ALCK2	
6789							
6790							
6791							

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 203
DATA COMPARISON CODE

.SBTTL DATA COMPARISON CODE

..**
FUNCTIONAL DESCRIPTION:

CMPSR - COMPARE CODE
THIS CODE COMPARES THE RECEIVED DATA AGAINST THE
EXPECTED AND FILLS THE EVENT LOG WITH 1 OF 3 MSGS.

NOTE: IF NO DATA CHECKING SKIP THIS CODE

- 1.) A DATA COMPARISON ENTRY WHICH REPORTS THE NUMBER OF COMPARISON ERRORS FOUND.
 - 2.) A DATA COMPARISON ENTRY WHICH REPORTS DIFFERENCES IN REC LENGTH TO COMPARE LENGTH.
 - 3.) A DATA COMPARISON STARTED ENTRY WHICH REPORTS ADDRESS OF RECEIVE BUFFER AND BYTE COUNT.
- THIS CODE ALSO REPORTS SOFT ERRORS FOR DATA COMPARISON (THE FIRST 5 ONLY), LENGTH ERROR, AND TOTAL NUMBER OF ERRORS

SUBORDINATE ROUTINES USED:

- 'LOGCMP' - SEE ITEM 3 ABOVE
- 'LOGCML' - SEE ITEM 2 ABOVE
- 'LOGCMD' - SEE ITEM 1 ABOVE

CALLING SEQUENCE:

JMP CMPSR ; JUMP TO DATA COMPARISON CODE

--

6824	061742	032737	000002	010224	CMPSR:	BIT	#DATCKB,PARAM	; IS DATA CHECKING TO BE DONE
6825	061750	001533				BEQ	CMPSX	; IF NOT THEN EXIT
6826	061752	012737	177777	007642		MOV	#-1,INDEX	
6827	061760	004737	030402		CMPSR:	JSR	PC,GTVIND	
6828	061764	022737	000010	007642		CMP	#8.,INDEX	
6829	061772	001522				BEQ	CMPSX	; END IF NO MORE LINES
6830								
6831	061774	004737	030474			JSR	PC,GRPTCP	
6832	062000	013737	010054	010152		MOV	CMPPTR,CPTRR	; AND START OF COMPARE POINTS TO CPTRR
6833	062006	013737	010112	010110		MOV	RXMTOT,DVRCT	
6834								
6835	062014				CMPS3:			
6836	062014	013702	010154			MOV	CPTR,R2	; MOVE CURRET RX PT. TO R2
6837	062020	011237	010170			MOV	(R2),TEMP2	; MOVE RX ADD TO EVENT LOG
6838	062024	012201				MOV	(R2)+,R1	; SET R1 TO START ADD OF RX
6839	062026	012237	010172			MOV	(R2)+,TEMP3	; SET CHAR COUNT TO EVENT LOG
6840	062032	010237	010154			MOV	R2,CPTR	; RESTORE RX POINT
6841								
6842	062036	013702	010152			MOV	CPTRR,R2	; PUT R2 AT COMPARE TABLE
6843	062042	012203				MOV	(R2)+,R3	; SET R3 TO COMPARE ADD
6844	062044	012204				MOV	(R2)+,R4	; SET R4 TO COMP CC
6845	062046	010237	010152			MOV	R2,CPTRR	; RESTORE POINTER
6846	062052	010437	010174			MOV	R4,TEMP4	
6847	062056	004737	024312			JSR	PC,LOGCMP	; LOG COMPARE START.

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 204
DATA COMPARISON CODE

```

6848
6849 062062 020437 010172      CMP      R4,TEMP3      ;IS COMPARE COUNT = TO RX COUNT
6850 062066 001410              BEQ      CMPS7          ;IF SO GO TO 7
6851 062070 005237 010124      INC      ERRCNT
6852          000001      ENUM = ^D<ENUM+1>    ;INCREMENT ERROR NUMBER
6853 062074          ERRSOFT ENUM,EDDL,ERR10 ;PRINT ERROR
6854 062074 104457              TRAP    C$ERSOFT
6855 062076 000001              .WORD  1
6856 062100 017661              .WORD  EDDL
6857 062102 023430              .WORD  ERR10
6858 062104 004737 024330      JSR      PC,LOGCML    ;LOG LENGTH ERROR
6859
6860 062110 005037 010174      CMPS7:  CLR      TEMP4      ;CLEAR BAD BYTE COUNTER
6861 062114 012737 000001 010162      MOV      #1,OFSET    ;SET OFFSET BYTE COUNT TO 1
6862 062122 122123      CMPS1:  CMPB     (R1)+,(R3)+ ;COMPARE RX WITH EXPETED
6863 062124 001422      BEQ      CMPS6          ;IF EQUAL THEN GO TO 6
6864
6865 062126 005237 010174      CMPS2:  INC      TEMP4      ;INC BAD COUNT
6866 062132 023727 010174 000005      CMP      TEMP4,#5    ;IS IT MORE THEN 5
6867 062140 101014      BHI      CMPS6          ;IF SO GO FOR MORE
6868 062142 114337 010204      MOVB    -(R3),GOOD   ;STORE GOOD BYTE FOR ERROR
6869 062146 114137 010205      MOVB    -(R1),BAD    ;STORE BAD BYTE FOR ERROR
6870 062152 005237 010124      INC      ERRCNT
6871          000002      ENUM = ^D<ENUM+1>    ;INCREMENT ERROR NUMBER
6872 062156          ERRSOFT ENUM,EDDE,ERR1 ;REPORT CGMPARISON FAILURE TO OPR.
6873 062156 104457              TRAP    C$ERSOFT
6874 062160 000002              .WORD  2
6875 062162 017716              .WORD  EDDE
6876 062164 023340              .WORD  ERR1
6877 062166 005201
6878 062170 005203
6879 062172 005237 010162      CMPS6:  INC      R1
6880 062176 005304      INC      R3
6881 062200 001350      INC      OFFSET      ;INC OFFSET
6882 062202 005737 010174      DEC      R4          ;ELSE DEC CHAR COUNT AND SEE IF 0
6883 062206 001410      BNE     CMPS1          ;IF NOT GO BACK
6884 062210 005237 010124      TST     TEMP4        ;SEE IF ANY CMP ERRS FOR THIS MSG
6885          000003      BEQ     CMPS5A        ;BR IF NONE
6886 062214          INC      ERRCNT
6887 062214 104457      ENUM = ^D<ENUM+1>    ;INCREMENT ERROR NUMBER
6888 062216 000003      ERRSOFT ENUM,EDDE,ERR2 ;REPORT # OF MISMATCHES FOR MESSAGE
6889 062220 017716              TRAP    C$ERSOFT
6890 062222 023402              .WORD  3
6891 062224 004737 024346      CMPS5:  JSR      PC,LOGCMD ;LOG DATA ERROR IN COMPARE
6892 062230          .WORD  EDDE
6893 062230 005337 010110      CMPS5A: DEC      DVRCT
6894 062234 001267      BNE     CMPS3
6895 062236 000400      BR      CMPSX        ;IF NOT ALL DONE GO BACK
6896          ;BRANCH TO END OF PASS CODE

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 205
INTERNAL END OF PASS CODE

.SBTTL INTERNAL END OF PASS CODE

6897
6898
6899
6900
6901
6902
6903
6904
6905
6906
6907
6908
6909
6910
6911
6912
6913
6914
6915
6916
6917
6918
6919
6920
6921
6922
6923
6924
6925
6926
6927
6928
6929

..++
FUNCTIONAL DESCRIPTION:
THIS CODE INCREMENTS THE PASS COUNT FOR THE
EVENT LOG. LOGS THE END OF PASS EVENT
IF "RPASS" IS A MINUS ONE RETURN TO MODE
DISPATCHER. IF NOT -1 THEN DECREMENT RPASS
AND IF "RPASS" IS THEN = TO 0 GO TO DCLT PROMPT
IN NOT = TO 0 THEN GO BACK TO MODE DISPATCHER

SUBORDINATE ROUTINES USED:

'LOGEOP' - LOG END OF PASS TO EVENT LOG

062240	005237	010122	CMPSEX: INC	PSCNT	:BUMP PASS COUNT
062244	013737	010116 010174	MOV	OPVAR,TEMP4	:LOG NO BUFFS
062252	013737	010122 010170	MOV	PSCNT,TEMP2	:LOG PASS COUNT
062260	013737	010124 010172	MOV	ERRCNT,TEMP3	
062266	004737	024372	JSR	PC,LOGEOP	:LOG END OF PASS
062272	022737	177777 010226	CMP	#-1,RPASS	:SEE IF RPASS=-1
062300	001403		BEQ	1\$:IF IT IS DON'T DECREMENT, LOOP FOREVER
062302	005337	010226	DEC	RPASS	:DEC PASS COUNT
062306	001402		BEQ	2\$:IF DONE GOTO DCLT> PROMPT
062310	000137	060606	1\$: JMP	GTRX2	:ELSE GO BACK AND DISPATCH
062314	004737	064020	2\$: JSR	PC,HLITRB	:GO HALT ALL LINES BEFORE GOING BACK
062320	000137	054512	JMP	GTRAS	:WHEN RPASS=0 GO BACK TO "DCLT>"

02

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 206
DOWN-LINE-LOAD SECTION

6930
6931
6932
6933
6934
6935
6936
6937
6938
6939
6940
6941
6942
6943
6944
6945
6946
6947
6948
6949
6950
6951
6952
6953
6954
6955
6956
6957
6958
6959
6960
6961
6962
6963
6964
6965
6966
6967
6968

.SBTTL DOWN-LINE-LOAD SECTION

++
FUNCTIONAL DESCRIPTION:
DOWN-LINE-LOAD SECTION
IN THIS MODE OF TESTING THE "HOST" OR ORIGINATING STATION
REQUESTS THE "SATELLITE" OR BOOT STATION TO ENTER MOP MODE.
THE BOOT STATION THEN SENDS A "REQUEST PROGRAM MESSAGE".
THE "HOST" THEN SENDS A "MEMORY LOAD WITH TRANSFER ADDRESS"
THAT CONTAINS IMAGE DATA TO BE LOADED BY THE BOOT STATION'S
DMP-11 FIRMWARE STARTING AT LOC. 0. THIS IMAGE DATA WILL CONTAIN A
PROGRAM THAT WILL PRINT A MSG THAT DOWN-LINE-LOAD WAS SUCCESSFUL.

SUBORDINATE ROUTINES USED:
"DLTXRX" - SPECIAL TX RX ROUTINE FOR DLL
"DVRXQ" - QUE RX BUFFER SPACE TO DEVICE
"LOGRXQ" - LOG RX SPACE QUED TO EVENT LOG
"LOGTXQ" - LOG TX BUFFER QUED TO EVENT LOG
"DVTXRX" - QUE TX BUFFER AND WAIT FOR RX OR TX TO COMPLETE
"LOGTXC" - LOG TX COMPLETED TO EVENT LOG
"LOGRXC" - LOG RX COMPLETED TO EVENT LOG

CALLING SEQUENCE:
JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
--

DLL:
PRINTF #DLLQ1 ;PRINT DOWN LINE LOAD NOT SUPPORTED
MOV #DLLQ1,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C\$PNTF
ADD #4,SP
JMP GTRA5 ;EXIT

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 207
TALK MODE SECTION

6969
6970
6971
6972
6973
6974
6975
6976
6977
6978
6979
6980
6981
6982
6983
6984
6985
6986
6987
6988
6989
6990
6991
6992
6993
6994
6995
6996
6997
6998
6999
7000
7001
7002
7003
7004
7005
7006
7007
7008
7009
7010
7011
7012
7013
7014
7015
7016
7017
7018
7019
7020
7021
7022
7023
7024

062350 012737 177777 007642
062356 004737 030402
062362 042737 000002 010224
062370 012702 002520
062374 012722 177777
062400 022702 002642
062404 001373
062406 104443
062410 000406
062412 002520
062414 000142
062416 016546
062420 000377
062422 000001
062424 000110
062426 005002
062430 122762 000377 002520
062436 001402
062440 005202
062442 000772
062444 010237 002162
062450 012737 002520 010064
062456 012737 002520 010170
062464 013737 002162 010172
062472 013737 002162 010066
062500 004737 024144
062504 052737 000210 010230
062512 005037 010152
062516 004737 064340
062522 013737 010064 010170
062530 013737 010066 010172
062536 004737 024162

.SBTTL TALK MODE SECTION

++
FUNCTIONAL DESCRIPTION:
TALK MODE SECTION
IN THIS MODE, THE "TALK" END OF THE LINK TRANSMITS OPERATOR
SPECIFIED MESSAGES UNTIL A "EXIT" MESSAGE IS TYPE. AT THAT POINT,
THIS END OF THE LINK GOES INTO "LISTEN" MODE.

SUBORDINATE ROUTINES USED:
"LOGTXQ" - LOG TX BUFFER QUED TO EVENT LOG
"DVTXRX" - QUE TX BUFFER TO DEVICE AND WAIT FOR COMPLETE
"LOGTXC" - LOG TX COMPLETE TO EVENT LOG

CALLING SEQUENCE:
JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2

TALCK: MOV #-1,INDEX
JSR PC,GTVIND ;GET FIRST LINE
BIC #DATCKB,PARAM ;SET NOCHECK
1\$: MOV #OPBUF,R2
MOV #-1,(R2)+ ;CLEAR OUT OPBUFFER FIRST
CMP #OPEND,R2
BNE 1\$
GMANID OPRMM,OPBUF,A,377,1,72.,NO ;GET TALK MESSAGE

TRAP CS\$GMAN
BR 10001\$
.WORD OPBUF
.WORD T\$CODE
.WORD OPRMM
.WORD 377
.WORD T\$LOLIM
.WORD T\$HILIM

10001\$:
2\$: CLR R2 ;NOW GET CHAR COUNT
CMPB #377,OPBUF(R2)
BEQ 3\$
INC R2
BR 2\$
3\$: MOV R2,OPCNT
MOV #OPBUF,DVTXA ;SET UP TX ADDR.
MOV #OPBUF,TEMP2
MOV OPCNT,TEMP3
MOV OPCNT,DVTCC ;SET UP TX CC
JSR PC,LOGTXQ
BIS #QTX+ETX,FLAG ;TRANSMIT QUEUED, EXPECT TRANSMIT COMPLETE.
CLR CPTRR ;CLEAR RX POINTER
JSR PC,DVTXRX
MOV DVTXA,TEMP2
MOV DVTCC,TEMP3
JSR PC,LOGTXC

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 208
TALK MODE SECTION

7025	062542	022737	054105	002520	CMP	#'EX,OPBUF	;CHECK FOR EXIT
7026	062550	001277			BNE	TALCK	
7027	062552	022737	052111	002522	CMP	#'IT,OPBUF+2	
7028	062560	001273			BNE	TALCK	
7029	062562	042737	000210	010230	BIC	#QTX+ETX,FLAG	;TRANSMIT QUEUED, EXPECT TRANSMIT COMPLETE
7030	062570	012737	000006	010216	MOV	#LIS,MODTYP	;CHANGE TO LISTEN MODE
7031	062576	000137	060606		JMP	GTRX2	;AND GO BACK TO DISPATCH

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 209
LISTEN MODE SECTION

.SBTTL LISTEN MODE SECTION

..**
FUNCTIONAL DESCRIPTION:
LISTEN MODE SECTION
IN THIS MODE, THE "LISTEN" END OF THE LINK PRINTS ALL OF THE MESSAGES
RECEIVED BY THE DEVICE ON THE OPERATOR'S CONSOLE. IF THE MESSAGE
RECEIVED IS AN "EXIT" MESSAGE, THEN THE NODE ENTERS "TALK" MODE.

SUBORDINATE ROUTINES USED:

"DVRXQ" - QUE RECEIVE BUFFER SPACE TO DEVICE
"LOGRXQ" - LOG RECEIVE BUFFER QUED TO EVENT LOG
"DVTXRX" - WAIT FOR RX TO COMPLETE
"LOGRXC" - LOG RX COMPLETE TO EVENT LOG

CALLING SEQUENCE:

JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2

```

7032
7033
7034
7035
7036
7037
7038
7039
7040
7041
7042
7043
7044
7045
7046
7047
7048
7049
7050
7051
7052 062602 012737 177777 007642 LISCK: MOV #-1,INDEX
7053 062610 004737 030402 JSR PC,GTVIND ;GET FIRST LINE
7054 062614 042737 000002 010224 BIC #DATCKB,PARAM ;CLEAR CHECK BIT
7055 062622 PRINTF #LISP ;PRINT PROMPT FOR OPR.
7056 062622 012746 016535
7057 062626 012746 000001
7058 062632 010600
7059 062634 104417
7060 062636 062706 000004
7061 062642 012737 002520 010104 LISCKA: MOV #OPBUF,DVRXA ;SET DEVICE UP TO REC AT OPBUF
7062 062650 012737 002520 010170 MOV #OPBUF,TEMP2
7063 062656 012737 000122 010106 MOV #82.,DVRCC ;SET UP CHAR COUNT TO 82.
7064 062664 012737 000122 010172 MOV #82.,TEMP3
7065 062672 052737 000104 010230 BIS #QRX+ERX,FLAG ;SET UP FLAG
7066 062700 005037 010154 CLR CPTR ;CLEAR THE TX.
7067
7068 062704 004737 064174 JSR PC,DVRXQ ;QUE RX
7069 062710 004737 024200 JSR PC,LOGRXQ
7070
7071 062714 004737 064340 JSR PC,DVTXRX ;GO TO DEVICE RX. SUBROUTINE
7072
7073 062720 013737 010104 010170 MOV DVRXA,TEMP2
7074 062726 013737 010106 010172 MOV DVRCC,TEMP3 ;SET UP ADDR.AND CC.
7075 062734 004737 024216 JSR PC,LOGRXC ;LOG COMPLETED
7076 062740 063737 010104 010106 ADD DVRXA,DVRCC
7077 062746 105077 125134 CLRB @DVRCC
7078 062752 PRINTF #OPBFPT
7079 062752 012746 002514
7080 062756 012746 000001
7081 062762 010600
7082 062764 104417
7083 062766 062706 000004
7084 062772 022737 054105 002520 CMP #'EX,OPBUF ;COMPARE FOR EX OF "EXIT"
7085 063000 001320 BNE LISCKA ;IF NOT EXIT THEN GO BACK
7086 063002 022737 052111 002522 CMP #'IT,OPBUF+2 ;IF FIRST HALF OK CHECK NEXT PART
7087 063010 001314 BNE LISCKA ;IF NOT EXIT THE GO BACK

```

MOV #LISP,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C\$PNTF
ADD #4,SP

MOV #OPBFPT,-(SP)
MOV #1,-(SP)
MOV SP,R0
TRAP C\$PNTF
ADD #4,SP

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 210
LISTEN MODE SECTION

7088 063012 012737 000005 010216
7089 063020 000137 060606
7090
7091

MOV #TAL,MODTYP ;CHANGE MODE TO TALK
JMP GTRX2 ;RETURN TO DISPATCHER

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 211
LISTEN MODE SECTION

7092
7093
7094
7095
7096
7097
7098
7099
7100
7101
7102
7103
7104
7105
7106
7107
7108
7109
7110
7111
7112
7113
7114
7115
7116
7117
7118
7119
7120
7121
7122
7123
7124
7125
7126
7127
7128
7129
7130
7131
7132
7133
7134
7135
7136
7137
7138
7139
7140
7141
7142
7143
7144
7145
7146
7147

.SBTTL DVINIT - DEVICE INITIALIZATION

++
FUNCTIONAL DESCRIPTION:
DVINIT - THIS SUBROUTINE STARTS THE KMS11-BD, TRANSFERS LINE, MODEM AND LOOPBACK PARAMETERS TO THE DEVICE, AND THEN REQUESTS THE KMS11-BD TO STARTUP A DDCMP LINK.

INPUTS:
MLTYP - LOOPBACK PARAMETER
MODCSR - MODEM CSR ADDRESS
TRIBN - ESTABLISHED LINE NUMBER
BDLBAS - ADDRESS OF BUFFER DESCRIPTION LIST

OUTPUTS:
NONE

SUBORDINATE ROUTINES USED:
LOGEO REPORT AND LOG TIMEOUT WAITING FOR KMS
TOORIO REQUEST INPUT INTERRUPT, WAIT FOR INTERRUPT OR TIME OUT
TOORO WAIT FOR OUTPUT INTERRUPT OR TIMEOUT
GTVIND GET ESTABLISHED LINE NUMBER COPIED TO TRIBN

CALLING SEQUENCE:
JSR PC,DVINIT

--
DVINIT:

063024

;REJECT DOWN LINE LOAD REQUEST
;SEE IF DOWN-LINE-LOAD (NOT SUPPORTED)
; BR IF NOT, ELSE EXIT TO PRINT MSG
;START KMS
;SET TIMER
;HALT KMS TO ENSURE CLEAN STARTUP
;MASTER CLEAR
;SPECIFY KMS NOT RUNNING
;(KMS CLEARS BSEL2 WHEN STARTED)
;SET RUN BIT TO START KMS
;WAIT FOR KMS TO START
;IS KMS RUNNING ?
;YES. CONTINUE
;NO. ALLOW CTRL/C EXIT WHILE WAITING
TRAP C\$BRK
;HAS TIME EXPIRED ?
;NO. CHECK AGAIN
;REPORT KMC NOT STARTED

CMP #DOW,MODTYP
BNE DVIN00
JMP DVINEX
DVIN00: MOV #1000,TIMER1
CLR @SELO
MOV #MCLR,@SELO
MOVB #377,@BSEL2
MOV #KRUN,@SELO
DVIN0: TSTB @BSEL2
BEQ 10\$
BREAK
TST TIMER1
BNE DVIN0

022737 000004 010216
063024 001002
063032 000137 063702
063040 012737 001000 010270
063046 005077 130320
063052 012777 040000 130312
063060 112777 000377 130310
063066 012777 100000 130276
063074 105777 130276
063100 001416
063102 104422
063104 005737 010270
063110 001371

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 212
DVINIT - DEVICE INITIALIZATION

```

7148 063112 012737 022006 010170      MOV      #DVEMO,TEMP2      ; TIME EXPIRED. PRINT ERROR MESSAGE
7149 063120 004737 067126      JSR      PC,LOGEO         ; LOG TIME OUT WAITING FOR RUN
7150      000004      ENUM = ^D<ENUM+1>        ; INCREMENT ERROR NUMBER
7151 063124      ERRSOFT ENUM,DVEMO,ERR13 ; RECORD ERROR
7152 063124 104457      TRAP     CSERSOFT
7153 063126 000004      .WORD   4
7154 063130 022006      .WORD   DVEMO
7155 063132 023460      .WORD   ERR13
7156 063134 000741      BR      DVIN00           ;RETRY KMS STARTUP
7157 063136      10$:
7158 063136 005037 011514      CLR     VSTAT            ;NEGATE "STATUS DATA VALID"
7159 063142 005237 011512      INC     ENRDST           ;ASSERT READ STATUS ENABLED
7160 063146 042737 000003 010230      BIC     #ININT+OTINT,FLAG ;CLEAR INTERRUPT INDICATORS
7161
7162 063154      DVIN1:
7163      ;SPECIFY SYSTEM INITIALIZATION (COMMA:ID PAIR 0,0)
7164
7165 063154 052737 000400 010230      BIS     #INITC,FLAG      ;SPECIFY WAITING FOR INITIALIZATION
7166 063162 004737 065422      JSR     PC,TOORIO        ;REQUEST PERMISSION
7167      ; TO SEND COMMAND TO KMS
7168      ;PERMISSION GRANTED TO SEND COMMAND
7169 063166 042777 177757 130202      BIC     #177757,@SEL2   ;SPECIFY COMMAND PAIR 0,0
7170 063174 005077 130202      CLR     @SEL4            ;NO PARAMETERS
7171 063200 005077 130202      CLR     @SEL6            ;REQUIRED
7172 063204 142777 000020 130164      BICB   #RDYI,@BSEL2     ;ASSERT READY TO TRANSFER COMMAND
7173      ;WAIT FOR INITIALIZATION COMPLETION
7174 063212 004737 065622      JSR     PC,TOORO        ;WAIT FOR RESPONSE
7175 063216 032737 000400 010230      BIT     #INITC,FLAG     ;IS INITIALIZATION COMPLETED ?
7176 063224 001401      BEQ     DVIN2           ;YES. CONTINUE
7177 063226 000752      BR      DVIN1          ;RETRY INITIALIZATION
7178
7179 063230      DVIN2:
7180      ;SPECIFY LOOP PARAMETER
7181
7182 063230 012737 000377 010164      MOV     #377,TEMP       ;ASSUME INTERNAL LOOPBACK NEGATED
7183 063236 022737 000001 010220      CMP     #1,MLTYP        ;IS INTERNAL LOOPBACK SPECIFIED ?
7184 063244 001002      BNE     10$            ;NO. ASSUMPTION WAS CORRECT
7185 063246 005037 010164      CLR     TEMP            ;SPECIFY INTERNAL LOOPBACK ASSERTED
7186 063252      10$:
7187 063252 004737 065422      JSR     PC,TOORIO        ;GIVE DEVICE INTERNAL LOOPBACK MODE SPECIFIER
7188      ;REQUEST PERMISSION TO SEND COMMAND
7189      ;PERMISSION GRANTED
7190 063256 042777 177757 130112      BIC     #177757,@SEL2   ;CLEAR ALL BUT RDYI
7191 063264 152777 000142 130104      BISB   #142,@BSEL2     ;SPECIFY WRITE DATA TO DEVICE
7192 053272 012777 007726 130102      MOV     #KMLTYP,@SEL4   ;SPECIFY DATA ADDRESS IN DEVICE
7193 063300 113777 010164 130100      MOVB   TEMP,@BSEL6     ;SPECIFY DEVICE LOOPBACK MODE
7194 063306 142777 000020 130062      BICB   #RDYI,@BSEL2     ;SPECIFY READY TO TRANSFER COMMAND
7195 063314 022737 000001 010220      CMP     #1,MLTYP        ;IS INTERNAL LOOPBACK SPECIFIED ?
7196 063322 001420      BEQ     20$            ;YES. OMIT SPECIFYING MODEM CONTROLLER ADDRESS
7197      ;SPECIFY MODEM CONTROLLER (DM-11) CSR ADDRESS
7198
7199 063324 004737 065422      JSR     PC,TOORIO        ;REQUEST PERMISSION TO SEND TO FIRMWARE
7200      ;PERMISSION GRANTED
7201 063330 042777 177757 130040      BIC     #177757,@SEL2   ;CLEAR ALL BUT RDYI
7202 063336 152777 000007 130032      BISB   #7,@BSEL2       ;SPECIFY COMMAND PAIR (0,7)
7203 063344 013777 013426 130030      MOV     MODCSR,@SEL4    ;SPECIFY DM11-BA CSR ADDRESS

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 213
DVINIT - DEVICE INITIALIZATION

```

7204 063352 005077 130030          CLR  @SEL6           :CLEAR SEL6
7205 063356 142777 000020 130012 BICB #RDYI,@BSEL2    :ASSERT READY TO TRANSFER COMMAND
7206
7207 063364          20$:          :SPECIFY BDL BASE ADDRESS
7208
7209
7210 063364 004737 065422          JSR  PC,TOORIO      :REQUEST PERMISSION TO SEND COMMAND
7211                                     :PERMISSION GRANTED
7212 063370 042777 177757 130000    BIC  #177757,@SEL2  :CLEAR ALL BUT RDYI
7213 063376 152777 000006 127772    BISB #6,@BSEL2      :SPECIFY COMMAND PAIR (0,6)
7214 063404 012777 013434 127770    MOV  #BDLBAS,@SEL4  :SPECIFY BDL BASE ADDRESS
7215 063412 005077 127770          CLR  @SEL6           :CLEAR HIGH ADDRESS BITS (BITS 17:16)
7216 063416 142777 000020 127752    BICB #RDYI,@BSEL2    :ASSERT READY TO TRANSFER COMMAND
7217
7218                                     :SPECIFY ESTABLISHED LINE NUMBER
7219
7220                                     :GET ESTABLISHED LINE NUMBER
7221 063424 012737 177777 007642    MOV  #-1,INDEX      :MAKE INDEX =-1. DO FOLLOWING FOR EACH LINE..
7222                                     :INDEX=-1 TO 8. LINE 0 FIRST, LINE 7 LAST
7223 063432 004737 030402          DVES1: JSR  PC,GTVIN     :GET VALID INDEX (NEXT LINE NUMBER TO ESTAB.)
7224 063436 022737 000010 007642    CMP  #8,INDEX        :DONE?
7225 063444 001516          BEQ  DVINEX          :IF SO EXIT (RETURN TO CALLER)
7226
7227                                     :SPECIFY LINE NUMBER
7228 063446 004737 065422          JSR  PC,TOORIO      :REQUEST PERMISSION TO TRANSFER COMMAND
7229                                     :PERMISSION GRANTED
7230 063452 042777 177757 127716    BIC  #177757,@SEL2  :CLEAR ALL EXCEPT RDYI
7231 063460 152777 000004 127710    BISB #4,@BSEL2      :SPECIFY COMMAND (0,4)
7232 063466 005077 127710          CLR  @SEL4           :CLEAR SEL4
7233 063472 013737 007636 010164    MOV  TRIBN,TEMP      :COPY LINE NUMBER
7234 063500 013777 010164 127700    MOV  TEMP,@SEL6      :SPECIFY NUMBER OF LINES TO SCAN
7235 063506 142777 000020 127662    BICB #RDYI,@BSEL2    :SPECIFY READY TO TRANSFER COMMAND
7236
7237                                     :ENABLE LINE
7238
7239 063514 004737 065422          DVEST: JSR  PC,TOORIO :REQUEST PERMISSION TO SEND COMMAND
7240                                     :PERMISSION GRANTED
7241 063520 042777 177757 127650    BIC  #177757,@SEL2  :CLEAR ALL EXCEPT RDYI
7242 063526 152777 000040 127642    BISB #40,@BSEL2     :SPECIFY COMMAND PAIR (1,0)
7243 063534 113777 007636 127636    MOVB TRIBN,@BSEL3    :SPECIFY LINE NUMBER
7244 063542 005077 127634          CLR  @SEL4           :CLEAR SEL4
7245 063546 005077 127634          CLR  @SEL6           :CLEAR SEL6
7246 063552 142777 000020 127616    BICB #RDYI,@BSEL2    :ASSERT READY TO TRANSFER COMMAND
7247
7248                                     :SET DDCMP MODE
7249
7250 063560 004737 065422          DVSET: JSR  PC,TOORIO :REQUEST PERMISSION TO SEND COMMAND
7251                                     :PERMISSION GRANTED
7252 063564 042777 177757 127604    BIC  #177757,@SEL2  :CLEAR ALL EXCEPT RDYI
7253 063572 152777 000001 127576    BISB #1,@BSEL2      :SPECIFY COMMAND PAIR (0,1)
7254 063600 113777 007636 127572    MOVB TRIBN,@BSEL3    :SPECIFY LINE NUMBER
7255 063606 052737 001000 010230    BIS  #SETC,FLAG      :INDICATE WAITING FOR SET DDCMP MODE COMPLETION
7256 063614 042737 004000 010230    BIC  #TXTO,FLAG      :CLEAR TRANSMIT TIMEOUT SPECIFIER
7257 063622 005077 127554          CLR  @SEL4           :CLEAR SEL4
7258 063626 005077 127554          CLR  @SEL6           :CLEAR SEL6
7259 063632 142777 000020 127536    BICB #RDYI,@BSEL2    :ASSERT READY TO TRANSFER COMMAND

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 214
DVINIT - DEVICE INITIALIZATION

```

7260
7261 063640 004737 065622          10$: JSR   PC,TOORO      ;WAIT FOR SET DDCMP MODE COMPLETION
7262 063644 032737 001000 010230  BIT   #SETC,FLAG    ;HAS COMPLETION MESSAGE OCCURRED ?
7263 063652 001405                BEQ   15$           ;YES. CONTINUE
7264 063654 032737 004000 010230  BIT   #TXTO,FLAG    ;DID DEVICE TIMEOUT ON TRANSMIT ?
7265 063662 001336                BNE   DVSET        ;YES. REISSUE REQUEST TO START PROTOCOL
7266 063664 000765                BR    10$          ;NO. KEEP WAITING
7267          063666          15$:
7268
7269                                ;DELAY TO COMPENSATE FOR THE FIRMWARE'S FAILURE TO ACKNOWLEDGE
7270                                ;A STACK MESSAGE RECEIVED.
7271
7272 063666 012737 000001 010274  MOV   #1,TIMERS     ;WAIT 1 SECOND
7273 063674 005737 010274          20$: TST   TIMERS        ;IS TIMER EXPIRED ?
7274 063700 001375                BNE   20$          ; WAIT
7275 063702 000207          DVINEX: RTS   PC      ;RETURN TO CALLER
7276
7277
7278
7279
7280 --
7281
7282

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 215
LNSTAT - GET LINE STATUS

.SBTTL LNSTAT - GET LINE STATUS

7283
7284
7285
7286
7287
7288
7289
7290
7291
7292
7293
7294
7295
7296
7297
7298
7299
7300
7301
7302
7303
7304
7305
7306
7307
7308
7309
7310
7311
7312
7313
7314
7315
7316
7317
7318
7319
7320
7321
7322
7323
7324
7325
7326
7327
7328
7329

```

++
: FUNCTIONAL DESCRIPTION:
: LNSTAT - IF READ LINE STATUS IS ENABLED THIS SUBROUTINE REQUESTS
: THE KMS TO REPORT THE VALUES OF THE FIRST EIGHT BYTES IN
: THE LINE STATUS TABLE FOR THE ESTABLISHED LINE.

: INPUTS:
: ENRDST - READ STATUS ENABLED IF <> 0 / DISABLED IF = 0
: TRIBN - ESTABLISHED LINE NUMBER

: OUTPUTS:
: IF STATUS REPORT COMPLETED SUCCESSFULLY THEN
: VSTAT - SET = 1 TO SHOW THAT STATUS DATA IS VALID
: DEVS1..DEVS4 - EIGHT BYTES OF LINE STATUS DATA FROM KMS

: SUBORDINATE ROUTINES USED:
: TOORIO REQUEST INPUT INTERRUPT. WAIT FOR INTERRUPT OR TIME OUT
: TOORO WAIT FOR OUTPUT INTERRUPT OR TIMEOUT

: CALLING SEQUENCE:
: JSR PC,LNSTAT
:--

LNSTAT:
10$: TST ENRDST ;REQUEST LINE STATUS
      BEQ 40$ ;IS READ STATUS ENABLED ?
      JSR PC,TOORIO ;NO. OMIT READING STATUS
           ;REQUEST PERMISSION TO SEND COMMAND
           ;PERMISSION GRANTED
           ;SPECIFY WAITING FOR STATUS REPORT
           ;CLEAR ALL BUT RDYI
           ;SPECIFY REQUEST LINE STATUS (3,0)
           ;SPECIFY LINE NO.
           ;INDICATE READY TO TRANSFER COMMAND
20$: JSR PC,TOORO ;WAIT FOR STATUS TO BE RETURNED
           ;WAIT FOR MESSAGE FROM DEVICE
           ;MESSAGE RECEIVED
           ;DID A STATUS REPORT OCCUR ?
           ;NO. KEEP WAITING
30$: JSR PC,TOORO ;SPECIFY WAITING FOR STATUS REPORT
           ;WAIT FOR SECOND HALF OF STATUS REPORT
           ;DID A STATUS REPORT OCCUR ?
           ;NO. KEEP WAITING
      INC VSTAT ;ASSERT VALID STATUS DATA
      RTS PC ;RETURN
40$:

```

```

063704
063704 005737 011512
063710 001442
063712 004737 065422
063716 052737 002000 010230
063724 042777 177757 127444
063732 152777 000140 127436
063740 113777 007636 127432
063746 142777 000020 127422
063754 004737 065622
063760 032737 002000 010230
063766 001372
063770 052737 002000 010230
063776 004737 065622
064002 032737 002000 010230
064010 001372
064012 005237 011514
064016 000207

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 216
HALT LINE SUBROUTINE

```

7330 .SBTTL HALT LINE SUBROUTINE
7331
7332
7333 :++
7334 : FUNCTIONAL DESCRIPTION:
7335 : HLTTRB - THIS SUBROUTINE SHUTS DOWN THE ESTABLISHED LINE
7336
7337 : INPUTS:
7338 : PARAM - PARAMETERS. THIS ROUTINE CHECKS THE DATCKB BIT
7339 : TRIBLS - ESTABLISHED LINE NUMBER
7340 : FLAG - BITS ININT AND OTINT INDICATE INTERRUPTS
7341
7342 : SUBORDINATE ROUTINES USED:
7343 : GTVIND - COPY VALID LINE NUMBER TO TRIBN
7344 : TOORIO REQUEST INPUT INTERRUPT. WAIT FOR INTERRUPT OR TIME OUT
7345 : OUTHDL HANDLES OUTPUT INTERRUPT DATA
7346
7347 : CALLING SEQUENCE:
7348 : JSR PC,HLTTRB
7349 :--
7350 064020 032737 000002 010224 HLTTRB: BIT #DATCKB,PARAM
7351 064026 001006 BNE HLTTR2 ;IF CHECK GO TO 2
7352 064030 012737 000002 010274 MOV #2,TIMERS ;SET UP FOR 2 SEC TIMER
7353 064036 005737 010274 HLTTR3: TST TIMERS
7354 064042 001375 BNE HLTTR3 ;WAIT FOR TIMER TO BE 0
7355 064044
7356
7357 ;SHUTDOWN ACTIVE LINE. ONLY THE 1ST ESTABLISHED LINE IS USED
7358
7359 064044 012737 177777 007642 HLTTR1: MOV #-1,INDEX
7360 064052 004737 030402 JSR PC,GTVIND ;GET 1ST ESTABLISHED LINE NUMBER
7361 064056 022737 000010 007642 CMP #8,INDEX ;IS THERE AN ACTIVE LINE ?
7362 064064 001422 BEQ HLTREX ;NO. EXIT
7363
7364 064066 004737 065422 JSR PC,TOORIO ;SPECIFY SET DOWN LINE COMMAND PAIR
7365 ;PERMISSION GRANTED ;REQUEST PERMISSION TO SEND COMMAND TO FIRMWARE
7366 064072 042777 177757 127276 BIC #177757,@SEL2 ;CLEAR ALL EXCEPT RDYI
7367 064100 152777 000042 127270 BISB #42,@BSEL2 ;SPECIFY COMMAND (1,2) IN BITS 6:5,2:0
7368 064106 113777 007636 127264 MOVB TRIBN,@BSEL3 ;SPECIFY LINE NO.
7369 064114 005077 127262 CLR @SEL4 ;CLEAR SEL4
7370 064120 005077 127262 CLR @SEL6 ;CLEAR SEL6
7371 064124 142777 000020 127244 BICB #RDYI,@BSEL2 ;INDICATE READY TO TRANSFER COMMAND
7372
7373 064132 032737 000003 010230 HLTREX: BIT #ININT+OTINT,FLAG ;HAVE ANY INTERRUPTS OCCURRED ?
7374 064140 001411 BEQ 10$ ;NO.
7375 064142 042737 000003 010230 BIC #ININT+OTINT,FLAG ;YES. CLEAR INTERRUPT INDICATORS
7376 064150 132777 000200 127220 BITB #RDYO,@BSEL2 ;IS THERE A MESSAGE FROM THE KMS ?
7377 064156 001402 BEQ 10$ ;NO.
7378 064160 004737 066040 JSR PC,OUTHDL ;YES. COLLECT IT (IT WILL BE THROWN
7379 ; AWAY SINCE CLNSET = -1)
7380 064164 142777 000020 127204 10$: BICB #RDYI,@BSEL2 ;IGNORE RDYI
7381 064172 000207 RTS PC ;RETURN
7382
7383
7384
7385

```

CZKMSAG KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 217
DEVICE QUEUE RECEIVE SPACE SUBROUTINE

```

7386 .SBTTL DEVICE QUEUE RECEIVE SPACE SUBROUTINE
7387
7388
7389 :++
7390 : FUNCTIONAL DESCRIPTION:
7391 : DVRXQ - THIS SUBROUTINE QUES THE REC BUFFER SPACE TO THE
7392 : DEVICE, THEN CLEARS THE QRX BIT OF THE FLAG WORD.
7393
7394 : INPUTS:
7395 : DVRXA = ADDRESS OF RX BUFFER SPACE
7396 : DVRCC = BYTE CHAR COUNT OF RX BUFFER
7397 : QRX FLAG BIT = SET BY CALLING ROUTINE
7398 : TRIBN = LINE WHICH RECEIVE BUFF IS TO BE QUEUED FOR
7399
7400 : OUTPUTS:
7401 : QRX FLAG BIT = CLEARED BY ROUTINE
7402
7403 : SUBORDINATE ROUTINES USED:
7404
7405 : CALLING SEQUENCE:
7406 : JSR PC,DVRXQ
7407 :--
7408
7409 DVRXQ:
7410 BIT #QRX,FLAG ;RX QUEUED
7411 BEQ DVREX ;IF NOT RX THEN EXIT
7412
7413 BIC #QRX,FLAG ;ELSE QUE RX
7414 ;CLEAR FLAG FOR RX QUEUED
7415
7416 MOV TRIBN,R2 ;GET CHANNEL NUMBER (WHICH IS MEMBER NUMBER)
7417 ASL R2 ;MULTIPLY BY 8
7418 ASL R2 ;FOR OFFSET IN TABLE FOR THIS BDL MEMBER
7419 ASL R2 ;(8 BYTES PER MEMBER IN TABLE)
7420 ADD #RXBDL,R2 ;TABLE ADDRESS FOR RECEIVE BDL MEMBERS
7421 MOV R2,BDLMAD ;CALCULATE BDL MEMBER ADDRESS, SAVE AT BDL MAD
7422 MOV TRIBN,BDLMNM ;SAVE THE CHANNEL NUMBER, BDL MEMBER NUMBER
7423 INC BDLNMN ;CALCULATE BDL MEMBER NUMBER (CHAN + 1)
7424 ;RANGE IS NOW 1-8 FOR CHAN 0-7
7425 ;LOAD BUFFER DESCRIPTOR LIST MEMBER...
7426 MOV #BITS,(R2)+ ;SET STATUS WORD IN BDL MMBR BLOCK (RX QUEUED)
7427 MOV DVRXA,(R2)+ ;DVRXA CONTAINS THE RECEIVE BUFFER ADDRESS
7428 MOV DVRCC,(R2)+ ;DVRCC CONTAINS LENGTH OF THAT BUFFER
7429 CLR (R2)+ ;CLEAR DDCMP MESSAGE NUMBER FIELD
7430 ;SPECIFY BDL MEMBER FOR RECEIVE BUFFER
7431 JSR PC,TOORIO ;REQUEST PERMISSION TO SEND COMMAND TO FIRMWARE
7432 ;PERMISSION GRANTED
7433 BIC #177757,@SEL2 ;CLEAR ALL EXCEPT RDYI
7434 BISB #101,@BSEL2 ;SPECIFY COMMAND PAIR (2,1)
7435 MOVB TRIBN,@BSEL3 ;SPECIFY LINE NUMBER
7436 CLR @SEL4 ;CLEAR SEL4
7437 CLRB @BSEL6 ;CLEAR BSEL6
7438 MOVB BDLNMN,@BSEL7 ;SPECIFY BDL MEMBER NUMBER
7439 ;BDL MEMBER NUMBER RANGE IS 1-8
7440 BICB #RDYI,@BSEL2 ;ASSERT READY TO TRANSFER COMMAND
7441
7442 DVREX: RTS PC ;RETURN TO CALLER

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 218
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

7441 .SBTTL                                DEVICE TRANSMIT AND RECEIVE SUBROUTINE
7442
7443
7444 : **
7445 : FUNCTIONAL DESCRIPTION:
7446 : DVTXRX-DEVICE TRANSMIT AND RECEIVE ROUTINE.
7447 : THIS CODE QUEUES THE TRANSMIT BUFFER TO THE DEVICE (IF NEEDED).
7448 : THE CODE THEN WAITS FOR A TX COMPLETE, AN RX COMPLETE OR A
7449 : RETRANSMIT REQUEST. THE CODE REPORTS A TIME OUT ERROR IF NONE
7450 : OF THESE IS REPORTED BACK IN 60 SECONDS. AFTER REPORTING ERROR,
7451 : THE TIMER IS RE STARTED AND THE DEVICE WILL CONTINUE TO WAIT FOR
7452 : AN INTERRUPT.
7453 : A NAK RESPONSE GENERATES A RETRANSMIT REQUEST. THIS CODE REQUEUES
7454 : THE TRANSMIT BUFFER TO THE DEVICE AND RESTARTS THE TIMER IF A
7455 : RETRANSMIT IS REQUESTED.
7456
7457 : INPUTS:
7458 : "DVTXA" = ADDRESS OF TRANSMIT MSG.
7459 : "DVTCC" = BYTE COUNT OF TRANSMIT MSG.
7460 : "QTX" BIT = SET IF TRANSMIT REQUESTED
7461 : "ETX" BIT = SET IF TRANSMIT EXPECTED
7462 : "ERX" BIT = SET IF RECEIVE EXPECTED
7463 : "RXMIT" BIT = SET IF RETRANSMIT REQUESTED
7464
7465 : OUTPUTS:
7466 : "DVTXA" = ADDRESS OF TX MSG. COMPLETED
7467 : "DVTCC" = BYTE COUNT OF TX MSG. COMPLETED
7468 : "QTX" = SET IF TX QUEUED
7469 : "DVRXA" = ADDRESS OF RX MSG. COMPLETED
7470 : "DVRCC" = BYTE COUNT OF RX MSG. COMPLETED
7471 : "QRX" = SET IF RX QUEUED
7472
7473 : SUBORDINATE ROUTINES USED: TCORIO, OUTHDL
7474
7475 : CALLING SEQUENCE: JSR PC,DVTXRX
7476 : --
7477
7478 064340 032737 000010 010230 DVTXRX: BIT #QTX,FLAG ;IS THERE A TRANSMIT QUEUED?
7479 064346 001457 BEQ DVTR3 ;IF NOT GO WAIT FOR OUTPUT
7480 064350 042737 000010 010230 BIC #QTX,FLAG ;TRANSMIT NOT QUEUED, TRANSMIT COMPLETE
7481
7482 ;SPECIFY "TRANSMIT MESSAGE" COMMAND
7483 064356 004737 065422 JSR PC,TCORIO ;REQUEST PERMISSION TO SEND COMMAND TO DEVICE
7484 ;PERMISSION GRANTED
7485 064362 013702 007636 MOV TRIBN,R2 ;GET THE DESIRED LINE NUMBER (0-7)
7486 064366 006302 ASL R2 ;\
7487 064370 006302 ASL R2 ;8*LINE_NUMBER=>R2 (0,8,16,24,32,40,48,56)
7488 064372 006302 ASL R2 ;/
7489 064374 062702 013534 ADD #TXBDL,R2 ;ADD THE XMIT BDL BASE ADDRESS
7490 064400 010237 013432 MOV R2,BDLMAD ;BDL MEMBER ADDRESS=>BDLMAD
7491 064404 013737 007636 013430 MOV TRIBN,BDLMNM ;LINE NUMBER = 0-7
7492 ;RECEIVE BDL MEMBER NUMBERS ARE 1-8
7493 ;TRANSMIT BDL MEMBER NUMBERS ARE 9-16
7494 064412 062737 000011 013430 ADD #9,BDLMNM ;CALCULATE XMIT BDL MEMBER NUMBER => BDLMNM
7495 064420 012722 100000 MOV #BIT15,(R2)+ ;SET STATUS WORD IN BDL MEMBER BLOCK
7496 064424 013722 010064 MOV DVTXA,(R2)+ ;BUFFER ADDRESS FOR XMISSION

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 219

DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

7497 064430 013722 010066      MOV      DVTCC,(R2)+      :CHARACTER COUNT FOR XMISSION
7498 064434 005022             CLR      (R2)+           :CLEAR THE DDCMP MESSAGE NUMBER FIELD
7499 064436 042777 177757 126732 BIC      #177757,@SEL2   :CLEAR ALL EXCEPT RDYI
7500 064444 152777 000100 126724 BISB     #100,@BSEL2     :SPECIFY COMMAND PAIR (2,0) IN BITS 6:5,2:0
7501 064452 113777 007636 126720 MOVB     TRIBN,@BSEL3    :SPECIFY LINE NO.
7502 064460 005077 126716      CLR      @BSEL4         :CLEAR UNUSED BSEL4
7503 064464 112777 000001 126714 MOVB     #1,@BSEL6       :SPECIFY NUMBER OF MSGS TO TX=1
7504 064472 113777 013430 126710 MOVB     BDLNM,@BSEL7    :SPECIFY BDL MEMBER NUMBER
7505 064500 142777 000020 126670 BICB     #RDYI,@BSEL2    :INDICATE READY TO TRANSFER COMMAND
7506 064506
7507
7508 064506 005737 013642      TST      RXMIT          :IS RETRANSMIT ASSERTED ?
7509 064512 001440             BEQ      10$           :NO. GO SET TIMER.
7510
7511                               :RETRANSMIT
7512 064514 004737 065422      JSR      PC,TOORIO      :REQUEST PERMISSION TO SEND DEVICE A COMMAND
7513 064520 042777 177757 126650 BIC      #177757,@SEL2   :CLEAR ALL EXCEPT RDYI BIT
7514 064526 152777 000100 126642 BISB     #100,@BSEL2     :SPECIFY TRANSMIT COMMAND
7515 064534 013702 007654      MOV      TSPTR,R2       :POINT TO TX SAVE AREA
7516 064540 014237 010164      MOV      -(R2),TEMP     :COPY SAVED SEL2
7517 064544 113777 010165 126626 MOVB     TEMP+1,@BSEL3   :SPECIFY LINE NUMBER
7518 064552 005077 126624      CLR      @SEL4         :CLEAR SEL4
7519 064556 014237 010164      MOV      -(R2),TEMP     :COPY SAVED SEL6
7520 064562 112737 000001 010164 MOVB     #1,TEMP         :SET NUMBER OF MSGS = 1
7521 064570 013777 010164 126610 MOV      TEMP,@SEL6     :SPECIFY BDL AND MSG NUMBERS
7522 064576 142777 000020 126572 BICB     #RDYI,@BSEL2    :SPECIFY READY TO TRANSFER COMMAND
7523 064604 010237 007654      MOV      R2,TSPTR      :RESTORE TX SAVE AREA POINTER
7524 064610 005337 013642      DEC      RXMIT         :DECREMENT RETRANSMIT COUNT
7525
7526 064614 012737 000074 010274 10$: MOV      #60.,TIMERS     :SET TIMER FOR 60 SECS !!!FOR DEBUG =30 !!!!
7527
7528                               :CHECK FOR TX OR RX COMPLETION
7529 064622
7530 064622 005737 013642      TST      RXMIT          :IF RETRANSMIT ASSERTED
7531 064626 001327             BNE     DVTR3          :GO REQUEUE TX BUFFER
7532 064630 005737 007646      TST      CRX           :IF RX COMPLETED
7533 064634 001053             BNE     DVTR4          :GO PROCESS COMPLETION
7534 064636 005737 007644      TST      CTX           :IF TX COMPLETED
7535 064642 001050             BNE     DVTR4          :GO PROCESS COMPLETION
7536 064644 005737 010274      TST      TIMERS        :IS TIMER EXPIRED
7537 064650 001027             BNE     TOIN1          :NO. GO CHECK INTERRUPTS
7538
7539 064652 012737 022223 010170 MOV      #DVEM2,TEMP2    :SPECIFY ERROR MESSAGE
7540 064660 005037 010172      CLR      TEMP3         :SPECIFY ERROR CODE
7541 064664 017737 126502 010174 MOV      @SELO,TEMP4     :COPY SELO
7542 064672 017737 126500 010176 MOV      @SEL2,TEMP5     :COPY SEL2
7543 064700 113737 010175 010173 MOVB     TEMP4+1,TEMP3+1 :COPY LINE NUMBER FROM TEMP4
7544 064706 005237 010124      INC      ERRCNT        :INCREMENT ERROR COUNT
7545 064712 004737 024226      JSR      PC,LGDVE      :LOG TIME OUT
7546 000005      ENUM = ^D<ENUM+1>    :INCREMENT ERROR NUMBER
7547 064716
7548 064716 104457             ERRSOF T ENUM,DVEM2,ERR13 :REPORT TIME OUT ERROR
7549 064720 000005
7550 064722 022223
7551 064724 023460
7552 064726 000667      BR      DVTR3         :REPEAT TIME CHECK

```

TRAP CSERSOF T
.WORD 5
.WORD DVEM2
.WORD ERR13

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 220
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

7553
7554                                     ;CHECK FOR INTERRUPT
7555 064730                               ;ALLOW CNTRL/C EXIT
7556 064730 104422                       TRAP   C$BRK
7557 064732 032737 000003 010230 TOIN1: BREAK
7558 064740 001730                               ;HAS AN INTERRUPT OCCURRED ?
7559                                     ;NO. KEEP CHECKING
7560 064742 042737 000003 010230 TOIN2: BIT #OTINT+ININT,FLAG ;INTERRUPT OCCURRED
7561 064750 132777 000200 126420 BEQ TOINOT ;CLEAR INTERRUPT INDICATORS
7562 064756 001721                               ;IS THERE A MESSAGE FROM THE DEVICE
7563                                     ;NO KEEP CHECKING
7564 064760 004737 066040 JSR PC,OUTHDL ;MESSAGE FROM DEVICE
7565                                     ;GO PROCESS MESSAGE FROM DEVICE
7566 064764 005737 013642 DVTR4: TST RXMIT ;CHECK IF RX OR TX COMPLETED
7567 064770 001246 BNE DVTR3 ;IS RETRANSMIT ASSERTED ?
7568 064772 005737 007644 TST CTX ;YES. GO RETRANSMIT
7569 064776 001004 BNE DVTR4A ;HAS A TX COMPLETED
7570 065000 005737 007646 10$: TST CRX ;YES. GO PROCESS COMPLETION
7571 065004 001077 BNE DVTR5 ;HAS AN RX COMPLETED ?
7572 065006 000705 BR TOINOT ;YES. GO PROCESS COMPLETION
7573                                     ;NO. KEEP CHECKING
7574 DVTR4A: ;TX COMPLETED
7575 065010 MOV TSPTR,R2 ;POINT TO TX SAVE AREA
7576 065014 013702 007654 MOV -(R2),TRIBN ;COPY SAVED TX LINE NUMBER
7577 065020 014237 007636 CLRB TRIBN ;CLEAR LOW BYTE
7578 065024 105037 007636 SWAB TRIBN ;POSITION LINE NUMBER IN LOW BYTE
7579 065030 014237 010164 MOV -(R2),TEMP ;COPY SAVED TX BDL NUMBER
7580 065034 105037 010164 CLRB TEMP ;CLEAR LOW BYTE
7581 065040 000337 010164 SWAB TEMP ;POSITION BDL NUMBER IN LOW BYTE
7582 065044 010237 007654 MOV R2,TSPTR ;RESTORE TX SAVE AREA POINTER
7583 065050 032737 000200 010230 BIT #ETX,FLAG ;YES. IS TX COMPLETION EXPECTED ?
7584 065056 001025 BNE DVTR4B ;YES. GO PROCESS TRANSMIT COMPLETED
7585 ;UNEXPECTED TRANSMIT COMPLETION
7586 065060 012737 022330 010170 MOV #DVEM4,TEMP2 ;LOG UNEXPECTED TRANSMIT ERROR
7587 065066 012737 000002 010172 MOV #2,TEMP3 ;SPECIFY ERROR CODE
7588 065074 113737 007636 010173 MOVB TRIBN,TEMP3+1 ;RECORD LINE NUMBER
7589 065102 013737 010164 010174 MOV TEMP,TEMP4 ;RECORD BDL NUMBER FOR ERROR
7590 065110 005237 010124 INC ERRCNT ;INCREMENT ERROR COUNT
7591 065114 004737 024226 JSR PC,LGDVE ;LOG EVENT
7592 000006 ENUM = *D<ENUM+1> ;INCREMENT ERROR NUMBER
7593 065120 ERRSOF T ENUM,DVEM4,ERR2 ;REPORT ERROR (UNEXPECTED CTX)
7594 065120 104457 TRAP   C$ERSOF T
7595 065122 000006 .WORD 6
7596 065124 022330 .WORD DVEM4
7597 065126 023542 .WORD ERR2
7598 065130 000423 BR DVTR4C ;THEN CLEAR COMPL.FLAG
7599 ;NOTE: COUNT REDUCED AT DVTR4C
7600 DVTR4B: ;EXPECTED TRANSMIT COMPLETE
7601 065132 013703 010164 MOV TEMP,R3 ;COPY BDL NUMBER
7602 065136 005303 DEC R3 ;COMPUTE OFFSET
7603 065140 006303 ASL R3 ;FOR
7604 065142 006303 ASL R3 ; THIS
7605 065144 006303 ASL R3 ; MEMBER
7606 065146 062703 013434 ADD #BDLBAS,R3 ;FORM BDL TABLE POINTER
7607 065152 005023 CLR (R3)+ ;CLEAR BDL STATUS WORD
7608 065154 012337 010064 MOV (R3)+,DVTXA ;UNLOAD TX MESSAGE ADDRESS

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 221

DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

7609 065160 012337 010066      MOV      (R3)+,DVTCC      ;UNLOAD TX MESSAGE CHAR COUNT
7610 065164 013737 007636 010070  MOV      TRIBN,DVTTB    ;STORE LINE NUMBER FOR LOGGING
7611 065172 052737 000010 010230  BIS      #QTX,FLAG      ;SET QUE-TRANSMIT FLAG BIT
7612 065200      DVTR4C:
7613 065200 005337 007644      DEC      CTX            ;DECREMENT TX COUNT
7614 065204      DVTR5:
7615 065204 005737 007646      TST      CRX            ;ANY RECEIVES COMPLETED ?
7616 065210 001503      BEQ      DVTREX        ;NO. GO RETURN
7617      ;RECEIVE COMPLETED
7618 065212 013702 007650      MOV      RSPTRS,R2      ;POINT TO RECEIVE SAVE AREA
7619 065216 012237 010164      MOV      (R2)+,TEMP     ;COPY SAVED BDL NUMBER
7620 065222 105037 010164      CLRB    TEMP           ;CLEAR LO BYTE
7621 065226 000337 010164      SWAB    TEMP           ;POSITION BDL NUMBER IN LO BYTE
7622 065232 012237 007636      MOV      (R2)+,TRIBN    ;COPY SAVED LINE NUMBER
7623 065236 105037 007636      CLRB    TRIBN          ;CLEAR LO BYTE
7624 065242 000337 007636      SWAB    TRIBN          ;POSITION LINE NUM IN LO BYTE
7625 065246 020227 010046      CMP     R2,#RXSKEN     ;POINTING TO END OF RX SAVE AREA ?
7626 065252 001002      BNE     2$             ;NO. KEEP POINTER
7627 065254 012702 007672      MOV     #RXSTAK,R2     ;YES. RESET POINTER
7628 065260 010237 007650      MOV     R2,RSPTRS      ;SAVE POINTER
7629 065264 032737 000100 010230  BIT     #ERX,FLAG      ;IS RX COMPLETE EXPECTED ?
7630 065272 001025      BNE     DVTR5B         ;YES. GO PROCESS RX COMPLETED
7631      ;UNEXPECTED RECEIVE COMPLETION
7632 065274 012737 022407 010170  MOV     #DVEM5,TEMP2   ;RECORD UNEXPECTED RX COMPLETION
7633 065302 012737 000002 010172  MOV     #2,TEMP3       ;SPECIFY ERROR CODE
7634 065310 113737 007636 010173  MOVB   TRIBN,TEMP3+1  ;RECORD LINE NUMBER
7635 065316 013737 010164 010174  MOV     TEMP,TEMP4     ;COPY BDL NUMBER
7636 065324 005237 010124      INC     ERRCNT         ;INCREMENT ERROR COUNT
7637 065330 004737 024226      JSR    PC,LGDVE        ;LOG EVENT
7638      ENUM = ^D<ENUM+1> ;INCREMENT ERROR NUMBER
7639 065334      ERRSOFT ENUM,DVEM5,ERR22 ;REPORT ERROR
7640 065334 104457      TRAP   C$ERSOFT
7641 065336 000007      .WORD 7
7642 065340 022407      .WORD DVEM5
7643 065342 023542      .WORD ERR22
7644 065344 000423      BR     DVTR5C         ; AND EXIT
7645
7646 065346      DVTR5B:
7647 065346 013703 010164      MOV     TEMP,R3        ;EXPECTED RX COMPLETED
7648 065352 005303      DEC     R3             ;COPY BDL MEMBER NUMBER
7649 065354 006303      ASL    R3             ;COMPUTE OFFSET
7650 065356 006303      ASL    R3             ; FOR
7651 065360 006303      ASL    R3             ; THIS
7652 065362 062703 013434      ADD     #BDLBAS,R3     ;POINT TO THIS MEMBER IN BDL AREA
7653 065366 005023      CLR    (R3)+          ;CLEAR BDL STATUS WORD
7654 065370 012337 010104      MOV    (R3)+,DVRXA    ;UNLOAD RX MESSAGE ADDRESS
7655 065374 012337 010106      MOV    (R3)+,DVRCC    ;UNLOAD RX MESSAGE CHAR COUNT
7656 065400 013737 007636 010102  MOV    TRIBN,DVRTB    ;STORE LINE NUMBER FOR LOGGING
7657 065406 052737 000004 010230  BIS    #QRX,FLAG      ;SET FLAG TO QUE ANOTHER RECEIVER
7658 065414      DVTR5C:
7659 065414 005337 007646      DEC     CRX
7660 065420 000207      DVTREX: RTS           PC
7661
7662
7663
7664

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 222
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

7665

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 223
TOORIO REQUEST PERMISSION TO GIVE DEVICE A CONTROL MESSAGE

.SBTTL TOORIO REQUEST PERMISSION TO GIVE DEVICE A CONTROL MESSAGE

..**
FUNCTIONAL DESCRIPTION:
TOORIO - THIS SUBROUTINE REQUESTS AN INPUT INTERRUPT THEN
WAITS FOR AN INTERRUPT OR A TIMEOUT.
IF AN OUTPUT INTERRUPT OCCURS OULTHDL IS CALLED
TO HANDLE THE OUTPUT DATA, AND THE REQUEST IS REISSUED.
IF AN INPUT INTERRUPT OCCURS CONTROL RETURNS TO THE CALLER.
IF A TIMEOUT OCCURS AN ERROR IS REPORTED AND THE REQUEST
IS REISSUED.

INPUTS:
FLAG - OTINT BIT - SET BY OUTPUT INT ROUTINE
ININT BIT - SET BY INPUT INT. ROUTINE
CLEARED BY THIS ROUTINE.

OUTPUTS:
ERRCNT - INCREMENTED IF A TIMEOUT OCCURS

SUBORDINATE ROUTINES USED:
LGDVE - LOG DEVICE ERRORS
OUTHDL - OUTPUT INTERRUPT HANDLER

CALLING SEQUENCE:
JSR PC,TOORIO

```

7694 065422 011637 010210 TOORIO: MOV (SP),PCADD ;SAVE ADDR. OF CALLING ROUTINE
7695 065426 152777 000021 125736 TOORIO: BISB #IEO!IEI,@BSELO ;ENABLE INTERRUPTS
7696 065434 TOOR11:
7697 065434 152777 000200 125730 TOOR11: BISB #RQI,@BSELO ;ASSERT RQI
7698 065442 012737 002000 010270 TOOR11: MOV #2000,TIMER1 ;SET TIMER
7699 065450 005737 010270 TOOR12: TST TIMER1 ;IS TIME EXPIRED ?
7700 065454 001027 TOOR12: BNE TOOR13 ;NO. CONTINUE
7701 ;TIME EXPIRED WAITING FOR RDI
7702 065456 142777 000200 125706 TOOR12: BICB #RQI,@BSELO ;YES. NEGATE RQI
7703 065464 012737 022275 010170 TOOR12: MOV #DVEM3,TEMP2 ;REPORT TIME OUT WAITING FOR RDI
7704 065472 005037 010172 TOOR12: CLR TEMP3 ;SPECIFY ERROR CODE
7705 065476 017737 125670 010174 TOOR12: MOV @SELO,TEMP4 ;COPY BSELO
7706 065504 017737 125666 010176 TOOR12: MOV @SEL2,TEMP5 ;COPY BSEL2
7707 065512 005237 010124 TOOR12: INC ENRCNT ;INCREMENT ERROR COUNT
7708 065516 004737 024226 TOOR12: JSR PC,LGDVE ;LOG THE DEVICE ERROR
7709 ENUM = *D<ENUM+1> ;INCREMENT ERROR NUMBER
7710 065522 000010 ERRSOF1 ENUM,DVEM3,ERR13 ;PRINT THE ERROR
7711 065522 104457 TRAP C$ERSOF1
7712 065524 000010 .WORD 8
7713 065526 022275 .WORD DVEM3
7714 065530 023460 .WORD ERR13
7715 065532 000740 TOOR13: BR TOOR11 ;GO BACK AND TRY AGAIN
7716 065534 TOOR13: BREAK ;ALLOW CNTRL/C EXIT
7717 065534 TRAP C$BRK
7718 065534 104422
7719
7720 065536 032737 000003 010230 TOOR13: BIT #OTINT+ININT,FLAG ;CHECK FOR AN INTERRUPT
7721 065544 001741 TOOR13: BEQ TOOR12 ;HAS AN INTERRUPT OCCURRED ?
;NO. WAIT FOR ONE

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 224

TOORIO REQUEST PERMISSION TO GIVE DEVICE A CONTROL MESSAGE

```

7722 065546 042737 000003 010230      BIC      #OTINT+ININT,FLAG      ;YES, CLEAR INTERRUPT OCCURED INDICATORS
7723                                     ;CHECK FOR A MESSAGE FROM THE FIRMWARE (RDYO)
7724 065554 132777 000200 125614      BITB     #RDYO,@BSEL2      ;IS THERE A MESSAGE FROM THE FIRMWARE ?
7725 065562 001407                                     BEQ      TOOR14             ;NO, LOOK FOR PERMISSION TO SEND
7726                                     ;MESSAGE RECEIVED FROM FIRMWARE
7727 065564 004737 066040                                     ;READ MESSAGE FROM FIRMWARE
7728 065570 132777 000020 125600      BITB     #RDYI,@BSEL2      ;IS PERMISSION TO SEND ASSERTED ?
7729 065576 001005                                     BNE      TOOR15             ;YES. NEGATE RQI AND RETURN
7730 065600 000715                                     BR       TOOR11             ;NO. RESET TIMER AND WAIT
7731
7732 065602                                     TOOR14: ;CHECK FOR RESPONSE TO RQI
7733 065602 132777 000020 125566      BITB     #RDYI,@BSEL2      ;IS IT OK TO SEND A MESSAGE TO FIRMWARE ?
7734 065610 001717                                     BEQ      TOOR12             ;NO. GO CHECK TIMER
7735 065612 142777 000200 125552      TOOR15: BICB     #RQI,@BSEL0      ;NEGATE RQI
7736 065620 000207                                     RTS      PC                 ;RETURN TO CALLER
7737
7738
7739
7740

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1(52) 23-DEC-82 14:01 PAGE 225
TOORO WAIT FOR AN EXPECTED CONTROL MESSAGE FROM DEVICE

.SBTTI TOORO WAIT FOR AN EXPECTED CONTROL MESSAGE FROM DEVICE

7741
7742
7743
7744
7745
7746
7747
7748
7749
7750
7751
7752
7753
7754
7755
7756
7757
7758
7759
7760
7761
7762
7763
7764
7765
7766
7767
7768
7769
7770
7771
7772
7773
7774
7775
7776
7777
7778
7779
7780
7781
7782
7783
7784
7785
7786
7787
7788
7789
7790
7791
7792
7793
7794
7795
7796

```

:++
:TOORO
:FUNCTIONAL DESCRIPTION:
:TOORO - THIS SUBROUTINE WAITS FOR AN OUTPUT INTERRUPT
:          A TIMEOUT OR A CONTROL-C TYPED BY THE OPERATOR.
:          IF AN OUTPUT INTERRUPT OCCURS OUTHDL IS CALLED TO
:          HANDLE THE INTERRUPT DATA.
:          IF A TIMEOUT OCCURS AN ERROR IS REPORTED AND LOGGED
:          AND THE TIMER IS RESTARTED.
:          IF CONTROL-C IS TYPED THE PROGRAM EXITS TO THE
:          DR> PROMPT.
    
```

```

:INPUTS:
:FLAG - OTINT BIT - SET BY OUTPUT INT ROUTINE
:ININT BIT - SET BY INPUT INTERRUPT ROUTINE
:          CLEARED BY THIS ROUTINE
    
```

```

:SUBORDINATE ROUTINES USED:
:LG DVE - LOG DEVICE ERRORS
:OUTHDL - OUTPUT INTERRUPT HANDLER
    
```

```

:CALLING SEQUENCE:
:JSR PC,TOORO
:--
    
```

```

TOORO: MOV (SP),PCADD ;SAVE ADDR. OF CALLING ROUTINE
        MOV #IEO+IEI,@BSEL0 ;ENABLE INTERRUPTS
        ;SET TIMER
TOORO1: MOV #2000,TIMER1 ;SET TIMER
        ;CHECK TIME
TOORO2: TST TIMER1 ;TIME EXPIRED ?
        BNE TOORO3 ;NO. CONTINUE
        ;TIME EXPIRED
        ;IS KMS INITIALIZATION REQUESTED ?
        BIT #INITC,FLAG ;NO
        BEQ 10$ ;SPECIFY ERROR MESSAGE
        MOV #DVEM1A,TEMP2
        BR 30$
10$: BIT #SETC,FLAG ;IS START PROTOCOL REQUESTED ?
        BEQ 20$ ;NO
        MOV #DVEM1B,TEMP2 ;SPECIFY ERROR MESSAGE
        BR 30$
20$: MOV #DVEM1,TEMP2 ;TIMED OUT. ERROR MESSAGE...
30$: MOV TEMP2,ERRMSG ;SPECIFY MESSAGE FOR ERROR REPORT
        CLR TEMP3 ;SPECIFY ERROR CODE
        MOV @SELO,TEMP4 ;COPY BSEL0
        MOV @SEL2,TEMP5 ;COPY BSEL2
        INC ERRCNT ;INCREMENT ERROR COUNT
        JSR PC,LGDVE ;LOG ERROR
ENUM = ^D<ENUM+1> ;INCREMENT ERROR NUMBER
    
```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 227
OUTPUT INTERRUPT HANDLER

7817
7818
7819
7820
7821
7822
7823
7824
7825
7826
7827
7828
7829
7830
7831
7832
7833
7834
7835
7836
7837
7838
7839
7840
7841
7842
7843
7844
7845
7846
7847
7848
7849
7850
7851
7852
7853
7854
7855
7856
7857
7858
7859
7860
7861
7862
7863
7864
7865
7866
7867
7868
7869
7870
7871
7872

.SBTTL OUTPUT INTERRUPT HANDLER

++
FUNCTIONAL DESCRIPTION:
OUTHDL - OUTPUT INTERRUPT HANDLER
THIS ROUTINE IS CALLED WHEN AN INTERRUPT HAS OCCURRED
WITH RDYO ASSERTED (BSEL2.BIT7 = 1).
COPY SEL2,SEL4,SEL6
CLEAR RDYO TO SAY MESSAGE TRANSFER COMPLETED

USE OF FLAGS:
"INITC" - SET BY DVINIT IF WAITING FOR COMPLETION
OF SYSTEM INITIALIZATION
CLEARED BY THIS ROUTINE IF COMMAND PAIR 0,0
IS RETURNED BY FIRMWARE
"CTX" - SET IF TRANSMIT COMPLETED
"CRX" - SET IF RECEIVE COMPLETED

SUBORDINATE ROUTINES USED:
"LGDVE" -LOG DEVICE ERRORS TO EVENT LOG

CALLING SEQUENCE
JSR PC,OUTHDL

--
OUTHDL: ;COPY FIRMWARE MESSAGE REGISTERS
MOV @SELO,SVSELO ;COPY SELO
MOV @SEL2,SVSEL2 ;COPY SEL2
MOV @SEL4,SVSEL4 ;COPY SEL4
MOV @SEL6,SVSEL6 ;COPY SEL6
;ASSERT MESSAGE TRANSFER COMPLETED
BICB #RDYO,@BSEL2 ;CLEAR RDYO
;CHECK FOR CLEANUP MODE
TST CLNSET ;IF CLEANUP MODE ENTERED (CLNSET = -1)
BNE OUTEND ; THEN RETURN
;ISOLATE COMMAND PAIR
MOV SVSEL2,R2 ;COPY SAVED SEL2
MOV R2,R3 ;COPY AGAIN
ASR R2 ;SHIFT COMMAND FROM BITS 5-6
ASR R2 ; TO BITS 3-4
BIC #^C<30>,R2 ;ISOLATE COMMAND IN R2 BITS 4:3
BIC #^C<7>,R3 ;ISOLATE SUBCOMMAND IN R3 BITS 2:0
;COMPUTE COMMAND PROCESSOR ADDRESS
BIS R2,R3 ;COMBINE COMMAND,SUBCOMMAND
MOV R3,CMDSUB ;SAVE COMMAND,SUBCOMMAND PAIR
;CHECK COMMAND PAIR IN VAILD RANGE (37:0)
CMP #37,R3 ;IS COMMAND PAIR WITHIN VALID RANGE ?
BHIS 10\$;YES. GO GET PROCESSOR ADDRESS
;COMMAND OUTSIDE VALID RANGE
MOV ADRINV,R3 ;NO. GET OFFSET FOR INVALID COMMAND PROCESSOR

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 228
OUTPUT INTERRUPT HANDLER

```

7873 066146 000403          BR      20$          ;GO DO PROCESS
7874                                ;COMMAND WITHIN VALID RANGE
7875 066150 006303          10$: ASL      R3          ;MULTIPLY COMMAND PAIR BY 2
7876 066152 016303 066166   MOV      CMDTBL(R3),R3    ;GET OFFSET TO COMMAND ACTION ROUTINE
7877 066156 062703 066166   20$: ADD      #CMDTBL,R3 ;ADD IN BASE ADDRESS
7878                                ;PROCESS COMMAND
7879 066162 004713          JSR      PC,(R3)        ;EXECUTE COMMAND ACTION ROUTINE
7880 066164 000207          OUTEND: RTS      PC          ;RETURN
7881
7882 066166          CMDTBL: ;BASE ADDRESS FOR ACCESSING COMMAND ACTION ROUTINES
7883                                ;TABLE OF COMMAND PROCESSOR ADDRESS OFFSETS
7884 066166 000110          .WORD    COS0-CMDTBL
7885 066170 000120          .WORD    COS1-CMDTBL
7886 066172 000102          .WORD    COS2-CMDTBL
7887 066174 000102          .WORD    COS3-CMDTBL
7888 066176 000174          .WORD    COS4-CMDTBL
7889 066200 000102          .WORD    COS5-CMDTBL
7890 066202 000102          .WORD    COS6-CMDTBL
7891 066204 000102          .WORD    COS7-CMDTBL
7892 066206 000102          .WORD    C1S0-CMDTBL
7893 066210 000102          .WORD    C1S1-CMDTBL
7894 066212 000102          .WORD    C1S2-CMDTBL
7895 066214 000222          .WORD    C1S3-CMDTBL
7896 066216 000230          .WORD    C1S4-CMDTBL
7897 066220 000344          .WORD    C1S5-CMDTBL
7898 066222 000102          .WORD    C1S6-CMDTBL
7899 066224 000446          .WORD    C1S7-CMDTBL
7900 066226 000512          .WORD    C2S0-CMDTBL
7901 066230 000552          .WORD    C2S1-CMDTBL
7902 066232 000102          .WORD    C2S2-CMDTBL
7903 066234 000102          .WORD    C2S3-CMDTBL
7904 066236 000102          .WORD    C2S4-CMDTBL
7905 066240 000102          .WORD    C2S5-CMDTBL
7906 066242 000102          .WORD    C2S6-CMDTBL
7907 066244 000102          .WORD    C2S7-CMDTBL
7908 066246 000560          .WORD    C3S0-CMDTBL
7909 066250 000616          .WORD    C3S1-CMDTBL
7910 066252 000642          .WORD    C3S2-CMDTBL
7911 066254 000660          .WORD    C3S3-CMDTBL
7912 066256 000704          .WORD    C3S4-CMDTBL
7913 066260 000102          .WORD    C3S5-CMDTBL
7914 066262 000102          .WORD    C3S6-CMDTBL
7915 066264 000102          .WORD    C3S7-CMDTBL
7916 066266 000102          ADRINV: .WORD    INVCMD-CMDTBL
7917

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 229
ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

.SBTTL ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

:THE FOLLOWING RESPONSES ARE CONSIDERED TO BE ILLEGAL
:IN THIS PROGRAM

7918
7919
7920
7921
7922 066270
7923 066270
7924 066270
7925 066270
7926 066270
7927 066270
7928 066270
7929 066270
7930 066270
7931 066270
7932 066270
7933 066270
7934 066270
7935 066270
7936 066270
7937 066270
7938 066270
7939 066270
7940 066270
7941 066270
7942 066274
7943

005237 013634
000207

COS2:
COS3:
COS5:
COS6:
COS7:
C1S0:
C1S1:
C1S2:
C1S6:
C2S2:
C2S3:
C2S4:
C2S5:
C2S6:
C2S7:
C3S5:
C3S6:
C3S7:
INVCMD:

INC
RTS
ILLRSP
PC

:ACTIVITY TERMINATED ON LINE (NOT USED IN THIS PROGRAM)
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:TOO MANY RECEIVE BUFFERS FOR LINE.MAX OF 2 PER LINE
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:RESERVED
:DEBUG

:COMMAND NOT WITHIN VALID RANGE
:COUNT ILLEGAL KMS RESPONSES
:RETURN

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 230
ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

```

7944
7945 066276 042737 000400 010230 COS0: BIC #INITC,FLAG ;INITIALIZATION COMPLETED (0,0)
7946 066304 000207 ;ASSERT INITIALIZATION COMPLETED
7947 ;RETURN
7948 066306 COS1: BIT #SETC,FLAG ;SET/RESET DDCMP MODE COMPLETED (0,1)
7949 066306 032737 001000 010230 BNE 10$ ;IS SET DDCMP EXPECTED ?
7950 066314 001016 ;YES.
7951
7952 ;UNEXPECTED RESTART DDCMP
7953 066316 105237 013636 INCB STARTC ;COUNT UNEXPECTED DDCMP RESTARTS
7954 066322 012737 022465 010170 MOV #DVEM11,TEMP2 ;SPECIFY ERROR MESSAGE
7955 066330 012737 000004 010172 MOV #4,TEMP3 ;SPECIFY ERROR CODE
7956 066336 004737 067146 JSR PC,LOGE1 ;LOG ERROR
7957 000012 ENUM = *D<ENUM+1> ;INCREMENT ERROR NUMBER
7958 066342 ERRSOF T ENUM,DVEM11,ERR14 ;REPORT ERROR
7959 066342 104457 TRAP C$ERSOF T
7960 066344 000012 .WORD 10
7961 066346 022465 .WORD DVEM11
7962 066350 023512 .WORD ERR14
7963 066352 042737 001000 010230 10$: BIC #SETC,FLAG ;ASSERT SET/RESET COMPLETED
7964 066360 000207 RTS PC ;RETURN
7965
7966 COS4: ;TRANSMIT COMPLETE
7967 066362 INC CTX ;INCREMENT TX COMPLETE COUNT
7968 066362 005237 007644 MOV TSPTR,R2 ;COPY POINTER
7969 066366 013702 007654 MOV SVSEL6,(R2)+ ;COPY SAVED SEL6
7970 066372 013722 067176 MOV SVSEL2,(R2)+ ;COPY SAVED SEL2
7971 066376 013722 067172 MOV R2,TSPTR ;SAVE UPDATED POINTER
7972 066402 010237 007654 RTS PC ;RETURN
7973 066406 000207
7974
7975 066410 C1S3: INCB ERRTHR ;ERROR THRESHOLD REACHED
7976 066410 105237 013640 RTS PC ;COUNT ERROR THRESHOLD MESSAGES
7977 066414 000207 ;RETURN
7978
7979

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 231
ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

```

7980 ;C1S4 TX TIMEOUT OCCURRED... TX CANNOT CONTINUE ON LINE
7981 ;POSSIBLE CAUSES : 1. LINE UNPLUGGED FROM MODEM
7982 ;2. XMIT BUF LENGTH = 0
7983 ;CONDITION WILL BE CLEARED BY:
7984 ;1. RECEPTION OF ANY MESSAGE (ACK, NAK, START, STACK, DATA)
7985 ;2. ISSUING DDCMP RESTART (INPUT 0,1) FOR THIS LINE
7986

```

```

7987 066416 C1S4: ;TX TIME OUT ON LINE
7988 066416 012737 022514 010170 MOV #DVEM14,TEMP2 ;SPECIFY TIME OUT ERROR MESSAGE
7989 066424 032737 001000 010230 BIT #SETC,FLAG ;IS PROTOCOL START REQUESTED ?
7990 066432 001406 BEQ 10$ ;NO
7991 066434 052737 004000 010230 BIS #TXTO,FLAG ;RECORD TRANSMIT TIMEOUT
7992 066442 012737 022563 010170 MOV #DVM14R,TEMP2 ;SPECIFY PROTOCOL START ERROR MESSAGE
7993 066450 013737 010170 013710 10$: MOV TEMP2,ERRMSG ;SPECIFY MESSAGE FOR ERROR REPORT
7994 066456 012737 000006 010172 MOV #6,TEMP3 ;SPECIFY ERROR CODE
7995 066464 013737 067174 010174 MOV SVSEL4,TEMP4 ;COPY SAVED SEL4
7996 066472 013737 067176 010176 MOV SVSEL6,TEMP5 ;COPY SAVED SEL6
7997 066500 004737 067146 JSR PC,LOGE1 ;LOG ERROR
7998 066504 012737 000003 013704 MOV #T.ESFT,ERRTYP ;SPECIFY ERROR TYPE
7999 000013 ENUM = ^D<ENUM+1> ;INCREMENT ERROR NUMBER
8000 066512 012737 000013 013706 MOV #ENUM,ERRNBR ;SPECIFY ERROR NUMBER
8001 066520 012737 023700 013712 MOV #ERR25,ERRBLK ;SPECIFY ERROR HANDLER
8002 066526 ERROR ;REPORT ERROR
8003 066526 104460 TRAP C$ERROR
8004 066530 000207 RTS PC ;RETURN

```


CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 232
ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

```

8005
8006 066532          C1S5:                               ;MSG NAKED (1,5)
8007
8008                                     ;GET NAK REASON MESSAGE
8009 066532 012737 022631 010170      MOV      #DVEM15,TEMP2      ;SPECIFY ERROR MESSAGE
8010 066540 012737 000005 010172      MOV      #5,TEMP3          ;SPECIFY ERROR CODE
8011 066546 005002                                     CLR      R2                ;CLEAR NAK REASON INDEX
8012 066550 123762 067176 013644 10$:  CMPB    SVSEL6,NAKX(R2)    ;IS THIS THE NAK REASON ?
8013 066556 001402                                     BEQ      20$               ;YES
8014 066560 005722                                     TST     (R2)+              ;IS THIS THE END OF THE NAK REASON LIST. ?
8015 066562 100372                                     BPL     10$                ;NO. TRY NEXT NAK REASON
8016 066564 016237 013664 010174 20$:  MOV      NAKMSG(R2),TEMP4   ;SPECIFY NAK REASON MESSAGE
8017 066572 004737 067146      JSR     PC,LOGE1           ;LOG ERROR
8018                                     ENUM = ^D<ENUM+1>         ;INCREMENT ERROR NUMBER
8019 066576                                     ERRSOFT ENUM,DVEM15,ERR24 ;REPORT ERROR
8020 066576 104457                                     TRAP    C$ERSOFT
8021 066600 000014                                     .WORD  12
8022 066602 022631                                     .WORD  DVEM15
8023 066604 023644                                     .WORD  ERR24
8024
8025 066606 013702 007654      MOV      TSPTR,R2          ;SPECIFY RETRANSMIT
8026 066612 013722 067176      MOV      SVSEL6,(R2)+      ;POINT TO TX SAVE AREA
8027 066616 013722 067172      MOV      SVSEL2,(R2)+      ;COPY SAVED SEL6
8028 066622 010237 007654      MOV      R2,TSPTR          ;COPY SAVED SEL2
8029 066626 005237 013642      INC     RXMIT              ;SAVE UPDATED POINTER
8030 066632 000207      RTS     PC                 ;SPECIFY RETRANSMIT
8031                                     ;RETURN
8032
8033 066634          C1S7:                               ;TRANSMIT REJECTED BEFORE COMPLETION
8034 066634 012737 022705 010170      MOV      #DVEM17,TEMP2    ;SPECIFY ERROR MESSAGE
8035 066642 012737 000002 010172      MOV      #2,TEMP3          ;SPECIFY ERROR CODE
8036 066650 013737 067176 010174      MOV      SVSEL6,TEMP4      ;COPY SAVED BDL AND MSG NUMBERS
8037 066656 000337 010174      SWAB    TEMP4              ;PUT BDL NUMBER IN BYTE0
8038 066662 004737 067146      JSR     PC,LOGE1           ;LOG ERROR
8039 066666 000015      ENUM = ^D<ENUM+1>         ;INCREMENT ERROR NUMBER
8040                                     ERRSOFT ENUM,DVEM17,ERR22 ;REPORT ERROR
8041 066666 104457                                     TRAP    C$ERSOFT
8042 066670 000015                                     .WORD  13
8043 066672 022705                                     .WORD  DVEM17
8044 066674 023542                                     .WORD  ERR22
8045 066676 000507      GOXOUT: BR      EXOUT      ; GOTO CLEANUP EXIT
8046

```

CZKMSAO KMS11-BD/BE DCLT
 CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 233
 ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

```

8047
8048 066700          C2S0:          ;MSG RCVD (2,0)
8049 066700 005237 007646 10$:      INC      CRX      ;INCREMENT RX COMPLETED COUNT
8050 066704 013702 007652      MOV      RSPTR,R2   ;POINT TO RX SAVE AREA ENTRY
8051 066710 013722 067176      MOV      SVSEL6,(R2)+ ;COPY SAVED SEL6 TO SAVE AREA
8052 066714 013722 067172      MOV      SVSEL2,(R2)+ ;COPY SAVED SEL2 TO SAVE AREA
8053 066720 022702 010046      CMP      #RXSKEN,R2  ;IS R2 POINTING TO THE END OF AREA ?
8054 066724 001002          BNE      20$        ;NO. KEEP POINTER
8055 066726 012702 007672      MOV      #RXSTAK,R2  ;YES. RESET POINTER
8056 066732 010237 007652 20$:      MOV      R2,RSPTR    ;SAVE POINTER
8057 066736 000207          RTS      PC         ;RETURN
8058
8059 066740          C2S1:          ;REQUEST BUFFER FOR MESSAGE RECEPTION (2,1)
8060 066740 005237 010116      INC      OPVAR      ;INCREMENT NO BUFF COUNTER
8061 066744 000207          RTS      PC         ;RETURN
8062
8063 066746          C3S0:          ;STATUS REPORT. LST BYTES 0..3
8064 066746 005037 011510      CLR      LINEST     ;CLEAR STATUS DATA LINE NUMBER
8065 066752 113737 067173 011510  MOVB     SVSEL2+1,LINEST ;COPY SAVED SEL2+1
8066 066760 013737 067174 011500  MOV      SVSEL4,DEVS1 ;COPY SAVED SEL4
8067 066766 013737 067176 011502  MOV      SVSEL6,DEVS2 ;COPY SAVED SEL6
8068 066774 042737 002000 010230  BIC      #STAT,FLAG  ;INDICATE STATUS REPORT OCCURRED
8069 067002 000207          RTS      PC         ;RETURN
8070
8071 067004          C3S1:          ;STATUS REPORT. LST BYTES 4..7
8072 067004 013737 067174 011504  MOV      SVSEL4,DEVS3 ;COPY SAVED SEL4
8073 067012 013737 067176 011506  MOV      SVSEL6,DEVS4 ;COPY SAVED SEL6
8074 067020 042737 002000 010230  BIC      #STAT,FLAG  ;INDICATE STATUS REPORT COMPLETED
8075 067026 000207          RTS      PC         ;RETURN
    
```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 234
ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

```

8076 067030
8077 067030 013737 067174 067200
8078 067036 013737 067176 067202
8079 067044 000207
8080
8081 067046
8082 067046 012737 022766 010170
8083 067054 004737 067126
8084 000016
8085 067060
8086 067060 104457
8087 067062 000016
8088 067064 022766
8089 067066 023460
8090 067070 000207
8091
8092 067072
8093 067072 012737 023030 010170
8094 067100 004737 067126
8095 000017
8096 067104
8097 067104 104457
8098 067106 000017
8099 067110 023030
8100 067112 023460
8101 067114 000207
8102
8103
8104

C3S2: ;KMC DATA MEMORY CONTENTS
      MOV SVSEL4,KADR ;SAVE DEVICE RAM ADDRESS
      MOV SVSEL6,KDATA ;SAVE DEVICE RAM CONTENTS (ONE BYTE)
      RTS PC

C3S3: ;KMS TIME OUT. WAITING FOR RDYI = 0
      MOV #DVEM33,TEMP2 ;POINT TO RDYI TIMEOUT MESSAGE
      JSR PC,LOGE0 ;LOG ERROR
      ENUM = ^D<ENUM+1> ;INCREMENT ERROR NUMBER
            ERRSOFT ENUM,DVEM33,ERR13 ;REPORT ERROR
      TRAP CSERSOFT
      .WORD 14
      .WORD DVEM33
      .WORD ERR13
      RTS PC ;RETURN

C3S4: ;KMS TIME OUT. WAITING FOR RDYO = 0
      MOV #DVEM34,TEMP2 ;SPECIFY ERROR MESSAGE
      JSR PC,LOGE0 ;LOG ERROR
      ENUM = ^D<ENUM+1> ;INCREMENT ERROR NUMBER
            ERRSOFT ENUM,DVEM34,ERR13 ;REPORT ERROR
      TRAP CSERSOFT
      .WORD 15
      .WORD DVEM34
      .WORD ERR13
      RTS PC ;RETURN

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 235
ERROR ROUTINES USED WITH KMS CSR OUTPUT ACTION ROUTINES

.SBTTL ERROR ROUTINES USED WITH KMS CSR OUTPUT ACTION ROUTINES

8105
8106
8107
8108 067116
8109 067116 004737 064020
8110 067122 000137 054512
8111
8112
8113 067126 005037 010172
8114 067132 013737 067170 010174
8115 067140 013737 067172 010176
8116 067146 113737 067173 010173
8117 067154 005237 010124
8118 067160 004737 024226
8119 067164 000207
8120
8121
8122
8123
8124 067166 000000
8125 067170 000000
8126 067172 000000
8127 067174 000000
8128 067176 000000
8129 067200 000000
8130 067202 000000
8131
8132

EXOUT: JSR PC,HLTTRB ; HALT LINE
JMP GTRA5 ; GOTO DCLT PROMPT

LOGE0: CLR TEMP3 ;LOG DEVICE ERROR
MOV SVSELO,TEMP4 ;SPECIFY ERROR CODE
MOV SVSEL2,TEMP5 ;COPY SAVED SEL0
LOGE1: MOV SVSEL2+1,TEMP3+1 ;COPY SAVED SEL2
INC ERRCNT ;COPY SAVED LINE NUMBER
JSR PC,LGDVE ;INCREMENT ERROR COUNT
RTS PC ;LOG DEVICE ERROR
;RETURN

;DATA SAVED FROM FIRMWARE TO HOST MESSAGES
;SAVED COMMAND, SUBCOMMAND PAIR
;SAVED COPY OF SEL0
;SAVED COPY OF SEL2
;SAVED COPY OF SEL4
;SAVED COPY OF SEL6
;SAVED DEVICE RAM ADDRESS
;SAVED DEVICE RAM DATA (ONE BYTE)

CMDSUB: .WORD 0
SVSELO: .WORD 0
SVSEL2: .WORD 0
SVSEL4: .WORD 0
SVSEL6: .WORD 0
KADR: .WORD 0
KDATA: .WORD 0

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 236
ERROR ROUTINES USED WITH KMS CSR OUTPUT ACTION ROUTINES

8133
8134
8135
8136
8137
8138
8139
8140
8141
8142
8143
8144
8145
8146
8147
8148
8149
8150
8151
8152
8153
8154

.SBTTL

DEVICE INTERRUPT SERVICE ROUTINES

067204
067204 052737 000001 010230
067212
067212 000002
067214
067214 052737 000002 010230
067222
067222 000002

BGNSRV DVINS
BIS #ININT,FLAG
ENDSRV
BGNSRV DVOUTS
BIS #OTINT,FLAG
ENDSRV

DVINS::
L10023: RTI
DVOUTS::
:OUTPUT INTERRUPT OCCURRED INDICATOR
L10024: RTI

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 237
DEVICE INTERRUPT SERVICE ROUTINES

8155
8156
8157
8158 067224
8159 067224
8160 067224 104401
8161

.EVEN
ENDTST

L10022: TRAP CSETST

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 238
DEVICE INTERRUPT SERVICE ROUTINES

8162
8163

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 239
HARDWARE PARAMETER CODING SECTION

.SBTTL HARDWARE PARAMETER CODING SECTION

..++
: THE HARDWARE PARAMETER CODING SECTION CONTAINS MACROS
: THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
: MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
: INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
: MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
: WITH THE OPERATOR.
:--

8164
8165
8166
8167
8168
8169
8170
8171
8172
8173
8174
8175
8176 067226
8177 067226 000023
8178 067230
8179
8180

BGNHRD

.WORD L10025-L\$HARD/2
L\$HARD::

.SBTTL DEVICE INDEPENDENT SECTION

8181
8182
8183 067230
8184 067230 000003
8185 067232 067433
8186
8187
8188
8189

DISPLAY DPLX

.WORD T\$CODE
.WORD DPLX

.SBTTL DEVICE DEPENDENT SECTION

8190
8191
8192 067234
8193 067234 000031
8194 067236 067276
8195 067240 160000
8196 067242 177776
8197 067244
8198 067244 001031
8199 067246 067324
8200 067250 000300
8201 067252 000776
8202 067254
8203 067254 002032
8204 067256 067357
8205 067260 000340
8206 067262 000004
8207 067264 000007
8208 067266
8209 067266 003031
8210 067270 067405
8211 067272 160000
8212 067274 177776

GPRMA CSRADR,0,0,160000,177776,YES

.WORD T\$CODE
.WORD CSRADR
.WORD T\$LOLIM
.WORD T\$HILIM

GPRMA VECTOR,2,0,300,776,YES

.WORD T\$CODE
.WORD VECTOR
.WORD T\$LOLIM
.WORD T\$HILIM

GPRMD PRIOR,4,0,340,4,7,YES

.WORD T\$CODE
.WORD PRIOR
.WORD 340
.WORD T\$LOLIM
.WORD T\$HILIM

GPRMA DM11BA,6,0,160000,177776,YES

.WORD T\$CODE
.WORD DM11BA
.WORD T\$LOLIM
.WORD T\$HILIM

ENDHWL: ENDHRD

.EVEN
L10025:

8213 067276
8214 067276
8215
8216 067276
8217
8218

.NLIST BEX

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 240
DEVICE DEPENDENT SECTION

:DEVICE INDEPENDENT QUESTIONS

:DEVICE DEPENDENT QUESTION

067276	046513	030503	025461	CSRADR:	.ASCIZ	/KMC11-B CSR ADDRESS: /
067324	047111	042524	051122	VECTOR:	.ASCIZ	/INTERRUPT VECTOR ADDRESS: /
067357	111	052116	051105	PRIOR:	.ASCIZ	/INTERRUPT PRIORITY : /
067405	104	030515	026461	DM11BA:	.ASCIZ	/DM11-BA CSR ADDRESS: /
067433	106	046125	020114	DPLX:	.ASCIZ	/FULL DUPLEX ONLY/
067454	000				.BYTE	0

.LIST BEX
.EVEN

8219
8220
8221

067456

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 241
DEVICE DEPENDENT SECTION

8222
8223
8224
8225
8226
8227
8228
8229
8230
8231
8232
8233
8234
8235
8236
8237
8238
8239
8240
8241
8242
8243
8244
8245
8246
8247
8248
8249
8250
8251
8252
8253
8254
8255
8256

067456
067456 000030

067536

067536 000000
067540 000000
067542
067542

000001

;.SBTTL SOFTWARE PARAMETER CODING SECTION

:+
: THE SOFTWARE PARAMETER CODING SECTION CONTAINS MACROS
: THAT ARE USED BY THE SUPERVISOR TO BUILD P-TABLES. THE
: MACROS ARE NOT EXECUTED AS MACHINE INSTRUCTIONS BUT ARE
: INTERPRETED BY THE SUPERVISOR AS DATA STRUCTURES. THE
: MACROS ALLOW THE SUPERVISOR TO ESTABLISH COMMUNICATIONS
: WITH THE OPERATOR.
:--

: BGNSFT

: ENDSFT

::: TEMPORARY PATCH AREA - FOR DEBUG PURPOSES :::

\$PATCH: .BLKW 30

LASTAD

L\$LAST:: ENDMOD

.END

.EVEN
.WORD 0
.WORD 0

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 243
CROSS REFERENCE TABLE -- USER SYMBOLS

ABO =	000026	2295#	5641						
ACT =	000003	2245#	5766	6220	6300				
ACTATV	057124	5989	6220#						
ACTBCR	056730	6007	6179#						
ACTCHK	057340	5969	6266#						
ACTCLB	056252	6068	6082#						
ACTCLP	057452	6003	6294#						
ACTCLR	055742	5967	6025#						
ACTCOP	056550	5977	6142#						
ACTCRC	057354	5998	6272#						
ACTCSE	056076	5972	6054#						
ACTCST	056170	5973	6070#						
ACTDLL	057172	5993	6234#						
ACTDME	056476	6009	6123	6126#					
ACTDMQ	056470	6010	6125#						
ACTDMS	056446	6008	6120#						
ACTDMX	056504	6127#							
ACTECH	057250	5997	6250#						
ACTEKE	060134	6399	6402#						
ACTEKT	060044	6019	6379#						
ACTEQO	056672	5981	6168#						
ACTETB	057550	6016	6314#						
ACTEXT	056026	6013	6043#						
ACTEXX	060172	6384	6396	6413#					
ACTHLP	055762	5971	6031#						
ACTKAL	057566	5772	6018	6319#	6400				
ACTKTB	057560	6017	6317#						
ACTLIS	057162	5992	6231#						
ACTLLP	057462	6004	6296#						
ACTLPX	057500	6291	6293	6295	6297	6300#			
ACTLXX	057542	6264	6285	6288	6301	6310#			
ACTMEX	057116	6161	6177	6199	6204	6210	6213	6217#	
ACTME1	057052	6188	6190	6192	6194	6196	6203#		
ACTMOP	057432	6001	6290#						
ACTMOS	057362	6012	6275#						
ACTMS0	056752	5982	6187#						
ACTMS1	056760	5983	6189#						
ACTMS2	056770	5984	6191#						
ACTMS3	057000	5985	6193#						
ACTMS4	057010	5986	6195#						
ACTMS5	057020	5987	6197#						
ACTMS6	057036	5988	6200#						
ACTM2X	057220	6221	6229	6232	6235	6238	6242#		
ACTNO	057240	5996	6247#						
ACTNUF	055732	6006	6022#						
ACTNUL	055740	5966	6023#						
ACTNUM	056560	5978	6145#						
ACTOPM	056652	5979	6163#						
ACTPAS	057134	5990	6223#						
ACTPRO	057370	5999	6278#						
ACTPRT	056036	6011	6045#						
ACTQFG	057374	6267	6270	6273	6276	6280#			
ACTRDE	027054	4126	4160	4163#					
ACTRDO	027046	4125	4162#						
ACTRDS	027024	4124	4157#						
ACTRDX	027062	4164#							

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 244
CROSS REFERENCE TABLE -- USER SYMBOLS

ACTREC	057154	5991	6228#						
ACTREX	026764	4120	4145#						
ACTRHL	026720	4119	4132#						
ACTRLG	026774	4121	4148#						
ACTRLP	057472	6005	6298#						
ACTRNF	026710	4123	4130#						
ACTRNL	026716	4118	4131#						
ACTRPS	057422	6000	6287#						
ACTRSL	027010	4122	4152#						
ACTRUN	056052	5970	6049#						
ACTSEX	057062	6014	6207#						
ACTSHO	055752	5968	6028#						
ACTSHW	056312	6060	6074	6093#	6112				
ACTSLS	057646	6015	6334#						
ACTSTE	056512	5974	6132#						
ACTSTS	057346	5980	6269#						
ACTSTT	056522	5975	6135#						
ACTSTX	056530	6133	6136#						
ACTSZE	056540	5976	6139#						
ACTTAL	057212	5995	6240#						
ACTTLP	057442	6002	6292#						
ACTTRA	057202	5994	6237#						
ADDCC	027422	4293#	5895	5939					
ADDC1	027516	4295	4310#						
ADR =	000020	G 2215#							
ADRINV	066266	7872	7916#						
ALCK	061050	2823	6619#						
ALCK1	061134	6676	6682#	6737					
ALCK2	061210	6683	6694#	6727					
ALCK2A	061470	6733	6739#						
ALCK3	061530	6697	6743	6749#					
ALCK3A	061652	6761	6773#						
ALCK3B	061674	6775	6778#						
ALCK3C	061664	6769	6776#	6784					
ALCK3D	061634	6764	6767#						
ALCK4	061716	6750	6782#						
ALCK4A	061730	6781	6785#						
ALCK5	061110	6675#	6777						
ALCK5B	061120	6678#							
ALL TR	061110	6551	6574	6598	6674#				
ASSEMB=	000010	1996							
ATVMOD=	000027	2351#	2983						
BAD	010205	2779#	3276	6869*					
BADCHR=	000051	2369#	3075						
BASM1	020051	3253#	4237						
BASM2	020042	3253#	4256						
BASM3	020033	3253#	4248						
BDCLK	016671	3253#	5452						
BDLBAS	013434	3166#	5774	7214	7606	7652			
BDLEND	013634	3183#	5776						
BDLMAD	013432	3165#	7418*	7490*					
BDLMNM	013430	3164#	7419*	7420*	7435	7491*	7494*	7504	
BIT0 =	000001	G 2188#	2271	2299	2426				
BIT00 =	000001	G 2177#	2188						
BIT01 =	000002	G 2176#	2187						
BIT02 =	000004	G 2175#	2186						

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 256
CROSS REFERENCE TABLE -- USER SYMBOLS

LNSTAT	063704	3772	5624	7309#			
LOE =	040000 G	2225#					
LOGAQR	031142	4518	4988#	6679			
LOGCMD	024346	3610#	6891				
LOGCML	024330	3606#	6858				
LOGCMP	024312	3602#	6847				
LOGDVI	024244	3593#	6499				
LOGEOP	024372	3616#	6919				
LOGEX	024642	3672#					
LOGEO	067126	7149	8083	8094	8113#		
LOGE1	067146	7956	7997	8017	8038	8116#	
LOGMSC	024410	3621#					
LOGRXC	024216	3586#	6701	7075			
LOGRXQ	024200	3581#	4995	7069			
LOGS1	024426	3574	3579	3584	3588	3626#	
LOGS2	024634	3666	3670#				
LOGS3	024434	3592	3600	3614	3619	3624	3628#
LOGS3A	024362	3605	3609	3613#			
LOGS4	024510	3636	3645#				
LOGS5	024534	3630	3632	3653#	5645		
LOGTXC	024162	3576#	6754	7024			
LOGTXQ	024144	3571#	6692	7016			
LOGUNT	010206	2784#	5467*	5469*	5470	5474	
LOOPS	003352	2672#	5028				
LOT =	000010 G	2214#					
LP0	016417	2672	2890	3253#	5027		
LP00	016420	3253#	5024				
LP1	016427	2673	3253#				
LP2	016440	2674	3253#				
LP3	016446	2675	3253#				
LP4	016461	2676	3253#				
LSACP	002110 G	2090#					
LSAPT	002036 G	2048#					
LSAU	054156 G	2075	5688#				
LSAUT	002070 G	2074#					
LSAUTO	054020 G	2091	5605#				
LSCCP	002106 G	2088#					
LSCLEA	054022 G	2089	5619#				
LSCO	002032 G	2044#					
LSDEPO	002011 G	2026#					
LSDESC	013730 G	2081	3238#				
LSDESP	002076 G	2080#					
LSDEVP	002060 G	2066#					
LSDISP	002124 G	2051	2108#				
LSDLY	002116 G	2096#					
LSDTP	002040 G	2050#					
LSDTYP	002034 G	2046#					
LSDU	054150 G	2077	5666#				
LSDUT	002072 G	2076#					
LSDVTY	013714 G	2067	3227#				
LSEF	002052 G	2061#					
LSENV1	002044 G	2054#					
LSERRT	013704 G	2085	3208#				
LSETP	002102 G	2084#					
LSEXP1	002046 G	2056#					
LSEXP4	002064 G	2070#					

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 259
CROSS REFERENCE TABLE -- USER SYMBOLS

NOCCLK	016715	3253#	5731
NODO	011520	2955#	
NOD1	011524	2956#	
NOD10	011602	2963#	
NOD100	012432	3031#	
NOD101	012436	3034#	
NOD102	012454	3035#	
NOD103	012460	3036#	
NOD104	012474	3037#	
NOD105	012500	3039#	
NOD106	012514	3040#	
NOD107	012520	3042#	
NOD11	011604	2964#	
NOD110	012534	3043#	
NOD111	012540	3045#	
NOD112	012554	3046#	
NOD113	012560	3048#	
NOD114	012564	3051#	
NOD115	012600	3052#	
NOD116	012604	3053#	
NOD117	012622	3054#	
NOD12	011616	2965#	
NOD120	012626	3055#	
NOD121	012642	3056#	
NOD122	012646	3057#	
NOD123	012662	3058#	
NOD124	012666	3059#	
NOD125	012702	3060#	
NOD126	012706	3061#	
NOD127	012722	3062#	
NOD13	011622	2966#	
NOD130	012726	3063#	
NOD131	012742	3064#	
NOD132	012746	3065#	
NOD133	012766	3066#	
NOD134	012772	3069#	
NOD135	012776	3070#	
NOD136	013002	3071#	
NOD137	013006	3072#	
NC914	011636	2967#	
NOD140	013012	3073#	
NOD141	013016	3074#	
NOD142	013022	3075#	
NOD143	013024	3078#	
NOD144	013030	3079#	
NOD145	013034	3080#	
NOD146	013050	3081#	
NOD147	013054	3082#	
NOD15	011642	2968#	
NOD150	013070	3083#	
NOD151	013074	3086#	
NOD152	013100	3087#	
NOD153	013104	3088#	
NOD154	013110	3090#	
NOD155	013114	3093#	
NOD156	013136	3094#	

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 260
CROSS REFERENCE TABLE -- USER SYMBOLS

NOD157	013142	3095#
NOD16	011656	2969#
NOD160	013156	3096#
NOD161	013162	3097#
NOD162	013204	3098#
NOD163	013210	3099#
NOD164	013232	3100#
NOD165	013236	3103#
NOD166	013242	3104#
NOD167	013246	3105#
NOD17	011662	2970#
NOD170	013252	3108#
NOD171	013256	3109#
NOD172	013272	3110#
NOD173	013274	3111#
NOD174	013314	3112#
NOD175	013320	3113#
NOD176	013334	3114#
NOD177	013336	3116#
NOD2	011530	2957#
NOD20	011676	2971#
NOD200	013342	3117#
NOD201	013354	3118#
NOD202	013360	3119#
NOD203	013364	3120#
NOD204	013370	3123#
NOD205	027070	4172#
NOD206	027074	4173#
NOD207	027100	4174#
NOD21	011702	2972#
NOD210	027102	4175#
NOD211	027116	4176#
NOD212	027120	4177#
NOD213	027134	4178#
NOD214	027136	4179#
NOD215	027162	4180#
NOD216	027164	4181#
NOD217	027176	4182#
NOD22	011706	2973#
NOD220	027200	4185#
NOD221	027214	4186#
NOD222	027220	4190#
NOD223	027224	4191#
NOD224	027230	4192#
NOD225	027234	4193#
NOD226	027240	4194#
NOD227	027244	4195#
NOD23	011720	2974#
NOD230	027250	4196#
NOD231	027254	4197#
NOD232	027260	4201#
NOD233	027262	4204#
NOD24	011724	2975#
NOD25	011736	2976#
NOD26	011742	2977#
NOD27	011744	2980#

CZKMSAO KMS11-BD/BE DCLT
 CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 264
 CROSS REFERENCE TABLE -- USER SYMBOLS

PNT	=	001000	G	2220#																	
PNTTRB		026616		3893	4063	4078	4101#														
PRI	=	002000	G	2221#																	
PRIOR		067357		8204	8218#																
PRI00	=	000000	G	2209#	5583																
PRI01	=	000040	G	2208#																	
PRI02	=	000100	G	2207#																	
PRI03	=	000140	G	2206#																	
PRI04	=	000200	G	2205#																	
PRI05	=	000240	G	2204#																	
PRI06	=	000300	G	2203#																	
PRI07	=	000340	G	2053	2202#	5634															
PRNT	=	000055		2373#	2960	6045															
PROTO	=	000041		2361#																	
PROTOB	=	000040		2276#	6278																
PRTEB		025764		3920	3921	3926#	3927	3928	3929	3930	3931	3932	3933								
PRTE0		026002		3927	3934#																
PRTE1		026034		3928	3946#																
PRTE2		026036		3929	3949#																
PRTE3		026074		3930	3963#																
PRTE4		026140		3931	3979#																
PRTE5		026170		3932	3989#																
PRTE6		026224		3933	4001#																
PRTNOS		016612		3253#	3777																
PSCNT		010122		2746#	5643	6495*	6914*	6917													
PST		016477		3253#	5042																
PTREND		007536		2702#																	
PTRTAB		006406		2697#	4452	4457	4641	5745	5878	6070	6077	6475									
PTR13		006502		2698#	5748	5921	6054	6064	6476												
PTR23		006576		2699#	4580	6477															
PSACT		003370		2681#	3707*	5135	5820*														
PSBUFA		003364		2679#	3705*	5093	5818*														
PSCNT		003372		2682#	5231*	5238	5300*	5307*	5312												
PSEXIT		031624		5096	5129#	5152	5157														
PSGDBD		003401		2686#	3690*	3710	5150*	5155*	5163	5213*	5256*	5806*	5823	6155*	6209*	6261*					
				6302*	6322*	6323	6336	6382	6447*	6449											
PSNNUF		003400		2685#	3691*	3720	4130*	4154*	4164*	5807*	5832	6022*	6050*	6058*	6072*	6127*					
				6217*	6224*	6243*	6310*	6319*	6334*	6381*											
PSNUM		003374		2683#	4157	4163	5226*	5229*	5231	5232*	5234*	5238*	5240*	5246*	6120	6126					
				6147	6157	6159	6287	6386													
PSRADX		003376		2684#	5186*	5189*	5196*	5203	5236	5244											
PSTREE		003366		2680#	3706*	5094	5819*														
PSTRV		031506		3709	5092#	5822															
PSTR5		031516		5095#	5104	5122	5127														
QCOPY	=	000013		2339#	3082	6142															
QRX	=	000004		2301#	4517	6547	6596	6625	6675	6696	6735	6744	6766	7065	7407	7410					
				7657																	
QTX	=	000010		2302#	6571	6625	6682	6736	6749	6762	6779	7017	7029	7478	7480	7611					
QUALFG		003250		2632#	3708*	5821*	6139*	6142*	6145	6247*	6250	6263*	6281	6284*							
QUALVL		003252		2633#	5902*	5946*	6136*	6159*	6160*												
RAMI	=	001000		2421#																	
RAMO	=	002000		2420#	5511																
RDMPPE	=	000010		2392#	4194																
RDMPQ	=	000007		2391#	4196																
RDMPSP	=	000006		2390#	3736	4159	4192														
RDYI	=	000020		2434#	7172	7193	7205	7216	7235	7246	7259	7318	7371	7380	7437	7505					

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 267
CRCSS REFERENCE TABLE -- USER SYMBOLS

STADD 010126
START 032706
STARTC 013636
STAT = 002000
STATB = 000001
STATUS= 000016
STATYP 013424
STEP = 000400
STXC 021664
STXQ 021653
SVCGBL= 000000

2748#	4157*	4233	6120*										
5383	5402#												
3187#	6486*	7953*											
2307#	7314	7322	7324	7326	8068	8074							
2271#	3631	5043	6269										
2342#	3034												
3148#													
2423#													
3253#	3577												
3253#	3572												
1996#	2015	2024	2026	2028	2030	2032	2034	2036	2038	2040	2042	2044	
2046	2048	2050	2052	2054	2056	2058	2061	2064	2066	2068	2070	2072	
2074	2076	2078	2080	2082	2084	2086	2088	2090	2092	2094	2096	2098	
2108	2120	2121	3208	3227	3238	3273	3290	3303	3320	3336	3352	3369	
3388	3403	3512	5333	5353	5369	5605	5619	5666	5688	8138	8146	8178	
8253#	8254												
1996#	2016	2017	2018	2019	2020	2021	2022	2023	2025	2027	2029	2031	
2033	2035	2037	2039	2041	2043	2045	2047	2049	2051	2053	2055	2057	
2059	2060	2062	2063	2065	2067	2069	2071	2073	2075	2077	2079	2081	
2083	2085	2087	2089	2091	2093	2095	2097	2099	2107	2109	2119	3228	
3230	3239	3246	3275	3276	3277	3278	3279	3280	3281	3282	3283	3284	
3287	3292	3293	3294	3295	3296	3297	3300	3305	3306	3307	3308	3309	
3310	3311	3314	3322	3323	3324	3325	3326	3327	3328	3331	3338	3339	
3340	3341	3342	3343	3344	3347	3354	3355	3356	3357	3358	3359	3360	
3361	3362	3365	3371	3372	3373	3374	3375	3376	3377	3378	3379	3380	
3381	3384	3390	3391	3392	3393	3394	3395	3396	3397	3400	3406	3407	
3408	3409	3410	3411	3412	3413	3414	3416	3417	3418	3419	3420	3421	
3422	3423	3424	3427	3430	3431	3538	3640	3641	3642	3643	3644	3648	
3649	3650	3651	3652	3685	3686	3687	3688	3689	3696	3697	3698	3699	
3700	3701	3702	3703	3713	3714	3715	3716	3717	3723	3724	3725	3726	
3727	3777	3778	3779	3780	3781	3785	3786	3787	3788	3789	3790	3792	
3793	3794	3795	3796	3797	3798	3799	3800	3802	3803	3804	3805	3806	
3807	3808	3809	3810	3812	3813	3814	3815	3816	3817	3818	3819	3820	
3822	3823	3824	3825	3826	3827	3828	3829	3830	3847	3848	3849	3850	
3851	3870	3871	3872	3873	3874	3880	3881	3882	3883	3884	3885	3886	
3887	3888	3895	3896	3897	3898	3899	3900	3901	3907	3908	3909	3910	
3911	3912	3936	3937	3938	3939	3940	3941	3942	3951	3952	3953	3954	
3955	3956	3957	3958	3959	3965	3966	3967	3968	3969	3970	3971	3972	
3973	3974	3975	3981	3982	3983	3984	3985	3986	3987	3991	3992	3993	
3994	3995	3996	3997	3998	4004	4005	4006	4007	4008	4009	4010	4011	
4012	4014	4015	4016	4017	4018	4019	4020	4021	4022	4048	4049	4050	
4051	4052	4053	4054	4055	4065	4066	4067	4068	4069	4070	4071	4072	
4080	4081	4082	4083	4084	4085	4086	4087	4103	4104	4105	4106	4107	
4108	4134	4135	4136	4137	4138	4139	4236	4237	4238	4239	4240	4241	
4246	4247	4248	4249	4250	4251	4252	4255	4256	4257	4258	4259	4260	
4300	4301	4302	4303	4304	5031	5032	5033	5034	5035	5036	5037	5038	
5039	5062	5063	5064	5065	5066	5067	5068	5069	5070	5208	5209	5210	
5211	5212	5251	5252	5253	5254	5255	5344	5376	5380	5381	5383	5385	
5386	5388	5390	5391	5393	5396	5397	5399	5404	5409	5410	5411	5413	
5419	5420	5421	5423	5432	5434	5439	5440	5441	5442	5443	5444	5445	
5446	5452	5453	5454	5455	5456	5474	5475	5476	5478	5522	5523	5524	
5525	5526	5528	5529	5530	5531	5532	5534	5535	5536	5537	5538	5539	
5540	5541	5543	5545	5556	5557	5558	5559	5560	5561	5566	5567	5568	
5569	5570	5571	5573	5574	5575	5576	5577	5578	5583	5584	5586	5587	
5594	5610	5632	5634	5635	5649	5650	5657	5670	5671	5678	5692	5693	
5700	5731	5732	5733	5734	5735	5780	5781	5782	5783	5784	5793	5795	

SVCINS= 000001

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 268
CROSS REFERENCE TABLE -- USER SYMBOLS

		5799	5800	5809	5810	5811	5812	5813	5814	5815	5816	5826	5827	5828
		5829	5830	5835	5836	5837	5838	5839	5855	5856	5868	5869	5870	5871
		5872	5873	5883	5884	5885	5886	5887	5888	5910	5911	5912	5913	5914
		5915	5927	5928	5929	5930	5931	5932	6033	6034	6035	6036	6037	6038
		6103	6104	6105	6106	6107	6108	6109	6150	6151	6152	6153	6154	6180
		6181	6182	6183	6184	6256	6257	6258	6259	6260	6305	6306	6307	6308
		6309	6343	6344	6345	6346	6347	6352	6353	6354	6355	6356	6361	6362
		6363	6364	6365	6366	6367	6372	6373	6374	6375	6376	6390	6391	6392
		6393	6394	6395	6431	6432	6433	6434	6435	6442	6443	6444	6445	6446
		6467	6468	6469	6470	6471	6854	6855	6856	6857	6873	6874	6875	6876
		6887	6888	6889	6890	6960	6961	6962	6963	6964	6996	6997	6998	6999
		7000	7001	7002	7003	7056	7057	7058	7059	7060	7079	7080	7081	7082
		7083	7143	7152	7153	7154	7155	7548	7549	7550	7551	7556	7594	7595
		7596	7597	7640	7641	7642	7643	7711	7712	7713	7714	7718	7801	7806
		7959	7960	7961	7962	8003	8020	8021	8022	8023	8041	8042	8043	8044
		8086	8087	8088	8089	8097	8098	8099	8100	8143	8151	8160	8177	8184
		8185	8193	8194	8195	8196	8198	8199	8200	8201	8203	8204	8205	8206
		8207	8209	8210	8211	8212	8215	8250	8251	8252				
SVCSUB=	000001	1996#												
SVCTAG=	000001	1996#	2145	3286	3299	3313	3330	3346	3364	3383	3399	3426	3537	3704
		5343	5447	5593	5609	5656	5677	5699	5817	7004	8142	8150	8159	8216
SVCTST=	000001	1996#	5715											
SVSELO	067170	7847*	8114	8125#										
SVSEL2	067172	7848*	7858	7971	8027	8052	8065	8115	8116	8126#				
SVSEL4	067174	7849*	7995	8066	8072	8077	8127#							
SVSEL6	067176	7850*	7970	7996	8012	8026	8036	8051	8067	8073	8078	8128#		
S&LSYM=	010000	1996#	2146#	3287#	3300#	3314#	3331#	3347#	3365#	3384#	3400#	3427#	3538#	3697
		3704	3705#	5344#	5440	5447	5448#	5594#	5610#	5657#	5678#	5700#	5810	5817
		5818#	6997	7004	7005#	8143#	8151#	8160#	8217#					
S1	032674	5393	5395#											
S2	032752	5413	5418#											
S3	033022	5423	5431#											
S4	033070	5434	5451#											
TABEX	016776	3253#	5883	5927										
TAL	= 000005	2248#	6240	7088										
TALCK	062350	2825	6988#	7026	7028									
TALMOD=	000035	2357#	2996											
TCURAD	010100	2736#	5752*	5755	5894	5900*	6080*							
TEMP	010164	2770#	3573*	3578*	3583*	3587*	3591*	3595*	3604*	3608*	3612*	3618*	3623*	7629
		3655	3656*	3657*	3658	4244*	4247	4306*	4307*	4308	4346*	4347*	4353	4582*
		4603*	4606*	4670	4754*	5021*	5034	5042*	5045*	5065	5641*	6101*	6103	6104
		6165*	6169	6172	6359*	6362	6460*	6503*	6507*	6723*	6724*	6727	7182*	7185*
		7192	7233*	7234	7516*	7517	7519*	7520*	7521	7579*	7580*	7581*	7589	7601
		7619*	7620*	7621*	7635	7647								
TEMP1	010166	2771#	3572*	3577*	3582*	3590*	3594*	3603*	3607*	3611*	3617*	3622*	3648	5028*
		5032	5046*	5049*	5064									
TEMP2	010170	2772#	3596*	3597*	3661	4349*	4351*	4355	4580*	4587	4670*	4755*	4757	4989*
		5029*	5031	5050*	5053*	5063	5518*	5536	5643*	6461*	6464	6514*	6517	6686*
		6698*	6707	6713	6720	6751*	6837*	6917*	7013*	7022*	7062*	7073*	7148*	7539*
		7586*	7632*	7703*	7781*	7786*	7789*	7790	7954*	7988*	7992*	7993	8009*	8034*
		8082*	8093*											
TEMP3	010172	2773#	3305	3339	3357	3376	3392	3598*	3662	4991*	5024*	5027*	5033	5519*
		5535	5644*	6688*	6699*	6708	6719	6725	6752*	6839*	6849	6918*	7014*	7023*
		7064*	7074*	7540*	7543*	7587*	7588*	7633*	7634*	7704*	7791*	7955*	7994*	8010*
		8035*	8113*	8116*										
TEMP4	010174	2774#	3292	3323	3355	3372	3374	3390	3407	3409	3599*	3663	5520*	5534

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 270
CROSS REFERENCE TABLE -- USER SYMBOLS

TXMTOT	010074	2734#	4464	4640	5758*	5877*	5879	5901*	6071	6075*	6428	6622		
TXONLY	060764	2821	6569#											
TXON2	060772	6570#												
TXPTR	010052	2725#	4457*	4459*	4460	4469*	4476	5745*	5756	5878*	5892*	5893	5898*	6077*
		6078	6475*											
TXQ	= 000000	2284#	3573											
TXSTAK	007656	2720#	6491											
TXTO	= 004000	2308#	7256	7264	7991									
T\$ARGC	= 000001	2016#	2017#	2018#	2019#	2020#	2021#	3275#	3284	3292#	3297	3305#	3311	3322#
		3328	3338#	3344	3354#	3362	3371#	3381	3390#	3397	3406#	3414	3416#	3424
		3640#	3644	3648#	3652	3685#	3689	3713#	3717	3723#	3727	3777#	3781	3785#
		3790	3792#	3800	3802#	3810	3812#	3820	3822#	3830	3847#	3851	3870#	3874
		3880#	3888	3895#	3901	3907#	3912	3936#	3942	3951#	3959	3965#	3975	3981#
		3987	3991#	3998	4004#	4012	4014#	4022	4048#	4055	4065#	4072	4080#	4087
		4103#	4108	4134#	4139	4236#	4241	4246#	4252	4255#	4260	4300#	4304	5031#
		5039	5062#	5070	5208#	5212	5251#	5255	5452#	5456	5522#	5526	5528#	5532
		5534#	5541	5731#	5735	5780#	5784	5826#	5830	5835#	5839	5868#	5873	5883#
		5888	5910#	5915	5927#	5932	6033#	6038	6103#	6109	6150#	6154	6180#	6184
		6256#	6260	6305#	6309	6343#	6347	6352#	6356	6361#	6367	6372#	6376	6390#
		6395	6431#	6435	6442#	6446	6467#	6471	6960#	6964	7056#	7060	7079#	7083
T\$CODE	= 003031	3699#	5442#	5812#	6999#	8184#	8193#	8198#	8203#	8209#	8209#	8209#	8209#	8209#
T\$ERRN	= 000017	1996#	6855#	6874#	6888#	7153#	7549#	7595#	7641#	7712#	7960#	8021#	8042#	8087#
		8098#												
T\$EXCP	= 000000	3699#	3704	5442#	5447	5812#	5817	6999#	7004	8193#	8197	8198#	8202	8203#
		8208	8209#	8213										
T\$FLAG	= 000040	3430#	3432	5586#	5649#	5670#	5672	5692#	5694	5799#	5855#			
T\$GMAN	= 000000	1996#	3696#	3699	3705#	5439#	5448#	5809#	5812	5818#	6996#	6999	7005#	
T\$HILI	= 177776	3699#	3703	5442#	5446	5812#	5816	6999#	7003	8193#	8196	8198#	8201	8203#
		8207	8209#	8212										
T\$LAST	= 000001	1996#	8251#											
T\$LOLI	= 160000	3699#	3702	5442#	5445	5812#	5815	6999#	7002	8193#	8195	8198#	8200	8203#
		8206	8209#	8211										
T\$LSYM	= 010000	1996#	2146	3287	3300	3314	3331	3347	3365	3384	3400	3427	3538	5344
		5594	5610	5657	5678	5700	8143	8151	8160	8217				
T\$LTNO	= 000001	8254#												
T\$NEST	= 177777	1996#	1998#	2119#	2145#	3273#	3286#	3290#	3299#	3303#	3313#	3320#	3330#	3336#
		3346#	3352#	3364#	3369#	3383#	3388#	3399#	3403#	3426#	3512#	3537#	5333#	5343#
		5353#	5360#	5369#	5593#	5605#	5609#	5619#	5656#	5666#	5677#	5688#	5699#	5716#
		8138#	8142#	8146#	8150#	8159#	8177#	8215#	8255#					
T\$NSO	= 000000	1998#	8255											
T\$NS1	= 000004	2119#	2145	3273#	3286	3290#	3299	3303#	3313	3320#	3330	3336#	3346	3352#
		3364	3369#	3383	3388#	3399	3403#	3426	3512#	3537	5333#	5343	5353#	5360
		5369#	5593	5605#	5609	5619#	5656	5666#	5677	5688#	5699	5716#	8159	8177#
		8215												
T\$NS2	= 000010	8138#	8142	8146#	8150									
T\$PTNU	= 000000	1996#												
T\$SAVL	= 177777	1996#												
T\$SEGL	= 177777	1996#												
T\$SUBN	= 000000	1996#	5715#											
T\$TAGL	= 177777	1996#												
T\$TAGN	= 010026	1996#	2119#	3273#	3290#	3303#	3320#	3336#	3352#	3369#	3388#	3403#	3512#	5333#
		5353#	5369#	5605#	5619#	5666#	5688#	5716#	8138#	8146#	8177#			
T\$TEMP	= 000000	2109#	2110#	2145#	3286#	3299#	3313#	3330#	3346#	3364#	3383#	3399#	3426#	3430#
		3431	3537#	3699#	5343#	5360#	5442#	5586#	5587	5593#	5609#	5649#	5650	5656#
		5670#	5671	5677#	5692#	5693	5699#	5799#	5800	5812#	5855#	5856	6999#	8142#
		8150#	8159#	8193#	8198#	8203#	8209#	8215#	8255#					

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 271
CROSS REFERENCE TABLE -- USER SYMBOLS

T\$TEST= 000001
T\$TSTM= 177777

1996#	5715#	8254															
1996#	3283	3287	3296	3300	3310	3314	3327	3331	3343	3347	3361	3365					
3380	3384	3396	3400	3413	3423	3427	3643	3651	3688	3696	3716	3726					
3780	3789	3799	3809	3819	3829	3850	3873	3887	3900	3911	3941	3958					
3974	3986	3997	4011	4021	4054	4071	4086	4107	4138	4240	4251	4259					
4303	5038	5069	5211	5254	5344	5376	5381	5386	5391	5397	5404	5410					
5420	5432	5439	5455	5475	5525	5531	5540	5543	5545	5560	5570	5577					
5584	5586	5594	5610	5632	5635	5649	5657	5678	5700	5734	5783	5793					
5799	5809	5829	5838	5855	5872	5887	5914	5931	6037	6108	6153	6183					
6259	6308	6346	6355	6366	6375	6394	6434	6445	6470	6854	6873	6887					
6963	6996	7059	7082	7143	7152	7548	7556	7594	7640	7711	7718	7801					
7806	7959	8003	8020	8041	8086	8097	8160										

T\$TSTS= 000001
T\$SAU = 010021
T\$SAUT= 010016
T\$SCLE= 010017
T\$SDU = 010020
T\$SHAR= 010025
T\$SHW = 010000
T\$SINI= 010015
T\$MSG= 010011

1996#	5716#																
5688#	5692	5699															
5605#	5609																
5619#	5649	5656															
5666#	5670	5677															
8177#	8216																
2119#	2145																
5369#	5586	5593															
3273#	3286	3290#	3299	3303#	3313	3320#	3330	3336#	3346	3352#	3364	3369#					
3383	3388#	3399	3403#	3426	3430												
5353#																	
5333#	5343																
3512#	3537	8138#	8142	8146#	8150												
5716#	5799	5855	8159														
2448#																	
2449#																	
2447#																	
2450#	7798	7998															
2109	5715#																
2218#																	

T\$SPRO= 010014
T\$SRPT= 010013
T\$SRV= 010024
T\$STES= 010022
T.EDF = 000001
T.EHRD= 000002
T.ESF = 000000
T.ESFT= 000003
T1 054164 G
UAM = 000200 G
ULRCLS 030552
ULRPLS 030454
ULTCLS 030652
ULTPLS 030612
UPTABL 061252
UPTA1 061336
UPTA4 061302
UPTEX 061436
VALTRB 003404
VECTOR 067324
VSTAT 011514
WRFLG 003402
X\$ = 000234

5353#																	
5333#	5343																
3512#	3537	8138#	8142	8146#	8150												
5716#	5799	5855	8159														
2448#																	
2449#																	
2447#																	
2450#	7798	7998															
2109	5715#																
2218#																	
4793#	6739																
4516	4732#	4896	6678	6717													
4869#	4946	6756															
4830#	4949	6705															
6703#																	
6710	6717#																
6704	6709#																
6715	6731#																
2689#	6380*	6409*															
8199	8218#																
2933#	3773	7158*	7328*														
2687#																	

1999#	2955#	2956#	2957#	2958#	2959#	2960#	2961#	2962#	2963#	2964#	2965#	2966#					
2967#	2968#	2969#	2970#	2971#	2972#	2973#	2974#	2975#	2976#	2977#	2980#	2981#					
2982#	2983#	2984#	2985#	2986#	2987#	2988#	2989#	2990#	2991#	2992#	2993#	2994#					
2995#	2996#	2997#	3001#	3002#	3003#	3004#	3005#	3010#	3011#	3012#	3013#	3014#					
3017#	3018#	3019#	3020#	3021#	3022#	3023#	3024#	3026#	3027#	3028#	3029#	3030#					
3031#	3034#	3035#	3036#	3037#	3039#	3040#	3042#	3043#	3045#	3046#	3048#	3051#					
3052#	3053#	3054#	3055#	3056#	3057#	3058#	3059#	3060#	3061#	3062#	3063#	3064#					
3065#	3066#	3069#	3070#	3071#	3072#	3073#	3074#	3075#	3078#	3079#	3080#	3081#					
3082#	3083#	3086#	3087#	3088#	3090#	3093#	3094#	3095#	3096#	3097#	3098#	3099#					
3100#	3103#	3104#	3105#	3108#	3109#	3110#	3111#	3112#	3113#	3114#	3116#	3117#					
3118#	3119#	3120#	3123#	4172#	4173#	4174#	4175#	4176#	4177#	4178#	4179#	4180#					
4181#	4182#	4185#	4186#	4190#	4191#	4192#	4193#	4194#	4195#	4196#	4197#	4201#					

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 275
CROSS REFERENCE TABLE -- MACRO NAMES

ENDINI	1#	1996#	5592												
ENDMOD	1#	1996#	8254												
ENDMSG	1#	1996#	3285	3298	3312	3329	3345	3363	3382	3398	3425				
ENDPRO	1#	1996#	5359												
ENDPTA	1#	1996#													
ENDRPT	1#	1996#	5342												
ENDSEG	1#	1996#													
ENDSET	1#	1996#													
ENDSFT	1#	1996#													
ENDSRV	1#	1996#	3536	8141	8149										
ENDSUB	1#	1996#													
ENDSW	1#	1996#													
ENDTST	1#	1996#	8158												
EQUALS	1#	1996#	2158												
ERRDF	1#	1996#													
ERRHRD	1#	1996#													
ERROR	1#	1996#	7800	8002											
ERRSF	1#	1996#													
ERRSOF	1#	1996#	6853	6872	6886	7151	7547	7593	7639	7710	7958	8019	8040	8085	8096
ERRTBL	1#	1996#	3207												
ESCAPE	1#	1996#													
EXIT	1#	1996#	3429	5585	5648	5669	5691	5798	5854						
FEQUAL	1#	1996#													
GETBYT	1#	1996#													
GETPRI	1#	1996#													
GETWOR	1#	1996#													
GMANIA	1#	1996#													
GMANID	1#	1996#	3695	5438	5808	6995									
GMANIL	1#	1996#													
GPHARD	1#	1996#	5473												
GPRMA	1#	1996#	8192	8197	8208										
GPRMD	1#	1996#	3696#	3699	5439#	5442	5809#	5812	6996#	6999	8202				
GPRML	1#	1996#													
HEADER	1#	1996#	2014												
INLOOP	1#	1996#													
IOSE TU	1#	1996#													
IOSTAR	1#	1996#													
KT11	1#	1996#													
LASTAD	1#	1996#	8249												
MANUAL	1#	1996#	5792												
MEMORY	1#	1996#													
MSBYTE	1#	1996#	2015#	2021	2022	2023									
MSCHEC	1#	1996#	3430#	5586#	5649#	5670#	5692#	5799#	5855#						
MSCNTO	1#	1996#	3699#	5442#	5812#	6999#	8193#	8198#	8203#	8209#					
MSCOUN	1#	1996#	3275#	3292#	3305#	3322#	3338#	3354#	3371#	3390#	3406#	3416#	3640#	3648#	3685#
	3713#	3723#	3777#	3785#	3792#	3802#	3812#	3822#	3847#	3870#	3880#	3895#	3907#	3936#	3951#
	3965#	3981#	3991#	4004#	4014#	4048#	4065#	4080#	4103#	4134#	4236#	4246#	4255#	4300#	5031#
	5062#	5208#	5251#	5452#	5522#	5528#	5534#	5731#	5780#	5826#	5835#	5868#	5883#	5910#	5927#
	6033#	6103#	6150#	6180#	6256#	6305#	6343#	6352#	6361#	6372#	6390#	6431#	6442#	6467#	6960#
	7056#	7079#													
MSDATA	1#	1996#	2015#	2024	2026	2028	2030	2032	2034	2036	2038	2040	2042	2044	2046
	2048	2050	2052	2054#	2056	2058	2061	2064	2066	2068	2070	2072	2074	2076	2078
	2080	2082	2084	2086	2088	2090	2092	2094	2096	2098	3227#	3238#			
MSDECR	1#	1996#	2145#	3286#	3299#	3313#	3330#	3346#	3364#	3383#	3399#	3426#	3537#	5343#	5360#
	5593#	5609#	5656#	5677#	5699#	8142#	8150#	8159#	8215#	8255#					
MSDEFA	1#	1996#	3699#	5442#	5812#	6999#	8193#	8198#	8203#	8209#					

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 276
CROSS REFERENCE TABLE -- MACRO NAMES

MSENDE	1#	1996#	2145#	3286#	3299#	3313#	3330#	3346#	3364#	3383#	3399#	3426#	3537#	5343#	5593#
	5609#	5656#	5677#	5699#	8142#	8150#	8159#	8215#	8255#						
MSEERRI	1#	1996#	6854#	6873#	6887#	7152#	7548#	7594#	7640#	7711#	7959#	8020#	8041#	8086#	8097#
MSESCA	1#	1996#													
MSESCS	1#	1996#													
MSEXCP	1#	1996#	3699#	5442#	5812#	6999#	8193#	8198#	8203#	8209#					
MSEXIT	1#	1996#	3430#	5586#	5587	5649#	5650	5670#	5692#	5799#	5800	5855#	5856		
MSEXSE	1#	1996#	3430#	5586#	5649#	5670#	5692#	5799#	5855#						
MSEXTJ	1#	1996#	3430#	3431	5586#	5649#	5670#	5671	5692#	5693	5799#	5855#			
M\$GEN	1#	1996#	2015#	2024#	2026#	2028#	2030#	2032#	2034#	2036#	2038#	2040#	2042#	2044#	2046#
	2048#	2050#	2052#	2054#	2056#	2058#	2061#	2064#	2066#	2068#	2070#	2072#	2074#	2076#	2078#
	2080#	2082#	2084#	2086#	2088#	2090#	2092#	2094#	2096#	2098#	2108#	2120#	2121#	2145#	3208#
	3227#	3238#	3273#	3286#	3290#	3299#	3303#	3313#	3320#	3330#	3336#	3346#	3352#	3364#	3369#
	3383#	3388#	3399#	3403#	3426#	3512#	3537#	3704#	5333#	5343#	5353#	5369#	5447#	5593#	5605#
	5609#	5619#	5656#	5666#	5677#	5688#	5699#	5715#	5817#	7004#	8138#	8142#	8146#	8150#	8159#
	8178#	8216#	8253#												
M\$GENB	1#	1996#	3696#	3697	5439#	5440	5809#	5810	6996#	6997					
M\$GETS	1#	1996#	2145#	3286#	3299#	3313#	3330#	3346#	3364#	3383#	3399#	3426#	3537#	5343#	5360#
	5593#	5609#	5656#	5677#	5699#	8142#	8150#	8159#	8215#	8255#					
M\$GETT	1#	1996#	3430#	5586#	5649#	5670#	5692#	5799#	5855#						
M\$GNGB	1#	1996#	1998#	2015#	2024#	2026#	2028#	2030#	2032#	2034#	2036#	2038#	2040#	2042#	2044#
	2046#	2048#	2050#	2052#	2054#	2056#	2058#	2061#	2064#	2066#	2068#	2070#	2072#	2074#	2076#
	2078#	2080#	2082#	2084#	2086#	2088#	2090#	2092#	2094#	2096#	2098#	2107#	2108	2119#	2120
	2121	3208#	3227#	3238#	3273#	3290#	3303#	3320#	3336#	3352#	3369#	3388#	3403#	3512#	5333#
	5353#	5369#	5605#	5619#	5666#	5688#	8138#	8146#	8177#	8178	8250#	8253			
M\$GNIN	1#	1996#	2015#	2016	2017	2018	2019	2020	2021#	2022#	2023#	2024#	2025	2026#	2027
	2028#	2029	2030#	2031	2032#	2033	2034#	2035	2036#	2037	2038#	2039	2040#	2041	2042#
	2043	2044#	2045	2046#	2047	2048#	2049	2050#	2051	2052#	2053	2054#	2055	2056#	2057
	2058#	2059	2060	2061#	2062	2063#	2064#	2065	2066#	2067	2068#	2069	2070#	2071	2072#
	2073	2074#	2075	2076#	2077	2078#	2079	2080#	2081	2082#	2083	2084#	2085	2086#	2087
	2088#	2089	2090#	2091	2092#	2093	2094#	2095	2096#	2097	2098#	2099	2107#	2109#	2119#
	3227#	3228	3230	3238#	3239	3246	3275#	3276	3277#	3278	3279#	3280#	3281#	3282	3283#
	3284	3287#	3292#	3293#	3294#	3295	3296#	3297	3300#	3305#	3306#	3307#	3308#	3309	3310#
	3311	3314#	3322#	3323#	3324#	3325#	3326	3327#	3328	3331#	3338#	3339	3340#	3341#	3342
	3343#	3344	3347#	3354#	3355	3356#	3357	3358#	3359#	3360	3361#	3362	3365#	3371#	3372
	3373#	3374	3375#	3376	3377#	3378#	3379	3380#	3381	3384#	3390#	3391#	3392	3393#	3394#
	3395	3396#	3397	3400#	3406#	3407	3408#	3409	3410#	3411#	3412	3413#	3414	3416#	3417
	3418#	3419	3420#	3421#	3422	3423#	3424	3427#	3430#	3431#	3537#	3538	3640#	3641#	3642
	3643#	3644	3648#	3649#	3650	3651#	3652	3685#	3686#	3687	3688#	3689	3696#	3697#	3698#
	3699#	3700	3701	3702	3703	3713#	3714#	3715	3716#	3717	3723#	3724#	3725	3726#	3727
	3777#	3778#	3779	3780#	3781	3785#	3786#	3787#	3788	3789#	3790	3792#	3793	3794#	3795
	3796#	3797#	3798	3799#	3800	3802#	3803	3804#	3805	3806#	3807#	3808	3809#	3810	3812#
	3813	3814#	3815	3816#	3817#	3818	3819#	3820	3822#	3823	3824#	3825	3826#	3827#	3828
	3829#	3830	3847#	3848#	3849	3850#	3851	3870#	3871#	3872	3873#	3874	3880#	3881#	3882#
	3883#	3884#	3885#	3886	3887#	3888	3895#	3896#	3897#	3898#	3899	3900#	3901	3907#	3908#
	3909#	3910	3911#	3912	3936#	3937#	3938#	3939#	3940	3941#	3942	3951#	3952	3953#	3954
	3955#	3956#	3957	3958#	3959	3965#	3966	3967#	3968	3969#	3970	3971#	3972#	3973	3974#
	3975	3981#	3982	3983#	3984#	3985	3986#	3987	3991#	3992#	3993	3994#	3995#	3996	3997#
	3998	4004#	4005	4006#	4007	4008#	4009#	4010	4011#	4012	4014#	4015	4016#	4017	4018#
	4019#	4020	4021#	4022	4048#	4049#	4050#	4051#	4052#	4053	4054#	4055	4065#	4066#	4067#
	4068#	4069#	4070	4071#	4072	4080#	4081#	4082#	4083#	4084#	4085	4086#	4087	4103#	4104#
	4105#	4106	4107#	4108	4134#	4135#	4136#	4137	4138#	4139	4236#	4237#	4238#	4239	4240#
	4241	4246#	4247	4248#	4249#	4250	4251#	4252	4255#	4256#	4257#	4258	4259#	4260	4300#
	4301#	4302	4303#	4304	5031#	5032#	5033#	5034#	5035#	5036#	5037	5038#	5039	5062#	5063#
	5064#	5065#	5066#	5067#	5068	5069#	5070	5208#	5209#	5210	5211#	5212	5251#	5252#	5253
	5254#	5255	5344#	5376#	5380#	5381#	5383#	5385#	5386#	5388#	5390#	5391#	5393#	5396#	5397#

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 277
CROSS REFERENCE TABLE -- MACRO NAMES

	5399#	5404#	5409#	5410#	5411#	5413#	5419#	5420#	5421#	5423#	5432#	5434#	5439#	5440#	5441#
	5442#	5443	5444	5445	5446	5452#	5453#	5454	5455#	5456	5474#	5475#	5476#	5478#	5522#
	5523#	5524	5525#	5526	5528#	5529#	5530	5531#	5532	5534#	5535#	5536#	5537#	5538#	5539
	5540#	5541	5543#	5545#	5556#	5557#	5558#	5559#	5560#	5561	5566#	5567#	5568#	5569#	5570#
	5571	5573#	5574#	5575#	5576#	5577#	5578	5583#	5584#	5586#	5587#	5594#	5610#	5632#	5634#
	5635#	5649#	5650#	5657#	5670#	5671#	5678#	5692#	5693#	5700#	5731#	5732#	5733	5734#	5735
	5780#	5781#	5782	5783#	5784	5793#	5795#	5799#	5800#	5809#	5810#	5811#	5812#	5813	5814
	5815	5816	5826#	5827#	5828	5829#	5830	5835#	5836#	5837	5838#	5839	5855#	5856#	5868#
	5869#	5870#	5871	5872#	5873	5883#	5884#	5885#	5886	5887#	5888	5910#	5911#	5912#	5913
	5914#	5915	5927#	5928#	5929#	5930	5931#	5932	6033#	6034#	6035#	6036	6037#	6038	6103#
	6104#	6105#	6106#	6107	6108#	6109	6150#	6151#	6152	6153#	6154	6180#	6181#	6182	6183#
	6184	6256#	6257#	6258	6259#	6260	6305#	6306#	6307	6308#	6309	6343#	6344#	6345	6346#
	6347	6352#	6353#	6354	6355#	6356	6361#	6362	6363#	6364#	6365	6366#	6367	6372#	6373#
	6374	6375#	6376	6390#	6391#	6392#	6393	6394#	6395	6431#	6432#	6433	6434#	6435	6442#
	6443#	6444	6445#	6446	6467#	6468#	6469	6470#	6471	6854#	6855#	6856#	6857#	6873#	6874#
	6875#	6876#	6887#	6888#	6889#	6890#	6960#	6961#	6962	6963#	6964	6996#	6997#	6998#	6999#
	7000	7001	7002	7003	7056#	7057#	7058	7059#	7060	7079#	7080#	7081	7082#	7083	7143#
	7152#	7153#	7154#	7155#	7548#	7549#	7550#	7551#	7556#	7594#	7595#	7596#	7597#	7640#	7641#
	7642#	7643#	7711#	7712#	7713#	7714#	7718#	7801#	7806#	7959#	7960#	7961#	7962#	8003#	8020#
	8021#	8022#	8023#	8041#	8042#	8043#	8044#	8086#	8087#	8088#	8089#	8097#	8098#	8099#	8100#
	8142#	8143	8150#	8151	8160#	8177#	8184#	8185#	8193#	8194	8195	8196	8198#	8199	8200
	8201	8203#	8204	8205	8206	8207	8209#	8210	8211	8212	8215#	8250#	8251#	8252#	
MSGNLS	1#	1996#	3696#	3704	5439#	5447	5809#	5817	6996#	7004					
MSGNSU	1#	1996#													
MSGNTA	1#	1996#	2145#	3286#	3299#	3313#	3330#	3346#	3364#	3383#	3399#	3426#	3537#	5343#	5593#
	5609#	5656#	5677#	5699#	8142#	8150#	8159#	8215#	8216						
MSGNTE	1#	1996#	5715#												
M\$HAPT	1#	1996#	2015#												
M\$HNAP	1#	1996#	2015#	2054											
M\$INCR	1#	1996#	1998#	2119#	3273#	3283#	3287#	3290#	3296#	3300#	3303#	3310#	3314#	3320#	3327#
	3331#	3336#	3343#	3347#	3352#	3361#	3365#	3369#	3380#	3384#	3388#	3396#	3400#	3403#	3413#
	3423#	3427#	3512#	3643#	3651#	3688#	3696#	3705	3716#	3726#	3780#	3789#	3799#	3809#	3819#
	3829#	3850#	3873#	3887#	3900#	3911#	3941#	3958#	3974#	3986#	3997#	4011#	4021#	4054#	4071#
	4086#	4107#	4138#	4240#	4251#	4259#	4303#	5038#	5069#	5211#	5254#	5333#	5344#	5353#	5369#
	5376#	5381#	5386#	5391#	5397#	5404#	5410#	5420#	5432#	5439#	5448	5455#	5475#	5525#	5531#
	5540#	5543#	5545#	5560#	5570#	5577#	5584#	5586#	5594#	5605#	5610#	5619#	5632#	5635#	5649#
	5657#	5666#	5678#	5688#	5700#	5715#	5716#	5734#	5783#	5793#	5799#	5809#	5818	5829#	5838#
	5855#	5872#	5887#	5914#	5931#	6037#	6108#	6153#	6183#	6259#	6308#	6346#	6355#	6366#	6375#
	6394#	6434#	6445#	6470#	6854#	6873#	6887#	6963#	6996#	7005	7059#	7082#	7143#	7152#	7548#
	7556#	7594#	7640#	7711#	7718#	7801#	7806#	7959#	8003#	8020#	8041#	8086#	8097#	8138#	8146#
	8160#	8177#													
M\$IOSE	1#	1996#													
M\$LDRO	1#	1996#	5380#	5385#	5390#	5396#	5409#	5419#	5474#	5583#	5634#				
M\$MASK	1#	1996#													
M\$MCHI	1#	1996#													
M\$MCLO	1#	1996#													
M\$MSK1	1#	1996#													
M\$POP	1#	1996#	2145#	3286#	3299#	3313#	3330#	3346#	3364#	3383#	3399#	3426#	3537#	5343#	5360#
	5593#	5609#	5656#	5677#	5699#	8142#	8150#	8159#	8215#	8255#					
M\$PRIN	1#	1996#	3275#	3292#	3305#	3322#	3338#	3354#	3371#	3390#	3406#	3416#	3640#	3648#	3685#
	3713#	3723#	3777#	3785#	3792#	3802#	3812#	3822#	3847#	3870#	3880#	3895#	3907#	3936#	3951#
	3965#	3981#	3991#	4004#	4014#	4048#	4065#	4080#	4103#	4134#	4236#	4246#	4255#	4300#	5031#
	5062#	5208#	5251#	5452#	5522#	5528#	5534#	5731#	5780#	5826#	5835#	5868#	5883#	5910#	5927#
	6033#	6103#	6150#	6180#	6256#	6305#	6343#	6352#	6361#	6372#	6390#	6431#	6442#	6467#	6960#
	7056#	7079#													
M\$PUSH	1#	1996#	1998#	2119#	3273#	3290#	3303#	3320#	3336#	3352#	3369#	3388#	3403#	3512#	5333#

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 278
CROSS REFERENCE TABLE -- MACRO NAMES

M\$PUT	5353#	5369#	5605#	5619#	5666#	5688#	5715#	5716	8138#	8146#	8177#				
	1#	1996#	3275#	3292#	3305#	3322#	3338#	3354#	3371#	3390#	3406#	3416#	3640#	3648#	3685#
	3713#	3723#	3777#	3785#	3792#	3802#	3812#	3822#	3847#	3870#	3880#	3895#	3907#	3936#	3951#
	3965#	3981#	3991#	4004#	4014#	4048#	4065#	4080#	4103#	4134#	4236#	4246#	4255#	4300#	5031#
	5062#	5208#	5251#	5452#	5522#	5528#	5534#	5556#	5566#	5573#	5731#	5780#	5826#	5835#	5868#
	5883#	5910#	5927#	6033#	6103#	6150#	6180#	6256#	6305#	6343#	6352#	6361#	6372#	6390#	6431#
	6442#	6467#	6960#	7056#	7079#										
M\$PUT1	1#	1996#	3275#	3277	3279	3280	3281	3292#	3293	3294	3305#	3306	3307	3308	3322#
	3323	3324	3325	3338#	3340	3341	3354#	3356	3358	3359	3371#	3373	3375	3377	3378
	3390#	3391	3393	3394	3406#	3408	3410	3411	3416#	3418	3420	3421	3640#	3641	3648#
	3649	3685#	3686	3713#	3714	3723#	3724	3777#	3778	3785#	3786	3787	3792#	3794	3796
	3797	3802#	3804	3806	3807	3812#	3814	3816	3817	3822#	3824	3826	3827	3847#	3848
	3870#	3871	3880#	3881	3882	3883	3884	3885	3895#	3896	3897	3898	3907#	3908	3909
	3936#	3937	3938	3939	3951#	3953	3955	3956	3965#	3967	3969	3971	3972	3981#	3983
	3984	3991#	3992	3994	3995	4004#	4006	4008	4009	4014#	4016	4018	4019	4048#	4049
	4050	4051	4052	4065#	4066	4067	4068	4069	4080#	4081	4082	4083	4084	4103#	4104
	4105	4134#	4135	4136	4236#	4237	4238	4246#	4248	4249	4255#	4256	4257	4300#	4301
	5031#	5032	5033	5034	5035	5036	5062#	5063	5064	5065	5066	5067	5208#	5209	5251#
	5252	5452#	5453	5522#	5523	5528#	5529	5534#	5535	5536	5537	5538	5556#	5557	5558
	5559	5566#	5567	5568	5569	5573#	5574	5575	5576	5731#	5732	5780#	5781	5826#	5827
	5835#	5836	5868#	5869	5870	5883#	5884	5885	5910#	5911	5912	5927#	5928	5929	6033#
	6034	6035	6103#	6104	6105	6106	6150#	6151	6180#	6181	6256#	6257	6305#	6306	6343#
	6344	6352#	6353	6361#	6363	6364	6372#	6373	6390#	6391	6392	6431#	6432	6442#	6443
	6467#	6468	6960#	6961	7056#	7057	7079#	7080							
M\$RADI	1#	1996#	3699#	5442#	5812#	6999#	8193#	8198#	8203#	8209#					
M\$RBRO	1#	1996#													
M\$RNRO	1#	1996#	5409#	5411	5419#	5421	5474#	5476							
M\$SETS	1#	1996#	1998#	2119#	3273#	3290#	3303#	3320#	3336#	3352#	3369#	3388#	3403#	3512#	5333#
M\$STAR	5353#	5369#	5605#	5619#	5666#	5688#	5716#	8138#	8146#	8177#					
M\$SVC	1#	1996#													
	3327	3330#	3331	3338#	3343	3346#	3347	3354#	3361	3364#	3365	3371#	3380	3383#	3384
	3390#	3396	3399#	3400	3406#	3413	3416#	3423	3426#	3427	3430#	3640#	3643	3648#	3651
	3685#	3688	3696#	3713#	3716	3723#	3726	3777#	3780	3785#	3789	3792#	3799	3802#	3809
	3812#	3819	3822#	3829	3847#	3850	3870#	3873	3880#	3887	3895#	3900	3907#	3911	3936#
	3941	3951#	3958	3965#	3974	3981#	3986	3991#	3997	4004#	4011	4014#	4021	4048#	4054
	4065#	4071	4080#	4086	4103#	4107	4134#	4138	4236#	4240	4246#	4251	4255#	4259	4300#
	4303	5031#	5038	5062#	5069	5208#	5211	5251#	5254	5343#	5344	5376#	5380#	5381	5385#
	5386	5390#	5391	5396#	5397	5404#	5409#	5410	5419#	5420	5432#	5439#	5452#	5455	5474#
	5475	5522#	5525	5528#	5531	5534#	5540	5543#	5545#	5556#	5560	5566#	5570	5573#	5577
	5583#	5584	5586#	5593#	5594	5609#	5610	5632#	5634#	5635	5649#	5656#	5657	5670#	5677#
	5678	5692#	5699#	5700	5731#	5734	5780#	5783	5793#	5799#	5809#	5826#	5829	5835#	5838
	5855#	5868#	5872	5883#	5887	5910#	5914	5927#	5931	6033#	6037	6103#	6108	6150#	6153
	6180#	6183	6256#	6259	6305#	6308	6343#	6346	6352#	6355	6361#	6366	6372#	6375	6390#
	6394	6431#	6434	6442#	6445	6467#	6470	6854	6873	6887	6960#	6963	6996#	7056#	7059
	7079#	7082	7143#	7152	7548	7556#	7594	7640	7711	7718#	7801#	7806#	7959	8003#	8020
	8041	8086	8097	8159#	8160										
M\$TLAB	1#	1996#	3283#	3287#	3296#	3300#	3310#	3314#	3327#	3331#	3343#	3347#	3361#	3365#	3380#
	3384#	3396#	3400#	3413#	3423#	3427#	3643#	3651#	3688#	3696#	3716#	3726#	3780#	3789#	3799#
	3809#	3819#	3829#	3850#	3873#	3887#	3900#	3911#	3941#	3958#	3974#	3986#	3997#	4011#	4021#
	4054#	4071#	4086#	4107#	4138#	4240#	4251#	4259#	4303#	5038#	5069#	5211#	5254#	5344#	5376#
	5381#	5386#	5391#	5397#	5404#	5410#	5420#	5432#	5439#	5455#	5475#	5525#	5531#	5540#	5543#
	5545#	5560#	5570#	5577#	5584#	5586#	5594#	5610#	5622#	5635#	5649#	5657#	5678#	5700#	5734#
	5783#	5793#	5799#	5809#	5829#	5838#	5855#	5872#	5887#	5914#	5931#	6037#	6108#	6153#	6183#
	6259#	6308#	6346#	6355#	6366#	6375#	6394#	6434#	6445#	6470#	6854#	6873#	6887#	6963#	6996#
	7059#	7082#	7143#	7152#	7548#	7556#	7594#	7640#	7711#	7718#	7801#	7806#	7959#	8003#	8020#

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 279
CROSS REFERENCE TABLE -- MACRO NAMES

MSTSTL	8041#	8086#	8097#	8160#											
	1#	1996#	3283#	3287#	3296#	3300#	3310#	3314#	3327#	3331#	3343#	3347#	3361#	3365#	3380#
	3384#	3396#	3400#	3413#	3423#	3427#	3643#	3651#	3688#	3696#	3716#	3726#	3780#	3789#	3799#
	3809#	3819#	3829#	3850#	3873#	3887#	3900#	3911#	3941#	3958#	3974#	3986#	3997#	4011#	4021#
	4054#	4071#	4086#	4107#	4138#	4240#	4251#	4259#	4303#	5038#	5069#	5211#	5254#	5344#	5376#
	5381#	5386#	5391#	5397#	5404#	5410#	5420#	5432#	5439#	5455#	5475#	5525#	5531#	5540#	5543#
	5545#	5560#	5570#	5577#	5584#	5586#	5594#	5610#	5632#	5635#	5649#	5657#	5678#	5700#	5734#
	5783#	5793#	5799#	5809#	5829#	5838#	5855#	5872#	5887#	5914#	5931#	6037#	6108#	6153#	6183#
	6259#	6308#	6346#	6355#	6366#	6375#	6394#	6434#	6445#	6470#	6854#	6873#	6887#	6963#	6996#
	7059#	7082#	7143#	7152#	7548#	7556#	7594#	7640#	7711#	7718#	7801#	7806#	7959#	8003#	8020#
	8041#	8086#	8097#	8160#											
M\$WORD	1#	1996#	2054#	2063	2107#	2109	3430#	3696#	3698	3699#	5439#	5441	5442#	5586#	5649#
	5670#	5692#	5799#	5809#	5811	5812#	5855#	6854#	6855	6856	6857	6873#	6874	6875	6876
	6887#	6888	6889	6890	6996#	6998	6999#	7152#	7153	7154	7155	7548#	7549	7550	7551
	7594#	7595	7596	7597	7640#	7641	7642	7643	7711#	7712	7713	7714	7959#	7960	7961
	7962	8020#	8021	8022	8023	8041#	8042	8043	8044	8086#	8087	8088	8089	8097#	8098
	8099	8100	8184#	8185	8193#	8198#	8203#	8209#	8251	8252					
M\$XFER	1#	1996#													
NODCL	2002#	2955	2956	2957	2958	2959	2960	2961	2962	2963	2964	2965	2966	2967	2968
	2969	2970	2971	2972	2973	2974	2975	2976	2977	2980	2981	2982	2983	2984	2985
	2986	2987	2988	2989	2990	2991	2992	2993	2994	2995	2996	2997	3001	3002	3003
	3004	3005	3010	3011	3012	3013	3014	3017	3018	3019	3020	3021	3022	3023	3024
	3026	3027	3028	3029	3030	3031	3034	3035	3036	3037	3039	3040	3042	3043	3045
	3046	3048	3051	3052	3053	3054	3055	3056	3057	3058	3059	3060	3061	3062	3063
	3064	3065	3066	3069	3070	3071	3072	3073	3074	3075	3078	3079	3080	3081	3082
	3083	3086	3087	3088	3090	3093	3094	3095	3096	3097	3098	3099	3100	3103	3104
	3105	3108	3109	3110	3111	3112	3113	3114	3116	3117	3118	3119	3120	3123	4172
	4173	4174	4175	4176	4177	4178	4179	4180	4181	4182	4185	4186	4190	4191	4192
	4193	4194	4195	4196	4197	4201	4204								
OPEN	1#	1996#													
POINTE	1#	1996#	2008												
PRINTB	1#	1996#	3274	3291	3304	3321	3337	3353	3370	3389	3405	3415			
PRINTF	1#	1996#	3639	3647	3684	3712	3722	4133	4235	4245	4254	4299	5207	5250	5451
	5521	5527	5533	5730	5779	5825	5834	5867	5882	5909	5926	6032	6102	6149	6179
	6255	6304	6430	6441	6466	6959	7055	7078							
PRINTS	1#	1996#	3776	3784	3791	3801	3811	3821	3846	3869	3879	3894	3906	3935	3950
	3964	3980	3990	4003	4013	4047	4064	4079	4102	5030	5061	6342	6351	6360	6371
	6389														
PRINTX	1#	1996#													
READBU	1#	1996#	5431												
REDEF	1#	1996#	5379	5384	5389	5395									
RFLAGS	1#	1996#													
SETPRI	1#	1996#	5582	5633											
SETVEC	1#	1996#	5555	5565	5572										
SLASH	1#	1996#													
STARS	1#	1996#													
SVC	1#	1996#													
XFER	1#	1996#	3430#	5586#	5649#	5670#	5692#	5799#	5855#						
XFERF	1#	1996#													
XFERT	1#	1996#													

. ABS. 067542 000

ERRORS DETECTED: 0

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 280
CROSS REFERENCE TABLE -- MACRO NAMES

CZKMSA.BIN,CZKMSA.LST/CRF/SOL/NL:TOC=SVC34R.MLB,CZKMSA.P11
RUN-TIME: 30 39 4 SECONDS
RUN-TIME RATIO: 117/75=1.5
CORE USED: 23K (45 PAGES)