

KMS-BD

KMS-BD DCLT
CZKMSAO

AH-S978A-MC
FICHE 1 OF 2

MAY 1983
COPYRIGHT © 1983
MADE IN USA

00000000

KMS-BD

KMS-BD DCLT
CZKMSAO

AH-S978A-MC
FICHE 2 OF 2

MAY 1983
COPYRIGHT © 1983
MADE IN USA



SVC.MLB SOURCE FILE MACY11 30A(1052) 23-DEC-82 14:01 PAGE 2
CZKMSA.P11 23-DEC-82 13:54

SEQ 1

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
DEC

PDP
DECUS

UNIBUS
DECTAPE

MASSBUS

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 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 4
CZKMSA.P11 23-DEC-82 13:54

.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

CZKMSAO KMS11-BD/BE DCLT MACY11 30A(1052) 23-DEC-82 14:01 PAGE 6
 CZKMSA.P11 23-DEC-82 13:54

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 (CHQUS?.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-D/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-KMC11-00
- 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 (CHQUS?.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	(LINE UNIT-MODEM
INTERACTION TEST	YM-2136D-3 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.

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 8

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".

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 9

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					

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 10

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 MACY11 30A(1052) 23-DEC-82 14:01 PAGE 12
 CZKMSA.P11 23-DEC-82 13:54

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
=ZEROS
=1ALT
=0ALT
=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:

=INTERNALTL 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 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 17
CZKMSA.P11 23-DEC-82 13:54

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

COMMAND -----	DESCRIPTION -----
RPT> HELP	PRINTS HELP INFORMATION FOR RPT>
RPT> ?	PRINTS HELP INFORMATION FOR RPT>
RPT> 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

MACV11 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
ZEROS - /SIZE=64/COPY=0
OALT - /SIZE=64/COPY=0
1ALT - /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

MACV11 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

MACV11 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

RECV=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
SEL0 = 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.

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 24

;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.

```

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 26

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 MACY11 30A(1052) 23-DEC-82 14:01 PAGE 28
CZKMSA.P11 23-DEC-82 13:54

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
CMF	ABOUT TO DO DATA CHECKING OF RECVD VS. EXPTD DATA
CML	LENGTH ERROR OCCURRED DURING DATA COMPARISON
CMO	DATA ERROR OCCURRED DURING DATA COMPARISON
EOP	END OF PASS

CZKMSAO KMS11-BD/BE DCLT MACY11 30A(1052) 23-DEC-82 14:01 PAGE 29
 CZKMSA.P11 23-DEC-82 13:54

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 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 30
 CZKMSA.P11 23-DEC-82 13:54

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
		:

CZKMSAO KMS11-BD/BE DCLT MACV11 30A(1052) 23-DEC-82 14:01 PAGE 31
 CZKMSA.P11 23-DEC-82 13:54

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".

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 33

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

INTERNALITL 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
ZEROS	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 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 35
CZKMSA.P11 23-DEC-82 13:54

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/NOMODEM
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=1/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=1/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 ITEP 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/NODEM
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 MACY11 30A(1052) 23-DEC-82 14:01 PAGE 39
 CZKMSA.P11 23-DEC-82 13:54

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 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 41
 CZKMSA.P11 23-DEC-82 13:54

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/NOMODEM
 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 MACY11 30A(1052) 23-DEC-82 14:01 PAGE 42
 CZKMSA.P11 23-DEC-82 13:54

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 MACY11 30A(1052) 23-DEC-82 14:01 PAGE 43
CZKMSA.P11 23-DEC-82 13:54

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

MACY11 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

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

NODE B

DCLT> (A) ? CL E
DCLT> (A) ? CL T
DCLT> (A) ? SE E=OALT/S=10
DCLT> (A) ? SE E=CCITT/S=20
DCLT> (A) ? SE E=ALPHA/S=30
DCLT> (A) ? SE T=ZERO/S=11
DCLT> (A) ? SE T=ONES/S=21
DCLT> (A) ? SE T=ITEP/S=31
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 ALPHANUMERICS
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/NOMODEM

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

MACV11 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 ITP 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=ITP/SIZE=58
MODE=ACTIVE/PASS=00001
/NOSTATUS/CHECK/NOECHO/NOMODEM

DCLT> (A) ?

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 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 47
 CZKMSA.P11 23-DEC-82 13:54

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

MACV11 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,1TEP,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=!NT,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 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 49
 CZKMSA.P11 23-DEC-82 13:54

7.3.4 EXAMPLES OF RUN COMMANDS

YOU CAN FIND SEVERAL EXPAMLES OF THE RUN COMMAND 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 CUMULATIVE ERRORS

DR>

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 50

7.3.5 EXAMPLES OF PRINT COMMANDS

THE PRINT COMMAND IS USED FROM THE SUPERVISOR DR> OR THE DCLT > LEVEL TO ENTER THE REPORT LEVEL. THE PROMPT AT THE REPORT LEVEL IS RPT>.

HERE ARE SOME EXAMPLES OF RPT> LEVEL COMMANDS

THE HELP COMMAND
RPT> HELP OR RPT> ?

PRINTS THE FOLLOWING:

DCLT REPORT CMDS:
EXIT OR LOG
STATUSONLINE

RPT> (A) ?

THE LOG COMMAND
RPT> (A) ? LOG

PRINTS A LOG ENTRY FOR THE MOST RECENT EVENTS LOGGED SINCE THE LAST DR> START OR DR> RESTART COMMAND. A MAXIMUM OF 45 EVENTS CAN BE PRINTED.

[illegible]

RPT> (A) ?

CZKMSAO KMS11-BD/BE DCLT MACY11 30A(1052) 23-DEC-82 14:01 PAGE 51
CZKMSA.P11 23-DEC-82 13:54

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 MACY11 30A(1052) 23-DEC-82 14:01 PAGE 52
CZKMSA.P11 23-DEC-82 13:54

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

002010

002011

002012

002012

002014

002014

002016

002016

002020

002020

002022

002022

002024

002024

002026

002026

002030

002030

002032

002032

002034

002034

002036

.SBTTL PROGRAM HEADER

BGNMOD

```

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

```

POINTER ERRIBL,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

LSIML::

.WORD 1800.

LSHPCP::

.WORD LSHARD

LSSPCP::

.WORD 0

LSHPTP::

.WORD LSHW

LSSTP::

.WORD 0

LSLADP::

.WORD LSLAST

LSSTA::

.WORD 0

LSCO::

.WORD 0

LSDTYP::

.WORD 0

LSAPT::

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

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
LSDEVP::	.WORD	LSDVTYP
LSREPP::	.WORD	LSRPT
LSEXP4::	.WORD	0
LSEXP5::	.WORD	0
LSAUT::	.WORD	LSAU
LSDUT::	.WORD	LSDU
LSLUN::	.WORD	0
LSDESP::	.WORD	LSDESC
LSLOAD::	EMT	ESLOAD
LSETP::	.WORD	LSERRTBL
LSICP::	.WORD	LSINIT
LSCCP::	.WORD	LSCLEAN
LSACP::	.WORD	LSAUTO
LSPRT::	.WORD	LSPROT
LSTEST::	.WORD	0
LSDLV::	.WORD	0
LSHIME::	.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
LSDISPATCH::
..WORD 11

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 56
DEFAULT HARDWARE P-TABLE

```

2111      ;**
2112      ; THE DEFAULT HARDWARE P-TABLE CONTAINS DEFAULT VALUES OF
2113      ; THE TEST-DEVICE PARAMETERS. THE STRUCTURE OF THIS TABLE
2114      ; IS IDENTICAL TO THE STRUCTURE OF THE HARDWARE P-TABLES,
2115      ; AND IS USED AS A "TEMPLATE" FOR BUILDING THE P-TABLES.
2116      ;--
2117
2118      002126      BGNHW      DFPTBL
2119      002126      000006
2120      002130
2121      002130
2122
2123
2124      ;INDEPENDENT SECTION
2125      ; THE NUMBERS IN BRACKETS ARE THE OFFSET VALUES USED IN THE PARAMETER
2126      ; CODING SECTION.
2127
2128
2129
2130
2131
2132      ;DEVICE DEPENDENT SECTION
2133      ; ADDING OR REMOVING WORDS FROM THIS TABLE EFFECTS THE "GET" CALLS IN
2134      ; THE HARDWARE PARAMETER CODING SECTION BY CHANGING "OFFSETS"
2135
2136      002130      164100      .WORD      164100      ;[0] CSR ADDRESS (KMC11-B)
2137      002132      000400      .WORD      400      ;[2] INTERRUPT VECTOR (KMC11-B)
2138      002134      000240      .WORD      240      ;[4] INTERRUPT PRIORITY (BR5) (KMC11-B)
2139      002136      164110      .WORD      164110      ;[6] CSR ADDRESS (DM11-BA MODEM CONTROL (CSR)
2140      002140      000000      .WORD      0      ;[10]SPARE
2141      002142      000000      .WORD      0      ;[12]SPARE
2142
2143
2144      002144      ENDPHW
2145      002144

```

L10000:

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 57
DEFAULT HARDWARE P-TABLE

2146
2147
2148

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

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

```

3. PRIORITY LEVEL DEFINITIONS

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

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 59
GLOEAL 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	MOE== 100000
2227		

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 60
GLCBAL 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      MSG LIM=15.    ; MAX NO. OF MESSAGES PER BUFFER
2235      ; (FOR EACH INCREMENT (+1) TO MSG LIM,
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 SUPPORT MODEM LOCAL,MODEM REMOTE OR MOP
2258      000003      MODLOC= 3    ; MODEM 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      MOCCHK= 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

: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)

2278
2279
2280
2281
2282
2283
2284 000000
2285 000002
2286 000004
2287 000006
2288 000010
2289 000012
2290 000014
2291 000016
2292 000020
2293 000022
2294 000024
2295 000026
2296
2297
2298

TXQ= 0
TXC= 2
RXQ= 4
RXC= 6
DER= 10
DVI= 12
DCK= 14
MSC= 16
DLE= 20
DDE= 22
EOP= 24
ABO= 26

:TRANSMIT MESSAGE QUEUED
:TRANSMIT COMPLETE
:RECEIVE BUFFER QUEUED
:RECEIVE COMPLETE
:DEVICE INFORMATION
:DEVICE ABOUT TO INIT
:DATA COMPARISON RESULTS
:MODEM STATUS CHANGE
:DATA COMPARISON LENGTH ERROR
:DATA COMPARISON DATA ERROR
:END OF PASS

:EQUATES FOR FLAG WORD

2299 000001
2300 000002
2301 000004
2302 000010
2303 000100
2304 000200
2305 000400
2306 001000
2307 002000
2308 004000
2309
2310
2311
2312
2313
2314

ININT = BIT0
OTINT = BIT1
QRX = BIT2
QTX = BIT3
ERX = BIT6
ETX = BIT7
INITC = BIT8
SETC = BIT9
STAT = BIT10
TXTO = BIT11

:INPUT INT. REC.
:OUTPUT INT REC.
:RX QUED /COMPL
:TX QUED/COMPL
:EXPECT TO GET A RX COMPLETED
:EXPECT TO GET A TX COMPLETED
:EXPECT INIT SYSTEM COMPLETED
:EXPECT SET LINE COMPLETED
:STATUS REPORT EXPECTED
:TRANSMIT TIMEOUT OCCURRED

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

2315 000000
2316 000001
2317 000002
2318 000003
2319 000004
2320 000005
2321 000006
2322 000007
2323 000010
2324 000011
2325 000012
2326

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
GLCBAL 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	CBLLLOP=45
2366	000046	LMLOP=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	SETEI=60
2377	000061	SLST=61
2378	000062	ETRB=62
2379	000063	KTRB=63
2380	000064	KALL=64
2381	000065	EKTB=65
2382		

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 63
GLOBAL EQUATES SECTION

SEQ 62

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
RDMPS=6
RDMPQ=7
RDMPE=10

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 64
GLOBAL EQUATES SECTION

```

2394
2395
2396      ; DEVICE DEPENDENT EQUATES
2397      ; MODEM SIGNAL BIT DEFINITIONS
2398      ; IF SIGNAL AVAILABLE IN DEVICE, EQUATE NAME TO BIT POSITION,
2399      ; ELSE EQUATE IT TO = 0
2400
2401      ; (KMS11-C FIRMWARE DOES NOT MAKE MODEM SIGNALS AVAILABLE)
2402      000000      CTS=      0      ;CLEAR TO SEND (CIRCUIT CB)
2403      000000      DSR=      0      ;DATA SET READY (CIRCUIT CC)
2404      000000      DCD=      0      ;DATA CARRIER DETECT (CIRCUIT CF)
2405      000000      RTS=      0      ;REQUEST TO SEND (CIRCUIT CA)
2406      000000      RI=      0      ;RING INDICATOR (CIRCUIT CE)
2407      000000      SQD=      0      ;SIGNAL QUALITY DETECT (CIRCUIT CG)
2408      000000      TM=      0      ;MODEM IN TEST MODE (RS 449 ONLY CIRCUIT TM)
2409
2410
2411      ; DEVICE BIT DEFINITIONS
2412
2413      ;KMS11-BD/BE REGISTER BIT DEFINITIONS
2414      ;CSR0:
2415      100000      KRUN= BIT15      ;RUN BIT
2416      040000      MCLR= BIT14      ;MASTER CLEAR
2417      ;BIT13      ;UNDEFINED
2418      010000      EXT1= BIT12      ;STEP LINE UNIT(OPTIONAL. USED WITH SOME LINE UNITS)
2419      004000      EXT0= BIT11      ;LINE UNIT LOOP (OPTIONAL. USED WITH SOME LINE UNITS)
2420      002000      RAM0= BIT10      ;LOAD VERIFY RAM
2421      001000      RAM1= BIT9       ;MAINTENANCE INSTRUCTION REGISTER CLEARS PC
2422      ;AND LOADS MAINT.INSTRUCTION INTO CRAM LOCATION ZERO
2423      000400      STEP= BIT8       ;SINGLE STEP MICROPROCESSOR
2424      000200      RQI=  BIT7       ;REQUEST PERMISSION TO TRANSFER COMMAND TO DEVICE
2425      000020      IE0=  BIT4       ;OUTPUT INTERRUPT ENABLE
2426      000001      IE1=  BIT0       ;INPUT INTERRUPT ENABLE
2427
2428      ;CSR2: USED FOR COMMUNICATION BETWEEN COMM DEVICE AND ITS LOCAL HOST PROCESSOR
2429
2430      000200      RDY0= BIT7       ;READY OUT BIT.
2431      ;SET BY DEVICE TO REQUEST HOST TO ACCEPT MESSAGE
2432      ;Cleared BY LOCAL HOST WHEN IT HAS ACCEPTED MESSAGE
2433
2434      000020      RDYI= BIT4       ;READY IN BIT
2435      ;SET BY DEVICE TO GRANT LOCAL HOST PERMISSION
2436      ; TO TRANSFER COMMAND TO DEVICE
2437      ;Cleared BY LOCAL HOST WHEN COMMAND HAS BEEN
2438      ; TRANSFERRED TO DEVICE CSRS
2439
2440      ;KMLTYP - ADDRESS IN KMC DATA RAM OF INTERNAL LOOPBACK MODE SPECIFIER
2441
2442      007726      KMLTYP= 7726      ;***THIS VALUE IS A KMC DATA ADDRESS
2443

```

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

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:

: 'INTERPROCESSOR TEST PROGRAM'S (ITEP)' MESSAGE
: #1 (DP1:)

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

.ASCII <15><12><001><177><177><177><177>

EMSG5:
MSG6:

: ALPHA-NUMERICS (OR FUTURE COMM TURNAROUND MSG)
.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 /

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

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

MACY1' 30A(1052) 23-DEC-82 17:01 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	.ASCII ?/[\]^_`?
2540	002510	022537	000		
2541	002513				EMSG6:
2542		002514			.EVEN

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 69
DEFAULT MESSAGE DEFINITIONS AND TABLES

```
2543 ; *****
2544 ; THESE THREE STORAGE AREAS MUST NOT BE SEPARATED !!!
2545
2546 002514 047045 040445 OPBFPT: .ASCII /%N%A/
2547 002520 000122 OPBUF: .BLKB 82. ;BUFFER FOR OPERATOR SPEC'D MESSAGES
2548 002642 OPEND:
2549
2550 ; THE ABOVE THREE LINES MUST BE KEPT TOGETHER
2551 ; *****
2552
2553 002642 033 MSG8: .BYTE 33 ;RECEIVE BUFFER FILL PATTERN
2554 002643 EMSG8:
2555
```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 70
DEFAULT MESSAGE DEFINITIONS AND TABLES

```

2556          *****KMS11-BD/BE DOES NOT SUPPORT DOWN LINE LOAD*****
2557          ; DOWN-LINE-LOAD MESSAGE DEFINITIONS
2558
2559          ;;;;ENTER MOP MODE MESSAGE FORMAT
2560          ;;;;THE NODE WILL ENTER MAINTENANCE MODE ONLY IF THE PASSWORD MATCHES.
2561          DLLM1: .BYTE 6          ;BINARY CODE FOR MAINTENANCE MODE
2562          PASS1: .BYTE 0          ;PASSWORD BYTE #1  LEGAL VALUE 0 - 255
2563          PASS2: .BYTE 0          ;VALUE IN BYTE 1 IS DUPLICATED HERE
2564          PASS3: .BYTE 0          ;AND HERE
2565          PASS4: .BYTE 0          ;AND HERE
2566          DLLM1E:                ;END ENTER MOP MODE MESSAGE FORMAT
2567          ;;;;MEMORY LOAD WITH TRANSFER ADDRESS MESSAGE FORMAT
2568          DLLM2: .BYTE 0          ;CODE
2569          .BYTE 0          ;LOAD NUMBER
2570          .BYTE 6          ;LOAD ADDRESS LSB
2571          .BYTE 0
2572          .BYTE 0
2573          .BYTE 0          ;LOAD ADDRESS
2574
2575          ;;'LOADER' IMAGE DATA
2576
2577          ;LOCATIONS (WORDS)
2578          ;ADDR  CONTENTS
2579          ;-----
2580          NOP                                ;'BYTE COUNT' :006 000240
2581          CLR @#6                          ; (240(8)=160.) :010 005037 000006
2582          MOV #1000,SP                     ;SET UP TTY      :014 012706 001000
2583          MOV #177560,R1                   ;MAKE ADDR.PIC  :020 012706 177560
2584          MOV PC,R0                        ;ADDRESS MSG.    :024 010700
2585          ADD #<MSG-.>,R0                  ;TTY READY?     :026 062700 000034
2586          1$: TSTB 4(R1)                   ;WAIT TIL YES   :032 105761 000004
2587          BPL 1$                           ;TYPE A CHAR    :036 100375
2588          MOVB (R0)+,6(R1)                 ;KEEP GOING     :040 112061 000006
2589          BNE 1$                           ;SETUP PWR FAIL :044 001372
2590          MOV #26,@#24                    ;ENSURE TBIT CLR:046 012737 000026 000024
2591          CLR @#26                         ;JMP ON YOURSELF:054 005037 000026
2592          BR                                ;                :060 000777
2593          MSG: .ASCII <12><15>/BOOT MESSAGE WAS RECE/ :062 006412 047502 .....
2594
2595          .ASCII /IVED SUCCESSFULLY -E/      ;
2596
2597          .ASCII /ND OF TEST!!/              ;
2598
2599          .ASCII <12><15>/....RELOAD PROGRAM..../:
2600
2601
2602
2603
2604
2605
2606

```

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      ;PADDING TO OBTAIN 240(8) BYTE LOADER
2608      ; (USE PART OF CCITT "WORST CASE BIT PATTERN"
2609
2610      .WORD 177603,157427,031011      ;202 177603 157427 031011
2611      .WORD 047321,163715,105221      ;210 047321 163715 105221
2612      .WORD 143325,142304,040041      ;216 143325 142304 040041
2613      .WORD 014116,052606,172334      ;224 014116 052606 172334
2614      .WORD 105025,123754,111337      ;232 105025 123754 111337
2615      .WORD 111523,030030,145064      ;240 111523 030030 145064
2616
2617      .WORD 126361      ;"CHECKSUM WORD";246 126361
2618
2619      .BYTE 6
2620      .BYTE 0
2621      .BYTE 0
2622      .BYTE 0
2623      DLLMZE:
2624      .EVEN
2625
2626

```

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          HLPEND:
2644 003276 015607          RHLPTB: .WORD RHLP1
2645 003300 015631          .WORD RHLP2
2646 003302 015646          .WORD RHLP3
2647 003304          RHLPEN:
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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 73
COMMAND LINE INTERPRETER BUFFER, LOCATIONS AND MESSAGES

```

2677      ;COMMAND LINE TRAVERSE LOCATIONS (USED BY 'PSTRV')
2678
2679 003364 000000 PSBUFA: .WORD 0      ;LOC. TO HOLD ADDR. OF CMD LINE BUFFER
2680 003366 000000 PSTREE: .WORD 0     ;LOC. TO HOLD ADDR. OF PARSING TREE
2681 003370 000000 PSACT:  .WORD 0     ;LOC. TO HOLD ADDR. OF ACTION ROUTINE
2682 003372 000000 PSCNT:  .WORD 0     ;LOC. TO BE A COUNTER LOCATION
2683 003374 000000 PSNUM:  .WORD 0     ;LOC. TO HOLD NUMERIC VALUE FROM PARSE
2684 003376 000000 PSRADX: .WORD 0     ;LOC. TO HOLD RADIX USED(LO) AND +/- (HI BYTE)
2685 003400      000 PSNNUF: .BYTE 0    ;RETURN =0 IF ENOUGH OF COMMAND FOUND
2686 003401      000 PSGDBD: .BYTE 0    ;RETURN CODE 0 IF NO ERROR FOUND
2687 003402      000 WRFLG:  .BYTE 0    ;WRITE FLAG
2688      003404 .EVEN
2689 003404 000000 VALTRB: .WORD 0      ;VALID LINE FLAG..IF SET -1 THEN VALID
2690

```

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	MSG LIM*2	:TABLE FOR MESSAGE ADDRS. & BYTE COUNTS
2698	006502	000036	PTR13: .BLKW	MSG LIM*2	:COMPARE MESSAGE POINTERS
2699	006576	000036	PTR23: .BLKW	MSG LIM*2	:RECEIVE MESSAGE POINTERS
2700	006672	000322		MSG LIM*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	CPTCLS: .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 (E ES) 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	TSPTR: .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	DVTIB: .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	LN CNT: .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	ERRCNT: .WORD	0	;ERROR COUNTER
2748	010126	000000	STADD: .WORD	0	;START ADDR.
2749	010130	000000	ENADD: .WORD	0	;END ADDR. FOR DUMP
2750	010132	000000	BYTBIT: .WORD	0	;BYTE BIT FOR DUMP ROUTINE
2751	010134	000000	CLNSET: .WORD	0	;CLEANSET FLAG SET AND CLEARED IN CLEAN UP
2752					;INDICATES TO OUTPUT HANDLER THAN NO OUTPUTS SHOULD
2753					;BE PRINTED
2754	010136	000000	RQIFLG: .WORD	0	;RQI FLAG
2755	010140	000000	FTLFLG: .WORD	0	;USED AS FATEL ERROR FLAG
2756	010142	000000	TSSFLG: .WORD	0	;USED AS TSS FLAG
2757	010144	000000	OVRCNT: .WORD	0	;USED FOR QUE OVERFLOW FLAG
2758					

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	MSGTYP: .WORD	0 ; TYPE OF DATA 0=0'S,1=1'S,2=10'S,3=01'S
2763				;4=CCITT,5=QUICK FOX,6=ALPHA/NUM,7=OPER
2764	010150	000000	CURCC: .WORD	0 ;TX/RX/CMP CHAR COUNT
2765	010152	000000	CPTRR: .WORD	0 ;CURRENT RX POINTER
2766	010154	000000	CPTR: .WORD	0 ;CURRENT POINTER
2767	010156	000000	CURADD: .WORD	0 ;CURRENT TX/RX/CMP START ADDD
2768	010160	000000	TUTCC: .WORD	0 ;TOTAL CHAR COUNT NOT MORE THEN 'BUFLIM'
2769	010162	000000	OFSET: .WORD	0 ;OFFSET COUNT
2770	010164	000000	TEMP: .WORD	0 ;TEMPORARY LOCATIONS (USED A LOT)
2771	010166	000000	TEMP1: .WORD	0
2772	010170	000000	TEMP2: .WORD	0
2773	010172	000000	TEMP3: .WORD	0
2774	010174	000000	TEMP4: .WORD	0
2775	010176	000000	TEMP5: .WORD	0
2776	010200	000000	SAVSP: .WORD	0 ;STACK POINTER SAVE AREA
2777	010202	000000	CONOTM: .WORD	0 ;CONTROL OUT ERROR MSG. ADDRESS AND TSS AND GSS MSGS.
2778	010204	000	GOOD: .BYTE	0 ;BYTE TO HOLD EXPECTED MESSAGE DATA BYTE FOR ERR REPORT
2779	010205	000	BAD: .BYTE	0 ;BYTE TO HOLD RECEIVED MESSAGE DATA BYTE FOR ERR REPORT
2780				

```

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

;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.
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	EVMRIS	;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)

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	017503
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

```

EVTLS: .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

```

```

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 " " "

```

```

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 RPTEOP ; 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

```

```

DEVS2: .WORD 0
DEVS3: .WORD 0
DEVS4: .WORD 0
LINEST: .WORD 0
ENRDS1: .WORD 0
VSTAT: .WORD 0

```

;LINE NUMBER OF STATUS DATA
 ;= 1 READ STATUS ENABLED. SET = 1 IN KSTART
 ;= 0 READ STATUS DISABLED SET = 0 IN BGNCLN
 ;= 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\$	;SKIP ANY LEADING SPACES
	CLI	<'?'>,HLP,N42\$	;IS THE FIRST NON-SP CHAR A '?'
	CLI	CLIEXI,0	; IF YES DO 'HLP' AND EXIT
N42\$:	CLI	CLISTR,HLP,N43\$,<'HELP'>	;ELSE, IS FIRST WORD A 'HELP'
	CLI	CLIEXI,0	; IF YES DO 'HLP' AND EXIT
N43\$:	CLI	CLISTR,PRNT,N44\$,<'PRINT'>	;ELSE, IS FIRST WORD A 'PRINT'
	CLI	CLIEXI,0	; IF YES DO 'PRINT' AND EXIT
N44\$:	CLI	CLISTR,EXIT,N45\$,<'EXIT'>	;ELSE, IS FIRST WORD A 'EXIT'
	CLI	CLIEXI,0	; IF YES DO 'EXIT' AND EXIT
N45\$:	CLI	CLISTR,RUN,N46\$,<'RUN'>	;ELSE, IS FIRST WORD A 'RUN'
	CLI	CLIBR,0,N80\$	; IF YES DO 'RUN' & GOTO N80\$
N46\$:	CLI	CLISTR,NOTNUF,N40\$,<'DUMP'>	;ELSE, IS FIRST WORD A 'DUMP'
	CLI	CLIBR,0,N50\$	; IF YES GOTO N80\$
N40\$:	CLI	CLISTR,CLEAR,N47\$,<'CLEAR'>	;ELSE, IS FIRST WORD A 'CLEAR'
	CLI	CLIBR,NOTNUF,N100\$	; IF YES DO 'CLR' & GOTO N100\$
N47\$:	CLI	CLISTR,NOTNUF,N20\$,<'LINE'>	;ELSE IS FIRST WORD LINE
	CLI	CLIBR,0,N105\$	
N20\$:	CLI	<'S'>,NOTNUF,N30\$	;ELSE, IS FIRST CHAR. A 'S'
	CLI	CLISTR,SHOW,N25\$,<'HOW'>	; IF YES IS REST OF WORD 'HOW'
	CLI	CLIBR,0,N100\$	; IF YES, DO 'SHOW',BR N100\$
N25\$:	CLI	CLISTR,0,N30\$,<'ET'>	; ELSE, IS REST OF WORD 'ET'
	CLI	CLIBR,0,N110\$	; IF YES, DO 'SET', BR N110\$
N30\$:	CLI	CLIERR,0	;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      N80$: CLI CLISPA,0,N30$      ;SKIP LEADING SPS, IF NONE-ERR
2980      N81$: CLI CLISTR,NOTNUF,N30$,<'MODE'>      ;IS NEXT WORD 'MODE='
2981      CLI <'='>,0,N30$      ; IF NO, IT'S WRONG -ERR -EXIT
2982      CLI CLISTR,ATVMOD,N82$,<'ACTIVE'>      ;IS NEXT WORD 'ACTIVE'
2983      CLI CLIBR,0,N115$      ; IF YES, DO 'ACTIVE',BR N115$
2984      N82$: CLI CLISTR,PASMOD,N83$,<'PASSIVE'>      ;IS NEXT WORD 'PASSIVE'
2985      CLI CLIBR,0,N115$      ; IF YES, DO 'PASSIVE',BR N115$
2986      N83$: CLI CLISTR,RECMOD,N84$,<'RECEIVE'>      ;IS NEXT WORD 'RECEIVE'
2987      CLI CLIBR,0,N115$      ; IF YES, DO 'RECEIVE',BR N115$
2988      N84$: CLI CLISTR,LISMCD,N85$,<'LISTEN'>      ;IS NEXT WORD 'LISTEN'
2989      CLI CLIBR,0,N115$      ; IF YES, DO 'LISTEN',BR N115$
2990      N85$: CLI CLISTR,DLLMOD,N86$,<'DOWNLINELOAD'>      ;IS NEXT WORD 'DOWNLINELOAD'
2991      CLI CLIBR,0,N115$      ; IF YES, DO 'DOWNLINELOAD',BR N115$
2992      N86$: CLI <'T'>,0,N30$      ;IS NEXT CHAR A 'T'
2993      CLI CLISTR,TRAMOD,N87$,<'TRANSMIT'>      ; IS REST OF WORD 'TRANSMIT'
2994      CLI CLIBR,0,N115$      ; IF YES, DO 'TRANSMIT',BR N115$
2995      N87$: CLI CLISTR,TALMOD,N30$,<'ALK'>      ; IS REST OF WORD 'ALK'
2996      CLI CLIBR,0,N115$      ; IF YES, DO 'ALK',BR N115$
2997      ; IF NO, ERROR - EXIT
2998
2999      ;SECOND KEYWORD (FOR CLEAR OR SHOW)
3000      N100$: CLI CLISPA,0,N30$      ;SKIP LEADING SPACES, NONE=ERR
3001      N102$: CLI CLISTR,CSHEXP,N104$,<'EXPECTBUFF'>      ;IS NEXT WORD 'EXPECTBUFF'
3002      CLI CLIBR,0,N120$      ; IF YES, DO CLR-EXP,EXIT
3003      N104$: CLI CLISTR,CSHTRN,N30$,<'TRANSMITBUFF'>      ;IS NEXT WORD 'TRANSMITBUFF'
3004      CLI CLIBR,0,N120$      ; IF YES, DO CLR-TRN,EXIT
3005      ; IF NO - ERROR - EXIT
3006
3007
3008      ;SECOND KEYWORD (FOR SET)
3009      N110$: CLI CLISPA,0,N30$
3010      N111$: CLI CLISTR,SETEXP,N112$,<'EXPECTMSG'>
3011      CLI CLIBR,0,N120$
3012      N112$: CLI CLISTR,SETTRN,N30$,<'TRANSMITMSG'>
3013      CLI CLIBR,0,N120$
3014
3015      ;GET ADDRESSES FOR DUMP COMMAND
3016      N50$: CLI CLIALP,0,N51$
3017      N51$: CLI CLISPA,0,N52$
3018      N52$: CLI CLIOCT,DMPD,N30$
3019      CLI <'-'>,NOTNUF,N125$
3020      CLI CLIOCT,DMPD,N30$
3021      CLI <'/'>,NOTNUF,N125$
3022      CLI <'B'>,DMPQ,N30$
3023      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$

```

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 86
COMMAND LINE ACTION TREE

```

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$: CLI      <'>,ENDQ0,N71$
3070 013002      CLI      CLIBR,0,N121$
3071 013006 N71$: CLI      CLISPA,0,N72$
3072 013012 N72$: CLI      CLIALN,0,N73$
3073 013016      CLI      CLIBR,0,N70$
3074 013022 N73$: CLI      CLIERR,BADCHR
3075
3076      ;GET QUALIFIERS (SIZE OR COPY) FOR SET MESSAGE COMMANDS
3077 013024 N121$: 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$

```

;ONLY A-Z,SP,TAB, OR 0-9 BETWEEN ''S

;PRINT ERROR IF NONE LEGAL CHAR FOR ''S

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      ;GET MAINTENANCE LOOP TYPE FOR RUN 'LOOP' QUALIFIER
3089 013110 N140$: CLI <'=>,0,N30$
3090
3091
3092 013114 N141$: CLI CLISTR,TTLLOP,N142$,<'INTERNAL TTL'>
3093 013136      CLI CLIBR,0,N115$
3094 013142 N142$: CLI CLISTR,CBLLOP,N143$,<'CABLE'>
3095 013156      CLI CLIBR,0,N115$
3096 013162 N143$: CLI CLISTR,LMDLOP,N144$,<'LOCAL MODEM'>
3097 013204      CLI CLIBR,0,N115$
3098 013210 N144$: CLI CLISTR,RMDLOP,N30$,<'REMOTE MODEM'>
3099 013232      CLI CLIBR,0,N115$
3100
3101      ;GET NUMBER FOR 'PASS' RUN QUALIFIER
3102 013236 N150$: CLI <'=>,0,N30$
3103 013242      CLI CLIDEC,PASC,N30$
3104 013246      CLI CLIBR,0,N115$
3105
3106      ;GET LINE SHOW OR LINE FOR KILL OR ESTABLISH
3107 013252 N105$: CLI CLISPA,NOTNUF,N106$
3108 013256 N106$: CLI CLISTR,SLST,N107$,<'SHOW'>
3109 013272      CLI CLIEXI,0
3110 013274 N107$: CLI CLISTR,ETRB,N108$,<'ESTABLISH'>
3111 013314      CLI CLIBR,0,N160$
3112 013320 N108$: CLI CLISTR,KTRB,N30$,<'KILL'>
3113 013334      CLI CLIEXI,0
3114
3115 013336 N160$: CLI <'=>,0,N30$
3116 013342 N161$: CLI CLISTR,KALL,N162$,<'ALL'>
3117 013354 N162$: CLI CLIDEC,EKTB,N30$
3118 013360 N163$: CLI 54,NOTNUF,N125$
3119 013364      CLI CLIBR,0,N161$
3120
3121      ;END-OF-LINE
3122 013370 N125$: CLI CLIEXI,0
3123

```

;LOOKING FOR ''

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 88
DEVICE DEPENDENT LOCATIONS

```

3124
3125
3126
3127
3128
3129 013372
3130 013372 000000
3131 013374 000000
3132 013376
3133 013376 000000
3134 013400 000000
3135 013402
3136 013402 000000
3137 013404 000000
3138 013406
3139 013406 000000
3140 013410 000000
3141
3142
3143 013412 000000
3144 013414 000000
3145 013416 000000
3146 013420 000000
3147 013422 000000
3148 013424 000000
3149 013426 000000
3150

```

;DEVICE DEPENDENT STORAGE LOCATIONS FOR
 ; CURRENT DEVICE PARAMETERS

SEL0:
 BSEL0: .WORD 0 ;ADDRESSES OF REGISTERS SEL0 THRU BSEL7
 BSEL1: .WORD 0
 SEL2:
 BSEL2: .WORD 0
 BSEL3: .WORD 0
 SEL4:
 BSEL4: .WORD 0
 BSEL5: .WORD 0
 SEL6:
 BSEL6: .WORD 0
 BSEL7: .WORD 0

INVEC: .WORD 0 ;INPUT INTERRUPT VECTOR ADDRESS
 OUTVEC: .WORD 0 ;OUTPUT INTERRUPT VECTOR ADDRESS
 INTPRI: .WORD 0 ;INTERRUPT PRIORITY
 OPTYP: .WORD 0 ;OPTION TYPE
 DEVPAR: .WORD 0
 STATYP: .WORD 0
 MODCSR: .WORD 0 ;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

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

;(KMS11-BD BUFFER DESCRIPTOR LISTS)

```

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

```

BDLMNM: .WORD 0 ;LOC TO CALCULATE BDL MEMBER NUMBER
BDLMAD: .WORD 0 ;LOC TO CALCULATE BDL MEMBER ADDRESS

BDLBAS: .WORD 0
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

013430 000000
013432 000000
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

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

L\$ERRTBL::

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
:

013714
013714
013714 046513 030523 026461
013722 042102 041054 000105

DEVTYP <KMS11-BD,BE>

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

.SBTTL PROGRAM IDENTIFICATION
: TEST DESCRIPTION
:

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

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

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

.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:	.ASCII	/DCLT>/
014010	047045	040445	044477	CLIERM:	.ASCII	/XN%?ILL CMD-BAD SYNTAX?/
014040	047045	040445	044477	CLINUF:	.ASCII	/XN%?INCMPLTE CMD?/
014063	045	022516	037501	CLINBG:	.ASCII	/XN%?NUM TOO BIG?/
014105	045	022516	037501	CLIBRX:	.ASCII	/XN%?BAD RADIX?/
014125	045	022516	037501	CLIBDL:	.ASCII	/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		.ASCII	/ TRY "L=C" & TEST CONNECTOR/
014307	045	022516	037501	CLINPS:	.ASCII	/XN%?"ECHO" VALID ONLY IN PASSIVE?/
014352	047045	040445	044477	CLIBCR:	.ASCII	/XN%?ILL CHR- "A-Z,0-9,SP,TAB" ONLY?/
014417	045	022516	037501	CLISEO:	.ASCII	/XN%?"SIZE=0" NOT VALID?/
014450	047045	040445	052077	CLIPW:	.ASCII	/XN%?TRANSMIT & EXPECT LIST MUST BE IDENTICAL FOR LOOP?/
014540	047045	040445	044124	HLP0:	.ASCII	/XN%THIS IS DCLT. TYPE "H" OR "?" FOR DETAILS/
014616	047045	052045	000	HLPF:	.ASCII	/XN%T/
014623	104	046103	020124	HLP1:	.ASCII	/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		.ASCII	? DUMP START-END/B?
014754	046040	047111	020105	HLP2B:	.ASCII	? LINE SHOW, LINE ESTABLISH=N,(N=0-7) OR LINE KILL?
015036	051440	052105	042440	HLP3:	.ASCII	? SET EXPECTMSG OR TRANSMITMSG=TYPE/SIZE=N OR /COPY=N?
015123	040	042523	020124	HLP3A:	.ASCII	/ SET EXPECT=TRANSMIT/
015150	020040	052040	050131	HLP4:	.ASCII	? TYPE=ONES,ZEROES,1ALT,0ALT,ITEP,CCITT,ALPHA?
015227	040	020040	020040	HLP4A:	.ASCII	/ OR "OPR SPCD=A-Z,SP,TAB,0-9 IN QUOTES"/
015305	040	052522	020116	HLP5:	.ASCII	? 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		.ASCII	/ LTP=INT,CAB/
015457	045	022516	042101	DLLQ1:	.ASCII	/XN%DOWN LINE LOAD NOT AVAILABLE FOR THE KMS11-BD,BE/
015544	050122	037124	000	CLISRP:	.ASCII	/RPT>/
015551	045	022516	052101	RHLP0:	.ASCII	/XN%ATYPE "H" OR "?" FOR HELP!/
015607	104	046103	020124	RHLP1:	.ASCII	/DCLT REPORT CMDS:/
015631	040	054105	052111	RHLP2:	.ASCII	/ EXIT OR LOG/
015646	051440	040524	052524	RHLP3:	.ASCII	/ STATUSONLINE/
015664	047045	040445	051515	SHMSG:	.ASCII	?XN%MSG: TYPE=X1%A/SIZE=XD3?
015720	042532	047522	051505	SHTYP0:	.ASCII	/ZEROES/
015727	117	042516	000123	SHTYP1:	.ASCII	/ONES/
015734	040461	052114	000	SHTYP2:	.ASCII	/1ALT/
015741	060	046101	000124	SHTYP3:	.ASCII	/0ALT/
015746	041503	052111	000124	SHTYP4:	.ASCII	/CCITT/
015754	052111	050105	000	SHTYP5:	.ASCII	/ITEP/
015761	101	050114	040510	SHTYP6:	.ASCII	/ALPHA/
015767	117	051120	051440	SHTYP7:	.ASCII	/OPR SPEC/
016000	047045	040445	047516	SHTRE:	.ASCII	\XN%AND LINE NUMBER IS ESTABLISHED\
016042	047045	040445	044514	SHTRH:	.ASCII	\XN%LINE NUMBER : \
016066	042045	000063		SHTAP:	.ASCII	\XD3\
				SHFL:	.ASCII	\XN%LINE NUMBER LIST FULL - LINE= %23%A NOT ADDED\
				SHUN:	.ASCII	\XN%?LINE NUMBER= %23%A IS NOT UNIQUE?\
				SHNF:	.ASCII	\XN%?LINE NUMBER= %23%A NOT FOUND?\
016072	047045	040445	040477	SHTLPA:	.ASCII	/XN%A A LINE MUST BE ESTABLISHED TO EXECUTE?/
016146	047045	040445	046077	SHTIV:	.ASCII	\XN%?LINE NUMBER= %23%A INVALID (NUMBERS 0-7 ARE VALID)?\
016237	045	022516	051101	SHTBR:	.ASCII	/XN%ARX BUFFER NOT BIG ENOUGHXN%ATOO MANY LINES OR MSGS/
016326	042522	042503	053111	MO0:	.ASCII	/RECEIVE/
016336	051124	047101	046523	MO1:	.ASCII	/TRANSMIT/
016347	120	051501	044523	MO2:	.ASCII	/PASSIVE/
016357	101	052103	053111	MO3:	.ASCII	/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	M04:	.ASCIIZ	/DOWNLINELOAD/
016403	124	046101	000113	M05:	.ASCIIZ	/TALK/
016410	044514	052123	047105	M06:	.ASCIIZ	/LISTEN/
016417	000			LP0:	.ASCIIZ	//
016420	046057	047517	036520	LP00:	.ASCIIZ	?/LOOP=?
016427	111	052116	051105	LP1:	.ASCIIZ	?INTERNAL?
016440	040503	046102	000105	LP2:	.ASCIIZ	?CABLE?
016446	047514	040503	046514	LP3:	.ASCIIZ	?LOCALMODEM?
016461	122	046505	052117	LP4:	.ASCIIZ	?REMOTEMODEM?
016475	116	117		PNST:	.ASCII	/NO/
016477	123	040524	052524	PST:	.ASCIIZ	/STATUS/
016506	047516			PNCK:	.ASCII	/NO/
016510	044103	041505	000113	PCK:	.ASCIIZ	/CHECK/
016516	047516			PNEC:	.ASCII	/NO/
016520	041505	047510	000	PEC:	.ASCIIZ	/ECHO/
016525	116	117		PNMS:	.ASCII	/NO/
016527	115	042117	046505	PMS:	.ASCIIZ	/MODEM/
016535	045	022516	046101	LISP:	.ASCIIZ	/%NXALIS>/
016546	046124	037113	000	OPRMM:	.ASCIIZ	/TLK>/
016553	124	044510	020123	LS060:	.ASCIIZ	/THIS A 50. OR 60. HZ. LSI-11:/
	016612				.EVEN	

```

; FORMAT STATEMENTS USED IN PRINT CALLS

```

016612	047045	040445	052123	PRTNOS:	.ASCIIZ	/XNZASTATUS NOT AVAILABLE UNTIL NEXT DCLT>RUNXN/
016671	045	022516	041501	BDCLK:	.ASCIIZ	/XNZACLOCK NOT FOUND/
016715	045	022516	041101	NOCLK:	.ASCIIZ	/XNZABAD CLOCK - PROGRAM WILL HANG ON "TIMEOUT"!!/
016776	040515	027130	041440	TABEX:	.ASCIIZ	/MAX. CHAR. MSG COUNT EXCEEDED -/
017036	052502	043106	051105	BUFEX:	.ASCIIZ	/BUFFER FULL -/
017054	047045	052045	040445	MSGTRN:	.ASCIIZ	/XNZTXA MSG. NOT BUILT !!/
017105	045	022516	041501	MSGTRU:	.ASCIIZ	/XNZACHAR. COUNT EXCEEDS BUFF LIMIT - MSG TRUNCATED/
017170	047045	051445	022465	SHFO:	.ASCIIZ	?XNZS5XAMODE=XTXTXTXA/PASS=XZ5?

```
017226 047045 051445 022465 SHF1: .ASCII ?%N%S%S%S%S%A/XT%A/XT%A/XT%A/XT?
017266 051445 022465 052101 EFM2: .ASCII /%S%ATOTAL MISMATCHES IN MSG = %D5/
017331      045 022516 031523 PCPM: .ASCII /%N%S3%ACALLED FROM PC=%O6/
017363      045 032523 040445 EFM11: .ASCII /%S%ACOMPARE COUNT=%D5%S3%ARECEIVE COUNT=%D5/
```

:EVENT DESCRIPTION MESSAGES

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 94
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

```

017716 040504 040524 041440 EDDDE: .ASCIZ /DATA COMPARISON DATA ERROR/
017751      105 042116 047440 EDEOP: .ASCIZ /END OF PASS/
017765      101 047102 051117 EDMOS: .ASCIZ /ABNORMAL MODEM STATUS CHANGE/
020022 041536 040440 047502 EDABO: .ASCIZ /^C ABORT/

```

EVENT REPORTING MESSAGES

:THE FOLLOWING ARE USED IN THE DUMP SUB ROUTINE

```

020033      045   031523   047445   BASM3: .ASCII /%S3%03/
020042    051445   022463   033117   BASM2: .ASCII /%S3%06/
020051      045   022516   033117   BASM1: .ASCII /%N%06/

020057      045   022516   052101   NULEVT: .ASCII /%N%ATHE DCLT EVENT LOG IS EMPTY/
020117      045   022516   037101   EVTF0: .ASCII /%N%A>>> DCLT EVENT LOG ENTRY <<<<<<<<<<<<<<<<<<<<<<<<</
020215      045   022516   032504   EVTF1: .ASCII /%N%D5%A:%Z2%A:%Z2%S3%T/
020244    047045   051445   022463   EVTF2: .ASCII /%N%S3%AADDR OF MSG=%06%S3%ABYTE COUNT=%D5/
020316    047045   051445   022463   EVTF3: .ASCII /%N%S3%TXN/
020330    040445   020040   042523   EFT3C: .ASCII /%A SEL0 = %06%A SEL2 = %06/
020366    051445   022463   033117   EVTF3D: .ASCII /%S3%06%S3%06%S3%T/
020410    051445   022463   033117   EVTF3F: .ASCII /%S3%06%S3 ^6%S3%ALINE NO.=%Z3/
020446    051445   022463   046101   EVTF40: .ASCII /%S3%ALINE NO. = %Z3/
020472    051445   022463   031517   EVTF42: .ASCII /%S3%03%S6%03/
020507      045   031523   047445   EVTF43: .ASCII /%S3%03%S6%03%S6%03/
020532    051445   022463   031517   EVTF44: .ASCII /%S3%03%S3%T/
020546    051445   022463   052101   EVTF6: .ASCII /%S3%ATO-FROM LINE NO.=%Z3/

020600    047045   051445   022463   EVTF4B: .ASCII /%N%S3%APASS=%D5%S3%AERRORS=%D5%S3%ANOBUFFS=%D5/
020657      045   022516   031523   EVTF4: .ASCII /%N%S3%AADDR OF MSG=%06%S3%ABYTE COUNT=%D5%S3%ANO. OF CMP ERRS=%D5/
020761      045   022516   031523   EVTF4A: .ASCII /%N%S3%AADDR OF MSG=%06%S3%ARX BYTES=%D5%S3%ACOMPARE BYTES=%D5/

021057      045   032523   040445   EVTF5A: .ASCII /%S5%ABYTE # IN MSG.= %D5%S3%AEXPTD=%03%S3%ARECVD=%03/

021143      045   022516   051501   EVTF51: .ASCII /%N%ASTATUS REPORT/<15><12>
021166    040445   044514   042516   .ASCII /%ALINE NUM = %03/
021207      045   022516   031523   EVTF52: .ASCII /%N%S3%ASTATUS = %03%S3%AiLAGS = %03/
021264    047045   051445   022463   EVTF53: .ASCII /%N%S3%ATEMP DATA = %03%S3%ALAST TX MSG = %03/
021341      045   022516   031523   EVTF: .ASCII /%N%S3%AACKS RCVD = %03%S3%AMSGS RCVD = %03/
021416    047045   051445   022463   EVTF5: .ASCII /%N%S3%ANAKS SENT = %03%S3%ANAKS RCVD = %03/
021473      045   022516   034523   EVMOCG: .ASCII /%N%S9%ACHANGED TO:/

```

: *****
: DO NOT SEPARATE THE NEXT LIST OF MESSAGES - MODEM SIGNAL HEADER AND REPORT

021516	047045	051445	022470	EVM0HD:	.ASCII	/XNXS8XAMODEM STATUS: CTS DSR DCD RTS RI SQD TM/
021576	047045	051445	022471	EVMOST:	.ASCII	/XNXS9XS9XS5XA/
021613	130	040	040	EVMCTS:	.BYTE	'X,40,40,40
021617	130	040	040	EVMDSR:	.BYTE	'X,40,40,40
021623	130	040	040	EVMDCD:	.BYTE	'X,40,40,40
021627	130	040	040	EVMRTS:	.BYTE	'X,40,40,40
021633	130	040	040	EVMRI:	.BYTE	'X,40,40,40
021637	130	040	040	EVMSQD:	.BYTE	'X,40,40,40
021643	130	040	040	EVMTM:	.BYTE	'X,40,40,40
021647	000				.BYTE	0
					.EVEN	

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

```

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

```

;DEVICE ERROR MESSAGES

```

022006 046513 020123 040506 ;DEVICE ERROR MESSAGES FROM ERRORS DETECTED IN PDP-11
022032 046513 020123 044504 DVEM0: .ASCIIZ /KMS FAILED TO START/
022075 127 044501 044524 DVEM1: .ASCIIZ /KMS DID NOT RESPOND TO CSR COMMAND/
022147 127 044501 044524 DVEM1A: .ASCIIZ /WAITING TO COMPLETE DEVICE INITIALIZATION/
022223 124 046511 020105 DVEM1B: .ASCIIZ /WAITING FOR START DDCMP REQUEST TO COMPLETE/
022275 113 051515 042040 DVEM2: .ASCIIZ /TIME OUT WAITING FOR TX OR RX TO COMPLETE/
022330 046111 042514 040507 DVEM3: .ASCIIZ /KMS DID NOT RESPOND TO ROI/
022361 015 020012 044514 DVEM4: .ASCIIZ /ILLEGAL TRANSMIT COMPLETE/
022407 111 046114 043505 DVEM5: .ASCIIZ <15><12>/ LINE NO BDL NO/
022437 015 020012 044514 .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 .ASCIIZ <15><12>/ LINE NO NAK REASON/
022705 124 020130 042522 DVEM17: .ASCIIZ /TX REJECTED BEFORE COMPLETION/
022742 005015 046040 047111 .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 MACY11 30A(1052) 23-DEC-82 14:01 PAGE 96
CZKMSA.P11 23-DEC-82 13:54 GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

023306 040516 020113 042522 NAKR00: .ASCII /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 023340      BGNMSG  ERR1
3273 023340
3274 023340      PRINTB  #EVT5A,OFFSET,<B,GOOD>,<B,BAD> ;INDIVIDUAL DATA COMPARE ERROR
3275 023340      005046
3276 023342      153716 010205
3277 023346      005046
3278 023350      153716 010204
3279 023354      013746 010162
3280 023360      012746 021057
3281 023364      012746 000004
3282 023370      010600
3283 023372      104414
3284 023374      062706 000012
3285 023400      ENDMSG
3286 023400
3287 023400      104423
3288
3289 023402      BGNMSG  ERR2
3290 023402
3291 023402      PRINTB  #EFM2,TEMP4
3292 023402      013746 010174
3293 023406      012746 017266
3294 023412      012746 000002
3295 023416      010600
3296 023420      104414
3297 023422      062706 000006
3298 023426      ENDMSG
3299 023426
3300 023426      104423
3301
3302 023430      BGNMSG  ERR10
3303 023430
3304 023430      PRINTB  #EFM11,R4,TEMP3
3305 023430      013746 010172
3306 023434      010446
3307 023436      012746 017363
3308 023442      012746 000003
3309 023446      010600
3310 023450      104414
3311 023452      062706 000010
3312 023456      ENDMSG
3313 023456
3314 023456      104423
3315
3316
3317

```

ERR1::
CLR -(SP)
BISB BAD,(SP)
CLR -(SP)
BISB GOOD,(SP)
MOV OFFSET, -(SP)
MOV #EVT5A, -(SP)
MOV #4, -(SP)
MOV SP,R0
TRAP C\$PNIB
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\$PNIB
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\$PNIB
ADD #10,SP
L10003: TRAP C\$MSG

;PRINT SELO = 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					: PRINT 'LINE NO = ' FOLLOWED BY OCTAL BYTE IN TEMP3+1		
3335	023512			BGNMSG	ERR14		
3336	023512					ERR14::	
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					: PRINT TWO OCTAL BYTES FROM TEMP3+1, TEMP4		
3351	023542			BGNMSG	ERR22		
3352	023542					ERR22::	
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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 99
GLOBAL ERROR REPORT SECTION

```

3367      : PRINT THREE OCTAL BYTES IN TEMP3+1, TEMP4, TEMP4+1
3368      BGNMSG  ERR23
3369      PRINTB  #EVT43,<B,TEMP3+1>,<B,TEMP4>,<B,TEMP4+1>
3370      ERR23::
3371      CLR      -(SP)
3372      BISB     TEMP4+1,(SP)
3373      CLR      -(SP)
3374      BISB     TEMP4,(SP)
3375      CLR      -(SP)
3376      BISB     TEMP3+1,(SP)
3377      MOV      #EVT43,-(SP)
3378      MOV      #4,-(SP)
3379      MOV      SP,R0
3380      TRAP     C$PNTB
3381      ADD      #12,SP
3382      ENDMSG
3383
3384      L10007:  TRAP     C$MSG
3385
3386      :PRINT 1 BYTE FOLLOWED BY MESSAGE POINTED TO IN TEMP4
3387      BGNMSG  ERR24
3388      PRINTB  #EVT44,<B,TEMP3+1>,TEMP4
3389      ERR24::
3390      MOV      TEMP4,-(SP)
3391      CLR      -(SP)
3392      BISB     TEMP3+1,(SP)
3393      MOV      #EVT44,-(SP)
3394      MOV      #3,-(SP)
3395      MOV      SP,R0
3396      TRAP     C$PNTB
3397      ADD      #10,SP
3398      ENDMSG
3399
3400      L10010:  TRAP     C$MSG

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 100
GLCBAL ERROR REPORT SECTION

3401						
3402	023700			BGNMSG	ERR25	
3403	023700					ERR25::
3404						;PRINT 4 STATUS BYTES FROM TEMP4,TEMP5
3405	023700			PRINTB	#EVTFS2,<B,TEMP4>,<B,TEMP4+1>	
3406	023700	005046				CLR -(SP)
3407	023702	153716	010175			BISB TEMP4+1,(SP)
3408	023706	005046				CLR -(SP)
3409	023710	153716	010174			BISB TEMP4,(SP)
3410	023714	012746	021207			MOV #EVTFS2,-(SP)
3411	023720	012746	000003			MOV #3,-(SP)
3412	023724	010600				MOV SP,R0
3413	023726	104414				TRAP (\$PNTB
3414	023730	062706	000010			ADD #10,SP
3415	023734			PRINTB	#EVTFS3,<B,TEMP5>,<B,TEMP5+1>	
3416	023734	005046				CLR -(SP)
3417	023736	153716	010177			BISB TEMP5+1,(SP)
3418	023742	005046				CLR -(SP)
3419	023744	153716	010176			BISB TEMP5,(SP)
3420	023750	012746	021264			MOV #EVTFS3,-(SP)
3421	023754	012746	000003			MOV #3,-(SP)
3422	023760	010600				MOV SP,R0
3423	023762	104414				TRAP (\$PNTB
3424	023764	062706	000010			ADD #10,SP
3425	023770			ENDMSG		
3426	023770					L10011:
3427	023770	104423				TRAP (\$MSG
3428						
3429	023772			EXIT	MSG	
3430	023772	000167				.WORD JSJMP
3431	023774	177772				.WORD L10011-2-
3432						
3433						

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

MACV11 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
      ASL      (R2)         ; THE PSW WITH A "SETVEC" CALL
      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 012122
023776 012112
024000 006312
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 .SBTTL          CLOCK INTERRUPT SERVICE ROUTINE
3480
3481 : **
3482 : FUNCTIONAL DESCRIPTION:
3483 : THIS IS THE CLOCK INTERRUPT SERVICE ROUTINE WHICH TAKES CARE OF
3484 : KEEPING THE "TIME-SINCE-START" AND COUNTING DOWN ANY OF THE
3485 : "EVENT" TIMERS. THE TIMERS ARE USED TO TIME COMPLETION OF DEVICE
3486 : REQUESTS. THE "TIME-SINCE-START" IS USED TO BE LOGGED WITH EACH ENTRY
3487 : INTO THE EVENT LOG.
3488
3489 : IMPLICIT INPUTS:
3490 : TIMTCK: THE CURRENT NO. OF TICKS LEFT TO BE COUNTED UNTIL A SECOND
3491 :         WAS BEEN COUNTED OFF
3492 : CLKHZ: THE NO. OF TICKS IN A SECOND, DETERMINED BY THE SYS. LINE FREQ.
3493 : TIMMIN & TIMSEC: CURRENT VALUE OF "TIME-SINCE-START"
3494 :                  IN MINUTES & SECONDS
3495 : TIMER 1,2, & S: CURRENT VALUES OF THE "EVENT TIMERS"
3496
3497 : IMPLICIT OUTPUTS:
3498 : NEW VALUE OF EVENT TIMER "1" DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
3499 : NEW VALUE OF EVENT TIMER "2" DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
3500 : NEW VALUE OF EVENT TIMER "S" DECREMENTED BY 1 SECOND IF IT WAS NON-ZERO
3501
3502 : FUNCTIONAL SIDE EFFECTS:
3503 : THE CLOCK IS DISABLED UPON ENTRY AND REENABLED WHEN LEAVING
3504
3505 : CALLING SEQUENCE:
3506 : THIS ROUTINE IS CALLED WHEN THE CLOCK INTERRUPTS THRU "CLKVEC".
3507 : THE ADDRESS OF THIS ROUTINE WAS LOADED INTO THE CLOCK'S INTERRUPT
3508 : VECTOR WITH A SUPERVISOR "SETVEC" CALL.
3509 : --
3510
3511 024022          BGNSRV CLKINT
3512 024022
3513
3514 024022 005077 164222          CLR @CLKCSR          ;DISABLE THE CLOCK FROM INTERRUPTING
3515 024026 005337 010266          DEC TIMTCK          ;DECREMENT THE # OF TICKS/SEC.
3516 024032 001015                BNE 1$              ;GO CHECK TIMERS (182-TICKS, 3-SECONDS)
3517 024034 013737 010256 010266 MOV CLKHZ,TIMTCK      ;RESET THE # OF TICKS/SEC.
3518 024042 005237 010264          INC TIMSEC          ;INC # OF SECS-SINCE-START
3519 024046 022737 000074 010264 CMP #60.,TIMSEC      ;SEE IF WE'VE COUNTED 60 SECS. YET
3520 024054 001004                BNE 1$              ;IF NOT, GO CHECK TIMERS
3521 024056 005237 010262          INC TIMMIN          ; ELSE INC MINUTES-SINCE-START
3522 024062 005037 010264          CLR TIMSEC          ; AND RESTART SECOND COUNTER
3523
3524 024066 005737 010270          1$: TST TIMER1        ;SEE IF TIMER #1, TIMING ANYTHING
3525 024072 001402                BEQ 2$              ; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
3526 024074 005337 010270          DEC TIMER1          ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
3527 024100 005737 010272          2$: TST TIMER2        ;SEE IF TIMER #2, TIMING ANYTHING
3528 024104 001402                BEQ 3$              ; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
3529 024106 005337 010272          DEC TIMER2          ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
3530 024112 005737 010274          3$: TST TIMERS        ;SEE IF TIMER #3, TIMING ANYTHING
3531 024116 001406                BEQ 4$              ; IF=0, NOTHING BEING TIMED, LEAVE
3532 024120 023737 010256 010266 CMP CLKHZ,TIMTCK      ;SEE IF A SECOND HAS BEEN COUNTED OFF
3533 024126 001002                BNE 4$              ; BR IF NO
3534 024130 005337 010274          DEC TIMERS          ; ELSE DECREMENT THE TIMER VALUE (BY 1 SEC.)

```

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 103
CLOCK INTERRUPT SERVICE ROUTINE

3535 024134 013777 010260 164106 48: MOV CLKEN,@CLKCSR ;REENABLE THE CLOCK TO INTERRUPT
3536 024142 ENDSRV
3537 024142
3538 024142 000002

L10012:

RTI

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 104
EVENT LOG SUBROUTINES

SEQ 103

```

3539 .SBTTL          EVENT LOG SUBROUTINES
3540
3541 : **
3542 : FUNCTIONAL DESCRIPTION:
3543 : THIS SUBROUTINE HAS A DIFFERENT ENTRY POINT
3544 : FOR EACH EVENT TO BE LOGGED AND ALWAYS PRINTS
3545 : THE SHORT "OPERATOR AWAKE" MESSAGE TO CONSOLE THEN LOGS THE
3546 : EVENT TYPE, TIME, AND THE OTHER 3 WORDS OF INFO PASSED TO THE
3547 : SUBROUTINE AT CALLING TIME
3548
3549 : INPUTS:
3550 : TIMMIN & TIMSEC:      CURRENT VALUE OF "TIME-SINCE-START"
3551 : TEMP2: WORD #1 OF EVENT LOG INFORMATION (FOR MOST EVENT TYPES)
3552 : TEMP3: WORD #2 OF EVENT LOG INFORMATION
3553 : TEMP4: WORD #3 OF EVENT LOG INFORMATION
3554
3555 : OUTPUTS:
3556 : "OPERATOR AWAKE" MESSAGE SENT TO THE CONSOLE
3557 : NEW EVENT LOGGED IN "EVTLOG" (EVENT LOG)
3558 : UPDATED "EVTPT:R" (EVENT LOG ENTRY POINTER)
3559
3560 : FUNCTIONAL SIDE EFFECTS:
3561 : TEMP: USED TO STORE ADDRESS OF "OPERATOR AWAKE" MESSAGE
3562 : TEMP1: USED TO SETUP THE VALUE OF THE "EVENT TYPE" BYTE FOR LOGGING
3563
3564 : CALLING SEQUENCE:
3565 : JSR      PC,LOGTXQ      ;CALL THE LOG EVENT SUBROUTINE WITH TEMP,TEMP1,
3566 :           ..           ; TEMP2, TEMP3, AND TEMP4 SETUP
3567 :           ..           ..
3568 : JSR      PC,LOGCMP
3569 : --
3570
3571 024144 LOGTXQ: MOV      #STXQ,TEMP1      ;SET UP MSG. TO PRINT
3572 024144 012737 021653 010166 MOV      #TXQ,TEMP      ;SET UP EVENT TYPE
3573 024152 012737 000000 010164 BR       LOGS1      ;GO LOG EVENT AND TIME
3574 024160 000522
3575
3576 024162 LOGTXC: MOV      #STXC,TEMP1      ;SET UP MSG. TO PRINT
3577 024162 012737 021664 010166 MOV      #TXC,TEMP      ;SET UP EVENT TYPE
3578 024170 012737 000002 010164 BR       LOGS1      ;GO LOG EVENT AND TIME
3579 024176 000513
3580
3581 024200 LOGRXQ: MOV      #SRXQ,TEMP1      ;SET UP MSG. TO PRINT
3582 024200 012737 021675 010166 MOV      #RXQ,TEMP      ;SET UP EVENT TYPE
3583 024206 012737 000004 010164 BR       LOGS1      ;GO LOG EVENT AND TIME
3584 024214 000504
3585
3586 024216 LOGRXC: MOV      #RXC,TEMP      ;SET UP EVENT TYPE
3587 024216 012737 000006 010164 BR       LOGS1      ;GO LOG EVENT AND TIME
3588 024224 000500
3589 024226 LGDVE: MOV      #SDVE,TEMP1      ;SET UP MSG. TO PRINT
3590 024226 012737 021706 010166 MOV      #DER,TEMP      ;SET UP EVENT TYPE
3591 024234 012737 000010 010164 BR       LOGS1      ;GO LOG EVENT AND TIME
3592 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				BEQ	LOGS5	;IF RXC DON'T PRINT	
3631	024444	032737	000001	010224		BIT	#STATB,PARAM		
3632	024452	001430				BEQ	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$PRINTF
        ADD #4,SP

LOGS5: MOV R3,-(SP) ;SAVE R3 ON THE STACK
        MOV EVIPTR,R3
        MOVB TEMP,(R3)+ ;LOG EVENT
        MOV CLKHZ,TEMP
        SUB TIMTCK,TEMP
        MOVB TEMP,(R3)+ ;LOG TIME SINCE START
        MOVB TIMSEC,(R3)+
        MOVB TIMMIN,(R3)+ ;TICKS,SECS AND MINS.
        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
        MOV #-1,(R3) ;IF EVENT LOG FULL GO
        MOV #EVILOG,R3 ;CONTINUE;ELSE GO TO 2
        MOV R3,EVIPTR ;LOG A TABLE END
        MOV (SP)+,R3 ;PUT R3 TO START OF TABLE
        RTS PC ;RESTORE POINTER
        ;RESTORE R3

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 107
DUMP EVENT LOG AND BASE TABLE

```

3675      .SBTTL      DUMP EVENT LOG AND BASE TABLE
3676
3677
3678      024644      010246      REPORT:  MOV      R2,-(SP)      ;SAVE R2,R3,R4 ON THE STACK
3679      024646      010346      MOV      R3,-(SP)
3680      024650      010446      MOV      R4,-(SP)
3681
3682      ;PRINT REPORT HELP MESSAGE
3683
3684      024652      PRINTF  #RHLPO
3685      024652      012746      015551      MOV      #RHLPO,-(SP)
3686      024656      012746      000001      MOV      #1,-(SP)
3687      024662      010600      MOV      SP,R0
3688      024664      104417      TRAP     C$PNTF
3689      024666      062706      000004      ADD      #4,SP
3690      024672      105037      003401      GETRCL:  CLRB     PSGDBD      ;CLEAR GOOD BAD FLAG
3691      024676      105037      003400      CLRB     PSNNUF
3692
3693      ;PRINT PROMPT RPT>
3694
3695      024702      GMANID  CLISRP,CMDBUF,A,377,1,72.,NO
3696      024702      104443
3697      024704      000406
3698      024706      003124
3699      024710      000142
3700      024712      015544
3701      024714      000377
3702      024716      000001
3703      024720      000110
3704      024722
3705      024722      012737      003124      003364      MOV      #CMDBUF,PSBUFA
3706      024730      012737      027070      003366      MOV      #CLIRT,PS TREE
3707      024736      012737      026650      003370      MOV      #CLIRAC,PSACT
3708      024744      005037      003250      CLR      QUALFG
3709      024750      004737      031506      JSR      PC,PS TRV
3710      024754      105737      003401      TSTB     PSGDBD
3711      024760      001412      BEQ      1$
3712      024762      PRINTF  #CLIERM
3713      024762      012746      014010
3714      024766      012746      000001
3715      024772      010600
3716      024774      104417
3717      024776      062706      000004
3718      025002      000137      024672      JMP      GETRCL
3719
3720      025006      105737      003400      1$:      TSTB     PSNNUF
3721      025012      001411      BEQ      10$
3722      025014      PRINTF  #CLINUF
3723      025014      012746      014040
3724      025020      012746      000001
3725      025024      010600
3726      025026      104417
3727      025030      062706      000004
3728      025034      000716      BR      GETRCL

```

```

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

```

```

;GO PRINT RPT> PROMPT AGAIN

```

```

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

```

```

MOV      #CLIERM,-(SP)
MOV      #1,-(SP)
MOV      SP,R0
TRAP     C$PNTF
ADD      #4,SP
;GO PRINT RPT> PROMPT AGAIN

```

10000\$:

```

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

```

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

MACV11 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      ;NO. CONTINUE CHECKING REQUEST
3733 025046 004737 025112      JSR PC,RPTSOL      ;DEVICE STATUS REPORT REQUESTED
3734 025052 000707      BR GETRCL      ;YES. REPORT DEVICE STATUS
3735      ;GO PRINT RPT> PROMPT AGAIN
3736 025054 023727 003246 000006 20$:  CMP KEYWD1,#RDMP$      ;IS DUMP REQUESTED ?
3737 025062 001003      BNE 30$      ;NO.
3738      ;DUMP DATA REQUESTED
3739 025064 004737 027264      JSR PC,DUMPSR      ;YES. PRINT REQUESTED DATA
3740 025070 000700      BR GETRCL      ;GO PRINT RPT> PROMPT
3741
3742 025072 023727 003246 000002 30$:  CMP KEYWD1,#RPEXT      ;IS EXIT REPORT SECTION REQUESTED ?
3743 025100 001274      BNE GETRCL      ;NO. GO PRINT RPT> PROMPT AGAIN
3744
3745      ;EXIT REPORT PROCESSOR REQUESTED
3746      ;YES. RESTORE R4,R3,R2
3747      ENDALL: MOV (SP)+,R4
3748      MOV (SP)+,R3
3749      MOV (SP)+,R2
3750      RTS PC      ;RETURN TO CALLING ROUTINE
3751
3752

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 109
RPTSOL -GET LINE STATUS REPORT FROM DEVICE-

```

3753 .SBTTL RPTSOL -GET LINE STATUS REPORT FROM DEVICE-
3754
3755
3756
3757
3758
3759
3760
3761
3762
3763
3764
3765
3766
3767 025112 RPTSOL: ;REQUEST LINE STATUS
3768 025112 005737 011512 TST ENRDST ;IS ENABLE READ STATUS ASSERTED ?
3769 025116 001407 BEQ 10$ ;NO. OMIT READING STATUS
3770 025120 005037 007636 CLR TRIBN ;YES. CLEAR LINE NUMBER
3771 025124 113737 007622 007636 MOVB TRIBLS,TRIBN ;SPECIFY LINE NUMBER
3772 025132 004737 063704 JSR PC,LNSTAT ;READ LINE STATUS
3773 025136 005737 011514 10$: TST VSTAT ;IS VALID STATUS DATA ASSERTED ?
3774 025142 001011 BNE 20$ ;YES. GO PRINT STATUS DATA
3775
3776 025144 PRINTS #PRTNOS ;STATUS NOT AVAILABLE
3777 025144 012746 016612 ;REPORT STATUS NOT AVAILABLE
3778 025150 012746 000001 MOV #PRTNOS,-(SP)
3779 025154 010600 MOV #1,-(SP)
3780 025156 104416 MOV SP,R0
3781 025160 062706 000004 TRAP C$PNTS
3782 025164 000502 ADD #4,SP
3783
3784 025166 20$: PRINTS #EVTFS1,LINEST ;PRINT STATUS
3785 025166 013746 011510 MOV LINEST,-(SP)
3786 025172 012746 021143 MOV #EVTFS1,-(SP)
3787 025176 012746 000002 MOV #2,-(SP)
3788 025202 010600 MOV SP,R0
3789 025204 104416 TRAP C$PNTS
3790 025206 062706 000006 ADD #6,SP
3791 025212 PRINTS #EVTFS2,<B,DEVS1>,<B,DEVS1+1>
3792 025212 005046 CLR -(SP)
3793 025214 153716 011501 BISB DEVS1+1,(SP)
3794 025220 005046 CLR -(SP)
3795 025222 153716 011500 BISB DEVS1,(SP)
3796 025226 012746 021207 MOV #EVTFS2,-(SP)
3797 025232 012746 000003 MOV #3,-(SP)
3798 025236 010600 MOV SP,R0
3799 025240 104416 TRAP C$PNTS
3800 025242 062706 000010 ADD #10,SP

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 110
RPT SOL -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>

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

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

308:

RTS PC

;RETURN

CLR -(SP)
BISB DEVS2+1,(SP)
CLR -(SP)
BISB DEVS2,(SP)
MOV #EVTFS3,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #10,SP

CLR -(SP)
BISB DEVS3+1,(SP)
CLR -(SP)
BISB DEVS3,(SP)
MOV #EVTFS4,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #10,SP

CLR -(SP)
BISB DEVS4+1,(SP)
CLR -(SP)
BISB DEVS4,(SP)
MOV #EVTFS5,-(SP)
MOV #3,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #10,SP

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 111
REPLOG PRINT EVENT LOG

.SBTTL REPLOG PRINT EVENT LOG

```

3836
3837
3838
3839 025374 010246      REPLOG: 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      EVTPTR,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$PNTS
3851 025432 062706 000004      ADD      #4,SP
3852 025436 000137 026606      JMP      ENDEVT      ;AND END
3853
3854 025442 162702 000014      RPT1:  SUB      #14,R2      ;NOW POINT BACK TO TOP OF ENTRY U
3855      ;JUST PRINTED
3856
3857 025446 020227 010300      CMP      R2,#EVTLOG      ;POINTING TO TOP OF EVNT LOG QUEUE?
3858 025452 001010      BNE      RPT1      ;BR IF NO
3859 025454 012702 011334      MOV      #EVTEND,R2      ;SET R2 TO POINT TO BOTTOM OF LOG
3860 025460 026227 177776 177777      CMP      -2(R2),#-1
3861 025466 001007      BNE      RPT0      ;IF END OF LOG IS NOT EMPTY
3862 025470 000137 026606      JMP      ENDEVT      ;CONTINUE...ELSE EXIT
3863
3864 025474 020237 010276      RPT1:  CMP      R2,EVTPTR      ;ARE WE BACK TO POINTER?
3865 025500 001002      BNE      RPT0      ;IF NOT CONTINUE
3866 025502 000137 026606      JMP      ENDEVT      ;IF SO EXIT....
3867
3868 025506 162702 000014      RPT0:  SUB      #14,R2      ;POINT R2 TO START OF ENTRY
3869 025512      RPTAA:  PRINTS      #EVTFO      ;PRINT EVENT ENTRY HEADER
3870 025512 012746 020117      MOV      #EVTFO,-(SP)
3871 025516 012746 000001      MOV      #1,-(SP)
3872 025522 010600      MOV      SP,R0
3873 025524 104416      TRAP      C$PNTS
3874 025526 062706 000004      ADD      #4,SP
3875 025532 112203      MOVB      (R2)+,R3      ;PUT EVENT TYPE INTO R3
3876 025534 112237 011430      MOVB      (R2)+,EVTICK      ;PUT EVENT TIME (TICKS,SECS,MINS IN TEMP LOC.S)
3877 025540 112237 011424      MOVB      (R2)+,EVTSEC
3878 025544 112237 011426      MOVB      (R2)+,EVTMIN
3879 025550      PRINTS      #EVTFF,EVTMIN,EVTSEC,EVTICK,EVTLSI(R3) ;PRINT EVENT TIME AND DESCRIPT.
3880 025550 016346 011374      MOV      EVTLSI(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      #EVTFF,-(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
REFLOG 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,PNTTB ;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
REFLOG 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      EVTMP,-(SP)
                                              MOV      #EVTFS,-(SP)
                                              MOV      #2,-(SP)
                                              MOV      SP,R0
                                              TRAP     ($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     ($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
REPLOG PRINT EVENT LOG

3949	026036			PRIE2:					
3950	026036				PRINTS	#EVT42,<B,DEV1+1>,<B,DEV2>			
3951	026036	005046						CLR	-(SP)
3952	026040	153716	011472					BISB	DEV2,(SP)
3953	026044	005046						CLR	-(SP)
3954	026046	153716	011471					BISB	DEV1+1,(SP)
3955	026052	012746	020472					MOV	#EVT42,-(SP)
3956	026056	012746	000003					MOV	#3,-(SP)
3957	026062	010600						MOV	SP,R0
3958	026064	104416						TRAP	C\$PNTS
3959	026066	062706	000010					ADD	#10,SP
3960	026072	000207			RTS	PC			;RETURN
3961									
3962									
3963	026074			PRIE3:					
3964	026074				PRINTS	#EVT43,<B,DEV1+1>,<B,DEV2>,<B,DEV2+1>			
3965	026074	005046						CLR	-(SP)
3966	026076	153716	011473					BISB	DEV2+1,(SP)
3967	026102	005046						CLR	-(SP)
3968	026104	153716	011472					BISB	DEV2,(SP)
3969	026110	005046						CLR	-(SP)
3970	026112	153716	011471					BISB	DEV1+1,(SP)
3971	026116	012746	020507					MOV	#EVT43,-(SP)
3972	026122	012746	000004					MOV	#4,-(SP)
3973	026126	010600						MOV	SP,R0
3974	026130	104416						TRAP	C\$PNTS
3975	026132	062706	000012					ADD	#12,SP
3976	026136	000207			RTS	PC			;RETURN
3977									
3978									
3979	026140			PRIE4:					
3980	026140				PRINTS	#EVT40,<B,DEV1+1>			
3981	026140	005046						CLR	-(SP)
3982	026142	153716	011471					BISB	DEV1+1,(SP)
3983	026146	012746	020446					MOV	#EVT40,-(SP)
3984	026152	012746	000002					MOV	#2,-(SP)
3985	026156	010600						MOV	SP,R0
3986	026160	104416						TRAP	C\$PNTS
3987	026162	062706	000006					ADD	#6,SP
3988	026166	000207			RTS	PC			

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 115
REFLOG 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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 116
REPLOG 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)+,EVTIMP
MOV (R2)+,TEMP5 ;DUMMY MOVE

;PRINT PASCOUNT ERROR COUNT RX THRES AND TX TTHRES

PRINTS #EVTf4B,EVTADD,EVTBCT,EVTIMP

MOV EVTIMP,-(SP)
MOV EVTBCT,-(SP)
MOV EVTADD,-(SP)
MOV #EVTf4B,-(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)+,EVTIMP ;STORE TOTAL # OF CMP ERRORS
JSR PC,PNTTRB ;PRINT LINE NO.
PRINTS #EVTf4,EVTADD,EVTBCT,EVTIMP ;PRINT ADDR, BYTE CNT, # CMP ERRS

MOV EVTIMP,-(SP)
MOV EVTBCT,-(SP)
MOV EVTADD,-(SP)
MOV #EVTf4,-(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			RPTDLE:			
4075	026526	012237	011432	RPTDCK:	MOV	(R2)+,EVTADD	;STORE MSG ADDR FOR PRINT
4076	026532	012237	011434		MOV	(R2)+,EVTBCT	;STORE BYTE COUNT
4077	026536	012237	011436		MOV	(R2)+,EVTIMP	;STORE BYTE COUNT COMP
4078	026542	004737	026616		JSR	PC,PNTTRB	;PRINT LINE NO.
4079	026546				PRINTS	#EVT4A,EVTADD,EVTBCT,EVTIMP	;PRINT ADDR,RXBYTES,CMPBYTES.
4080	026546	013746	011436				MOV
4081	026552	013746	011434				MOV
4082	026556	013746	011432				MOV
4083	026562	012746	020761				MOV
4084	026566	012746	000004				MOV
4085	026572	010500					MOV
4086	026574	104416					MOV
4087	026576	062706	000012				TRAP
4088							ADD
4089	026602	000137	025442		JMP	RPT	;THEN GO GET NEXT EVENT ENTRY
4090							
4091							
4092							
4093	026606			ENDEVT:			;RETURN TO CALLER AFTER REG RESTORE
4094	026606	012604			MOV	(SP)+,R4	;RESTORE R4,R3,R2
4095	026610	012603			MOV	(SP)+,R3	
4096	026612	012602			MOV	(SP)+,R2	
4097	026614	000207			RTS	PC	;RETURN TO CALLING ROUTINE
4098							
4099							;PRINT LINE NO.
4100							
4101	026616	012237	010176	PNTTRB:	MOV	(R2)+,TEMP5	
4102	026622				PRINTS	#EVT6,TEMP5	;PRINT LINE NUMBER.
4103	026622	013746	010176				MOV
4104	026626	012746	020546				MOV
4105	026632	012746	000002				MOV
4106	026636	010600					MOV
4107	026640	104416					TRAP
4108	026642	062706	000006				ADD
4109	026646	000207			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                                     .SBTTL      CLI FOR REPORT CODING SECTION
4111                                     CLIRAC:
4112 026650                                ASL      R2
4113 026652 006302                        MOV      10$(R2),R2      ;FORM ADDRESS OF ACTION ROUTINE
4114 026656 016202 026666                ADD      #10$,R2
4115 026662 062702 026666                JSR      PC,(R2)
4116 026664 004712                        RTS      PC
4117 026664 000207
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$      ;RDMPS
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 .SBTTL REPORT COMMAND ACTION ROUTINES
4130 026710 112737 177777 003400 ACTRNF: MOVB #-1,P$NNUF ;SET FLAG TO SAY MORE NEEDED
4131 026716 000207 ACTRNL: RTS PC
4132 026720 012702 003276 ACTRHL: MOV #RHLPTB,R2 ;SETUP R2 AS A POINTER TO HELP MSG TABLE
4133 026724 012246 1$: PRINTF #HLPF,(R2)+ ;PRINT HELP INFORMATION MESSAGES
4134 026724 012746 014616 MOV (R2)+,-(SP)
4135 026726 012746 000002 MOV #HLPF,-(SP)
4136 026732 012746 000002 MOV #2,-(SP)
4137 026736 010600 MOV SP,R0
4138 026740 104417 TRAP C$PNTF
4139 026742 062706 000006 ADD #6,SP
4140 026746 020227 003304 CMP R2,#RHLPEN ;SEE IF ALL INFO PRINTED YET
4141 026752 001364 BNE 1$ ;IF NO KEEP PRINTING
4142 026754 012737 000001 003246 MOV #RPHLP,KEYWD1
4143 026762 000207 RTS PC
4144
4145 026764 012737 000002 003246 ACTREX: MOV #RPEXT,KEYWD1 ;SET UP EXIT WORD
4146 026772 000207 RTS PC
4147
4148 026774 004737 025374 ACTRLG: JSR PC,REPLOG ;GO REPORT DCLT EVENT LOG
4149 027000 012737 000003 003246 MOV #RPLOG,KEYWD1
4150 027006 000207 RTS PC
4151
4152 027010 ACTRSL:
4153 027010 012737 000004 003246 MOV #RPSOL,KEYWD1 ;SET UP KEY WORD
4154 027016 105037 003400 CLR P$NNUF ;CLEAR NOT ENOUGH FLAG
4155 027022 000207 RTS PC ;AND RETURN
4156
4157 027024 013737 003374 010126 ACTRDS: MOV P$NUM,STADD ;SETUP STARTING ADDRESS FOR DUMP
4158 027032 005037 010132 CLR BYTBIT ;SET DEFAULT OF WORD DUMP
4159 027036 012737 000006 003246 MOV #RDMPS,KEYWD1 ;FLAG THAT A DUMP WAS TYPED
4160 027044 000403 BR ACTRDE
4161
4162 027046 012737 177777 010132 ACTRDO: MOV #-1,BYTBIT ;SET DUMP FLAG TO 'DUMP-BYTE'
4163 027054 013737 003374 010130 ACTRDE: MOV P$NUM,ENADD ;SETUP END ADDRESS FOR DUMP (=START IF NO 'EEE'
4164 027062 105037 003400 ACTRDX: CLR P$NNUF ;CLEAR NOT-ENOUGH FLAG, 'DUMP N-N/B' IS VALID
4165 027066 000207 RTS PC
4166
4167
4168

```

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

MACV11 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      CLI      CLIEXI,0              ;EXIT
4174 027102  R11$:   CLI      CLISTR,RPHLP,R12$,<'HELP'>
4175 027116      CLI      CLIEXI,0
4176 027120  R12$:   CLI      CLISTR,RPEXT,R13$,<'EXIT'>
4177 027134      CLI      CLIEXI,0
4178 027136  R13$:   CLI      CLISTR,RPSOL,R14$,<'STATUSONLINE'>
4179 027162      CLI      CLIEXI,0
4180 027164  R14$:   CLI      CLISTR,RPLOG,R46$,<'LOG'>
4181 027176      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      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$:   CLI      CLIOCT,RDMPS,R30$
4192 027234      CLI      <'-'>,RNOTNF,R125$
4193 027240      CLI      CLIOCT,RDMPE,R30$
4194 027244      CLI      <'/'>,RNOTNF,R125$
4195 027250      CLI      <'B'>,RDMPQ,R30$
4196 027254      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

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
PRINTF #BASM2,(R2)+ ;PRINT WORD

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

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

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

MACV11 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      .SBTTL      UPDATE  TAL CHAR. COUNT SUBROUTINE
4271
4272      :++
4273      : FUNCTIONAL DESCRIPTION:
4274      :     UPDATES TOTAL CHAR. COUNT TOTCC BASED ON CURCC.
4275      :     LAST MESSAGE IS TRUNCATED TO FIT INTO THE
4276      :     BUFFER IF TOTAL CHAR. COUNT EXCEEDS 'BUFLIM' A MESSAGE
4277      :     IS PRINTED TELLING THE OPERATOR THE TRUNCATION OCCURRED.
4278
4279      : INPUTS:
4280      :     CURCC= CHAR. COUNT OF MESSAGE BEING ADDED
4281      :     TOTCC= TOTAL CHAR COUNT OF BUFFER ITS BEING ADDED TO
4282
4283      : OUTPUTS:
4284      :     MESSAGE TO OPERATOR IF MESSAGE TRUNCATED TO FIT
4285
4286      : FUNCTIONAL SIDE EFFECTS:
4287      :     LOCATION "TEMP" USED FOR CALCULATIONS
4288
4289      : CALLING SEQUENCE:
4290      :     JSR      PC,ADDCC      ;UPDATED TOTAL CHAR. COUNT
4291      :--
4292
4293      027422 063737 010150 010160 ADDCC: ADD    CURCC,TOTCC      ;ADD CURRENT TO TOTAL
4294      027430 022737 001000 010160      CMP    #BUFLIM,TOTCC    ; COMPARE TO 'BUFLIM'
4295      027436 103027      BHIS   ADDC1      ;IF NOT MORE THEN 'BUFLIM' EXIT
4296
4297      ; PRINT MESSAGE AND TRUNCATE COUNT
4298
4299      PRINTF #MSGTRU
4300
4301      027440 012746 017105      MOV    #MSGTRU,-(SP)
4302      027444 012746 000001      MOV    #1,-(SP)
4303      027450 010600      MOV    SP,R0
4304      027452 104417      TRAP   C$PNTF
4305      027454 062706 000004      ADD    #4,SP
4306
4307      027460 163737 010150 010160      SLB    CURCC,TOTCC      ;SUB CURRENT FROM TOTAL
4308      027466 012737 001000 010164      MOV    #BUFLIM,TEMP    ;MOV 'BUFLIM' TO TEMP
4309      027474 163737 010160 010164      SUB    TOTCC,TEMP      ;SUB TOTAL FROM 'BUFLIM'
4310      027502 013737 010164 010150      MOV    TEMP,CURCC      ;AND ESTABLISH NEW CURRENT
4311      027510 063737 010150 010160      ADD    CURCC,TOTCC    ;ADD "ADJUSTED CURRENT" TO TOTAL CHAR. CNT.
4312      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 :FUNCTIONAL DESCRIPTION:
4316 :BLDBUF-- BUILD POINTER TABLE AND BUFFERS
4317
4318 :THIS SUBROUTINE ADDS A MESSAGE TO THE TRANSMIT OR EXPECT LIST
4319 :USING THE POINTER, BYTE COUNT, AND ADDRESS PASSED TO IT.
4320
4321 :INPUTS:
4322 :CURCC= CHAR. COUNT OF MESSAGE TO BE ADDED
4323 :CURADD= ADDRESS OF MESSAGE TO BE ADDED
4324 :CPTR= ADDRESS OF POINTER TABLE WORD WHERE MESSAGE POINTERS ARE
4325 :TO BE BUILT
4326 :MSGTYP= VALUE TO USE AS AN INDEX TO FIND SOURCE OF MESSAGE DATA
4327 :INDEX INTO DMSGCT() AND DMSGAD().
4328
4329 :OUTPUTS:
4330 :A MESSAGE ADDED TO EITHER TXBUF OR CMPBUF
4331 :APPROPRIATE POINTERS IN PTRTAB POINTER TABLE
4332
4333 :CALLING SEQUENCE:
4334 :JSR PC,BLDBUF ;BUILD MESSAGE IN BUFFER AND ADD PTRS.
4335 :--
4336
4336 027520 BLDBUF: MOV R2,-(SP) ;SAVE R2 AND R3 ON THE STACK
4337 027520 010246 MOV R3,-(SP)
4338 027522 010346 MOV CPTR,R2
4339 027524 013702 010154
4340
4341 027530 013722 010156 BLDB1: MOV CURADD,(R2)+ ;PUT CURRENT ADD ON POINTER TAB
4342 027534 013722 010150 MOV CURCC,(R2)+ ;PUT CURRENT CC ON POINTER TAB
4343 027540 010237 010154 MOV R2,CPTR ;PUT UPDATED R2 BACK TO CURRENT POINT
4344 027544 013702 010146 MOV MSGTYP,R2 ;GET MESSAGE TYPE TO USE AS INDEX
4345 027550 006302 ASL R2 ;DOUBLE FOR WORD INDEX
4346 027552 013737 010156 010164 MOV CURADD,TEMP ;MOVE CURRENT ADD TO TEMP
4347 027560 063737 010150 010164 ADD CURCC,TEMP ;ADD CHAR COUNT TO IT TO GET END
4348 027566 013703 010156 MOV CURADD,R3 ;SET R3 TO CURRENT START ADD
4349 027572 016237 002144 010170 BLDB2: MOV DMSGCT(R2),TEMP2 ;GET BYTE COUNT
4350 027600 016204 002172 MOV DMSGAD(R2),R4 ;PUT STARTING FROM ADD IN R4
4351 027604 060437 010170 ADD R4,TEMP2 ;ADD IT TO TEMP2 TO GET END OF FROM
4352 027610 112423 BLDB3: MOVB (R4)+,(R3)+ ;MOV BYTE FROM PATTERN TO BUFFER
4353 027612 020337 010164 CMP R3,TEMP ;ALL DONE?
4354 027616 001404 BEQ BLDBEX ;IF SO EXIT
4355 027620 020437 010170 CMP R4,TEMP2 ;IS PATTERN COUNT EXPIRED
4356 027624 001762 BEQ BLDB2 ;IF SO GO START AGAIN
4357 027626 000770 BR BLDB3 ;IF NOT GET ANOTHER BYTE
4358 027630 063737 010150 010156 BLDBEX: ADD CURCC,CURADD ;BUMP CURADD
4359 027636 012603 MOV (SP)+,R3 ;RESTORE R3 AND R2
4360 027640 012602 MOV (SP)+,R2
4361 027642 000207 RTS PC ;RETURN TO CALLER
4362

```

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

SEQ 124

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
:--

```

```

:CHECK CURRENT ADDRESS
FCURAD:  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

```

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

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

.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
MSGLIN = 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

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

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 127
CREATE FACSIMILE OF TX BUFFER AND MESSAGE LIST

```

4440
4441 027672          FACSIMILE:
4442 027672 010146      MOV    R1,-(SP)          ;SAVE R1
4443 027674 005001      CLR    R1              ;INIT R1
4444 027676 116161 003406 004406 10$:  MOVB   TXBUF(R1),CMPBUF(R1)  ;COPY TX BUFFER TO EXPECTED BUFFER
4445 027704 005201      INC    R1              ;BUMP INDEX
4446 027706 020127 001000      CMP    R1,#BUFLIM  ;ALL DATA COPIED ?
4447 027712 001371      BNE    10$            ;NO,BRANCH
4448
4449 027714 012701 000017      20$:  MOV    #MSGLIM,R1          ;MESSAGE LIMIT
4450 027720 006301      ASL    R1              ;MULTIPLY BY 2
4451 027722 006301      ASL    R1              ;MULTIPLY BY 2
4452 027724 012737 006406 010054      MOV    #PTRTAB,CMPPTR      ;TOP OF POINTER TABLE
4453 027732 060137 010054      ADD    R1,CMPPTR      ;START OF EXPECTED POINTER TABLE
4454 027736 005001      CLR    R1              ;INIT R1
4455
4456      ;SET UP WHILE - DO LOOP
4457 027740 012737 006406 010052      MOV    #PTRTAB,TXPTR      ;TX POINTER NOW AT TOP OF TABLE
4458 027746 012737 004406 010062      MOV    #CMPBUF,CCURAD      ;TRANSFER ADDRESS OF 1ST MESSAGE
4459 027754 062737 000002 010052      ADD    #2,TXPTR          ;BUMP POINTER
4460 027762 017737 160054 010060      MOV    @TXPTR,CTOTCC      ;BYTE COUNTER 1ST MESSAGE
4461 027770 005037 010056      CLR    CMPTOT      ;INIT EXPECTED MESSAGE COUNT
4462
4463      ;WHILE TX MESSAGE TOTAL <> EXPECTED MESSAGE TOTAL DO
4464 027774 023737 010074 010056 30$:  CMP    TXMTOT,CMPTOT      ;ALL MESSAGES COPIED ?
4465 030002 001436      BEQ    40$            ;YES,BRANCH
4466 030004 013777 010062 160042      MOV    CCURAD,@CMPPTR      ;TRANSFER ADDRESS OF MESSAGE
4467 030012 062737 000002 010054      ADD    #2,CMPPTR          ;BUMP POINTER
4468 030020 013777 010060 160026      MOV    CTOTCC,@CMPPTR      ;BYTE COUNT OF MESSAGE
4469 030026 062737 000004 010052      ADD    #4,TXPTR          ;BUMP TX MESSAGE POINTER
4470 030034 063737 010060 010062      ADD    CTOTCC,CCURAD      ;CALC. TRANSFER ADDRESS
4471      ;ADJUST ODD ADDRESS
4472 030042 032737 000001 010062      BIT    #1,CCURAD          ;IS COMPUTED ADDRESS ODD ?
4473 030050 001402      BEQ    35$            ;NO. OMIT ADJUSTMENT
4474 030052 005237 010062      INC    CCURAD      ;YES. INCREMENT ADDRESS
4475 030056      35$:
4476 030056 017737 157770 010060      MOV    @TXPTR,CTOTCC      ;BYTE COUNT NEXT MESSAGE
4477 030064 062737 000002 010054      ADD    #2,CMPPTR          ;BUMP POINTER
4478 030072 005237 010056      INC    CMPTOT      ;INCREMENT MESSAGE COUNT
4479 030076 000736      BR     30$            ;DO IT AGAIN
4480      ;END WHILE - DO
4481 030100 013737 010076 010060 40$:  MOV    TTOTCC,CTOTCC      ;COPY TOTAL CHARACTER COUNT
4482
4483      ;END ROUTINE
4484 030106 012601      MOV    (SP)+,R1          ;RESTORE R1
4485 030110 000207      RTS    PC              ;RETURN
4486
4487

```

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 128
QUEUE UP ALL REC BUFFERS FOR MULTIPOINT

```

:++
: FUNCTIONAL DESCRIPTION:      RXQUAL - QUEUE ALL REC BUFFERS

```

: 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

```

• —

4510							
4511							
4512	030112	012737	177777	007642	RXQUAL: MOV	#-1,INDEX	:SET INDEX TO -1
4513	030120	004737	030402		RXQU1: JSR	PC,GTVIND	:GET NEXT VALID INDEX
4514	030124	022737	000010	007642	CMP	#8.,INDEX	:IS ALL DONE
4515	030132	001411			BEQ	RXQUEX	:IF SO EXIT
4516	030134	004737	030454		JSR	PC,ULRPLS	:LOAD CPTRR FOR THIS LINE
4517	030140	052737	000004	010230	BIS	#QRX,FLAG	:INDICATE RX QUEUED
4518	030146	004737	031142		JSR	PC,LOGAQR	
4519	030152	004737	030434		JSR	PC,LDRPLS	:RELOAD RX PTR LIST
4520	030156	000207			RXQUEX: RTS	PC	:RETURN TO CALLER
4521							

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	:IS IT 8?
	#8,INDEX	:BRANCH IF 8.
	BEQ	:IF NOT LOAD CPTRLS FOR THIS LINE.
	JSR	:RETURN TO CALLER WHEN DONE WITH ALL.
	PC,LCPRL1	
LCPREX: RTS	PC	

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

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

```

4557      .SBTTL      LOAD CPTRLS AND DVRCLS FROM INDEX
4558
4559      :++
4560      : FUNCTIONAL DESCRIPTION:      LCPRL1 - LOAD CPTRLS AND DVRCLS FROM INDEX
4561      :
4562      :      THIS ROUTINE LOADS UP THE CPTRLS LIST FOR THE
4563      :      INDEX VALUE AND THE DVRCLS IS LOADED WITH RXMTOT.
4564      :
4565      : INPUTS:      RXMTOT - TOTAL NUMBER OF RX MSGS PER LINE
4566      :      PTR23 - START OF RX POINTER TABLE
4567      :
4568      : OUTPUTS:     CPTRLS - LOADED WITH POINTERS TO THE RXPTR LIST
4569      :      DVRCLS - LOADED WITH RXMTOT COUNT
4570      :
4571      : SUBORDINATE ROUTINES USED:
4572      :      MTPLY - MULTIPLIES VALUE IN INDEX BY VALUE IN
4573      :      TEMP1 AND THEN ADDS THAT RESULT TO VALUE
4574      :      IN TEMP2 AND PUTS FINAL RESULT IN TEMP2
4575      :
4576      : CALLING SEQUENCE:
4577      :      JSR      PC,LCPRL1
4578      :--
4579
4580      030210 012737 006576 010170 LCPRL1: MOV      #PTR23,TEMP2 ;SET UP TEMP 2 AS BASE
4581      030216 013737 007642 010046      MOV      INDEX,MPLY ;SET UP MULTIPLIER
4582      030224 012737 000074 010164      MOV      #60.,TEMP ;SET UP MULTIPLICAN
4583      030232 004737 030356      JSR      PC,MTPLY ;GO MULTIPY
4584      030236 013703 007642      LCPRL2: MOV      INDEX,R3
4585      030242 113763 010112 007602      MOVB     RXMTOT,DVRCLS(R3) ;LOAD UP COUNT LIST
4586      030250 006303      ASL      R3 ;MAKE R3 WORD INDEX
4587      030252 013763 010170 007542      MOV      TEMP2,CPTRLS(R3) ;SET UP POINTER TABLE
4588      030260 000207      RTS      PC ;RETURN TO CALLER
4589
4590

```

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      .SBTTL      CLEAR RECEIVE POINTER LIST
4592
4593      :++
4594      : FUNCTIONAL DESCRIPTION:      CLRPLS - CLEAR RX POINTER LIST
4595      : THIS ROUTINE CLEARS ALL 8 SLOTS OF THE CTPRLS
4596      :
4597      : OUTPUTS:      CPTRLS - IS ZEROED IN ALL SLOTS
4598      :
4599      : CALLING SEQUENCE:
4600      : JSR      PC,CLRPLS
4601      :--
4602
4603      030262 012737 000010 010164 CLRPLS: MOV      #8,TEMP
4604      030270 012703 007542      MOV      #CPTRLS,R3      ;LOAD START OF LIST TO R3
4605      030274 005023      CLRPL1: CLR      (R3)+      ;CLEAR THIS SLOT
4606      030276 005337 010164      DEC      TEMP
4607      030302 001374      BNE      CLRPL1      ;IF NOT DONE GO BACK
4608      030304 000207      CLRPEX: RTS      PC      ;RETURN TO CALLER WHEN DONE
4609
4610
4611

```

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 .SBTTL          LOAD TX POINTER LIST INITIALLY
4613
4614 :++
4615 : FUNCTIONAL DESCRIPTION:      LCPTLS - LOAD TRANSMIT POINTER LIST
4616 : THIS ROUTINE LOADS CPTLS WITH TX POINTERS
4617 : FOR EACH VALID LINE.
4618 :
4619 : INPUTS:
4620 :       TXMTOT - TOTAL NUMBER OF TX MSGS
4621 :       PTRTAB - POINTER TO TOP OF TX POINTER TABLE
4622 :
4623 : OUTPUTS:
4624 :       CCTLS - LOADED WITH POINTERS TO TX POINTER TABLE
4625 :               FOR ALL VALID LINES
4626 :       DVTCLS - TX MSG COUNT LIST LOADED WITH MSG COUNTS
4627 :               FOR ALL VALID LINES
4628 :
4629 : SUBORDINATE ROUTINES USED:
4630 :       GTVIND - GETS NEXT VALID INDEX BY
4631 :               CHECKING TRIBLS FOR NON ZERO ENTRY
4632 :
4633 :       LDTPLS - LOADS VALUE FROM CPTR TO CPTLS INDEXED
4634 :               BY TRIBN
4635 :
4636 :       LDTCLS - LOADS DVTCT TO DVTCLS INDEXED BY TRIBN
4637 :
4638 : CALLING SEQUENCE:
4639 :       JSR      PC,LCPTLS
4640 :
4641 :
4642 :
4643 :
4644 :
4645 :
4646 :
4647 :
4648 :
4649 :

```

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      .SBTTL      MULTIPY
4651
4652      : **
4653      : FUNCTIONAL DESCRIPTION:      MTPLY= MULTIPY
4654      : THIS ROUTINE MULTIPLIES THE VALUE IN MPLY BY
4655      : THE VALUE IN TEMP AND THEN ADDS IN THE VALUE OF TEMP2
4656      : WITH THE RELSUT GOING TO TEMP2
4657
4658      : INPUTS:      TEMP2 - INITIALLY VALUE
4659      :                TEMP  - VALUE TO MULTIPLY BY
4660      :                MPLY  - NUMBER OF TIMES TO MULITPLY
4661
4662      : OUTPUTS:      TEMP2 - RESULT OF [MPLY * TEMP]+TEMP2
4663
4664      : CALLING SEQUENCE:
4665      :                JSR      PC,MTPLY
4666      : --
4667
4668      030356 005737 010046      MTPLY:  TST      MPLY
4669      030362 001406              BEQ      MTPLEX
4670      030364 063737 010164 010170      ADD      TEMP,TEMP2
4671      030372 005337 010046              DEC      MPLY
4672      030376 000767              BR       MTPLY
4673      030400 000207      MTPLEX: RTS      PC
4674

```

:IF MULITPLIER IS ZERO QUIT
:ADD THE FACTOR TO BASE
:COUNT DOWN THE MULTIPLIER
:GO BACK FOR MORE
: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
..--

GTVIND: MOV INDEX,R3
GTVI1: INC R3
MOV 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

030402 013703 007642
030406 005203
030410 116337 007622 007636
030416 123727 007636 000377
030424 001770
030426 010337 007642
030432 000207

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 135
LOAD REC POINTER LIST

```

4700      .SBTTL      LOAD REC POINTER LIST
4701      :
4702      : **
4703      : FUNCTIONAL DESCRIPTION:      LDRPLS - LOAD RX POINTER LIST FROM CPTRR
4704      : THIS ROUTINE MOVES DATA FROM CPTRR TO THE SLOT IN THE
4705      : CPTRLS INDEXED BY INDW.
4706      : INPUTS:      INDW - WORD INDEX INTO LIST
4707      : OUTPUTS:      CPTRLS - CORRECT SLOT LOADED WITH DATA FROM CPTRR
4708      : SUBORDINATE ROUTINES USED:
4709      : GETIND - GETS INDW FOR THIS TRIBN
4710      : CALLING SEQUENCE
4711      : JSR      PC,LDRPLS
4712      : --
4713      LDRPLS: JSR      PC,GETIND      ;GET INDW FOR THIS TRIBN
4714      MOV      INDW,R3      ;MOVE WORD INDEX TO R3
4715      MOV      CPTRR,CPTRLS(R3)      ;LOAD CPTRLS LIST
4716      RTS      PC      ;RETURN TO CALLER
4717
4718      .SBTTL      UNLOAD CPTRR LIST
4719      :
4720      : **
4721      : FUNCTIONAL DESCRIPTION:      ULRPLS - UNLOAD RX POINTER LIST
4722      : THIS ROUTINE MOVES DATA FROM CPTRLS SLOT INDEXED
4723      : BY INDW TO CPTRR.
4724      : IMPLICIT INPUTS:
4725      : TRIBN - NUMBER OF CURRENT LINE
4726      : CPTRR - VALUE FROM CPTRLS
4727      : OUTPUTS:
4728      : SUBORDINATE ROUTINES USED:
4729      : GETIND - GET INDW FOR THIS TRIBN
4730      : CALLING SEQUENCE:
4731      : JSR      PC,ULRPLS
4732      : --
4733      ULRPLS: JSR      PC,GETIND      ;GET INDEX
4734      MOV      INDW,R3      ;MOVE WORD INDEX TO R3
4735      MOV      CPTRLS(R3),CPTRR      ;LOAD CPTRR FROM LIST INDEX
4736      RTS      PC      ;RETURN TO CALLER

```

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

MACV11 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:
: MTPY - MULTIPLIES INDEX BY TEMP AND ADDS TEMP2 TO RESULT
: CALLING SEQUENCE:
: JSR PC,GRPTCP
: --

```

```

GRPTCP: MOV INDEX,MPLY ;SET UP MULPILIER
        MOV #60.,TEMP
        MOV RXPTR,TEMP2
        JSR PC,MTPY ;[INDEX VALUE X 60.] + RXPTR = POINTER ADDRESS
        MOV TEMP2,CPTR ;SET UP POINTER ADDR.
        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      :               DVRCT - COUNT VALUE TO GO TO LIST
4766      : OUTPUTS:      DVRCLS - VALUE OF DVRCT
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 DVRCT
4785      : OUTPUTS:      DVRCT - 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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 138
LOAD CPTR LIST (TRANSMIT POINTER)

```

4797      .SBTTL      LOAD CPTR LIST (TRANSMIT POINTER)
4798
4799      : **
4800      : FUNCTIONAL DESCRIPTION:      LDTPLS - LOAD TX POINTER LIST
4801      : THIS ROUTINE LOADS THE VALUE FROM CPTR TO
4802      : THE TX POINTER LIST INDEXED BY TRIBN INDEX.
4803      : INPUTS:      TRIBN - NUMBER OF LINE IN USE
4804      : OUTPUTS:     CPTILS - SLOT LOADED WITH CPTR DATA
4805      : SUBORDINATE ROUTINES USED:
4806      : GETIND - GET INDEX VALUE FROM TRIBLS
4807      : CALLING SEQUENCE:
4808      : JSR      PC,LDTPLS
4809      : --
4810
4811      LDTPLS: JSR      PC,GETIND      ;GET INDEX
4812      MOV      INDW,R3      ;MOVE INDEX TO R3
4813      MOV      CPTR,CPTILS(R3) ;LOAD LIST
4814      RTS      PC      ;RETURN TO CALLER
4815
4816      .SBTTL      UNLOAD CPTR LIST (TRANSMIT POINTER)
4817
4818      : **
4819      : FUNCTIONAL DESCRIPTION:      ULTPLS - UNLOAD TX POINTER LIST
4820      : THIS ROUTINE MOVES DATA FROM TX POINTER LIST
4821      : TO CPTR.
4822      : INPUTS:      TRIBN - NUMBER OF LINE IN USE
4823      : OUTPUTS:     CPTR - VALUE FROM THE TX POINTER LIST
4824      : SUBORDINATE ROUTINES USED:
4825      : GETIND - GET INDEX FROM TRIBLS
4826      : CALLING SEQUENCE:
4827      : JSR      PC,ULTPLS
4828      : --
4829
4830      ULTPLS: JSR      PC,GETIND      ;GET INDEX
4831      MOV      INDW,R3      ;MOVE WORD INDEX TO R3
4832      MOV      CPTILS(R3),CPTR ;GET PTR FROM LIST
4833      RTS      PC      ;RETURN TO CALLER
4834

```

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 .SBTTL LOAD DVTCT LIST (TRANSMIT COUNT)
4836
4837 : **
4838 : FUNCTIONAL DESCRIPTION: LDTCLS - LOAD TX COUNT LIST
4839 : THIS ROUTINE LOADS A VALUE FROM DVTCT TO
4840 : THE TX COUNT LIST (DVTCLS). INDEXED BY TRIBN.
4841
4842 : INPUTS: TRIBN - NUMBER OF LINE IN USE
4843 : DVTCT - CURRENT TX COUNT FOR LINE
4844 : OUTPUTS: DVTCLS - SLOT LOADED WITH DVTCT
4845 : SUBORDINATE ROUTINES USED:
4846 : GETIND - GET INDEX FROM TRIBLS
4847 : CALLING SEQUENCE:
4848 : JSR PC,LDTCLS
4849 : --
4850
4851 030632 004737 031102 LDTCLS: JSR PC,GETIND ;GET INDEX
4852 030636 013703 007642 MOV INDEX,R3 ;MOVE BYTE INDEX TO R3
4853 030642 113763 010072 007612 MOVB DVTCT,DVTCLS(R3);LOAD LIST
4854 030650 000207 RTS PC ;RETURN TO CALLER
4855
4856 .SBTTL UNLOAD DVTCT LIST (TX COUNT)
4857
4858 : **
4859 : FUNCTIONAL DESCRIPTION: ULTCLS - UNLOAD TX COUNT LIST
4860 : THIS ROUTINE TAKES DATA FROM DVTCLS AND MOVES
4861 : IT TO DVTCT
4862 : INPUTS: TRIBN - NUMBER OF LINE IN USE
4863 : OUTPUTS: DVTCT - TRANSMIT COUNT VALUE
4864 : SUBORDINATE ROUTINES USED:
4865 : GETIND - GET INDEX VALUE FROM TRIBLS
4866 : CALLING SEQUENCE:
4867 : JSR PC,ULTCLS
4868 : --
4869
4869 030652 004737 031102 ULTCLS: JSR PC,GETIND ;GET INDEX
4870 030656 013703 007642 MOV INDEX,R3 ;MOVE BYTE INDEX TO R3
4871 030662 116337 007612 010072 MOVB DVTCLS(R3),DVTCT
4872 030670 000207 RTS PC ;RETURN TO CALLER
4873
4874

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 140
GET ALL RX POINTERS FROM LIST TO CPTRR

```

4875      .SBTTL      GET ALL RX POINTERS FROM LIST TO CPTRR
4876
4877      : **
4878      : FUNCTIONAL DESCRIPTION:      GARPFL - GET ALL RX POINTERS FROM LIST
4879      : THIS ROUTINE CHECKS ALL RX POINTERS FOR VALID LINES
4880      : IN CPTRLS AND MAKES SURE THEY ARE ALL ZERO.
4881      : OUTPUTS:      CPTRR - ZERO IF ALL CPTRLS IS ZERO
4882      : NON ZERO IF NOT.
4883      :
4884      : SUBORDINATE ROUTINES USED:
4885      : GTVIND - GET VALID INDEX
4886      : ULRPLS - UNLOAD CPTRR LIST TO CPTRR
4887      : CALLING SEQUENCE:
4888      : JSR      PC,GARPFL
4889      : --
4890
4891      030672 013737 007636 010176 GARPFL: MOV      TRIBN,TEMP5
4892      030700 012737 177777 007642      MOV      #-1,INDEX
4893      030706 004737 030402      GARP1: JSR      PC,GTVIND      ;GET VALID INDEX
4894      030712 022737 000010 007642      CMP      #8.,INDEX      ;COMPARE INDEX
4895      030720 001405      BEQ      GARPEX      ;EXIT IF DONE
4896      030722 004737 030454      JSR      PC,ULRPLS      ;LOAD CPTRR WITH VALUE
4897      030726 005737 010152      TST      CPTRR      ;TEST THE VALUE
4898      030732 001765      BEQ      GARP1      ;IF ZERO CHECK NEXT
4899      030734 013737 010176 007636 GARPEX: MOV      TEMP5,TRIBN
4900      030742 000207      RTS      PC      ;RETURN TO CALLER WHEN DONE
4901

```

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      .SBTTL      GET ALL TX COUNTS FROM LIST TO DVTCT
4903      :RETURN WITH DVTCT=1 IF ANY COUNT HAS SOME IN IT
4904      :IF ALL COUNTS ARE ZERO EXIT
4905
4906      :**
4907      : FUNCTIONAL DESCRIPTION:      GATCFL - GET ALL TX COUNTS FROM LIST
4908      : THIS ROUTINE GETS AND CHECKS ALL TX COUNTS TO BE ZERO
4909      : OUTPUTS:      DVTCT - ZERO IF LIST IS ZERO
4910      : NON ZERO IF NOT
4911      : SUBORDINATE ROUTINES USED:
4912      : GTVIND - GET NEXT VALID INDEX
4913      : CALLING SEQUENCE:
4914      : JSR      PC,GATCFL
4915      :--
4916 030744 013737 007636 010176 GATCFL: MOV      TRIBN,TEMPS
4917 030752 012737 177777 007642      MOV      #-1,INDEX
4918 030760 005037 010072      CLR      DVTCT      ;CLEAR COUNT
4919 030764 004737 030402      GATC1: JSR      PC,GTVIND      ;GET VALID INDEX
4920 030770 022737 000010 007642      CMP      #8,,INDEX      ;IS INDEX =8 ALL DONE
4921 030776 001410      BEQ      GATCEX      ;IF SO EXIT
4922 031000 013703 007642      MOV      INDEX,R3
4923 031004 105763 007612      TSTB     DVTCLS(R3)      ;IS THIS COUNT 0
4924 031010 001765      BEQ      GATC1      ;IF THIS ONE IS ZERO
4925 031012 012737 000001 010072      MOV      #01,DVTCT      ;LOAD COUNT WITH A 1
4926 031020 013737 010176 007636 GATCEX: MOV      TEMPS,TRIBN
4927 031026 000207      RTS      PC      ;RETURN TO CALLERR
4928

```

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 142
GET NEXT TX POINTER FROM LIST

```
.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 MSG
: 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      PC             ;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     TRIBL(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      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 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      .SBTTL      SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS
4999
5000      :++
5001      : FUNCTIONAL DESCRIPTION:
5002      : SHWOP - SHOW MODE OF OPERATION, LOOP, QUALIFIERS
5003      : PRINTED ON THE OPERATOR'S CONSOLE.
5004
5005      : INPUTS:
5006      : DEV1=  MODE TYPE (MODTYP)
5007      : DEV2=  MAINT LOOP TYPE (MLTYP)
5008      : DEV3=  'RUN PASS' COUNT (RPASS) - COUNT DOWN
5009      : DEV4=  PARAMETERS WORD (PARAM)
5010
5011      : IMPLICIT INPUTS:
5012      : MODES= TABLE OF ADDRESSES OF MODE NAME STRINGS
5013      : LOOPS= TABLE OF ADDRESSES OF LOOP TYPE NAMES
5014
5015      : CALLING SEQUENCE:
5016      : JSR PC,SHWOP
5017      :--
5018
5019      SHWOP:  MOV     DEV1,R2      ;GET THE MODE TYPE IN R2
5020      ASL     R2                  ;MAKE IT A WORD TABLE OFFSET
5021      MOV     MODES(R2),TEMP      ;GET ADDRESS OF MODE-IN-ASCII
5022      MOV     DEV2,R2            ;GET MAINTENANCE LOOP TYPE
5023      ASL     R2
5024      MOV     #LPO0,TEMP3        ;LOAD TEMP3 TO POINT TO '/LOOP='
5025      TST     R2                  ;SEE IF /LOOP=XXXXX OR NONE
5026      BNE     10$                ;BR IF /LOOP= OF SOME KIND
5027      MOV     #LPO,TEMP3         ;IF NO LOOP THEN DON'T PRINT '/LOOP='
5028      10$:  MOV     LOOPS(R2),TEMP1 ;GET ADDRESS OF LOOP-IN-ASCII
5029      MOV     DEV3,TEMP2        ;GET NUMBER OF PASSES
5030      PRINTS  #SHF0,TEMP,TEMP3,TEMP1,TEMP2
5031
5032      MOV     TEMP2,-(SP)
5033      MOV     TEMP1,-(SP)
5034      MOV     TEMP3,-(SP)
5035      MOV     TEMP,-(SP)
5036      MOV     #SHF0,-(SP)
5037      MOV     #5,-(SP)
5038      MOV     SP,R0
5039      TRAP    C$PNTS
5040      ADD     #14,SP
5041
5042      CLR     R2                  ;NOW SET UP FOR QUALIFIERS IN ASCII
5043      MOV     #PST,TEMP
5044      BIT     #STATB,DEV4        ;SEE IF /STATUS OR /NOSTATUS
5045      BNE     1$                ;BR IF /STATUS
5046      MOV     #PNST,TEMP
5047      MOV     #PCK,TEMP1
5048      BIT     #DATCKB,DEV4      ;SEE IF /CHECK OR /NOCHECK
5049      BNE     2$                ;BR IF /CHECK
5050      MOV     #PNCK,TEMP1
5051      MOV     #PEC,TEMP2
5052      BIT     #ECHOB,DEV4      ;SEE IF /ECHO OR /NOECHO
5053      BNE     3$                ;BR IF /ECHO
5054      MOV     #PNEC,TEMP2

```

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

MACV11 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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 146
TRAVERSE COMMAND LINE SUBROUTINES

```

5074 .SBTTL TRAVERSE COMMAND LINE SUBROUTINES
5075
5076
5077
5078
5079
5080
5081
5082
5083
5084
5085
5086
5087
5088
5089
5090
5091
5092 031506
5093 031506 013704 003364
5094 031512 013703 003366
5095 031516 105714
5096 031520 001441
5097 031522 121327 000013
5098 031526 003023
5099 031530 111305
5100 031532 006305
5101 031534 016505 031550
5102 031540 062705 031550
5103 031544 004715
5104 031546 000763
5105
5106 031550 000114
5107 031552 000134
5108 031554 000152
5109 031556 000162
5110 031560 000204
5111 031562 000270
5112 031564 000604
5113 031566 000650
5114 031570 000270
5115 031572 000256
5116 031574 000736
5117
5118
5119 031576 121314
5120 031600 001403
5121 031602 004737 031646
5122 031606 000743
5123 031610 004737 031626
5124 031614 062703 000004
5125
5126 031620 005204
5127 031622 000735
5128
5129 031624 000207

P$TRV SUBROUTINE
: PARSE THE COMMAND LINE SUBROUTINE
: TAKE ACTIONS (VIA ACTION TREE) AS PARSING LINE
: PARSING DIRECTIONS FROM "CLI PARSING NODES"
: REGS USED:
:     P$NUM=NUMERIC CODE FROM DATA
:     R1,R5=SCRATCH
:     R2=ACTION CODE PARAMETER FROM TREE
:     R3=PARSE TREE POINTER
:     R4=INPUT STRING POINTER
: CALLING SEQUENCE:
: JSR PC,P$TRV
: --

P$TRV: MOV P$BUFA,R4
      MOV P$TREE,R3
P$TR5: TSTB (R4)
      BEQ P$EXIT
      CMPB (R3),#11.
      BGT 20$
      MOVB (R3),R5
      ASL R5
      MOV 10$(R5),R5
      ADD #10$,R5
      JSR PC,(R5)
      BR P$TR5

10$: .WORD TRVERR-10$
     .WORD TRVEXI-10$
     .WORD TRVBR-10$
     .WORD TRVBIF-10$
     .WORD TRVSPA-10$
     .WORD TRVNUM-10$
     .WORD TRVALP-10$
     .WORD TRVALN-10$
     .WORD TRVOCT-10$
     .WORD TRVDEC-10$
     .WORD TRVSTR-10$

20$: CMPB (R3),(R4)
     BEQ 22$
     JSR PC,TRVBRC
     BR P$TR5
22$: JSR PC,TRVACT
     ADD #4,R3
     INC R4
     BR P$TR5

P$EXIT: RTS PC

:SEE IF ANY CHARS LEFT IN INPUT STRING
:BR IF NO
:SEE IF SPECIAL CLI CHAR CODE OR ASCII
:BR IF REGULAR ASCII CHAR.
:GET SPECIAL CHAR CODE INTO R5
:BUILD TRAVERSE ROUTINE ADDRESS
:JSR TO SPECIAL CLI TRAVERSE ROUTINE
:GO SEE IF MORE OF STRING LEFT
:TRAVERSE TABLE FOR "CLI FUNCTIONS"
:1
:2
:3
:4
:5
:6
:7
:8
:9
:10
;NOT A SPECIAL CODE
:SEE IF FIRST CHAR OF STRING IS A MATCH
:BR IF A MATCH
:IF NOT A MATCH, GO TAKE MISS BRANCH
: THEN GO BACK PT'G TO MISS NODE
:IF A MATCH, GO DO ACTION DEFINED BY
: ACTION CODE IN CLI NODE, THEN
: ADJUST PTR TO NEXT CLI NODE
:ADJUST BUF PTR TO NEXT CHAR IF MATCH
:RETURN FROM PARSER

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 147
TRAVERSE COMMAND LINE SUBROUTINES

```

5130 ;-----
5131 ;GOTO USER ACTION ROUTINE
5132 TRVACT: MOV 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.,P$RADX ;USE DECIMAL AS RADIX AND ASSUME +
5187 032034 000137 032046 TRVNMA: JMP TRVNMA
5188 032040 TRVOCT: ;(SAME AS TRVNUM SINCE DEFAULT RADIX IS OCTAL)
5189 032040 012737 000010 003376 TRVNUM: MOV #8.,P$RADX ;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 P$RADX ALREADY SAYS +, JUST BR
5194 032060 121427 000055 10$: CMPB (R4),#'- ;SEE IF THERE'S A - SIGN THERE
5195 032064 001004 BNE 1$ ;BR IF NO
5196 032066 112737 177777 003377 MOVB #-1,P$RADX+1 ;SET 'MINUS FLAG' (HI BYTE OF P$RADX)
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 P$RADX,#10. ;SEE IF IN DECIMAL MODE
5204 032120 001417 BEQ 12$ ;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 0'4105 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 P$NUM ;CLEAR LOC. WHERE VALUE WILL BE STORED
5227 032210 112102 3$: MOVB (R1)+,R2 ;GET ASCII CHAR AND CONVERT IT TO A #
5228 032212 162702 000060 SUB #60,R2
5229 032216 006337 003374 ASL P$NUM
5230 032222 103437 BCS 7$ ;SHIFT CURRENT VALUE TO MAKE ROOM
5231 032224 013737 003374 003372 MOV P$NUM,P$CNT ;ERROR IF NUMBER TOO BIG
5232 032232 006337 003374 ASL P$NUM ;SAVE FOR LATER IN CASE DECIMAL RADIX
5233 032236 103431 BCS 7$ ;ERROR IF NUMBER TOO BIG
5234 032240 006337 003374 ASL P$NUM
5235 032244 103426 BCS 7$ ;ERROR IF NUMBER TOO BIG
5236 032246 123727 003376 000012 CMPB P$RADX,#10. ;SEE IF DECIMAL RADIX
5237 032254 001004 BNE 4$ ;BR IF NOT EQUAL
5238 032256 063737 003372 003374 ADD P$CNT,P$NUM
5239 032264 103416 BCS 7$ ;ERROR IF NUMBER TOO BIG
5240 032266 060237 003374 4$: ADD R2,P$NUM

```

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	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	MOVB	#-1,PSGDBD	;SET ERROR RETURN FLAG
5257	032350	000137	031646	5\$:	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.
:--

BGNRPT

LSRPT::

JSR PC,REPORT

:CALL SUBROUTINE TO DUMP EVENT LOG
: AND BASE TABLE OR TO GET LINE STATUS

ENDRPT

L10013: TRAP C\$RPT

5324
5325
5326
5327
5328
5329
5330
5331
5332 032576
5333 032576
5334
5335
5336 032576 004737 024644
5337
5338
5339
5340
5341
5342 032602
5343 032602
5344 032602 104425

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
5353
5354
5355
5356
5357
5358
5359
5360

032604
032604

032604 177777
032606 177777
032610 177777

032612

BGNPROT

-1
-1
-1

ENDPROT

;OFFSET INTO P-TABLE FOR CSR ADDRESS
;OFFSET INTO P-TABLE FOR MASSBUS ADDRESS
;OFFSET INTO P-TABLE FOR DRIVE NUMBER

LSPROT::

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 153
INITIALIZE SECTION

```
:** THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED  
: AT THE BEGINNING OF EACH PASS.  
:--
```

LSINIT::

CLR	KEYWD1
TST	DCLFLG
BEQ	INIT1
CLR	DCLFLG
DOCLN	

```

;CLEAR USER COMMAND SPECIFIER
;CLEAN UP AND EXIT REQUESTED ?
;BRANCH IF NOT
;CLEAR CLEANUP REQUEST SPECIFIER
;GO CLEAN UP
                                TRAP      C$DCLN

```

```
MOV      #-1,RESFLG
READEF   #FF,START
```

```

;SET RESTART FLAG
;IF HERE CAUSE OF START,DO SOME INIT
                                MOV     #EF.START,RO
                                TRAP     C$REFG

```

BCOMplete START

```

;IF HERE CAUSE OF RESTART, DO SOME INIT
BCS START
MOV #EF.RESTART,R0
TRAP C$REFG

```

READER REF. RESTART

BCOMPLETE RESIRT

```

;SEE IF WE'RE HERE CAUSE OF A CONTINUE
                                BCS     RESTRT
                                MOV     #EF.CONTINUE,R0
                                TRAP     CSREFG

```

READER REF. CONTINUE

;BR IF NOT HERE CAUSE OF CONTINUE
BCC S1

```

      JMP      ENDIT
      READEF   #EF.NEW

```

```

;JMP IF HERE CAUSE OF A CONTINUE
;SEE IF THIS IS A 'NEW PASS'
                                MOV     #EF.NEW,RO
                                TRAP    CSREFG
;IF YES, BR AROUND LOGUNIT # SETUP
                                BCS     NEW

```

BCOMplete NEW

BR GETPRM

```
START: CLR RESFLG
        BRESET
```

```

;CLEAR RESTART FLAG SINCE HERE ON START
;INIT ENTIRE BUS

```

CLR CLKVEC

```

:INIT ENTIRE BUS
:TRAP CSRESET
:CLEAR CLK VECTOR PTR. AS A FLAG IN
: NO CLOCK IS FOUND.
:SETUP R2 AS A PTR. TO CLOCK INFO BLOCK
:LOOK FOR A LINE CLOCK

```

```
MOV      #CLKCSR,R2
CLOCK    L,R1
```

```

:LOOK FOR A LINE CLOCK      MOV      #1,L,RO
                             TRAP      C$CLK
                             MOV      RO,R1
: IF NONE THERE GO LOOK FOR A P-CLOCK
                             BCC      S2
: GO SET UP CLOCK INFO TABLE & CLK VEC.
:SETUP THE ENABLE LINE CLOCK DATA

```

BNCOMPLETE S2

```
JSR    PC,CLKSET
MOV    #LCLKEN,CLKEN
BR      RESTRT
```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 154
INITIALIZE SECTION

SEQ 153

```

5417
5418 032752          S2:  CLOCK  P,R1          ;LOOK FOR A P-CLOCK SINCE NO LINE CLOCK
5419 032752 012700 000120          MOV      #'P,R0
5420 032756 104462          TRAP      C$CLK
5421 032760 010001          MOV      R0,R1
5422 032762          BNCOMPLETE  S3          ; IF NONE THERE GO SEE IF THIS IS LSI
5423 032762 103017          BCC      S3
5424 032764 004737 023776          JSR      PC,CLKSET
5425 032770 062737 000002 010250      ADD      #2,CLKCSR
5426 032776 012777 001600 155244      MOV      #PCLKCT,@CLKCSR
5427 033004 162737 000002 010250      SUB      #2,CLKCSR
5428 033012 012737 000111 010260      MOV      #PCLKEN,CLKEN
5429 033020 000433          BR      RESTRT
5430
5431 033022          S3:  READBUS          ;READ BUS TYPE TO SEE IF ON AN LSI
5432 033022 104407          TRAP      C$RDBU
5433 033024          BNCOMPLETE  S4          ;BR IF NOT, NO CHANCE OF A CLOCK
5434 033024 103021          BCC      S4
5435 033026 012737 000100 010254      MOV      #100,CLKVEC
5436 033034 005037 010252          CLR      CLKBR
5437 033040 012737 010260 010250      MOV      #CLKEN,CLKCSR
5438 033046          GMANID  L5060,CLKHZ,D,377,50,,60,,YES
5439 033046 104443          ;LOAD 100 AS CLK VECTOR
5440 033050 000406          ;LOAD 0 AS CLK INT. LEVEL
5441 033052 010256          ;KLUDGE UP THE CSR & ENABLE DATA LOCS
5442 033054 000052          TRAP      C$GMAN
5443 033056 016553          BR      10000$
5444 033060 000377          .WORD    CLKHZ
5445 033062 000062          .WORD    T$CODE
5446 033064 000074          .WORD    L5060
5447 033066          .WORD    377
5448 033066 000410          .WORD    T$LO! IM
5449          .WORD    T$HILIM
5450          10000$:
5451 033070          S4:  PRINTF  #BDCLK
5452 033070 012746 016671          MOV      #BDCLK,-(SP)
5453 033074 012746 000001          MOV      #1,-(SP)
5454 033100 010600          MOV      SP,R0
5455 033102 104417          TRAP      C$PRINTF
5456 033104 062706 000004          ADD      #4,SP
5457 033110 005037 010262          RESTRT: CLR      TIMMIN
5458 033114 005037 010264          CLR      TIMSEC
5459 033120 013737 010256 010266      MOV      CLKHZ,TIMTCK
5460 033126 012702 010300          MOV      #EVTLOG,R2
5461 033132 010237 010276          MOV      R2,EVTPIR
5462 033136 012722 177777          1$:  MOV      #-1,(R2)+
5463 033142 020227 011334          CMP      R2,#EVTEND
5464 033146 001373          BNE      1$
5465
5466 033150          NEW:
5467 033150 012737 177777 010206      MOV      #-1,LOGUNT
5468
5469 033156 005237 010206          GETPRM: INC      LOGUNT
5470 033162 023737 010206 002012      CMP      LOGUNT,LSUNIT
5471 033170 002367          BGE      NEW
5472
;INITIALIZE LOGICAL UNIT #
;POINT TO NEXT LOGICAL UNIT
;SEE IF PAST MAX. LOG. UNIT #
;BR IF YES, AND START OVER

```

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 155
INITIALIZE SECTION

```

5473 033172          GPWARD LOGUNT,R1          ;GET THE P-TABLE FOR THIS LOG. UNIT
5474 033172 013700 010206          MOV          LOGUNT,R0
5475 033176 104442          TRAP          C$GPWRD
5476 033200 010001          MOV          R0,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,R0
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				MOV	#1,-(SP)
5530	033462	010600					MOV	SP,R0
5531	033464	104417					TRAP	C\$PNTF
5532	033466	062706	000004				ADD	#4,SP
5533	033472			PRINTF	#DATAID,TEMP2,TEMP3,TEMP4 ;PRINT DATA			
5534	033472	013746	010174				MOV	TEMP4,-(SP)
5535	033476	013746	010172				MOV	TEMP3,-(SP)
5536	033502	013746	010170				MOV	TEMP2,-(SP)
5537	033506	012746	053771				MOV	#DATAID,-(SP)
5538	033512	012746	000004				MOV	#4,-(SP)
5539	033516	010600					MOV	SP,R0
5540	033520	104417					TRAP	C\$PNTF
5541	033522	062706	000012				ADD	#12,SP
5542	033526			DOCLN	;GO CLEAN UP			
5543	033526	104444					TRAP	C\$DOCLN
5544	033530			BREAK	;ALLOW OPERATION INTERVENTION VIA CTRL/C			
5545	033530	104422					TRAP	C\$BRK
5546	033532	000712		BR	LDFIRM ;RETRY LOADING FIRMWARE			
5547								
5548								
5549								
5550					;CONTINUE LOADING FIRMWARE			
5551	033534	005205		10\$: INC	R5 ;UPDATE THE CRAM ADDRESS			
5552	033536	022704	053660	CMP	#DDCMPE,R4 ;TEST FOR DONE			
5553	033542	001314		BNE	2\$;LOOP TIL FIRMWARE LOADED			
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	013410				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:				
5582	033646			SETPRI	#PRI00 ;SET THE 'RUN' PRIORITY TO 0			
5583	033646	012700	000000				MOV	#PRI00,R0
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 C\$EXIT
.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	^0	120720	^0	103015	^0	16400	^0	100411	^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	^0	100075	^0	10061	^0	4017	^0	2777	^0	110574	^0	100534	^0	410
033760	010050	004017	062620	.WORD	^0	10050	^0	4017	^0	62620	^0	110574	^0	10041	^0	4017	^0	51225	^0	62605
034000	100457	040365	101456	.WORD	^0	100457	^0	40365	^0	101456	^0	3200	^0	40360	^0	101057	^0	2400	^0	42231
034020	000404	061233	100000	.WORD	^0	404	^0	61233	^0	100000	^0	110574	^0	100516	^0	10042	^0	4017	^0	43225
034040	000400	060365	101474	.WORD	^0	400	^0	60365	^0	101474	^0	100516	^0	4002	^0	10000	^0	40371	^0	101516
034060	043621	004014	010000	.WORD	^0	43621	^0	4014	^0	10000	^0	50011	^0	40361	^0	101516	^0	120720	^0	4014
034100	010000	052410	101110	.WORD	^0	10000	^0	52410	^0	101110	^0	76605	^0	62601	^0	65620	^0	410	^0	61233
034120	100000	110574	010061	.WORD	^0	100000	^0	110574	^0	10061	^0	4017	^0	42411	^0	43220	^0	530	^0	160400
034140	100436	100444	100464	.WORD	^0	100436	^0	100444	^0	100464	^0	100432	^0	10044	^0	4017	^0	57220	^0	40360
034160	101542	100443	120620	.WORD	^0	101542	^0	100443	^0	120620	^0	3077	^0	103443	^0	63677	^0	61231	^0	10050
034200	004017	043220	000554	.WORD	^0	4017	^0	43220	^0	554	^0	160400	^0	100570	^0	100645	^0	114672	^0	114672
034220	114672	100675	100737	.WORD	^0	114672	^0	100675	^0	100737	^0	114672	^0	100746	^0	110502	^0	110515	^0	110570
034240	010051	004017	057220	.WORD	^0	10051	^0	4017	^0	57220	^0	40360	^0	101605	^0	3007	^0	42407	^0	101201
034260	100602	002404	000405	.WORD	^0	100602	^0	2404	^0	405	^0	100440	^0	4017	^0	10053	^0	57220	^0	40360
034300	101630	052410	004016	.WORD	^0	101630	^0	52410	^0	4016	^0	57222	^0	101604	^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	^0	100437
034340	120400	103633	100443	.WORD	^0	120400	^0	103633	^0	100443	^0	3020	^0	61202	^0	102237	^0	100643	^0	120620
034360	003200	063700	061231	.WORD	^0	3200	^0	63700	^0	61231	^0	401	^0	100733	^0	120440	^0	103343	^0	2410
034400	010067	004003	123441	.WORD	^0	10067	^0	4003	^0	123441	^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	^0	122460	^0	56667
034440	136520	136500	136560	.WORD	^0	136520	^0	136500	^0	136560	^0	136540	^0	100443	^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	^0	120400	^0	103326	^0	100732	^0	120620	^0	3300
034540	063700	061231	000406	.WORD	^0	63700	^0	61231	^0	406	^0	4017	^0	10107	^0	2410	^0	100440	^0	120440
034560	103743	002400	100443	.WORD	^0	103743	^0	2400	^0	100443	^0	407	^0	61233	^0	100203	^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	^0	160403	^0	104403	^0	104572	^0	104730	^0	417	^0	61233	^0	100100
034640	004000	013400	003377	.WORD	^0	4000	^0	13400	^0	3377	^0	16400	^0	63077	^0	101777	^0	100773	^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,^0	100437,^0	16450,^0	76613,^0	62614,^0	4003,^0	10075,^0	40371
035720	101443	042412	010074	.WORD	^0101443,^0	42412,^0	10074,^0	42411,^0	100437,^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,^0	123304,^0	137337,^0	54220,^0	111446,^0	110450,^0	40220,^0	111466
036000	000510	070004	111054	.WORD	^0 510,^0	70004,^0	111054,^0	110455,^0	64077,^0	14000,^0	54220,^0	111461
036020	110463	040220	111466	.WORD	^0110463,^0	40220,^0	111466,^0	646,^0	61222,^0	100723,^0	402,^0	63236
036040	000475	063237	000414	.WORD	^0 475,^0	63237,^0	414,^0	61233,^0	100021,^0	620,^0	76707,^0	62603
036060	000411	100440	010043	.WORD	^0 411,^0	100440,^0	10043,^0	40012,^0	111506,^0	100443,^0	10044,^0	57220
036100	040360	111513	100443	.WORD	^0 40360,^0	111513,^0	100443,^0	400,^0	100440,^0	4017,^0	10043,^0	40012
036120	111522	100443	010044	.WORD	^0111522,^0	100443,^0	10044,^0	4017,^0	57220,^0	40360,^0	111530,^0	100443
036140	014000	057220	040360	.WORD	^0 14000,^0	57220,^0	40360,^0	111535,^0	100443,^0	10071,^0	4003,^0	43220
036160	000402	063236	000547	.WORD	^0 402,^0	63236,^0	547,^0	63237,^0	413,^0	61233,^0	100075,^0	123306
036200	123327	016777	016777	.WORD	^0123327,^0	16777,^0	16777,^0	16777,^0	16777,^0	510,^0	70006,^0	111161
036220	110562	064067	016777	.WORD	^0110562,^0	64067,^0	16777,^0	16777,^0	16777,^0	16777,^0	413,^0	100440
036240	002400	000602	061222	.WORD	^0 2400,^0	602,^0	61222,^0	100723,^0	10043,^0	4017,^0	43220,^0	601
036260	170400	110626	110653	.WORD	^0170400,^0	110626,^0	110653,^0	110664,^0	110653,^0	110715,^0	110653,^0	114446
036300	114674	110653	114535	.WORD	^0114674,^0	110653,^0	114535,^0	110653,^0	114556,^0	114674,^0	110653,^0	114535
036320	110653	114556	110653	.WORD	^0110653,^0	114556,^0	110653,^0	114640,^0	110653,^0	114655,^0	10046,^0	4017
036340	057220	040360	111643	.WORD	^0 57220,^0	40360,^0	111643,^0	3005,^0	42405,^0	111237,^0	110640,^0	2401
036360	010043	002401	100522	.WORD	^0 10043,^0	2401,^0	100522,^0	10044,^0	57220,^0	40360,^0	101522,^0	42410
036400	010043	002403	100522	.WORD	^0 10043,^0	2403,^0	100522,^0	120600,^0	102122,^0	42411,^0	123620,^0	112261
036420	100522	000554	061271	.WORD	^0100522,^0	554,^0	61271,^0	110663,^0	120620,^0	103522,^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,^0	101522,^0	4017,^0	110643,^0	120620,^0	103522,^0	2405
036520	010045	050220	004010	.WORD	^0 10045,^0	50220,^0	4010,^0	3007,^0	43667,^0	63220,^0	57621,^0	113375
036540	060521	117430	060601	.WORD	^0 60521,^0	117430,^0	60601,^0	113775,^0	402,^0	63236,^0	743,^0	63237
036560	000413	061233	100146	.WORD	^0 413,^0	61233,^0	100146,^0	14000,^0	0,^0	120700,^0	56225,^0	42224
036600	042410	111353	110763	.WORD	^0 42410,^0	111353,^0	110763,^0	70220,^0	56411,^0	111357,^0	110763,^0	14000
036620	003100	042400	110764	.WORD	^0 3100,^0	42400,^0	110764,^0	14000,^0	3300,^0	43660,^0	61620,^0	61620
036640	061620	061620	060711	.WORD	^0 61620,^0	61620,^0	60711,^0	61230,^0	114702,^0	43221,^0	63161,^0	3400

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

MACV11 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	0115631	^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	0120600	^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

MACY11 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	^0	0160400
037700	100437	100602	100602	.WORD	^0	0100437	^0	0100602	^0	0100602	^0	0100602	^0	0100602	^0	0100602	^0	0100620	^0	0100644
037720	104443	100421	104613	.WORD	^0	0104443	^0	0100421	^0	0104613	^0	0104613	^0	0104750	^0	0110431	^0	0110556	^0	0110703
037740	114410	000000	000000	.WORD	^0	0114410	^0	0	^0	0	^0	0	^0	0	^0	400	^0	61233	^0	0100063
037760	000404	063236	000447	.WORD	^0	404	^0	63236	^0	447	^0	63237	^0	417	^0	61233	^0	0100222	^0	40620
040000	102052	100431	020660	.WORD	^0	0102052	^0	0100431	^0	20660	^0	0102055	^0	0100434	^0	400	^0	62232	^0	0123316
040020	000412	070016	016400	.WORD	^0	412	^0	70016	^0	16400	^0	2400	^0	23200	^0	0121273	^0	0123244	^0	0114423
040040	004002	070215	016401	.WORD	^0	4002	^0	70215	^0	16401	^0	3201	^0	62600	^0	40361	^0	0101505	^0	3005
040060	040365	101434	070215	.WORD	^0	40365	^0	0101434	^0	70215	^0	2420	^0	0100434	^0	507	^0	0160407	^0	0100543
040100	100540	100535	100532	.WORD	^0	0100540	^0	0100535	^0	0100532	^0	0100527	^0	0100524	^0	0100521	^0	10020	^0	4005
040120	100545	010007	004005	.WORD	^0	0100545	^0	10007	^0	4005	^0	0100545	^0	10376	^0	4004	^0	0100545	^0	10365
040140	004004	100545	010354	.WORD	^0	4004	^0	0100545	^0	10354	^0	4004	^0	0100545	^0	10343	^0	4004	^0	0100545
040160	010332	004004	100545	.WORD	^0	10332	^0	4004	^0	0100545	^0	10321	^0	4004	^0	54220	^0	0101550	^0	0100434
040200	040220	101553	100434	.WORD	^0	40220	^0	0101553	^0	0100434	^0	60607	^0	63221	^0	64217	^0	402	^0	70016
040220	042412	101563	100434	.WORD	^0	42412	^0	0101563	^0	0100434	^0	404	^0	63226	^0	572	^0	63227	^0	407
040240	061233	100312	016402	.WORD	^0	61233	^0	0100312	^0	16402	^0	76601	^0	16400	^0	16400	^0	16400	^0	16400
040260	002602	100434	020660	.WORD	^0	2602	^0	0100434	^0	20660	^0	0102205	^0	0100434	^0	56411	^0	400	^0	62232
040300	120700	070000	022600	.WORD	^0	0120700	^0	70000	^0	22600	^0	43220	^0	0121273	^0	0123244	^0	0114423	^0	0100434
040320	020660	102223	100434	.WORD	^0	20660	^0	0102223	^0	0100434	^0	0123314	^0	56411	^0	400	^0	62232	^0	3000
040340	121273	123244	114423	.WORD	^0	0121273	^0	0123244	^0	0114423	^0	0121273	^0	0123244	^0	0114423	^0	64217	^0	70216
040360	023200	040360	101434	.WORD	^0	23200	^0	40360	^0	0101434	^0	0100670	^0	20660	^0	0102247	^0	0100434	^0	400
040400	062232	123314	000404	.WORD	^0	62232	^0	0123314	^0	404	^0	63236	^0	661	^0	63237	^0	417	^0	61233
040420	100222	123316	123337	.WORD	^0	0100222	^0	0123316	^0	0123337	^0	413	^0	70016	^0	23200	^0	40360	^0	0101774
040440	004002	070214	002420	.WORD	^0	4002	^0	70214	^0	2420	^0	401	^0	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	^0	0160404	^0	0100772	^0	0100760	^0	0100746	^0	0100743
040540	100772	100772	100735	.WORD	^0	0100772	^0	0100772	^0	0100735	^0	2440	^0	0100736	^0	2401	^0	4013	^0	410
040560	070006	002400	100434	.WORD	^0	70006	^0	2400	^0	0100434	^0	16404	^0	62603	^0	0100434	^0	16402	^0	62602
040600	004013	000400	070006	.WORD	^0	4013	^0	400	^0	70006	^0	477	^0	62673	^0	0100434	^0	70220	^0	0100761
040620	120700	016420	076602	.WORD	^0	0120700	^0	16420	^0	76602	^0	14000	^0	40362	^0	0101767	^0	0100736	^0	70220
040640	002400	100434	000421	.WORD	^0	2400	^0	0100434	^0	421	^0	0110743	^0	4002	^0	70214	^0	16410	^0	405

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

MACV11 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

MACV11 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 13:24

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	16601	^0	0100434
042700	022700	003376	042676	.WORD	^0	22700	^0	3376	^0	42676	^0	42234	^0	3001	^0	0	^0	0	^0	42701
042720	042234	002400	100434	.WORD	^0	42234	^0	2400	^0	0100434	^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	0160064	^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	0117177	^0	60603	^0	60721	^0	0116130	^0	10320	^0	0114531
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	0117144	^0	0114523	^0	63064	^0	70205	^0	76603	^0	76602
043200	063221	123305	070206	.WORD	^0	63221	^0	0123305	^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	0123306	^0	0114513	^0	61202
043260	000400	061233	100032	.WORD	^0	400	^0	61233	^0	0100032	^0	40620	^0	4017	^0	10106	^0	56411	^0	0115611
043300	114653	042411	115614	.WORD	^0	0114653	^0	42411	^0	0115614	^0	0114653	^0	0116624	^0	0123001	^0	577	^0	60661
043320	061220	000406	063221	.WORD	^0	61220	^0	406	^0	63221	^0	0114626	^0	410	^0	63221	^0	0123042	^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	0100312	^0	76601	^0	0136460	^0	0136500	^0	0136520	^0	0136540	^0	0136560	^0	2603
043400	004017	010050	002400	.WORD	^0	4017	^0	10050	^0	2400	^0	400	^0	61233	^0	0100043	^0	10042	^0	4017
043420	040620	061620	061620	.WORD	^0	40620	^0	61620	^0	61620	^0	61620	^0	63220	^0	667	^0	0174400	^0	10140
043440	114706	010143	114706	.WORD	^0	0114706	^0	10143	^0	0114706	^0	10146	^0	0114706	^0	10151	^0	0114706	^0	10154
043460	114706	010157	114706	.WORD	^0	0114706	^0	10157	^0	0114706	^0	10162	^0	0114706	^0	10165	^0	4005	^0	60616
043500	061233	160217	010051	.WORD	^0	61233	^0	0160217	^0	10051	^0	4017	^0	407	^0	63220	^0	42400	^0	0115321
043520	114722	002404	057220	.WORD	^0	0114722	^0	2404	^0	57220	^0	40360	^0	0115732	^0	70200	^0	4011	^0	60606
043540	061233	160207	114732	.WORD	^0	61233	^0	0160207	^0	0114732	^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	0114761	^0	1141	^0	0114763	^0	1121	^0	0114765	^0	1101	^0	0114767
043640	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 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	16777	^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

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 167
INITIALIZE SECTION

;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	117004,^0	100644,^0	42411,^0	123316,^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	100644,^0	60605
047040	061620	061620	061620	.WORD	^0 61620,^0	61620,^0	61620,^0	61620,^0	63220,^0	477,^0	174400,^0	114533
047060	114530	114525	114522	.WORD	^0114530,^0	114525,^0	114522,^0	114517,^0	114514,^0	114511,^0	10017,^0	4005
047100	114535	010006	004005	.WORD	^0114535,^0	10006,^0	4005,^0	114535,^0	10375,^0	4004,^0	114535,^0	10364
047120	004004	114535	010353	.WORD	^0 4004,^0	114535,^0	10353,^0	4004,^0	114535,^0	10342,^0	4004,^0	114535
047140	010331	004004	114535	.WORD	^0 10331,^0	4004,^0	114535,^0	10320,^0	4004,^0	60616,^0	61233,^0	160217
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	114576,^0	114574,^0	114572,^0	114570,^0	114566,^0	114564,^0	114562
047220	010177	114577	010166	.WORD	^0 10177,^0	114577,^0	10166,^0	114577,^0	10155,^0	14577,^0	10144,^0	114577
047240	010133	114577	010122	.WORD	^0 10133,^0	114577,^0	10122,^0	114577,^0	10111,^0	14577,^0	10100,^0	60616
047260	061233	160217	000000	.WORD	^0 61233,^0	160217,^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	114761,^0	1141,^0	114763,^0	1121,^0	114765,^0	1101,^0	114767
047640	001061	114771	001041	.WORD	^0 1061,^0	114771,^0	1041,^0	114773,^0	1021,^0	114775,^0	1001,^0	114777

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 169
INITIALIZE SECTION

: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	32^0
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

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 170
INITIALIZE SECTION

: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 16400,^0 76603,^0 2601,^0 60602

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

MACY11 3CA(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.^0160064
053060 000502 174407 114505 .WORD ^0 502.^0174407.^0114505.^0114564.^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.^0100312.^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.^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.^0100043.^0 14000.^0 57221.^0 43222.^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 ^0100312.^0 16404.^0 16400.^0 76602.^0 76601.^0 76603.^0 16400.^0 2603
053300 114561 014000 057221 .WORD ^0114561.^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.^0114631
053340 040620 003007 063667 .WORD ^0 40620.^0 3007.^0 63667.^0 63226.^0 636.^0174406.^0114672.^0114667
053360 114664 114661 114656 .WORD ^0114664.^0114661.^0114656.^0114653.^0114650.^0 10300.^0 4001.^0114674
053400 010200 004001 114674 .WORD ^0 10200.^0 4001.^0114674.^0 10100.^0 4001.^0114674.^0 10000.^0 4001
053420 114674 010300 004000 .WORD ^0114674.^0 10300.^0 4000.^0114674.^0 10200.^0 4000.^0114674.^0 10100
053440 004000 114674 010000 .WORD ^0 4000.^0114674.^0 10000.^0 4000.^0 60616.^0 61233.^0160217.^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.^0114763.^0 1121.^0114765.^0 1101.^0114767
053640 001061 114771 001041 .WORD ^0 1061.^0114771.^0 1041.^0114773.^0 1021.^0114775.^0 1001.^0114777

```

053660

DDCMPE:

:END OF DCLT FIRMWARE IMAGE

020000

MCSIZ = .-DDCMPS

:SIZE OF FIRMWARE IMAGE

```

053660 047045 051445 022463 FIRMLD: .ASCII /%N%$3%AFailure Loading Firmware/
053720 047045 022462 031523 DATAHD: .ASCII /%N2%$3%ADDRESS%$3%LOADED%$3%AREAD BACK/
053771 045 022516 031523 DATAID: .ASCII /%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

MACv11 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.
;--

BGNAUTO

LSAUTO::

ENDAUTO

L10016: TRAP CSAUTO

5595
5596
5597
5598
5599
5600
5601
5602
5603
5604 054020
5605 054020
5606
5607
5608 054020
5609 054020
5610 054020 104461

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

MACV11 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.
:--

BGNCLN

L\$CLEAN::

```

5611
5612
5613
5614
5615
5616
5617
5618 054022
5619 054022
5620
5621
5622 054022 005737 011512          TST      ENRDST      ;IS READ STATUS ENABLED ?
5623 054026 001402          BEQ      10$          ;NO. OMIT READING STATUS
5624 054030 004737 063704          JSR      PC,LNSTAT    ;READ LINE STATUS
5625 054034 005237 011516          10$: INC      CNTCLN      ;INCREMENT CLEANUP COUNT
5626 054040 005037 011512          CLR      ENRDST      ;NEGATE READ STATUS ENABLED
5627 054044 012737 177777 010134  MOV      #-1,CLNSET  ;SET THE CLEANUP FLAG
5628 054052 004737 064020          JSR      PC,HLTTRB    ;HALT LINE
5629 054056 005037 010134          CLR      CLNSET      ;CLEAR CLEANUP FLAG
5630 054062 005077 134162          CLR      @CLKCSR     ;DISABLE CLOCK
5631 054066
5632 054066 104433
5633 054070          SETPRI   #PRI07      ;SET PROCESSOR PRIORITY BACK TO 7
5634 054070 012700 000340          TRAP    7          C$RESET
5635 054074 104441          MOV      #PRI07,R0
5636          TRAP    C$SPRI
5637 054076 022737 000057 003246  CMP      #EXIT,KEYWD1 ;"EXIT" COMMAND?
5638 054104 001416          BEQ      EXITCLN      ;YES.
5639
5640          :LOG ^C ABORT IN EVENT LOG
5641 054106 012737 000026 010164  MOV      #ABO,TEMP    ;LOAD EVENT TYPE
5642 054114 013737 010116 010174  MOV      OPVAR,TEMP4   ;NUMBER OF NO-BUFFS
5643 054122 013737 010122 010170  MOV      PSCNT,TEMP2   ;NUMBER OF PASSES
5644 054130 013737 010124 010172  MOV      ERRCNT,TEMP3  ;ERRORS
5645 054136 004737 024534          JSR      PC,LOGS5    ; GO LOG IT
5646
5647 054142          EXITCLN:
5648 054142          EXIT      CLN
5649 054142 104432          TRAP    C$EXIT
5650 054144 000002          .WORD   L10017-.
5651
5652
5653          .EVEN
5654
5655 054146          ENDCLN
5656 054146          L10017:
5657 054146 104412          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.
:--

BGNDU

L\$DU::

EXIT DU

.WORD JSJMP
.WORD L10020-2-.

.EVEN

ENDDU

L10020:

TRAP C\$DU

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

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
054156 000167
054160 000000

054162
054162
054162 104452

BGNAU

EXIT AU

.EVEN

ENDAU

LSAU::

.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.
:--

```

BGNTST

T!::

.SBTTL PROGRAM SETUP SECTION

```

5703
5704
5705
5706
5707
5708
5709
5710
5711
5712
5713
5714 054164
5715 054164
5716
5717
5718
5719
5720
5721 054164 013777 010260 134056      MOV      CLKEN,@CLKCSR      ;ENABLE THE CLOCK
5722
5723 054172
5724 054172 005001
5725 054174 012737 000001 010270      GTXRXB:
5726 054202 005737 010270      GTRA2:  CLR      R1
5727 054206 001412      MOV      #1,TIMER1      ;SET TIMER TO COUNT 1 TICK
5728 054210 005301      TST      TIMER1      ;CHECK FOR IT TO BE COUNTED OFF
5729 054212 001373      BEQ      GTRA3      ;BRANCH IF CLOCK EXISTS (COUNTED A TICK)
5730 054214      DEC      R1
5731 054214 012746 016715      BNE      1$      ;KEEP CHECKING UNTIL R1 DOES FULL COUNTDOWN
5732 054220 012746 000001      PRINTF  #NOCLK      ;PRINT BAD CLK MSG AND WARN OF HANG IF TIMEOUT
5733 054224 010600      MOV      #NOCLK,-(SP)
5734 054226 104417      MOV      #1,-(SP)
5735 054230 062706 000004      MOV      SP,R0
5736      TRAP    C$PNTF
5737      ADD     #4,SP
5738 054234 005737 010214      GTRA3:  TST      RESFLG      ;SEE IF HERE AFTER A RESTART
5739 054240 001124      BNE      GTRA5      ;BR IF HERE BECAUSE OF A RESTART
5740      ; CLEAR COUNTS AND SET UP DEFAULTS
5741
5742 054242 005037 010160      GTRA4:  CLR      TOTCC      ;CLEAR TOTAL CHAR. COUNT TEMP. LOC
5743 054246 005037 010076      CLR      TTOTCC      ; CLEAR TOTAL CHAR. COUNT FOR TX BUFF
5744 054252 005037 010060      CLR      CTOTCC      ; CLEAR TOTAL CHAR. COUNT FOR CMP BUFF
5745 054256 012737 006406 010052      MOV      #PTRTAB, TXPTR ;INIT TRANSMIT MESSAGE POINTER
5746
5747 054264 005037 010050      CLR      RXPTR      ; ZERO RX POINTER
5748 054270 012737 006502 010054      MOV      #PTR13, CMPPTR ;INIT COMP POINTER
5749
5750 054276 012737 000005 010146      MOV      #5,MSGTYP      ;SET UP DEFAULT MSG TYPE (QUICK FOX - ITEX MSG)
5751 054304 013737 002156 010150      MOV      MSG5C,CURCC      ;SET UP DEFAULT CHAR COUNT
5752 054312 012737 003406 010100      MOV      #TXBUF,TCURAD      ;SET UP CURRENT ADD TO START OF TX BUFFER
5753 054320 012737 004406 010062      MOV      #CMPBUF,CCURAD      ;SET UP CURRENT ADD TO START OF CMP BUFFER
5754
5755 054326 013737 010100 010156      MOV      TCURAD,CURADD      ;SETUP CURRENT ADDR TO START OF TXBUF
5756 054334 013737 010052 010154      MOV      TXPTR,CPTR      ;SETUP CURRENT POINTER TABLE POINTER FOR TXBUF
5757 054342 004737 027520      JSR      PC,BLDBUF      ; GO BUILD POINTER TABLE AND BUFFER
5758 054346 012737 000001 010074      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 MSG 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                                     ;OPERATOR STATUS MSGS. PRINT OFF
5771 054444 012737 000063 003246      MOV      #KTRB,KEYWD1    ;SET UP KEYWRD.
5772 054452 004737 057566              JSR      PC,ACTKAL        ;INIT LINE LIST
5773
5774 054456 012702 013434              MOV      #BDLBAS,R2      ;INIT MSG BUFFER DESCRIPT LIST
5775 054462 005022 1$:                CLR      (R2)+
5776 054464 020227 013634              CMP      R2,#BDLEND
5777 054470 001374 1$                  BNE
5778
5779                                     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      PS$GDBD      ;CLEAR CMD LINE PARSING ERROR FLAGS
5807      054602      105037      003400      CLRB      PS$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$CODE
5813      054616      014002      .WORD      CLISPM
5814      054620      000377      .WORD      377
5815      054622      000001      .WORD      T$LOLIM
5816      054624      000110      .WORD      T$HILIM
5817      054626      10000$:
5818      054626      012737      003124      003364      MOV      #CMDBUF,PS$BUFA
5819      054634      012737      011520      003366      MOV      #CLITRE,PS$TREE
5820      054642      012737      055540      003370      MOV      #CLIACT,PS$ACT
5821      054650      005037      003250      CLR      QUALFG      ;CLEAR QUALIFIER FLAG LOCATION
5822      054654      004737      031506      JSR      PC,PS$TRV      ;GO PARSE COMMAND LINE
5823      054660      105737      003401      TSTB      PS$GDBD      ;SEE IF PARSED OK OR AN ERROR
5824      054664      001412      BEQ      1$
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      PS$NNUF      ;SEE IF INCOMPLETE COMMAND TYPED
5833      054716      001412      BEQ      10$
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$:      CMP      KEYWD1,#DMPS      ;IS IT DUMP
5848      054776      001004      BNE      14$
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$:      CMP      KEYWD1,#EXIT      ;IS IT EXIT
5852      055016      001005      BNE      40$      ;BRANCH IF NOT
5853      055020      012737      177777      010212      MOV      #-1,DCLFLG      ;SET DO CLEAN FLAG
5854      055026      EXIT      TST
5855      055026      104432      TRAP      C$EXIT
5856      055030      012174      .WORD      L10022-.
5857      055032      023727      003246      000010      40$:      CMP      KEYWD1,#SETEXP      ;SEE IF SET EXPECTED
5858      055040      001001      BNE      4$      ; BR IF YES (A SETEXP WAS TYPED)
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				BEO	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,R0
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		;SEE IF MSG COUNT EXCEEDED.
5881	055156	002414				BLT	17\$		; BR IF NO
5882	055160					PRINTF	#MSGTRN,#TABEX		; ELSE TELL OPR. AND DON'T BUILD MSG.
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,R0
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		;SETUP CHAR. COUNT, CURRENT ADDR, & PTR
5894	055226	013737	010100	010156		MOV	TCURAD, CURADD		
5895	055234	004737	027422			JSR	PC, ADDCC		;ADD IN CHAR. COUNT AND CHECK TOTAL
5896	055240	004737	027520			JSR	PC, BLDBUF		;GO BUILD MESSAGE IN BUFFER AND PTRS.
5897	055244	004737	027644			JSR	PC, FCURAD		;MAKE SURE ADDRESS IS WORD BOUNDARY
5898	055250	013737	010154	010052		MOV	CPTR, TXPTR		
5899	055256	013737	010160	010076		MOV	TOTCC, TTOTCC		;UPDATE CHAR. COUNT, CURR ADDR, & PTR
5900	055264	013737	010156	010100		MOV	CURADD, TCURAD		
5901	055272	005237	010074			INC	TXMTOT		
5902	055276	005337	003252			DEC	QUALVL		;DEC THE COPY COUNT
5903	055302	001266				BNE	5\$		
5904	055304	000137	054576			JMP	GETCL		
5905									
5906	055310	013737	010060	010160	2\$:	MOV	CTOTCC, TOTCC		;SETUP CHAR. COUNT, CURR. ADDR. & PTR
5907	055316	023727	010160	001000		CMP	TOTCC, #BUFLIM		;SEE IF BUFFER ALREADY FULL
5908	055324	002414				BLT	16\$		; BR IF NOT FULL (BUFLIM # OF CHARS.)
5909	055326					PRINTF	#MSGTRN, #BUFEX		; ELSE TELL OPR. AND DON'T BUILD MSG.
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,R0
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 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 PSGDBD 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 ACTSZE-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$ :TALT
.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 ACTNUP-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
.WORD ACTDME-10$ :DUMP MEMORY END 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	.WORD	ACTDMQ-10\$	:DUMP WORD
6011	055710	000260	.WORD	ACTPRT-10\$	:PRINT
6012	055712	001604	.WORD	ACTMOS-10\$	:MODEM STATUS CHANGE
6013	055714	000250	.WORD	ACTEXT-10\$	:EXIT
6014	055716	001304	.WORD	ACTSEX-10\$	:SET E=T COMMAND
6015	055720	002070	.WORD	ACTSLS-10\$	:SHOW LINE LIST
6016	055722	001772	.WORD	ACTETB-10\$	:ESTABLISH LINE
6017	055724	002002	.WORD	ACTKTB-10\$	:KILL LINE
6018	055726	002010	.WORD	ACTKAL-10\$	:KILL ALL
6019	055730	002266	.WORD	ACTEKT-10\$	:FLAG LINE KILLED
6020					

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 184
ACTION TABLE AND ROUTINES

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	MSGSC, 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					
6111	056404	005302			ADD	#4, R1		;BUMP R1 TO NEXT SET OF POINTERS
6112	056406	001341			DEC	R2		
6113	056410	013737	010216	011470	BNE	ACTSHW		
6114	056416	013737	010220	011472	MOV	MODTYP, DEV1		
6115	056424	013737	010226	011474	MOV	MLTYP, DEV2		
6116	056432	013737	010224	011476	MOV	RPASS, DEV3		
6117	056440	004737	031204		MOV	PARAM, DEV4		
6118	056444	000207			JSR	PC, SHWOP		;SHOW THE OPERATOR THE CURRENT MODE..... ALSO
6119					RTS	PC		
6120	056446	013737	003374	010126	ACTDMS: MOV	PSNUM, 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	PSNUM, ENADD		;SETUP END ADDRESS FOR DUMP (=START IF NO 'EEE'
6127	056504	105037	003400		ACTDMX: CLRB	PSNUF		;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	#LLISEO		
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,PSGDL3	;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,PSGDBD	;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	PSNNUF	;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,PSGDBD	
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	PSNUM,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,PSGDBD	

;BE SURE IN PASSIVE MODE IF
 ;IF TRYING TO SET /ECHO
 MOV #CLINPS,-(SP)
 MOV #1,-(SP)
 MOV SP,R0
 TRAP C\$PNTF
 ADD #4,SP
 ;CLEAR 'NO' OUT OF QUALIFIE.. 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	CLR	MLTYP		;CLEAR ANY LOOP TYPE THAT MAY HAVE GOT SET	
6304	057522			PRINTF	#CLIBDL			
6305	057522	012746	014125					MOV #CLIBDL,-(SP)
6306	057526	012746	000001					MOV #1,-(SP)
6307	057532	010600						MOV SP,R0
6308	057534	104417						TRAP C\$PNTF
6309	057536	062706	000004					ADD #4,SP
6310	057542	105037	003400	ACTLXX: CLR	B	P\$NNUF	;CLEAR NOT-ENOUGH FLAG	
6311	057546	000207		RTS	PC			
6312								

CZKMSAO RMS11-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: CLRB P$NNUF ; CLEAR INCOMPLETE INFO FLAG
6320 057572 022737 000063 003246      CMP #KTRB,KEYWD1 ; BE SURE 'ALL' IS AFTER A 'KILL'
6321 057600 001403          BEQ 11$ ; BR IF YES
6322 057602 112737 177777 003401      MOV #1,P$GDBD ; ELSE ERROR IN CMD
6323 057610 105737 003401      11$: TSTB P$GDBD ; SEE IF WAS AN ERROR FROM ..KTB
6324 057614 001401          BEQ 10$ ; 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$: RTS PC ; RETURN TO CALL
6333
6334 057646 105037 003400          ACTSLS: CLRB P$NNUF ; 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      TSTB P$GDBD ; SEE IF WAS AN ERROR FROM ..KTB
6337 057664 001401          BEQ 10$ ; 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$PNTS
6347 057716 062706 000004          ADD #4,SP
6348 057722 000437          BR 4$
6349 057724 012702 007622      1$: MOV #TRIBLS,R2 ; POINT R2 TO THE LINE NUMBER LIST
6350 057730 012705 000010      MOV #8,R5 ; SETUP R5 AS A COUNTER
6351 057734          PRINTS #SHTRH ; PRINT LINE LIST HEADER
6352 057734 012746 016042          MOV #SHTRH,-(SP)
6353 057740 012746 000001          MOV #1,-(SP)
6354 057744 010600          MOV SP,R0
6355 057746 104416          TRAP C$PNTS
6356 057750 062706 000004          ADD #4,SP
6357 057754 122712 000377      2$: CMPB #377,(R2) ; SEE IF A NULL ENTRY
6358 057760 001415          BEQ 3$ ; BR IF YES
6359 057762 111237 010164      MOVB (R2),TEMP
6360 057766          PRINTS #SHTAP,<B,TEMP>
6361 057766 005046          CLR -(SP)
6362 057770 153716 010164          BISB TEMP,(SP)
6363 057774 012746 016066          MOV #SHTAP,-(SP)
6364 060000 012746 000002          MOV #2,-(SP)
6365 060004 010600          MOV SP,R0
6366 060006 104416          TRAP C$PNTS
6367 060010 062706 000006          ADD #6,SP
6368 060014 005202      3$: 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

5$: RTS PC ; RETURN TO CALL

ACTEKT:
    CLR VALTRB ; INIT. VALID LINE FLAG
    CLR PSNUF ; CLEAR NOT ENOUGH INFO FLAG
    TSTB PSGBDB ; SEE IF WAS AN ERROR FROM ..KTB
    BEQ 10$ ; BR IF NO
    BR ACTEXX ; ELSE EXIT

10$: MOV PSNUM,R1 ;SEE THAT ITS A VALID LINE NUM (0-7)
     CMP #7,R1
     BHS 1$
     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
     MOVB 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
     MOV #-1,VALTRB ; 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:
6422          ; RX ALLOCATE CODE
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$PNTF
6435 060236 062706 000004                    ADD      #4, SP
6436 060242 000137 054576      JMP      GETCL                    ;GO GET NEXT COMMAND
6437
6438          44$:
6439 060246 005737 007634      TST      TRBTOT      ;CHECK FOR AT LEAST ONE LINE ENABLED
6440 060252 001013                    BNE      4$                ;IS LINE TOTAL
6441 060254                    PRINTF    #SHTLPA      ;ZERO?..BR IF NOT
6442 060254 012746 016072                    ;PRINT ERROR MUST ESTABLISH LINE
6443 060260 012746 000001                    MOV      #SHTLPA, -(SP)
6444 060264 010600                    MOV      #1, -(SP)
6445 060266 104417                    MOV      SP, R0
6446 060270 062706 000004                    TRAP     C$PNTF
6447 060274 112737 177777 003401      ADD      #4, SP
6448 060302          4$:
6449 060302 105737 003401      MOVB     #-1, P$GDBD      ;SET ERROR FLAG
6450 060306 001043          ;CHECK ERROR FLAG
6451          TSTB     P$GDBD      ;TEST ERROR FLAG
6452          BNE      12$          ;BRANCH IF ERROR
6453          ;CHECK IF DEFAULT RUN COMMAND
6454 060310 013737 010060 010150      MOV      CTOTCC, CURCC      ;MAKE CURRENT COUNT= COMPARE COUNT
6455 060316 005737 010150      TST      CURCC      ;TEST TOTAL COMPARE COUNT
6456 060322 001003          BNE      1$                ;BRANCH IF NON DEFAULT
6457 060324 012737 000072 010150      MOV      #58., CURCC      ;SET UP DEFAULT
6458 060332          1$:
6459 060332 032737 000002 010224      BIT      #DATCKB,PARAM
6460 060340 001430                    BEQ      2$                ;BRANCH IF NOT CHECKING
6461 060342 013737 007634 010046      MOV      TRBTOT, MPLY
6462 060350 013737 010150 010164      MOV      CURCC, TEMP
6463 060356 005037 010170      CLR      TEMP2
6464 060362 004737 030356      JSR      PC, MPLY      ;MULTIPY TRBTOT BY CURCC
6465          ;RESULT IN TEMP2
6466 060366 022737 001000 010170      CMP      #BUFLIM, TEMP2      ;IS IT MUCH TO MUCH
6467 060374 002012                    BGE      2$                ;NO EVERTHING IS HUNKY DORY
6468 060376                    PRINTF    #SHTBR      ;ERROR
6469          2$:
6470 060376 012746 016237                    MOV      #SHTBR, -(SP)
6471 060402 012746 000001                    MOV      #1, -(SP)
6472 060406 010600                    MOV      SP, R0
6473 060410 104417                    TRAP     C$PNTF
6474 060412 062706 000004                    ADD      #4, SP
6475 060416 000137 054576      JMP      GETCL                    ;GO BACK TO GET NEW COMMAND

```

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      2$:      ;INITIALIZE POINTERS, COUNTERS AND FLAGS
6475 060422      012737 006406 010052      MOV      #PTRTAB, TXPTR      ;INIT TRANSMIT MESSAGE POINTER
6476 060430      012737 006502 010054      MOV      #PTR13, CMPPTR      ;INIT COMPARE MESSAGE POINTER
6477 060436      012737 006576 010050      MOV      #PTR23, RXPTR      ;INIT RECEIVE MESSAGE POINTER
6478 060444      013737 010056 010112      MOV      CMPTOT, RXMTOT      ;MAKE COMPARE AND RX MESSAGE COUNTS EQUAL
6479 060452      032737 000002 010224      BIT      #DATCKB, PARAM      ;IS IT DATA CHECK
6480 060460      001003      010112      BNE      GTREX      ;BRANCH IF CHECKING
6481 060462      012737 000001      010112      MOV      #1, RXMTOT      ;IF NOCHK MAKE RXCOUNT =1
6482 060470      005037 010134      GTREX:      CLR      CLNSET      ;CLEAR CLEANUP IN PROGRESS SPECIFIER
6483 060474      005037 007644      CLR      CTX      ;CLEAR TX COMPLETE COUNT
6484 060500      005037 007646      CLR      CRX      ;CLEAR RX COMPLETE COUNT
6485 060504      005037 013634      CLR      ILLRSP      ;CLEAR ILLEGAL RESPONSE COUNT
6486 060510      005037 013636      CLR      STARTC      ;CLEAR DDCMP START COUNT
6487 060514      005037 013640      CLR      ERRTHR      ;CLEAR ERROR THRESHHOLD RESPONSE COUNT
6488 060520      005037 013642      CLR      RXMIT      ;CLEAR RETRANSMIT COUNT
6489 060524      012737 007672 007650      MOV      #RXSTAK, RSPTRS      ;POINT TO START OF RX SAVE AREA
6490 060532      012737 007672 007652      MOV      #RXSTAK, RSPTRE      ;AGAIN. POINT TO START OF RX SAVE AREA
6491 060540      012737 007656 007654      MOV      #TXSTAK, TSPTR      ;POINT TO START OF TX SAVE AREA
6492 060546      005037 010230      CLR      FLAG      ;CLEAR FLAG
6493 060552      005037 010116      CLR      OPVAR      ;CLEAR NO BUFFER COUNTER
6494 060556      005037 010120      CLR      OPVAR1      ;CLEAR OPVAR1
6495 060562      005037 010122      CLR      PSCNT      ;CLEAR PASS COUNT
6496 060566      005037 010124      CLR      ERRCNT      ;CLEAR ERROR COUNT
6497 060572      005037 010114      CLR      LNCNT      ;CLEAR COUNTER THAT IS USED FOR STATUS
6498      ;INITIALIZE COMMUNICATIONS DEVICE
6499 060576      004737 024244      JSR      PC, LOGDVI      ;LOG ABOUT TO INIT DEVICE
6500 060602      004737 063024      JSR      PC, DVINIT      ;INIT DEVICE
6501
6502 060606      012737 177777 007642      GTRX2:      MOV      #-1, INDEX      ;MAKE INDEX =-1
6503 060614      013737 010150 010164      GTRX2C:      MOV      CURCC, TEMP      ;
6504 060622      032737 000002 010224      BIT      #DATCKB, PARAM      ;IS THERE DATA CHECKING
6505 060630      001005      BNE      GTRX2A      ;BRANCH IF CHECKING
6506 060632      012737 001000 010150      MOV      #BUFLIM, CURCC      ;SET UP CHAR COUNT TO 'BUFLIM'
6507 060640      005037 010164      CLR      TEMP      ;
6508 060644      004737 030402      GTRX2A:      JSR      PC, GTVIND      ;GET VALID INDEX
6509 060650      022737 000010 007642      CMP      #8, INDEX      ;IS IT 8
6510 060656      001422      BEQ      GTRX2B      ;YES... ALL DONE GO EXECUTE MODE
6511
6512      ;GET RXBUF PTR FIGURE
6513
6514 060660      012737 005406 010170      MOV      #RXBUF, TEMP2      ;TEMP = 0 FOR PTP OR MTP/W NO CHK
6515 060666      013737 007642 010046      MOV      INDEX, MPLY      ;INDEX X TEMP+ RXBUF ADDR =
6516 060674      004737 030356      JSR      PC, MPLY      ;NEW RXBUF ADDR.
6517 060700      013737 010170 010156      MOV      TEMP2, CURADD      ;SET UP RX BUFFER ADDRESS
6518
6519      ;GET CURRENT POINTER FIGURE
6520
6521 060706      004737 030474      JSR      PC, GRPTCP
6522
6523      ;GO LOAD '33' TO BUFFER
6524
6525 060712      012737 000010 010146      MOV      #10, MSGTYP      ;SET UP FOR 33 TO FILL RX BUFFERS
6526 060720      004737 027520      JSR      PC, BLDBUF      ;CLEAR RX BUFFER
6527 060724      013702 010216      GTRX2B:      MOV      MODTYP, R2
6528 060730      006302      ASL      R2
6529 060732      000172 010232      JMP      @MODE(R2)      ;MODE DISPATCH

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 196
RECEIVE MODE SECTION

```

6531 .SBTTL RECEIVE MODE SECTION
6532 :++
6533 :FUNCTIONAL DESCRIPTION:
6534 :RECEIVE-ONLY (OR ONE-WAY-IN) ROUTINE
6535 :IN THIS MODE OF TESTING THE DEVICE'S RECEIVER IS ENABLED IN EXPECTATION
6536 :OF RECEIVING A MESSAGE. AFTER RECEIVING AN "EXPECTED" NUMBER OF
6537 :MESSAGES, THE DATA RECEIVED CAN BE COMPARED AGAINST A LIST OF "EXPECT
6538 :TO RECEIVE" MESSAGES IF DATA-CHECKING IS ENABLED.
6539 :
6540 :SUBORDINATE ROUTINES USED:
6541 : "ALLTR"
6542 :
6543 :CALLING SEQUENCE:
6544 :JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
6545 :--
6546 :
6547 060736 052737 000104 010230 RXONLY: BIS #QRX+ERX,FLAG ;SET UP RX QUE: RX QUEUED, RX EXPECTED
6548 060744 004737 030160 JSP PC,LCPRLS ;LOAD CPTRLS (RX PTRS)
6549 060750 004737 030112 JSR PC,RXQUAL ;GO QUE ALL VALID RX'S
6550 060754 005037 010154 RXON3: CLR CPTR
6551 060760 000137 061110 JMP ALLTR ;GO RX.
6552

```

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      .SBTTL      TRANSMIT MODE SECTION
6554
6555      : **
6556      : FUNCTIONAL DESCRIPTION:
6557      : TRANSMIT-ONLY (OR ONE-WAY-OUT) ROUTINE
6558      : IN THIS MODE OF TESTING A LIST OF MESSAGES IS TRANSMITTED WITHOUT
6559      : EXPECTING ANY DATA TO BE RECEIVED. A REPETITION COUNT CAN BE
6560      : SPECIFIED TO REPETITIVELY TRANSMIT THE LIST.
6561
6562      : SUBORDINATE ROUTINES USED:
6563      : "ALLTR"
6564
6565      : CALLING SEQUENCE:
6566      : JMP      @MODE(R2)      ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
6567      : --
6568
6569      060764 042737 000002 010224 TXONLY: BIC      #JATCKB,PARAM      ;SET NOCHECK
6570      060772 004737 030306 TXON2:  JSR      PC,LCPTLS      ;LOAD TX POINTERS AND TX COUNTS
6571      060776 052737 000210      BIS      #QTX+ETX,FLAG      ;TX QUEUED, TX COMPLETE EXPECTED
6572      061004 004737 030262      JSR      PC,CLRPLS      ;CLEAR RXPRT LIST
6573      061010 012737 000010      MOV      #8,INDEX
6574      061016 000137 061110      JMP      ALLTR      ;GO TX.
6575

```

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      .SBTTL      PASSIVE MODE SECTION
6577
6578      :++
6579      : FUNCTIONAL DESCRIPTION:
6580      : PASSIVE MODE SECTION
6581      : IN THIS MODE OF TESTING, THE DEVICE'S RECEIVER IS ENABLED IN
6582      : EXPECTATION OF RECEIVING A MESSAGE. THEN EVERY TIME A MESSAGE IS
6583      : RECEIVED, A MESSAGE IS TRANSMITTED. DATA CHECKING CAN BE DONE ON THE
6584      : RECEIVED DATA.
6585
6586      : SUBORDINATE ROUTINES USED:
6587
6588      : "ALLTR"
6589
6590      : CALLING SEQUENCE:
6591      : JMP      @MODE(R2)      ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
6592      :--
6593
6594      061022 004737 030306      PLCK: JSR      PC,LCPTLS      ;LOAD TX POINTERS AND TX COUNTS
6595      061026 004737 030160      JSR      PC,LCPRLS      ;SET UP CPTRR TO REC POINTERS
6596      061032 052737 000104 010230 BIS      #QRX+ERX,FLAG ;RX QUEUED AND EXPECT RX
6597      061040 004737 030112      JSR      PC,RXQUAL      ;QUE ALL
6598      061044 000137 061110      JMP      ALLTR      ;AND GO RX FIRST MSG.
6599

```

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      .SBTTL      ACTIVE MODE SECTION
6601
6602      :++
6603      : FUNCTIONAL DESCRIPTION:
6604      : ACTIVE MODE SECTION
6605      : IN THIS MODE OF TESTING, A LIST OF MESSAGES IS TRANSMITTED AND THOSE
6606      : MESSAGES ARE EXPECTED TO BE RECEIVED. RECEIVED DATA CAN BE COMPARED
6607      : AGAINST "EXPECTED" DATA IF DATA-CHECKING IS ENABLED.
6608      : NOTE: IF BOTH ENDS OF THE LINK ARE IN ACTIVE MODE, THEN THE
6609      : LINK MUST BE A FULL DUPLEX LINK!
6610
6611      : SUBORDINATE ROUTINES USED:
6612
6613      : "ALLTR"
6614
6615      : CALLING SEQUENCE:
6616      : JMP      @MODE(R2)      ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
6617      :--
6618
6619      061050      ALCK:
6620      061050      032737 000002 010224      BIT      #DATCKB,PARAM      ;IS IT DATA CHECK
6621      061056      001003      BNE      1$      ;BRANCH IF CHECK
6622      061060      013737 010074 010112      MOV      TXMTOT,RXMTOT      ;IF NOCH MAKE RX=TX
6623      061066      004737 030306      1$:      JSR      PC,LCPTLS      ;LOAD TX POINTERS AND COUNTS
6624      061072      004737 030160      JSR      PC,LCPRLS      ;LOAD RX POINTERS
6625      061076      052737 000314 010230      BIS      #QRX+QTX+ETX+ERX,FLAG      ;RX QUEUED, TRANSMIT QUEUED
6626      ;EXPECT RECEIVE, EXPECT TRANSMIT
6627      061104      004737 030112      JSR      PC,RXQUAL      ;QUE UP 1 RX BUFFER FOR ALL VALID LINES
6628

```

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 200
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

.SBTTL TRANSMIT - RECEIVE FOR ALL STANDARD MODES

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 QUEUED'.
- 2.) IF TX BUFFERS ARE TO BE QUEUED, TELL DEVICE
CODE TO QUE THEM AND LOG 'TRANSMIT QUEUED'.
- 3.) WAIT FOR EITHER RECEIVE 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 QUEUED 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 QUEUED 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 QUEUED 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	CPIR,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,CPIR	: 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	CPIR,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,CPIRR	: 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	CPIRR,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,CPIRR	: 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    DVRT          ;IS IT ALL DONE
6743 061510 001007              BNE     ALCK3          ;NO. GO CHECK TX
6744 061512 042737 000004 010230 BIC     #QTX,FLAG      ;CLEAR THE RX QUEUED FLAG
6745 061520 005037 010152      CLR     CPTR          ;YES. CLEAR POINTER
6746 061524 004737 030434      JSR     PC,LDRPLS     ;AND RELOAD LIST
6747
6748                               ;CHECK FOR MESSAGE TRANSMITTED
6749 061530 032737 000010 010230 ALCK3: BIT     #QTX,FLAG      ;IS A TX QUEUED?
6750 061536 001467              BEQ     ALCK4          ;IF NOT TX THEN GO BACK (NEITHER RX NOR TX)
6751 061540 013737 010064 010170 MOV     DVTXA,TEMP2     ;TRANSMIT COMPLETED- LOG IT
6752 061546 013737 010066 010172 MOV     DVTCC,TEMP3     ;LOG TX COMPLETED
6753 061554 013737 010070 007636 MOV     DVTTB,TRIBN    ;LOG EVENT-
6754 061562 004737 024162      JSR     PC,LOGTXC      ;TRANSMIT COMPLETE
6755
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 010632      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          ;IF NO MORE MESSG TO RX FOR THIS LINE
6764 061624 001403              BEQ     ALCK3D          ;EXIT WITHOUT RESETTING QRX (RX STILL QUEUED)
6765
6766 061626 052737 000104 010230 ALCK3D: BIS    #QTX+ERX,FLAG ;AND SET THE RX FLAGS
6767 061634 004737 030744      JSR     PC,GATCFL
6768 061640 005737 010072      TST     DVTCT
6769 061644 001007              BNE     ALCK3C
6770 061646 000137 061742      JMP     CMPSR
6771
6772                               ;NOT PASSIVE MODE
6773 061652 004737 030744      ALCK3A: JSR    PC,GATCFL      ;GET ALL TX COUNTS FROM LIST
6774 061656 005737 010072      TST     DVTCT          ;IS IT ALL DONE? ALL TRANSMISSIONS COMPLETE?
6775 061662 001404              BEQ     ALCK3B          ;IF NOT GO BACK TO 5
6776 061664 004737 031102      ALCK3C: JSR    PC,GETIND     ;ALL TRANSMISSIONS COMPLETE FOR CURRENT LINE
6777 061670 000137 061110      JMP     ALCK5
6778 061674 005037 010154      ALCK3B: CLR     CPTR          ;NOW GET ANOTHER VALID LINE.
6779 061700 042737 000010 010230 BIC     #QTX,FLAG      ;CLEAR POINTER
6780 061706 032737 000002 010224 BIT     #DATCKB,PARAM  ;TRANSMIT COMPLETE, NO LONGER QUEUED.
6781 061714 001405              BEQ     ALCK4A          ;IS IT DAT CHECK
6782 061716 004737 030672      ALCK4: JSR     PC,GARFPL     ;IF NOT THEN END WORKING RX.
6783 061722 005737 010152      TST     CPTR
6784 061726 001356              BNE     ALCK3C
6785 061730 005737 010154      ALCK4A: TST    CPTR
6786 061734 001402              BEQ     CMPSR
6787 061736 000137 061210      JMP     ALCK2
6788
6789
6790
6791

```

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

MACV11 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	CMPSR: 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	CMPTTR,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	CPTTR,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,CPTTR	; 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      ERRSOF1 ENUM,EDDL,ERR10      ;PRINT ERROR
6854 062074 104457      TRAP      C$ERSOFT
6855 062076 000001      .WORD      1
6856 062100 017661      .WORD      EDDL
6857 062102 023430      .WORD      ERR10
6858 062104 004737 024330      JSR      PC,LOGCML      ;LOG LENGTH ERROR
6859
6860 062110 005037 010174      CMPS7:  CLR      TEMP4      ;CLEAR BAD BYTE COUNTER
6861 062114 012737 000001 010162      MOV      #1,OFFSET      ;SET OFFSET BYTE COUNT TO 1
6862 062122 122123      CMPS1:  CMPB     (R1)+,(R3)+      ;COMPARE RX WITH EXPETED
6863 062124 001422      BEQ      CMPS6      ;IF EQUAL THEN GO TO 6
6864
6865 062126 005237 010174      CMPS2:  INC      TEMP4      ;INC BAD COUNT
6866 062132 023727 010174 000005      CMP      TEMP4,#5      ;IS IT MORE THEN 5
6867 062140 101014      BHI      CMPS6      ;IF SO GO FOR MORE
6868 062142 114337 010204      MOVB     -(R3),GOOD      ;STORE GOOD BYTE FOR ERROR
6869 062146 114137 010205      MOVB     -(R1),BAD      ;STORE BAD BYTE FOR ERROR
6870 062152 005237 010124      INC      ERRCNT
6871      000002      ENUM = ^D<ENUM+1>      ;INCREMENT ERROR NUMBER
6872 062156      ERRSOF1 ENUM,EDDE,ERR1      ;REPORT CMPARISON FAILURE TO OPR.
6873 062156 104457      TRAP      C$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      ENUM = ^D<ENUM+1>      ;INCREMENT ERROR NUMBER
6887 062214      ERRSOF1 ENUM,EDDE,ERR2      ;REPORT # OF MISMATCHES FOR MESSAGE
6888 062216 000003      TRAP      C$ERSOFT
6889 062220 017716      .WORD      3
6890 062222 023402      .WORD      EDDE
6891 062224 004737 024346      CMPS5:  JSR      PC,LOGCMD      ;LOG DATA ERROR IN COMPARE
6892 062230      CMPS5A:
6893 062230 005337 010110      DEC      DVRC1
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

```

:++
: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

```

-----
CMPSEX: INC      PSCNT      ;BUMP PASS COUNT
MOV      OPVAR,TEMP4      ;LOG NO BUFFS
MOV      PSCNT,TEMP2      ;LOG PASS COUNT
MOV      ERRCNT,TEMP3
JSR      PC,LOGEOP      ;LOG END OF PASS

CMP      #-1,RPASS      ;SEE IF RPASS=-1
BEQ      1$      ;IF IT IS DON'T DECREMENT, LOOP FOREVER
DEC      RPASS      ;DEC PASS COUNT
BEQ      2$      ;IF DONE GOTO DCLT> PROMPT
JMP      GTRX2      ;ELSE GO BACK AND DISPATCH
JSR      PC,HLTTRB      ;GO HALT ALL LINES BEFORE GOING BACK
JMP      GTRAS      ;WHEN RPASS=0 GO BACK TO 'DCLT>'

1$:
2$:

```

```

6897
6898
6899
6900
6901
6902
6903
6904
6905
6906
6907
6908
6909
6910
6911
6912
6913
6914 062240 005237 010122
6915
6916 062244 013737 010116 010174
6917 062252 013737 010122 010170
6918 062260 013737 010124 010172
6919 062266 004737 024372
6920
6921 062272 022737 177777 010226
6922 062300 001403
6923 062302 005337 010226
6924 062306 001402
6925 062310 000137 060606
6926 062314 004737 064020
6927 062320 000137 054512
6928
6929

```

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 SUCESSFUL.

```

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

```

```

062324
062324
062324 012746 015457
062330 012746 000001
062334 010600
062336 104417
062340 062706 000004
062344 000137 054512

```

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

.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 QUEUED 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$:      CLR    R2
        CMPB   #377,OPBUF(R2) ;NOW GET CHAR COUNT
        BEQ    3$
        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,MODIYP	;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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 209
LISTEN MODE SECTION

```

7032 .SBTTL LISTEN MODE SECTION
7033
7034 : **
7035 : FUNCTIONAL DESCRIPTION:
7036 : LISTEN MODE SECTION
7037 : IN THIS MODE, THE "LISTEN" END OF THE LINK PRINTS ALL OF THE MESSAGES
7038 : RECEIVED BY THE DEVICE ON THE OPERATOR'S CONSOLE. IF THE MESSAGE
7039 : RECEIVED IS AN "EXIT" MESSAGE, THEN THE NODE ENTERS "TALK" MODE.
7040
7041 : SUBORDINATE ROUTINES USED:
7042
7043 : "DVRXQ" - QUE RECEIVE BUFFER SPACE TO DEVICE
7044 : "LOGRXQ" - LOG RECEIVE BUFFER QUED TO EVENT LOG
7045 : "DVTXRX" - WAIT FOR RX TO COMPLETE
7046 : "LOGRXC" - LOG RX COMPLETE TO EVENT LOG
7047
7048 : CALLING SEQUENCE:
7049 : JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
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 MOV #LISP,-(SP)
7057 062626 012746 000001 MOV #1,-(SP)
7058 062632 010600 MOV SP,R0
7059 062634 104417 TRAP C$PNTF
7060 062636 062706 000004 ADD #4,SP
7061 062642 012737 002520 010104 LISCKA: MOV #OPBUF,DVRXA ;SET DEVICE UP TO REC AT OPBUF
7062 062650 012737 002520 010170 MOV #OPBUF,TEMP2
7063 062656 012737 000122 010106 MOV #82.,DVRCC ;SET UP CHAR COUNT TO 82.
7064 062664 012737 000122 010172 MOV #82.,TEMP3
7065 062672 052737 000104 010230 BIS #QRY+ERX,FLAG ;SET UP FLAG
7066 062700 005037 010154 CLR CPTR ;CLEAR THE TX.
7067
7068 062704 004737 064174 JSR PC,DVRXQ ;QUE RX
7069 062710 004737 024200 JSR PC,LOGRXQ
7070
7071 062714 004737 064340 JSR PC,DVTXRX ;GO TO DEVICE RX. SUBROUTINE
7072
7073 052720 013737 010104 010170 MOV DVRXA,TEMP2
7074 062726 013737 010106 010172 MOV DVRCC,TEMP3 ;SET UP ADDR.AND CC.
7075 062734 004737 024216 JSR PC,LOGRXC ;LOG COMPLETED
7076 062740 063737 010104 010106 ADD DVRXA,DVRCC
7077 062746 105077 125134 CLRB @DVRCC
7078 062752 PRINTF #OPBFPT
7079 062752 012746 002514 MOV #OPBFPT,-(SP)
7080 062756 012746 000001 MOV #1,-(SP)
7081 062762 010600 MOV SP,R0
7082 062764 104417 TRAP C$PNTF
7083 062766 062706 000004 ADD #4,SP
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

```

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	MOV	#TAL,MODTYP	;CHANGE MODE TO TALK
7089	063020	000137	060606		JMP	GTRX2	;RETURN TO DISPATCHER
7090							
7091							

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

MACV11 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:

;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

063024

063024 022737 000004 010216
063032 001002
063034 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

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

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      ERRSOF 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$:      CLR      VSTAT      ;NEGATE 'STATUS DATA VALID'
7158 063136 005037 011514      INC      ENRDST      ;ASSERT READ STATUS ENABLED
7159 063142 005237 011512      BIC      #ININT+OTINT,FLAG ;CLEAR INTERRUPT INDICATORS
7160 063146 042737 000003 010230
7161      DVIN1:
7162 063154      ;SPECIFY SYSTEM INITIALIZATION (COMMAND PAIR 0,0)
7163
7164      BIS      #INITC,FLAG      ;SPECIFY WAITING FOR INITIALIZATION
7165 063154 052737 000400 010230      JSR      PC,TOORIO      ;REQUEST PERMISSION
7166 063162 004737 065422      ; TO SEND COMMAND TO KMS
7167      ;PERMISSION GRANTED TO SEND COMMAND
7168      BIC      #177757,@SEL2      ;SPECIFY COMMAND PAIR 0,0
7169 063166 042777 177757 130202      CLR      @SEL4      ;NO PARAMETERS
7170 063174 005077 130202      CLR      @SEL6      ;REQUIRED
7171 063200 005077 130202      BICB     #RDYI,@BSEL2      ;ASSERT READY TO TRANSFER COMMAND
7172 063204 142777 000020 130164      ;WAIT FOR INITIALIZATION COMPLETION
7173      JSR      PC,TOORO      ;WAIT FOR RESPONSE
7174 063212 004737 065622      BIT      #INITC,FLAG      ;IS INITIALIZATION COMPLETED ?
7175 063216 032737 000400 010230      BEQ      DVIN2      ;YES. CONTINUE
7176 063224 001401      BR      DVIN1      ;RETRY INITIALIZATION
7177 063226 000752
7178      DVIN2:
7179 063230      ;SPECIFY LOOP PARAMETER
7180
7181      MOV      #377,TEMP      ;ASSUME INTERNAL LOOPBACK NEGATED
7182 063230 012737 000377 010164      CMP      #1,MLTYP      ;IS INTERNAL LOOPBACK SPECIFIED ?
7183 063236 022737 000001 010220      BNE      10$      ;NO. ASSUMPTION WAS CORRECT
7184 063244 001002      CLR      TEMP      ;SPECIFY INTERNAL LOOPBACK ASSERTED
7185 063246 005037 010164      10$:      JSR      PC,TOORIO      ;GIVE DEVICE INTERNAL LOOPBACK MODE SPECIFIER
7186 063252 004737 065422      ;REQUEST PERMISSION TO SEND COMMAND
7187      ;PERMISSION GRANTED
7188      BIC      #177757,@SEL2      ;CLEAR ALL BUT RDYI
7189 063256 042777 177757 130112      BISB     #142,@BSEL2      ;SPECIFY WRITE DATA TO DEVICE
7190 063264 152777 000142 130104      MOV      #KMLTYP,@SEL4      ;SPECIFY DATA ADDRESS IN DEVICE
7191 053272 012777 007726 130102      MOV      TEMP,@BSEL6      ;SPECIFY DEVICE LOOPBACK MODE
7192 063300 113777 010164 130100      BICB     #RDYI,@BSEL2      ;SPECIFY READY TO TRANSFER COMMAND
7193 063306 142777 000020 130062      CMP      #1,MLTYP      ;IS INTERNAL LOOPBACK SPECIFIED ?
7194 063314 022737 000001 010220      BEQ      20$      ;YES. OMIT SPECIFYING MODEM CONTROLLER ADDRESS
7195 063322 001420
7196      ;SPECIFY MODEM CONTROLLER (DM-11) CSR ADDRESS
7197
7198      JSR      PC,TOORIO      ;REQUEST PERMISSION TO SEND TO FIRMWARE
7199 063324 004737 065422      ;PERMISSION GRANTED
7200      BIC      #177757,@SEL2      ;CLEAR ALL BUT RDYI
7201 063330 042777 177757 130040      BISB     #7,@BSEL2      ;SPECIFY COMMAND PAIR (0,7)
7202 063336 152777 000007 130032      MOV      MODCSR,@SEL4      ;SPECIFY DM11-BA CSR ADDRESS
7203 063344 013777 013426 130030

```

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  BICB    #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,GTVINI          :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 #TXID,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

```

**
: 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
      BIS    #STAT,FLAG  ;SPECIFY WAITING FOR STATUS REPORT
      BIC    #177757,@SEL2 ;CLEAR ALL BUT RDYI
      BISB   #BIT6+BIT5,@BSEL2 ;SPECIFY REQUEST LINE STATUS (3,0)
      MOVB   TRIBN,@BSEL3 ;SPECIFY LINE NO.
      BICB   #RDYI,@BSEL2 ;INDICATE READY TO TRANSFER COMMAND
                        ;WAIT FOR STATUS TO BE RETURNED
20$:   JSR    PC,TOORO   ;WAIT FOR MESSAGE FROM DEVICE
                        ;MESSAGE RECEIVED
      BIT    #STAT,FLAG  ;DID A STATUS REPORT OCCUR ?
      BNE    20$         ;NO.KEEP WAITING
      BIS    #STAT,FLAG  ;SPECIFY WAITING FOR STATUS REPORT
30$:   JSR    PC,TOORO   ;WAIT FOR SECOND HALF OF STATUS REPORT
      BIT    #STAT,FLAG  ;DID A STATUS REPORT OCCUR ?
      BNE    30$         ;NO. KEEP WAITING
      INC    VSTAT       ;ASSERT VALID STATUS DATA
40$:   RTS    PC        ;RETURN

```

```

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 063704
7310 063704 005737 011512
7311 063710 001442
7312 063712 004737 065422
7313
7314 063716 052737 002000 010230
7315 063724 042777 177757 127444
7316 063732 152777 000140 127436
7317 063740 113777 007636 127432
7318 063746 142777 000020 127422
7319
7320 063754 004737 065622
7321
7322 063760 032737 002000 010230
7323 063766 001372
7324 063770 052737 002000 010230
7325 063776 004737 065622
7326 064002 032737 002000 010230
7327 064010 001372
7328 064012 005237 011514
7329 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 : FUNCTIONAL DESCRIPTION:
7334 :   HLTTRB - THIS SUBROUTINE SHUTS DOWN THE ESTABLISHED LINE
7335 :
7336 : INPUTS:
7337 :   PARAM - PARAMETERS. THIS ROUTINE CHECKS THE DATCKB BIT
7338 :   TRIBLS - ESTABLISHED LINE NUMBER
7339 :   FLAG - BITS ININT AND OTINT INDICATE INTERRUPTS
7340 :
7341 : SUBORDINATE ROUTINES USED:
7342 :   GTVIND - COPY VALID LINE NUMBER TO TRIBN
7343 :   TOORIO - REQUEST INPUT INTERRUPT. WAIT FOR INTERRUPT OR TIME OUT
7344 :   OUTHDL - HANDLES OUTPUT INTERRUPT DATA
7345 :
7346 : CALLING SEQUENCE:
7347 :   JSR    PC,HLTTRB
7348 :--
7349
7350 064020 032737 000J02 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
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 :
7410 :
7411 :
7412 :
7413 :
7414 :
7415 :
7416 :
7417 :
7418 :
7419 :
7420 :
7421 :
7422 :
7423 :
7424 :
7425 :
7426 :
7427 :
7428 :
7429 :
7430 :
7431 :
7432 :
7433 :
7434 :
7435 :
7436 :
7437 :
7438 :
7439 :
7440 :

```

064174	032737	000004	010230	DVRXQ:	BIC	#QRX,FLAG	;RX QUEUED
064174	001455				BEQ	DVREX	;IF NOT RX THEN EXIT
064204	042737	000004	010230		BIC	#QRX,FLAG	;ELSE QUE RX
							;CLEAR FLAG FOR RX QUEUED
064212	013702	007636			MOV	TRIBN,R2	;GET CHANNEL NUMBER (WHICH IS MEMBER NUMBER)
064216	006302				ASL	R2	;MULTIPLY BY 8
064220	006302				ASL	R2	;FOR OFFSET IN TABLE FOR THIS BDL MEMBER
064222	006302				ASL	R2	; (8 BYTES PER MEMBER IN TABLE)
064224	062702	013434			ADD	#RXBDL,R2	;TABLE ADDRESS FOR RECEIVE BDL MEMBERS
064230	010237	013432			MOV	R2,BDLMAD	;CALCULATE BDL MEMBER ADDRESS, SAVE AT BDLMD
064234	013737	007636	013430		MOV	TRIBN,BDLMNM	;SAVE THE CHANNEL NUMBER, BDL MEMBER NUMBER
064242	005237	013430			INC	BDLMNM	;CALCULATE BDL MEMBER NUMBER (CHAN + 1)
							;RANGE IS NOW 1-8 FOR CHAN 0-7
							;LOAD BUFFER DESCRIPTOR LIST MEMBER...
064246	012722	100000			MOV	#BIT1,(R2)+	;SET STATUS WORD IN BDL MMBR BLOCK (RX QUEUED)
064252	013722	010104			MOV	DVRXA,(R2)+	;DVRXA CONTAINS THE RECEIVE BUFFER ADDRESS
064256	013722	010106			MOV	DVRCC,(R2)+	;DVRCC CONTAINS LENGTH OF THAT BUFFER
064262	005022				CLR	(R2)+	;CLEAR DDCMP MESSAGE NUMBER FIELD
064264	004737	065422			JSR	PC,TOORIO	;SPECIFY BDL MEMBER FOR RECEIVE BUFFER
							;REQUEST PERMISSION TO SEND COMMAND TO FIRMWARE
							;PERMISSION GRANTED
064270	042777	177757	127100		BIC	#177757,@SEL2	;CLEAR ALL EXCEPT RDYI
064276	152777	000101	127072		BISB	#101,@BSEL2	;SPECIFY COMMAND PAIR (2,1)
064304	113777	007636	127066		MOVB	TRIBN,@BSEL3	;SPECIFY LINE NUMBER
064312	005077	127064			CLR	@SEL4	;CLEAR SEL4
064316	105077	127064			CLRB	@BSEL6	;CLEAR BSEL6
064322	113777	013430	127060		MOVB	BDLMNM,@BSEL7	;SPECIFY BDL MEMBER NUMBER
							;BDL MEMBER NUMBER RANGE IS 1-8
064330	142777	000020	127040		BICB	#RDYI,@BSEL2	;ASSERT READY TO TRANSFER COMMAND
064336	000207			DVREX:	RTS	PC	;RETURN TO CALLER

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 218
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

7441 .SBTTL                                DEVICE TRANSMIT AND RECEIVE SUBROUTINE
7442
7443 : **
7444 : FUNCTIONAL DESCRIPTION:
7445 : DVTXRX-DEVICE TRANSMIT AND RECEIVE ROUTINE.
7446 : THIS CODE QUEUES THE TRANSMIT BUFFER TO THE DEVICE (IF NEEDED).
7447 : THE CODE THEN WAITS FOR A TX COMPLETE, AN RX COMPLETE OR A
7448 : RETRANSMIT REQUEST. THE CODE REPORTS A TIME OUT ERROR IF NONE
7449 : OF THESE IS REPORTED BACK IN 60 SECONDS. AFTER REPORTING ERROR,
7450 : THE TIMER IS RE STARTED AND THE DEVICE WILL CONTINUE TO WAIT FOR
7451 : AN INTERRUPT.
7452 : A NAK RESPONSE GENERATES A RETRANSMIT REQUEST. THIS CODE REQUEUES
7453 : THE TRANSMIT BUFFER TO THE DEVICE AND RESTARTS THE TIMER IF A
7454 : RETRANSMIT IS REQUESTED.
7455
7456 : INPUTS:
7457 :   "DVTXA" = ADDRESS OF TRANSMIT MSG.
7458 :   "DVTCC" = BYTE COUNT OF TRANSMIT MSG.
7459 :   "QTX" BIT = SET IF TRANSMIT REQUESTED
7460 :   "ETX" BIT = SET IF TRANSMIT EXPECTED
7461 :   "ERX" BIT = SET IF RECEIVE EXPECTED
7462 :   "RXMIT" BIT = SET IF RETRANSMIT REQUESTED
7463
7464 : OUTPUTS:
7465 :   "DVTXA" = ADDRESS OF TX MSG. COMPLETED
7466 :   "DVTCC" = BYTE COUNT OF TX MSG. COMPLETED
7467 :   "QTX" = SET IF TX QUEUED
7468 :   "DVRXA" = ADDRESS OF RX MSG. COMPLETED
7469 :   "DVRCC" = BYTE COUNT OF RX MSG. COMPLETED
7470 :   "QRX" = SET IF RX QUEUED
7471
7472 : SUBORDINATE ROUTINES USED: TCORIO, OUTHDL
7473
7474 : CALLING SEQUENCE: JSR PC,DVTXRX
7475 : --
7476
7478 064340 032737 000010 010230 DVTXRX: BIT    #QTX,FLAG    ;IS THERE A TRANSMIT QUEUED?
7479 064346 001457                BEQ     DVTIR3        ;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 NUMBEF OF MSGS TO TX=1
7504 064472 113777 013430 126710 MOVB   BDLMMN,@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
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 TRANS*IT COMMAND
7515 064534 013702 007654      MOV    TSPTTR,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,TSPTTR    ;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
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    ;TIME EXPIRED. REPORT ERROR
7540 064660 005037 010172      CLR    TEMP3      ;SPECIFY ERROR MESSAGE
7541 064664 017737 126502 010174      MOV    @SELO,TEMP4    ;SPECIFY ERROR CODE
7542 064672 017737 126500 010176      MOV    @SEL2,TEMP5    ;COPY SELO
7543 064700 113737 010175 010173      MOVB   TEMP4+1,TEMP3+1 ;COPY SFL2
7544 064706 005237 010124      INC    ERRCNT      ;COPY LINE NUMBER FROM TEMP4
7545 064712 004737 024226      JSR    PC,LGDVE      ;INCREMENT ERROR COUNT
7546
7547 064716
7548 064716 104457
7549 064720 000005
7550 064722 022223
7551 064724 023460
7552 064726 000667

```

DVTR3:

TOINOT:

ENUM =

BR

DVTR3

;REPEAT TIME CHECK

TRAP CSERSOFT
.WORD 5
.WORD DVEM2
.WORD ERR13

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 220
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

7553
7554
7555 064730          TOIN1: BREAK          ;CHECK FOR INTERRUPT
7556 064730 104422          ;ALLOW CNTRL/C EXIT          TRAP C$BRK
7557 064732 032737 000003 010230 TOIN2: BIT  #OTINT+ININT,FLAG ;HAS AN INTERRUPT OCCURRED ?
7558 064740 001730          BEQ TOINOT ;NO. KEEP CHECKING
7559          ;INTERRUPT OCCURRED
7560 064742 042737 000003 010230 BIC  #OTINT+ININT,FLAG ;CLEAR INTERRUPT INDICATORS
7561 064750 132777 000200 126420 BITB  #RDY0,@BSEL2 ;IS THERE A MESSAGE FROM THE DEVICE
7562 064756 001721          BEQ TOINOT ;NO KEEP CHECKING
7563          ;MESSAGE FROM DEVICE
7564 064760 004737 066040          JSR  PC,OUTHDL ;GO PROCESS MESSAGE FROM DEVICE
7565          ;CHECK IF RX OR TX COMPLETED
7566 064764 005737 013642          DVTR4: TST  RXMIT ;IS RETRANSMIT ASSERTED ?
7567 064770 001246          BNE  DVTR3 ;YES. GO RETRANSMIT
7568 064772 005737 007644          TST  CTX ;HAS A TX COMPLETED
7569 064776 001004          BNE  DVTR4A ;YES. GO PROCESS COMPLETION
7570 065000 005737 007646          10$: TST  CRX ;HAS AN RX COMPLETED ?
7571 065004 001077          BNE  DVTR5 ;YES. GO PROCESS COMPLETION
7572 065006 000705          BR  TOINOT ;NO. KEEP CHECKING
7573
7574          DVTR4A: ;TX COMPLETED
7575 065010          MOV  TSPTX,R2 ;POINT TO TX SAVE AREA
7576 065014 014237 007636          MOV  -(R2),TRIBN ;COPY SAVED TX LINE NUMBER
7577 065020 105037 007636          CLRB TRIBN ;CLEAR LOW BYTE
7578 065024 000337 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,TSPTX ;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          MOV  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          ENUM = *D<ENUM+1> ;INCREMENT ERROR NUMBER
7593 065120          ERRSOF1 ENUM,DVEM4,ERR22 ;REPORT ERROR (UNEXPECTED CTX)
7594 065120 104457          TRAP C$ERSOF1
7595 065122 000006          .WORD 6
7596 065124 022330          .WORD DVEM4
7597 065126 023542          .WORD ERR22
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)+,DVTR4A ;UNLOAD TX MESSAGE ADDRESS

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P11 23-DEC-P2 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 NUMBFR FOR LOGGING
7611 065172 052737 000010 010230      BIS      #QTX,FLAG      ;SET QUE-TRANSMIT FLAG BIT
7612 065200      DVTR4C:      DEC      CTX      ;DECREMENT TX COUNT
7613 065200 005337 007644      DVTR5:      TST      CRX      ;ANY RECEIVES COMPLETED ?
7614 065204      BEQ      DVTREX      ;NO. GO RETURN
7615 065204 005737 007646      ;RECEIVE COMPLETED
7616 065210 001503      MOV      RSPTRS,R2      ;POINT TO RECEIVE SAVE AREA
7617      MOV      (R2)+,TEMP      ;COPY SAVED BDL NUMBER
7618 065212 013702 007650      CLRB     TEMP      ;CLEAR LO BYTE
7619 065216 012237 010164      SWAB     TEMP      ;POSITION BDL NUMBER IN LO BYTE
7620 065222 105037 010164      MOV      (R2)+,TRIBN      ;COPY SAVED LINE NUMBER
7621 065226 000337 010164      CLRB     TRIBN      ;CLEAR LO BYTE
7622 065232 012237 007636      SWAB     TRIBN      ;POSITION LINE NUM IN LO BYTE
7623 065236 105037 007636      CMP      R2,#RXSKEN      ;POINTING TO END OF RX SAVE AREA ?
7624 065242 000337 007636      BNE      2$      ;NO. KEEP POINTER
7625 065246 020227 010046      MOV      #RXSTAK,R2      ;YES. RESET POINTER
7626 065252 001002      MOV      R2,RSPTRS      ;SAVE POINTER
7627 065254 012702 007672      BIT      #ERX,FLAG      ;IS RX COMPLETE EXPECTED ?
7628 065260 010237 007650      BNE      DVTREX      ;YES. GO PROCESS RX COMPLETED
7629 065264 032737 000100 010230      ;UNEXPECTED RECEIVE COMPLETION
7630 065272 001025      MOV      #DVEM5,TEMP2      ;RECORD UNEXPECTED RX COMPLETION
7631      MOV      #2,TEMP3      ;SPECIFY ERROR CODE
7632 065274 012737 022407 010170      MOV      TRIBN,TEMP3+1      ;RECORD LINE NUMBER
7633 065302 012737 000002 010172      MOV      TEMP,TEMP4      ;COPY BDL NUMBER
7634 065310 113737 007636 010173      INC      ERRCNT      ;INCREMENT ERROR COUNT
7635 065316 013737 010164 010174      JSR      PC,LGDVE      ;LOG EVENT
7636 065324 005237 010124      ENUM = ^D<ENUM+1>      ;INCREMENT ERROR NUMBER
7637 065330 004737 024226      ERRSOFT ENUM,DVEM5,ERR22      ;REPORT ERROR
7638 065334      TRAP      CSERSOFT
7639 065334 104457      .WORD      7
7640 065336 000007      .WORD      DVEM5
7641 065340 022407      .WORD      ERR22
7642 065342 023542
7643 065344 000423      BR      DVTR5C      ; AND EXIT
7644      ;EXPECTED RX COMPLETED
7645      ;COPY BDL MEMBER NUMBER
7646 065346      ;COMPUTE OFFSET
7647 065346 013703 010164      MOV      TEMP,R3      ;FOR
7648 065352 005303      ; THIS
7649 065354 006303      ; MEMBER
7650 065356 006303      ; POINT TO THIS MEMBER IN BDL AREA
7651 065360 006303      ;CLEAR BDL STATUS WORD
7652 065362 062703 013434      MOV      (R3)+,DVRXA      ;UNLOAD RX MESSAGE ADDRESS
7653 065366 005023      MOV      (R3)+,DVRCC      ;UNLOAD RX MESSAGE CHAR COUNT
7654 065370 012337 010104      MOV      TRIBN,DVRTB      ;STORE LINE NUMBER FOR LOGGING
7655 065374 012337 010106      BIS      #QRX,FLAG      ;SET FLAG TO QUE ANOTHER RECEIVER
7656 065400 013737 007636 010102      DVTR5C:      DEC      CRX
7657 065406 052737 000004 010230      DVTREX:      RTS      PC
7658 065414
7659 065414 005337 007646
7660 065420 000207
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 OUTHDL 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		BISB	#IEO!IEI,@BSEL0		;ENABLE INTERRUPTS
7696	065434				TOOR11:				
7697	065434	152777	000200	125730		BISB	#RQ1,@BSEL0		;ASSERT RQ1
7698	065442	012737	002000	010270		MOV	#2000,TIMER1		;SET TIMER
7699	065450	005737	010270		TOOR12:	TST	TIMER1		;IS TIME EXPIRED ?
7700	065454	001027				BNE	TOOR13		;NO. CONTINUE
7701									;TIME EXPIRED WAITING FOR RDYI
7702	065456	142777	000200	125706		BICB	#RQ1,@BSEL0		;YES. NEGATE RQ1
7703	065464	012737	022275	010170		MOV	#DVEM3,TEMP2		;REPORT TIME OUT WAITING FOR RDYI
7704	065472	005037	010172			CLR	TEMP3		;SPECIFY ERROR CODE
7705	065476	017737	125670	010174		MOV	@SEL0,TEMP4		;COPY BSEL0
7706	065504	017737	125666	010176		MOV	@SEL2,TEMP5		;COPY BSEL2
7707	065512	005237	010124			INC	ERRCNT		;INCREMENT ERROR COUNT
7708	065516	004737	024226			JSR	PC,LGDVE		;LOG THE DEVICE ERROR
7709		000010			ENUM =	*D<ENUM+1>		;INCREMENT ERROR NUMBER	
7710	065522					ERRSOFT ENUM,DVEM3,ERR13		;PRINT THE ERROR	
7711	065522	104457							TRAP C\$ERSOFT
7712	065524	000010							.WORD 8
7713	065526	022275							.WORD DVEM3
7714	065530	023460							.WORD ERR13
7715	065532	000740							
7716	065534				TOOR13:	BR	TOOR11		;GO BACK AND TRY AGAIN
7717	065534					BREAK			;ALLOW CNTRL/C EXIT
7718	065534	104422							TRAP C\$BRK
7719									
7720	065536	032737	000003	010230		BIT	#OTINT+ININT,FLAG		;CHECK FOR AN INTERRUPT
7721	065544	001741				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

TOOR10 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 (RDY0)
7724 065554 132777 000200 125614      BITB      #RDY0,@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      JSR      PC,OUTHDL      ;READ MESSAGE FROM FIRMWARE
7728 065570 132777 000020 125600      BITB      #RDY1,@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      #RDY1,@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      #RQ1,@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

.SBTT, TOORO WAIT FOR AN EXPECTED CONTROL MESSAGE FROM DEVICE

```

:++
: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:
:   LGDVE - LOG DEVICE ERRORS
:   OUTHDL - OUTPUT INTERRUPT HANDLER

```

```

:CALLING SEQUENCE:
:   JSR PC,TOORO
:--

```

7770	065622	011637	010210		TOORO:	MOV	(SP),PCADD		:SAVE ADDR. OF CALLING ROUTINE
7771	065626	112777	000021	125536		MOVB	#IEO+IEI,@BSEL0		:ENABLE INTERRUPTS
7772									:SET TIMER
7773	065634				TOORO1:	MOV	#2000,TIMER1		:SET TIMER
7774	065634	012737	002000	010270					:CHECK TIME
7775									:TIME EXPIRED ?
7776	065642	005737	010270		TOORO2:	TST	TIMER1		:NO. CONTINUE
7777	065646	001055				BNE	TOORO3		:TIME EXPIRED
7778									:IS KMS INITIALIZATION REQUESTED ?
7779	065650	032737	000400	010230		BIT	#INITC,FLAG		:NO
7780	065656	001404				BEQ	10\$		:SPECIFY ERROR MESSAGE
7781	065660	012737	022075	010170		MOV	#DVEM1A,TEMP2		:
7782	065666	000413				BR	30\$		:
7783									:
7784	065670	032737	001000	010230	10\$:	BIT	#SETC,FLAG		:IS START PROTOCOL REQUESTED ?
7785	065676	001404				BEQ	20\$		:NO
7786	065700	012737	022147	010170		MOV	#DVEM1B,TEMP2		:SPECIFY ERROR MESSAGE
7787	065706	000403				BR	30\$		:
7788	065710				20\$:				:
7789	065710	012737	022032	010170		MOV	#DVEM1,TEMP2		:TIMED OUT. ERROR MESSAGE...
7790	065716	013737	010170	013710	30\$:	MOV	TEMP2,ERRMSG		:SPECIFY MESSAGE FOR ERROR REPORT
7791	065724	005037	010172			CLR	TEMP3		:SPECIFY ERROR CODE
7792	065730	017737	125436	010174		MOV	@SEL0,TEMP4		:COPY BSEL0
7793	065736	017737	125434	010176		MOV	@SEL2,TEMP5		:COPY BSEL2
7794	065744	005237	010124			INC	ERRCNT		:INCREMENT ERROR COUNT
7795	065750	004737	024226			JSR	PC,LGDVE		:LOG ERROR
7796		000011			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 226
TOORO WAIT FOR AN EXPECTED CONTROL MESSAGE FROM DEVICE

7797	065754	012737	000011	013706	MOV	#ENUM,ERRNBR	;RECORD ERROR NUMBER
7798	065762	012737	000003	013704	MOV	#T.E\$FT,ERRTYP	;SPECIFY ERROR TYPE
7799	065770	012737	023460	013712	MOV	#ERR13,ERRBLK	;SPECIFY ERROR HANDLER
7800	065776				ERROR		;REPORT ERROR
7801	065776	104460					TRAP C\$ERROR
7802	066000	000715			BR	TOORO1	;REPEAT WAIT FOR ,INTERRUPT
7803							
7804							;CHECK FOR INTERRUPT
7805	066002				TOORO3: BREAK		;PERMIT CONTROL-C EXIT WHILE WAITING
7806	066002	104422					TRAP C\$BRK
7807	066004	032737	000003	010230	BIT	#ININT!OTINT,FLAG	;HAS INTERRUPT OCCURRED ?
7808	066012	001713			BEQ	TOORO2	;NO. KEEP WAITING
7809							;INTERRUPT OCCURRED
7810	066014	042737	000003	010230	BIC	#ININT!OTINT,FLAG	;CLEAR INTERRUPT INDICATOR
7811	066022	132777	000200	125346	BITB	#RDYO,@BSEL2	;IS THERE A MESSAGE FROM THE DEVICE ?
7812	066030	001704			BEQ	TOORO2	;NO. KEEP WAITING
7813							;MESSAGE FROM DEVICE
7814	066032	004737	066040		JSR	PC,OUTHDL	;PROCESS MESSAGE FROM DEVICE
7815	066036	000207			RTS	PC	;RETURN TO CALLER
7816							

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      .SBTTL                                OUTPUT INTERRUPT HANDLER
7818
7819      :++
7820      : FUNCTIONAL DESCRIPTION:
7821      :   OUTHDL - OUTPUT INTERRUPT HANDLER
7822      :   THIS ROUTINE IS CALLED WHEN AN INTERRUPT HAS OCCURRED
7823      :   WITH RDYO ASSERTED (BSEL2.BIT7 = 1).
7824      :   COPY SEL2,SEL4,SEL6
7825      :   CLEAR RDYO TO SAY MESSAGE TRANSFER COMPLETED
7826
7827
7828      : USE OF FLAGS:
7829      :   "INITC" - SET BY DVINIT IF WAITING FOR COMPLETION
7830      :               OF SYSTEM INITIALIZATION
7831      :               CLEARED BY THIS ROUTINE IF COMMAND PAIR 0,0
7832      :               IS RETURNED BY FIRMWARE
7833      :   "CTX"    - SET IF TRANSMIT COMPLETED
7834      :   "CRX"    - SET IF RECEIVE COMPLETED
7835
7836
7837      : SUBORDINATE ROUTINES USED:
7838
7839      :   "LGDVE" -LOG DEVICE ERRORS TO EVENT LOG
7840
7841      : CALLING SEQUENCE
7842      :   JSR      PC,OUTHDL
7843
7844      :--
7845
7846      066040      OUTHDL:                      ;COPY FIRMWARE MESSAGE REGISTERS
7847      066040      017737      125326      067170      MOV      @SEL0,SVSEL0          ;COPY SEL0
7848      066046      017737      125324      067172      MOV      @SEL2,SVSEL2          ;COPY SEL2
7849      066054      017737      125322      067174      MOV      @SEL4,SVSEL4          ;COPY SEL4
7850      066062      017737      125320      067176      MOV      @SEL6,SVSEL6          ;COPY SEL6
7851
7852      066070      142777      000200      125300      BICB      #RDYO,@BSEL2          ;ASSERT MESSAGE TRANSFER COMPLETED
7853
7854      066076      005737      010134          TS?      CLNSET                      ;CHECK FOR CLEANUP MODE
7855      066102      001030          BNE      OUTEND                      ;IF CLEANUP MODE ENTERED (CLNSET = -1)
7856
7857
7858      066104      013702      067172          MOV      SVSEL2,R2          ;ISOLATE COMMAND PAIR
7859      066110      010203          MOV      R2,R3          ;COPY SAVED SEL2
7860      066112      006202          ASR      R2          ;COPY AGAIN
7861      066114      006202          ASR      R2          ;SHIFT COMMAND FROM BITS 5-6
7862      066116      042702      177747          BIC      #^C<30>,R2          ; TO BITS 3-4
7863      066122      042703      177770          BIC      #^C<7>,R3          ;ISOLATE COMMAND IN R2 BITS 4:3
7864
7865      066126      050203          BIS      R2,R3          ;ISOLATE SUBCOMMAND IN R3 BITS 2:0
7866      066130      010337      067166          MOV      R3,CMDSUB          ;COMPUTE COMMAND PROCESSOR ADDRESS
7867
7868
7869      066134      022703      000037          CMP      #37,R3          ;COMBINE COMMAND,SUBCOMMAND
7870      066140      103003          BHIS     10$          ;SAVE COMMAND,SUBCOMMAND PAIR
7871
7872      066142      013703      066266          MOV      ADRINV,R3          ;CHECK COMMAND PAIR IN VAILD RANGE (37:0)
7873
7874
7875
7876
7877
7878
7879
7880
7881
7882
7883
7884
7885
7886
7887
7888
7889
7890
7891
7892
7893
7894
7895
7896
7897
7898
7899
7900
7901
7902
7903
7904
7905
7906
7907
7908
7909
7910
7911
7912
7913
7914
7915
7916
7917
7918
7919
7920
7921
7922
7923
7924
7925
7926
7927
7928
7929
7930
7931
7932
7933
7934
7935
7936
7937
7938
7939
7940
7941
7942
7943
7944
7945
7946
7947
7948
7949
7950
7951
7952
7953
7954
7955
7956
7957
7958
7959
7960
7961
7962
7963
7964
7965
7966
7967
7968
7969
7970
7971
7972
7973
7974
7975
7976
7977
7978
7979
7980
7981
7982
7983
7984
7985
7986
7987
7988
7989
7990
7991
7992
7993
7994
7995
7996
7997
7998
7999
8000

```

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

MACV11 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$: ASL      R3          ;MULTIPLY COMMAND PAIR BY 2
7876 066152 016303 066166    MOV      CMTDBL(R3),R3    ;GET OFFSET TO COMMAND ACTION ROUTINE
7877 066156 062703 066166    20$: ADD      #CMTDBL,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          CMTDBL:          ;BASE ADDRESS FOR ACCESSING COMMAND ACTION ROUTINES
7883                                     ;TABLE OF COMMAND PROCESSOR ADDRESS OFFSETS
7884 066166 000110          .WORD     COS0-CMTDBL
7885 066170 000120          .WORD     COS1-CMTDBL
7886 066172 000102          .WORD     COS2-CMTDBL
7887 066174 000102          .WORD     COS3-CMTDBL
7888 066176 000174          .WORD     COS4-CMTDBL
7889 066200 000102          .WORD     COS5-CMTDBL
7890 066202 000102          .WORD     COS6-CMTDBL
7891 066204 000102          .WORD     COS7-CMTDBL
7892 066206 000102          .WORD     C1S0-CMTDBL
7893 066210 000102          .WORD     C1S1-CMTDBL
7894 066212 000102          .WORD     C1S2-CMTDBL
7895 066214 000222          .WORD     C1S3-CMTDBL
7896 066216 000230          .WORD     C1S4-CMTDBL
7897 066220 000344          .WORD     C1S5-CMTDBL
7898 066222 000102          .WORD     C1S6-CMTDBL
7899 066224 000446          .WORD     C1S7-CMTDBL
7900 066226 000512          .WORD     C2S0-CMTDBL
7901 066230 000552          .WORD     C2S1-CMTDBL
7902 066232 000102          .WORD     C2S2-CMTDBL
7903 066234 000102          .WORD     C2S3-CMTDBL
7904 066236 000102          .WORD     C2S4-CMTDBL
7905 066240 000102          .WORD     C2S5-CMTDBL
7906 066242 000102          .WORD     C2S6-CMTDBL
7907 066244 000102          .WORD     C2S7-CMTDBL
7908 066246 000560          .WORD     C3S0-CMTDBL
7909 066250 000616          .WORD     C3S1-CMTDBL
7910 066252 000642          .WORD     C3S2-CMTDBL
7911 066254 000660          .WORD     C3S3-CMTDBL
7912 066256 000704          .WORD     C3S4-CMTDBL
7913 066260 000102          .WORD     C3S5-CMTDBL
7914 066262 000102          .WORD     C3S6-CMTDBL
7915 066264 000102          .WORD     C3S7-CMTDBL
7916 066266 000102          ADRINV: .WORD     INVCMD-CMTDBL
7917

```

CZKMSAO KMS11-BD/BE DCLT
CZKMSA.P1' 23-DEC-82 13:54

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 229
ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

```

7918      .SBTTL ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES
7919
7920      ;THE FOLLOWING RESPONSES ARE CONSIDERED TO BE ILLEGAL
7921      ;IN THIS PROGRAM
7922      COS2:      ;ACTIVITY TERMINATED ON LINE (NOT USED IN THIS PROGRAM)
7923      COS3:      ;RESERVED
7924      COS5:      ;RESERVED
7925      COS6:      ;RESERVED
7926      COS7:      ;RESERVED
7927      C1S0:      ;RESERVED
7928      C1S1:      ;RESERVED
7929      C1S2:      ;RESERVED
7930      C1S6:      ;TOO MANY RECEIVE BUFFERS FOR LINE.MAX OF 2 PER LINE
7931      C2S2:      ;RESERVED
7932      C2S3:      ;RESERVED
7933      C2S4:      ;RESERVED
7934      C2S5:      ;RESERVED
7935      C2S6:      ;RESERVED
7936      C2S7:      ;RESERVED
7937      C3S5:      ;RESERVED
7938      C3S6:      ;RESERVED
7939      C3S7:      ;DEBUG
7940      INVCMD:
7941      INC      ILLRSP      ;COMMAND NOT WITHIN VALID RANGE
7942      RTS      PC          ;COUNT ILLEGAL KMS RESPONSES
7943      PC          ;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 066306 000207          PC          ;ASSERT INITIALIZATION COMPLETED
7947          ;RETURN
7948 066306          COS1:          ;SET/RESET DDCMP MODE COMPLETED (0,1)
7949 066306 032737 001000 010230 BIT      #SETC,FLAG      ;IS SET DDCMP EXPECTED ?
7950 066314 001016          BNE      10$      ;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          ENUM = ^D<ENUM+1>      ;INCREMENT ERROR NUMBER
7958 066342          ERRSOF T ENUM,DVEM11,ERR14      ;REPORT ERROR
7959 066342 104457          TRAP      CSERSOF 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          PC          ;RETURN
7965
7966          COS4:          ;TRANSMIT COMPLETE
7967 066362          INC      CTX          ;INCREMENT TX COMPLETE COUNT
7968 066362 005237 007644          MOV      TSPTIR,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,TSPTIR      ;SAVE UPDATED POINTER
7972 066402 010237 007654          RTS      PC          ;RETURN
7973 066406 000207
7974
7975 066410          C1S3:          ;ERROR THRESHOLD REACHED
7976 066410 105237 013640          INCB     ERRTHR      ;COUNT ERROR THRESHOLD MESSAGES
7977 066414 000207          RTS      PC          ;RETURN
7978
7979

```

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

MACY11 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      #DVEM14,TEMP2          ;SPECIFY PROTOCOL START ERROR MESSAGE
7993 066450 013737 010170 013710 10$:    MOV      TEMP2,ERRMSG          ;SPECIFY MESSAGE FOR ERROR REPORT
7994 066456 012737 000006 010172          MOV      #6,TEMP3              ;SPECIFY ERROR CODE
7995 066464 013737 067174 010174          MOV      SVSEL4,TEMP4          ;COPY SAVED SEL4
7996 066472 013737 067176 010176          MOV      SVSEL6,TEMP5          ;COPY SAVED SEL6
7997 066500 004737 067146                                JSR      PC,LOGE1              ;LOG ERROR
7998 066504 012737 000003 013704          MOV      #T.ESFT,ERRTYP          ;SPECIFY ERROR TYPE
7999                                ENUM = ^D<ENUM+1>          ;INCREMENT ERROR NUMBER
8000 066512 012737 000013 013706          MOV      #ENUM,ERRNBR          ;SPECIFY ERROR NUMBER
8001 066520 012737 023700 013712          MOV      #ERR25,ERRBLK          ;SPECIFY ERROR HANDLER
8002 066526                                ERROR              ;REPORT ERROR
8003 066526 104460                                TRAP      C$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          000014          ENUM = ^D<ENUM+1>                    ;INCREMENT ERROR NUMBER
8019 066576          ERRSOFT ENUM,DVEM15,ERR24                    ;REPORT ERROR
8020 066576 1 1457
8021 066600 C 1014
8022 066602 022631
8023 066604 023644
8024
8025 066606 013702 007654          MOV      TSPTIR,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,TSPTIR             ;COPY SAVED SEL2
8029 066626 005237 013642          INC      RXMIT                ;SAVE UPDATED POINTER
8030 066632 000207          RTS      PC                          ;SPECIFY RETRANSMIT
8031
8032
8033          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 0C0337 010174          SWAB     TEMP4                ;PUT BDL NUMBER IN BYTE0
8038 066662 004737 067146          JSR      PC,LOGE1              ;LOG ERROR
8039          000015          ENUM = ^D<ENUM+1>                    ;INCREMENT ERROR NUMBER
8040 066666          ERRSOFT ENUM,DVEM17,ERR22                    ;REPORT ERROR
8041 066666 104457
8042 066670 000015
8043 066672 022705
8044 066674 023542
8045 066676 000507
8046
      GOXOUT: BR      EXOUT                      ; GOTO CLEANUP EXIT

```

TRAP CSERSOFT
.WORD 12
.WORD DVEM15
.WORD ERR24

TRAP CSERSOFT
.WORD 13
.WORD DVEM17
.WORD ERR22

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 233
ACTION ROUTINES FOR KMS RESPONSES AND MESSAGES

8047					C2S0:		;MSG RCVD (2,0)
8048	066700				10\$:	INC CRX	:INCREMENT RX COMPLETED COUNT
8049	066700	005237	007646			MOV RSPTR,R2	:POINT TO RX SAVE AREA ENTRY
8050	066704	013702	007652			MOV SVSEL6,(R2)+	:COPY SAVED SEL6 TO SAVE AREA
8051	066710	013722	067176			MOV SVSEL2,(R2)+	:COPY SAVED SEL2 TO SAVE AREA
8052	066714	013722	067172			CMP #RXSKEN,R2	:IS R2 POINTING TO THE END OF AREA ?
8053	066720	022702	010046			BNE 20\$	:NO. KEEP POINTER
8054	066724	001002				MOV #RXSTAK,R2	:YES. RESET POINTER
8055	066726	012702	007672		20\$:	MOV R2,RSPTR	:SAVE POINTER
8056	066732	010237	007652			RTS PC	:RETURN
8057	066736	000207					
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 C3S2:      ;KMC DATA MEMORY CONTENTS
8078 067036 013737 067176 067202      MOV      SVSEL4,KADR      ;SAVE DEVICE RAM ADDRESS
8079 067044 000207      MOV      SVSEL6,KDATA      ;SAVE DEVICE RAM CONTENTS (ONE BYTE)
8080      RTS      PC
8081 067046
8082 067046 012737 022766 010170 C3S3:      ;KMS TIME OUT. WAITING FOR RDYI = 0
8083 067054 004737 067126      MOV      #DVEM33,TEMP2      ;POINT TO RDYI TIMEOUT MESSAGE
8084      JSR      PC,LOGEO      ;LOG ERROR
8085 067060      ENUM = ^D<ENUM+1>      ;INCREMENT ERROR NUMBER
8086 067060 104457      ERRSOF1 ENUM,DVEM33,ERR13      ;REPORT ERROR
8087 067062 000016      TRAP      C$ERSOF1
8088 067064 022766      .WORD      14
8089 067066 023460      .WORD      DVEM33
8090 067070 000207      .WORD      ERR13
8091      RTS      PC      ;RETURN
8092 067072
8093 067072 012737 023030 010170 C3S4:      ;KMS TIME OUT. WAITING FOR RDYO = 0
8094 067100 004737 067126      MOV      #DVEM34,TEMP2      ;SPECIFY ERROR MESSAGE
8095      JSR      PC,LOGEO      ;LOG ERROR
8096 067104      ENUM = ^D<ENUM+1>      ;INCREMENT ERROR NUMBER
8097 067104 104457      ERRSOF1 ENUM,DVEM34,ERR13      ;REPORT ERROR
8098 067106 000017      TRAP      C$ERSOF1
8099 067110 023030      .WORD      15
8100 067112 023460      .WORD      DVEM34
8101 067114 000207      .WORD      ERR13
8102      RTS      PC      ;RETURN
8103
8104

```

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

MACV11 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,HLTTTB      ; HALT LINE
      JMP    GTRA5          ; GOTO DCLT PROMPT

;LOG DEVICE ERROR
LOGE0: CLR     TEMP3        ;SPECIFY ERROR CODE
      MOV    SVSEL0,TEMP4   ;COPY SAVED SEL0
      MOV    SVSEL2,TEMP5   ;COPY SAVED SEL2
LOGE1: MOV     SVSEL2+1,TEMP3+1 ;COPY SAVED LINE NUMBER
      INC    ERRCNT         ;INCREMENT ERROR COUNT
      JSR    PC,LGDVE       ;LOG DEVICE ERROR
      RTS    PC             ;RETURN

;DATA SAVED FROM FIRMWARE TO HOST MESSAGES
CMDSUB: .WORD 0             ;SAVED COMMAND, SUBCOMMAND PAIR
SVSEL0: .WORD 0             ;SAVED COPY OF SEL0
SVSEL2: .WORD 0             ;SAVED COPY OF SEL2
SVSEL4: .WORD 0             ;SAVED COPY OF SEL4
SVSEL6: .WORD 0             ;SAVED COPY OF SEL6
KADR:   .WORD 0             ;SAVED DEVICE RAM ADDRESS
KDATA:  .WORD 0             ;SAVED DEVICE RAM DATA (ONE BYTE)

```

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 067204          BGNSRV  DVINS
8138 067204          DVINS::
8139 067204 052737 000001 010230  BIS  #ININT,FLAG
8140
8141 067212          ENDSRV
8142 067212          L10023:
8143 067212 000002          RTI
8144
8145 067214          BGNSRV  DVOUTS
8146 067214          DVOUTS::
8147
8148 067214 052737 000002 010230  BIS  #OTINT,FLAG  ;OUTPUT INTERRUPT OCCURRED INDICATOR
8149 067222          ENDSRV
8150 067222          L10024:
8151 067222 000002          RTI
8152
8153
8154

```

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

MACV11 304(1052) 23-DEC-82 14:01 PAGE 237
DEVICE INTERRUPT SERVICE ROUTINES

8155
8156
8157

.EVEN

8158 067224

ENDTST

8159 067224

8160 067224 104401

8161

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.
:--

```

BGNHRD

.WORD L10025-L\$HARD/2
L\$HARD::

.SBTTL DEVICE INDEPENDENT SECTION

DISPLAY DPLX

.WORD T\$CODE
.WORD DPLX

.SBTTL DEVICE DEPENDENT SECTION

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:

.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: .ASCII	/KMC11-B CSR ADDRESS: /
067324	047111	042524	051122	VECTOR: .ASCII	/INTERRUPT VECTOR ADDRESS: /
067357	111	052116	051105	PRIOR: .ASCII	/INTERRUPT PRIORITY: /
067405	104	030515	026461	DM11BA: .ASCII	/DM11-BA CSR ADDRESS: /
067433	106	046125	020114	DPLX: .ASCII	/FULL DUPLEX ONLY/
067454	000			.BYTE	0

.LIST BEX
.EVEN

8219 067456
8220
8221

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

MACV11 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

LSLAST::
ENDMOD

.END

.EVEN
.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 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#					
ACTRDQ	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					
ADDCl	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#							
ALLTR	061110	6551	6574	6598	6674#				
ASSEMB=	000010	1996							
ATVMD=	030027	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						

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 245
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 246
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 247
CROSS REFERENCE TABLE -- USER SYMBOLS

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 248
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 249
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 250
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

[illegible]

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 252
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 253
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 254
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 255
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 256
CRCSS 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#				
LSOLY	002116 G	2096#					
LSUTP	002040 G	2050#					
LSDTYP	002034 G	2046#					
LSDU	054150 G	2077	5666#				
LSDUT	002072 G	2076#					
LSDVTV	013714 G	2067	3227#				
LSFF	002052 G	2061#					
LSENVI	002044 G	2054#					
LSERRT	013704 G	2085	3208#				
LSETP	002102 G	2084#					
LSXP1	002046 G	2056#					
LSXP4	002064 G	2070#					

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 257
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 258
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 259
CROSS REFERENCE TABLE -- USER SYMBOLS

NOCCLK	016715	3253#	5731
NOD0	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#	
NOD14	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#

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 261
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 262
CROSS REFERENCE TABLE -- USER SYMBOLS

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 263
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CZKMSAO KMS11-BD/BE DCLT MACY11 30A(1052) 23-DEC-82 14:01 PAGE 264
 CZKMSA.P11 23-DEC-82 13:54 CRSSS REFERENCE TABLE -- USER SYMBOLS

PNT = 001000 G	2220#																
PNTTRB 026616	3893	4063	4078	4101#													
PR1 = 002000 G	2221#																
PRIOR 067357	8204	8218#															
PR100 = 000000 G	2204#	5583															
PR101 = 000040 G	2208#																
PR102 = 000100 G	2207#																
PR103 = 000140 G	2206#																
PR104 = 000200 G	2205#																
PR105 = 000240 G	2204#																
PR106 = 000300 G	2203#																
PR107 = 000340 G	2053	2202#	5634														
PRNT = 000055	2373#	2960	6045														
PROTO = 000041	2361#																
PROTOB = 000040	2276#	6278															
PRTE1B 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										
PSFREE 003366	2680#	3706*	5094	5819*													
PSIRV 031506	3709	5092#	5822														
PSIRS 031516	5095#	5104	5122	5127													
QCOPY = 000013	2339#	3082	6142														
QRX = 000004	2301#	4517	6547	6596	6625	6675	6696	6735	6744	6765	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															
RDMPE = 000010	2392#	4194															
RDMPQ = 000007	2391#	4196															
RDMP5 = 000006	2390#	3736	4159	4192													
RDVI = 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

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 265
CRCSS REFERENCE TABLE -- USER SYMBOLS

RDYO = 000200	7522	7728	7733						
REC = 000000	2430#	7376	7561	7724	7811	7852			
RECMOD= 000031	2242#								
REPLOG 025374	2353#	2987							
REPORT 024644	3839#	4148							
RESFLG 010214	3678#	5336	6046						
RESTR 033110	2787#	5378*	5402*	5737					
RMLPEN 003304	5388	5416	5429	5448	5457#				
RMLPTB 003276	2647#	4140							
RMLPO 015551	2644#	4132							
RMLP1 015607	3253#	3685							
RMLP2 015631	2644	3253#							
RMLP3 015646	2645	3253#							
RI = 000000	2646	3253#							
RMDLOP= 000047	2406#	2863							
RNOTNF= 000005	2367#	3099							
RPASS 010226	2389#	4185	4193	4195					
RPEXT = 000002	2803#	3598	5768*	5788	5796	6051*	6115	6287*	6921
RPHLP = 000001	2386#	3742	4145	4177					6923*
RPLOG = 000003	2385#	4142	4173	4175					
RPSOL = 000004	2387#	4149	4181						
RPT 025442	2388#	3730	4153	4179					
RPTAA 025512	3854#	3902	3924	4035	4057	4073	4089		
RPTABO 026366	3869#								
RPTDCK 026526	2918	4039#							
RPTDDE 026446	2913	4075#							
RPTDER 025666	2916	4060#							
RPTDLE 026526	2911	3904#							
RPTDSP 011440	2915	4074#							
RPTDVI 026316	2907#	3889							
RPTTEOP 026366	2912	4025#							
RPTSOL 025112	2917	4040#							
RPTTXQ 025614	3733	3767#							
RPTO 025506	2907	2908	2909	2910	3890#				
RPT1 025474	3845	3861	3865	3868#					
RQI = 000200	3858	3864#							
RQIFLG 010136	2424#	7697	7702	7735					
RSPTRE 007652	2754#								
KSPTPS 007650	2718#	6490*	8050	8056*					
RTS = 000000	2717#	6489*	7618	7628*					
RUN = 000004	2405#	2862							
RXBOL 013434	2332#	2964	5844	6049					
RXBUF 005406	3167#	7417							
RXC = 000006	2694#	6514							
RXMIT 013642	2287#	3587	3629						
RXMTOT 010112	3189#	6488*	7508	7524*	7530	7566	8029*		
RXONLY 060736	2741#	4585	6478*	6481*	6622*	6833			
RXON3 060754	2820	6547#							
RXPTR 010050	6550#								
RXQ = 000004	2724#	4755	5747*	6477*					
RXQUAL 030112	2286#	3583							
RXQUEX 030156	4512#	6549	6597	6627					
RXQU1 030120	4515	4520#							
RXSKEN 010046	4513#								
RXSTAK 007672	2722#	7625	8053						
	2721#	6489	6490	7627	8055				

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 266
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CZKMSAO KMS11-BD/BE DCLT 23-DEC-82 13:54 MACV11 30A(1052) 23-DEC-82 14:01 PAGE 267
 CZKMSA.P11 23-DEC-82 13:54 CRCSS REFERENCE TABLE -- USER SYMBOLS

STADD 010126	2748#	4157*	4233	6120*									
START 032706	5383	5402#											
STARTC 013636	3187#	6486*	7953*										
STAT = 002000	2307#	7314	7322	7324	7326	8068	8074						
STATB = 000001	2271#	3631	5043	6269									
STATUS= 000016	2342#	3034											
STATYP 013424	3148#												
STEP = 000400	2423#												
STXC 021664	3253#	3577											
STXD 021653	3253#	3572											
SVCGBL= 000000	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											
SVCINS= 000001	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

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											
T&URAD	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	6354#	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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 269
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

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#	4454	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									
TSARGC= 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
TS CODE= 003031	3699#	5442#	5812#	6999#	8184#	8193#	8198#	8203#	8209#				
TSERRN= 000017	1996#	6855#	6874#	6888#	7153#	7549#	7595#	7641#	7712#	7960#	8021#	8042#	8087#
	8098#												
TS EXCP= 000000	3699#	3704	5442#	5447	5812#	5817	6999#	7004	8193#	8197	8198#	8202	8203#
	8208	8209#	8213										
TS FLAG= 000040	3430#	3432	5586#	5649#	5670#	5672	5692#	5694	5799#	5855#			
TS GMAN= 000000	1996#	3696#	3699	3705#	5439#	5448#	5809#	5812	5818#	6996#	6999	7005#	
TS HILI= 177776	3699#	3703	5442#	5446	5812#	5816	6999#	7003	8193#	8196	8198#	8201	8203#
	8207	8209#	8212										
TS LAST= 000001	1996#	8251#											
TS LOLI= 160000	3699#	3702	5442#	5445	5812#	5815	6999#	7002	8193#	8195	8198#	8200	8203#
	8206	8209#	8211										
TS LSYM= 010000	1996#	2146	3287	3300	3314	3331	3347	3365	3384	3400	3427	3538	5344
	5594	5610	5657	5678	5700	8143	8151	8160	8217				
TS LINO= 000001	8254#												
TS 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#					
TS NSO = 000000	1998#	8255											
TS 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												
TS NS2 = 000010	8138#	8142	8146#	8150									
TS PTNU= 000000	1996#												
TS SAVL= 177777	1996#												
TS SEGL= 177777	1996#												
TS SUBN= 000000	1996#	5715#											
TS TAGL= 177777	1996#												
TS 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#			
TS 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

TSTEST= 000001	1996#	5715#	8254										
TSTSTM= 177777	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					
TSTSTS= 000001	1996#	5716#											
TSSAU = 010021	5688#	5692	5699										
TSSAUT= 010016	5605#	5609											
TSSCLE= 010017	5619#	5649	5656										
TSSDU = 010020	5666#	5670	5677										
TSSHAR= 010025	8177#	8216											
TSSHW = 010000	2119#	2145											
TSSINI= 010015	5369#	5586	5593										
TSSMSG= 010011	3273#	3286	3290#	3299	3303#	3313	3320#	3330	3336#	3346	3352#	3364	3369#
	3383	3388#	3399	3403#	3426	3430							
TSSPRO= 010014	5353#												
TSSRPT= 010013	5333#	5343											
TSSSRV= 010024	3512#	3537	8138#	8142	8146#	8150							
TSSTES= 010022	5716#	5799	5855	8159									
T.EDF = 000001	2448#												
T.EHRD= 000002	2449#												
T.ESF = 000000	2447#												
T.ESFT= 000003	2450#	7798	7998										
T1 = 054164 G	2109	5715#											
UAM = 000200 G	2218#												
ULRCLS 030552	4793#	6739											
ULRPLS 030454	4516	4732#	4896	6678	6717								
ULTCLS 030652	4869#	4946	6756										
ULTPLS 030612	4830#	4949	6705										
UPTABL 061252	6703#												
UPTA1 061336	6710	6717#											
UPTA4 061302	6704	6709#											
UPTEX 061436	6715	6731#											
VALTRB 003404	2689#	6380*	6409*										
VECTOR 057324	8199	8218#											
VSTAT 011514	2933#	3773	7158*	7328*									
WRFLG 003402	2687#												
X\$ = 000234	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

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 272
CROSS REFERENCE TABLE -- USER SYMBOLS

X\$ALWA= 000000	4204#													
X\$FALS= 000040	1996#													
X\$OFFS= 000400	1996#													
X\$TRUE= 000020	1996#													
\$PATCH 067456	8245#													
. = 067542	1996#	2542#	2547#	2583	2590	2629#	2662#	2688#	2692#	2693#	2694#	2697#	2698#	
	2699#	2700#	2704#	2705#	2706#	2707#	2708#	2709#	2720#	2721#	2848#	2849#	2958#	
	2962#	2966#	2970#	2975#	2981#	2983#	2989#	2991#	3002#	3004#	3028#	3030#	3034#	
	3036#	3039#	3045#	3051#	3053#	3055#	3057#	3059#	3065#	3080#	3082#	3097#	3109#	
	3113#	3167#	3168#	3169#	3170#	3171#	3172#	3173#	3174#	3175#	3176#	3177#	3178#	
	3179#	3180#	3181#	3182#	3246#	3253#	3431	4175#	4177#	4179#	4185#	5587	5589	
	5650	5671	5693	5800	5856	8219#	8246#							

MACY11 30A(1052) 23-DEC-82 14:01 PAGE 274
CROSS REFERENCE TABLE -- MACRO NAMES

BCOMPL	1#	1996#	5382	5387	5398	5794											
BERROR	1#	1996#															
BGNAU	1#	1996#	5687														
BGNAUT	1#	1996#	5604														
BGNCLN	1#	1996#	5618														
BGNDU	1#	1996#	5665														
BGNHRD	1#	1996#	8176														
BGNHW	1#	1996#	2118														
BGNINI	1#	1996#	5368														
BGNMOD	1#	1996#	1997														
BGNMSG	1#	1996#	3272	3289	3302	3319	3335	3351	3368	3387	3402						
BGNPRO	1#	1996#	5352														
BGNPTA	1#	1996#															
BGNRPT	1#	1996#	5332														
BGNSEG	1#	1996#															
BGNSET	1#	1996#															
BGNSFT	1#	1996#															
BGNSRV	1#	1996#	3511	8137	8145												
BGNSUB	1#	1996#															
BGNSW	1#	1996#															
BGNTST	1#	1996#	5714														
BNCOMP	1#	1996#	5392	5412	5422	5433	5477										
BNERRO	1#	1996#															
BREAK	1#	1996#	5544	7142	7555	7717	7805										
BRESET	1#	1996#	5403	5631													
CKLOOP	1#	1996#															
CLI	2000#	2954	2955	2956	2957	2958	2959	2960	2961	2962	2963	2964	2965	2966	2967		
	2968	2969	2970	2971	2972	2973	2974	2975	2976	2979	2980	2981	2982	2983	2984		
	2985	2986	2987	2988	2989	2990	2991	2992	2993	2994	2995	2996	3000	3001	3002		
	3003	3004	3009	3010	3011	3012	3013	3016	3017	3018	3019	3020	3021	3022	3023		
	3025	3026	3027	3028	3029	3030	3033	3034	3035	3036	3038	3039	3041	3042	3044		
	3045	3047	3050	3051	3052	3053	3054	3055	3056	3057	3058	3059	3060	3061	3062		
	3063	3064	3065	3068	3069	3070	3071	3072	3073	3074	3077	3078	3079	3080	3081		
	3082	3085	3086	3087	3089	3092	3093	3094	3095	3096	3097	3098	3099	3102	3103		
	3104	3107	3108	3109	3110	3111	3112	3113	3115	3116	3117	3118	3119	3122	4171		
	4172	4173	4174	4175	4176	4177	4178	4179	4180	4181	4184	4185	4189	4190	4191		
	4192	4193	4194	4195	4196	4200	4203										

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#													
IOSETU	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#						
MSERRI	1#	1996#	6854#	6873#	6887#	7152#	7548#	7594#	7640#	7711#	7959#	8020#	8041#	8086#	8097#
MSERCA	1#	1996#													
MSERCS	1#	1996#													
MSXCP	1#	1996#	3699#	5442#	5812#	6999#	8195#	8198#	8203#	8209#					
MSXIT	1#	1996#	3430#	5586#	5587#	5649#	5650#	5670#	5692#	5799#	5800	5855#	5856		
MSXSE	1#	1996#	3430#	5586#	5649#	5670#	5692#	5799#	5855#						
MSXTJ	1#	1996#	3430#	3431	5586#	5649#	5670#	5671	5692#	5693	5799#	5855#			
MSGEN	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#												
MSGENB	1#	1996#	3696#	3697	5439#	5440	5809#	5810	6996#	6997					
MSGETS	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#					
MSGETT	1#	1996#	3430#	5586#	5649#	5670#	5692#	5799#	5855#						
MSGNGB	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			
MSGNIN	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#												
MSHAPT	1#	1996#	2015#												
MSHNAP	1#	1996#	2015#	2054#											
MSINCR	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#													
MSIOSE	1#	1996#													
MSLDRO	1#	1996#	5380#	5385#	5390#	5396#	5409#	5419#	5474#	5583#	5634#				
MSMASK	1#	1996#													
MSMCHI	1#	1996#													
MSMCLO	1#	1996#													
MSMSK1	1#	1996#													
MSPOP	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#					
MSPRIN	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#													
MSPUSH	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

MSPUT	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#
MSPUT1	6442#	6467#	6960#	7056#	7079#										
	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							
MSRADI	1#	1996#	3699#	5442#	5812#	6999#	8193#	8198#	8203#	8209#					
MSRBRO	1#	1996#													
MSRNRO	1#	1996#	5409#	5411	5419#	5421	5474#	5476							
MSSETS	1#	1996#	1998#	2119#	3273#	3290#	3303#	3320#	3336#	3352#	3369#	3388#	3403#	3512#	5333#
	5353#	5369#	5605#	5619#	5666#	5688#	5716#	8138#	8146#	8177#					
MSSTAR	1#	1996#													
MS SVC	1#	1996#	3275#	3283	3286#	3287	3292#	3296	3299#	3300	3305#	3310	3313#	3314	3322#
	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										
MS TL AB	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#

MACV11 30A(1052) 23-DEC-82 14:01 PAGE 279
CROSS REFERENCE TABLE -- MACRO NAMES

ERRORS DETECTED: 0

CZKMSAO KMS11-BD/BE DCLT MACV11 30A(1052) 23-DEC-82 14:01 PAGE 280
CZKMSA.P11 23-DEC-82 13:54 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)