

DUP-11

DUP-11 DCLT
CZDCLB0

AH-S975B-MC
FICHE 1 OF 2

OCT 1983
COPYRIGHT © 82-83
MADE IN USA



Table with 16 columns and 16 rows. Each cell contains a small, illegible document snippet, likely a microfiche frame. The snippets appear to be technical or administrative documents, possibly related to the 'DUP-11' and 'AH-S975B-MC' identifiers. The text within the snippets is too small to be transcribed accurately.

DUP-11

DUP-11 DCLT
CZDCLB0

AH-S975B-MC
FICHE 2 OF 2

OCT 1983
COPYRIGHT © 82-83
MADE IN USA



CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 2

1

.TITLE CZDCLB DUP-11 DATA COMM. LINK TEST

.REM 8

IDENTIFICATION

PRODUCT CODE: AC-S974B-MC
PRODUCT NAME: CZDCLB0 DUP-11 DCLT
PRODUCT DATE: JUNE 1983
MAINTAINER: MERRIMACK DIAGNOSTIC ENGINEERING
AUTHOR: ERNIE COOPER

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) 1982,1983 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL
DEC

PDP
DECUS

UNIBUS
DECTAPE

MASSBUS

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 3
CZDCLB.P11 19-JUL-83 17:12

REVISION HISTORY:

REV ---	DATE ----	AUTHOR -----	REASON -----
A	24-MAR-82	ERNIE COOPER	ORIGINAL ISSUE, DCLT FOR THE DUP-11
**JPB B	29-JUN-83	JOHN BEIKE	DTR WAS NOT BEING SET WHILE IN MODEM LOCAL LOOP-BACK WHICH CAUSED CTS NEVER TO COME TRUE. FIX: SET DTR WITH RTS.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 4

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 STATISTICAL COMMANDS
 - 2.5.3 RUN COMMANDS
 - 2.5.4 DEFAULTS
 - 2.5.5 PRINT COMMANDS
 - 2.5.6 MISC COMMANDS
 - 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 ERRORS
 - 3.2.3 DEVICE ERRORS
- 4.0 PERFORMANCE AND PROGRESS REPORTS
 - 4.1 PRINTING EVENT LOG
 - 4.2 OPERATOR STATUS MESSAGES
 - 4.3 PRINTING DDCMP STATISTICAL AND ERROR LOG
- 5.0 DEVICE INFORMATION TABLES

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 5
CZDCLB.P11 19-JUL-83 17:12

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 MODE
- 6.1.6 TALK MODE
- 6.1.7 LISTEN MODE
- 6.1.8 MAINTENANCE MODE

6.2 MESSAGE DESCRIPTIONS

7.0 OTHER INFORMATION

- 7.1 INTERFACING TO AN "ITEP" NODE
- 7.2 TROUBLESHOOTING HINTS

- 7.2.1 INTERNAL LOOP AT EACH NODE
- 7.2.2 TRANSMIT ON ONE NODE-RECEIVE ON THE OTHER
- 7.2.3 ONE NODE ACTIVE-THE OTHER NODE PASSIVE
- 7.2.4 BOTH NODES ACTIVE
- 7.2.5 TALK AND LISTEN NODES FOR COMMUNICATIONS

7.3 EXAMPLE OF COMMANDS

- 7.3.1 MESSAGES COMMANDS
- 7.3.2 STATISTICAL COMMANDS
- 7.3.3 RUN COMMANDS
- 7.3.4 PRINT COMMANDS
- 7.3.5 EXIT COMMAND

7.4 THINGS TO WATCH OUT FOR

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 6
CZDCLB.P11 19-JUL-83 17:12

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 DUP-11 COMMUNICATION LINKS. THIS PROGRAM ALLOWS THE DUP-11 TO COMMUNICATE WITH OTHER SYNCHRONOUS (INCLUDING DDCMP) DEVICES ON POINT TO POINT OR MULTIDROP NETWORKS. THIS DCLT PROGRAM WILL PROVIDE THE COVERAGE NECESSARY TO DETECT FAILURES TO 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, APT, 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 DUP DCLT PROGRAM, THE FOLLOWING MINIMUM HARDWARE IS REQUIRED:

- A PDP-11 CPU
- MINIMUM OF 24K WORDS OF MEMORY
- A WORKING CLOCK
- A CONSOLE TERMINAL
- ANY XXDP+ SUPPORTED LOAD MEDIA
- DUP11-DA: M7867 MODULE
 - BC05C-25 CABLE
 - BC02-1D CABLE
 - H325 TEST CONNECTOR

1.3 RELATED DOCUMENTS AND STANDARDS

- XXDP+ USER'S MANUAL (CHQUS?.SEQ WHERE ? IS THE REV. LEVEL OF THE MANUAL - 'C' IS THE CURRENT REV.).

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 7
CZDCLB.P11 19-JUL-83 17:12

1.4 DIAGNOSTIC HIERARCY PREREQUISITES

THE GOAL OF THE DATA COMM. LINK TEST PROGRAM IS TO TEST THE COMMUNICATION LINK AND THEREFORE ASSUMES THAT THE CPU'S, CLOCKS, DUP-11 AND THE DEVICES AT THE OTHER END OF THE LINK HAVE ALREADY BEEN TESTED.

IF A WORKING CLOCK IS NOT FOUND, THE PROGRAM WILL CONTINUE 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.

IT IS NOT THE INTENTION OF A DATA COMM. LINK TEST PROGRAM TO TEST THE DUP-11'S, BUT TO TEST THE COMMUNICATION LINK TO WHICH THEY ARE CONNECTED.

SOME OF THE DIAGNOSTICS THAT COULD BE RUN IF THE DUP-11 LOOKS BAD:

DZDPEXX CONFIDENCE TEST
DZDPBXX BASIC TRANSMITTER TESTS
DZDPCXX RECEIVER, MODEM CONTROL AND INTERRUPT TEST
DZDPDXX DATA AND FUNCTION TESTS
DXDPBXX DECX11 MODULE

XX= LATEST REVISION

1.5 ASSUMPTIONS - RESTRICTIONS

IT IS ASSUMED THAT THE COMMUNICATIONS DEVICE HAS BEEN TESTED USING THE PREREQUISTE 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.

THIS DIAGNOSTIC DOES NOT RUN THE DUP-11 IN BIT STUFF MODE. IT IS ASSUMED THAT IF THE LINK WORKS IN CHAR MODE THE LINK WILL WORK IN BIT STUFF MODE.

THE DUP-11 IS NOT A DMA DEVICE AND THUS MUST RELY ON THE SOFTWARE FOR SERVICE.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 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 (CHQUS).

2.1 COMMANDS

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

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

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

2.2 SWITCHES

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

SWITCH -----	EFFECT -----
/TESTS:LIST	EXECUTE ONLY THOSE TESTS SPECIFIED IN THE LIST. LIST IS A STRING OF TEST NUMBERS, FOR EXAMPLE - /TESTS:1:5:7-10. THIS LIST WILL CAUSE TESTS 1,5,7,8,9,10 TO BE RUN. ALL OTHER TESTS WILL NOT BE RUN.
/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. LIST EXAMPLE - /UNITS:0:5:10-12

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 9

USE UNITS 0,5,10,11,12 (UNIT NUMBERS = 0-63)

EXAMPLE OF SWITCH USAGE:

START/TESTS:1-5/PASS:1000/EOP:100

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

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

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

2.3 FLAGS

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

FLAG	EFFECT
----	-----
HOE	HALT ON ERROR - CONTROL IS RETURNED TO RUNTIME SERVICES COMMAND MODE
LOE	LOOP ON ERROR
IER*	INHIBIT ALL ERROR REPORTS
IBE*	INHIBIT ALL ERROR REPORTS EXCEPT FIRST LEVEL (FIRST LEVEL CONTAINS ERROR TYPE, NUMBER, PC, TEST AND UNIT)
IXE*	INHIBIT EXTENDED ERROR REPORTS (THOSE CALLED BY PRINTX MACRO'S)
PRI	DIRECT MESSAGES TO LINE PRINTER
PNT	PRINT TEST NUMBER AS TEST EXECUTES
BOE	'BELL' ON ERROR
UAM	UNATTENDED MODE (NO MANUAL INTERVENTION)

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 10

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

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 DUP-11 DATA COMM. LINK TEST PROGRAM WILL NOT USE MORE THAN ONE UNIT. FOR THE DUP-11, THE HARDWARE INFORMATION REQUESTED WILL BE:

UNITS (D) ? 1<CR>

UNIT 0
FULL DUPLEX OPERATION : (L) Y ?
DEVICE CSR ADDRESS : (D) 160170 ?
INTERRUPT VECTOR ADDRESS: (D) 300 ?
REMOTE NODE 'ITEP' : (L) N ?
IS THIS A MULTIPOINT NETWORK: (L) N ?

THE FULL DUPLEX QUESTION SHOULD BE ANSWERED 'Y' WHEN USING FULL DUPLEX MODEMS, OR NULL MODEM, OR MODEM ELIMINATORS. ANSWER 'N' FOR HALF DUPLEX MODEMS.

REMOTE NODE ITEP SHOULD BE ANSWERED 'Y' IF OTHER NODE IS RUNNING SOFTWARE THAT IS USING 'ITEP' FORMATS (I.E. PDP-11 RUNNING ITEP).

IF OTHER NODE IS USING ITEP, THE ABOVE 'MULTIPOINT NETWORK' QUESTION WILL NOT APPEAR.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 11

IF TO THE 'MULTIPOINT NETWORK' QUESTION, YOU RESPOND WITH 'Y'
THEN

ADDRESS THIS STATION: (D) 1 ?

WILL BE DISPLAYED. INPUT THE DECIMAL TRIBUTARY ADDRESS (1-255)
OF THIS DUP-11.

2.5 DATA COMM. LINK TEST COMMANDS

THE 'DCLT>' COMMAND LEVEL FOLLOWS THE ANSWERING OF THE HARDWARE P-TABLE
QUESTIONS. THESE COMMANDS CAN BE TYPED WHEN THE 'DCLT> (A) ?' PROMPT
IS PRINTED.

MESSAGE COMMANDS AVAILABLE:

YOU ONLY HAVE TO TYPE ENOUGH CHARACTERS TO UNIQUELY SPECIFY A COMMAND.

THE COMMAND LINE IS INTERPRETED FROM LEFT TO RIGHT. THEREFORE,
IF A QUALIFIER ON THE COMMAND LINE IS RELATED OR EFFECTS A QUALIFIER
TO THE LEFT ON THE COMMAND LINE, 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 BE INDICATED IN THE
PARAMETERS WORD.)

REFER TO SECTION 6.0 FOR A DESCRIPTION OF THE DIFFERENT MODES OF
OPERATION AND THE TYPES OF MESSAGES AVAILABLE.

2.5.1 MESSAGE COMMANDS

COMMAND

CLEAR EXPECTLIST

CLEAR TRANSMITLIST

DESCRIPTION

ZEROES THE EXPECTLIST (00'S)
AND THEN PUTS DEFAULT ITP MSG
IN SO NOT REALLY EMPTY

FILLS TRANSMITLIST (000'S)
AND THEN PUTS DEFAULT ITP MSG
IN SO NOT REALLY EMPTY

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 12

SET EXPECTMSG=TYPE/QUAL

DEFINE A MESSAGE TO BE PUT ON
THE EXPECTED LIST

WHERE: 'TYPE' IS:

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

WHERE THE OPTIONAL 'QUAL' IS:

/SIZE=NNN

MAKE THE MESSAGE 'NNN' BYTES
LONG. (DEFAULT VALUE IS
SIZE OF MESSAGE SPEC'D BY
OPERATOR OR DEFAULTS.)

/COPY=NN

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

NOTE: SET'S ADD MESSAGES TO THE LIST IN THE ORDER THEY'RE
DEFINED. 'NNN' IS A DECIMAL NUMBER. THE FIRST SET
OVERWRITES THE DEFAULT ITEP MESSAGE PLACED THERE BY
INITIALIZATION OR A 'CLEAR' COMMAND.

SEE SECTION 6.2 FOR A DESCRIPTION OF THE PRE-DEFINED
MESSAGES THAT ARE AVAILABLE. (ZEROS,ONES ...)

SET TRANSMITMSG=TYPE/QUAL

DEFINE A MESSAGE TO BE PUT ON
THE TRANSMIT LIST
(SEE DESCRIPT FOR SET EXP)

SET EXPECT=TRANSMIT

MAKES A COPY OF THE TRANSMIT
LIST IN THE EXPECT LIST.

SHOW EXPECTLIST

LISTS THE MESSAGE SIZE AND TYPE
FOR THE MESSAGES IN THE
EXPECT LIST

SHOW TRANSMITLIST

LISTS THE MESSAGE SIZE AND TYPE
FOR THE MESSAGES IN THE
TRANSMIT LIST

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 13

2.5.2 STATISTICAL COMMANDS

COMMAND

DESCRIPTION

PRINT

TAKES THE OPERATOR TO THE
REPORT LEVEL 'RPT>'. FROM
HERE YOU CAN EXAMINE THE
EVENT LOG OR IF '/PROTOCOL'
IS SELECTED, THE DDCMP
STATISTICAL AND ERROR COUNTERS.

DUMP SSSSSS-EEEEEE/B

PRINTS THE CONTENTS OF THE
MEMORY LOCATIONS BETWEEN
OCTAL ADDRESSES 'SSSSSS' AND
'EEEEEE' WHERE 'SSSSSS' IS
THE START ADDRESS AND
'-EEEEEE' IS THE END ADDRESS.

WHERE '/B' IS OPTIONAL:
DEFAULT IS PRINT WORDS
'/B' CAUSES PRINT BYTES

IF '-EEEEEE' IS NOT SPECIFIED
THEN THE CONTENTS OF 'SSSSSS'
IS PRINTED IN WORD FORMAT.
IS PRINTED IN WORD FORMAT.

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

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 14

2.5.3 RUN COMMAND

COMMAND

DESCRIPTION

RUN MODE=MTYPE/QUAL

STARTS DCLT EXECUTING IN THE
MODE SPECIFIED

NOTE: MODE=ACTIVE IS NOT DEFAULT, A MODE=MTYPE MUST BE TYPED
----- EACH TIME A RUN IS TYPED

WHERE THE '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)

=DOWNLINELOAD (DOWN-LINE-LOADING IS NOT SUPPORTED
FOR DUP-11 TO DUP-11 LINKS).

(FORCING NO LOOPING MEANS IT MUST BE
SPECIFIED AS A QUALIFIER ANY TIME ITS
DESIRED, THERE IS NO DEFAULT)

AND OPTIONAL 'QUAL' IS ANY COMBINATION OF THE FOLLOWING:

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

NOTE: IF BOTH MODES 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.

/STATUS/NOSTATUS ENABLES/DISABLES PRINTING OF PROGRAM
STATUS MESSAGES TO THE OPERATOR

/ECHO/NOECHO ENABLES/DISABLES THE RETRANSMISSION OF
THE DATA RECEIVED IN PASSIVE MODE.
(IGNORED IN MODES OTHER THAN PASSIVE)

/MODEM/NO MODEM/ ENABLES/DISABLES THE REPORTING OF MODEM STATUS
INTERRUPT CHANGES.

/LOOP=LTYPE SPECIFIES WHICH, IF ANY, TYPE OF
MAINTENANCE LOOPBACK IS BEING USED.
(IGNORED IN MODES OTHER THAN ACTIVE)
MUST BE SPECIFIED EACH TIME ELSE NO

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 15

LOOP IS USED.

'LTYPE' IS:

=INTERNALTTL LOOPS DATA INTERNAL TO USYNRT
=CABLE USE THIS FOR TESTING WITH H325
 TURNAROUND CONNECTOR ON END OF CABLE.

NOTE: THIS SKIPS OVER THE CHECK
FOR MODEM READY WHEN DTR IS SET.

=LOCALMODEM NOT SUPPORTED BY DUP-11
=REMOTEMODEM

/PASS=NN SPECIFIES NUMBER OF ITERATIONS TO MAKE BEFORE
 END-OF-PASS. DEFAULT VALUE OF 1
 WILL BE USED ON ANY RUN THAT A /PASS=N
 IS NOT ADDED TO THE 'RUN ...' COMMAND.
 IF A '-1' IS TYPED, THEN THE PROGRAM
 RUN UNTIL A ^C IS TYPED.

/PROTOCOL ENABLES SUBSET OF DDCMP PROTOCOL- THE DUP-11
 CAN NOW COMMUNICATE WITH OTHER 'INTELLIGENT'
 SYNCHRONOUS DEVICES THAT SUPPORT DDCMP IN
 THEIR MICROCODE. (DMR,DMC,DMV OR DMP).

THIS SWITCH IS NOT SUPPORTED BY ALL DCLT'S.

/NOPROTOCOL DISABLES PROTOCOL- THE DUP-11 CAN COMMUNICATE
 ONLY WITH ANOTHER DUP-11 OR DPV-11 RUNNING
 DCLT OR ITEP.

NOTE: SEE SECTION 6.1 FOR A DESCRIPTION
----- OF THE 'RUN MODES' AND 'LOOP MODES'

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 16

2.5.4 DEFAULTS

IF NO 'SET'S' THEN THE DEFAULT IS SAME AS IF TYPED:

SET TRANSMITMSG=ITEP/SIZE=58/COPY=0
SET EXPECTMSG=ITEP/SIZE=58/COPY=0

THE DEFAULT COPY AND SIZE FOR EACH OF THE MESSAGE TYPES:

ONES - /SIZE=64/COPY=0
ZEROES - /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

FOR THE RUN COMMAND THE DEFAULTS ARE:

RUN MODE=ACTIVE/NOSTATUS/CHECK/NOECHO/NOMODEM/PASS=1/NOPROTOCOL

NOTE: MODE=ACTIVE IS NOT DEFAULT, A MODE=MTYPE MUST BE TYPED
----- EACH TIME A RUN IS TYPED

IF THE DCLT PROGRAM IS RUN IN UNATTENDED MODE (UAM FLAG=1 OR CHAINED),
THE DEFAULTS ARE AS IF THESE SETUP AND RUN COMMANDS WERE TYPED:

SET TRANS=ITEP
SET EXPECT=ITEP
RUN MODE=ACTIVE/LOOP=INTERNAL/NOSTAT/NOECHO/NOMODEM/CHECK
/PASS=1/NOPROTOCOL

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

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 17
 CZDCLB.P11 19-JUL-83 17:12

2.5.5 PRINT COMMAND

THE PRINT COMMAND TAKES YOU TO THE REPORT LEVEL 'RPT>'.
 THE COMMANDS AVAILABLE IN RPT> ARE ...

COMMAND	DESCRIPTION
-----	-----
HELP OR ?	PRINTS HELP INFORMATION FOR RPT>
EXIT	RETURNS YOU TO THE LEVEL THAT YOU ENTERED FROM. (DCLT> OR DR>)
LOG	PRINTS THE DCLT EVENT LOG
COUNTERS/FULL	PRINTS THE ENTIRE DDCMP STATISTICAL AND ERROR LOG. SEE SECTION 4.3
COUNTERS/ERRORS	PRINTS ONLY THE DDCMP ERROR LOCATIONS OF THE LOG.
COUNTERS/OFFSET=NN	PRINTS A SINGLE LOCATION OF THE LOG AS SPECIFIED BY THE OCTAL WORD OFFSET VALUE(NN).
	NOTE:: THE DDCMP COUNTERS WILL BE VALID ONLY WITH PROTOCOL ENABLED(/PROTOCOL).

2.5.6 MISC COMMANDS

COMMANDS	DESCRIPTION
-----	-----
EXIT	FROM THE DCLT> LEVEL RETURNS YOU TO DR>
HELP OR ?	PRINTS HELP INFORMATION

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 NAME'', WHERE NAME IS THE NAME OF THE BIN OR BIC
FILE FOR THIS PROGRAM
4. TYPE "START"

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 18
CZDCLB.P11 19-JUL-83 17:12

5. ANSWER THE "CHANGE HW" QUESTION WITH 'Y'
6. ANSWER ALL THE HARDWARE QUESTIONS. THE NUMBER OF UNITS THAT 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. AFTER THE 'DCLT> (A) ?' PROMPT, TYPE
'RUN MODE=ACTIVE<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.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 19
 CZDCLB.P11 19-JUL-83 17:12

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

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 20

OCTAL ADDRESSES ARE EXPECTED.

- ? '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 THE TRANSMIT LIST AND EXPECT LIST MUST BE EQUAL. IF THEY ARE NOT THIS ERROR WILL BE DISPLAYED. USE 'SE E=T' COMMAND.

3.2.2 DCLT OR DEVICE 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 THE CLOCK FOUND ON THE SYSTEM 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.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 21

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.

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 MESSAGE WAS TRUNCATED TO COMPLETELY
FILL THE BUFFER. NO MORE MESSAGES CAN
BE ADDED TO THAT BUFFER.

3.2.3 DEVICE ERROR MESSAGE

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

RECD=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

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 22

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 DIFFENT LENGTH THEN THE MESSAGE
THAT WAS EXPECTED.

MODEM STATUS CHANGES FOR THIS PASS WERE..
HARD CHANGES=XXXXX GLITCHES=XXXXX

WHERE XXXXX IS A 5 DIGIT DECIMAL NUMBER
THIS MSG IS ONLY PRINTED IF NUMBER OF
EITHER HARD CHANGES OR GLITCHES IS
GREATER THAN 0. A HARD CHANGE IS ONE
WHERE THE DUP WAS ABLE TO LATCH UP A
DIFFERENCE IN THE MODEM STATUS. A
GLITCH IS WHEN A MODEM STATUS INTERRUPT
OCCURS BUT THE DUP CANNOT FIND A
DIFFERENCE IN STATUS BIT.

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

MASTER RESET DID NOT WORK
RXCSR TXCSR
XXXXXX XXXXXXXX

THIS MEANS THAT AFTER A MASTER
RESET WAS ISSUED TO DUP THE
RXCSR REGISTER WAS NON ZERO.

NO CLEAR TO SEND FROM MODEM
RXCSR TXCSR
XXXXXXX XXXXXXXX

WHEN REQUEST TO SEND (RTS)
IS SET, MODEM DOES NOT RESPOND
WITH CLEAR TO SEND(CTS).

TIME OUT WAITING FOR RX OR TX TO COMPLETE
RXCSR TXCSR
XXXXXXX XXXXXXXX

THIS USUALLY MEANS AN OPEN
COMMUNICATION LINK.

MODEM DID NOT RETURN MODEM READY
RXCSR TXCSR
XXXXXXX XXXXXXXX

WHEN THE DTR SIGNAL WAS SET,
DATA SET READY WAS NOT RETURNED

CRC IN ERROR
RXDBUF RXCSR

A CRC ERROR WAS DETECTED BY
THE DUP ON AN INCOMING MESSAGE.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 23
 CZDCLB.P11 19-JUL-83 17:12

XXXXXX

XXXXXX

RECEIVER OVERRUN

RXDBUF

XXXXXXX

RXCSR

XXXXXXX

THE RECEIVER WASN'T SERVICED
 FAST ENOUGH (SOFTWARE)--
 CAUSING A CHARACTER TO BE LOST.

TIMED OUT IN START,STACK ACK SEQ

RDATA

XXXXXXX

SDATA

XXXXXXX

THIS USUALLY MEANS THAT THE DUP
 IS UNABLE TO ESTABLISH A
 CONNECTION WITH THE OTHER
 DEVICE BEING TESTED. THE VALUES
 IN RDATA AND SDATA SHOW THE
 RECIEVED(RDATA) AND TRANSMITTED
 (SDATA). SEE DDCMP SPEC. FOR
 FURTHER EXPLANATION OF STARTUP
 SEQUENCE.

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' IS REPORTED
 WHEN 'X' NUMBER OF DCLT PASSES' ARE 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
 'N' 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. THIS WILL TAKE YOU TO THE RPT> LEVEL. NOW
 INPUT THE 'LOG' COMMAND. THE EVENTS ARE PRINTED IN A 'LAST-IN
 FIRST-OUT' ORDER.

EVENT TIME IS TYPED OUT AS MM:SS:TT (LIKE 254:36:07) WHERE MM,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' 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 1ST BYTE OF MESSAGE,
 TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 24
CZDCLB.P11 19-JUL-83 17:12

TRANSMIT MESSAGE COMPLETED:
EVENT TIME, ADDRESS OF 1ST BYTE OF MESSAGE,
TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

RECEIVE SPACE QUEUED:
EVENT TIME, ADDRESS OF 1ST BYTE OF MESSAGE,
TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

RECEIVE MESSAGE COMPLETED:
EVENT TIME, ADDRESS OF 1ST BYTE OF MESSAGE,
TOTAL NO. OF BYTES, MODEM STATUS AT THAT TIME.

DATA COMPARISON STARTED:
EVENT TIME, 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 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 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:
^C ABORT:
EVENT TIME, 'DCLT' PASS COUNT, 'DCLT' ERROR COUNT,
AND THE 'STRT-TO'(COUNT OF START TIME OUTS).

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 25

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
MSC	ABNORMAL MODEM STATUS CHANGE
CMP	ABOUT TO DO DATA CHECKING OF RECVD VS. EXPTD DATA
CML	LENGTH ERROR OCCURRED DURING DATA COMPARISON
CMD	DATA ERROR OCCURRED DURING DATA COMPARISON
EOP	END OF PASS

NOTE:: BECAUSE THE DUP IS AN INTERRUPT DRIVEN DEVICE, IT IS BEST TO DISABLE STATUS TO PREVENT OVERRUN ERRORS.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 26
 CZDCLB.P11 19-JUL-83 17:12

4.3 PRINTING DDCMP STATISTICAL AND ERROR LOG

IF YOU ARE RUNNING THIS PROGRAM WITH DDCMP PROTOCOL ENABLED, YOU CAN EXAMINE (VIA 'RPT>' COMMAND) DDCMP STATISTICAL AND ERROR COUNTERS TO GET A BETTER UNDERSTANDING OF WHAT IS HAPPENING ON THE LINK. FOR A FULL DESCRIPTION OF THESE COUNTERS SEE (DIGITAL DATA COMMUNICATION MESSAGE PROTOCOL) SPECIFICATION VERSION 4.1.

BELOW IS A BRIEF DESCRIPTION OF EACH COUNTER. THE MOST IMPORTANT OF THESE ARE DATA MESSAGES SENT/RECEIVED AND DATA ERRORS IN/OUT.

OCTAL #	MESSAGE	MEANING
000000	STATUS FLAGS	USED ONLY IN SOFTWARE DEVELOPMENT.
000000	DATA MSGS TX	# MESSAGES TX'ED DURING THE TEST. RESET TO ZERO AT START OR RESTART. LATCHES AT -1.
000000	DATA MSGS RX	# MESSAGES RX'ED DURING THE TEST. RESET TO ZERO AT START OR RESTART. LATCHES AT -1.
000	HIGHEST MSG TX	MODULO 255 COUNTER. HIGHEST MESSAGE # SENT AND ACK'ED BY REMOTE STATION.
000	HIGHEST MSG ACK	MODULO 255 COUNTER. HIGHEST MESSAGE # RX'D BY REMOTE NODE. (WITH NO ERRORS)
000	NEXT MSG # TO TX	MODULO 255 COUNTER. ALWAYS 1 GREATER THEN CURRENT MESSAGE NUMBER BEING SENT.
000	LAST MSG # TX'ED	MODULO 255 COUNTER. ALWAYS SAME AS HIGHEST # SENT.
000	HIGHEST MSG# RX	NUMBER OF LAST MESSAGE RX'ED AND ACK'ED.
000	TRIB ADDR	IF MULTIPOINT THEN ADDRESS THIS STATION.
000	REMOTE TIME OUTS	MODULO 255 COUNTER. REPLYS RECEIVED AND ACK'ED.
000	GLOBAL CRC ERRORS	IF MULTIPOINT NETWORK-CRC ERRORS DETECTED.
000	NAK REASON	REASON FOR SENDING LAST NAK.
000	SEL THRESH ERRS	HALF/DUPLEX ONLY. SELECT TIME OUTS.
000	RX THRESH ERRS	INCREMENTED WHEN ERROR DETECTED IN INCOMING MESSAGE. (MODULO 8 COUNTER) RESET WHEN GOOD MESSAGE RECEIVED.
000	TX THRESH ERRS	INCREMENTED WHEN NAK RECEIVED. RESET WHEN ACK RECEIVED. (MODULO 8 COUNTER)

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 27

000	DATA ERRORS OUT	NAKS RECEIVED BECAUSE OF HEADER CRC ERROR OR DATA CRC ERRORS OR MESSAGE NOT RECEIVED AT ALL(REP). INDICATES NOISE ON TRANSMIT LINE.
000	DATA ERRORS IN	NAKS SENT BECAUSE HEADER CRC ERROR OR DATA CRC ERROR DETECTED IN INCOMING MESSAGE. MESSAGE TAKING NOISE HITS.
000	LOCAL BUFFER ERRS	EITHER NO BUFFER WAS AVAILABLE FOR INCOMING MESSAGE OR BUFFER THAT WAS AVAILABLE WAS TOO SMALL FOR INCOMING MESSAGE. USUALLY A SOFTWARE SPEED PROBLEM.
000	REMOTE BUFFER ERRS	SAME AS LOCAL BUT BUFFER PROBLEMS AT REMOTE STATION.
000	REMOTE STA ERRS	RX OVERRUN ERRORS(RX WASN'T SERVICED FAST ENOUGH) OR IF FORMAT ERROR A CRC EXISTED AND WASN'T DETECTED BY HARDWARE.
000	LOCAL STA ERRS	SAME AS REMOTE STATION ERRORS.
000	TX / RX THRESH ERR	OVERFLOW FROM RX OR TX THRESHOLD COUNTERS. INDICATES A PERSISTENT LINK PROBLEM THAT ISN'T CORRECTED AFTER 7 RETRIES.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 28
 CZDCLB.P11 19-JUL-83 17:12

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	1	:[0] FULL OR HALF DUPLEX FLAG (BIT0=1 IF FULL)
.WORD	160170	:[2] CSR ADDRESS
.WORD	300	:[4] INTERRUPT VECTOR
.WORD	240	:[6] INTERRUPT PRIORITY
.WORD	0	:[10] PT-PT=0 MULTIPOINT=1
.WORD	1	:[12] TRIB ADDRESS THIS STATION
.WORD	0	:[14] REMOTE NODE "ITEP"

6.0 MODE AND MESSAGE DESCRIPTIONS

6.1 MODE DESCRIPTIONS

THE FOLLOWING MODE DESCRIPTIONS REFER TO MESSAGE LISTS BEING TRANSMITTED AND RECEIVED BUT BE AWARE THAT OTHER DATA IS ALSO SENT AND RECEIVED. IF '/PROTOCOL' IS SELECTED THE DATA IS ENCLOSED IN A DDCMP ENVELOPE AND CONTROL MESSAGE WILL ALSO APPEAR ON THE LINK.

6.1.1 TRANSMIT MODE

A LIST OF MESSAGES IS TRANSMITTED WITHOUT EXPECTING ANY DATA TO BE RECEIVED. HOWEVER WITH '/PROTOCOL' ENABLED EACH MESSAGE SENT MUST BE ACKNOWLEDGED(ACK).

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

THEN EVERY TIME A MESSAGE IS RECEIVED, A MESSAGE IS TRANSMITTED.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 29
CZDCLB.P11 19-JUL-83 17:12

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.

NOTE: IF BOTH ENDS OF THE LINK ARE IN ACTIVE MODE, THEN THE
LINK MUST BE A FULL DUPLEX LINK!

6.1.5 DOWN-LINE-LOAD

DOWN-LINE-LOADING IS NOT SUPPORTED IN THE SOFTWARE FOR DUP-11
TO DUP-11 LINKS. HOWEVER IT IS POSSIBLE TO 'REQUEST SECONDARY
LOAD' FROM A HOST STATION (IF SUPPORTED) IF THERE IS A DUP(M9312)
DECNET BOOTSTRAP MODULE IN YOUR MACHINE.
SEE BOOTSTRAP OPERATOR'S MANUAL.

6.1.6 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

6.1.7 LISTEN MODE

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

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 30
 CZDCLB.P11 19-JUL-83 17:12

6.1.8 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 !

INTERNAL TTL LOOPS DATA INTERNAL TO THE USYNRT

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):

HALF DUPLEX START	STATION A 'HOST' NODE	'/LOOP' ALLOWED?	STATION B 'REMOTE' NODE	DUPLEX
B	TALK	NO	LISTEN*, RECEIVE	HALF OR FULL
A	LISTEN	NO	TALK*, TRANSMIT	HALF OR FULL
B	TRANSMIT	NO	RECEIVE*, LISTEN	HALF OR FULL
A	RECEIVE	NO	TRANSMIT*, TALK	HALF OR FULL
A	PASSIVE	NO	ACTIVE*	HALF OR FULL
-NA-	ACTIVE	YES	ACTIVE*	FULL
B	ACTIVE	YES	PASSIVE*	HALF OR FULL
-NA-	DOWNLINELOAD	** DOWN-LINE-LOADING IS NOT SUPPORTED FOR DUP-11		

*= MOST LIKELY TO BE IN THAT MODE

NOTE: H/D START COLUMN INDICATES WHICH NODE TO START FIRST ON A HALF DUPLEX LINK

IF PROTOCOL IS ENABLED, THE H/D START COLUMN CAN BE IGNORED.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 31

6.2 MESSAGE DESCRIPTIONS

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

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 32
 CZDCLB.P11 19-JUL-83 17:12

7.0 OTHER INFORMATION

7.1 INTERFACING TO AN "ITEP" NODE

THESE ARE THE RULES WHEN USING ITEP/WITH A DUV TO TALK
 TO A DUP USING DCLT.

ITEP NODE	DCLT NODE
-----------	-----------

ANSWER ALL QUESTION TO THE
 SET SWITCHES PROMPT.

ANSWER ALL QUESTIONS TO THE
 DCLT> PROMPT.

FOR ONE WAY OUT...
 SET SWITCHES TO 1221

CLEAR EXPECTED
 SET E=ITEP/S=56
 RUN MODE=REC/STATUS/CHECK/NOPROTOCOL

NOTE: DUV ITEP SENDS ONLY 56 CHARS

FOR ONE WAY IN.....

SET SWITCHES TO1222

RUN MODE=TRA/STATUS/NOPROTOCOL

FOR EXTERNAL LOOPBACK....

SET SWITCHES.....1224

CLEAR EXPECTED
 SET EXP=ITEP/S=56
 RUN MODE=ACTIVE/STATUS/CHECK/NOPROTOCOL

FOR INTERNAL LOOPBACK.....

SET SWITCHES.....1260

CLEAR EXPECTED
 SET EXP=ITEP/S=56
 RUN MODE=ACTIVE/STATUS/CHECK/NOPR

NOTE: DO NOT USE SWITCH 8 WITH ITEP GOING TO DCLT
 THE ONLY MESG. DCLT SUPPORTS IS MSG 1.
 DCLT IGNORES CRC ERRORS WHEN REC DATA FROM ITEP
 BECAUSE ITPE SENDS NO CRC.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 33
 CZDCLB.P11 19-JUL-83 17:12

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.

HINT::: IF THIS DOCUMENT IS TOO LARGE TO CONSUME, GET A COPY OF DEC'S COMMUNICATION OPTIONS MINI REFERENCE GUIDE(EK-CM1N1-RM-001).

REMEMBER, IF YOU ARE HAVING TROUBLE WITH RX OVERRUN ERRORS OR MISSED MESSAGES, OR DATA CHECK ERRORS-- DISABLE STATUS(/NOSTATUS). THE CPU IS HAVING A HARD TIME SERVICING BOTH THE TTY AND DUP.

EVEN IF YOU ARE CHECKING OUT DUP-11 TO DUP-11 LINKS, IT IS A GOOD IDEA TO ENABLE PROTOCOL(/PROTOCOL). BY EXAMINING THE DDCMP STATISTICAL AND ERROR LOG, YOU WILL GET A COMPLETE PICTURE OF WHAT IS HAPPENING ON THE LINK. NOISY COMM LINKS WILL BE DETECTED BETTER IF LARGE MESSAGES(512 CHARS) ARE SENT.

BOTH NODES MUST EITHER ENABLE('/PROTOCOL') OR DISABLE('/NOPROTOCOL').

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 INTERNAL LOOP AT EACH NODE

RUN EACH END OF THE LINK IN ACTIVE MODE WITH LOOP=INTERNAL.
 TRANSMIT TWO OR THREE MESSAGES WITH NO DATA CHECKING.
 STATUS PRINTING COULD BE TURNED OFF IF ON, BUT SEEING THE SEQUENCE
 OF EVENTS MIGHT BE INFORMATIVE.

A POSSIBLE COMMAND SEQUENCE IS:

```
C E
C T
SE T=ONES/S=20/C=2
R M=A/LO=1/NOCH/STAT/NPR
```

WHAT THE ABOVE COMMAND SEQUENCE MEANS:

THE 'C E' AND THE 'C T' INITIALIZES THE 'EXPECT'
 LIST AND THE 'TRANSMIT LIST'. THE 'SE T=ONES/S=20/C=2'
 SETS THE TRANSMIT LIST TO CONTAIN 3 MESSAGES. THE MESSAGES
 CONTAIN DATA OF ALL ONES AND EACH ONE IS 20 BYTES IN LENGTH.
 THE 'R M=A/LO=1/NOCH/STAT' SETS THE MODE TO RUN IN TO BE
 ACTIVE AND LOOP TYPE TO BE INTERNAL TTL. THE PROGRAM WILL
 NOT BE CHECKING DATA SO THERE WAS NO NEED TO SET UP AN
 EXPECT LIST. THE PROGRAM WILL BE PRINTING STATUS MESSAGES.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 34
 CZDCLB.P11 19-JUL-83 17:12

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

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

THIS GIVES YOU A IDEA IF THE COMM. DEVICE CAN EVEN TRANSMIT AND
 RECEIVE. ANY ERRORS REPORTED WILL PROBABLY BE DUE TO INCORRECT
 DEVICE ADDRESSES BEING USED OR A FAULTY DEVICE. CHECK ADDRESSES
 WITH 'DISPLAY' AND RUN THE PREREQUISITE DIAGNOSTICS FOR THE COMM.
 DEVICE.

NOW TRY RUNNING EACH NODE THE SAME WAY WITH DATA CHECKING ENABLED.
 A POSSIBLE COMMAND SEQUENCE IS:

```
SE E=T
R M=A/LO=1/CH/PAS=3/PR
```

WHAT THIS SEQUENCE MEANS:

THIS SEQUENCE IS SIMILAR TO THE ONE ABOVE. THE 'SE E=T'
 MAKES A COPY OF THE TRANSMIT LIST IN THE EXPECT LIST.
 THE EXPECT LIST NOW CONTAINS 3 MESSAGES. THE MESSAGES WILL
 HAVE ALL ONES FOR DATA AND BE 20 BYTES EACH IN LENGTH.
 THE RUN COMMAND IS THE SAME WITH THE ADDITION OF TWO
 SWITCHES '/CH/PAS=3'. THE 'CH' SWITCH TELLS THE PROGRAM
 TO CHECK THE RECEIVED DATA AGAINST THE 'EXPECTED LIST'.
 THE 'PAS=3' SWITCH TELLS THE PROGRAM TO RUN 3 PASSES BEFORE
 RETURNING TO THE DCLT> PROMPT. ON NON-DDCMP LINKS, THE
 '/PROTOCOL' SWITCH IS OPTIONAL.

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 35
 CZDCLB.P11 19-JUL-83 17:12

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

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

IF A CABLE TURNAROUND CONNECTOR IS AVAILABLE, PUT IT ON THE END OF
 THE CABLE JUST BEFORE THE MODEM AND RUN IN ACTIVE MODE WITH THE
 "/LOOP=CABLE" SWITCH.

POSSIBLE COMMAND SEQUENCE IS:

R M=A/L=C/CH/PAS=3

WHAT THIS SEQUENCE MEANS:

THIS SEQUENCE HAS THE "/LO=C". THIS INFORMS THE SOFTWARE
 NOT TO CHECK FOR DATA SET READY SIGNAL FROM THE MODEM.
 ALSO A CLOCK SIGNAL IS FURNISHED BY THE DUP.

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

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

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
-----	-----
C E	C E
C T	C T
R M=TR/NOCH/PAS=3/NPR	R M=R/NOCH/PAS=3/NPR

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 36
 CZDCLB.P11 19-JUL-83 17:12

WHAT THIS SEQUENCE MEANS:

THE 'C E ' AND 'C T' INITIALIZE BOTH THE TRANSMIT AND EXPECT LISTS. THE 'R M=TR/PAS=3' SETS THE RUN MODE OF NODE A TO BE TRANSMIT AND THE PASS COUNT IS SET TO 3. THE 'R M=R/NOCH/PAS=3' SETS THE RUN MODE OF NODE B TO RECEIVE WITH NO DATA CHECKING AND THE PASS COUNT IS SET TO THREE. PROTOCOL CAN BE OPTIONAL BUT IT MUST BE ENABLE OR DISABLE ON BOTH ENDS.

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/NOPROTOCOL
DCLT> (A) ?
```

FOR NODE B:

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

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

R M=TR/PAS=3

R M=R/CH/PAS=3

WHAT THIS SEQUENCE MEANS:

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/CHECK/NOECHO/NOMODEM/PROTOCOL
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.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 37

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
-----	-----
C E	C E
C T	C T
SE T=CCITT/S=10/C=2	SE T=1ALT/S=20/C=2
R M=ACT/NOCH/PAS=3	R M=P/NOCH/PAS=3

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. 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. THE NODE IS THEN 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 RXQ TXQ TXC RXQ
TXQ TXC EOP RXQ TXQ TXC RXQ TXQ
TXC RXQ TXQ TXC EOP RXQ TXQ TXC
RXQ TXQ TXC RXQ TXQ TXC EOP
MODE=ACTIVE/PASS=00000
/STATUS/NOCHECK/NOECHO/NOMODEM/NOPROTOCL
DCLT> (A) ?
```

FOR NODE B:

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

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 38
CZDCLB.P11 19-JUL-83 17:12

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.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 39

7.2.4 BOTH NODES ACTIVE

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

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

WHAT THIS SEQUENCE MEANS:

NODE A SETS UP IS 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
DATA CHECKING AND PASS COUNT EQUAL TO THREE.

WHAT YOU SHOULD SEE AFTER ENTERING THE RUN COMMAND
IF THINGS ARE RUNNING CORRECTLY :
ON BOTH NODES A AND B:

```

MODE=ACTIVE/PASS=00000
/NOSTATUS/CHECK/NOECHO/NODEM/NOPROTOCOL
DCLT> (A) ?

```

A GOOD VARIATION THAT CAN BE USED IS TO LOAD THE TRANSMIT LIST AND
EXPECT LIST WITH A LARGE MESSAGE(512 CHARACTERS),ENABLE PROTOCOL,
AND RUN MANY PASSES ON BOTH ENDS.

```

DCLT>(A)? CL T
DCLT>(A)? CL E
DCLT>(A)? SET T=CCITT/SIZE=512
DCLT>(A)? SET E=T
DCLT>(A)? R M=A/NST/CH/PA=255/PR

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 40

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.

R M=LIS/NOST
LIS>

R M=TA/NOST
TLK>

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 MESSAGES COMMANDS

THE CLEAR COMMANDS .

C E
C T

THIS WILL INITIALIZE THE TRANSMIT AND EXPECT LIST TO 1 MESSAGE OF 58 BYTES. THE DATA OF THE MESSAGE WILL BE THE ITEP MESSAGE.

IF THESE COMMANDS ARE FOLLOWED BY A SHOW COMMAND

SH E

SUCH AS THE SHOW EXPECT LIST. WHAT YOU WOULD SEE IS

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

DCLT> (A) ?

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

SE E=A/S=35/C=3

AND FOLLOWED IT WITH A SHOW EXPECT LIST COMMAND

SH E

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 41
CZDCLB.P11 19-JUL-83 17:12

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/NOPROTOCOL

DCLT> (A) ?

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 42
 CZDCLB.P11 19-JUL-83 17:12

7.3.2 EXAMPLES STATISTICAL COMMANDS

IF YOU TYPE A HELP COMMAND

HELP

WHAT YOU WILL SEE IS

DCLT CMDS:

CLEAR OR SHOW EXPECTLIST OR TRANSMITLIST

PRINT

EXIT

DUMP START-END/B

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

SET EXPECT=TRANSMIT

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

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

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

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

LTP=INT,CAB,LOC,REM/

DCLT> (A) ?

THE SAME WILL HAPPEN IF YOU USE THE ?

THE DUMP COMMAND WORKS LIKE THIS

DUM 41260-41300

THIS WILL DUMP THE DATA FROM ADDRESSES 41260 TO
 41300 IN THE FOLLOWING MANNER

41260 104423 000167 177772 021122 012112 006312 006312 006312

41300 006312

IF YOU HAD USED THE /B SWITCH

DUM 41260-41300/B

WHAT YOU WOULD SEE IS

41260 023 211 167 000 372 377 122 024

41270 112 024 312 014 312 014 312 014

41300 312

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 43

7.3.3 EXAMPLES RUN COMMANDS

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

IF YOU WERE TO EXECUTE THE RUN COMMAND

R M=TR/NOST/CH/PAS=4

WHAT WOULD HAPPEN IS AFTER 4 PASSES THE PROGRAM WOULD RETURN TO THE DCLT PROMPT AND PRINT

MODE=TRANSMIT/PASS=00000

/NOSTATUS/CHECK/NOECHO/NOMODEM/NOPRTCOL

DCLT> (A) ?

IF YOU WERE TO EXECUTE THE RUN COMMAND

C E

C T

R M=A/LO=1/ST/CH/PAS=3/PROTOCOL

WHAT YOU WOULD SEE (IF USING DEFAULT TRANSMIT AND EXPECT MESSAGES) IS

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/PROTOCOL

DCLT> (A) ?

IF YOU USE THE EXIT COMMAND

EXIT

WHAT YOU WOULD SEE IS

CZDCL EOP

0 CUMULATIVE ERRORS

DR>

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 45
CZDCLB.P11 19-JUL-83 17:12

7.3.5 EXAMPLE EXIT COMMAND

THE EXIT COMMAND WORKS LIKE THIS. IF YOU ENTERED THE REPORT LEVEL FROM THE SUPERVISOR (DR>) THEN TYPING

EXIT

WILL RETURN YOU TO THE SUPERVISOR.

DR>

IF YOU ENTERED REPORT FROM THE DCLT LEVEL THEN TYPING

EXIT

WILL RETURN YOU TO THE DCLT LEVEL.

DCLT>

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 PASSIVE 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. ALSO DISABLE DATACHECKING('/NOCHECK').

IF YOU ARE RUNNING HALF/DUPLEX IT IS BEST TO USE THE '/NOMODEM' SWITCH BECAUSE EVERY TIME THE LINE IS TURNAROUND A MODEM CHANGE WILL BE REPORTED.

IF YOU ARE RUNNING WITH '/PROTOCOL' SELECTED THE MODEM STATUS AS REPORTED IN THE EVENT LOG MAY NOT INDICATE THE TRUE CONDITION OF THE MODEM SIGNALS. THIS IS BECAUSE THE EVENT IS LOGGED BEFORE THE MESSAGE IS PASSED TO THE DDCMP PROTOCOL LAYER WHERE THE RX,TX AND MODEM SIGNALS ARE MANIPULATED.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 46

1987
1988
1989
1990
1991
1992
1993
1994
1995 002000
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006 002000
2007
2008
2009
2010
2011
2012 002000
2013 002000
2014 002000 103
2015 002001 132
2016 002002 104
2017 002003 103
2018 002004 114
2019 002005 000
2020 002006 000
2021 002007 000
2022 002010
2023 002010 102
2024 002011
2025 002011 060
2026 002012
2027 002012 000000
2028 002014
2029 002014 003410
2030 002016
2031 002016 046230
2032 002020
2033 002020 000000
2034 002022
2035 002022 002130
2036 002024
2037 002024 000000
2038 002026
2039 002026 046616
2040 002030
2041 002030 000000
2042 002032

.SBTTL PROGRAM HEADER

BGNMOD

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

POINTER BGNRPT,BGNAU,BGNDU

HEADER CZDCL,B,0,1800.,0,#PRI07

LSNAME::
 .ASCII /C/
 .ASCII /Z/
 .ASCII /D/
 .ASCII /C/
 .ASCII /L/
 .BYTE 0
 .BYTE 0
 .BYTE 0
LSREV::
 .ASCII /B/
LSDEPO::
 .ASCII /O/
LSUNIT::
 .WORD 0
LSTIML::
 .WORD 1800.
LSHPCP::
 .WORD LSHARD
LSSPCP::
 .WORD 0
LSHPTP::
 .WORD LSHW
LSSPTP::
 .WORD 0
LSLADP::
 .WORD LSLAST
L\$STA::
 .WORD 0
L\$CO::

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 47
PROGRAM HEADER

2043 002032 000000
2044 002034
2045 002034 000000
2046 002036
2047 002036 000000
2048 002040
2049 002040 002124
2050 002042
2051 002042 000340
2052 002044
2053 002044 000000
2054 002046
2055 002046 000000
2056 002050
2057 002050 003
2058 002051 003
2059 002052
2060 002052 000000
2061 002054 000000
2062 002056
2063 002056 000000
2064 002060
2065 002060 011524
2066 002062
2067 002062 025336
2068 002064
2069 002064 000000
2070 002066
2071 002066 000000
2072 002070
2073 002070 026334
2074 002072
2075 002072 026326
2076 002074
2077 002074 000000
2078 002076
2079 002076 011534
2080 002100
2081 002100 104035
2082 002102
2083 002102 000000
2084 002104
2085 002104 025352
2086 002106
2087 002106 026240
2088 002110
2089 002110 026236
2090 002112
2091 002112 025344
2092 002114
2093 002114 000000
2094 002116
2095 002116 000000
2096 002120
2097 002120 000000
2098

LSDTYP:: .WORD 0
LSAPT:: .WORD 0
LSDTP:: .WORD 0
LSPRIO:: .WORD LSDISPATCH
LSENV1:: .WORD #PRI07
LSEXP1:: .WORD 0
LSMREV:: .WORD 0
LSEF:: .BYTE CSREVISION
 .BYTE CSREDIT
LSSPC:: .WORD 0
LSDEVP:: .WORD 0
LSREPP:: .WORD LSDVTYP
LSEXP4:: .WORD LSRPT
LSEXP5:: .WORD 0
LSAUT:: .WORD 0
LSDUT:: .WORD LSAU
LSLUN:: .WORD LSDU
LSDESP:: .WORD 0
LSLOAD:: .WORD LDESC
LSETP:: EMT ESLOAD
LSICP:: .WORD 0
LSCCP:: .WORD LSINIT
LSACP:: .WORD LSCLEAN
LSPRT:: .WORD LSAUTO
LSTEST:: .WORD LSPROT
LSDLY:: .WORD 0
LSHIME:: .WORD 0

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 48
DISPATCH TABLE

.SBTTL DISPATCH TABLE

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

DISPATCH 1

2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110

002122
002122 000001
002124
002124 026342

.WORD 1
L\$DISPATCH::
.WORD T1

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 49
DEFAULT HARDWARE P-TABLE

```

2111
2112
2113
2114
2115
2116
2117
2118
2119
2120 002126
2121 002126 000010
2122 002130
2123 002130
2124
2125
2126
2127
2128
2129
2130
2131 002130 000001
2132
2133
2134
2135
2136
2137
2138
2139 002132 160170
2140 002134 000300
2141 002136 000240
2142 002140 000000
2143 002142 000001
2144 002144 000000
2145 002146 000000
2146
2147
2148 002150
2149 002150

```

.SBTTL DEFAULT HARDWARE P-TABLE

```

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

```

BGNHW DFPTBL

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

```

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

```

.WORD 1 ;[0] FULL OR HALF DUPLEX FLAG (BIT0=1 IF FULL)

```

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

```

.WORD 160170 ;[2] CSR ADDRESS
.WORD 300 ;[4] INTERRUPT VECTOR
.WORD 240 ;[6] INTERRUPT PRIORITY (5)
.WORD 0 ;[10] MULTI POINT =1 PT TO PT = 0
.WORD 1 ;[12] TRIB ADDRESS THIS STATION
.WORD 0 ;[14] OTHER NODE "ITEP"
.WORD 0 ;[16] SPARE

ENDHW

L10000:

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 50
DEFAULT HARDWARE P-TABLE

.SBTTL GLOBAL EQUATES SECTION

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

EQUALS

: BIT DEFINITIONS

2165	002150	
2166		
2167		
2168		
2169	100000	BIT15== 100000
2170	040000	BIT14== 40000
2171	020000	BIT13== 20000
2172	010000	BIT12== 10000
2173	004000	BIT11== 4000
2174	002000	BIT10== 2000
2175	001000	BIT09== 1000
2176	000400	BIT08== 400
2177	000200	BIT07== 200
2178	000100	BIT06== 100
2179	000040	BIT05== 40
2180	000020	BIT04== 20
2181	000010	BIT03== 10
2182	000004	BIT02== 4
2183	000002	BIT01== 2
2184	000001	BIT00== 1
2185		:
2186	001000	BIT9== BIT09
2187	000400	BIT8== BIT08
2188	000200	BIT7== BIT07
2189	000100	BIT6== BIT06
2190	000040	BIT5== BIT05
2191	000020	BIT4== BIT04
2192	000010	BIT3== BIT03
2193	000004	BIT2== BIT02
2194	000002	BIT1== BIT01
2195	000001	BIT0== BIT00

: EVENT FLAG DEFINITIONS

: EF32:EF17 RESERVED FOR SUPERVISOR TO PROGRAM COMMUNICATION

2200	000040	EF.START== 32.	: START COMMAND WAS ISSUED
2201	000037	EF.RESTART== 31.	: RESTART COMMAND WAS ISSUED
2202	000036	EF.CONTINUE== 30.	: CONTINUE COMMAND WAS ISSUED
2203	000035	EF.NEW== 29.	: A NEW PASS HAS BEEN STARTED
2204	000034	EF.PWR== 28.	: A POWER-FAIL/POWER-UP OCCURRED
2205		:	

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 51
GLOBAL EQUATES SECTION

2206		:
2207		: PRIORITY LEVEL DEFINITIONS
2208		:
2209	000340	PRI07== 340
2210	000300	PRI06== 300
2211	000240	PRI05== 240
2212	000200	PRI04== 200
2213	000140	PRI03== 140
2214	000100	PRI02== 100
2215	000040	PRI01== 40
2216	000000	PRI00== 0
2217		:
2218		: OPERATOR FLAG BITS
2219		:
2220	000004	EVL== 4
2221	000010	LOT== 10
2222	000020	ADR== 20
2223	000040	IDU== 40
2224	000100	ISR== 100
2225	000200	UAM== 200
2226	000400	BOE== 400
2227	001000	PNT== 1000
2228	002000	PRI== 2000
2229	004000	IXE== 4000
2230	010000	IBE== 10000
2231	020000	IER== 20000
2232	040000	LOE== 40000
2233	100000	HOE== 100000
2234		

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 52
GLOBAL EQUATES SECTION

```

2235      ;***** INDEPENDENT EQUATES
2236
2237      001000      BUFLIM=512.      ;MAX BUFFER SIZE IN BYTES
2238
2239      000017      MSGLIM=15.      ;APPLIES TO TX,RX AND CMP BUFFS
2240      ;MAX NO. OF MESSAGES PER BUFFER
2241      ;(FOR EACH INCREMENT (+1) TO MSGLIM,
2242      ;ADD 6 WORDS TO THE POINTER TABLE
2243      ;(PTRTAB:) SINCE THIS MEANS 2 MORE
2244      ;'POINTER' WORDS PER BUFFER.
2245
2246      ;MODE OF OPERATION EQUATES
2247      000000      REC=0      ;RECEIVE MODE
2248      000001      TRA=1      ;TRANSMIT MODE
2249      000002      PAS=2      ;PASSIVE MODE
2250      000003      ACT=3      ;ACTIVE MODE
2251      000004      DOW=4      ;DOWN-LINE-LOAD MODE
2252      000005      TAL=5      ;TALK MODE
2253      000006      LIS=6      ;LISTEN MODE
2254
2255      ;MAINT LOOP TYPE EQUATES
2256      000000      NONE= 0      ;NO LOOP
2257      000001      TTL= 1      ;INTERNAL TTL
2258      000002      CABLE= 2      ;CABLE LOOP
2259      000003      MODLOC= 3      ;MODEM LOCAL
2260      000004      MODREM= 4      ;MODEM REMOTE
2261      000005      MOP= 5      ;MOP
2262
2263      ;CLOCK ENABLE VALUES TO BE LOADED IN CLK'S CSR
2264      000100      LCLKEN= 100      ;L-CLOCK CSR VALUE TO ENABLE THE CLOCK
2265      000111      PCLKEN= 111      ;P-CLOCK CSR VALUE TO ENABLE THE CLOCK
2266      001600      PCLKCT= 1600      ;P-CLOCK COUNT SET REGISTER FOR COUNTER
2267
2268      ;PARAM WORD EQUATES
2269      000001      STATB= BIT0      ;OPERATOR AWAKE ASKED FOR
2270      000002      DATCKB= BIT1      ;DATA CHECK BIT
2271      000004      ECHOB= BIT2      ;ECHO BIT
2272      000010      MOCHK= BIT3      ;MODEM STATUS CHECK BIT
2273      000020      CRCB= BIT4      ;CRC CALCUALTE ASKED FOR
2274      000040      PROTOB= BIT5      ;PROTOCOL PROCESSING ASKED FOR
2275      000100      PRORUN= BIT6      ;PROTOCOL IS RUNNING(NOT STARTING OR MAINT)
2276      000200      ABORT= BIT7      ;FATAL PROTOCOL ERROR(SET IN PROTOCOL CODE)
2277
2278      ;OPTION TYPE EQUATES
2279
2280      000000      DPV= 0      ;CODE FOR DPV CHAR MODE
2281
2282      ;EVENT LOG MESSAGE TYPES (USED TO LOCATE EVENT DESCRIPTION IN EVENT TABLE
2283      ;AND DISPATCHING TO SEPERATE SECTIONS OF THE EVENT REPORTING SECTION)
2284
2285      000000      TXQ= 0      ;TRANSMIT MESSAGE QUEUED
2286      000002      TXC= 2      ;TRANSMIT COMPLETE
2287      000004      RXQ= 4      ;RECEIVE BUFFER QUEUED
2288      000006      RXC= 6      ;RECEIVE COMPLETE
2289      000010      DER= 10      ;DEVICE INFORMATION
2290      000012      DVI= 12      ;DEVICE ABOUT TO INIT

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 53
GLOBAL EQUATES SECTION

2291	000014	DCK= 14	;DATA COMPARISON RESULTS
2292			
2293	000016	MSC= 16	;MODEM STATUS CHANGE
2294			
2295	000020	DLE= 20	;DATA COMPARISON LENGH ERROR
2296	000022	DDE= 22	;DATA COMPARISON DATA ERROR
2297	000024	EOP= 24	;END OF PASS
2298	000026	ABO= 26	;^C ABORT
2299			
2300		;EQUATES FOR FLAG WORD	
2301			
2302	000001	ININT= BIT0	;INPUT INT. REC.
2303	000002	OTINT= BIT1	;OUTPUT INT REC
2304	000004	QRX= BIT2	;RX QUED /COMPL
2305	000010	QTX= BIT3	;TX QUED/COMPL
2306	000100	ERX= BIT6	;EXPECT TO GET A RX COMPLED
2307	000200	ETX= BIT7	;EXPECT TO GET A TX COMPLETED
2308			
2309			
2310	000020	TXM= BIT4	;INDICATES TO TX INTERRUPT ROUTINE
2311			;THAT IT IS TIME TO TRANSMIT BODY OF MSG.
2312	000040	RXM= BIT5	;INDICATES TO RX INTERUPPT ROUTINE
2313			;THAT IT IS TIME TO REC MSG BODY
2314	000400	BCC= BIT8	;TIME FOR CRC CHECK.
2315			
2316	001000	PAD= BIT9	;INDICATES THAT PAD MUST BE SENT
2317			
2318	002000	INOVN= BIT10	;INIT OVER
2319			
2320	004000	FIRST= BIT11	;FIRST TIME FOR CTS
2321			
2322		; SPECIAL CLI CODES FOR 'CHAR' ARGUMENT IN CLI CALLS	
2323		; (COMMAND LINE INTERPRETER DEFINITIONS)	
2324	000000	CLIERR= 0	
2325	000001	CLIEXI= 1	
2326	000002	CLIBR= 2	
2327	000003	CLIBIF= 3	
2328	000004	CLISPA= 4	
2329	000005	CLINUM= 5	
2330	000006	CLIALP= 6	
2331	000007	CLIALN= 7	
2332	000010	CLIOCT= 8.	
2333	000011	CLIDEC= 9.	
2334	000012	CLISTR= 10.	
2335			
2336		; DEFS FOR COMMAND LINE INTERPRETATION ACTION VALUES	
2337	000000	NULL=0	
2338	000001	CLEAR=1	
2339	000002	SHOW=2	
2340	000003	CHECK=3	
2341	000004	RUN=4	
2342	000005	HLP=5	
2343	000006	CSHEXP=6	
2344	000007	CSHTRN=7	
2345	000010	SETEXP=10	
2346	000011	SETTRN=11	

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 54
GLOBAL EQUATES SECTION

2347	000012	SIZE=12	
2348	000013	QCOPY=13	
2349	000014	NUM=14	
2350	000015	OPRMSG=15	
2351	000016	STATUS=16	
2352	000017	ENDQO=17	
2353	000020	CMSG0=20	
2354	000021	CMSG1=21	
2355	000022	CMSG2=22	
2356	000023	CMSG3=23	
2357	000024	CMSG4=24	
2358	000025	CMSG5=25	
2359	000026	CMSG6=26	
2360	000027	ATVMOD=27	
2361	000030	PASMOD=30	
2362	000031	RECMOD=31	
2363	000032	LISMOD=32	
2364	000033	DLLMOD=33	
2365	000034	TRAMOD=34	
2366	000035	TALMOD=35	
2367	000036	NO=36	
2368	000037	ECHO=37	
2369	000040	CRC=40	
2370	000041	PROTO=41	
2371	000042	PASC=42	
2372	000043	MOP=43	
2373	000044	TTLLOP=44	
2374	000045	CBLLLOP=45	
2375	000046	LMDLOP=46	
2376	000047	RMDLOP=47	
2377	000050	NOTNUF=50	
2378	000051	BADCHR=51	
2379	000052	DMPS=52	
2380	000053	DMPE=53	
2381	000054	DMPQ=54	
2382	000055	PRNT=55	
2383	000056	MOSC=56	
2384	000057	EXIT=57	
2385	000060	SETET=60	
2386		;FOLLOWING EQUATES USED IN REPORT CLI	
2387	000001	RPHLP=1	;PRINT HELP MESSAGE
2388	000002	RPEXT=2	;EXIT
2389	000003	RPLOG=3	;REPORT EVENT LOG
2390	000004	RPERR=4	; 'COUNTER/ERROR'
2391	000005	RPFUL=5	; 'COUNTER/FULL'
2392	000006	RNOTNF=6	;MORE COMMAND NEEDED
2393	000007	RPSWO=7	;VALIDATE OFFSET
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	020000	CTS= BIT13	;CLEAR TO SEND (CIRCUIT CB)
2402	001000	DSR= BIT9	;DATA SET READY (CIRCUIT CC)

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 55
GLOBAL EQUATES SECTION

2403	010000	DCD=	BIT12	;DATA CARRIER DETECT (CIRCUIT CF)
2404	000004	RTS=	BIT2	;REQUEST TO SEND (CIRCUIT CA)
2405	040000	RI=	BIT14	;RING INDICATOR (CIRCUIT CE)
2406	004000	SRD=	BIT11	;SECONDARY RECEIVE DATA
2407				
2408				
2409				
2410				
2411		; DEVICE SIGNALS		
2412	000002	DTR=	BIT1	;DATA TERMINAL READY
2413	000010	HDPLX=	BIT3	;HALF DUPLEX MODE
2414	000020	RXENA=	BIT4	;RECEIVER ENABLE
2415	000040	DSITEN=	BIT5	;DATA SET CHANGE ENABLE
2416	000100	RINTEN=	BIT6	;REC INT. ENABLE
2417	000200	RXDONE=	BIT7	;REC DATA READY
2418	004000	RXACT=	BIT11	;REC ACTIVE
2419	000400	RESET=	BIT8	;MASTER RESET
2420	001000	TXACT=	BIT9	;TX ACTIVE
2421	000200	TXDONE=	BIT7	;TX BUFFER EMPTY
2422	004000	TTLL=	BIT11	;TTL LOOP BIT(INTERNAL)
2423	010000	CABLOP=	BIT12	;CABLE LOOP(TURN AROUND)
2424	000020	SEND=	BIT4	;TX ENABLE
2425	000100	TINTEN=	BIT6	;TX INT ENABLE
2426	000400	TSOM=	BIT8	;TX START OF MSG.
2427	001000	TEOM=	BIT9	;TX END OF MSG.
2428	100000	TERR=	BIT15	;TX ERROR
2429	100000	RERR=	BIT15	;REC OVER RUN
2430	010000	CRCOK=	BIT12	;CRC CHAR OK
2431	100000	DSCA=	BIT15	;DATA SET CHANGE A
2432	000400	STRIP=	BIT8	;SYNC STRIP
2433	000226	SYN=	226	;SYNC WORD
2434				

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 56
GLOBAL DATA SECTION

2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
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
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490

002150
002150 000001
002152 000001
002154 000001
002156 000001
002160 000100
002162 000072
002164 000101
002166 000000
002170 000001

002172
002172 002214
002174 002215
002176 002216
002200 002217
002202 002220
002204 002320
002206 002412
002210 002520
002212 002642

002214 000
002215
002215 377
002216
002216 252
002217
002217 125
002220
002220 177603 157427 031011
002226 047321 163715 105221
002234 143325 142304
002240 040041 014116 052606
002246 172334 105025 123754
002254 111337 111523
002260 030030 145064 137642
002266 143531 063617 135075
002274 066730 026575
002300 052012 053627 070071
002306 151172 165044 031605
002314 166632 016741
002320

.SBTTL GLOBAL DATA SECTION
.SBTTL DEFAULT MESSAGE DEFINITIONS AND TABLES

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

;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

;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:			

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 57
DEFAULT MESSAGE DEFINITIONS AND TABLES

2491	002320				MSG5:	;'INTERPROCESSOR TEST PROGRAM'S (ITEP)'' MESSAGE
2492						: #1 (DP1:)
2493	002320	077577	040444	052040	.ASCII	<177><177>/SA THE QUICK BROWN FOX JUMPED OVER THE LAZY DOG./
2494	002326	042510	050440	044525		
2495	002334	045503	041040	047522		
2496	002342	047127	043040	054117		
2497	002350	045040	046525	042520		
2498	002356	020104	053117	051105		
2499	002364	052040	042510	046040		
2500	002372	055101	020131	047504		
2501	002400	027107				
2502	002402	005015	077401	077577	.ASCIIZ	<15><12><001><177><177><177><177>
2503	002410	000177				
2504	002412				MSG5:	
2505	002412				MSG6:	:ALPHA-NUMERICS (OR FUTURE COMM TURNAROUND MSG)
2506	002412	022043	021041	023040	.ASCII	/NS!' &'()*+,-.0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ/
2507	002420	024047	025051	026053		
2508	002426	027055	030460	031462		
2509	002434	032464	033466	034470		
2510	002442	035472	036474	037476		
2511	002450	040500	041502	042504		
2512	002456	043506	044510	045512		
2513	002464	046514	047516	050520		
2514	002472	051522	052524	053526		
2515	002500	054530	132			
2516	002503	057	056133	057135	.ASCIIZ	?/[\\]^_`?
2517	002510	022537	000			
2518	002513				MSG6:	
2519		002514			.EVEN	
2520						
2521					: *****	
2522					:THESE THREE STORAGE AREAS MUST NOT BE SEPERATED !!!!	
2523						
2524	002514	047045	040445		OPBFPT: .ASCII	/XNXA/
2525	002520	000122			OPBUF: .BLKB	82. ;BUFFER FOR OPERATOR SPEC'D MESSAGES
2526	002642				OPEND:	
2527						
2528					: THE ABOVE THREE LINES MUST BE KEPT TOGETHER	
2529					: *****	
2530						
2531	002642	033			MSG8: .BYTE	33 ;RECEIVE BUFFER FILL PATTERN
2532	002643				MSG8:	
2533		002644			.EVEN	

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 58
DEFAULT MESSAGE DEFINITIONS AND TABLES

2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560

002644 000
002645 201
002646
002646 000000
002650 001
002651 001
002652 001
002653
002654
002654 000006

002656 000
002657 201
002660
002660 000000
002662 001
002663 001
002664 001
002664 002666

.....
: THE FOLLOWING IS THE AREA USED TO TRANSMIT AND REC THE :
: HEADER MSGS. AND THE START,STACK ACK SEQUENCES. :
:.....

:: THE TRANSMIT HEADER MESSAGE WILL BE STORED HERE
HDMMSG: .BYTE 0 ;FILLER
HDMID: .BYTE 201 ;MESSAGE TYPE WILL BE STORED HERE
HDMTYP: ;IF CONTROL MESSAGE, TYPE IS STORED HERE
HDMCC: .WORD 0 ;CHAR COUNT GOES HERE
HDMREP: .BYTE 1 ;RESPONSE NUMBER
HDMNUM: .BYTE 1 ;MSG. NUMBER
HMDADR: .BYTE 1 ;ADDR TO.
HSMSE:
HDMC: .EVEN
.WORD 6 ;CHARACTER COUNT OF HEADER

:: THE RECEIVED HEADER WILL BE STORED HERE
RHDMMSG: .BYTE 0
RHDMID: .BYTE 201 ;MESSAGE TYPE GOES HERE
RHDTYP: ;IF CONTROL MESSAGE, TYPE GOES HERE
RHDMCC: .WORD 0 ;BYTE COUNT GOES HERE
RHDMREP: .BYTE 1 ;RESP NUM
RHDMNUM: .BYTE 1 ;MSG NUM
RHMDADR: .BYTE 1 ;ADDRESS OF TRIB
.EVEN

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 59
DEFAULT MESSAGE DEFINITIONS AND TABLES

```

;COMMAND LINE BUFFER, DATA LOCATIONS AND MESSAGES FOR ACTION ROUTINES

```

```

CMBUF: .BLKB      82.      ;BUFFER FOR OPERATOR COMMANDS
KEYWD1: .WORD      0       ;THIS LOC WILL =1 IF CLEAR TYPED, 2 FOR SHOW,
                             ; A 4 IF RUN WAS TYFED, 5 IF HELP WAS TYPED
QUALFG: .WORD      0       ;THIS LOC HOLDS QUALIFIER VALUE (SIZE OR COPY)
QUALVL: .WORD      0
HLPTAB: .WORD      HLP1
        .WORD      HLP2
        .WORD      HLP3
        .WORD      HLP3A
        .WORD      HLP4
        .WORD      HLP4A
        .WORD      HLP5
        .WORD      HLP6

```

```

HLPEND:
;INDEX TABLE FOR REPORT 'RPT>' HELP MESSAGES

```

```
RHLPT8: .WORD RHLP1
         .WORD RHLP2
         .WORD RHLP3
         .WORD RHLP4
         .WORD RHLP5
         .WORD RHLP6
```

RHLPEN:

SHTYTB: .WORD SHTYP0,SHTYP1,SHTYP2,SHTYP3,SHTYP4,SHTYP5,SHTYP6,SHTYP7

```
; THE LIST OF BYTES BELOW ARE THE FIRST BYTES OF THE PREDEFINED MESSAGES
; USED TO 'SHOW' THE TRANSMIT AND COMPARE BUFFER CONTENTS.
```

SHTAB: .BYTE 0,377,252,125,203,177,043

SHTEND: .EVEN

```

MODES:  .WORD    M00      ;ADDRESSES OF MODE TYPES IN ASCII
        .WORD    M01
        .WORD    M02
        .WORD    M03
        .WORD    M04
        .WORD    M05
        .WORD    M06

```

```

LOOPS:  .WORD    LP0      ;ADDRESSES OF LOOP TYPES IN ASCII
        .WORD    LP1
        .WORD    LP2
        .WORD    LP3
        .WORD    LP4

```

```

;COMMAND LINE TRAVERSE LOCATIONS (USED BY 'P$TRV')

```

```

PSBUFA: .WORD    0           ;LOC. TO HOLD ADDR. OF CMD LINE BUFFER
PSTREE: .WORD    0           ;LOC. TO HOLD ADDR. OF PARSING TREE

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 60
DEFAULT MESSAGE DEFINITIONS AND TABLES

2617	003136	000000	PSACT: .WORD	0	;LOC. TO HOLD ADDR. OF ACTION ROUTINE
2618	003140	000000	PSCNT: .WORD	0	;LOC. TO BE A COUNTER LOCATION
2619	003142	000000	PSNUM: .WORD	0	;LOC. TO HOLD NUMERIC VALUE FROM PARSE
2620	003144	000000	PSRADX: .WORD	0	;LOC. TO HOLD RADIX USED(LJ) AND +/- (HI BYTE)
2621	003146	000	PSNNUF: .BYTE	0	;RETURN =0 IF ENOUGH OF COMMAND FOUND
2622	003147	000	PSGDBD: .BYTE	0	;RETURN CODE 0 IF NO ERROR FOUND
2623					

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 61
MESSAGE BUFFERS AND POINTER TABLES

2624			.SBTTL		MESSAGE BUFFERS AND POINTER TABLES
2625					
2626	003150	001000	TXBUF:	.BLKB	BUFLIM ; TRANSMITTER BUFFERS
2627	004150	001000	RXBUF:	.BLKB	BUFLIM ; RECEIVER BUFFERS
2628	005150	001000	CMPBUF:	.BLKB	BUFLIM ; COMPARISON BUFFERS
2629	006150	000036	PTRTAB:	.BLKW	MSG LIM*2 ; TABLE FOR MESSAGE ADDRS. & BYTE COUNTS
2630	006244	000036	PTR13:	.BLKW	MSG LIM*2
2631	006340	000036	PTR23:	.BLKW	MSG LIM*2
2632	006434		PTREND:		; END OF MSG. PTR. TABLE
2633					
2634	006434	000002		.BLKW 2	; FILLER FOR OVERFLOW OF RX POINTER TABLE
2635					
2636	006440	000000	RXPTR:	.WORD 0	; RECEIVER MESSAGE POINTER
2637	006442	000000	TXPTR:	.WORD 0	; TRANSMITTER BUFFER POINTER
2638	006444	000000	CMPPTR:	.WORD 0	; COMPARISON BUFFER POINTER
2639	006446	000000	CMPTOT:	.WORD 0	; CMP MSG TOTAL
2640	006450	000000	CTOTCC:	.WORD 0	; COMPARE BUFFER CHAR. COUNT
2641	006452	000000	CCURAD:	.WORD 0	; CURRENT ADDR OF CMP BUFF TO ADD AT
2642					
2643	006454	000000	DVTXA:	.WORD 0	; DEVICE TX ADDR
2644	006456	000000	DVTCC:	.WORD 0	; DEVICE TX CHAR COUNT
2645	006460	000000	DVTCT:	.WORD 0	; DEVICE TX MESSAGE COUNT
2646	006462	000000	TXMTOT:	.WORD 0	; TX MSG TOTAL
2647	006464	000000	TTOTCC:	.WORD 0	; TX BUFFER CHAR. COUNT
2648	006466	000000	TCURAD:	.WORD 0	; CURRENT ADDR. OF TX BUFF TO ADD AT
2649					
2650	006470	000000	DVRXA:	.WORD 0	; DEVICE RX ADDR
2651	006472	000000	DVRCC:	.WORD 0	; DEVICE RX CHAR COUNT
2652	006474	000000	DVRCT:	.WORD 0	; DEVICE RX MESSAGE COUNT
2653	006476	000000	RXMTOT:	.WORD 0	; RX MSG TOTAL
2654					
2655	006500	000000	LN CNT:	.WORD 0	; NUMBER OF OPERATOR AWAKE MSGS
2656	006502	000000	OPVAR:	.WORD 0	; OPTIONAL VARIABLE LOCATION
2657	006504	000000	PSCNT:	.WORD 0	; PASS COUNTER
2658	006506	000000	ERRCNT:	.WORD 0	; ERROR COUNTER
2659	006510	000000	STADD:	.WORD 0	; START ADDR.
2660	006512	000000	ENADD:	.WORD 0	; END ADDR. FOR DUMP
2661	006514	000000	BYTBIT:	.WORD 0	; BYTE BIT FOR DUMP ROUTINE
2662					
2663					; OTHER MESSAGE RELATED STORAGE LOCATIONS
2664					
2665	006516	000000	MSGTYP:	.WORD 0	; TYPE OF DATA 0=0'S, 1=1'S, 2=10'S, 3=01'S ; 4=CCITT, 5=QUICK FOX, 6=ALPHA/NUM, 7=OPER
2666					
2667	006520	000000	CURCC:	.WORD 0	; TX/RX/CMP CHAR COUNT
2668	006522	000000	CPTRR:	.WORD 0	; CURRENT RX POINTER
2669	006524	000000	CPTR:	.WORD 0	; CURRENT POINTER
2670	006526	000000	CURADD:	.WORD 0	; CURRENT TX/RX/CMP START ADDD
2671	006530	000000	TOTCC:	.WORD 0	; TOTAL CHAR COUNT NOT MORE THEN 'BUFLIM'
2672	006532	000000	OFSET:	.WORD 0	; OFFSET COUNT
2673	006534	000000	TEMP:	.WORD 0	; TEMPORARY LOCATIONS (USED A LOT)
2674	006536	000000	TEMP1:	.WORD 0	
2675	006540	000000	TEMP2:	.WORD 0	
2676	006542	000000	TEMP3:	.WORD 0	
2677	006544	000000	TEMP4:	.WORD 0	
2678	006546	000000	TEMP5:	.WORD 0	
2679	006550	000000	CONOTM:	.WORD 0	; CONTROL OUT ERROR MSG. ADDRESS

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 62
MESSAGE BUFFERS AND POINTER TABLES

2680 006552 000000
2681 006554 000
2682 006555 000
2683 006556 000000
2684

CONTIN: .WORD 0 ;WORD FOR CONTROL IN
GOOD: .BYTE 0 ;BYTE TO HOLD EXPECTED MESSAGE DATA BYTE FOR ERR REPORT
BAD: .BYTE 0 ;BYTE TO HOLD RECEIVED MESSAGE DATA BYTE FOR ERR REPORT
DATAWORD: .WORD 0 ;STORAGE LOCATION FOR TRANSMIT DATA

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 63
MESSAGE BUFFERS AND POINTER TABLES

;MORE INDEPENDENT CODE STORAGE LOCATIONS

2685					
2686					
2687	006560	000000	LOGUNT: .WORD	0	;LOC. TO HOLD LOGICAL UNIT NUMBER
2688	006562	000000	PCADD: .WORD	0	;LOC. HOLD PC OF CALLING ROUTINE
2689	006564	000000	DCLFLG: .WORD	0	;CLEANUP & EXIT FLAG -1 = EXIT TEST
2690	006566	000000	RESFLG: .WORD	0	;LOC TO HOLD FLAG (-1) THAT A RESTART WAS GIVEN
2691	006570	000000	MODTYP: .WORD	0	;DCLT MODE OF OPERATION TYPE
2692					(0=REC-ONLY, 1=TX-ONLY, 2=PASSIVE-LOOPBK,
2693					3=ACTIVE-LOOPBK, 4=DOWN L.L., 5=TALK, 6=LISTEN)
2694	006572	000000	MLTYP: .WORD	0	;MAINTENANCE LOOP TYPE (0=NONE, 1=INTERNAL TTL,
2695					2=CABLE, 3=MODEM-ANALOG LOOPBK (LOCAL),
2696					4=MODEM-DIGITAL LOOPBK (REMOTE), 5=MOP)
2697	006574	000000	FHDPLX: .WORD	0	;FULL OR HALF DUPLEX FLAG (1=FULL FROM P-TABLE)
2698	006576	000002	PARAM: .WORD	2	;PROGRAM PARAMETERS
2699					BIT0= STATUS MSGS TO OPR PRINTED (1=YES)
2700					BIT1= DATA CHECKING DONE ON RCD MSGS (1=YES)
2701					BIT2= ECHO (TRANSMIT) RCV'D MSG.(PASSIVE)(1=YES)
2702					BIT3= MODEM STATUS CHECK (1=YES)
2703					BIT4= CRC CALC./CHECK DONE (1=YES)
2704					BIT5= PROTOCOL EMULATION (1=YES)
2705					BIT6= PROTOCOL IS RUNNING
2706					BIT7= FATAL FAULT IN PROTOCOL--ABORTING!!
2707					
2708	006600	000000	RPASS: .WORD	0	;PASS NUMBER FROM RUN COMMAND
2709	006602	000000	FLAG: .WORD	0	;DEVICE FLAG WORD
2710					
2711			;MODE DISPATCH TABLE		
2712	006604	032210	MODE: .WORD	RXONLY	;RX ONLY DISPATCH
2713	006606	032242	.WORD	TXONLY	;TX ONLY DISPATCH
2714	006610	032302	.WORD	PLCK	;PASSIVE LOOP BACK DISP
2715	006612	032336	.WORD	ALCK	;ACTIVE LOOP BACK DISP
2716	006614	033570	.WORD	DLL	;DOWN LINE LOAD DISP
2717	006616	033614	.WORD	TALCK	;TALK MODE DISPATCH
2718	006620	034060	.WORD	LISCK	;LISTEN MODE DISPATCH
2719					
2720					
2721			.SBTTL		CLOCK TABLES, EVENT LOG AND POINTERS
2722	006622	000000	CLKCSR: .WORD	0	;CLOCK CSR ADDRESS
2723	006624	000000	CLKBR: .WORD	0	;CLOCK INTERRUPT LEVEL
2724	006626	000000	CLKVEC: .WORD	0	;CLOCK INTERRUPT VECTOR
2725	006630	000074	CLKHZ: .WORD	60.	;CLOCK'S HERTZ RATE
2726	006632	000000	CLKEN: .WORD	0	;CLOCK'S CSR VALUE TO INTRPT. ENABLE IT
2727					
2728	006634	000000	TIMMIN: .WORD	0	;PLACE TO KEEP TIME-SINCE-START
2729	006636	000000	TIMSEC: .WORD	0	
2730	006640	000000	TIMTCK: .WORD	0	;PLACE TO KEEP # OF TICKS/SEC
2731					
2732	006642	000000	TIMER1: .WORD	0	;EVENT TIMER #1 (TICKS)
2733	006644	000000	TIMER2: .WORD	0	;EVENT TIMER #2 (TICKS)
2734	006646	000000	TIMERS: .WORD	0	;EVENT TIMER #3 (SECONDS)
2735					

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 64
CLOCK TABLES, EVENT LOG AND POINTERS

2736			;EVENT LOG TABLE AND ITS NEXT ENTRY POINTER		
2737	006650	006652	EVTPTN: .WORD	EVTLOG	;POINTER TO NEXT FREE SPACE IN EVENT LOG
2738	006652	000341	EVTLOG: .BLKW	225.	;EVENT LOG BUFFER
2739	007554	000001	EVTEND: .BLKW	1.	;APPROXIMATE END OF EVENT TABLE (ALLOWS CIRCULAR QUEUE)
2740					
2741			.SBTTL MODEM DATA SECTION		
2742					
2743	007556	000000	MODS: .WORD	0	;MODEM STATUS
2744					
2745					

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 65
MODEM DATA SECTION

: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 SRD      ;SECONDARY RECEIVE DATA (CIRCUIT SBB)
MOBITE:
```

:TABLE OF ADDRESSES OF MODEM SIGNAL MESSAGE POSITIONS

```

MOMSGS: .WORD  EVMCTS      ;CLEAR TO SEND (CIRCUIT CB)
        .WORD  EVMDSR      ;DATA SET READY (CIRCUIT CC)
        .WORD  EVMDCD      ;DATA CARRIER DETECT (CIRCUIT CF)
        .WORD  EVMRTS      ;REQUEST TO SEND (CIRCUIT CA)
        .WORD  EVMRI       ;RING INDICATOR (CIRCUIT CE)
        .WORD  EVMSRD      ;SECONDARY RECEIE DATA

```

```

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

```

```

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      EDMOS      ;MODEM STATUS CHANGE
        .WORD      EDDLE      ;DATA COMPARE LENGTH ERROR
        .WORD      EDDDE      ;DATA COMPARE DATA ERROR
        .WORD      EDEOP      ;END OF PASS
        .WORD      EDABO      ;^C ABORT

```

:LOCATIONS USED DURING EVENT REPORTING

```

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

```

:REPORT CODING DISPATCH TABLE

```

RPTDSP: .WORD  RPTTXQ  ;TRANSMIT QUEUED ENTRY DECODING
         .WORD  RPTTXQ  ;TRANSMIT COMPLETE ENTRY DECODING
         .WORD  RPTTXQ  ;RECEIVER QUEUED ENTRY DECODING
         .WORD  RPTTXQ  ;RECEIVER COMPLETE ENTRY DECODING
         .WORD  RPTDER  ;DEVICE ERROR ENTRY DECODING
         .WORD  RPTDVI  ;DEVICE INIT ENTRY DECODING
         .WORD  RPTDCK  ;DATA COMPARISON ENTRY DECODING
         .WORD  RPTMSC  ;REPORT MODEM STATUS CHANGE
         .WORD  RPTDLE  ;DATA COMPARISON LENGTH ERROR

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 66
MODEM DATA SECTION

2802	007676	022566	.WORD	RPTDDE	;DATA COMPAPISON DATA ERROR
2803	007700	022512	.WORD	RPTEOP	;END OF PASS
2804	007702	022512	.WORD	RPTABO	;^C ABORT
2805					
2806	007704	000000	DEV1:	.WORD	0 ;TEMP LOCS TO HOLD DATA FOR EVENT REPORTING
2807	007706	000000	DEV2:	.WORD	0 ; AND SHOW MODE,... SUBROUTINE
2808	007710	000000	DEV3:	.WORD	0
2809	007712	000000	DEV4:	.WORD	0
2810					

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 67
COMMAND LINE ACTION TREE

2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866

007714

007714
007720
007724
007726
007742
007744
007760
007762
007776
010000
010012
010016
010032
010036
010052
010056
010062
010074
010100
010112
010116

010120
010124
010140
010144
010162
010166
010204
010210
010226
010232
010250
010254
010300
010304

.SBTTL COMMAND LINE ACTION TREE

: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
ASCIZ 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 '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,N20\$,<'CLEAR'>	:ELSE, IS FIRST WORD A 'CLEAR'
	CLI	CLIBR,NOTNUF,N100\$	: IF YES DO 'CLR' & GOTO N100\$
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

:SECOND KEYWORD (MODE=) FOR RUN COMMAND

N80\$:	CLI	CLISPA,0,N30\$	:SKIP LEADING SPS, IF NONE-ERR
N81\$:	CLI	CLISTR,NOTNUF,N30\$,<'MODE'>	:IS NEXT WORD 'MODE='
	CLI	<'='>,0,N30\$	: IF NO, IT'S WRONG -ERR -EXIT
	CLI	CLISTR,ATVMOD,N82\$,<'ACTIVE'>	:IS NEXT WORD 'ACTIVE'
	CLI	CLIBR,0,N115\$	: IF YES, DO 'ACTIVE',BR N115\$
N82\$:	CLI	CLISTR,PASMOD,N83\$,<'PASSIVE'>	:IS NEXT WORD 'PASSIVE'
	CLI	CLIBR,0,N115\$	: IF YES, DO 'PASSVE',BR N115\$
N83\$:	CLI	CLISTR,RECMOD,N84\$,<'RECEIVE'>	:IS NEXT WORD 'RECEIVE'
	CLI	CLIBR,0,N115\$	: IF YES, DO 'RECVE',BR N115\$
N84\$:	CLI	CLISTR,LISMOD,N85\$,<'LISTEN'>	:IS NEXT WORD 'LISTEN'
	CLI	CLIBR,0,N115\$	: IF YES, DO 'LISTEN',BR N115\$
N85\$:	CLI	CLISTR,DLLMOD,N86\$,<'DOWNLINELOAD'>	:IS NEXT WORD 'DOW'
	CLI	CLIBR,0,N115\$	: IF YES, DO 'DWNLL',BR N115\$
N86\$:	CLI	<'T'>,0,N30\$	:IS NEXT CHAR A 'T'

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 68
COMMAND LINE ACTION TREE

```

2867 010310      CLI      CLISTR,TRAMOD,N87$,<'RANSMIT'>  : IS REST OF WORD 'RANSMIT'
2868 010326      CLI      CLIBR,0,N115$                : IF YES, DO 'TRANSM',BR N115$
2869 010332      N87$:  CLI      CLISTR,TALMOD,N30$,<'ALK'>  : IS REST OF WORD 'ALK'
2870 010344      CLI      CLIBR,0,N115$                : IF YES, DO 'TALK',BR N115$
2871                                     : IF NO, ERROR - EXIT
2872
2873      ;SECOND KEYWORD (FOR CLEAR OR SHOW)
2874 010350      N100$:  CLI      CLISPA,0,N30$
2875 010354      N102$:  CLI      CLISTR,CSHEXP,N104$,<'EXPECTBUFF'> : SKIP LEADING SPACES, NONE=ERR,
2876 010376      CLI      CLIBR,0,N115$                : IS NEXT WORD 'EXPE...'
2877 010400      N104$:  CLI      CLISTR,CSHTRN,N30$,<'TRANSMITBUFF'> : IF YES, DO CLR-EXP,EXIT
2878 010424      CLI      CLIBR,0,N115$                : IS NEXT WORD 'TRANS...'
2879                                     : IF YES, DO CLR-TRN,EXIT
2880                                     : IF NO - ERROR - EXIT
2881
2882      ;SECOND KEYWORD (FOR SET)
2883 010426      N110$:  CLI      CLISPA,0,N30$
2884 010432      N111$:  CLI      CLISTR,SETEXP,N112$,<'EXPECT'>
2885 010450      CLI      CLIBR,0,N120$
2886 010454      N112$:  CLI      CLISTR,SETTRN,N30$,<'TRANSMIT'>
2887 010474      CLI      CLIBR,0,N120$
2888
2889      ;GET ADDRESSES FOR DUMP COMMAND
2890 010500      N50$:  CLI      CLIALP,0,N51$
2891 010504      N51$:  CLI      CLISPA,0,N52$
2892 010510      N52$:  CLI      CLIOCT,DMP5,N30$
2893 010514      CLI      <'>,NOTNUF,N125$
2894 010520      CLI      CLIOCT,DMPE,N30$
2895 010524      CLI      <'>,NOTNUF,N125$
2896 010530      CLI      <'B>,DMPQ,N30$
2897 010534      CLI      CLIBR,0,N125$
2898
2899      ;QUALIFIERS FOR THE RUN COMMAND
2900 010540      N115$:  CLI      CLIALP,0,N114$
2901 010544      N114$:  CLI      <'>,NOTNUF,N125$
2902 010550      CLI      CLISTR,NO,N116$,<'NO'>
2903 010562      N116$:  CLI      <'C>,0,N117$
2904 010566      CLI      CLISTR,CHECK,N117$,<'HECK'>
2905 010602      CLI      CLIBR,0,N115$
2906
2907      N117$:  CLI      CLISTR,STATUS,N118$,<'STATUS'>
2908 010606      CLI      CLIBR,0,N115$
2909 010624      N118$:  CLI      CLISTR,ECHO,N119$,<'ECHO'>
2910 010630      CLI      CLIBR,0,N115$
2911 010644
2912      N119$:  CLI      <'P>,0,N132$
2913 010650      CLI      CLISTR,PROTO,N130$,<'ROTOCOL'>
2914 010654      CLI      CLIBR,0,N115$
2915 010672      N130$:  CLI      CLISTR,0,N30$,<'ASS'>
2916 010676      CLI      CLIBR,0,N150$
2917 010710
2918      N132$:  CLI      CLISTR,MOSC,N131$,<'MODEM'>
2919 010714      CLI      CLIBR,0,N115$
2920 010730
2921      N131$:  CLI      CLISTR,0,N30$,<'LOOP'>
2922 010734

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 69
COMMAND LINE ACTION TREE

```

2923 010750          CLI      CLIBR,0,N140$
2924
2925          ;GET MESSAGE TYPE FOR SET MESSAGE COMMANDS
2926 010754          N120$: CLI      <'=>,0,N30$
2927
2928          ; LOOK FOR DEFAULT MESSAGE NAME
2929 010760          N60$:  CLI      CLISTR,CMMSG1,N61$,<'ONES'>
2930 010774          CLI      CLIBR,0,N121$
2931 011000          N61$:  CLI      CLISTR,CMMSG0,N62$,<'ZEROES'>
2932 011016          CLI      CLIBR,0,N121$
2933 011022          N62$:  CLI      CLISTR,CMMSG2,N63$,<'1ALT'>
2934 011036          CLI      CLIBR,0,N121$
2935 011042          N63$:  CLI      CLISTR,CMMSG3,N64$,<'0ALT'>
2936 011056          CLI      CLIBR,0,N121$
2937 011062          N64$:  CLI      CLISTR,CMMSG5,N65$,<'ITEP'>
2938 011076          CLI      CLIBR,0,N121$
2939 011102          N65$:  CLI      CLISTR,CMMSG4,N66$,<'CCITT'>
2940 011116          CLI      CLIBR,0,N121$
2941 011122          N66$:  CLI      CLISTR,CMMSG6,N67$,<'ALPHA'>
2942 011136          CLI      CLIBR,0,N121$
2943 011142          N67$:  CLI      CLISTR,SETET,N68$,<'TRANSMIT'>
2944 011162          CLI      CLIBR,0,N125$
2945          ; LOOK FOR QUOTED MESSAGE
2946 011166          N68$:  CLI      <'>,OPRMSG,N30$
2947 011172          N70$:  CLI      <'>,ENDQ0,N71$
2948 011176          CLI      CLIBR,0,N121$
2949 011202          N71$:  CLI      CLISPA,0,N72$
2950 011206          N72$:  CLI      CLIALN,0,N73$
2951 011212          CLI      CLIBR,0,N70$
2952 011216          N73$:  CLI      CLIERR,BADCHR
2953
2954          ;GET QUALIFIERS (SIZE OR COPY) FOR SET MESSAGE COMMANDS
2955 011220          N121$: CLI      CLIALP,0,N123$
2956 011224          N123$: CLI      <'>,NOTNUF,N125$
2957 011230          CLI      CLISTR,SIZE,N122$,<'SIZE'>
2958 011244          CLI      CLIBR,0,N126$
2959 011250          N122$: CLI      CLISTR,QCOPY,N30$,<'COPY'>
2960 011264          CLI      CLIBR,0,N126$
2961
2962          ;NUMER FOR SIZE OR COPY
2963 011270          N126$: CLI      <'=>,0,N30$
2964 011274          CLI      CLIDEC,NUM,N30$
2965 011300          CLI      CLIBR,0,N121$
2966
2967          ;GET MAINTENANCE LOOP TYPE FOR RUN 'LOOP' QUALIFIER
2968 011304          N140$: CLI      <'=>,0,N30$
2969
2970
2971 011310          N141$: CLI      CLISTR,TTLLOP,N142$,<'INTERNALTTL'>
2972 011332          CLI      CLIBR,0,N115$
2973 011336          N142$: CLI      CLISTR,CBLLOP,N143$,<'CABLE'>
2974 011352          CLI      CLIBR,0,N115$
2975 011356          N143$: CLI      CLISTR,LMDLOP,N144$,<'LOCALMODEM'>
2976 011400          CLI      CLIBR,0,N115$
2977 011404          N144$: CLI      CLISTR,RMDLOP,N30$,<'REMOTEMODEM'>
2978 011426          CLI      CLIBR,0,N115$

```

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

;PRINT ERROR IF NONE LEGAL CHAR FOR ''S

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 70
COMMAND LINE ACTION TREE

2979
2980
2981 011432
2982 011436
2983 011442
2984
2985
2986
2987
2988 011446
2989

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

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

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 71
COMMAND LINE ACTION TREE

```

2990
2991
2992      ;DEVICE DEPENDENT STORAGE LOCATIONS FOR
2993      ;  CURRENT DEVICE PARAMETERS
2994
2995
2996 011450 000000      RXCSR: .WORD 0      ;RECEIVE STATUS REGISTER
2997 011452 000000      PARCSR: .WORD 0      ;STATUS REGISTER
2998 011454 000000      RXDBUF: .WORD 0      ;RECEIVE DATA BUFFER
2999 011456 000000      TXCSR: .WORD 0      ;TRANSMIT STATUS REGISTER
3000 011460 000000      TXDBUF: .WORD 0      ;TRANSMIT DATA BUFFER
3001
3002
3003 011462 000000      INVEC: .WORD 0      ;INPUT INTERRUPT VECTOR ADDRESS
3004 011464 000000      OUTVEC: .WORD 0      ;OUTPUT INTERRUPT VECTOR ADDRESS
3005 011466 000000      INTPRI: .WORD 0      ;INTERRUPT PRIORITY
3006
3007
3008 011470 100226      DUPPAR: .WORD 100226 ;THIS WORD IS BROKEN DOWN AS FOLLOWS
3009                                          ;BITS 0-7 =SYNC WORD
3010                                          ;BIT 9 = CRC ENABLE
3011
3012                                          ;BIT 15 = DDCMP MODE
3013
3014
3015 011472 000000      CMODS: .WORD 0      ;CURRENT MODEM
3016 011474 000000      IRXCSR: .WORD 0      ;IMAGE OF RXCSR
3017 011476 000000      IRXDBUF: .WORD 0      ;IMAGE OF RXDBUF
3018 011500 000000      MSGPTR: .WORD 0      ;MSG PTR.FOR HEADER OR CONTROL
3019 011502 000000      MSGCC: .WORD 0      ;MSG COUNTER OR CC
3020 011504 000000      SYNCW: .WORD 0      ;SYNC CHAR COUNT.
3021 011506 000000      SYNCC: .WORD 0      ;SYNC WORD.PLUS TSOM BIT.
3022 011510 000000      RMSGPT: .WORD 0      ;MSG PTR FOR REC
3023 011512 000000      RMSGCC: .WORD 0      ;CHAR COUNTER FOR REC
3024 011514 000000      BCCW: .WORD 0      ;CRC HOLDING LOC.
3025 011516 000000      MGLCNT: .WORD 0      ;COUNT OF GLITCH ERRORS
3026 011520 000000      MHRCNT: .WORD 0      ;COUNT OF HARD ERRORS
3027 011522 000000      RNODE: .WORD 0      ;1=REMOTE NODE ITEP,0=NON ITEP
3028
3029
3030      ;      ERR_TBL

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 72
GLOBAL TEXT SECTION

.SBTTL GLOBAL TEXT SECTION

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

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

DEV TYP <DUP-11>

L\$DVTYP::
.ASCIZ /DUP-11/
.EVEN

.SBTTL PROGRAM IDENTIFICATION
: TEST DESCRIPTION
:

DESCRIPT <DUP-11 DATA COMM LINK TEST >

L\$DESC::
.ASCIZ /DUP-11 DATA COM
.EVEN

.EVEN

3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067

011524				
011524				
011524	052504	026520	030461	
011532	000			
	011534			
011534				
011534				
011534	052504	026520	030461	
011542	042040	052101	020101	
011550	047503	046515	046040	
011556	047111	020113	042524	
011564	052123	000040		

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 73
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

```

3068
3069
3070 011570 041504 052114 000076 CLISPM: .ASCIIZ /DCLT>/
011576 050122 037124 000 CLISRP: .ASCIIZ /RPT>/
011603 045 022516 037501 CLIERM: .ASCIIZ /XNZ?ILL CMD-BAD SYNTAX?/
011633 045 022516 037501 CLINUF: .ASCIIZ /XNZ?INCMPLTE CMD?/
011656 047045 040445 047077 CLINBG: .ASCIIZ /XNZ?NUM TOO BIG?/
011700 047045 040445 041077 CLIBRX: .ASCIIZ /XNZ?BAD RADIX?/
011720 047045 040445 021077 CLIBDL: .ASCIIZ /XNZ?'LOOP' VALID ONLY IN ACTIVE?/
011762 047045 040445 021077 CLINPS: .ASCIIZ /XNZ?'ECHO' VALID ONLY IN PASSIVE?/
012025 045 022516 037501 CLIBCR: .ASCIIZ /XNZ?ILL CHR- 'A-Z,0-9,SP,TAB' ONLY?/
012072 047045 040445 021077 CLISE0: .ASCIIZ /XNZ?'SIZE=0' NOT VALID?/
012123 045 022516 037501 CLIPW: .ASCIIZ /XNZ?TRANSMIT & EXPECT LIST MUST BE IDENTICAL FOR LOOP?/
012213 045 022516 052101 HLP0: .ASCIIZ /XNZ?THIS IS DCLT. TYPE 'H' OR '?' FOR DETAILS/
012271 045 022516 000124 HLPF: .ASCIIZ /XNZT/
012276 041504 052114 041440 HLP1: .ASCIIZ /DCLT CMDS:/
012311 040 046103 040505 HLP2: .ASCIIZ / CLEAR OR SHOW EXPECTLIST OR TRANSMITLIST/<15><12>
012365 040 051120 047111 .ASCIIZ / PRINT/<15><12>
012375 040 054105 052111 .ASCIIZ / EXIT/<15><12>
012404 042040 046525 020120 .ASCIIZ ? DUMP START-END/B?
012426 051440 052105 042440 HLP3: .ASCIIZ ? SET EXPECTMSG OR TRANSMITMSG=TYPE/SIZE=N OR /COPY=N?
012513 040 042523 020124 HLP3A: .ASCIIZ / SET EXPECT=TRANSMIT/
012540 020040 052040 050131 HLP4: .ASCIIZ ? TYPE=ONES,ZEROES,1ALT,0ALT,ITEP,CCITT,ALPHA?
012617 040 020040 020040 HLP4A: .ASCIIZ / OR 'OPR SPCD=A-Z,SP,TAB,0-9 IN QUOTES'/
012675 040 052522 020116 HLP5: .ASCIIZ ? RUN MODE=MTYP/LOOP=LTP/CHECK,PROTOCOL,STATUS,ECHO,MODEM,PASS=N?
012776 020040 046440 054524 HLP6: .ASCIIZ / MTYP=TRAN,REC,ACT,PAS,TAL,LIS,DOWN/<15><12>
013045 040 020040 052114 .ASCIIZ / LTP=INT,CAB,LOC,REM/

013075 045 022516 052101 RHLP0: .ASCIIZ /XNZ?ATYPE 'H' OR '?' FOR HELP!/
013133 104 046103 020124 RHLP1: .ASCIIZ /DCLT REPORT CMDS:/
013155 114 043517 026440 RHLP2: .ASCIIZ /LOG - PRINT DCLT EVENT LOG/
013210 054105 052111 026440 RHLP3: .ASCIIZ /EXIT - EXIT REPORT LEVEL/
013241 110 046105 020120 RHLP4: .ASCIIZ /HELP - PRINT THIS MESSAGE/
013273 103 052517 052116 RHLP5: .ASCIIZ ?COUNTERS/SW - PRINT DDCMP COUNTERS?
013336 044127 051105 020105 RHLP6: .ASCIIZ ?WHERE /SW=FULL, /ERRORS, /OFFSET=NN(0)?
013405 045 022516 047501 RPTIV: .ASCIIZ /XNZ?OFFSET INVALID/
013430 047045 040445 042104 RPTNV: .ASCIIZ /XNZ?DDCMP COUNTERS VALID ONLY WITH PROTOCOL SELECTED./
013516 047045 040445 051515 SHMSG: .ASCIIZ ?XNZ?MSG: TYPE=XNZ/SIZE=XD3?
013552 042532 047522 051505 SHTYP0: .ASCIIZ /ZEROES/
013561 117 042516 000123 SHTYP1: .ASCIIZ /ONES/
013566 040461 052114 000 SHTYP2: .ASCIIZ /1ALT/
013573 060 046101 000124 SHTYP3: .ASCIIZ /0ALT/
013600 041503 052111 000124 SHTYP4: .ASCIIZ /CCITT/
013606 052111 050105 000 SHTYP5: .ASCIIZ /ITEP/
013613 101 050114 040510 SHTYP6: .ASCIIZ /ALPHA/
013621 117 051120 051440 SHTYP7: .ASCIIZ /OPR SPEC/
013632 042522 042503 053111 MO0: .ASCIIZ /RECEIVE/
013642 051124 047101 046523 MO1: .ASCIIZ /TRANSMIT/
013653 120 051501 044523 MO2: .ASCIIZ /PASSIVE/
013663 101 052103 053111 MO3: .ASCIIZ /ACTIVE/
013672 047504 047127 044514 MO4: .ASCIIZ /DOWNLINELOAD/
013707 124 046101 000113 MO5: .ASCIIZ /TALK/
013714 044514 052123 047105 MO6: .ASCIIZ /LISTEN/
013723 000 LPO: .ASCIIZ //
013724 046057 047517 036520 LP00: .ASCIIZ ?/LOOP=?
013733 111 052116 051105 LP1: .ASCIIZ ?INTERNAL?

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 74
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

013744	040503	046102	000105	LP2:	.ASCII	?CABLE?
013752	047514	040503	046514	LP3:	.ASCII	?LOCALMODEM?
013765	122	046505	052117	LP4:	.ASCII	?REMOTEMODEM?
014001	116	117		PNST:	.ASCII	/NO/
014003	123	040524	052524	PST:	.ASCII	/STATUS/
014012	047516			PNCK:	.ASCII	/NO/
014014	044103	041505	000113	PCK:	.ASCII	/CHECK/
014022	047516			PNEC:	.ASCII	/NO/
014024	041505	047510	000	PEC:	.ASCII	/ECHO/
014031	116	117		PNMS:	.ASCII	/NO/
014033	115	042117	046505	PMS:	.ASCII	/MODEM/
014041	116	117		PNPR:	.ASCII	/NO/
014043	120	047522	047524	PPR:	.ASCII	/PROTOCOL/
014054	047045	040445	044514	LISP:	.ASCII	/XN%ALIS>/
014065	124	045514	000076	OPRMM:	.ASCII	/TLK>/
014072	044124	051511	040440	L5060:	.ASCII	/THIS A 50. OR 60. HZ. LSI-11:/
						.EVEN

:
: FORMAT STATEMENTS USED IN PRINT CALLS
:

014130	047045	040445	047504	DLLCM:	.ASCII	/XN%ADOWN LINE LOAD NOT SUPPORTED BY THIS DEVICE/
014210	047045	040445	046103	BDCLK:	.ASCII	/XN%ACLOCK NOT FOUND/
014234	047045	040445	040502	NOCLK:	.ASCII	/XN%ABAD CLOCK - PROGRAM WILL HANG ON "TIMEOUT"!!/
014315	115	054101	020056	TABEX:	.ASCII	/MAX. CHAR. MSG COUNT EXCEEDED -/
014355	102	043125	042506	BUFEX:	.ASCII	/BUFFER FULL -/
014373	045	022516	022524	MSGTRN:	.ASCII	/XN%T% MSG. NOT BUILT !!/
014424	047045	040445	044103	MSGTRU:	.ASCII	/XN%ACHAR. COUNT EXCEEDS BUFF LIMIT - MSG TRUNCATED/
014507	045	022516	032523	SHF0:	.ASCII	?XN%SS%AMODE=X%T%T%T%/PASS=X%25?
014545	045	022516	032523	SHF1:	.ASCII	?XN%SS%SS%SS%A/X%T%A/X%T%A/X%T%A/X%T%A/X%T?
014612	051445	022465	052101	EFM2:	.ASCII	/X%SS%ATOTAL MISMATCHES IN MSG = X%5/
014655	045	022516	031523	PCPM:	.ASCII	/XN%S3%ACALLED FROM PC=X%06/
014707	045	032523	040445	EFM11:	.ASCII	/X%SS%ACOMPARE COUNT=X%5X%S3%ARECEIVE COUNT=X%5/
014764	047515	042504	020115	MSCMS:	.ASCII	/MODEM STATUS CHANGES FOR THIS PASS WERE../
015036	051445	022465	044101	EFM13:	.ASCII	/X%SS%AHARD CHANGES=X%5X%A%S3%AGLITCHES=X%5/

:EVENT DESCRIPTION MESSAGES

015107	124	040522	051516	EDTXQ:	.ASCII	/TRANSMIT MSG QUEUED/
015133	124	040522	051516	EDTXC:	.ASCII	/TRANSMIT MSG COMPLETED/
015162	042522	042503	053111	EDRXQ:	.ASCII	/RECEIVE SPACE QUEUED/
015207	122	041505	044505	EDRXC:	.ASCII	/RECEIVE MSG COMPLETED/
015235	104	053105	041511	EDDER:	.ASCII	/DEVICE ERROR/
015252	040504	040524	041440	EDDCK:	.ASCII	/DATA COMPARISON STARTED/
015302	042504	044526	042503	EDDVI:	.ASCII	/DEVICE INIT AND SETUP/
015330	040504	040524	041440	EDDLE:	.ASCII	/DATA COMPARISON LENGTH ERROR/
015365	104	052101	020101	EDDDE:	.ASCII	/DATA COMPARISON DATA ERROR/
015420	047105	020104	043117	EDEOP:	.ASCII	/END OF PASS/
015434	047515	042504	020115	EDMOS:	.ASCII	/MODEM STATUS CHANGE/

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 75
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

;EVENT REPORTING MESSAGES

```

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

```

```

:EXECUTION STATUS MESSAGES TO BE PRINTED TO KEEP OPERATOR AWAKE
CR:      .ASCIIZ  /%N/      ;CR FOR LINES IN A ROW
STXQ:    .ASCIIZ  /%S3%ATXQ/ ;ABOUT TO TRANSMIT
STXC:    .ASCIIZ  /%S3%ATXC/ ;TX COMPLETED
SRXQ:    .ASCIIZ  /%S3%ARXQ/ ;ABOUT TO RECEIVE
SDVE:    .ASCIIZ  /%S3%AEOR/  ;DEVICE ERROR
SCM:      .ASCIIZ  /%S3%ACMP/  ;ABOUT TO DO DATA CHECKING OF RECVD VS. EXPTD
SDVI:    .ASCIIZ  /%S3%AINI/   ;DEVICE ABOUT TO BE INITIALIZED
SCML:    .ASCIIZ  /%S3%ACML/   ;COMPARE LENGTH ERROR
SCMD:    .ASCIIZ  /%S3%ACMD/   ;COMPARE DATA ERROR
SEOP:    .ASCIIZ  /%S3%AEOP/   ;END OF PASS
        .EVEN

```

```
016671      115  042117  046505  GLMSG:  .ASCIZ  /MODEM STATUS GLITCHED/
016717      115  042117  046505  HRDMSG:  .ASCIZ  /MODEM STATUS HARD ERROR/
```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 76
GLOBAL FORMAT STATEMENTS, MESSAGES, AND ASCII INFO

016747	115	051501	042524	DVEM0:	.ASCII	/MASTER RESET DID NOT WORK/
017000	005015	020040	051040		.ASCIIZ	<15><12>/ RXCSR TXCSR /
017026	047516	041440	042514	DVEM1:	.ASCII	/NO CLEAR TO SEND FROM MODEM /
017062	005015	020040	051040		.ASCIIZ	<15><12>/ RXCSR TXCSR /
017110	044524	042515	047440	DVEM2:	.ASCII	/TIME OUT WAITING FOR RX OR TX TO COMPLETE/
017161	015	020012	020040		.ASCIIZ	<15><12>/ RXCSR TXCSR/
017205	103	041522	044440	DVEM3:	.ASCII	/CRC IN ERROR/
017221	015	020012	051040		.ASCIIZ	<15><12>/ RXDBUF RXCSR/
017246	042522	042503	053111	DVEM4:	.ASCII	/RECEIVER OVERRUN/
017266	005015	020040	054122		.ASCIIZ	<15><12>/ RXDBUF RXCSR/
017313	124	046511	042105	DVEM5:	.ASCII	/TIMED OUT IN START,STACK,ACK SEQ/
017353	015	020012	020040		.ASCIIZ	<15><12>/ RDATA SDATA/
017377	115	042117	046505	DVEM6:	.ASCII	/MODEM DID NOT RETURN MODEM READY/
017437	015	020012	020040		.ASCIIZ	<15><12>/ RXCSR TXCSR/

.EVEN

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 77
GLOBAL ERROR REPORT SECTION

.SBTTL 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.
--

3071	017464		BGNMSG	ERR1		
3072	017464				ERR1::	
3073	017464		PRINTB	#EVTF5A,OFSET,<B,GOOD>,<B,BAD>	;INDIVIDUAL DATA COMPARE ERROR	
3074	017464	005046			CLR	-(SP)
3075	017466	153716	006555		BISB	BAD,(SP)
3076	017472	005046			CLR	-(SP)
3077	017474	153716	006554		BISB	GOOD,(SP)
3078	017500	013746	006532		MOV	OFSET, -(SP)
3079	017504	012746	016302		MOV	#EVTF5A, -(SP)
3080	017510	012746	000004		MOV	#4, -(SP)
3081	017514	010600			MOV	SP,R0
3082	017516	104414			TRAP	C\$PNTB
3083	017520	062706	000012		ADD	#12,SP
3084	017524		ENDMSG			
3085	017524				L10001:	
3086	017524	104423			TRAP	C\$MSG
3087						
3088	017526		BGNMSG	ERR2		
3089	017526				ERR2::	
3090	017526		PRINTB	#EFM2,TEMP4	;TOTAL DATA COMPARE FAILS ERROR	
3091	017526	013746	006544		MOV	TEMP4, -(SP)
3092	017532	012746	014612		MOV	#EFM2, -(SP)
3093	017536	012746	000002		MOV	#2, -(SP)
3094	017542	010600			MOV	SP,R0
3095	017544	104414			TRAP	C\$PNTB
3096	017546	062706	000006		ADD	#6,SP
3097	017552		ENDMSG			
3098	017552				L10002:	
3099	017552	104423			TRAP	C\$MSG
3100						
3101	017554		BGNMSG	ERR10		
3102	017554				ERR10::	
3103	017554		PRINTB	#EFM11,R4,TEMP3	;LENGH COMPARISON ERROR	
3104	017554	013746	006542		MOV	TEMP3, -(SP)
3105	017560	010446			MOV	R4, -(SP)
3106	017562	012746	014707		MOV	#EFM11, -(SP)
3107	017566	012746	000003		MOV	#3, -(SP)
3108	017572	010600			MOV	SP,R0
3109	017574	104414			TRAP	C\$PNTB
3110	017576	062706	000010		ADD	#10,SP
3111	017602		ENDMSG			
3112	017602				L10003:	
3113	017602	104423			TRAP	C\$MSG
3114						

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 78
GLOBAL ERROR REPORT SECTION

3115	017604			BGNMSG	ERR4		
3116	017604					ERR4::	
3117	017604			PRINTB	#EFM13,MHRCNT,MGLCNT	;MODEM STATUS CHANGE	
3118	017604	013746	011516			MOV	MGLCNT,-(SP)
3119	017610	013746	011520			MOV	MHRCNT,-(SP)
3120	017614	012746	015036			MOV	#EFM13,-(SP)
3121	017620	012746	000003			MOV	#3,-(SP)
3122	017624	010600				MOV	SP,R0
3123	017626	104414				TRAP	C\$PNTB
3124	017630	062706	000010			ADD	#10,SP
3125	017634			ENDMSG			
3126	017634					L10004:	
3127	017634	104423				TRAP	C\$MSG
3128							
3129							
3130							
3131				:			
3132				:PRINT THE 2 OCTAL #'S IN TEMP3/4			
3133				:			
3134							
3135	017636			BGNMSG	ERR13	ERR13::	
3136	017636						
3137	017636			PRINTB	#EVT3C,TEMP3,TEMP4		
3138	017636	013746	006544			MOV	TEMP4,-(SP)
3139	017642	013746	006542			MOV	TEMP3,-(SP)
3140	017646	012746	015764			MOV	#EVT3C,-(SP)
3141	017652	012746	000003			MOV	#3,-(SP)
3142	017656	010600				MOV	SP,R0
3143	017660	104414				TRAP	C\$PNTB
3144	017662	062706	000010			ADD	#10,SP
3145	017666			ENDMSG			
3146	017666					L10005:	
3147	017666	104423				TRAP	C\$MSG
3148							
3149				:			
3150				:PRINT THE 2 OCTAL #'S IN TEMP3/4			
3151				: AND THE MSG. WHOSE ADDR. IS IN CONOTM			
3152				:			
3153							
3154	017670			BGNMSG	ERR14	ERR14::	
3155	017670						
3156	017670			PRINTB	#EVT3D,TEMP3,TEMP4,CONOTM		
3157	017670	013746	006550			MOV	CONOTM,-(SP)
3158	017674	013746	006544			MOV	TEMP4,-(SP)
3159	017700	013746	006542			MOV	TEMP3,-(SP)
3160	017704	012746	016001			MOV	#EVT3D,-(SP)
3161	017710	012746	000004			MOV	#4,-(SP)
3162	017714	010600				MOV	SP,R0
3163	017716	104414				TRAP	C\$PNTB
3164	017720	062706	000012			ADD	#12,SP
3165	017724			ENDMSG			
3166	017724					L10006:	
3167	017724	104423				TRAP	C\$MSG
3168							
3169	017726			EXIT	MSG		
3170	017726	000167				.WORD	JSJMP

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 79
GLOBAL ERROR REPORT SECTION

3171 017730 177772
3172
3173

.WORD L10006-2-.

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 80
GLOBAL SUBROUTINES SECTION

.SBTTL GLOBAL SUBROUTINES SECTION

++
: 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

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
3207
3208
3209 017732
3210 017732 012122
3211 017734 012112
3212 017736 006312
3213 017740 006312
3214 017742 006312
3215 017744 006312
3216 017746 006322
3217 017750 012122
3218 017752 012122
3219 017754 000207
3220

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 81
CLOCK SETUP SUBROUTINE

3221		.SBTTL	CLOCK INTERRUPT SERVICE ROUTINE
3222		:++	
3223		:FUNCTIONAL DESCRIPTION:	
3224		:	THIS IS THE CLOCK INTERRUPT SERVICE ROUTINE WHICH TAKES CARE OF
3225		:	KEEPING THE 'TIME-SINCE-START' AND COUNTING DOWN ANY OF THE
3226		:	'EVENT' TIMERS. THE TIMERS ARE USED TO TIME COMPLETION OF DEVICE
3227		:	REQUESTS. THE 'TIME-SINCE-START' IS USED TO BE LOGGED WITH EACH ENTRY
3228		:	INTO THE EVENT LOG.
3229		:	
3230		:	
3231		:IMPLICIT INPUTS:	
3232		:	TIMTCK: THE CURRENT NO. OF TICKS LEFT TO BE COUNTED UNTIL A SECOND
3233		:	HAS BEEN COUNTED OFF
3234		:	CLKHZ: THE NO. OF TICKS IN A SECOND, DETERMINED BY THE SYS. LINE FREQ.
3235		:	TIMMIN & TIMSEC: CURRENT VALUE OF 'TIME-SINCE-START'
3236		:	IN MINUTES & SECONDS
3237		:	TIMER 1,2, & S: CURRENT VALUES OF THE 'EVENT TIMERS'
3238		:	
3239		:IMPLICIT OUTPUTS:	
3240		:	NEW VALUE OF EVENT TIMER '1' DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
3241		:	NEW VALUE OF EVENT TIMER '2' DECREMENTED BY 1 TICK IF IT WAS NON-ZERO
3242		:	NEW VALUE OF EVENT TIMER 'S' DECREMENTED BY 1 SECOND IF IT WAS NON-ZERO
3243		:	
3244		:FUNCTIONAL SIDE EFFECTS:	
3245		:	THE CLOCK IS DISABLED UPON ENTRY AND REENABLED WHEN LEAVING
3246		:	
3247		:CALLING SEQUENCE:	
3248		:	THIS ROUTINE IS CALLED WHEN THE CLOCK INTERRUPTS THRU 'CLKVEC'.
3249		:	THE ADDRESS OF THIS ROUTINE WAS LOADED INTO THE CLOCK'S INTERRUPT
3250		:	VECTOR WITH A SUPERVISOR 'SETVEC' CALL.
3251		:--	
3252			
3253	017756	BGNSRV CLKINT	
3254	017756		CLKINT::
3255			
3256	017756	005077 166640	CLR @CLKCSR ;DISABLE THE CLOCK FORM INTERRUPTING
3257	017762	005337 006640	DEC TIMTCK ;DECREMENT THE # OF TICKS/SEC.
3258	017766	001015	BNE 1\$;GO CHECK TIMERS (1&2-TICKS, 3-SECONDS)
3259	017770	013737 006630 006640	MOV CLKHZ,TIMTCK ;RESET THE # OF TICKS/SEC.
3260	017776	005237 006636	INC TIMSEC ;INC # OF SECS-SINCE-START
3261	020002	022737 000074 006636	CMP #60.,TIMSEC ;SEE IF WE'VE COUNTED 60 SECS. YET
3262	020010	001004	BNE 1\$;IF NOT, GO CHECK TIMERS
3263	020012	005237 006634	INC TIMMIN ; ELSE INC MINUTES-SINCE-START
3264	020016	005037 006636	CLR TIMSEC ; AND RESTART SECOND COUNTER
3265			
3266	020022	005737 006642	1\$: TST TIMER1 ;SEE IF TIMER #1, TIMING ANYTHING
3267	020026	001402	BEQ 2\$; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
3268	020030	005337 006642	DEC TIMER1 ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
3269	020034	005737 006644	2\$: TST TIMER2 ;SEE IF TIMER #2, TIMING ANYTHING
3270	020040	001402	BEQ 3\$; IF=0, NOTHING BEING TIMED CHECK NEXT TIMER
3271	020042	005337 006644	DEC TIMER2 ; ELSE DECREMENT THE TIMER VALUE (BY 1 TICK)
3272	020046	005737 006646	3\$: TST TIMERS ;SEE IF TIMER #3, TIMING ANYTHING
3273	020052	001406	BEQ 4\$; IF=0, NOTHING BEING TIMED, LEAVE
3274	020054	023737 006630 006640	CMP CLKHZ,TIMTCK ;SEE IF A SECOND HAS BEEN COUNTED OFF
3275	020062	001002	BNE 4\$; BR IF NO
3276	020064	005337 006646	DEC TIMERS ; ELSE DECREMENT THE TIMER VALUE (BY 1 SEC.)

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 82
CLOCK INTERRUPT SERVICE ROUTINE

3277 020070 013777 006632 166524 48: MOV CLKEN,@CLKCSR ;REENABLE THE CLOCK TO INTERRUPT
3278 020076 ENDSRV
3279 020076 L10007:
3280 020076 000002 RTI

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 83
EVENT LOG SUBROUTINES

.SBTTL EVENT LOG SUBROUTINES

```

++
FUNCTIONAL DESCRIPTION:
THIS SUBROUTINE HAS A DIFFERENT ENTRY POINT
FOR EACH EVENT TO BE LOGGED AND ALWAYS PRINTS
THE SHORT 'OPERATOR AWAKE' MESSAGE TO CONSOLE THEN LOGS THE
EVENT TYPE, TIME, AND THE OTHER 3 WORDS OF INFO PASSED TO THE
SUBROUTINE AT CALLING TIME

INPUTS:
TIMMIN & TIMSEC:      CURRENT VALUE OF 'TIME-SINCE-START'
TEMP2:  WORD #1 OF EVENT LOG INFORMATION (FOR MOST EVENT TYPES)
TEMP3:  WORD #2 OF EVENT LOG INFORMATION
TEMP4:  WORD #3 OF EVENT LOG INFORMATION
MODS:   CURRENT VALUE OF THE MODEM SIGNALS AVAILABLE FROM THE DEVICE

OUTPUTS:
'OPERATOR AWAKE' MESSAGE SENT TO THE CONSOLE
NEW EVENT LOGGED IN 'EVTLOG' (EVENT LOG)
UPDATED 'EVTPTN' (EVENT LOG ENTRY POINTER)

SUBORDINATE ROUTINES USED:
'DVMODS' THE DEVICE SUBROUTINE THAT RETURNS MODEM STATUS IN 'MODS'
(FOR SOME EVENT TYPES)

FUNCTIONAL SIDE EFFECTS:
TEMP:  USED TO STORE ADDRESS OF 'OPERATOR AWAKE' MESSAGE
TEMP1: USED TO SETUP THE VALUE OF THE 'EVENT TYPE' BYTE FOR LOGGING

CALLING SEQUENCE:
      JSR      PC,LOGTXQ      ;CALL THE LOG EVENT SUBROUTINE WITH TEMP,TEMP1,
      ..      ..      ; TEMP2, TEMP3, AND TEMP4 SETUP
      JSR      PC,LOGCMP
--

LOGTXQ:
      MOV      #STXQ,TEMP1    ;SET UP MSG. TO PRINT
      MOV      #TXQ,TEMP      ;SET UP EVENT TYPE
      BR       LOGS1          ;GO LOG EVENT AND TIME

LOGTXC:
      MOV      #STXC,TEMP1    ;SET UP MSG. TO PRINT
      MOV      #TXC,TEMP      ;SET UP EVENT TYPE
      BR       LOGS1          ;GO LOG EVENT AND TIME

LOGRXQ:
      MOV      #SRXQ,TEMP1    ;SET UP MSG. TO PRINT
      MOV      #RXQ,TEMP      ;SET UP EVENT TYPE
      BR       LOGS1          ;GO LOG EVENT AND TIME

LOGRXC:
      MOV      #RXC,TEMP      ;SET UP EVENT TYPE
      BR       LOGS1          ;GO LOG EVENT AND TIME

LGDVE:

```

```

3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318 020100
3319 020100 012737 016537 006536
3320 020106 012737 000000 006534
3321 020114 000517
3322
3323 020116
3324 020116 012737 016550 006536
3325 020124 012737 000002 006534
3326 020132 000510
3327
3328 020134
3329 020134 012737 016561 006536
3330 020142 012737 000004 006534
3331 020150 000501
3332
3333 020152
3334 020152 012737 000006 006534
3335 020160 000475
3336 020162

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 84
EVENT LOG SUBROUTINES

3337	020162	012737	016572	006536	MOV	#SDVE,TEMP1	:SET UP MSG. TO PRINT
3338	020170	012737	000010	006534	MOV	#DER,TEMP	:SET UP EVENT TYPE
3339	020176	000503			BR	LOGS3	:GO LOG EVENT AND TIME
3340							
3341	020200				LOGDVI:		
3342	020200	012737	016614	006536	MOV	#SDVI,TEMP1	:SET UP MSG. TO PRINT
3343	020206	012737	000012	006534	MOV	#DVI,TEMP	:SET UP EVENT TYPE
3344	020214	113737	006570	006540	MOVB	MODTYP,TEMP2	
3345	020222	113737	006572	006541	MOVB	MLTYP,TEMP2+1	
3346	020230	013737	006600	006542	MOV	RPASS,TEMP3	
3347	020236	013737	006576	006544	MOV	PARAM,TEMP4	:SET UP EVNT ENTRIES
3348	020244	000460			BR	LOGS3	:GO LOG EVENT AND TIME
3349							
3350	020246				LOGCMP:		
3351	020246	012737	016603	006536	MOV	#SCM,TEMP1	:SET UP MSG. TO PRINT
3352	020254	012737	000014	006534	MOV	#DCK,TEMP	:SET UP EVENT TYPE
3353	020262	000451			BR	LOGS3	
3354	020264				LOGCML:		
3355	020264	012737	016625	006536	MOV	#SCML,TEMP1	
3356	020272	012737	000020	006534	MOV	#DLE,TEMP	:SET UP MSG. AND TYPE
3357	020300	000442			BR	LOGS3	:GO LOG EVENT AND TIME
3358	020302				LOGCMD:		
3359	020302	012737	016636	006536	MOV	#SCMD,TEMP1	
3360	020310	012737	000022	006534	MOV	#DDE,TEMP	
3361	020316	000433			BR	LOGS3	:GO LOG MSG TYPE AND TIME
3362	020320				LOGEOP:		
3363	020320	012737	016647	006536	MOV	#SEOP,TEMP1	
3364	020326	012737	000024	006534	MOV	#EOP,TEMP	
3365	020334	000424			BR	LOGS3	:GO LOG MSG TYPE AND TIME
3366							
3367							
3368	020336				LOGMSC:		
3369	020336	012737	016660	006536	MOV	#SMSC,TEMP1	
3370	020344	012737	000016	006534	MOV	#MSC,TEMP	
3371	020352	000415			BR	LOGS3	
3372							
3373							
3374	020354	013746	006506		LOGS1:	MOV	ERRCNT, -(SP)
3375	020360	004737	035152			JSR	PC,DVMODS
3376	020364	012604				MOV	(SP)+,R4
3377	020366	020437	006506			CMP	R4,ERRCNT
3378	020372	001402				BEQ	1\$
3379	020374	000137	020610			JMP	LOGEX
3380							
3381	020400	013737	007556	006544	1\$:	MOV	MODS,TEMP4
3382							
3383	020406				LOGS3:		
3384	020406	022737	000006	006534	CMP	#RXC,TEMP	
3385	020414	001434			BEQ	LOGS5	:IF RXC DONT PRINT
3386	020416	032737	000001	006576	BIT	#STATB,PARAM	
3387	020424	001430			BEQ	LOGS5	:IF NO STATUS SELECTED
3388							:GO TO 5
3389							
3390	020426	022737	000010	006500	CMP	#10,LNCNT	:HAVE WE DONE 10?
3391	020434	001012			BNE	LOGS4	:IF NOT GO TO 4
3392	020436	005037	006500		CLR	LNCNT	:ESLE CLEAR IT

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 85
EVENT LOG SUBROUTINES

```

3393
3394 020442          PRINTF  #CR          :ELSE PRINT CR
3395 020442 012746 016534
3396 020446 012746 000001
3397 020452 010600
3398 020454 104417
3399 020456 062706 000004
3400 020462
3401 020462 005237 006500
3402 020466
3403 020466 013746 006536
3404 020472 012746 000001
3405 020476 010600
3406 020500 104417
3407 020502 062706 000004
3408 020506 010346
3409 020510 013703 006650
3410 020514 113723 006534
3411 020520 013737 006630 006534
3412 020526 163737 006640 006534
3413 020534 113723 006534
3414 020540 113723 006636
3415 020544 113723 006634
3416 020550 013723 006540
3417 020554 013723 006542
3418 020560 013723 006544
3419 020564 020327 007554
3420 020570 103404
3421
3422 020572 012713 177777
3423 020576 012703 006652
3424 020602 010337 006650
3425 020606 012603
3426 020610 000207
3427

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

LOGS5: MOV R3,-(SP)        :SAVE R3 ON THE STACK
        MOV EVTPTR,R3
        MOVB TEMP,(R3)+    :LOG EVENT
        MOV CLKH2,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
        CMP R3,#EVTEND
        BLO LOGS2          :IF EVENT LOG FULL GO
                           :CONTINUE;ELSE GO TO 2
                           :LOG A TABLE END
                           :PUT R3 TO START OF TABLE
LOGS2: MOV R3,EVTPTR      :RESTORE POINTER
LOGEX: MOV (SP)+,R3        :RESTORE R3
        RTS PC

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 86
REPORT EVENT LOG

.SBTTL REPORT EVENT LOG

```

::RPT> LOG
::      HELP
::      EXIT
::      COUNTER/FULL,ERROR,OFFSET=NN(0)

```

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

```

:PRINT HELP MESSAGE
PRINTF #RHLPO ;BASIC HELP MESSAGE

```

```

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

```

GETRCL: CLRB PSGBD ;INIT GOOD/BAD FLAG -1=BAD INPUT
CLRB PSNNUF ;INIT MORE COMMAND LINE INPUT NEEDED

```

:PRINT PROMPT 'RPT>'
GMANID CLISRP,CMDBUF,A,-1,1,72.,NO

```

```

TRAP C$GMAN
BR 10000$
.WORD CMDBUF
.WORD T$CODE
.WORD CLISRP
.WORD -1
.WORD T$LOLIM
.WORD T$HILIM

```

10000\$:

```

MOV #CMDBUF,PSBUFA ;INPUT BUFFER
MOV #CLIRT,PSFREE ;REPORT CLI TREE
MOV #CLIRAC,PSACT ;ACTION ROUTINES
CLR QUALFG
JSR PC,PSTRV ;GO PARSE COMMAND LINE
TSTB PSGBD ;COMMAND OK ?
BEQ 1$ ;YES,BRANCH
PRINTF #CLIERM ;PRINT INVALID INPUT MESSAGE

```

```

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

```

JMP GETRCL ;TRY AGAIN

```

1$: TSTB PSNNUF ;MORE COMMAND NEEDED ?
BEQ 10$ ;NO,BRANCH
PRINTF #CLINUF ;INCOMPLETE MESSAGE

```

```

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

```

```

3428
3429
3430
3431
3432
3433
3434
3435 020612 010246
3436 020614 010346
3437 020616 010446
3438
3439
3440 020620
3441 020620 012746 013075
3442 020624 012746 000001
3443 020630 010600
3444 020632 104417
3445 020634 062706 000004
3446
3447 020640 105037 003147
3448 020644 105037 003146
3449
3450
3451 020650
3452 020650 104443
3453 020652 000406
3454 020654 002666
3455 020656 000142
3456 020660 011576
3457 020662 177777
3458 020664 000001
3459 020666 000110
3460 020670
3461 020670 012737 002666 003132
3462 020676 012737 021032 003134
3463 020704 012737 021234 003136
3464 020712 005037 003012
3465 020716 004737 024246
3466 020722 105737 003147
3467 020726 001412
3468 020730
3469 020730 012746 011603
3470 020734 012746 000001
3471 020740 010600
3472 020742 104417
3473 020744 062706 000004
3474 020750 000137 020640
3475
3476 020754 105737 003146
3477 020760 001412
3478 020762
3479 020762 012746 011633
3480 020766 012746 000001
3481 020772 010600
3482 020774 104417
3483 020776 062706 000004

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 87
REPORT EVENT LOG

3484	021002	000137	020640		JMP	GETRCL	:TRY AGAIN
3485							
3486	021006	023727	003010	000002	10\$:	CMP	KEYWD1,#RPEXT
3487	021014	001402				BEQ	20\$
3488	021016	000137	020640			JMP	GETRCL
3489	021022	012604			20\$:	MOV	(SP)+,R4
3490	021024	012603				MOV	(SP)+,R3
3491	021026	012602				MOV	(SP)+,R2
3492	021030	000207				RTS	PC

:EXIT COMMAND ?
:YES,BRANCH
:GET ANOTHER COMMAND
:RESTORE R4
:RESTORE R3
:RESTORE R2
:RETURN

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 88
COMMAND LINE PARSING TREE FOR REPORT

```

3493 .SBTTL COMMAND LINE PARSING TREE FOR REPORT
3494 CLIRT: CLI CLISPA,0,R10$ ;SKIP SPACES IN COMMAND LINE
3495 R10$: CLI <'?'>,RPHLP,R11$ ;IF INPUT = ? THEN PRINT HELP MESSAGE
3496 CLI CLIEXI,0 ;AND EXIT PARSE
3497 R11$: CLI CLISTR,RPHLP,R12$,<'HELP'> ;IF INPUT = 'HELP' THEN PRINT HELP
3498 CLI CLIEXI,0 ;MESSAGE AND EXIT PARSE
3499 R12$: CLI CLISTR,RPEXT,R13$,<'EXIT'> ;IF INPUT = 'EXIT' THEN SET KEYWORD =
3500 CLI CLIEXI,0 ;RPEXT AND EXIT PARSE
3501 R13$: CLI CLISTR,RPLOG,R14$,<'LOG'> ;IF INPUT = 'LOG' THEN GO PRINT EVENT
3502 CLI CLIEXI,0 ;LOG AND EXIT PARSE
3503 R14$: CLI CLISTR,RNOTNF,R30$,<'COUNTERS'>;IF INPUT = 'COUNTERS'
3504 CLI CLIBR,0,R20$ ;THEN GET SWITCH
3505 R20$: CLI <'/'>,RNOTNF,R30$
3506 CLI CLISTR,RPERR,R21$,<'ERROR'> ; REPORT ERROR COUNTERS
3507 CLI CLIEXI,0
3508 R21$: CLI CLISTR,RPFUL,R22$,<'FULL'> ; REPORT ALL STATUS
3509 CLI CLIEXI,0
3510 R22$: CLI CLISTR,RNOTNF,R30$,<'OFFSET'> ; REPORT ONE LOCATION
3511 CLI <'='>,0,R30$
3512 CLI CLIOCT,RPSWO,R30$
3513 CLI CLIEXI,0
3514 R30$: CLI CLIERR,0
3515 R125$: CLI CLIEXI,0

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 89
CLI ACTION DISPATCHER AND ROUTINES

```

3516 .SBTTL CLI ACTION DISPATCHER AND ROUTINES
3517 CLIRAC: ASL R2 ;SET UP INDEX
3518 MOV 10$(R2),R2 ;
3519 ADD #10$,R2 ;
3520 JSR PC,(R2) ;GO DO ACTION
3521 RTS PC ;RETURN
3522 10$: .WORD ACTRNL-10$ ;NULL
3523 .WORD ACTRHL-10$ ;HELP ROUTINE
3524 .WORD ACTREX-10$ ;EXIT ROUTINE
3525 .WORD ACTRLG-10$ ;REPORT EVENT LOG ROUTINE
3526 .WORD ACTERR-10$ ;REPORT ONLY ERROR COUNTERS
3527 .WORD ACTFUL-10$ ;REPORT ALL COUNTERS
3528 .WORD ACTRNF-10$ ;MORE COMMAND NEEDED
3529 .WORD ACTRSO-10$ ;VALIDATE OFFSET
3530
3531 :::::ACTION ROUTINES FOR REPORT::::::
3532 021272 113737 177777 003146 ACTRNF: MOVB -1,PSNNUF ;SET 'MORE COMMAND NEEDED' FLAG
3533 021300 000207 ACTRNL: RTS PC ;NULL
3534
3535 ;PRINT HELP MESSAGE
3536 021302 012702 003036 ACTRHL: MOV #RHLPTB,R2 ;INDEX FOR HELP MESSAGES
3537 021306 1$: PRINTF #HLPF,(R2)+ ;PRINT IT
3538 021306 012246
3539 021310 012746 012271
3540 021314 012746 000002
3541 021320 010600
3542 021322 104417
3543 021324 062706 000006
3544 021330 020227 003052
3545 021334 001364
3546 021336 012737 000001 003010
3547 021344 000207
3548
3549
3550
3551 021346 012737 000002 003010 ACTREX: MOV #RPEXT,KEYWD1 ;SET KEYWORD AND RETURN
3552 021354 000207
3553
3554
3555 021356 004737 022060 ACTRLG: JSR PC,REPLOG ;GO PRINT EVENT LOG
3556 021362 012737 000003 003010 MOV #RPLOG,KEYWD1 ;SET KEYWORD
3557 021370 000207 RTS PC ;RETURN
3558
3559
3560 021372 012737 000000 037254 ACTFUL: MOV #0,FIR ;STARTING INDEX
3561 021400 012737 000036 037252 MOV #36,LAST ;LAST INDEX
3562 021406 004737 021542 JSR PC,STAPRI ;GO PRINT IT
3563 021412 000207 RETURN
3564
3565
3566 021414 012737 000014 037254 ACTERR: ;PRINT ONLY DDCMP ERROR COUNTERS
3567 021422 012737 000036 037252 MOV #14,FIR ;FIRST ERROR
3568 021430 004737 021542 MOV #36,LAST ;LAST ERROR
3569 021434 000207 JSR PC,STAPRI ;GO PRINT IT
3570 RETURN
3571

```

```

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

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 90
CLI ACTION DISPATCHER AND ROUTINES

```

3572
3573 021436 105037 003146 003142 ACTRSO: ::VERIFY OFFSET VALUE
3574 021442 032737 000001 003142 CLR B PSNUF :CLEAR 'NOT ENOUGH FLAG'
3575 021450 001020 000001 003142 BIT #BIT0,PSNUM :IS IT ODD ?
3576 021452 005737 003142 BNE 20$ :YES,BRANCH
3577 021456 100415 000001 003142 TST PSNUM :NEGATIVE # ?
3578 021460 023727 003142 000036 BMI 20$ :YES,BRANCH
3579 021466 003011 003142 000036 CMP PSNUM,#36 :INDEX LARGER THEN 36 ?
3580 021470 013737 003142 037254 BGT 20$ :YES,BRANCH
3581 021476 013737 003142 037252 MOV PSNUM,FIR :STARTING INDEX
3582 021504 004737 021542 037252 MOV PSNUM,LAST :LAST LOCATION
3583 021510 000413 000001 003142 JSR PC,STAPRI :PRINT SINGLE LOCATION
3584 021512 012746 013405 000001 003142 BR 30$ :EXIT
3585 021512 012746 000001 003142 20$: PRINTS #RPTIV :INVALID
3586 021516 012746 000001 003142 MOV #RPTIV,-(SP)
3587 021522 010600 000001 003142 MOV #1,-(SP)
3588 021524 104416 000001 003142 MOV SP,R0
3589 021526 062706 000004 003147 TRAP C$PNTS
3590 021532 112737 177777 003147 ADD #4,SP
3591 021540 000207 000001 003147 30$: MOVB #-1,PSGDBD :SET BAD DATA FLAG
3592 021540 000207 000001 003147 RETURN :OFFSET OK - EXIT
3593
3594 :: PRINT ROUTINES
3595 021542 010146 000040 006576 STAPRI: MOV R1,-(SP) :SAVE R1
3596 021544 032737 000040 006576 BIT #PROTOB,PARAM :'/PROTOCOL'?
3597 021552 001011 000040 006576 BNE 5$ :YES,BRANCH
3598 021554 012746 013430 000001 006576 PRINTF #RPTNV :'COUNTERS VALID ONLY WITH PROTOCOL SELECTED'
3599 021554 012746 013430 000001 006576 MOV #RPTNV,-(SP)
3600 021560 012746 000001 006576 MOV #1,-(SP)
3601 021564 010600 000001 006576 MOV SP,R0
3602 021566 104417 000001 006576 TRAP C$PNTF
3603 021570 062706 000004 006576 ADD #4,SP
3604 021574 000420 000001 006576 BR 20$ :EXIT
3605 021576 013701 037254 000001 006576 5$: MOV FIR,R1 :FIRST INDEX
3606 021602 016137 037152 037256 10$: MOV STALST(R1),MES :MESSAGE ADDRESS
3607 021610 016137 037052 037260 MOV PRSTAT(R1),MESDATA :MESSAGE DATA
3608 021616 004771 037212 037260 JSR PC,STAINDR(R1) :JUMP TO PROPER PRINT ROUTINE
3609 021622 062701 000002 037260 ADD #2,R1 :BUMP INDEX
3610 021626 020137 037252 037260 CMP R1,LAST :ALL MESSAGES PRINTED
3611 021632 003001 000002 037260 BGT 20$ :YES,BRANCH
3612 021634 000762 000002 037260 BR 10$ :PRINT NEXT MESSAGE
3613 021636 012601 000002 037260 20$: MOV (SP)+,R1 :RESTORE R1
3614 021640 000207 000002 037260 RETURN :EXIT
3615
3616 :: PRINT WORD LOCATION
3617 PRIW: PRINTS MES,MESDATA :PRINT WORD LOCATION
3618 021642 013746 037260 000002 037260 MOV MESDATA,-(SP)
3619 021642 013746 037256 000002 037260 MOV MES,-(SP)
3620 021646 013746 037256 000002 037260 MOV #2,-(SP)
3621 021652 012746 000002 037260 MOV SP,R0
3622 021656 010600 000002 037260 TRAP C$PNTS
3623 021660 104416 000002 037260 ADD #6,SP
3624 021662 062706 000006 037260
3625 021666 000207 000006 037260 RETURN
3626
3627

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 91
CLI ACTION DISPATCHER AND ROUTINES

:: PRINT TWO BYTES OF DATA
PRIBB: PRINTS MES,<B,MESDATA>,<B,MESDATA+1>

3628
3629 021670 005046
3630 021670 153716 037261
3631 021672 005046
3632 021676 153716 037260
3633 021700 013746 037256
3634 021704 012746 000003
3635 021710 010600
3636 021714 104416
3637 021716 062706 000010
3638 021720 000207
3639 021724

RETURN

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

:: PRINT SPECIAL BYTE MASK

3640
3641
3642 021726 005037 006536
3643 021732 005037 006540
3644 021736 005037 006542
3645 021742 132737 000001 037261
3646 021750 001402
3647 021752 005237 006536
3648 021756 132737 000002 037261 10\$:
3649 021764 001402
3650 021766 005237 006540
3651 021772 132737 000004 037261 20\$:
3652 022000 001402
3653 022002 005237 006542
3654 022006 30\$:
3655 022006 005046
3656 022010 153716 006542
3657 022014 005046
3658 022016 153716 006540
3659 022022 005046
3660 022024 153716 006536
3661 022030 005046
3662 022032 153716 037260
3663 022036 013746 037256
3664 022042 012746 000005
3665 022046 010600
3666 022050 104416
3667 022052 062706 000014
3668 022056 000207
3669

RETURN

CLR -(SP)
BISB TEMP3,(SP)
CLR -(SP)
BISB TEMP2,(SP)
CLR -(SP)
BISB TEMP1,(SP)
CLR -(SP)
BISB MESDATA,(SP)
MOV MES,-(SP)
MOV #5,-(SP)
MOV SP,R0
TRAP C\$PNTS
ADD #14,SP

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 92
DUMP EVENT LOG

.SBTTL DUMP EVENT LOG

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

3670									
3671									
3672									
3673	022060	010246							
3674	022062	010346							
3675	022064	010446							
3676									
3677									
3678									
3679									
3680	022066	013702	006650		MOV	EVTPT,R2		;MAKE R2 A POINTER TO EVENT TABLE	
3681	022072	023727	006652	177777	CMP	EVTLOG,#-1		;SEE IF EVENT TABLE IS EMPTY	
3682	022100	001034			BNE	RPT0		;BR IF NO	
3683	022102				PRINTS	#NULEVT		;IF EMPTY TELL OPERATOR.	
3684	022102	012746	015515						MOV #NULEVT,-(SP)
3685	022106	012746	000001						MOV #1,-(SP)
3686	022112	010600							MOV SP,R0
3687	022114	104416							TRAP C\$PNTS
3688	022116	062706	000004						ADD #4,SP
3689	022122	000137	023006		JMP	ENDEVT		;AND END	
3690									
3691	022126	162702	000012		RPT: SUB	#12,R2		;NOW POINT BACK TO TOP OF ENTRY U	
3692								;JUST PRINTED	
3693									
3694	022132	020227	006652		CMP	R2,#EVTLOG		;POINTING TO TOP OF EVNT LOG QUEUE?	
3695	022136	001010			BNE	RPT1		; BR IF NO	
3696	022140	012702	007554		MOV	#EVTEND,R2		;SET R2 TO POINT TO BOTTOM OF LOG	
3697	022144	026227	177776	177777	CMP	-2(R2),#-1			
3698	022152	001007			BNE	RPT0		;IF END OF LOG IS NOT EMPTY	
3699	022154	000137	023006		JMP	ENDEVT		;CONTINUE...ELSE EXIT	
3700									
3701	022160	020237	006650		RPT1: CMP	R2,EVTPT,R		;ARE WE BACK TO POINTER?	
3702	022164	001002			BNE	RPT0		;IF NOT CONTINUE	
3703	022166	000137	023006		JMP	ENDEVT		;IF SO EXIT....	
3704									
3705	022172	162702	000012		RPT0: SUB	#12,R2		;POINT R2 TO START OF ENTRY	
3706	022176				RPTAA: PRINTS	#EVTFO		;PRINT EVENT ENTRY HEADER	
3707	022176	012746	015555						MOV #EVTFO,-(SP)
3708	022202	012746	000001						MOV #1,-(SP)
3709	022206	010600							MOV SP,R0
3710	022210	104416							TRAP C\$PNTS
3711	022212	062706	000004						ADD #4,SP
3712	022216	112203			MOVB	(R2)+,R3		;PUT EVENT TYPE INTO R3	
3713	022220	112237	007644		MOVB	(R2)+,EVTICK			
3714	022224	112237	007640		MOVB	(R2)+,EVTSEC		;PUT EVENT TIME (TICKS,SECS,MINS IN TEMP LOC.S)	
3715	022230	112237	007642		MOVB	(R2)+,EVTMIN			
3716	022234				PRINTS	#EVTF1,EVTMIN,EVTSEC,EVTICK,EVTLS(R3)		;PRINT EVENT TIME AND DESCRIPT.	
3717	022234	016346	007610						MOV EVTLST(R3),-(SP)
3718	022240	013746	007644						MOV EVTTCK,-(SP)
3719	022244	013746	007640						MOV EVTSEC,-(SP)
3720	022250	013746	007642						MOV EVTMIN,-(SP)
3721	022254	012746	015651						MOV #EVTF1,-(SP)
3722	022260	012746	000005						MOV #5,-(SP)
3723	022264	010600							MOV SP,R0
3724	022266	104416							TRAP C\$PNTS
3725	022270	062706	000014						ADD #14,SP

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 93
DUMP EVENT LOG

```

3726 022274 000173 007654      JMP      @RPTDSP(R3)      ;DISPATCH TO DECODING SECTION FOR SPECIFIC TYPE
3727
3728 022300 012237 007646      RPTTXQ: MOV      (R2)+,EVTADD      ;STORE MESSAGE ADDRESS FOR PRINTING
3729 022304 012237 007650      MOV      (R2)+,EVTBCT      ;STORE BYTE COUNT FOR PRINTING
3730 022310 012203              MOV      (R2)+,R3          ;STORE MODEM STATUS FOR PRINTING
3731 022312              PRINTS  #EVTF2,EVTADD,EVTBCT      ;PRINT ADDR,BYTE CNT
3732 022312 013746 007650              MOV      EVTBCT,-(SP)
3733 022316 013746 007646              MOV      EVTADD,-(SP)
3734 022322 012746 015700              MOV      #EVTF2,-(SP)
3735 022326 012746 000003              MOV      #3,-(SP)
3736 022332 010600              MOV      SP,R0
3737 022334 104416              TRAP     C$PNTS
3738 022336 062706 000010              ADD      #10,SP
3739 022342 004737 023016      JSR      PC,RPTMSB      ;GO PRINT MODEM STATUS
3740 022346 000137 022126      JMP      RPT          ;GO BACK FOR NEXT EVENT ENTRY
3741
3742 022352 012237 007652      RPTDER: MOV      (R2)+,EVTTMP      ;GET ADDRESS OF DEVICE INFO MESSAGE
3743 022356 012237 007704      MOV      (R2)+,DEV1      ;STORE DEVICE REG CONTENTS FOR PRINTING
3744 022362 012237 007706      MOV      (R2)+,DEV2
3745 022366              PRINTS  #EVTF3,EVTTMP      ;PRINT DEV.CE REG CONTENTS.
3746 022366 013746 007652              MOV      EVTTMP,-(SP)
3747 022372 012746 015752              MOV      #EVTF3,-(SP)
3748 022376 012746 000002              MOV      #2,-(SP)
3749 022402 010600              MOV      SP,R0
3750 022404 104416              TRAP     C$PNTS
3751 022406 062706 000006              ADD      #6,SP
3752 022412              PRINTS  #EVTF3C,DEV1,DEV2
3753 022412 013746 007706              MOV      DEV2,-(SP)
3754 022416 013746 007704              MOV      DEV1,-(SP)
3755 022422 012746 015764              MOV      #EVTF3C,-(SP)
3756 022426 012746 000003              MOV      #3,-(SP)
3757 022432 010600              MOV      SP,R0
3758 022434 104416              TRAP     C$PNTS
3759 022436 062706 000010              ADD      #10,SP
3760 022442 000137 022126      JMP      RPT          ;GO BACK FOR NEXT EVENT ENTRY
3761
3762 022446 005037 007704      RPTDVI: CLR      DEV1
3763 022452 005037 007706      CLR      DEV2      ;CLEAR UPPER BYTES OF DEV1 & DEV2 BEFORE USE
3764 022456 112237 007704      MOV      (R2)+,DEV1      ;STORE SETUP OPERATION PARAMETERS FOR PRINTING
3765 022462 112237 007706      MOV      (R2)+,DEV2
3766 022466 012237 007710      MOV      (R2)+,DEV3
3767 022472 012237 007712      MOV      (R2)+,DEV4
3768 022476 010246              MOV      R2,-(SP)      ;SAVE R2 ON THE STACK
3769 022500 004737 023714      JSR      PC,SHWOP      ;GO PRINT MODE, MAINT-LOOP TYPE, PARAMTERS.
3770 022504 012602              MOV      (SP)+,R2      ;RESTORE R2
3771 022506 000137 022126      JMP      RPT          ;GO BACK FOR NEXT EVENT ENTRY
3772
3773      ;:REPORT END OF PASS OR ^C ABORT
3774 022512      RPTABO:
3775 022512 012237 007646      RPTTEOP: MOV      (R2)+,EVTADD      ;PASSES
3776 022516 012237 007650      MOV      (R2)+,EVTBCT      ;ERRORS
3777 022522 012237 007652      MOV      (R2)+,EVTTMP      ;START TIME OUTS
3778 022526              PRINTS  #EVTF4B,EVTADD,EVTBCT,EVTTMP      ;PRINT ADDR,RXBYTES,CMPBYTES.
3779 022526 013746 007652              MOV      EVTTMP,-(SP)
3780 022532 013746 007650              MOV      EVTBCT,-(SP)
3781 022536 013746 007646              MOV      EVTADD,-(SP)

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 94
DUMP EVENT LOG

```

3782 022542 012746 016223
3783 022546 012746 000004
3784 022552 010600
3785 022554 104416
3786 022556 062706 000012
3787 022562 000137 022126
3788
3789 022566 012237 007646
3790 022572 012237 007650
3791 022576 012237 007652
3792 022602
3793 022602 013746 007652
3794 022606 013746 007650
3795 022612 013746 007646
3796 022616 012746 016023
3797 022622 012746 000004
3798 022626 010600
3799 022630 104416
3800 022632 062706 000012
3801 022636 000137 022126
3802
3803 022642
3804 022642 012237 007646
3805 022646 012237 007650
3806 022652 012237 007652
3807 022656
3808 022656 013746 007652
3809 022662 013746 007650
3810 022666 013746 007646
3811 022672 012746 016125
3812 022676 012746 000004
3813 022702 010600
3814 022704 104416
3815 022706 062706 000012
3816
3817 022712 000137 022126
3818
3819
3820
3821
3822
3823 022716 012237 007652
3824 022722
3825 022722 013746 007652
3826 022726 012746 015752
3827 022732 012746 000002
3828 022736 010600
3829 022740 104416
3830 022742 062706 000006
3831 022746 012203
3832 022750 004737 023016
3833 022754
3834 022754 012746 016365
3835 022760 012746 000001
3836 022764 010600
3837 022766 104416

;THEN GO GET NEXT EVENT ENTRY
RPTDDE: MOV (R2)+,EVTADD ;STORE MESSAGE ADDRESS FOR PRINTING
        MOV (R2)+,EVTBCT ;STORE BYTE COUNT FOR PRINTING
        MOV (R2)+,EVTTMP ;STORE TOTAL # OF CMP ERRORS
        PRINTS #EVT4,EVTADD,EVTBCT,EVTTMP ;PRINT ADDR, BYTE CNT, # CMP ERRS
        MOV EVTTMP,-(SP)
        MOV EVTBCT,-(SP)
        MOV EVTADD,-(SP)
        MOV #EVT4,-(SP)
        MOV #4,-(SP)
        MOV SP,R0
        TRAP C$PNTS
        ADD #12,SP

;THEN GO GET NEXT EVENT ENTRY
RPTDLE: MOV (R2)+,EVTADD ;STORE MSG ADDR FOR PRINT
        MOV (R2)+,EVTBCT ;STORE BYTE COUNT
        MOV (R2)+,EVTTMP ;STORE BYTE COUNT COMP
        PRINTS #EVT4A,EVTADD,EVTBCT,EVTTMP ;PRINT ADDR,RXBYTES,CMPBYTES.
        MOV EVTTMP,-(SP)
        MOV EVTBCT,-(SP)
        MOV EVTADD,-(SP)
        MOV #EVT4A,-(SP)
        MOV #4,-(SP)
        MOV SP,R0
        TRAP C$PNTS
        ADD #12,SP

;THEN GO GET NEXT EVENT ENTRY
RPTMSC: MOV (R2)+,EVTTMP
        PRINTS #EVT3,EVTTMP ;PRINT CHANGE TYPE
        MOV EVTTMP,-(SP)
        MOV #EVT3,-(SP)
        MOV #2,-(SP)
        MOV SP,R0
        TRAP C$PNTS
        ADD #6,SP

;PUT OLD MODEM STATUS IN R3 FOR PRINTING
        MOV (R2)+,R3
        JSR PC,RPTMSB ;GO PRINT OLD MODEM STATUS
        PRINTS #EVMOCG ;GO PRINT "CHANGED TO:"
        MOV #EVMOCG,-(SP)
        MOV #1,-(SP)
        MOV SP,R0
        TRAP C$PNTS

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 95
DUMP EVENT LOG

```

3838 022770 062706 000004
3839 022774 012203
3840 022776 004737 023016
3841 023002 000137 022126
3842
3843
3844 023006 012604
3845 023010 012603
3846 023012 012602
3847 023014 000207
3848
3849
3850
3851
3852
3853 023016
3854 023016 012746 016410
3855 023022 012746 000001
3856 023026 010600
3857 023030 104416
3858 023032 062706 000004
3859 023036 012704 007560
3860 023042 012705 007574
3861 023046 005714
3862 023050 001004
3863 023052 112735 000130
3864 023056 005724
3865 023060 000407
3866 023062 032403
3867 023064 001403
3868 023066 112735 000061
3869 023072 000402
3870 023074 112735 000060
3871 023100 020427 007574
3872 023104 002760
3873 023106
3874 023106 012746 016465
3875 023112 012746 000001
3876 023116 010600
3877 023120 104416
3878 023122 062706 000004
3879 023126 000207
3880
3881

      MOV      (R2)+,R3      ;PUT NEW MODEM STATUS IN R3 FOR PRINTING
      JSR      PC,RPTMSB    ;GO PRINT NEW MODEM STATUS
RPTMSE: JMP      RPT        ;THEN GO GET NEXT EVENT

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

;REPORT MODEM STATUS SUBROUTINE
;PART OF STATISICAL REPORTING (DUMPING EVENT LOG)

RPTMSB: PRINTS  #EVMOHD      ;PRINT MODEM STATUS HEADER
                                MOV      #EVMOHD,-(SP)
                                MOV      #1,-(SP)
                                MOV      SP,R0
                                TRAP     C$PNTS
                                ADD      #4,SP
                                ;MAKE R4 A POINTER TO MODEM SIG. BIT DEF. TABLE
                                ;MAKE R5 A POINTER TO MODEM MSG. POSITION TABLE
                                ;SEE IF BIT AVAILABLE FROM DEVICE
                                ;BR IF THAT MODEM SIG. AVAILABLE
                                ;ELSE PUT 'X' IN REPORT IF SIGNAL NOT AVAILABLE
                                ;BUMP R4 TO POINT TO NEXT BIT DEFINITION
                                ;GO SEE IF CHECKED ALL MODEM SIGNALS
                                ;IF THERE, SEE IF THAT BIT IN DEVICE'S ENTRY=1
                                ;BR IF BIT (SIGNAL) VALUE =0
                                ;IF=1, PUT '1' IN REPORT MESSAGE
                                ;GO SEE IF ALL MODEM SIGNALS CHECKED
                                ;IF BIT(SIGNAL)=0, PUT '0' IN REPORT MESSAGE
                                ;SEE IF ALL BITS(SIGNALS) CHECKED
                                ;LOOP UNTIL ALL SIGNALS(BITS) CHECKED
                                ;THEN PRINT MODEM SIGNAL VALUE MESSAGE
                                MOV      #EVMOST,-(SP)
                                MOV      #1,-(SP)
                                MOV      SP,R0
                                TRAP     C$PNTS
                                ADD      #4,SP

      6$:  MOV      #MOBITS,R4
      MOV      #MOMSGS,R5
      TST      (R4)
      BNE      7$
      MOVB     #'X,@(R5)+
      TST      (R4)+
      BR       9$
      7$:  BIT      (R4)+,R3
      BEQ      8$
      MOVB     #'1,@(R5)+
      BR       9$
      8$:  MOVB     #'0,@(R5)+
      9$:  CMP      R4,#MOBITE
      BLT      6$
      PRINTS   #EVMOST

      RTS      PC            ;RETURN TO EVENT DECODING

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 96
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

--

3882
3883
3884
3885
3886
3887
3888
3889
3890
3891
3892
3893
3894
3895
3896
3897
3898
3899
3900
3901
3902
3903
3904
3905
3906
3907
3908
3909 023130 013702 006510
3910 023134 005003
3911 023136
3912 023136 010246
3913 023140 012746 015507
3914 023144 012746 000002
3915 023150 010600
3916 023152 104417
3917 023154 062706 000006
3918 023160 005737 006514
3919 023164 001416
3920 023166 112237 006534
3921 023172
3922 023172 005046
3923 023174 153716 006534
3924 023200 012746 015471
3925 023204 012746 000002
3926 023210 010600
3927 023212 104417
3928 023214 062706 000006
3929 023220 000411
3930 023222
3931 023222 012246
3932 023224 012746 015500
3933 023230 012746 000002
3934 023234 010600
3935 023236 104417
3936 023240 062706 000006
3937 023244 020237 006512

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

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

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

DUM2: CMP R2,ENADD ;COMPARE FOR LAST ADD

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

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

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

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 97
DUMP BYTES OR WORDS

3938 023250 003005
3939 023252 005203
3940 023254 022703 000010
3941 023260 001725
3942 023262 000736
3943
3944 023264 000207
3945

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

DUMEX: RTS PC

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

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 98
UPDATE TOTAL CHAR. COUNT SUBROUTINE

```

3946 .SBTTL UPDATE TOTAL CHAR. COUNT SUBROUTINE
3947
3948 :++
3949 : FUNCTIONAL DESCRIPTION:
3950 : UPDATES TOTAL CHAR. COUNT TOTCC BASED ON CURCC.
3951 : LAST MESSAGE IS TRUNCATED TO FIT INTO THE
3952 : BUFFER IF TOTAL CHAR. COUNT EXCEEDS 'BUFLIM' A MESSAGE
3953 : IS PRINTED TELLING THE OPERATOR THE TRUNCATION OCCURED.
3954
3955 : INPUTS:
3956 : CURCC= CHAR. COUNT OF MESSAGE BEING ADDED
3957 : TOTCC= TOTAL CHAR COUNT OF BUFFER ITS BEING ADDED TO
3958
3959 : OUTPUTS:
3960 : MESSAGE TO OPERATOR IF MESSAGE TRUNCATED TO FIT
3961
3962 : FUNCTIONAL SIDE EFFECTS:
3963 : LOCATION 'TEMP' USED FOR CALCULATIONS
3964
3965 : CALLING SEQUENCE:
3966 : JSR PC,ADCC ;UPDATED TOTAL CHAR. COUNT
3967 :--
3968
3969 023266 063737 006520 006530 ADDCC: ADD CURCC,TOTCC ;ADD CURRENT TO TOTAL
3970 023274 022737 001000 006530 CMP #BUFLIM,TOTCC ; COMPARE TO 'BUFLIM'
3971 023302 103027 BHIS ADDC1 ;IF NOT MORE THEN 'BUFLIM' EXIT
3972
3973 ; PRINT MESSAGE AND TRUNCATE COUNT
3974
3975 PRINTF #MSGTRU
3976
3976 023304 012746 014424 MOV #MSGTRU,-(SP)
3977 023310 012746 000001 MOV #1,-(SP)
3978 023314 010600 MOV SP,R0
3979 023316 104417 TRAP C$PNTF
3980 023320 062706 000004 ADD #4,SP
3981 023324 163737 006520 006530 SUB CURCC,TOTCC ;SUB CURRENT FROM TOTAL
3982 023332 012737 001000 006534 MOV #BUFLIM,TEMP ;MOV 'BUFLIM' TO TEMP
3983 023340 163737 006530 006534 SUB TOTCC,TEMP ;SUB TOTAL FROM 'BUFLIM'
3984 023346 013737 006534 006520 MOV TEMP,CURCC ;AND ESTABLISH NEW CURRENT
3985 023354 063737 006520 006530 ADD CURCC,TOTCC ;ADD "ADJUSTED CURRENT" TO TOTAL CHAR. CNT.
3986 023362 000207 ADDC1: RTS PC ;RETURN TO CALLER
3987

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 99
BUILD MESSAGE BUFFERS SUBROUTINE

.SBTTL BUILD MESSAGE BUFFERS SUBROUTINE

++
FUNCTIONAL DESCRIPTION:
BLDBUF-- BUILD POINTER TABLE AND BUFFERS

THIS SUBROUTINE ADDS A MESSAGE TO THE TRANSMIT OR EXPECT LIST
USING THE POINTER, BYTE COUNT, AND ADDRESS PASSED TO IT.

INPUTS:

CURCC= CHAR. COUNT OF MESSAGE TO BE ADDED
CURADD= ADDRESS OF MESSAGE TO BE ADDED
CPTR= ADDRESS OF POINTER TABLE WORD WHERE MESSAGE POINTERS ARE
TO BE BUILT
MSGTYP= VALUE TO USE AS AN INDEX TO FIND SOURCE OF MESSAGE DATA
INDEX INTO DMSGCT() AND DMSGAD().

OUTPUTS:

A MESSAGE ADDED TO EITHER TXBUF OR CMPBUF
APPROPRIATE POINTERS IN PTRTAB POINTER TABLE

CALLING SEQUENCE:

JSR PC,BLDBUF

;BUILD MESSAGE IN BUFFER AND ADD PTRS.

--

BLDBUF:

MOV R2,-(SP) ;SAVE R2 AND R3 ON THE STACK
MOV R3,-(SP)
MOV CPTR,R2

BLDB1: MOV CURADD,(R2)+ ;PUT CURRENT ADD ON POINTER TAB
MOV CURCC,(R2)+ ;PUT CURRENT CC ON POINTER TAB
MOV R2,CPTR ;PUT UPDATED R2 BACK TO CURRENT POINT
MOV MSGTYP,R2 ;GET MESSAGE TYPE TO USE AS INDEX
ASL R2 ;DOUBLE FOR WORD INDEX
MOV CURADD,TEMP ;MOVE CURRENT ADD TO TEMP
ADD CURCC,TEMP ;ADD CHAR COUNT TO IT TO GET END
MOV CURADD,R3 ;SET R3 TO CURRENT START ADD
BLDB2: MOV DMSGCT(R2),TEMP2 ;GET BYTE COUNT
MOV DMSGAD(R2),R4 ;PUT STARTING FROM ADD IN R4
ADD R4,TEMP2 ;ADD IT TO TEMP2 TO GET END OF FROM
BLDB3: MOVB (R4)+,(R3)+ ;MOV BYTE FROM PATTERN TO BUFFER
CMP R3,TEMP ;ALL DONE?
BEQ BLDBEX ;IF SO EXIT
CMP R4,TEMP2 ;IS PATTERN COUNT EXPIRED
BEQ BLDB2 ;IF SO GO START AGAIN
BR BLDB3 ;IF NOT GET ANOTHER BYTE
BLDBEX: ADD CURCC,CURADD ;BUMP CURADD
MOV (SP)+,R3 ;RESTORE R3 AND R2
MOV (SP)+,R2
RTS PC ;RETURN TO CALLER

3988
3989
3990
3991
3992
3993
3994
3995
3996
3997
3998
3999
4000
4001
4002
4003
4004
4005
4006
4007
4008
4009
4010
4011
4012 023364
4013 023364 010246
4014 023366 010346
4015 023370 013702 006524
4016
4017 023374 013722 006526
4018 023400 013722 006520
4019 023404 010237 006524
4020 023410 013702 006516
4021 023414 006302
4022 023416 013737 006526 006534
4023 023424 063737 006520 006534
4024 023432 013703 006526
4025 023436 016237 002150 006540
4026 023444 016204 002172
4027 023450 060437 006540
4028 023454 112423
4029 023456 020337 006534
4030 023462 001404
4031 023464 020437 006540
4032 023470 001762
4033 023472 000770
4034 023474 063737 006520 006526
4035 023502 012603
4036 023504 012602
4037 023506 000207
4038

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 100
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
EXPECTED LIST AND EXPECTED BUFFER. THE ROUTINE IS
NORMALLY CALLED WHEN USER COMMAND 'SET E [EXPECT]=
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
TTOTCC = 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

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 = TTOTCC
END FACSIMILE ROUTINE

4039
4040
4041
4042
4043
4044
4045
4046
4047
4048
4049
4050
4051
4052
4053
4054
4055
4056
4057
4058
4059
4060
4061
4062
4063
4064
4065
4066
4067
4068
4069
4070
4071
4072
4073
4074
4075
4076
4077
4078
4079
4080
4081
4082
4083
4084
4085
4086
4087
4088
4089
4090
4091
4092
4093
4094

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 101
CREATE FACSIMILE OF TX BUFFER AND MESSAGE LIST

```

4095 023510          FACSIMILE:
4096
4097 023510 010146    MOV      R1,-(SP)          ;SAVE R1
4098 023512 005001    CLR      R1              ;INIT R1
4099 023514 116161 003150 005150 10$:  MOVB    TXBUF(R1),CMPBUF(R1) ;COPY TX BUFFER TO EXPECTED BUFFER
4100 023522 005201    INC      R1              ;BUMP INDEX
4101 023524 020127 001000    CMP      R1,#BUFLIM ;ALL DATA COPIED ?
4102 023530 001371    BNE      10$            ;NO,BRANCH
4103
4104 023532 012701 000017      20$:  MOV      #MSG LIM,R1          ;MESSAGE LIMIT
4105 023536 006301    ASL      R1              ;MULTIPLY BY 2
4106 023540 006301    ASL      R1              ;MULTIPLY BY 2
4107 023542 012737 006150 006444    MOV      #PTRTAB,CMPPTR      ;TOP OF POINTER TABLE
4108 023550 060137 006444    ADD      R1,CMPPTR      ;START OF EXPECTED POINTER TABLE
4109 023554 005001    CLR      R1              ;INIT R1
4110
4111          ;SET UP WHILE - DO LOOP
4112 023556 012737 006150 006442    MOV      #PTRTAB, TXPTR      ;TX POINTER NOW AT TOP OF TABLE
4113 023564 012737 005150 006452    MOV      #CMPBUF, CCURAD      ;TRANSFER ADDRESS OF 1ST MESSAGE
4114 023572 062737 000002 006442    ADD      #2, TXPTR          ;BUMP POINTER
4115 023600 017737 162636 006450    MOV      @TXPTR, CTOTCC      ;BYTE COUNTER 1ST MESSAGE
4116 023606 005037 006446    CLR      CMPTOT          ;INIT EXPECTED MESSAGE COUNT
4117
4118          ;WHILE TX MESSAGE TOTAL <> EXPECTED MESSAGE TOTAL DO
4119 023612 023737 006462 006446 30$:  CMP      TXMTOT, CMPTOT      ;ALL MESSAGES COPIED ?
4120 023620 001430    BEQ      40$            ;YES,BRANCH
4121 023622 013777 006452 162614    MOV      CCURAD, @CMPPTR      ;TRANSFER ADDRESS OF MESSAGE
4122 023630 062737 000002 006444    ADD      #2, CMPPTR      ;BUMP POINTER
4123 023636 013777 006450 162600    MOV      CTOTCC, @CMPPTR      ;BYTE COUNT OF MESSAGE
4124 023644 062737 000004 006442    ADD      #4, TXPTR          ;BUMP TX MESSAGE POINTER
4125 023652 063737 006450 006452    ADD      CTOTCC, CCURAD      ;CALC. TRANSFER ADDRESS
4126 023660 017737 162556 006450    MOV      @TXPTR, CTOTCC      ;BYTE COUNT NEXT MESSAGE
4127 023666 062737 000002 006444    ADD      #2, CMPPTR      ;BUMP POINTER
4128 023674 005237 006446    INC      CMPTOT          ;INCREMENT MESSAGE COUNT
4129 023700 000744    BR       30$            ;DO IT AGAIN
4130          ;END WHILE - DO
4131 023702 013737 006464 006450 40$:  MOV      TTOTCC, CTOTCC      ;COPY TOTAL CHARACTER COUNT
4132
4133          ;END ROUTINE
4134 023710 012601    MOV      (SP)+, R1          ;RESTORE R1
4135 023712 000207    RTS      PC              ;RETURN
4136
4137

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 102
SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS

```

4138 .SBTTL          SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS
4139
4140 :++
4141 : FUNCTIONAL DESCRIPTION:
4142 :   SHWOP - SHOW MODE OF OPERATION, LOOP, QUALIFIERS
4143 :   PRINTED ON THE OPERATOR'S CONSOLE.
4144 :
4145 : INPUTS:
4146 :   DEV1=  MODE TYPE (MODTYP)
4147 :   DEV2=  MAINT LOOP TYPE (MLTYP)
4148 :   DEV3=  'RUN PASS' COUNT (RPASS) - COUNT DOWN
4149 :   DEV4=  PARAMETERS WORD (PARAM)
4150 :
4151 : IMPLICIT INPUTS:
4152 :   MODES= TABLE OF ADDRESSES OF MODE NAME STRINGS
4153 :   LOOPS= TABLE OF ADDRESSES OF LOOP TYPE NAMES
4154 :
4155 : CALLING SEQUENCE:
4156 :   JSR PC,SHWOP
4157 :--
4158
4159 023714 013702 007704      SHWOP:  MOV     DEV1,R2          ;GET THE MODE TYPE IN R2
4160 023720 006302           ASL      R2              ;MAKE IT A WORD TABLE OFFSET
4161 023722 016237 003102 006534  MOV     MODES(R2),TEMP ;GET ADDRESS OF MODE-IN-ASCII
4162 023730 013702 007706      MOV     DEV2,R2          ;GET MAINTENANCE LOOP TYPE
4163 023734 006302           ASL      R2
4164 023736 012737 013724 006542  MOV     #LP00,TEMP3      ;LOAD TEMP3 TO POINT TO '/LOOP='
4165 023744 005702           TST      R2              ;SEE IF /LOOP=XXXXX OR NONE
4166 023746 001003           BNE     10$              ;BR IF /LOOP= OF SOME KIND
4167 023750 012737 013723 006542  MOV     #LP0,TEMP3      ;IF NO LOOP THEN DON'T PRINT '/LOOP='
4168 023756 016237 003120 006536 10$:  MOV     LOOPS(R2),TEMP1 ;GET ADDRESS OF LOOP-IN-ASCII
4169 023764 013737 007710 006540  MOV     DEV3,TEMP2      ;GET NUMBER OF PASSES
4170 023772           PRINTS  #SHF0,TEMP,TEMP3,TEMP1,TEMP2
4171 023772 013746 006540           MOV     TEMP2,-(SP)
4172 023776 013746 006536           MOV     TEMP1,-(SP)
4173 024002 013746 006542           MOV     TEMP3,-(SP)
4174 024006 013746 006534           MOV     TEMP,-(SP)
4175 024012 012746 014507           MOV     #SHF0,-(SP)
4176 024016 012746 000005           MOV     #5,-(SP)
4177 024022 010600           MOV     SP,R0
4178 024024 104416           TRAP    C$PNTS
4179 024026 062706 000014           ADD     #14,SP
4180
4181 024032 005002           CLR      R2              ;NOW SET UP FOR QUALIFIERS IN ASCII
4182 024034 012737 014003 006534  MOV     #PST,TEMP
4183 024042 032737 000001 007712  BIT     #STATB,DEV4
4184 024050 001003           BNE     1$              ;SEE IF /STATUS OR /NOSTATUS
4185 024052 012737 014001 006534  MOV     #PNST,TEMP      ;BR IF /STATUS
4186 024060 012737 014014 006536 1$:  MOV     #PCK,TEMP1
4187 024066 032737 000002 007712  BIT     #DATCKB,DEV4
4188 024074 001003           BNE     2$              ;SEE IF /CHECK OR /NOCHECK
4189 024076 012737 014012 006536  MOV     #PNCK,TEMP1      ;BR IF /CHECK
4190 024104 012737 014024 006540 2$:  MOV     #PEC,TEMP2
4191 024112 032737 000004 007712  BIT     #ECHOB,DEV4
4192 024120 001003           BNE     4$              ;SEE IF /ECHO OR /NOECHO
4193 024122 012737 014022 006540  MOV     #PNEC,TEMP2      ;BR IF /ECHO

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 103
SHOW MODE OF OPERATION, LOOP TYPE AND QUALIFIERS

```

4194
4195 024130 012737 014043 006544 4$: MOV #PPR,TEMP4
4196 024136 032737 000040 007712 BIT #PROTOB,DEV4 ;SEE OF /PROTOCOL OR /NOPROTOCOL
4197 024144 001003 BNE 3$ ;BR IF /PROTOCOL
4198 024146 012737 014041 006544 MOV #PNPR,TEMP4
4199
4200 024154 012737 014033 006546 3$: MOV #PMS,TEMP5
4201 024162 032737 000010 007712 BIT #MOCHK,DEV4 ;SEE IF /MODEM OR /NOMODEM
4202 024170 001003 BNE 5$ ;BR IF MODEM
4203 024172 012737 014031 006546 MOV #PNMS,TEMP5
4204
4205
4206 024200 5$: PRINTS #SHF1,TEMP,TEMP1,TEMP2,TEMP5,TEMP4 ;,TEMP3
4207 024200 013746 006544 MOV TEMP4,-(SP)
4208 024204 013746 006546 MOV TEMP5,-(SP)
4209 024210 013746 006540 MOV TEMP2,-(SP)
4210 024214 013746 006536 MOV TEMP1,-(SP)
4211 024220 013746 006534 MOV TEMP,-(SP)
4212 024224 012746 014545 MOV #SHF1,-(SP)
4213 024230 012746 000006 MOV #6,-(SP)
4214 024234 010600 MOV SP,R0
4215 024236 104416 TRAP C$PNTS
4216 024240 062706 000016 ADD #16,SP
4217 024244 000207 RTS PC ;RETURN
4218
4219

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 104
TRAVERSE COMMAND LINE SUBROUTINES

```

4220 .SBTTL          TRAVERSE COMMAND LINE SUBROUTINES
4221
4222 :++
4223 :      P$TRV SUBROUTINE
4224 :
4225 :      PARSE THE COMMAND LINE SUBROUTINE
4226 :      TAKE ACTIONS (VIA ACTION TREE) AS PARSING LINE
4227 :      PARSING DIRECTIONS FROM 'CLI PARSING NODES'
4228 :      REGS USED:
4229 :
4230 :      R1,R5=SCRATCH                                P$NUM=NUMERIC CODE FROM DATA
4231 :      R2=ACTION CODE PARAMETER FROM TREE
4232 :      R3=PARSE TREE POINTER
4233 :      R4=INPUT STRING POINTER
4234 :      CALLING SEQUENCE:
4235 :      JSR      PC,P$TRV
4236 :      --
4237
4238 P$TRV:
4239 024246 013704 003132      MOV      P$BUFA,R4
4240 024252 013703 003134      MOV      P$TREE,R3
4241 024256 105714           P$TR5:  TSTB      (R4)                ;SEE IF ANY CHARS LEFT IN INPUT STRING
4242 024260 001441           BEQ        P$EXIT                ;BR IF NO
4243 024262 121327 000013      CMPB      (R3),#11.            ;SEE IF SPECIAL CLI CHAR CODE OR ASCII
4244 024266 003023           BGT        20$                  ;BR IF REGULAR ASCII CHAR.
4245 024270 111305           MOVB      (R3),R5                ;GET SPECIAL CHAR CODE INTO R5
4246 024272 006305           ASL        R5
4247 024274 016505 024310      MOV      10$(R5),R5            ;BUILD TRAVERSE ROUTINE ADDRESS
4248 024300 062705 024310      ADD      #10$,R5
4249 024304 004715           JSR      PC,(R5)
4250 024306 000763           BR        P$TR5                ;JSR TO SPECIAL CLI TRAVERSE ROUTINE
4251 :                                           ;GO SEE IF MORE OF STRING LEFT
4252
4253 024310 000114           10$:   .WORD    TRVERR-10$        ;TRAVERSE TABLE FOR 'CLI FUNCTIONS'
4254 024312 000134           .WORD    TRVEXI-10$              ;1
4255 024314 000152           .WORD    TRVBR-10$               ;2
4256 024316 000162           .WORD    TRVBIF-10$             ;3
4257 024320 000204           .WORD    TRVSPA-10$             ;4
4258 024322 000270           .WORD    TRVNUM-10$             ;5
4259 024324 000604           .WORD    TRVALP-10$             ;6
4260 024326 000650           .WORD    TRVALN-10$             ;7
4261 024330 000270           .WORD    TRVOCT-10$             ;8
4262 024332 000256           .WORD    TRVDEC-10$             ;9
4263 024334 000736           .WORD    TRVSTR-10$             ;10
4264
4265 ;NOT A SPECIAL CODE
4266
4267 024336 121314           20$:   CMPB      (R3),(R4)        ;SEE IF FIRST CHAR OF STRING IS A MATCH
4268 024340 001403           BEQ        22$                  ;BR IF A MATCH
4269 024342 004737 024406      JSR      PC,TRVBRC            ;IF NOT A MATCH, GO TAKE MISS BRANCH
4270 024346 000743           BR        P$TR5                ; THEN GO BACK PT'G TO MISS NODE
4271 024350 004737 024366      22$:   JSR      PC,TRVACT        ;IF A MATCH, GO DO ACTION DEFINED BY
4272 024354 062703 000004      ADD      #4,R3                ; ACTION CODE IN CLI NODE, THEN
4273 :                                           ; ADJUST PTR TO NEXT CLI NODE
4274 024360 005204           INC        R4                    ;ADJUST BUF PTR TO NEXT CHAR IF MATCH
4275 024362 000735           BR        P$TR5

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 105
TRAVERSE COMMAND LINE SUBROUTINES

```

4276
4277 024364 000207      P$EXIT: RTS      PC      ;RETURN FROM PARSER
4278
4279      ;-----
4280
4281      ;GOTO USER ACTION ROUTINE
4282 024366 116302 000001 TRVACT: MOV      1(R3),R2      ;GET ACTION CODE FROM CLI NODE
4283 024372 042702 177400      BIC      #177400,R2      ;CLEAR ANY SIGN EXTENSION
4284 024376 013705 003136      MOV      P$ACT,R5      ;GET ADDRESS OF CLI ACTION ROUTINE
4285 024402 004715      JSR      PC,(R5)      ;GO DO ACTION DEFINED BY CODE
4286 024404 000207      RTS      PC      ;RETURN TO CALLING CODE
4287
4288      ;TAKE BRANCH IN TREE
4289 024406 016305 000002 TRVBRC: MOV      2(R3),R5      ;GET BRANCH DISPLACEMENT FROM TREE
4290 024412 060503      ADD      R5,R3      ; AND POINT R3 TO THE 'MISS' NODE
4291 024414 000207      RTS      PC      ; RETURN TO P$TRV
4292
4293      ;NO BRANCH TAKEN
4294 024416 062703 000004 TRVNOB: ADD      #4,R3      ;THINGS OK, UPDATE R3 TO POINT TO NEXT
4295 024422 000207      RTS      PC      ; NODE AND RETURN TO P$TRV
4296
4297      ;-----
4298 024424 004737 024366 TRVERR: JSR      PC,TRVACT      ;TAKE ERROR ACTION
4299 024430 112737 177777 003147      MOV      #-1,P$GDBD      ;SET ERROR RETURN FLAG
4300 024436 005726      TST      (SP)+      ;GET RID OF 'JSR PUSH TO TRVERR'
4301 024440 000137 024364      JMP      P$EXIT      ;RETURN DIRECT TO EXIT OF P$TRV ROUTINE
4302
4303 024444 004737 024366 TRVEXI: JSR      PC,TRVACT      ;TAKE EXIT ACTION
4304 024450 105037 003147      CLRB      P$GDBD      ;SET GOOD/BAD FLAG TO 'SUCCESS (0)'
4305 024454 005726      TST      (SP)+      ;GET RID OF 'JSR PUSH TO TRVEXI'
4306 024456 000137 024364      JMP      P$EXIT      ;RETURN DIRECT TO EXIT OF P$TRV ROUTINE
4307
4308 024462 004737 024366 TRVBR: JSR      PC,TRVACT      ;GO TAKE BRANCH ACTION
4309 024466 000137 024406      JMP      TRVBRC
4310
4311 024472 004737 024366 TRVBIF: JSR      PC,TRVACT
4312 024476 105737 003147      TSTB      P$GDBD
4313 024502 001402      BEQ      1$      ;SEE IF P$GDBD SET OR CLEARED BY ACTION
4314 024504 000137 024406      JMP      TRVBRC      ;IF CLEAR FALL THRU TO NEXT NODE
4315 024510 000137 024416      JMP      TRVNOB      ;ELSE TAKE THE 'MISS' BRANCH
4316
4317 024514 005005      TRVSPA: CLR      R5      ;CLEAR 'SPACE OR TAB FOUND' FLAG
4318 024516 121427 000011 1$:      CMPB      (R4),#11      ;SEE IF CHAR. IN CMD LINE= TAB
4319 024522 001003      BNE      2$      ;BR IF NO, NOT A TAB
4320 024524 005204      INC      R4      ;INC INPUT STRING POINTER
4321 024526 005205      INC      R5      ;INDICATE A TAB FOUND
4322 024530 000772      BR      1$      ;GO CHECK NEXT CHAR
4323
4324 024532 121427 000040 2$:      CMPB      (R4),#40      ;SEE IF CHAR. IN CMD LINE= SPACE
4325 024536 001003      BNE      10$      ;BR IF NO, NON-SPACE OR NON-TAB CHAR.
4326 024540 005204      INC      R4      ;INC INPUT STRING POINTER
4327 024542 005205      INC      R5      ;INDICATE A SPACE FOUND
4328 024544 000764      BR      1$      ;GO CHECK NEXT CHAR
4329 024546 005705      10$:      TST      R5      ;SEE IF ANY SPACES OR TABS FOUND
4330 024550 001404      BEQ      15$      ;BR IF NO, TAKE NO ACTION
4331 024552 004737 024366      JSR      PC,TRVACT      ;GO TAKE ACTION IF ANY FOUND

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 106
TRAVERSE COMMAND LINE SUBROUTINES

```

4332 024556 000137 024416      JMP      TRVNOB      ;JUST GO UPDATE R3 TO NEXT NODE IF OK
4333 024562 000137 024406      15$: JMP      TRVBRC      ;TAKE BRANCH (MISS) IF NONE FOUND
4334
4335
4336 024566 012737 000012 003144 TRVDEC: MOV      #10.,P$RADX      ;USE DECIMAL AS RADIX AND ASSUME +
4337 024574 000137 024606      JMP      TRVNMA
4338 024600      TRVOCT: ;(SAME AS TRVNUM SINCE DEFAULT RADIX IS OCTAL)
4339 024600 012737 000010 003144 TRVNUM: MOV      #8.,P$RADX      ;USE OCTAL AS RADIX AND ASSUME +
4340 024606 005005      TRVNMA: CLR      R5      ;CLEAR DIGIT COUNTER
4341 024610 121427 000053      CMPB     (R4),#'+      ;SEE IF THERE'S A + SIGN THERE
4342 024614 001001      BNE      10$      ;BR IF NO
4343 024616 000406      BR       11$      ;ELSE P$RADX ALREADY SAYS +, JUST BR
4344 024620 121427 000055      10$: CMPB     (R4),#'-      ;SEE IF THERE'S A - SIGN THERE
4345 024624 001004      BNE      1$      ;BR IF NO
4346 024626 112737 177777 003145 MOVB     #-1,P$RADX+1      ;SET 'MINUS FLAG' (HI BYTE OF P$RADX)
4347 024634 005204      11$: INC      R4      ;BUMP R4 TO POINT TO FIRST CHAR
4348
4349 024636 121427 000060      1$:  CMPB     (R4),#60      ;SEE IF CHAR. LESS THAN A '0'
4350 024642 002434      BLT      2$      ;BR IF YES (NOT NUMERIC)
4351 024644 121427 000067      CMPB     (R4),#67      ;SEE IF CHAR. GREATER THAN A '7'
4352 024650 003426      BLE      13$      ;BR IF YES
4353 024652 123727 003144 000012 CMPB     P$RADX,#10.      ;SEE IF IN DECIMAL MODE
4354 024660 001417      BEQ      12$      ;BR IF YES (CAN USE HIGHER LIMIT)
4355 024662 121427 000071      CMPB     (R4),#71      ;SEE IF DIGIT WAS A 8 OR 9
4356 024666 003022      BGT      2$      ;BR IF NON-NUMERIC
4357 024670      PRINTF   #CLIBRX      ;ELSE WAS A 8 OR 9 WHEN IN OCTAL RADIX
4358 024670 012746 011700      MOV      #CLIBRX,-(SP)
4359 024674 012746 000001      MOV      #1,-(SP)
4360 024700 010600      MOV      SP,R0
4361 024702 104417      TRAP     C$PNTF
4362 024704 062706 000004      ADD      #4,SP
4363 024710 112737 177777 003147 MOVB     #-1,P$GDBD      ;SET ERROR RETURN FLAG
4364 024716 000474      BR       5$      ;PRINT ERROR AND TAKE MISS
4365
4366 024720 121427 000071      12$: CMPB     (R4),#71      ;SEE IF CHAR. GREATER THAN A '9'
4367 024724 003003      BGT      2$      ;BR IF YES (NOT NUMERIC)
4368 024726 005204      13$: INC      R4      ;UPDATE CMD LINE PTR TO NEXT CHAR.
4369 024730 005205      INC      R5      ;INDICATE A NUMERIC FOUND
4370 024732 000741      BR       1$      ;GO LOOK AT NEXT CHAR.
4371
4372 024734 005705      2$:  TST      R5      ;SEE IF FOUND ANY NUMERICS
4373 024736 001464      BEQ      5$      ;BR IF NO, TAKE 'MISS' BRANCH
4374 024740 010401      MOV      R4,R1      ;GET POINTER TO START OF NUMERIC STRING
4375 024742 160501      SUB      R5,R1
4376 024744 005037 003142      CLR      P$NUM      ;CLEAR LOC. WHERE VALUE WILL BE STORED
4377 024750 112102      3$: MOVB     (R1)+,R2      ;GET ASCII CHAR AND CONVERT IT TO A #
4378 024752 162702 000060      SUB      #60,R2
4379 024756 006337 003142      ASL      P$NUM      ;SHIFT CURRENT VALUE TO MAKE ROOM
4380 024762 103437      BCS      7$      ;ERROR IF NUMBER TOO BIG
4381 024764 013737 003142 003140 MOV      P$NUM,P$CNT      ;SAVE FOR LATER IN CASE DECIMAL RADIX
4382 024772 006337 003142      ASL      P$NUM
4383 024776 103431      BCS      7$      ;ERROR IF NUMBER TOO BIG
4384 025000 006337 003142      ASL      P$NUM
4385 025004 103426      BCS      7$      ;ERROR IF NUMBER TOO BIG
4386 025006 123727 003144 000012 CMPB     P$RADX,#10.      ;SEE IF DECIMAL RADIX
4387 025014 001004      BNE      4$      ;BR IF NOT EQUAL

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 107
TRAVERSE COMMAND LINE SUBROUTINES

4388	025016	063737	003140	003142		ADD	P\$CNT,P\$NUM	
4389	025024	103416				BCS	7\$	;ERROR IF NUMBER TOO BIG
4390	025026	060237	003142		4\$:	ADD	R2,P\$NUM	
4391	025032	103413				BCS	7\$	;ERROR IF NUMBER TOO BIG
4392	025034	005305				DEC	R5	
4393	025036	001344				BNE	3\$	
4394	025040	105737	003145			TSTB	P\$RADX+1	;SEE IF NUM WAS PRECEDED BY A - SIGN
4395	025044	001402				BEQ	15\$	; BR IF NO
4396	025046	005437	003142			NEG	P\$NUM	; ELSE NEGATE THE NUMBER BEFORE LEAVING
4397	025052	004737	024366		15\$:	JSR	PC,TRVACT	;SINCE NUMERIC FOUND, GO TAKE ACTION
4398	025056	000137	024416			JMP	TRVNOB	;GO POINT R3 TO NEXT NODE
4399								
4400	025062				7\$:	PRINTF	#CLINBG	;PRINT NUMBER TOO BIG ERROR
4401	025062	012746	011656					MOV #CLINBG,-(SP)
4402	025066	012746	000001					MOV #1,-(SP)
4403	025072	010600						MOV SP,R0
4404	025074	104417						TRAP C\$PNTF
4405	025076	062706	000004					ADD #4,SP
4406	025102	112737	177777	003147		MOVB	#-1,P\$GDBD	;SET ERROR RETURN FLAG
4407	025110	000137	024406		5\$:	JMP	TRVBRC	;TAKE 'MISS' BRANCH
4408								
4409								
4410	025114	005005			TRVALP:	CLR	R5	;CLEAR ALPHA FOUND FLAG
4411	025116	121427	000101		1\$:	CMPB	(R4),#101	;SEE IF CHAR. LESS THAN A 'A'
4412	025122	002406				BLT	2\$	;BR IF YES (NOT ALPHA)
4413	025124	121427	000132			CMPB	(R4),#132	;SEE IF CHAR. GREATER THAN A 'Z'
4414	025130	003003				BGT	2\$	;BR IF YES (NOT ALPHA)
4415	025132	005204				INC	R4	;UPDATE CMD LINE PTR TO NEXT CHAR
4416	025134	005205				INC	R5	;INDICATE AN ALPHA WAS FOUND
4417	025136	000767				BR	1\$	;GO LOOK AT NEXT CHAR.
4418	025140	005705			2\$:	TST	R5	;SEE IF ANY ALPHA'S WERE FOUND
4419	025142	001404				BEQ	3\$	;BR IF NO
4420	025144	004737	024366			JSR	PC,TRVACT	;IF ANY FOUND TAKE ACTION
4421	025150	000137	024416			JMP	TRVNOB	;THEN UPDATE R3 TO NEXT NODE -NO BRANCH
4422	025154	000137	024406		3\$:	JMP	TRVBRC	;NONE FOUND, TAKE MISS BRANCH
4423								
4424	025160	005005			TRVALN:	CLR	R5	;CLEAR ALPHANUM FOUND FLAG
4425	025162	121427	000060		10\$:	CMPB	(R4),#60	;SEE IF CHAR. LESS THAN A '0'
4426	025166	002417				BLT	2\$	;BR IF YES (NOT NUMERIC OR ALPHA)
4427	025170	121427	000072			CMPB	(R4),#72	;SEE IF CHAR. GREATER THAN A '9'
4428	025174	003003				BGT	1\$	;BR IF YES (NOT NUMERIC)
4429	025176	005204				INC	R4	;UPDATE CMD LINE PTR TO NEXT CHAR.
4430	025200	005205				INC	R5	;INDICATE A NUMERIC FOUND
4431	025202	000767				BR	10\$	;GO LOOK AT NEXT CHAR.
4432	025204	121427	000101		1\$:	CMPB	(R4),#101	;SEE IF CHAR. LESS THAN A 'A'
4433	025210	002406				BLT	2\$	;BR IF YES (NOT ALPHA)
4434	025212	121427	000132			CMPB	(R4),#132	;SEE IF CHAR. GREATER THAN A 'Z'
4435	025216	003003				BGT	2\$	;BR IF YES (NOT ALPHA)
4436	025220	005204				INC	R4	;UPDATE CMD LINE PTR TO NEXT CHAR
4437	025222	005205				INC	R5	;INDICATE AN ALPHA FOUND
4438	025224	000756				BR	10\$	;GO LOOK AT NEXT CHAR.
4439	025226	005705			2\$:	TST	R5	;SEE IF ANY ALPHANUM'S WERE FOUND
4440	025230	001404				BEQ	3\$	;BR IF NO
4441	025232	004737	024366			JSR	PC,TRVACT	;IF ANY FOUND TAKE ACTION
4442	025236	000137	024416			JMP	TRVNOB	;THEN UPDATE R3 TO NEXT NODE -NO BRANCH
4443	025242	000137	024406		3\$:	JMP	TRVBRC	;NONE FOUND, TAKE MISS BRANCH

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 108
TRAVERSE COMMAND LINE SUBROUTINES

```

4444
4445
4446
4447 025246 010401          TRVSTR: MOV      R4,R1          ;POINT R1 TO CMD STRING
4448 025250 010305          MOV      R3,R5
4449 025252 062705 000006   ADD      #6,R5          ;POINT R5 TO MATCH STRING FROM CLI NODE
4450 025256 005037 003140   CLR      P$CNT          ;CLEAR CHAR MATCH COUNT
4451 025262 105715          2$:  TSTB     (R5)          ;SEE IF END OF MATCH STRING YET
4452 025264 001411          BEQ      10$          ;BR IF YES
4453 025266 105711          TSTB     (R1)          ;SEE IF END OF CMD LINE YET
4454 025270 001407          BEQ      10$          ;BR IF YES
4455 025272 121115          CMPB     (R1),(R5)        ;SEE IF CHARACTERS MATCH
4456 025274 001005          BNE      10$          ;BR IF NO
4457 025276 005237 003140   INC      P$CNT          ;MATCH -INCREMENT MATCH COUNT
4458 025302 005201          INC      R1          ;UPDATE STRING POINTERS
4459 025304 005205          INC      R5
4460 025306 000765          BR       2$          ;BR TO CONTINUE CHECKING CHARS.
4461
4462 025310 005737 003140   10$:  TST      P$CNT          ;WHEN DONE SEE IF ANY MATCHES FOUND
4463 025314 001406          BEQ      15$          ;BR IF NO, GO TAKE THE MISS BRANCH
4464 025316 010104          MOV      R1,R4          ;POINT CMD POINTER TO END OF STRING &
4465 025320 004737 024366   JSR      PC,TRVACT        ;IF A MATCH FOUND, GO DO MATCH ACTION
4466 025324 066303 000004   ADD      4(R3),R3          ;UPDATE R3 TO NEXT NODE (NO BRANCH)
4467 025330 000207          RTS      PC          ; (NO RETURN THRU TRVNOB SINCE DIFFERNT
4468                                     ; DISPLACEMENT DUE TO MATCH STRING)
4469 025332 000137 024406   15$:  JMP      TRVBRC          ; GO TAKE BRANCH
4470
4471                                     ; (PARSED OK), -1 IF ILL CMD.....
4472                                     ;-----
4473

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 109
REPORT CODING SECTION

.SBTTL REPORT CODING SECTION

;;++
; THE REPORT CODING SECTION CONTAINS THE
; 'PRINTS' CALLS THAT GENERATE STATISTICAL REPORTS.
;--

4474
4475
4476
4477
4478
4479
4480
4481

4482 025336
4483 025336

4484
4485
4486 025336 004737 020612

4487
4488
4489
4490
4491

4492 025342
4493 025342

4494 025342 104425

BGNRPT

L\$RPT::

JSR PC,REPORT

;CALL SUBROUTINE TO DUMP EVENT LOG
; AND BASE TABLE

ENDRPT

L10010: TRAP CSRPT

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 110
PROTECTION TABLE

4495
4496
4497
4498
4499
4500
4501
4502
4503
4504
4505
4506
4507
4508
4509
4510

025344
025344

025344 177777
025346 177777
025350 177777

025352

.SBTTL PROTECTION TABLE

:++
: THIS TABLE IS USED BY THE RUNTIME SERVICES
: TO PROTECT THE LOAD MEDIA.
:--

BGNPROT

-1
-1
-1

ENDPROT

L\$PROT::

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

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 111
INITIALIZE SECTION

```

;--
; THE INITIALIZE SECTION CONTAINS THE CODING THAT IS PERFORMED
; AT THE BEGINNING OF EACH PASS.
;--

```

	.SBTTL	INITIALIZE SECTION			
4511					
4512					
4513					
4514					
4515					
4516					
4517					
4518	025352	BGNINIT			
4519	025352				L\$INIT::
4520					
4521	025352	005037	003010		:INIT COMMAND STORAGE VARIABLE
4522	025356	005737	006564		: 'EXIT' COMMAND ?
4523	025362	001403			: NO,BRANCH
4524	025364	005037	006564		: INIT 'DO CLEAN VARIABLE'
4525	025370				: GO CLEANUP AND EXIT
4526	025370	104444			TRAP CSDCLN
4527					
4528	025372	012737	177777	006566	INIT1:
4529	025400				MOV #-1,RESFLG
4530	025400	012700	000040		READEF #EF.START
4531	025404	104447			
4532	025406				BCOMplete START
4533	025406	103417			
4534	025410				READEF #EF.RESTART
4535	025410	012700	000037		
4536	025414	104447			
4537	025416				BCOMplete RESTRT
4538	025416	103513			
4539	025420				READEF #EF.CONTINUE
4540	025420	012700	000036		
4541	025424	104447			
4542	025426				BNCOMplete S1
4543	025426	103002			
4544	025430	000137	026120		S1:
4545	025434				JMP ENDIT
4546	025434	012700	000035		READEF #EF.NEW
4547	025440	104447			
4548	025442				BCOMplete NEW
4549	025442	103521			
4550	025444	000523			BR GETPRM
4551					
4552	025446	005037	006566		START:
4553	025452	005037	006626		CLR RESFLG
4554					CLR CLKVEC
4555	025456	012702	006622		
4556	025462				MOV #CLKCSR,R2
4557	025462	012700	000114		CLOCK L,R1
4558	025466	104462			
4559	025470	010001			
4560	025472				BNCOMplete S2
4561	025472	103006			
4562	025474	004737	017732		JSR PC,CLKSET
4563	025500	012737	000100	006632	MOV #LCLKEN,CLKEN
4564	025506	000457			BR RESTRT
4565					
4566	025510				S2: CLOCK P,R1
					:LOOK FOR A P-CLOCK SINCE NO LINE CLOCK

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 112
INITIALIZE SECTION

```

4567 025510 012700 000120      MOV      #'P,RO
4568 025514 104462      TRAP      C$CLCK
4569 025516 010001      MOV      RO,R1
4570 025520      BNCOMPLETE      S3      ; IF NONE THERE GO SEE IF THIS IS LSI
4571 025520 103017      BCC      S3
4572 025522 004737 017732      JSR      PC,CLKSET      ; ELSE GO SET UP CLOCK INFO & VECTOR
4573 025526 062737 000002 006622      ADD      #2,CLKCSR      ;POINT CLKCSR TO P-CLK COUNT SET REG.
4574 025534 012777 001600 161060      MOV      #PCLKCT,@CLKCSR      ;LOAD CLK SET REG. WITH COUNT VALUE
4575 025542 162737 000002 006622      SUB      #2,CLKCSR      ;POINT CLKCSR BAC TO P-CLK CSR
4576 025550 012737 000111 006632      MOV      #PCLKEN,CLKEN      ;SETUP THE ENABLE THE P-CLK DATA
4577 025556 000433      BR      RESTRT
4578
4579 025560      S3:      READBUS      ;READ BUS TYPE TO SEE IF ON AN LSI
4580 025560 104407      TRAP      C$RDBU
4581 025562      BNCOMPLETE      S4      ;BR IF NOT, NO CHANCE OF A CLOCK
4582 025562 103021      BCC      S4
4583 025564 012737 000100 006626      MOV      #100,CLKVEC      ;LOAD 100 AS CLK VECTOR
4584 025572 005037 006624      CLR      CLKBR      ;LOAD 0 AS CLK INT. LEVEL
4585 025576 012737 006632 006622      MOV      #CLKEN,CLKCSR      ;KLUDGE UP THE CSR & ENABLE DATA LOCS
4586 025604      GMANID      L5060,CLKHZ,D,377,50.,60.,YES
4587 025604 104443      TRAP      C$GMAN
4588 025606 000406      BR      10000$
4589 025610 006630      .WORD      CLKHZ
4590 025612 000052      .WORD      T$CODE
4591 025614 014072      .WORD      L5060
4592 025616 000377      .WORD      377
4593 025620 000062      .WORD      T$LOLIM
4594 025622 000074      .WORD      T$HILIM
4595 025624      10000$:
4596 025624 000410      BR      RESTRT
4597
4598 025626      S4:      PRINTF      #BDCLK      ;INFORM OPR. NO CLOCK, & EXIT INIT
4599 025626 012746 014210      MOV      #BDCLK,-(SP)
4600 025632 012746 000001      MOV      #1,-(SP)
4601 025636 010600      MOV      SP,RO
4602 025640 104417      TRAP      C$PNTF
4603 025642 062706 000004      ADD      #4,SP
4604
4605 025646 005037 006634      RESTRT:      CLR      TIMMIN      ;CLEAR TIME SINCE START LOCATIONS
4606 025652 005037 006636      CLR      TIMSEC
4607 025656 013737 006630 006640      MOV      CLKHZ,TIMTCK      ;LOAD TICKS/SEC
4608 025664 012702 006652      MOV      #EVTLOG,R2      ;INIT EVENT TABLE TO ALL 1'S AFTER EACH
4609 025670 010237 006650      MOV      R2,EVTPTN      ; START OR RES AND INIT TABLE POINTER
4610 025674 012722 177777      1$:      MOV      #-1,(R2)+
4611 025700 020227 007554      CMP      R2,#EVTEND
4612 025704 001373      BNE      1$
4613
4614 025706 012737 177777 006560      NEW:      MOV      #-1,LOGUNT      ;INITIALIZE LOGICAL UNIT #
4615
4616 025714 005237 006560      GETPRM:      INC      LOGUNT      ;POINT TO NEXT LOGICAL UNIT
4617 025720 023737 006560 002012      CMP      LOGUNT,L$UNIT      ;SEE IF PAST MAX. LOG. UNIT #
4618 025726 002367      BGE      NEW      ;BR IF YES, AND START OVER
4619
4620 025730      GPWARD      LOGUNT,R1      ;GET THE P-TABLE FOR THIS LOG. UNIT
4621 025730 013700 006560      MOV      LOGUNT,RO
4622 025734 104442      TRAP      C$GPWARD

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 113
INITIALIZE SECTION

```

4623 025736 010001
4624 025740
4625 025740 103365
4626
4627 025742 011137 006574
4628
4629
4630
4631
4632 025746 016137 000002 011450
4633
4634
4635 025754 016137 000002 011452
4636 025762 062737 000002 011452
4637 025770 016137 000002 011454
4638 025776 062737 000002 011454
4639 026004 016137 000002 011456
4640 026012 062737 000004 011456
4641 026020 016137 000002 011460
4642 026026 062737 000006 011460
4643
4644 026034 016137 000004 011462
4645 026042 016137 000004 011464
4646 026050 062737 000004 011464
4647 026056 016137 000006 011466
4648 026064 016137 000014 011522
4649 026072 016137 000010 037132
4650 026100 001004
4651 026102 112737 000001 037065
4652 026110 000403
4653 026112 116137 000012 037065 10$:
4654
4655
4656 026120
4657 026120 012746 000340
4658 026124 012746 017756
4659 026130 013746 006626
4660 026134 012746 000003
4661 026140 104437
4662 026142 062706 000010
4663
4664
4665
4666 026146
4667 026146 012746 000240
4668 026152 012746 035556
4669 026156 013746 011462
4670 026162 012746 000003
4671 026166 104437
4672 026170 062706 000010
4673 026174
4674 026174 012746 000240
4675 026200 012746 036230
4676 026204 013746 011464
4677 026210 012746 000003
4678 026214 104437

BNCOMPLETE GETPRM ;IF NO P-TABLE AVAIL., GO GET NEXT ONE
MOV (R1),FHDPLX ;PUT FULL OR HALF DUPLEX ANSWER IN LOC.

;DEVICE DEPENDENT PART OF GETTING INFO FROM P-TABLE
MOV 2(R1),RXCSR ;STORE AWAY CSR ADDRESSES

MOV 2(R1),PARCSR
ADD #2,PARCSR
MOV 2(R1),RXDBUF
ADD #2,RXDBUF
MOV 2(R1),TXCSR
ADD #4,TXCSR
MOV 2(R1),TXDBUF
ADD #6,TXDBUF

MOV 4(R1),INVEC ;STORE AWAY INPUT INTERRUPT VECTOR
MOV 4(R1),OUTVEC
ADD #4,OUTVEC ;BUILD OUTPUT INTERRUPT VECTOR
MOV 6(R1),INTPRI ;STORE AWAY INTERRUPT PRIORITY
MOV 14(R1),RNODE ;STORE AWAY THE REMOTE NODE TYPE
MOV 10(R1),MPPTP ;MULTI-POINT =1
BNE 10$ ;IF MULTI-POINT GET ADDRESS FROM PTABLE
MOVB #1,TRIBN ;IF POINT-POINT ADDRESS ALWAYS =1
BR ENDIT ;EXIT
MOVB 12(R1),TRIBN ;STORE AWAY TRIB ADDRESS

ENDIT: SETVEC CLKVEC,#CLKINT,#340 ;SETUP CLOCK VECTOR

MOV #340,-(SP)
MOV #CLKINT,-(SP)
MOV CLKVEC,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP

;DEVICE DEPENDENT VECTOR SETUP
SETVEC INVEC,#DVRXI,#PRI05 ;SETUP INPUT INTERRUPT VECTOR

MOV #PRI05,-(SP)
MOV #DVRXI,-(SP)
MOV INVEC,-(SP)
MOV #3,-(SP)
TRAP C$SVEC
ADD #10,SP

SETVEC OUTVEC,#DVTXI,#PRI05 ;SETUP OUTPUT INTERRUPT VECTOR

MOV #PRI05,-(SP)
MOV #DVTXI,-(SP)
MOV OUTVEC,-(SP)
MOV #3,-(SP)
TRAP C$SVEC

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 114
INITIALIZE SECTION

4679	026216	062706	000010			ADD	#10,SP
4680							
4681	026222			SETPRI	#PRI00		
4682	026222	012700	000000			;SET THE 'RUN' PRIORITY TO 0	
4683	026226	104441				MOV	#PRI00,R0
4684	026230			EXIT	INIT	TRAP	C\$SPRI
4685	026230	104432				TRAP	C\$EXIT
4686	026232	000002				.WORD	L10012-.
4687							
4688							
4689				.EVEN			
4690							
4691	026234			ENDINIT			
4692	026234						
4693	026234	104411				L10012: TRAP	C\$INIT

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 115
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.
:--

4694
4695
4696
4697
4698
4699
4700
4701
4702
4703
4704
4705
4706
4707
4708
4709

026236
026236

026236
026236

104461

BGNAUTO

ENDAUTO

LSAUTO::

L10013: TRAP CSAUTO

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 116
CLEANUP CODING SECTION

4710			.SBTTL	CLEANUP CODING SECTION			
4711							
4712			::++				
4713			:	THE CLEANUP CODING SECTION CONTAINS THE CODING THAT IS PERFORMED			
4714			:	AFTER THE HARDWARE TESTS HAVE BEEN PERFORMED.			
4715			::--				
4716							
4717	026240		BGNCLN				
4718	026240				L\$CLEAN::		
4719							
4720	026240	005077	160356	CLR	@CLKCSR	:DISABLE CLOCK	
4721	026244			SETPRI	#PRI07	:SET PROCESSOR PRIORITY BACK TO 7	
4722	026244	012700	000340				
4723	026250	104441				MOV TRAP	#PRI07,R0
4724	026252	022737	000057	003010	CMP	#EXIT,KEYWD1	C\$SPRI
4725	026260	001416			BEQ	EXITCLN	
4726						:YES,BRANCH	
4727							
4728	026262	012737	000026	006534	::LOG ^C ABORT IN EVENT LOG		
4729	026270	013737	006502	006544	MOV	#ABO,TEMP	:EVENT TYPE
4730	026276	013737	006504	006540	MOV	OPVAR,TEMP4	:START TIME-OUTS
4731	026304	013737	006506	006542	MOV	PSCNT,TEMP2	:PASSES
4732	026312	004737	020506		MOV	ERRCNT,TEMP3	:ERRORS
4733	026316				JSR	PC,LOG\$5	:GO LOG IT
4734	026316	104433			EXITCLN:BRESET	:CLEAR ALL BEFORE END	
4735							TRAP C\$RESET
4736	026320				EXIT CLN		
4737	026320	104432					TRAP C\$EXIT
4738	026322	000002				.WORD	L10014-
4739							
4740							
4741					.EVEN		
4742							
4743	026324				ENDCLN		
4744	026324						
4745	026324	104412				L10014:	TRAP C\$CLEAN

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 117
DROP UNIT SECTION

.SBTTL DROP UNIT SECTION

;;
; THE DROP-UNIT SECTION CONTAINS THE CODING THAT CAUSES A DEVICE
; TO NO LONGER BE TESTED.
;--

4746
4747
4748
4749
4750
4751
4752
4753
4754
4755
4756
4757
4758
4759
4760
4761
4762
4763
4764
4765
4766

026326
026326

026326
026326 000167
026330 000000

BGNDU

EXIT DU

.EVEN

ENDDU

L\$DU::

.WORD JSJMP
.WORD L10015-2-.

L10015:
TRAP C\$DU

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 118
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.
:--

4767
4768
4769
4770
4771
4772
4773
4774
4775
4776
4777
4778
4779
4780
4781
4782
4783
4784
4785
4786
4787
4788
4789
4790

026334
026334

BGNAU

LSAU::

EXIT AU

.WORD JSJMP
.WORD L10016-2-.

.EVEN

ENDAU

L10016: TRAP CSAU

026340
026340
026340 104452

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 119
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 ISOLATE FAILURES TO THE COMPUTER
: EQUIPMENT, THE COMMUNICATION LINK, OR THE MODEM.
:--

```

4791
4792
4793
4794
4795
4796
4797
4798
4799
4800
4801
4802
4803
4804
4805
4806
4807
4808
4809
4810
4811
4812
4813
4814
4815
4816
4817
4818
4819
4820
4821
4822
4823
4824
4825
4826
4827
4828
4829
4830
4831
4832
4833
4834
4835
4836
4837
4838
4839
4840
4841
4842
4843
4844
4845
4846

026342
026342

BGNTST

T1::

.SBTTL PROGRAM SETUP SECTION

```

MOV CLKEN, @CLKCSR ;ENABLE THE CLOCK

GTXRXB:
GTRA2: CLR R1
        MOV #1, TIMER1 ;SET TIMER TO COUNT 1 TICK
        TST TIMER1 ;CHECK FOR IT TO BE COUNTED OFF
        BEQ GTRA3 ;BRANCH IF CLOCK EXISTS (COUNTED A TICK)
        DEC R1
        BNE 1$ ;KEEP CHECKING UNTIL R1 DOES FULL COUNTDOWN
        PRINTF #NOCLK ;PRINT BAD CLK MSG AND WARN OF HANG IF TIMEOUT
                                MOV #NOCLK, -(SP)
                                MOV #1, -(SP)
                                MOV SP, R0
                                TRAP C$PNTF
                                ADD #4, SP

GTRA3: TST RESFLG ;SEE IF HERE AFTER A RESTART.
        BNE GTRA5 ;BR IF HERE CAUSE OF A RESTART

; CLEAR COUNTS AND SET UP DEFAULTS

GTRA4: CLR TOTCC ;CLEAR TOTAL CHAR. COUNT TEMP. LOC.
        CLR TTOTCC ;CLEAR TOTAL CHAR. COUNT FOR TX BUFF
        CLR CTOTCC ;CLEAR TOTAL CHAR. COUNT FOR CMP BUFF
        MOV #PTRTAB, R1 ;INIT TRANSMIT MESSAGE POINTER
        MOV R1, TXPTR
        CLR RXPTR ;ZERO RX POINTER

MOV #PTR13, CMPPTR ;INIT COMPARE MESSAGE POINTER

MOV #5, MSGTYP ;SET UP DEFAULT MSG TYPE (QUICK FOX - ITEP MSG)
MOV MSG5C, CURCC ;SET UP DEFAULT CHAR COUNT
MOV #TXBUF, TCURAD ;SET UP CURRENT ADD TO START OF TX BUFFER
MOV #CMPBUF, CCURAD ;SET UP CURRENT ADD TO START OF CMP BUFFER

MOV TCURAD, CURADD ;SETUP CURRENT ADDR TO START OF TXBUF
MOV TXPTR, CPTR ;SETUP CURRENT POINTER TABLE POINTER FOR TXBUF
JSR PC, BLDBUF ;GO BUILD POINTER TABLE AND BUFFER
MOV #1, TXMTOT ;BUMP TOTAL MESSAGE COUNT

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 120
PROGRAM SETUP SECTION

```

4847
4848 026534 013737 006444 006524      MOV      CMPPTR,CPTR      ;SET UP START OF COMPARE POINTER TABLE
4849 026542 013737 006452 006526      MOV      CCURAD,CURADD   ;SET UP CURRENT ADDR. TO START OF CMPBUF
4850 026550 012737 000005 006516      MOV      #5,MSGTYP
4851 026556 013737 002162 006520      MOV      MSGSC,CURCC
4852 026564 004737 023364              JSR      PC,BLDBUF       ;PUT DEFAULT MESSAGE INTO CMPBUF
4853 026570 012737 000001 006446      MOV      #1,CMPTOT      ;BUMP THE COMP MSG COUNT
4854 026576 012737 000003 006570      MOV      #ACT,MODTYP    ;SET DEFAULT MODE= ACTIVE
4855 026604 005037 006572              CLR      MLTYP          ;SET DEFAULT MAINTENANCE LOOP MODE =NONE
4856 026610 012737 000001 006600      MOV      #1,RPASS      ;SET UP DEFAULT 'RUN PASS' COUNT TO 1
4857 026616 012737 000002 006576      MOV      #2,PARAM      ;SET UP PROG. PARAMETERS - DATACHECKING ENABLED
4858                                     ;
4859                                     PRINTF  #HLP0              ;
4860 026624 012746 012213              MOV      #HLP0,-(SP)
4861 026630 012746 000001              MOV      #1,-(SP)
4862 026634 010600                      MOV      SP,R0
4863 026636 104417                      TRAP     C$PNTF
4864 026640 062706 000004              ADD      #4,SP
4865 026644              GTRAS: SETVEC  INVEC,#DVRXI,#PRI05 ;DEFAULT NON-PROTOCOL RX INTERRUPT ROUTINE
4866 026644 012746 000240              MOV      #PRI05,-(SP)
4867 026650 012746 035556              MOV      #DVRXI,-(SP)
4868 026654 013746 011462              MOV      INVEC,-(SP)
4869 026660 012746 000003              MOV      #3,-(SP)
4870 026664 104437                      TRAP     C$SVEC
4871 026666 062706 000010              ADD      #10,SP
4872 026672 042737 000300 006576      BIC      #PRORUN!ABORT,PARAM ;INIT PROTOCOL VARIABLES
4873 026700 013737 006570 007704      MOV      MODTYP,DEV1
4874 026706 013737 006572 007706      MOV      MLTYP,DEV2
4875 026714 013737 006600 007710      MOV      RPASS,DEV3
4876 026722 013737 006576 007712      MOV      PARAM,DEV4
4877 026730 004737 023714              JSR      PC,SHWOP       ;PRINT TO OPERATOR THE CURRENT MODE.....
4878
4879 026734              MANUAL              ;SEE IF MANUAL INTERVENTION ALLOWED
4880 026734 104450                      TRAP     C$MANI
4881 026736              BCOMPLETE      GETCL ; BR IF YES (UAM=0 AND NOT CHAINED)
4882 026736 103412                      BCS      GETCL
4883 026740 005737 006600              TST      RPASS          ;SEE IF THIS IS FIRST 'DCLT PASS'
4884 026744 001002                      BNE      1$             ; BR IF NOT COMPLETED 1 PASS
4885 026746                      EXIT      TST              ; IF DONE 1 PASS IN UNATTENDED MODE - EXIT
4886 026746 104432                      TRAP     C$EXIT
4887 026750 017254                      .WORD    L10017-.
4888 026752 012737 000001 006572 1$:  MOV      #TTL,MLTYP      ;SET UP DEFAULT FOR UNATTENDED MODE
4889 026760 000137 031736              JMP      GTR9          ; 'R M=ACT/LC-1/PAS=1/NOST/CH' AND RUN
4890
4891              .SBTTL      COMMAND LINE FETCH & INTERPRETATION SECTION
4892
4893 026764 105037 003147              GETCL: CLRB      P$GDBD      ;CLEAR CMD LINE PARSING ERROR FLAGS
4894 026770 105037 003146              CLRB      P$NNUF
4895 026774              GMANID      CLISPM,CMDBUF,A,-1,1,72.,NO ;GET A COMMAND LINE FROM OPR.
4896 026774 104443                      TRAP     C$GMAN
4897 026776 000406                      BR        10000$
4898 027000 002666                      .WORD    CMDBUF
4899 027002 000142                      .WORD    T$CODE
4900 027004 011570                      .WORD    CLISPM
4901 027006 177,77                      .WORD    -1
4902 027010 000001                      .WORD    T$LOLIM

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 121
COMMAND LINE FETCH & INTERPRETATION SECTION

```

4903 027012 000110
4904 027014
4905 027014 012737 002666 003132      MOV    #CMDBUF,PSBUFA
4906 027022 012737 007714 003134      MOV    #CLITRE,PSTREE
4907 027030 012737 027740 003136      MOV    #CLIACT,PSACT
4908 027036 005037 003012      CLR    QUALFG          ;CLEAR QUALIFIER FLAG LOCATION
4909 027042 004737 024246      JSR    PC,PSTRV        ;GO PARSE COMMAND LINE
4910 027046 105737 003147      TSTB   PSGDBD         ;SEE IF PARSED OK OR AN ERROR
4911 027052 001412      BEQ    1$
4912 027054      PRINTF  #CLIERM
4913 027054 012746 011603
4914 027060 012746 000001
4915 027064 010600
4916 027066 104417
4917 027070 062706 000004
4918 027074 000137 026764
4919 027100 105737 003146      1$:    JMP    GETCL
4920 027104 001412      TSTB   PSNUF
4921 027106      BEQ    10$
4922 027106 012746 011633      PRINTF #CLINUF
4923 027112 012746 000001
4924 027116 010600
4925 027120 104417
4926 027122 062706 000004
4927 027126 000137 026764      JMP    GETCL
4928
4929 027132 023727 003010 000060 10$:  CMP    KEYWD1,#SETET    ;WAS 'SET EXPECT=TRANSMIT' TYPED ?
4930 027140 001711      BEQ    GETCL          ;YES,BRANCH
4931 027142 023727 003010 000005      CMP    KEYWD1,#HLP    ;SEE IF HELP WAS TYPED
4932 027150 001705      BEQ    GETCL          ;GO GET CMD AGAIN IF YES
4933 027152 023727 003010 000055      CMP    KEYWD1,#PRNT   ;SEE IF PRINT WAS TYPED
4934 027160 001701      BEQ    GETCL          ;GO GET CMD AGAIN IF YES
4935 027162 023727 003010 000004      CMP    KEYWD1,#RUN    ;SEE IF RUN WAS TYPED
4936 027170 001002      BNE    11$           ;BR IF NO
4937 027172 000137 031736      JMP    GTR9          ;START EXEC. IF YES
4938 027176 023727 003010 000052 11$:  CMP    KEYWD1,#DMPS    ;SEE IF DUMP WAS TYPED
4939 027204 001004      BNE    12$           ;BR IF NO
4940 027206 004737 023130      JSR    PC,DUMPSR     ;ELSE, DUMP PART OF MEMORY
4941 027212 000137 026764      JMP    GETCL         ;THEN RETURN TO GET ANOTHER CMD.
4942 027216 023727 003010 000057 12$:  CMP    KEYWD1,#EXIT    ;EXIT ?
4943 027224 001005      BNE    13$           ;NO,BRANCH
4944 027226 012737 000001 006564      MOV    #1,DCLFLG     ;SET CLEANUP FLAG
4945 027234      EXIT    TST        ;GO BACK TO INIT
4946 027234 104432
4947 027236 016766
4948
4949 027240 023727 003010 000001 13$:  CMP    KEYWD1,#CLEAR   ;SEE IF CLEAR WAS TYPED
4950 027246 001646      BEQ    GETCL          ;IF YES, BACK TO GET ANOTHER CMD.
4951 027250 023727 003010 000002      CMP    KEYWD1,#SHOW   ;SEE IF SHOW WAS TYPED
4952 027256 001642      BEQ    GETCL          ;IF YES, BACK TO GET ANOTHER CMD.
4953 027260 023727 003010 000010 4$:    CMP    KEYWD1,#SETEXP  ;SEE IF SET EXPECTED
4954 027266 001512      BEQ    2$            ;BR IF YES (A SETEXP WAS TYPED)
4955 027270 013737 006464 006530 5$:    MOV    TOTCC,TOTCC
4956 027276 023727 006530 001000      CMP    TOTCC,#BUFLIM ;SEE IF BUFFER ALREADY FULL
4957 027304 002414      BLT    15$           ;BR IF NOT FULL (BUFLIM # OF CHARS.)
4958 027306      PRINTF #MSGTRN,#BUFEX ;ELSE TELL OPR. AND DON'T BUILD MSG.

```

10000\$: .WORD TSHILIM

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

;SEE IF INCOMPLETE COMMAND TYPED

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

TRAP C\$EXIT
.WORD L10017-

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 122
COMMAND LINE FETCH & INTERPRETATION SECTION

4959	027306	012746	014355				MOV	#BUFEX,-(SP)
4960	027312	012746	014373				MOV	#MSGTRN,-(SP)
4961	027316	012746	000002				MOV	#2,-(SP)
4962	027322	010600					MOV	SP,R0
4963	027324	104417					TRAP	C\$PNTF
4964	027326	062706	000006				ADD	#6,SP
4965	027332	000137	026764			JMP	GETCL	: THEN GO GET A NEW COMMAND
4966	027336	005737	006464	15\$:		TST	TTOTCC	:IF FIRST 'SET' THEN GET RID OF DEFAULT
4967	027342	001002				BNE	6\$	
4968	027344	005037	006462			CLR	TXMTOT	
4969	027350	012737	006150	006442	6\$:	MOV	#PTRTAB, TXPTR	:GET POSITION OF END OF TX LIST
4970	027356	013701	006462			MOV	TXMTOT, R1	
4971	027362	020127	000017			CMP	R1, #MSGGLIM	:SEE IF MSG COUNT EXCEEDED.
4972	027366	002414				BLT	17\$	: BR IF NO
4973	027370					PRINTF	#MSGTRN, #TABEX	: ELSE TELL OPR. AND DON'T BUILD MSG.
4974	027370	012746	014315				MOV	#TABEX,-(SP)
4975	027374	012746	014373				MOV	#MSGTRN,-(SP)
4976	027400	012746	000002				MOV	#2,-(SP)
4977	027404	010600					MOV	SP,R0
4978	027406	104417					TRAP	C\$PNTF
4979	027410	062706	000006				ADD	#6,SP
4980	027414	000137	026764			JMP	GETCL	: THEN GO GET A NEW COMMAND.
4981	027420	006301			17\$:	ASL	R1	:# OF MSGS *4 = NEXT FREE PTR BLOCK
4982	027422	006301				ASL	R1	
4983	027424	060137	006442			ADD	R1, TXPTR	
4984	027430	013737	006442	006524		MOV	TXPTR, CPTR	:SETUP CHAR. COUNT, CURRENT ADDR, & PTR
4985	027436	013737	006466	006526		MOV	TCURAD, CURADD	
4986	027444	004737	023266			JSR	PC, ADDCC	:ADD IN CHAR. COUNT AND CHECK TOTAL
4987	027450	004737	023364			JSR	PC, BLDBUF	:GO BUILD MESSAGE IN BUFFER AND PTRS.
4988	027454	013737	006524	006442		MOV	CPTR, TXPTR	
4989	027462	013737	006530	006464		MOV	TOTCC, TTOTCC	:UPDATE CHAR. COUNT, CURR ADDR, & PTR
4990	027470	013737	006526	006466		MOV	CURADD, TCURAD	
4991	027476	005237	006462			INC	TXMTOT	
4992	027502	005337	003014			DEC	QUALVL	:DEC THE COPY COUNT
4993	027506	001270				BNE	5\$	
4994	027510	000137	026764			JMP	GETCL	
4995								
4996	027514	013737	006450	006530	2\$:	MOV	CTOTCC, TOTCC	:SETUP CHAR. COUNT, CURR. ADDR, & PTR
4997	027522	023727	006530	001000		CMP	TOTCC, #BUFLIM	:SEE IF BUFFER ALREADY FULL
4998	027530	002414				BLT	16\$	: BR IF NOT FULL (BUFLIM # OF CHARS.)
4999	027532					PRINTF	#MSGTRN, #BUFEX	: ELSE TELL OPR. AND DON'T BUILD MSG.
5000	027532	012746	014355				MOV	#BUFEX,-(SP)
5001	027536	012746	014373				MOV	#MSGTRN,-(SP)
5002	027542	012746	000002				MOV	#2,-(SP)
5003	027546	010600					MOV	SP,R0
5004	027550	104417					TRAP	C\$PNTF
5005	027552	062706	000006				ADD	#6,SP
5006	027556	000137	026764			JMP	GETCL	: THEN GO GET A NEW COMMAND
5007	027562	005737	006450		16\$:	TST	CTOTCC	:IF FIRST 'SET' THEN GET RID OF DEFAULT
5008	027566	001002				BNE	7\$	
5009	027570	005037	006446			CLR	CMPTOT	
5010	027574				7\$:			
5011	027574	012737	006244	006444		MOV	#PTR13, CMPPTR	:INIT COMPARE MESSAGE POINTER
5012	027602	013701	006446			MOV	CMPTOT, R1	
5013	027606	020127	000017			CMP	R1, #MSGGLIM	:SEE IF MSG COUNT EXCEEDED.
5014	027612	002414				BLT	18\$	: BR IF NO

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 123
COMMAND LINE FETCH & INTERPRETATION SECTION

5015	027614				PRINTF	#MSGTRN,#TABEX		; ELSE TELL OPR. AND DON'T BUILD MSG.
5016	027614	012746	014315					MOV #TABEX,-(SP)
5017	027620	012746	014373					MOV #MSGTRN,-(SP)
5018	027624	012746	000002					MOV #2,-(SP)
5019	027630	010600						MOV SP,R0
5020	027632	104417						TRAP C\$PNTF
5021	027634	062706	000006					ADD #6,SP
5022	027640	000137	026764					; THEN GO GET A NEW COMMAND.
5023	027644	006301		18\$:	JMP	GETCL		;# OF MSGS *4 = NEXT FREE PTR BLOCK
5024	027646	006301			ASL	R1		
5025	027650	060137	006444		ASL	R1		
5026	027654	013737	006444	006524	ADD	R1,CMPPTR		
5027	027662	013737	006452	006526	MOV	CMPPTR,CPTR		
5028	027670	004737	023266		MOV	CCURAD,CURADD		
5029	027674	004737	023364		JSR	PC,ADDCC		;ADD IN XHAR. COUNT AND CHECK TOTAL
5030	027700	013737	006524	006444	JSR	PC,BLDBUF		
5031	027706	005237	006446		MOV	CPTR,CMPPTR		
5032	027712	013737	006526	006452	INC	CMPTOT		;UPDATE CHAR. COUNT, CURR ADDR. & PTR
5033	027720	013737	006530	006450	MOV	CURADD,CCURAD		
5034	027726	005337	003014		MOV	TOTCC,CTOTCC		
5035	027732	001270			DEC	QUALVL		;IF COPY WAS GIVEN, PUT MSG IN BUFF
5036	027734	000137	026764		BNE	2\$		; AGAIN
5037					JMP	GETCL		;GO BACK UNTIL GET A 'RUN'
5038								
5039								
5040								
5041								

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 124
COMMAND LINE FETCH & INTERPRETATION SECTION

5042					
5043					
5044					
5045					
5046	027740				
5047	027740	006302			
5048	027742	016202	027756		
5049	027746	062702	027756		
5050	027752	004712			
5051	027754	000207			
5052					
5053					
5054	027756	000150			
5055	027760	000152			
5056	027762	000162			
5057	027764	001550			
5058	027766	000262			
5059	027770	000172			
5060	027772	000306			
5061	027774	000400			
5062	027776	000722			
5063	030000	000732			
5064	030002	000750			
5065	030004	000760			
5066	030006	000770			
5067	030010	001062			
5068	030012	001556			
5069	030014	001102			
5070	030016	001162			
5071	030020	001170			
5072	030022	001200			
5073	030024	001210			
5074	030026	001220			
5075	030030	001230			
5076	030032	001246			
5077	030034	001334			
5078	030036	001344			
5079	030040	001364			
5080	030042	001372			
5081	030044	001402			
5082	030046	001412			
5083	030050	001422			
5084	030052	001450			
5085	030054	001460			
5086	030056	001564			
5087	030060	001600			
5088	030062	001632			
5089	030064	001642			
5090	030066	001652			
5091	030070	001662			
5092	030072	001672			
5093	030074	001702			
5094	030076	000142			
5095	030100	001140			
5096	030102	000656			
5097	030104	000706			

.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) MODE				
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:		
			;BRIEF DESCRIPTION OF ACTONS TAKEN		
10\$:	.WORD	ACTNUL-10\$	;NULL		
	.WORD	ACTCLR-10\$	;CLEAR		
	.WORD	ACTSHO-10\$	;SHOW		
	.WORD	ACTCHK-10\$	;CHECK		
	.WORD	ACTRUN-10\$	;RUN		
	.WORD	ACTHLP-10\$	;HELP		
	.WORD	ACTCSE-10\$	;CLEAR OR SHOW EXPECT		
	.WORD	ACTCST-10\$	;CLEAR OR SHOW TRANSMIT		
	.WORD	ACTSTE-10\$	;SET EXPECT		
	.WORD	ACTSTT-10\$	;SET TRANSMIT		
	.WORD	ACTSZE-10\$	;SIZE		
	.WORD	ACTCOP-10\$	;COPY		
	.WORD	ACTNUM-10\$	;NUMERIC VALUE FOR SIZE OR COPY		
	.WORD	ACTOPM-10\$	;QUOTED MESSAGE FROM USER		
	.WORD	ACTSTS-10\$	;STATUS		
	.WORD	ACTEQO-10\$	;END OF QUOTED MESSAGE FROM USER		
	.WORD	ACTMSO-10\$	;ONES AS DATA		
	.WORD	ACTMS1-10\$	;ZEROS AS DATA		
	.WORD	ACTMS2-10\$	;1ALT AS DATA		
	.WORD	ACTMS3-10\$	;OACT AS DATA		
	.WORD	ACTMS4-10\$	;ITEP AS DATA		
	.WORD	ACTMS5-10\$	;CCITT AS DATA		
	.WORD	ACTMS6-10\$	;ALPHA AS DATA		
	.WORD	ACTATV-10\$	;ACTIVE MODE		
	.WORD	ACTPAS-10\$	;PASSIVE MODE		
	.WORD	ACTREC-10\$	;RECEIVE MODE		
	.WORD	ACTLIS-10\$	;LISTEN MODE		
	.WORD	ACTDLL-10\$	;DOWNLINE LOAD		
	.WORD	ACTTRA-10\$	;TRANSMIT MODE		
	.WORD	ACTTAL-10\$	;TALK MODE		
	.WORD	ACTNO-10\$	;NO IE /NOCHECK		
	.WORD	ACTECH-10\$	;ECHO		
	.WORD	ACTCRC-10\$	;CRC		
	.WORD	ACTPRO-10\$	;PROTOCOL		
	.WORD	ACTRPS-10\$	;STATUS		
	.WORD	ACTMOP-10\$	;SATELLITE IN MAINTENANCE LOOP MODE		
	.WORD	ACTTLP-10\$	;INTERNALTL		
	.WORD	ACTCLP-10\$	;CABLE LOOP		
	.WORD	ACTLLP-10\$	;LOCAL MODEM LOOP		
	.WORD	ACTRLP-10\$	;REMOTE MODEM LOOP		
	.WORD	ACTNUF-10\$	;MORE COMMAND NEEDED		
	.WORD	ACTBCR-10\$	;BAD CHARACTER IN OPERATOR MESSAGE		
	.WORD	ACTDMS-10\$	;DUMP MEMORY START ADDRESS		
	.WORD	ACTDME-10\$	;DUMP MEMORY END ADDRESS		

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 125
ACTION TABLE AND ROUTINES

5098	030106	000700	.WORD	ACTDMQ-10\$	:DUMP WORD
5099	030110	000246	.WORD	ACTPRT-10\$	:PRINT
5100	030112	001572	.WORD	ACTMOS-10\$	:MODEM STATUS
5101	030114	000236	.WORD	ACTEXT-10\$	:EXIT ROUTINE
5102	030116	001272	.WORD	ACTSEX-10\$	:SET EX=TR
5103					

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 126
ACTION TABLE AND ROUTINES

5104											
5105	030120	112737	177777	003146	ACTNUF: MOV	#-1,PSNNUF				:SET FLAG TO SAY NEED MORE OF COMMAND	
5106	030126	000207			ACTNUL: RTS	PC				:RETURN TO PARSER	
5107											
5108	030130	012737	000001	003010	ACTCLR: MOV	#CLEAR,KEYWD1				:SET LOC TO SAY A CLEAR WAS TYPED	
5109	030136	000207			RTS	PC					
5110											
5111	030140	012737	000002	003010	ACTSHO: MOV	#SHOW,KEYWD1				:SET LOC. TO SAY A SHOW WAS TYPED	
5112	030146	000207			RTS	PC					
5113											
5114	030150	012702	003016		ACTHLP: MOV	#HLP,KEYWD1				:SETUP R2 AS A POINTER TO HELP MSG TABLE	
5115	030154				1\$: PRINTF	#HLPF,(R2)+				:PRINT HELP INFORMATION MESSAGES	
5116	030154	012246									MOV (R2)+,-(SP)
5117	030156	012746	012271								MOV #HLPF,-(SP)
5118	030162	012746	000002								MOV #2,-(SP)
5119	030166	010600									MOV SP,R0
5120	030170	104417									TRAP C\$PNTF
5121	030172	062706	000006								ADD #6,SP
5122	030176	020227	003036		CMP	R2,#HLPEND				:SEE IF ALL INFO PRINTED YET	
5123	030202	001364			BNE	1\$				:IF NO KEEP PRINTING	
5124	030204	012737	000005	003010	MOV	#HLP,KEYWD1				:SET LOC. TO SAY A HELP WAS TYPED	
5125	030212	000207			RTS	PC					
5126											
5127	030214	012737	000057	003010	ACTEXT: MOV	#EXIT,KEYWD1				:EXIT COMMAND WAS INPUT	
5128	030222	000207			RTS	PC				:RETURN	
5129											
5130	030224	012737	000055	003010	ACTPRT: MOV	#PRNT,KEYWD1				:SET LOC. TO SAY A HELP WAS TYPED	
5131	030232	004737	020612		JSR	PC,REPORT				:CALL ROUTINE TO PRINT EVENT LOG AND BASE TABLE	
5132	030236	000207			RTS	PC					
5133											
5134	030240	012737	000004	003010	ACTRUN: MOV	#RUN,KEYWD1				:SET RUN FLAG	
5135	030246	112737	177777	003146	MOVB	#-1,PSNNUF				:SET FLAG TO SAY NEED MORE OF COMMAND	
5136	030254	012737	000001	006600	MOV	#1,RPASS				:SET DEFAULT RUN 'PASS' TO 1	
5137	030262	000207			RTS	PC					
5138											
5139	030264	012737	006244	006444	ACTCSE: MOV	#PTR13,CMPPTR				:INIT COMPARE MESSAGE POINTER	
5140	030272	013701	006444		MOV	CMPPTR,R1					
5141											
5142	030276	013702	006446		MOV	CMPTOT,R2					
5143	030302	105037	003146		CLRB	PSNNUF				:FLAG THAT HAVE VALID COMMAND AT THIS PT.	
5144	030306	023727	003010	000002	CMP	KEYWD1,#SHOW				:SEE IF A CLEAR OR SHOW WAS TYPED	
5145	030314	001471			BEQ	ACTSHW				:BR IF A SHOW WAS TYPED	
5146	030316	012737	000001	006446	MOV	#1,CMPTOT				:CLEAR COMPARE MESSAGE COUNT, CHAR. COUNT	
5147	030324	005037	006450		CLR	CTOTCC				: AND RESET POINTER	
5148											
5149	030330	012737	006244	006444	MOV	#PTR13,CMPPTR				:INIT COMPARE MESSAGE POINTER	
5150	030336	013737	006444	006524	MOV	CMPPTR,CPTR				:SET UP TO FILL IN DEFAULT MESSAGE	
5151	030344	012701	005150		MOV	#CMPBIJF,R1					
5152	030350	010137	006452		MOV	R1,CCURAD					
5153	030354	000431			BR	ACTCLB					
5154											
5155	030356	012701	006150		ACTCST: MOV	#PTRTAB,R1					
5156	030362	013702	006462		MOV	TXMTOT,R2					
5157	030366	105037	003146		CLRB	PSNNUF				:FLAG THAT HAVE VALID COMMAND AT THIS PT.	
5158	030372	023727	003010	000002	CMP	KEYWD1,#SHOW				:SEE IF A CLEAR OR SHOW WAS TYPED	
5159	030400	001437			BEQ	ACTSHW				:BR IF A SHOW WAS TYPED	

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 127
ACTION TABLE AND ROUTINES

5160	030402	012777	000001	006462	MOV	#1,TXMTOT	:CLEAR TRANSMIT MESSAGE COUNT, CHAR. COUNT
5161	030410	005037	006464		CLR	TTOTCC	: AND RESET POINTER
5162	030414	012737	006150	006442	MOV	#PTRTAB,TXPTR	
5163	030422	013737	006442	006524	MOV	TXPTR,CPTR	
5164	030430	012701	003150		MOV	#TXBUF,R1	
5165	030434	010137	006466		MOV	R1,TCURAD	
5166							
5167	030440	012702	001000		ACTCLB: MOV	#BUFLIM,R2	
5168	030444	010137	006526		MOV	R1,CURADD	:SET UP TO PUT DEFAULT MSG IN LIST AFTER 033'S
5169	030450	012737	000005	006516	MOV	#5,MSGTYP	
5170	030456	013737	002162	006520	MOV	MSG5C,CURCC	
5171	030464	105021			1\$: CLR	(R1)+	:FILL EXPT OR TRAN BUFFER WITH 0'S IF A CLEAR
5172	030466	005302			DEC	R2	:DO 'BUFLIM' NUMBER OF BYTE LOCATIONS
5173	030470	001375			BNE	1\$	
5174	030472	004737	023364		JSR	PC,BLDBUF	: 'CLEAR' REALLY MEANS TO PUT DEFAULT MSG IN
5175	030476	000207			RTS	PC	:WHEN DONE, RETURN TO PARSER
5176							
5177							
5178	030500	012705	003072		ACTSHW: MOV	#SHTAB,R5	
5179	030504	122571	000000		5\$: CMPB	(R5)+,0(R1)	:LOOK AT FIRST BYTE OF MSG TO DECIPHER TYPE
5180	030510	001404			BEQ	6\$	
5181	030512	020527	003101		CMP	R5,#SHTEND	:SEE IF LOOKED AT ALL OF DEFAULTS YET
5182	030516	001372			BNE	5\$	
5183	030520	005205			INC	R5	:MUST BE OPR. SPEC'D THEN
5184	030522	162705	003073		6\$: SUB	#SHTAB+1,R5	
5185	030526	006305			ASL	R5	
5186	030530	016137	000002	006534	MOV	2(R1),TEMP	
5187	030536				PRINTF	#SHMSG,SHTYTB(R5),TEMP	:PRINT MSG SIZE & TYPE
5188	030536	013746	006534				MOV TEMP,-(SP)
5189	030542	016546	003052				MOV SHTYTB(R5),-(SP)
5190	030546	012746	013516				MOV #SHMSG,-(SP)
5191	030552	012746	000003				MOV #3,-(SP)
5192	030556	010600					MOV SP,R0
5193	030560	104417					TRAP C\$PNTF
5194	030562	062706	000010				ADD #10,SP
5195	030566	062701	000004				
5196	030572	005302			ADD	#4,R1	:BUMP R1 TO NEXT SET OF POINTERS
5197	030574	001341			DEC	R2	
5198	030576	013737	006570	007704	BNE	ACTSHW	
5199	030604	013737	006572	007706	MOV	MODTYP,DEV1	
5200	030612	013737	006600	007710	MOV	MLTYP,DEV2	
5201	030620	013737	006576	007712	MOV	RPASS,DEV3	
5202	030626	004737	023714		MOV	PARAM,DEV4	
5203	030632	000207			JSR	PC,SHWOP	:SHOW THE OPERATOR THE CURRENT MODE..... ALSO
5204					RTS	PC	
5205	030634	013737	003142	006510	ACTDMS: MOV	P\$NUM,STADD	:SETUP STARTING ADDRESS FOR DUMP
5206	030642	005037	006514		CLR	BYTBIT	:SET DEFAULT OF WORD DUMP
5207	030646	012737	000052	003010	MOV	#DMP\$KEYWD1	:FLAG THAT A DUMP WAS TYPED
5208	030654	000403			BR	ACTDME	
5209							
5210	030656	012737	177777	006514	ACTDMQ: MOV	#-1,BYTBIT	:SET DUMP FLAG TO 'DUMP-WORD'
5211	030664	013737	003142	006512	ACTDME: MOV	P\$NUM,ENADD	:SETUP END ADDRESS FOR DUMP (=START IF NO 'EEE'
5212	030672	105037	003146		ACTDMX: CLR	P\$NUF	:CLEAR NOT-ENOUGH FLAG, 'DUMP N-N/B' IS VALID
5213	030676	000207			RTS	PC	
5214							

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 128
ACTION TABLE AND ROUTINES

```

5215
5216
5217
5218 030700 012737 000010 003010 ACTSTE: MOV #SETEXP,KEYWD1
5219 030706 000403 BR ACTSTX
5220
5221 030710 012737 000011 003010 ACTSTT: MOV #SETTRN,KEYWD1
5222 030716 012737 000001 003014 ACTSTX: MOV #1,QUALVL ;SET UP DEFAULT COPY TO 1 (/COPY=0)
5223 030724 000207 RTS PC
5224
5225 030726 012737 000012 003012 ACTSIZE: MOV #SIZE,QUALFG
5226 030734 000207 RTS PC
5227
5228 030736 012737 000013 003012 ACTCOP: MOV #QCOPY,QUALFG
5229 030744 000207 RTS PC
5230
5231 030746 023727 003012 000012 ACTNUM: CMP QUALFG,#SIZE ;SEE IF A SIZE OR COPY TYPED
5232 030754 001023 BNE 1$ ;BR IF IT WAS A COPY
5233 030756 005737 003142 TST PSNUM ;CHECK TO BE SURE DIDN'T TRY SIZE=0
5234 030762 001014 BNE 3$ ; BR IF NO
5235 030764 PRINTF #CLISEO
5236 030764 012746 012072 MOV #CLISEO,-(SP)
5237 030770 012746 000001 MOV #1,-(SP)
5238 030774 010600 MOV SP,R0
5239 030776 104417 TRAP C$PNTF
5240 031000 062706 000004 ADD #4,SP
5241 031004 112737 177777 003147 MOVB #-1,PSGDBD ;SEE ERROR-IN-CMD FLAG
5242 031012 000411 BR 2$
5243 031014 013737 003142 006520 3$: MOV PSNUM,CURCC ;IF A SIZE LOAD CURCC WITH BYTE COUNT
5244 031022 000405 BR 2$
5245 031024 013737 003142 003014 1$: MOV PSNUM,QUALVL ;IF A COPY, LOAD COPY COUNT
5246 031032 005237 003014 INC QUALVL ;INCREMENT SO FIRST DEC MAKES IT REAL #
5247 031036 000522 2$: BR ACTMEX
5248
5249 031040 012737 000007 006516 ACTOPM: MOV #7,MSGTYP
5250 031046 010437 006534 MOV R4,TEMP ;KEEP TRACK OF START OF QUOTED TEXT
5251 031052 005237 006534 INC TEMP ; SO CAN CALC OPCNT AT END OF QUOTES
5252 031056 000207 RTS PC
5253
5254 031060 010402 ACTEQO: MOV R4,R2
5255 031062 163702 006534 SUB TEMP,R2
5256 031066 010237 006520 MOV R2,CURCC ;CALC BYTE COUNT FOR QUOTED TEXT
5257 031072 010237 002166 MOV R2,OPCNT
5258 031076 013701 006534 MOV TEMP,R1
5259 031102 012705 002520 MOV #OPBUF,R5
5260 031106 112125 1$: MOVB (R1)+,(R5)+ ;COPY QUOTED TEXT TO OPBUF
5261 031110 005302 DEC R2
5262 031112 001375 BNE 1$
5263 031114 000473 BR ACTMEX
5264
5265 031116 ACTBCR: PRINTF #CLIBCR ;BAD CHAR. IN OPR. QUOTED STRING
5266 031116 012746 012025 MOV #CLIBCR,-(SP)
5267 031122 012746 000001 MOV #1,-(SP)
5268 031126 010600 MOV SP,R0
5269 031130 104417 TRAP C$PNTF
5270 031132 062706 000004 ADD #4,SP

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 129
ACTION TABLE AND ROUTINES

					RTS	PC	
5271	031136	000207					
5272							
5273	031140	005037	006516		ACTMS0:	CLR	MSGTYP
5274	031144	000435				BR	ACTME1
5275	031146	012737	000001	006516	ACTMS1:	MOV	#1,MSGTYP
5276	031154	000431				BR	ACTME1
5277	031156	012737	000002	006516	ACTMS2:	MOV	#2,MSGTYP
5278	031164	000425				BR	ACTME1
5279	031166	012737	000003	006516	ACTMS3:	MOV	#3,MSGTYP
5280	031174	000421				BR	ACTME1
5281	031176	012737	000004	006516	ACTMS4:	MOV	#4,MSGTYP
5282	031204	000415				BR	ACTME1
5283	031206	012737	000005	006516	ACTMS5:	MOV	#5,MSGTYP
5284	031214	013737	002162	006520		MOV	MSG5C,CURCC
5285	031222	000430				BR	ACTMEX
5286	031224	012737	000006	006516	ACTMS6:	MOV	#6,MSGTYP
5287	031232	013737	002164	006520		MOV	MSG6C,CURCC
5288							
5289	031240	012737	000100	006520	ACTME1:	MOV	#64.,CURCC
5290	031246	000416				BR	ACTMEX
5291							
5292	031250	022737	000010	003010	ACTSEX:	CMP	#SETEXP,KEYWD1
5293	031256	001404				BEQ	10\$
5294	031260	112737	000001	003147		MOVB	#1,PSGDBD
5295	031266	000406				BR	ACTMEX
5296	031270	004737	023510		10\$:	JSR	PC,FACSIMILE
5297	031274	012737	000060	003010		MOV	#SETET,KEYWD1
5298	031302	000400				BR	ACTMEX
5299							
5300	031304	105037	003146		ACTMEX:	CLRB	PSNNUF
5301	031310	000207				RTS	PC
5302							

;SET MESSAGE TYPE = ALL ONES
;SET MESSAGE TYPE = ONES & ZEROS
;SET MESSAGE TYPE = ZEROS & ONES
;SET MESSAGE TYPE = CCITT
;SET MESS TYPE = QUICK FOX
;SETUP DEFAULT SIZE FOR THIS TYPE
;SET MESSAGE TYPE = ALPHA/NUM
;SETUP DEFAULT SIZE FOR THIS TYPE
;SETUP DEFAULT SIZE FOR MSG0-4
;BRANCH TO EXIT
;DID WE GET HERE FROM 'SET E='COMMAND?
;YES,BRANCH
;SET ERROR FLAG
;GO EXIT SUBROUTINE
;GO COPY TRANSMIT LIST TO EXPECT LIST
;SET FLAG TO BE USED IN T1::
;EXIT SUBROUTINE
;CLEAR NOT-ENOUGH FLAG

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 130
 CZDCLB.P11 19-JUL-83 17:12 ACTION TABLE AND ROUTINES

5303	031312	012737	000003	006570	ACTATV: MOV	#ACT,MODTYP	;MODE = ACTIVE
5304	031320	000432			BR	ACTM2X	
5305							
5306	031322	012737	000002	006570	ACTPAS: MOV	#PAS,MODTYP	;MODE = PASSIVE
5307	031330	105037	003146		CLRB	P\$NNUF	;CLEAR NOT-ENOUGH FLAG
5308	031334	005037	006572		CLR	MLTYP	;CLEAR MAINT LOOP TYPE
5309	031340	000207			RTS	PC	
5310							
5311	031342	005037	006570		ACTREC: CLR	MODTYP	;MODE = RECEIVE
5312	031346	000417			BR	ACTM2X	
5313							
5314	031350	012737	000006	006570	ACTLIS: MOV	#LIS,MODTYP	;MODE = LISTEN
5315	031356	000413			BR	ACTM2X	
5316							
5317	031360	012737	000004	006570	ACTDLL: MOV	#DOW,MODTYP	;MODE = DOWNLINE LOAD
5318	031366	000407			BR	ACTM2X	
5319							
5320	031370	012737	000001	006570	ACTTRA: MOV	#TRA,MODTYP	;MODE = TRANSMIT
5321	031376	000403			BR	ACTM2X	
5322							
5323	031400	012737	000005	006570	ACTTAL: MOV	#TAL,MODTYP	;MODE = TALK
5324							
5325	031406	042737	000004	006576	ACTM2X: BIC	#ECHOB,PARAM	;DISABLE /ECHO (ALL BUT PASSIVE MODE)
5326	031414	105037	003146		CLRB	P\$NNUF	;CLEAR NOT-ENOUGH FLAG
5327	031420	005037	006572		CLR	MLTYP	;CLEAR MAINT LOOP TYPE
5328	031424	000207			RTS	PC	
5329							

CZDCLB DUP-11 DATA COMM. LINK TEST MACY11 30A(1052) 20-JUL-83 13:19 PAGE 131
 CZGCLB.P11 19-JUL-83 17:12 ACTION TABLE AND ROUTINES

5330	031426	012737	000036	003012	ACTNO:	MOV	#NO,QUALFG	
5331	031434	000207				RTS	PC	
5332								
5333	031436	022737	000036	003012	ACTECH:	CMP	#NO,QUALFG	
5334	031444	001422				BEQ	1\$	
5335	031446	052737	000004	006576		BIS	#ECHOB,PARAM	
5336	031454	022737	000002	006570		CMP	#PAS,MODTYP	
5337	031462	001416				BEQ	2\$	;BE SURE IN PASSIVE MODE IF
5338	031464					PRINTF	#CLINPS	;IF TRYING TO SET /ECHO
5339	031464	012746	011762					MOV #CLINPS,-(SP)
5340	031470	012746	000001					MOV #1,-(SP)
5341	031474	010600						MOV SP,R0
5342	031476	104417						TRAP C\$PNTF
5343	031500	062706	000004					ADD #4,SP
5344	031504	112737	177777	003147		MOVB	#-1,\$GDBD	
5345	031512	042737	000004	006576	1\$:	BIC	#ECHOB,PARAM	
5346	031520	005037	003012		2\$:	CLR	QUALFG	;CLEAR 'NO' OUT OF QUALIFIER FLAG
5347	031524	000501				BR	ACTLXX	
5348								
5349	031526	012701	000002		ACTCHK:	MOV	#DATCKB,R1	;SET DATA CHECK BIT
5350	031532	000413				BR	ACTQFG	
5351								
5352	031534	012701	000001		ACTSTS:	MOV	#STATB,R1	;SET THE STATUS BIT
5353	031540	000410				BR	ACTQFG	
5354								
5355	031542	012701	000020		ACTCRC:	MOV	#CRCB,R1	;SET THE CRC BIT
5356	031546	000405				BR	ACTWFG	
5357								
5358	031550	012701	000010		ACTMOS:	MOV	#MOCHK,R1	;SET THE MODEM BIT
5359	031554	000402				BR	ACTQFG	
5360								
5361	031556	012701	000040		ACTPRO:	MOV	#PROTOB,R1	;SET THE PROTOCOL BIT
5362								
5363	031562	050137	006576		ACTQFG:	BIS	R1,PARAM	
5364	031566	022737	000036	003012		CMP	#NO,QUALFG	
5365	031574	001002				BNE	1\$	
5366	031576	040137	006576			BIC	R1,PARAM	
5367	031602	005037	003012		1\$:	CLR	QUALFG	;CLEAR 'NO' OUT OF QUALIFIER FLAG
5368	031606	000450				BR	ACTLXX	
5369								
5370	031610	013737	003142	006600	ACTRPS:	MOV	\$NUM,RPASS	;GET NUMBER OF 'RUN PASSES'
5371	031616	000444				BR	ACTLXX	
5372								
5373	031620	012737	000005	006572	ACTMOP:	MOV	#5,MLTYP	
5374	031626	000417				BR	ACTLPX	
5375	031630	012737	000001	006572	ACTTLP:	MOV	#1,MLTYP	
5376	031636	000413				BR	ACTLPX	
5377	031640	012737	000002	006572	ACTCLP:	MOV	#2,MLTYP	
5378	031646	000407				BR	ACTLPX	
5379	031650	012737	000003	006572	ACTLLP:	MOV	#3,MLTYP	
5380	031656	000403				BR	ACTLPX	
5381	031660	012737	000004	006572	ACTRLP:	MOV	#4,MLTYP	
5382								
5383	031666	022737	000003	006570	ACTLPX:	CMP	#ACT,MODTYP	;BE SURE IN ACTIVE IF TRYING TO SET LOOP
5384	031674	001415				BEQ	ACTLXX	; BR IF IN ACTIVE
5385	031676	112737	177777	003147		MOVB	#-1,\$GDBD	

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 132
ACTION TABLE AND ROUTINES

5386 031704 005037 006572
5387 031710
5388 031710 012746 011720
5389 031714 012746 000001
5390 031720 010600
5391 031722 104417
5392 031724 062706 000004
5393 031730 105037 003146
5394 031734 000207
5395

CLR MLTYP
PRINTF #CLIBDL

;CLEAR ANY LOOP TYPE THAT MAY HAVE GOT SET

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

ACTLXX: CLRB PSNNUF
RTS PC

;CLEAR NOT-ENOUGH FLAG

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 133
ACTION TABLE AND ROUTINES

```

5396
5397
5398 031736 005737 006572      GTR9:  TST      MLTYP      ;LOOP MODE ?
5399 031742 001422              BEQ      10$      ;NO,BRANCH
5400 031744 032737 000002 006576  BIT      #DATCKB,PARAM ;DATA CHECK ?
5401 031752 001416              BEQ      10$      ;NO,BRANCH
5402 031754 023737 006446 006462  CMP      CMPTOT,TXMTOT ;TX = EX ?
5403 031762 001412              BEQ      10$      ;YES,BRANCH
5404 031764              PRINTF  #CLIPW      ;PRINT WARNING
5405 031764 012746 012123              MOV      #CLIPW,-(SP)
5406 031770 012746 000001              MOV      #1,-(SP)
5407 031774 010600              MOV      SP,R0
5408 031776 104417              TRAP     C$PNTF
5409 032000 062706 000004              ADD      #4,SP
5410 032004 000137 026764              JMP      GETCL      ;TRY AGAIN
5411
5412
5413      ; RX ALLOCATE CODE
5414 032010 012737 006150 006442 10$:  MOV      #PTRTAB,TXPTR ;INIT TRANSMIT MESSAGE POINTER
5415 032016 012737 006244 006444      MOV      #PTR13,CMPPTR ;INIT COMPARE MESSAGE POINTER
5416 032024 012737 006340 006440      MOV      #PTR23,RXPTR ;INIT RECEIVE MESSAGE POINTER
5417
5418 032032 013737 006446 006476      MOV      CMPTOT,RXMTOT ;MAKE COMPARE AND RX MESSAGE COUNTS EQUAL
5419
5420
5421 032040 005037 006602      GTREX:  CLR      FLAG      ;CLEAR FLAG
5422 032044 005037 006502      CLR      OPVAR      ;CLEAR OPTIONAL VARIABLE COUNTER
5423 032050 005037 006504      CLR      PSCNT      ;CLEAR PASS COUNT
5424 032054 005037 006506      CLR      ERRCNT      ;CLEAR ERROR COUNT
5425 032060 005037 011516      CLR      MGLCNT      ;CLEAR GLITCH COUNT
5426 032064 005037 011520      CLR      MHRCNT      ;CLEAR HARD ERR. COUNT
5427 032070 005037 006500      CLR      LNCNT      ;CLEAR LINE COUNTER
5428 032074 012737 000626 011506      MOV      #626,SYNCW ;SET UP SYNCW FOR 226 SYNC +TSOM
5429 032102 052737 000200 011470      BIS      #BIT7,DUPPAR ;RX SYNC=226
5430 032110 005737 011522      TST      RNODE
5431 032114 001406              BEQ      1$      ;IF NON ITEP GO TO 1
5432 032116 042737 000200 011470      BIC      #BIT7,DUPPAR ;SET UP FOR 26 SYNC WORD ON RX.
5433 032124 012737 000426 011506      MOV      #426,SYNCW ;ELSE SET UP SYNC FOR 26 AND TSOM
5434 032132 004737 020200      JSR      PC,LOGDVI ;LOG ABOUT TO INIT DEVICE
5435 032136 004737 034306      JSR      PC,DVINIT ;INIT DEVICE
5436
5437 032142 012737 001000 006520 GTRX2:  MOV      #BUFLIM,CURCC ;SET CHAR COUNT TO 'BUFLIM' NO. OF BYTES
5438 032150 012737 004150 006526      MOV      #RXBUF,CURADD ;SET UP RX BUFFER AS CURRENT ADD.
5439 032156 013737 006440 006524      MOV      RXPTR,CPTR
5440 032164 012737 000010 006516      MOV      #10,MSGTYP ;SET UP FOR 33 TO FILL RX BUFFERS
5441 032172 004737 023364      JSR      PC,BLDBUF ;CLEAR RX BUFFER
5442 032176 013702 006570      MOV      MODTYP,R2
5443 032202 006302      ASL      R2
5444 032204 000172 006604      JMP      @MODE(R2) ;MODE DISPATCH
5445

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 134
RECEIVE MODE SECTION

```

5446 .SBTTL          RECEIVE MODE SECTION
5447 :++
5448 : FUNCTIONAL DESCRIPTION:
5449 :   RECEIVE-ONLY (OR ONE-WAY-IN) ROUTINE
5450 :   IN THIS MODE OF TESTING THE DEVICE'S RECEIVER IS ENABLED IN EXPECTATION
5451 :   OF RECEIVING A MESSAGE. AFTER RECEIVING AN 'EXPECTED' NUMBER OF
5452 :   MESSAGES, THE DATA RECEIVED CAN BE COMPARED AGAINST A LIST OF 'EXPECT
5453 :   TO RECEIVE' MESSAGES IF DATA-CHECKING IS ENABLED.
5454 :
5455 : SUBORDINATE ROUTINES USED:
5456 :   'ALLTR'
5457 :
5458 : CALLING SEQUENCE:
5459 :   JMP      @MODE(R2)      ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
5460 :--
5461 :
5462 :RXONLY:
5463 :RXON2:  MOV      RXPTR,CPTRR
5464 :         MOV      RXMTOT,DVRCT      ;SET UP MESSAGE COUNT
5465 :         BIS      #QRX+#ERX,FLAG    ;SET UP RX QUE
5466 :         CLR      CPTR              ;CLEAR THE TX POINTER
5467 :         JMP      ALLTR              ;GO RX.
5468
032210
032210 013737 006440 006522
032216 013737 006476 006474
032224 052737 000104 006602
032232 005037 006524
032236 000137 032400

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 135
TRANSMIT MODE SECTION

```

5469 .SBTTL          TRANSMIT MODE SECTION
5470
5471 :++
5472 : FUNCTIONAL DESCRIPTION:
5473 : TRANSMIT-ONLY (OR ONE-WAY-OUT) ROUTINE
5474 : IN THIS MODE OF TESTING A LIST OF MESSAGES IS TRANSMITTED WITHOUT
5475 : EXPECTING ANY DATA TO BE RECEIVED. A REPETITION COUNT CAN BE
5476 : SPECIFIED TO REPETITIVELY TRANSMIT THE LIST.
5477
5478 : SUBORDINATE ROUTINES USED:
5479 : 'ALLTR'
5480
5481 : CALLING SEQUENCE:
5482 :     JMP      @MODE(R2)      ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
5483 : --
5484
5485 032242 042737 000002 006576 TXONLY: BIC      #DATCKB,PARAM ;SET NOCHECK
5486 032250 013737 006442 006524 TXON2:  MOV      TXPTR,CPTR
5487 032256 013737 006462 006460      MOV      TXMTOT,DVTCT ;COPY COUNTER FOR THIS PASS
5488 032264 052737 000210 006602      BIS      #QTX+#ETX,FLAG ;SET THE QUE TX FLAG
5489 032272 005037 006522      CLR      CPTR      ;CLEAR RX POINTER
5490 032276 000137 032400      JMP      ALLTR      ;GO TX.

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 136
PASSIVE MODE SECTION

```

5491 .SBTTL          PASSIVE MODE SECTION
5492
5493 :++
5494 : FUNCTIONAL DESCRIPTION:
5495 : PASSIVE MODE SECTION
5496 : IN THIS MODE OF TESTING, THE DEVICE'S RECEIVER IS ENABLED IN
5497 : EXPECTATION OF RECEIVING A MESSAGE. THEN EVERY TIME A MESSAGE IS
5498 : RECEIVED, A MESSAGE IS TRANSMITTED. DATA CHECKING CAN BE DONE ON THE
5499 : RECEIVED DATA.
5500
5501 : SUBORDINATE ROUTINES USED:
5502 :
5503 :         'ALLTR'
5504 :
5505 : CALLING SEQUENCE:
5506 :         JMP      @MODE(R2)          ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
5507 :--
5508
5509 032302          PLCK:
5510 032302 013737 006462 006460 PLCK2: MOV      TXMTOT,DVTCT      ;SET UP THE TRANSMIT COUNT
5511 032310 013737 006442 006524      MOV      TXPTR,CPTR      ;SET UP CPTR TO TRANSMIT POINTER
5512 032316 013737 006440 006522 PLCK3: MOV      RXPTR,CPTRR     ;SET UP CPTRR TO REC POINTER
5513 032324 052737 000104 006602      BIS      #QRX+#ERX,FLAG ;SET UP Q AND EXPECT RX
5514 032332 000137 032400          JMP      ALLTR          ;AND GO RX FIRST MSG.
5515

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 137
ACTIVE MODE SECTION

```

5516 .SBTTL          ACTIVE MODE SECTION
5517
5518 :++
5519 : FUNCTIONAL DESCRIPTION:
5520 :   ACTIVE MODE SECTION
5521 :   IN THIS MODE OF TESTING A LIST OF MESSAGES IS TRANSMITTED AND
5522 :   MESSAGES ARE EXPECTED TO BE RECEIVED.  RECEIVED DATA CAN BE COMPARED
5523 :   AGAINST 'EXPECTED' DATA IF DATA-CHECKING IS ENABLED.
5524 :   NOTE: IF BOTH ENDS OF THE LINK ARE IN ACTIVE MODE, THEN THE
5525 :   LINK MUST BE A FULL DUPLEX LINK!
5526
5527 : SUBORDINATE ROUTINES USED:
5528
5529 :   'ALLTR'
5530
5531 : CALLING SEQUENCE:
5532 :   JMP      @MODE(R2)      ;DISPATCH TO MODE BASED ON MODE TYPE IN R2
5533 :--
5534
5535 032336 013737 006462 006460 ALCK:  MOV    TXMTOT,DVTCT    ;# OF MESSAGES TO TRANSMIT(DEVICE TX COUNT)
5536 032344 013737 006442 006524      MOV    TXPTR,CPTR      ;SETUP TX MESSAGE LIST POINTER
5537 032352 013737 006476 006474      MOV    RXMTOT,DVRCT    ;# OF MESSAGES TO RECEIVE(DEVICE RX COUNT)
5538 032360 013737 006440 006522      MOV    RXPTR,CPTRR    ;SETUP RX MESSAGE LIST POINTER
5539 032366 052737 000314 006602      BIS    #QRX+#QTX+#ETX+#ERX,FLAG
5540 032374 000137 032400              JMP    ALLTR
5541
5542
5543

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 138
TRANSMIT - RECEIVE FOR ALL STANDARD MODES

```

5544 .SBTTL          TRANSMIT - RECEIVE FOR ALL STANDARD MODES
5545
5546
5547 : **
5548 : FUNCTIONAL DESCRIPTION:
5549 :   THIS CODE PERFORMS THE FOLLOWING FUNCTIONS
5550 :   1.) IF RX BUFFERS ARE TO BE QUEUED, TELL DEVICE
5551 :       CODE TO QUE THEM, LOG RECEIVE QUEUED.
5552 :   2.) IF TX BUFFERS ARE TO BE QUEUED, TELL DEVICE
5553 :       CODE TO QUE THEM, LOG TRANSMIT QUEUED.
5554 :   3.) WAIT FOR EITHER RECEIVE BUFFER OR TRANSMIT BUFFER OR
5555 :       BOTH TO COMPLETE
5556 :   4.) IF RECEIVE COMPLETE LOG IT UPDATE RX TABLE IF DATA
5557 :       CHECKING.
5558 :   5.) IF TRANSMIT COMPLETE LOG IT.
5559 :   6.) WHEN BOTH TRANSMIT AND RECEIVE LISTS ARE DONE
5560 :       GO TO THE COMPARE BUFFER CODE
5561
5562 : SUBORDINATE ROUTINES USED:
5563 :   'DVRXQ' - QUE RECEIVE BUFFER SPACE TO DEVICE
5564 :   'LOGRXQ' - LOG RECEIVE BUFFER SPACE TO EVENT LOG
5565 :   'LOGTXQ' - LOG TRANSMIT BUFFER QUEUED TO EVENT LOG
5566 :   'DVTXRX' - QUE TRANSMIT BUFFER AND WAIT FOR RX
5567 :               OR TX TO COMPLETE
5568 :   'LOGRXC' - LOG RECEIVE BUFFER COMPLETED TO EVENT LOG
5569 :   'LOGTXC' - LOG TRANSMIT BUFFER COMPLETED TO EVENT LOG
5570
5571 : USE OF FLAG BITS:
5572 :   QRX - SET ON INPUT TO ALLTR IF REC IS TO BE QUEUED TO
5573 :         DEVICE. CLEARED BY DVRXQ AND THEN SET BY DVTXRX
5574 :         WHEN RX BUFFER IS COMPLETED.
5575 :   QTX - SET ON INPUT TO ALLTR IF TRANSMIT IS TO BE QUEUED TO
5576 :         DEVICE. CLEARED ON ENTRY TO DVTXRX AND SET BY DVTXRX
5577 :         WHEN TX BUFFER IS COMPLETED.
5578 :   ETX - USED BY DVTXRX TO DETERMINE IF TX BUFFER COMPLETED IS
5579 :         EXPECTED.
5580 :   ERX - USED BY DVTXRX TO DETERMINE IF RX BUFFER COMPLETED IS
5581 :         EXPECTED.
5582
5583 : CALLING SEQUENCE:
5584 :   JMP      ALLTR          ;GO TO TRANSMIT-RECEIVE FOR ALL STANDARD MODES
5585 : --
5586
5587
5588 032400          ALLTR:
5589 032400 032737 000004 006602 ALCK5: BIT      #QRX,FLAG      ;QUE RX ADDRESS & COUNT? (ARE WE RECEIVING?)
5590 032406 001424          BEQ      ALCK1      ;NO, BRANCH ?
5591 032410 013702 006522          MOV      CPTRR,R2      ;GET CURRENT RX MESSAGE LIST POINTER (POINTS
5592 :                               ;TO RX MESSAGE LIST POINTER TABLE)
5593 032414 011237 006540          MOV      (R2),TEMP2      ;SAVE RX ADDRESS FOR LOG
5594 032420 012237 006470          MOV      (R2)+,DVRXA     ;DEVICE RX ADDRESS
5595 032424 011237 006542          MOV      (R2),TEMP3      ;SAVE RX CHAR COUNT FOR LOG
5596 032430 011237 006472          MOV      (R2),DVRCC     ;DEVICE RX CHARACTER COUNT
5597 032434 010237 006522          MOV      R2,CPTRR       ;STORE UPDATED RX POINTER
5598 032440 004737 020134          JSR      PC,LOGRXQ      ;LOG RECEIVER QUEUED
5599 032444 032737 000040 006576 10$: BIT      #PROTOB,PARAM ;'/PROTOCOL' ?

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 139
TRANSMIT - RECEIVE FOR ALL STANDARD MODES

5600	032452	001002			BNE	ALCK1	:YES,BRANCH
5601	032454	004737	035170		JSR	PC,DVRXQ	:GO QUE RX BUFFERS & ENABLE RECEIVER
5602							
5603	032460	032737	000010	006602	ALCK1:	BIT	:#QTX,FLAG
5604	032466	001416				BEQ	:ALCK2
5605	032470	013702	006524			MOV	:CPTX,R2
5606	032474	011237	006540			MOV	:TEMP2
5607	032500	012237	006454			MOV	:R2,DVTXA
5608	032504	011237	006542			MOV	:R2,TEMP3
5609	032510	012237	006456			MOV	:R2,DVTCC
5610	032514	010237	006524			MOV	:R2,CPTX
5611	032520	004737	020100			JSR	:PC,LOGTXQ
5612							:LOG TX QUE
5613	032524	032737	000040	006576	ALCK2:	BIT	:#PROTOB,PARAM
5614	032532	001410				BEQ	:10\$
5615	032534	004737	040706			JSR	:PC,PROTCC
5616	032540	032737	000200	006576		BIT	:#ABORT,PARAM
5617	032546	001404				BEQ	:20\$
5618	032550	000137	026644			JMP	:GTRAS
5619							:ABORT!! AND RETURN TO 'DCLT >' PROMPT
5620	032554	004737	035272		10\$:	JSR	:PC,DVTXRX
5621							:IF TRANSMITTING QUE TX BUFFERS & ENABLE TX
5622	032560	032737	000004	006602	20\$:	BIT	:#QTX,FLAG
5623	032566	001514				BEQ	:ALCK3
5624	032570	013737	006470	006540		MOV	:DVRXA,TEMP2
5625	032576	013737	006472	006542		MOV	:DVRCC,TEMP3
5626	032604	004737	020152			JSR	:PC,LOGRXC
5627	032610	032737	000004	006576	UPTABL:	BIT	:#ECHOB,PARAM
5628	032616	001406				BEQ	:UPTA4
5629	032620	013702	006524			MOV	:CPTX,R2
5630	032624	013722	006540			MOV	:TEMP2,(R2)+
5631	032630	013712	006542			MOV	:TEMP3,(R2)
5632	032634	032737	000002	006576	UPTA4:	BIT	:#DATCKB,PARAM
5633	032642	001015				BNE	:UPTA1
5634	032644	012737	000001	006474		MOV	:#01,DVRCT
5635	032652	013737	006440	006522		MOV	:RXPTR,CPTX
5636	032660	022737	000003	006570		CMP	:#ACT,MODTYP
5637	032666	001002				BNE	:UPTA3
5638	032670	005237	006474			INC	:DVRCT
5639	032674	000424			UPTA3:	BR	:UPTX
5640	032676	013702	006522		UPTA1:	MOV	:CPTX,R2
5641	032702	011237	006534			MOV	:R2,TEMP
5642	032706	163737	006542	006534		SUB	:TEMP3,TEMP
5643	032714	013722	006542			MOV	:TEMP3,(R2)+
5644	032720	063737	006542	006540		ADD	:TEMP3,TEMP2
5645	032726	013722	006540			MOV	:TEMP2,(R2)+
5646	032732	013712	006534			MOV	:TEMP,(R2)
5647	032736	162702	000002			SUB	:#2,R2
5648							:STORE OF NEW ADD
5649	032742	010237	006522			MOV	:R2,CPTX
5650							:AND RESTORE IT.
5651	032746				UPTX:		
5652	032746	022737	000002	006570		CMP	:#PAS,MODTYP
5653	032754	001007				BNE	:ALCK2A
5654	032756	042737	000104	006602		BIC	:#QTX+RERX,FLAG
5655	032764	052737	000210	006602		BIS	:#QTX+RCTX,FLAG

:IF NOT PASSIVE LOOP THEN GO TO 2A
:CLEAR BOTH EXPECTED AND COMPLETED FLAGS
:SET THE TX FLAGS

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 140
TRANSMIT - RECEIVE FOR ALL STANDARD MODES

5656	032772	000632		BR	ALCK1	
5657						
5658	032774	005337	006474	ALCK2A:	DEC	DVRCT ;DEC REC COUNT
5659	033000	005737	006474		TST	DVRCT ;IS IT ALL DONE
5660	033004	001005			BNE	ALCK3 ;NO. GO CHECK TX
5661	033006	042737	000004	006602	BIC	#QRX,FLAG ;CLEAR THE RX FLAG
5662	033014	005037	006522		CLR	CPTRR ;YES. CLEAR POINTER
5663	033020	032737	000010	006602	ALCK3:	BIT ;IS IT TX
5664	033026	001447			BEQ	ALCK4 ;IF NOT TX THEN GO BACK
5665	033030	013737	006454	006540	MOV	DVTXA,TEMP2
5666	033036	013737	006456	006542	MOV	DVTCC,TEMP3 ;LOG TX COMPLETED
5667	033044	004737	020116		JSR	PC,LOGTXC
5668	033050	005337	006460		DEC	DVTCT ;DEC TX COUNT
5669	033054	022737	000002	006570	CMP	#PAS,MODTYP
5670	033062	001013			BNE	ALCK3A ;IF NOT PASSIVE MODE GO TO 3A
5671	033064	042737	000210	006602	BIC	#QTX+ETX,FLAG ;CLEAR THE TX FLAGS
5672	033072	052737	000104	006602	BIS	#QRX+ERX,FLAG ;AND SET THE RX FLAGS
5673	033100	005737	006460		TST	DVTCT
5674	033104	001005			BNE	ALCK3C ;IF MORE RX'S DO IT
5675	033106	000137	033166		JMP	CMPSR ; ELSE COMPARE
5676	033112	005737	006460		ALCK3A:	TST ;IS IT ALL DONE
5677	033116	001402			BEQ	ALCK3B ;IF NOT GO BACK TO 5
5678	033120	000137	032400		ALCK3C:	JMP
5679	033124	005037	006524		ALCK3B:	CLR ;IF SO CLEAR POINTER
5680	033130	042737	000010	006602	BIC	#QTX,FLAG ;CLEAR TX FLAG
5681	033136	032737	000002	006576	BIT	#DATCKB,PARAM ;IS IT DAT CK
5682	033144	001403			BEQ	ALCK4A ;IF NOT THEN END WO CKING RX.
5683	033146	005737	006522		ALCK4:	TST CPTRR
5684						
5685	033152	001362			BNE	ALCK3C ;IF SOME RX'S LEFT GO BACK
5686	033154	005737	006524		ALCK4A:	TST CPTRR
5687	033160	001402			BEQ	ALCK4B ;BRANCH IF ANY TX'S LEFT
5688	033162	000137	032524		JMP	ALCK2
5689	033166				ALCK4B:	
5690						
5691						
5692						

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 141
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

--

5725	033166	032737	000002	006576	CMPSR:	BIT	#DATCKB,PARAM	:IS DATA CHECKING TO BE DONE
5726	033174	001522				BEQ	CMPSX	:IF NOT THEN EXIT
5727	033176	013737	006440	006524		MOV	RXPTR,CPTR	:PUT START OF RX POINTERS TO CPTR
5728	033204	013737	006444	006522		MOV	CMPPTR,CPTRR	: AND START OF COMPARE POINTS TO CPTRR
5729	033212	013737	006476	006474		MOV	RXMTOT,DVRCT	
5730								
5731	033220				CMPS3:			
5732	033220	013702	006524			MOV	CPTR,R2	:MOVE CURRET RX PT. TO R2
5733	033224	011237	006540			MOV	(R2),TEMP2	:MOVE RX ADD TO EVENT LOG
5734	033230	012201				MOV	(R2)+,R1	:SET R1 TO START AD. OF RX
5735	033232	012237	006542			MOV	(R2)+,TEMP3	:SET CHAR COUNT TO EVENT LOG
5736	033236	010237	006524			MOV	R2,CPTR	:RESTORE RX POINT
5737								
5738	033242	013702	006522			MOV	CPTRR,R2	:PUT R2 AT COMPARE TABLE
5739	033246	012203				MOV	(R2)+,R3	:SET R3 TO COMPARE ADD
5740	033250	012204				MOV	(R2)+,R4	:SET R4 TO COMP CC
5741	033252	010237	006522			MOV	R2,CPTRR	:RESTORE POINTER
5742	033256	010437	006544			MOV	R4,TEMP4	
5743	033262	004737	020246			JSR	PC,LOGCMP	:LOG COMPARE START.
5744								
5745	033266	020437	006542			CMP	R4,TEMP3	:IS COMPARE COUNT = TO RX COUNT
5746	033272	001410				BEQ	CMPS7	:IF SO GO TO 7
5747	033274	005237	006506			INC	ERRCNT	
5748	033300					ERRSOFT 1,EDDLE,ERR10		:PRINT ERROR

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 142
DATA COMPARISON CODE

Address	Op Code	Op 2	Op 3	Op 4	Op 5	Op 6	Op 7	Op 8	Op 9	Op 10	Op 11	Op 12	Op 13	Op 14	Op 15	Op 16	Op 17	Op 18	Op 19	Op 20	Op 21	Op 22	Op 23	Op 24	Op 25	Op 26	Op 27	Op 28	Op 29	Op 30	Op 31	Op 32	Op 33	Op 34	Op 35	Op 36	Op 37	Op 38	Op 39	Op 40	Op 41	Op 42	Op 43	Op 44	Op 45	Op 46	Op 47	Op 48	Op 49	Op 50	Op 51	Op 52	Op 53	Op 54	Op 55	Op 56	Op 57	Op 58	Op 59	Op 60	Op 61	Op 62	Op 63	Op 64	Op 65	Op 66	Op 67	Op 68	Op 69	Op 70	Op 71	Op 72	Op 73	Op 74	Op 75	Op 76	Op 77	Op 78	Op 79	Op 80	Op 81	Op 82	Op 83	Op 84	Op 85	Op 86	Op 87	Op 88	Op 89	Op 90	Op 91	Op 92	Op 93	Op 94	Op 95	Op 96	Op 97	Op 98	Op 99	Op 100	Op 101	Op 102	Op 103	Op 104	Op 105	Op 106	Op 107	Op 108	Op 109	Op 110	Op 111	Op 112	Op 113	Op 114	Op 115	Op 116	Op 117	Op 118	Op 119	Op 120	Op 121	Op 122	Op 123	Op 124	Op 125	Op 126	Op 127	Op 128	Op 129	Op 130	Op 131	Op 132	Op 133	Op 134	Op 135	Op 136	Op 137	Op 138	Op 139	Op 140	Op 141	Op 142	Op 143	Op 144	Op 145	Op 146	Op 147	Op 148	Op 149	Op 150	Op 151	Op 152	Op 153	Op 154	Op 155	Op 156	Op 157	Op 158	Op 159	Op 160	Op 161	Op 162	Op 163	Op 164	Op 165	Op 166	Op 167	Op 168	Op 169	Op 170	Op 171	Op 172	Op 173	Op 174	Op 175	Op 176	Op 177	Op 178	Op 179	Op 180	Op 181	Op 182	Op 183	Op 184	Op 185	Op 186	Op 187	Op 188	Op 189	Op 190	Op 191	Op 192	Op 193	Op 194	Op 195	Op 196	Op 197	Op 198	Op 199	Op 200	Op 201	Op 202	Op 203	Op 204	Op 205	Op 206	Op 207	Op 208	Op 209	Op 210	Op 211	Op 212	Op 213	Op 214	Op 215	Op 216	Op 217	Op 218	Op 219	Op 220	Op 221	Op 222	Op 223	Op 224	Op 225	Op 226	Op 227	Op 228	Op 229	Op 230	Op 231	Op 232	Op 233	Op 234	Op 235	Op 236	Op 237	Op 238	Op 239	Op 240	Op 241	Op 242	Op 243	Op 244	Op 245	Op 246	Op 247	Op 248	Op 249	Op 250	Op 251	Op 252	Op 253	Op 254	Op 255	Op 256	Op 257	Op 258	Op 259	Op 260	Op 261	Op 262	Op 263	Op 264	Op 265	Op 266	Op 267	Op 268	Op 269	Op 270	Op 271	Op 272	Op 273	Op 274	Op 275	Op 276	Op 277	Op 278	Op 279	Op 280	Op 281	Op 282	Op 283	Op 284	Op 285	Op 286	Op 287	Op 288	Op 289	Op 290	Op 291	Op 292	Op 293	Op 294	Op 295	Op 296	Op 297	Op 298	Op 299	Op 300	Op 301	Op 302	Op 303	Op 304	Op 305	Op 306	Op 307	Op 308	Op 309	Op 310	Op 311	Op 312	Op 313	Op 314	Op 315	Op 316	Op 317	Op 318	Op 319	Op 320	Op 321	Op 322	Op 323	Op 324	Op 325	Op 326	Op 327	Op 328	Op 329	Op 330	Op 331	Op 332	Op 333	Op 334	Op 335	Op 336	Op 337	Op 338	Op 339	Op 340	Op 341	Op 342	Op 343	Op 344	Op 345	Op 346	Op 347	Op 348	Op 349	Op 350	Op 351	Op 352	Op 353	Op 354	Op 355	Op 356	Op 357	Op 358	Op 359	Op 360	Op 361	Op 362	Op 363	Op 364	Op 365	Op 366	Op 367	Op 368	Op 369	Op 370	Op 371	Op 372	Op 373	Op 374	Op 375	Op 376	Op 377	Op 378	Op 379	Op 380	Op 38
---------	---------	------	------	------	------	------	------	------	------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	--------	-------

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 143
MODEM CHANGE REPORTS

.SBTTL MODEM CHANGE REPORTS

++
FUNCTIONAL DESCRIPTION:
THIS SECTION REPORTS THE NUMBER OF MODEM STATUS CHANGES
THAT OCCUR ON EACH PASS. THE ERROR IS ONLY REPORTED IF
THERE WERE ANY CHANGES IN OTHER WORDS A COUNT OF ZERO IS
NOT REPORTED. THE CHANGES ARE REPORTED IN TWO CLASSES ..
HARD ERRORS AND GLITCHES. HARD ERRORS ARE WHEN THE DEVICE
IS ABLE TO LATCH UP THE BAD MODEM STATUS. GLITCHES OCCUR
WHEN THE MODEM STATUS CHANGES TO CAUSE A DATA SET CHANGE
INTERRUPT BUT THE CHANGE DOES NOT OCCUR LONG ENOUGH FOR
THE DEVICE TO LATCH THE DATA

INPUTS:

'MGLCNT' - CONTAINS NUMBER OF GLITCH ERRORS
'MHRCNT' - CONTAINS NUMBER OF HARD ERRORS

OUTPUTS:

'MGLCNT' -ZEROED BY THIS SECTION
'MHRCNT' -ZEROED BY THIS SECTION

;;
--

CMPSEX: TST MGLCNT ;CHECK FOR ANY GLITCH ERRORS
BNE MCREP ;IF NON ZERO REPORT THEM
TST MHRCNT ;CHECK FOR ANY HARD ERRORS
BEQ ENDPS ;IF NONE GO TO END OF PASS
;REPORT ANY MODEM ERRORS HERE

MCREP: INC ERRCNT ;BUMP ERROR COUNT
ERRSOFT 4, MSCMS, ERR4

TRAP CSERSOFT
.WORD
.WORD MSCMS
.WORD ERR4

CLR MGLCNT ;CLEAR GLITCH COUNT
CLR MHRCNT ;CLEAR THE HARD COUNT

5789
5790
5791
5792
5793
5794
5795
5796
5797
5798
5799
5800
5801
5802
5803
5804
5805
5806
5807
5808
5809
5810
5811
5812
5813
5814 033442 005737 011516
5815 033446 001003
5816 033450 005737 011520
5817 033454 001412
5818
5819
5820
5821 033456 005237 006506
5822 033462
5823 033462 104457
5824 033464 000004
5825 033466 014764
5826 033470 017604
5827 033472 005037 011516
5828 033476 005037 011520
5829

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 144
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 PROMT
:   IN NOT = TO 0 THEN GO BACK TO MODE DISPATCHER

```

: SUBORDINATE ROUTINES USED:

: 'LOGEOP' - LOG END OF PASS TO EVENT LOG

:-----

```

5830
5831
5832
5833
5834
5835
5836
5837
5838
5839
5840
5841
5842
5843
5844
5845
5846
5847 033502 005237 006504      ENDPS:  INC      PSCNT      ;BUMP PASS COUNT
5848
5849 033506 013737 006502 006544      MOV      OPVAR,TEMP4
5850 033514 013737 006504 006540      MOV      PSCNT,TEMP2
5851 033522 013737 006506 006542      MOV      ERRCNT,TEMP3
5852 033530 004737 020320      JSR      PC,LOGEOP      ;LOG END OF PASS
5853
5854 033534 022737 177777 006600      CMP      #-1,RPASS      ;SEE IF RPASS=-1
5855 033542 001403      BEQ      1$      ;IF IT IS DON'T DECRMNT, LOOP FOREVER
5856 033544 005337 006600      DEC      RPASS      ;DEC PASS COUNT
5857 033550 001402      BEQ      2$      ;IF DONE EXIT TEST
5858 033552 000137 032142 1$:      JMP      GTRX2      ;ELSE GO BACK AND DISPATCH
5859 033556 042777 000120 155664 2$:  BIC      #RINTEN!RXENA,@RXCSR      ;TURN OFF RX
5860 033564 000137 026644      JMP      GTRAS      ;WHEN RPASS=0 GO BACK TO 'DCLT>'
5861

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 145
DOWN-LINE-LOAD SECTION

5862
5863
5864
5865
5866
5867
5868
5869
5870
5871
5872
5873
5874
5875
5876
5877
5878
5879
5880
5881

.SBTTL DOWN-LINE-LOAD SECTION

:++

: FUNCTIONAL DESCRIPTION:

: DOWN LINE LOAD IS NOT SUPPORTED BY THIS DEVICE..
: IF THIS MODE IS CALLED BY THE COMMAND LINE INTERPRETER
: THEN A MESSAGE WILL BE PRINTEDTHAT SAYS DOWN LINE
: LOAD IS NOT!! SUPPORTED BY THIS DEVICE.
:--

DLL:

PRINTF #DLLCM

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

JMP GTRAS

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 146
TALK MODE SECTION

.SBTTL TALK MODE SECTION

++
FUNCTIONAL DESCRIPTION:
TALK MODE SECTION
IN THIS MODE, THE 'TALK' END OF THE LINK TRANSMITS OPERATOR
SPECIFIED MESSAGES UNTIL A 'EXIT' MESSAGE IS TYPE. AT THAT POINT,
THIS END OF THE LINK GOES INTO 'LISTEN' MODE.

SUBORDINATE ROUTINES USED:

'LOGTXQ' - LOG TX BUFFER QUED TO EVENT LOG
'DVTXRX' - QUE TX BUFFER TO DEVICE AND WAIT FOR COMPLETE
'LOGTXC' - LOG TX COMPLETE TO EVENT LOG

CALLING SEQUENCE:

JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2

--

TALCK:

BIC #DATCKB,PARAM ;SET NOCHECK
MOV #OPBUF,R2
1\$: MOV #-1,(R2)+ ;CLEAR OUT OPBUFFER FIRST
CMP #OPEND,R2
BNE 1\$
GMANID OPRMM,OPBUF,A,-1,1,72.,NO ;GET TALK MESSAGE

TRAP
BR
.WORD
.WORD
.WORD
.WORD
.WORD
.WORD
CSGMAN
10001\$
OPBUF
T\$CODE
OPRMM
-1
T\$LOLIM
T\$HILIM

10001\$:

CLR R2 ;NOW GET CHAR COUNT
CMPB #377,OPBUF(R2)
BEQ 3\$
INC R2
BR 2\$
3\$: MOV R2,OPCNT

MOV #OPBUF,DVTXA ;SET UP TX ADDR.

MOV #OPBUF,TEMP2

MOV OPCNT,TEMP3

MOV OPCNT,DVTCC ;SET UP TX CC

JSR PC,LOGTXQ

BIS #QTX+QETX,FLAG ;SET UP FLAGS

CLR CPTRR ;CLEAR RX POINTER

::THIS CODE ADDED FOR PROTOCOL

BIT #PROTOB,PARAM ;'/PROTOCOL'?

BNE 20\$;YES,BRANCH

JSR PC,DVTXRX

BR 25\$;JUMP AROUND PROTOCOL

20\$: BIC #RXQ,FLAG ;MAKE SURE NOT TO RECEIVE

5882
5883
5884
5885
5886
5887
5888
5889
5890
5891
5892
5893
5894
5895
5896
5897
5898
5899
5900
5901 033614
5902 033614 042737 000002 006576
5903 033622 012702 002520
5904 033626 012722 177777
5905 033632 022702 002642
5906 033636 001373
5907 033640
5908 033640 104443
5909 033642 000406
5910 033644 002520
5911 033646 000142
5912 033650 014065
5913 033652 177777
5914 033654 000001
5915 033656 000110
5916 033660
5917 033660 005002
5918 033662 122762 000377 002520 2\$:
5919 033670 001402
5920 033672 005202
5921 033674 000772
5922 033676 010237 002166 3\$:
5923
5924 033702 012737 002520 006454
5925 033710 012737 002520 006540
5926 033716 013737 002166 006542
5927 033724 013737 002166 006456
5928 033732 004737 020100
5929 033736 052737 000210 006602
5930 033744 005037 006522
5931
5932
5933 033750 032737 000040 006576
5934 033756 001003
5935 033760 004737 035272
5936 033764 000405
5937 033766 042737 000004 006602 20\$:

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 147
TALK MODE SECTION

5938	033774	004737	040706		CALL	PROTOCOL	;DO DDCMP PROTOCOL
5939							
5940	034000	013737	006454	006540	25\$:	MOV	DVTXA,TEMP2
5941	034006	013737	006456	006542		MOV	DVTCC,TEMP3
5942	034014	004737	020116			JSR	PC,LOGTXC
5943	034020	022737	054105	002520		CMP	#'EX,OPBUF ;CHECK FOR EXIT
5944	034026	001272				BNE	TALCK
5945	034030	022737	052111	002522		CMP	#'IT,OPBUF+2
5946	034036	001266				BNE	TALCK
5947	034040	042737	000210	006602		BIC	#QTX+NETX,FLAG ;CLEAR THE TX BITS
5948	034046	012737	000006	006570		MOV	#LIS,MODTYP ;CHANGE TO LISTEN MODE
5949	034054	000137	032142			JMP	GTRX2 ;AND GO BACK TO DISPATCH

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 148
LISTEN MODE SECTION

.SBTTL LISTEN MODE SECTION

++
FUNCTIONAL DESCRIPTION:
LISTEN MODE SECTION
IN THIS MODE, THE 'LISTEN' END OF THE LINK PRINTS ALL OF THE MESSAGES
RECEIVED BY THE DEVICE ON THE OPERATOR'S CONSOLE. IF THE MESSAGE
RECEIVED IS AN 'EXIT' MESSAGE, THEN THE NODE ENTERS 'TALK' MODE.

SUBORDINATE ROUTINES USED:

'DVRXQ' - QUE RECEIVE BUFFER SPACE TO DEVICE
'LOGRXQ' - LOG RECEIVE BUFFER QUED TO EVENT LOG
'DVTXRX' - WAIT FOR RX TO COMPLETE
'LOGRXC' - LOG RX COMPLETE TO EVENT LOG

CALLING SEQUENCE:

JMP @MODE(R2) ;DISPATCH TO MODE BASED ON MODE TYPE IN R2

--

5970	034060	042737	000002	006576	LISCK: BIC	#DATCKB,PARAM	;CLEAR CHECK BIT		
5971	034066				PRINTF	#LISP	;PRINT PROMPT FOR OPR.		
5972	034066	012746	014054					MOV	#LISP,-(SP)
5973	034072	012746	000001					MOV	#1,-(SP)
5974	034076	010600						MOV	SP,R0
5975	034100	104417						TRAP	C\$PNTF
5976	034102	062706	000004					ADD	#4,SP
5977	034106	012737	002520	006470	LISCKA: MOV	#OPBUF,DVRXA	;SET DEVICE UP TO REC AT OPBUF		
5978	034114	012737	002520	006540	MOV	#OPBUF,TEMP2			
5979	034122	012737	000122	006472	MOV	#82.,DVRCC	;SET UP CHAR COUNT TO 82.		
5980	034130	012737	000122	006542	MOV	#82.,TEMP3			
5981	034136	052737	000104	006602	BIS	#QRX+MERX,FLAG	;SET UP FLAG		
5982	034144	005037	006524		CLR	CPTR	;CLEAR THE TX.		
5983									
5984					:: WAS	PROTOCOL	SELECTED ?		
5985	034150	032737	000040	006576	BIT	#PROTOB,PARAM	;' /PROTOCOL' ?		
5986	034156	001007			BNE	20\$	;YES,BRANCH		
5987									
5988	034160	004737	035170		JSR	PC,DVRXQ	;QUE RX		
5989	034164	004737	020134		JSR	PC,LOGRXQ			
5990	034170	004737	035272		JSR	PC,DVTXRX	;GO TO DEVICE RX. SUBROUTINE		
5991	034174	000402			BR	25\$			
5992	034176	004737	040706		20\$: CALL	PROTOD	;DO DDCMP PROTOCOL		
5993	034202	013737	006470	006540	25\$: MOV	DVRXA,TEMP2			
5994	034210	013737	006472	006542	MOV	DVRCC,TEMP3	;SET UP ADDR.AND CC.		
5995	034216	004737	020152		JSR	PC,LOGRXC	;LOG COMPLETED		
5996	034222	063737	006470	006472	ADD	DVRXA,DVRCC			
5997	034230	105077	152236		CLRB	@DVRCC			
5998	034234				PRINTF	#OPBFPT			
5999	034234	012746	002514					MOV	#OPBFPT,-(SP)
6000	034240	012746	000001					MOV	#1,-(SP)
6001	034244	010600						MOV	SP,R0
6002	034246	104417						TRAP	C\$PNTF
6003	034250	062706	000004					ADD	#4,SP
6004	034254	022737	054105	002520	CMP	#'EX,OPBUF	;COMPARE FOR EX OF 'EXIT'		
6005	034262	001311			BNE	LISCKA	;IF NOT EXIT THEN GO BACK		

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 149
LISTEN MODE SECTION

6006	034264	022737	052111	002522	CMP	#'IT,OPBUF+2	;IF FIRST HALF OK CHECK NEXT PART
6007	034272	001305			BNE	LISCKA	;IF NOT EXIT THE GO BACK
6008	034274	012737	000005	006570	MOV	#TAL,MODTYP	;CHANGE MODE TO TALK
6009	034302	000137	032142		JMP	GTRX2	;RETURN TO DISPATCHER
6010							
6011							

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 150
DEVICE FUNCTION SUBROUTINES

.SBTTL DEVICE FUNCTION SUBROUTINES

.SBTTL DEVICE INIT SUBROUTINE

```

:++
: FUNCTIONAL DESCRIPTION:
: DVINIT- DEVICE INIT ROUTINE
: THIS ROUTINE IS DEVICE DEPENDENT CODE THAT INITIS
: THE DEVICE BEING TESTED.
: IT SETS THE DEVICE UP TO THE MODE IT IS TO RUN IN AND
: INITIATES THE START,STACK,ACK SEQUENCE IF THE 'RNODE'(REMOTE
: NODE)INPUT INDICATES THE REMOTE NODE IS NON-ITEP.
:
: INPUTS:      'FHDPLX' INDICATES IF MODE IS FULL OR HALF DUPLEX. (1=FULL)
:              ADDRESS POINTERS (SELO,...) ALREADY POINT TO DEVICE'S REG.S
:
:              'MLTYP' INDICATES THE LOOP TYPE (1=TTL,2=CAB,3=RM,4=LM)
:              'RNODE' INDICATES THE TYPE OF REMOTE NODE (ITEP=1,NON-ITEP=0)
:
: SUBORDINATE ROUTINES USED:
:
:              'CTSSR' - CLEAR TO SEND SUB ROUTINE
:              'DVIN31' - SEND CONTROL AND REC OR TIME OUT
:              'CLRRTS' - CLEAR REQUEST TO SEND ROUTINE
:              'LGDVE' - LOG DEVICE ERROR TO EVENT LOG
:
: CALLING SEQUENCE:
:              JSR      PC,DVINIT
:--

```

DVINIT:

```

: DO MASTER CLEAR
20$: MOV      #RESET,@TXCSR      ;DO A MASTER CLEAR
      NOP                      ;WAIT A WHILE FOR
      NOP                      ;IT TO CLEAR
      MOV      #RESET,@TXCSR      ;DO A CLEAR AGAIN(NECESSARY WITH LOOP=CABLE)
      MOVB     @RXCSR,TEMP3      ;SEE IF IT WORKED
      BEQ      DVINI             ;BRANCH IF OK
      BREAK
                                     TRAP      CSBRK

      ;REPORT ERROR IF RESET
      ;DOES NOT WORK

      MOV      #DVEMO,TEMP2
      MOV      @RXCSR,TEMP3
      MOV      @TXCSR,TEMP4      ;LOAD UP ERRM. AND REG OUTPUTS
      JSR      PC,LGDVE          ;LOG TIME OUT WAITING FOR RUN
      INC      ERRCNT
      ERRSOFT 5,DVEMO,ERR13
                                     TRAP      CSERSOFT

```

6012				
6013				
6014				
6015				
6016				
6017				
6018				
6019				
6020				
6021				
6022				
6023				
6024				
6025				
6026				
6027				
6028				
6029				
6030				
6031				
6032				
6033				
6034				
6035				
6036				
6037				
6038				
6039				
6040				
6041				
6042				
6043				
6044				
6045				
6046	034306			
6047				
6048				
6049	034306	012777	000400	155142
6050	034314	000240		
6051	034316	000240		
6052	034320	012777	000400	155130
6053	034326	117737	155116	006542
6054	034334	001423		
6055	034336			
6056	034336	104422		
6057				
6058				
6059				
6060				
6061	034340	012737	016747	006540
6062	034346	017737	155076	006542
6063	034354	017737	155076	006544
6064	034362	004737	020162	
6065	034366	005237	006506	
6066	034372			
6067	034372	104457		

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 151
DEVICE INIT SUBROUTINE

```

6068 034374 000005
6069 034376 016747
6070 034400 017636
6071 034402 000741
6072
6073
6074
6075 034404 042737 000003 006602 DVIN1: BIC #3,FLAG ;CLEAR INPUT AND OUTPUT INT FLAGS
6076 034412 042777 004000 155036 BIC #TTL,@TXCSR ;CLEAR INTERNAL LOOP
6077 034420 022737 000001 006572 CMP #TTL,MLTYP ;IS TTL SELECTED
6078 034426 001004 BNE DVIN3 ;IF NOT GO TO 3
6079 034430 052777 004000 155020 BIS #TTL,@TXCSR ;ELSE SET INTERNAL LOOP
6080 034436 000461 BR DVIN37
6081
6082 034440 022737 000002 006572 DVIN3: CMP #CABLE,MLTYP ;CABLE LOOP ?
6083 034446 001004 BNE 10$ ;NO,BRANCH
6084 034450 052777 010000 155000 BIS #CABLOP,@TXCSR ;SET EXTERNAL LOOP (TURN AROUND CONNECTOR)
6085 034456 000451 BR DVIN37
6086
6087 034460 022737 000004 006570 10$: CMP #DOW,MODTYP ;CHECK IF DLL
6088 034466 001002 BNE DVIN3A ;BRANCH IF NOT DLL
6089 034470 000137 035130 JMP DVINEX ;ELSE EXIT
6090
6091 034474 012777 000002 154746 DVIN3A: MOV #DTR,@RXCSR ;SET UP DTR.
6092
6093 034502 012737 002000 006642 DVIN38: MOV #2000,TIMER1
6094 034510 005737 006642 TST TIMER1
6095 034514 001022 BNE DVIN39 ;IF TIMER NOT OUT GO TO 39
6096
6097
6098
6099 034516 012737 017377 006540 MOV #DVEM6,TEMP2
6100 034524 017737 154720 006542 MOV @RXCSR,TEMP3
6101 034532 017737 154720 006544 MOV @TXCSR,TEMP4
6102 034540 004737 020162 JSR PC,LGDVE
6103 034544 005237 006506 INC ERRCNT
6104 034550 ERRSOF 11,DVEM6,ERR13
6105 034550 104457
6106 034552 000013
6107 034554 017377
6108 034556 017636
6109 034560 000745
6110 034562
6111 034562 104422
6112 034564 017737 154660 011474 MOV @RXCSR,IRXCSR ;GET COPY OF RXCSR
6113 034572 032737 001000 011474 BIT #BIT9,IRXCSR ;IS MODEM READY SET
6114 034600 001743 BEQ DVIN38
6115 034602 013777 011470 154642 DVIN37: MOV DUPPAR,@PARCSR ;SET PARAMETER REGISTER
6116 034610 005737 011522 TST RNODE ;REMOTE ITEP ?
6117 034614 001145 BNE DVINEX ;YES,BRANCH
6118 034616 005737 006574 TST FHDPLX ;FULL DUPLEX ?
6119 034622 001542 BEQ DVINEX ;NO,BRANCH
6120 034624 032737 000040 006576 BIT #PROTOB,PARAM ;'/PROTOCOL' ?
6121 034632 001136 BNE DVINEX ;YES,BRANCH
6122
6123

```

BR DVINIT ;GO BACK AND TRY MSTR CLR AGAIN IF ERROR

;SET TTL LOOP IF REQU'D

TRAP C\$ERSOFT 11,DVEM6,ERR13

TRAP C\$BRK

:: THIS START-STACK ROUTINE USED IN NON-PROTOCOL,NON-ITEP,FULL DUPLEX MODE

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 152
DEVICE INIT SUBROUTINE

```

6124
6125      ;SET UP TO SEND STRT
6126 034634 112737 000005 002645      MOV      #5,HDMSC+1      ;SET UP ENQ
6127 034642 052737 000060 006602      BIS      #RXM!TXM,FLAG      ;SET FLAG WORD
6128 034650 012737 000074 006646      MOV      #60.,TIMERS      ;SET TIMER FOR 1 MINUTE
6129 034656 004737 036576              JSR      PC,CTSSR      ;SET CTS IF NESC.
6130 034662 012737 000006 002646 DVIN41: MOV      #6,HDMCC      ;SET UP STRT CODE
6131 034670 004737 036432              JSR      PC,DVIN31      ;GO TX STRT AND CHK FOR RX.
6132 034674 005737 006646              TST      TIMERS
6133 034700 001466              BEQ      DVIN81      ;IF TIMER EXPIERED EXIT
6134
6135 034702 022737 000006 002660 DVIN4:  CMP      #6,RHDMCC      ;IS THE RCVD=STRT
6136 034710 001441              BEQ      DVIN8      ;IF SO GO TO ASTRT
6137 034712 022737 000007 002660      CMP      #7,RHDMCC      ;IS IT A STACK
6138 034720 001360              BNE      DVIN41      ;IF NOT STACK ETIHER GO BACK
6139
6140 034722 004737 036576              JSR      PC,CTSSR      ;SET REQUEST TO SEND
6141 034726 042737 001010 006602      BIC      #QTX!PAD,FLAG      ;CLEAR TX COMPT FLAG.
6142 034734 012737 000001 002646      MOV      #1,HDMCC      ;SET UP ACK
6143 034742 012737 002645 011500      MOV      #HDMSC+1,MSGPTR      ;SET UP POINTER
6144 034750 013737 002654 011502      MOV      HDMC,MSGCC
6145 034756 012737 000010 011504      MOV      #8.,SYNCC      ;SET UP SYNC COUNT
6146 034764 052777 000120 154464      BIS      #SEND!TINTEN,@TXCSR ;TURN ON TX, ENABLE TX INTERRUPT
6147 034772 032737 000010 006602 DIVN91: BIT      #QTX,FLAG
6148 035000 001053              BNE      DVINEX      ;EXIT IF ACK SENT
6149 035002              BREAK
6150 035002 104422              TRAP      CSBRK
6151 035004 005737 006646      TST      TIMERS
6152 035010 001370              BNE      DIVN91      ;IF NOT TIMER EXPIRED RECHK TX.
6153 035012 000421              BR       DVIN81      ;IF TIMER OUT REPORT IT
6154
6155 035014 012737 000007 002646 DVIN8:  MOV      #7,HDMCC      ;SET POTINTER TO STACK
6156 035022 004737 036432              JSR      PC,DVIN31      ;AND GO SEND STACK
6157 035026 005737 006646              TST      TIMERS
6158 035032 001411              BEQ      DVIN81      ;REPORT ERROR IF TIME OUT
6159 035034 022737 000001 002660      CMP      #1,RHDMCC      ;IS IT ACK RCVD?
6160 035042 001432              BEQ      DVINEX      ;IF SO EXIT
6161 035044 022737 000007 002660      CMP      #7,RHDMCC      ;IS IT STACK RCVD
6162 035052 001723              BEQ      DVIN9      ;IF SO SEND ACK
6163 035054 000757              BR       DVIN8      ;IF NEITHER SEND ANOTHER ACK
6164
6165      ;DO ERROR AND REPEAT
6166
6167 035056 012737 017313 006540 DVIN81: MOV      #DVEM5,TEMP2
6168 035064 013737 002660 006542      MOV      RHDMCC,TEMP3
6169 035072 013737 002646 006544      MOV      HDMCC,TEMP4
6170 035100 004737 020162              JSR      PC,LGDVE
6171 035104 005237 006506              INC      ERRCNT
6172 035110              ERRSOF 10.,DVEM5,ERR13
6173 035110 104457              TRAP      CSERSOF 10
6174 035112 000012              .WORD   DVEM5
6175 035114 017313              .WORD   ERR13
6176 035116 017636
6177 035120 005237 006502      INC      OPVAR
6178 035124 000137 034306      JMP      DVINIT      ;COUNT HOW MANY TIMES WE DO THIS.
6179                                ;TRY ALL OVER AGAIN

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 153
DEVICE INIT SUBROUTINE

6180	035130	004737	037010		DVINEX: JSR	PC,CLRRTS	:CLEAR RTS IF NESC
6181	035134	042737	173777	006602	BIC	#173777,FLAG	:CLEAR FLAG WORD
6182	035142	052737	002000	006602	BIS	#INOV,FLAG	:SET THE INITT OVER FLAG
6183	035150	000207			RTS	PC	:RETURN TO CALLER
6184							
6185							

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 154
DEVICE GET MODEM STATUS SUBROUTINE

.SBTTL DEVICE GET MODEM STATUS SUBROUTINE

```

:++
: FUNCTIONAL DESCRIPTION:
:   'DVMODS' GET MODEM STATUS
:
: IMPLICIT INPUTS:
:   THE BIT POSITION AND AVAILABILITY OF THE MODEM SIGNALS CTS,DSR,...RI,..
:   FOUND IN THE DEPENDENT PORTION OF THE GLOBAL EQUATES SECTION.
:
: OUTPUTS:
:   CURRENT MODEM SIGNAL VALUES IN 'MODS'
:
: CALLING SEQUENCE:
:   JSR    PC,DVMODS
:--

```

6186
6187
6188
6189
6190
6191
6192
6193
6194
6195
6196
6197
6198
6199
6200
6201
6202
6203
6204
6205
6206
6207
6208
6209
6210
6211

```

035152 017737 154272 007556 DVMODS: MOV    @RXCSR,MODS    ;READ MODEM STATUS
035160 042737 104761 007556 DVMEX:  BIC    #104761,MODS    ;CLEAR BITS NOT RELATING TO MODEM
035166 000207                RTS      PC                ;RETURN TO CALLER

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 155
DEVICE QUEUE RECEIVE SPACE SUBROUTINE

.SBTTL DEVICE QUEUE RECEIVE SPACE SUBROUTINE

```

:++
: FUNCTIONAL DESCRIPTION:
:   DVRXQ - THIS SUBROUTINE QUEUES THE RECIEVER BUFFER SPACE TO THE
:           DEVICE, THEN CLEARS THE QRX BIT OF THE FLAG WORD.
:
: INPUTS:
:   DVRXA = ADDRESS OF RX BUFFER SPACE
:   DVRCC = BYTE CHAR COUNT OF RX BUFFER
:   QRX FLAG BIT = SET BY CALLING ROUTINE
:
: OUTPUTS:
:   QRX FLAG BIT = CLEARED BY ROUTINE
:
: CALLING SEQUENCE:
:   JSR      PC,DVRXQ
:--

```

```

DVRXQ:
BIT      #QRX,FLAG      ;ARE WE RECEIVING ?
BEQ      DVREX          ;NO,BRANCH
BIC      #QRX+#BCC+#RXM,FLAG ;CLEAR FLAG FOR RX
TST      RNODE          ;ITEP MODE ?
BEQ      DVRX2          ;NO,BRANCH
BIS      #RXM+#BCC,FLAG  ;GET JUST THE DATA NO CRC.
MOV      DVRXA,RMSGPT    ;RECEIVE DATA BUFFER ADDRESS
MOV      #72,RMSGCC      ;SET UP RX TO GET ITEP MSG.
MOV      #70,DVRCC
BR       DVRX3

;ENABLE RX, RX INTERRUPTS,AND DATA SET INTERRUPTS

DVRX2: MOV      #RHMSG+1,RMSGPT ;SETUP RX BUFFER ADDRESS
        MOV      HDMC,RMSGCC    ;SETUP CHARACTER COUNT
DVRX3: BIS      #RINTEN!RXENA!#DSITEN!#STRIP,@RXCSR ;ENABLE RECEIVER
DVREX: RTS      PC             ;RETURN TO CALLER

```

```

6212
6213
6214
6215
6216
6217
6218
6219
6220
6221
6222
6223
6224
6225
6226
6227
6228
6229
6230
6231
6232 035170
6233 035170 032737 000004 006602
6234 035176 001434
6235 035200 042737 000444 006602
6236 035206 005737 011522
6237 035212 001415
6238 035214 052737 000440 006602
6239 035222 013737 006470 011510
6240 035230 012737 000072 011512
6241 035236 012737 000070 006472
6242 035244 000406
6243
6244
6245
6246 035246 012737 002657 011510 DVRX2:
6247 035254 013737 002654 011512      MOV
6248 035262 052777 000560 154160 DVRX3:
6249 035270 000207      DVREX:
6250

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 156
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

.SBTTL DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

:++
: FUNCTIONAL DESCRIPTION:
: DVTXRX-DEVICE TRANSMIT AND RECEIVE ROUTINE
: THIS CODE QUES THE TRANSMIT BUFFER TO THE DEVICE
: IF NEEDED. THE CODE THEN WAITS FOR A TX COMPLE,
: RX COMPLETE OR BOTH. THE CODE REPORTS A TIME OUT
: ERROR IF NO OUTPUT INTERRUPT IS RECIEVED BEFORE
: 60 SECONDS. AFTER REPORTING ERROR TIMER IS RE STARTED
: AND DEVICE WILL CONTINUE TO WAIT FOR INTERRUPT.

```

```

: INPUTS:
: 'DVTXA' = ADDRESS OF TRANSMIT MSG.
: 'DVTCC' = BYTE COUNT OF TRANSMIT MSG.
: 'QTX' BIT = SET IF TRANSMIT REQUESTED
: 'ETX' BIT = SET IF TRNASMIT EXPECTED
: 'ERX' BIT = SET IF RECIEVE EXPECTED

```

```

: OUTPUTS:
: 'DVTXA' = ADDRESS OF TX MSG. COMPLETED
: 'DVTCC' = BYTE COUNT OF TX MSG. COMPLETED
: 'QTX' = SET IF TX COMPLETED
: 'DVRXA' = ADDRESS OF RX MSG. COMPLETED
: 'DVRCC' = BYTE COUNT OF RX MSG. COMPLETED
: 'QRX' = SET IF RX COMPLETED

```

```

: SUBORDINATE ROUTINES USED:
: 'LGDVE' - LOG DEVICE ERROR TO EVENT LOG

```

```

: CALLING SEQUENCE:
: JSR PC,DVTXRX
:--

```

6287					DVTXRX: BIT	#QTX,FLAG	;ANY TX TO QUE ?(ARE WE TRANSMITTING?)
6288	035272	032737	000010	006602	BEQ	DVTR3	;NO,BRANCH
6289	035300	001444			BIC	#QTX+#TXM+PAD,FLAG	;CLEAR FLAG
6290	035302	042737	001030	006602	JSR	PC,CTSSR	;GO SET CTS
6291	035310	004737	036576		TST	RNODE	;REMOTE NODE = ITEP ?
6292	035314	005737	011522		BEQ	DVTR1	;NO,BRANCH
6293	035320	001412			BIS	#TXM,FLAG	;TX ONLY DATA MESSAGE
6294	035322	052737	000020	006602	MOV	DVTXA,MSGPTR	;MESSAGE ADDRESS
6295	035330	013737	006454	011500	MOV	DVTCC,MSGCC	;MESSAGE CHAR COUNT
6296	035336	013737	006456	011502	BR	DVTR2	;
6297	035344	000414				;ENABLE TX AND TX INTER.	
6298							
6299							
6300	035346	112737	000201	002645	DVTR1: MOVB	#201,HDMMSG+1	;SET UP SOH
6301	035354	012737	002645	011500	MOV	#HDMMSG+1,MSGPTR	;SET POINTER TO HEADER
6302	035362	013737	006456	002646	MOV	DVTCC,HDMCC	
6303	035370	013737	002654	011502	MOV	HDMC,MSGCC	;SET CC FOR HEADER
6304	035376	012737	000177	011504	DVTR2: MOV	#177,SYNCC	;SET UP FOR 177 SYNC.
6305							; (SO MANY SYNC'S ARE NECESSARY
6306							; IF THE OTHER NODE H/S A

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 157
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

```

6307                                     :::SLOWER CPU.)
6308
6309 035404 052777 000120 154044      BIS      #SEND!#TINTEN,@TXCSR      ;TURN ON TX
6310
6311 035412 012737 000074 006646  DVTR3: MOV      #60.,TIMERS          ;SET TIMER FOR 60 SECS
6312
6313 035420      DVTR8: BREAK
6314 035420 104422
6315 035422 005737 006646      TST      TIMERS          ;IS TIMER EXPIRED      TRAP      C$BRK
6316 035426 001022      BNE      TOINOT
6317
6318      ;LOG ERROR TIME OUT RX OR TX NOT COMPLETED
6319
6320 035430 012737 017110 006540      MOV      #DVEM2,TEMP2
6321 035436 017737 154006 006542      MOV      @RXCSR,TEMP3
6322 035444 017737 154006 006544      MOV      @TXCSR,TEMP4
6323 035452 004737 020162      JSR      PC,LGDVE
6324 035456 005237 006506      INC      ERRCNT
6325 035462      ERRSOF 7,DVEM2,ERR13
6326 035462 104457
6327 035464 000007
6328 035466 017110
6329 035470 017636
6330 035472 000747
6331
6332 035474 032737 000010 006602  TOINOT: BIT      #QTX,FLAG          ;IS IT TX COMPL?
6333 035502 001406      BEQ      DVTR4          ;BRANCH IF TX NOT DONE.
6334 035504 004737 037010      JSR      PC,CLRRTS
6335 035510 032737 000100 006602      BIT      #ERX,FLAG          ;ARE WE EXPECTING TO RX
6336 035516 001416      BEQ      DVTREX          ;BRANCH IF NOT.
6337
6338 035520 032737 000004 006602  DVTR4: BIT      #QRX,FLAG          ;IS RX DONE
6339 035526 001734      BEQ      DVTR8          ;GO BACK AND TIME IF NOT
6340
6341 035530 032737 000200 006602      BIT      #ETX,FLAG          ;ARE WE EXPECTG TO TX.
6342 035536 001406      BEQ      DVTREX          ;BRANCH IF NOT.
6343
6344 035540 032737 000010 006602      BIT      #QTX,FLAG          ;IS IT TX COMPLETED
6345 035546 001724      BEQ      DVTR8          ;GO BACK AND TIME OUT
6346 035550 004737 037010      JSR      PC,CLRRTS          ;CLEAR RTS IF NESC.
6347 035554 000207      DVTREX: RTS      PC          ;AND EXIT
6348

```

TRAP C\$ERSOFT
.WORD 7
.WORD DVEM2
.WORD ERR13

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 158
DEVICE TRANSMIT AND RECEIVE SUBROUTINE

; DEVICE DEPENDENT SUBROUTINES

.SBTTL DEVICE INTERRUPT SERVICE ROUTINES

```

++
: FUNCTIONAL DESCRIPTION:
: RECEIVER INTERRUPT ROUTINE. WHEN A RX INT. OCCURS
: THIS ROUTINE DECIDES IF IT IS A DATA SET
: CHANGE OR DATA INTERRUPT. IF IT IS A DATA SET CHANGE
: INTERRUPT IT PUTS THE STATUS IN 'CMODS' AND COMPARES
: THAT STATUS TO THE OLD STATUS IN 'MODS'. IF THEY ARE
: THE SAME THAT MEANS THE INTERRUPT WAS CAUSED BY A GLITCH
: ON ONE OF THE LINES. IF THEY ARE DIFFERENT THEN A HARD
: MODEM ERROR HAS OCCURED. IN ANY EVENT THE MODEM STATUS
: CHANGE IS LOGGED.
: IF A DATA INT. OCCURS THE ROUTINE PUTS THE DATA AWAY
: IN A BUFFER POINTED TO BY 'RMSGPT' THE MSG. COUNT IS
: DECREMENTED BY ONE BYTE. IF COUNT IS EQUAL TO ZERO AND
: 'BCC' BIT AND 'RXM' BIT IS SET THEN RX IS DISABLED AND
: 'QRX' BIT IS SET. IF COUNT IS ZERO AND 'BCC' BIT IS SET
: BUT 'RXM' BIT IS NOT SET THEN MSG COUNT IS SET TO LENGHT
: RECD IN HEADER AND 'RMSGPT' IS SET TO RX BUFFER LOCATION
: AND 'RXM' BIT IS SET.
: IF COUNT IS EQUAL TO ZERO AND 'BCC' IS NOT SET THEN
: COUNT IS SET TO 2 AND 'RMSGPT' IS SET TO 'BCCW' AND
: 'BCC' BIT IS SET.

```

IF THE OVERRUN ERROR BIT IS SET THEN
AN ERROR IS LOGGED AND 'QRX' IS SET AND THE RX IS DISABLED.

INPUTS:

RMSGPT - ADDRESS OF RX BUFFER
RMSCC - COUNT OF DATA TO BE RXED.

SUBORDINATE ROUTINES USED:

'LOGMSC' - LOG MODEM STATUS CHANGE
'LGDVE' - LOG DEVICE ERROR

--

```

6349
6350
6351
6352
6353
6354
6355
6356
6357
6358
6359
6360
6361
6362
6363
6364
6365
6366
6367
6368
6369
6370
6371
6372
6373
6374
6375
6376
6377
6378
6379
6380
6381
6382
6383
6384
6385
6386
6387
6388
6389
6390
6391
6392
6393 035556
6394 035556
6395 035556 010246
6396 035560 017737 153664 011474
6397 035566 032737 000010 006576
6398 035574 001447
6399 035576 032737 002000 006602
6400 035604 001443
6401 035606 005737 011474
6402 035612 100040
6403 035614 013737 011474 011472
6404 035622 042737 104761 011472

```

BGNSRV DVRXI

```

MOV R2,-(SP) ;SAVE R2
MOV @RXCSR,IRXCSR ;MOV RX CSR TO IMAGE
BIT #MOCHK,PARAM ;ANY MODEM CHANGES TO REPORT
BEQ RXIN21 ;IF NOT IGNORE DS CHANGE.
BIT #INQVR,FLAG ;IS INIT OVER
BEQ RXIN21 ;NO THEN IGNORE DS CHANGE.
TST IRXCSR
BPL RXIN21 ;IF DATA SET CHANGE IS NOT SET BR
MOV IRXCSR,CMODS ;MOV THE NEW MODEM STATUS IN
BIC #104761,CMODS ;CLEAR BITS NOT RELATING TO MODEM STATUS

```

DVRXI::

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 159
DEVICE INTERRUPT SERVICE ROUTINES

```

6405 035630 013737 011472 006542 RXIN2: MOV CMODS,TEMP3
6406 035636 013737 007556 006544      MOV MODS,TEMP4
6407 035644 023737 006544 006542      CMP TEMP4,TEMP3      ;COMPARE OLD TO CURRENT
6408 035652 001406          BEQ 10$      ;INC GLITCH COUNT
6409 035654 005237 011520          INC MHRCNT      ;INC HARD COUNT
6410 035660 012737 016717 006540      MOV #HRDMSG,TEMP2      ;SET UP HARD MESSG.
6411 035666 000405          BR RXIN1
6412 035670 005237 011516          10$: INC MGLCNT      ;INC GLITCH COUNT
6413 035674 012737 016671 006540      MOV #GLMSG,TEMP2      ;SET UP GLITCH
6414 035702 004737 020336          RXIN1: JSR PC,LOGMSC      ;GO LOG MODEM STATUS CHANGE
6415 035706 013737 011472 007556      MOV CMODS,MODS      ;MOVE CURRENT TO OLD
6416
6417      ;TEST FOR DATA
6418
6419 035714 032737 000200 011474 RXIN21: BIT #RXDONE,IRXCSR      ;RX DONE ?
6420 035722 001540          BEQ RXINEX      ;NO,BRANCH
6421 035724 017737 153524 011476      MOV @RXDBUF,IRXDBUF      ;READ DATA
6422 035732 032737 100000 011476      BIT #RERR,IRXDBUF      ;OVERRUN ERROR ?
6423 035740 001055          BNE RXIN3      ;YES,BRANCH
6424
6425      ;GET HERE WITH GOOD DATA
6426
6427 035742 013702 011510          RXIN4: MOV RMSGPT,R2      ;SET RX MESSAGE POINTER
6428 035746 113722 011476          MOVB IRXDBUF,(R2)+      ;STORE DATA AWAY
6429 035752 010237 011510          MOV R2,RMSGPT      ;SAVE UPDATED MESSAGE POINTER
6430
6431
6432 035756 005337 011512          DEC RMSGCC      ;ALL DATA RECEIVED ?
6433 035762 001120          BNE RXINEX      ;NO,BRANCH
6434 035764 032737 000400 006602      BIT #BCC,FLAG      ;CHECK CRC ?
6435 035772 001426          BEQ RXIN6      ;YES,BRANCH
6436 035774 032737 010000 011476      BIT #CRCOK,IRXDBUF      ;CRC GOOD ?
6437 036002 001056          BNE RXIN5      ;YES,BRANCH
6438 036004 013737 011476 006542      MOV IRXDBUF,TEMP3      ;SET UP TO
6439 036012 013737 011474 006544      MOV IRXCSR,TEMP4      ;LOG AND
6440 036020 012737 017205 006540      MOV #DVEM3,TEMP2      ;PRINT CRC ERROR
6441 036026 004737 020162          JSR PC,LGDVE      ;LOG ERROR
6442 036032 005237 006506          INC ERRCNT      ;BUMP COUNT
6443 036036          ERRSOFT 8,DVEM3,ERR13      ;PRINT ERROR TO USER
6444 036036 104457
6445 036040 000010
6446 036042 017205
6447 036044 017636
6448
6449 036046 000463          BR RXIN8      ;DISABLE INTERRUPTS AND EXIT
6450
6451      ;:::IN ORDER TO CHECK CRC, WE MUST READ 2 MORE CHARACTERS(CRC)
6452 036050 052737 000400 006602 RXIN6: BIS #BCC,FLAG      ;SET CRC ALREADY CHECKED FLAG
6453 036056 012737 000002 011512      MOV #2,RMSGCC      ;COUNT TWO CHARACTERS
6454 036064 012737 011514 011510      MOV #BCCW,RMSGPT      ;CRC STORAGE ADDRESS
6455 036072 000454          BR RXINEX      ;EXIT
6456
6457
6458 036074          RXIN3: ;LOG OVERRUN ERROR
6459
6460 036074 012737 017246 006540      MOV #DVEM4,TEMP2

```

```

TRAP      CSERSOFT
.WORD      8
.WORD      DVEM3
.WORD      ERR13

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 160
DEVICE INTERRUPT SERVICE ROUTINES

6461	036102	013737	011476	006542		MOV	IRXDBUF,TEMP3		
6462	036110	013737	011474	006544		MOV	IRXCSR,TEMP4		
6463	036116	004737	020162			JSR	PC,LGDVE		
6464	036122	005237	006506			INC	ERRCNT		
6465	036126					ERRSOFT	9,DVEM4,ERR13		
6466	036126	104457						TRAP	C\$ERSOFT
6467	036130	000011						.WORD	9
6468	036132	017246						.WORD	DVEM4
6469	036134	017636						.WORD	ERR13
6470	036136	000424				BR	RXIN7		
6471									
6472	036140	032737	000040	006602	RXIN5:	BIT	#RXM,FLAG	:IS THE RX M BODY BIT SET	
6473	036146	001020				BNE	RXIN7	:IF YES THEN ALL DONE	
6474	036150	052737	000040	006602		BIS	#RXM,FLAG		
6475	036156	042737	000400	006602		BIC	#BCC,FLAG	:CLEAR BCC AND SET RXM	
6476	036164	013737	006470	011510		MOV	DVRXA,RMSGPT	:MOVE ADDRESS TO POINTER	
6477	036172	013737	002660	011512		MOV	RHDMCC,RMSGCC	:MOVE THE CHAR COUNT IN	
6478	036200	013737	002660	006472		MOV	RHDMCC,DVRCC	:SET THE CC TO AMOUNT IN HEADER	
6479	036206	000406				BR	RXINEX	:AND FINISH.	
6480									
6481	036210	052737	000004	006602	RXIN7:	BIS	#QRX,FLAG	:SET MESSAGE RECEIVED IN FLAG	
6482									
6483	036216	042777	000120	153224	RXIN8:	BIC	#RINTEN+RXENA,QRXCSR	:CLEAR INTAND RX ENABLE	
6484									
6485	036224	012602			RXINEX:	MOV	(SP)+,R2	:RESTORE R2	
6486	036226					ENDSRV			
6487	036226								
6488	036226	000002						L10020:	RTI

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 161
DEVICE TRANSMIT INTERRUPT ROUTINE

.SBTTL DEVICE TRANSMIT INTERRUPT ROUTINE

++
FUNCTIONAL DESCRIPTION:
DEVICE TRANSMIT INT. ROUTINE

WHEN A TRANSMIT BUFFER EMPTY CAUSES AN INTERRUPT TO OCCUR
THE PROGRAM COMES TO THIS ROUTINE.
IF THE SYNC COUNT 'SYNCC' IS NON ZERO TSOM IS SET
A SYNC CHAR IS LOADED TO TXDBUF AND THE SYNC COUNT IS
DECREMENTED.

IF THE SYNC COUNT IS ZERO TSOM AND TEOM ARE RESET
AND THE 'PAD' BIT IN FLAG WORD IS CHECKED IF IT IS
SET THEN A PAD(377) CHAR IS LOADED TO TXDBUF AND TX
INTERRUPT ENABLE IS CLEARD.

IF THE SYNC COUNT IS ZERO AND THE 'PAD' FLAG IS
CLEAR THEN A BYTE IS PUT IN TXDBUF FROM THE ADDRESS
IN MSGPTR AND THE MSG COUNT IS DECREMENTED

IF THE MSG COUNT GOES TO ZERO THE 'TXM' BIT IS
CHECKED IF IT IS SET THE 'PAD' FLAG IS SET
IF IT IS CLEAR THEN IT GETS SET AND MSGPTR IS
LOADED WITH THE ADDRESS OF TXBUF AND THE MSG
COUNT IS LOADED WITH THE COUNT OF THE MSG TO
BE TRANSMITTED.

INPUTS:

MSGPTR - IS SET TO THE ADDRESS OF THE MSG OR HEADER TO BE TX'D
MSGCC - IS SET TO THE COUNT OF MSG TO BE TX'D

OUTPUTS:

QTX - THIS BIT IS SET WHEN MSG IS TX'D OK.

--

6524	036230					BGNSRV	DVTXI		
6525	036230							DVTXI::	
6526	036230	010246				MOV	R2,-(SP)	:SAVE R2	
6527	036232	005737	011504			TST	SYNCC	:ANY SYNCs TO SEND	
6528	036236	001406				BEQ	TXIN1	:IF NOT GO TO 1	
6529	036240	013777	011506	153212		MOV	SYNCC,@TXDBUF	:ELSE SET TSOM AND SYNC WORD	
6530	036246	005337	011504			DEC	SYNCC	:DEC SYNC COUNT	
6531	036252	001065				BNE	TXINEX	:IF NOT ZERO EXIT	
6532	036254	032737	001000	006602	TXIN1:	BIT	#PAD,FLAG	:IS THE PAD BIT SET	
6533	036262	001414				BEQ	TXIN2	:GO TO 2 IF NOT SET	
6534	036264	012777	000377	153166		MOV	#377,@TXDBUF	:LOAD FF TO TX DATA REG.	
6535	036272	042777	000120	153156		BIC	#TINTEN!SEND,@TXCSR	:CLEAR TX INT ENABLE	
6536	036300	005237	037126			INC	TXREADY	:TELL PROTOCOL MODULE WE'RE DONE	
6537	036304	052737	000010	006602		BIS	#QTX,FLAG	:SET THE TX COMPLETE	
6538	036312	000445				BR	TXINEX	:AND EXIT	
6539	036314	005737	011502		TXIN2:	TST	MSGCC	:ALL DATA SENT ?	
6540	036320	001416				BEQ	TXIN4	:YES,BRANCH	
6541	036322	005037	006556			CLR	DATAWORD	:BE SURE ITS CLEAR	
6542	036326	013702	011500			MOV	MSGPTR,R2	:LOAD R2 WITH TX BUFFER POINTER ADDR.	
6543	036332	112237	006556			MOVB	(R2)+,DATAWORD	:PUT DATA IN LOW BYTE	
6544	036336	010237	011500			MOV	R2,MSGPTR	:RESTORE UPDATED POINTER	

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 162
DEVICE TRANSMIT INTERRUPT ROUTINE

6545	036342	013777	006556	153110		MOV	DATAWORD,@TXDBUF	;HI BYTE TSOM=0.. LO BYTE = DATA
6546	036350	005337	011502			DEC	MSGCC	;BUMP CHAR COUNT
6547	036354	000424				BR	TXINEX	
6548	036356	012777	001000	153074	TXIN4:	MOV	#TEOM,@TXDBUF	;SEND CRC CHARACTER
6549	036364	032737	000020	006602		BIT	#TXM,FLAG	;IS THIS THE END OF DATA MSG.
6550	036372	001012				BNE	TXIN3	;IF SO SET THE PAD BIT
6551	036374	052737	000020	006602		BIS	#TXM,FLAG	;IF NOT MUST BE END OF HEADER
6552	036402	013737	006454	011500		MOV	DVTXA,MSGPTR	;SO SET UP MSGPTR FOR MSG
6553	036410	013737	006456	011502		MOV	DVTCC,MSGCC	;AND THE CC FOR MSG.
6554	036416	000403				BR	TXINEX	
6555	036420	052737	001000	006602	TXIN3:	BIS	#PAD,FLAG	;SET THE PAD BIT
6556								
6557	036426	012602			TXINEX:	MOV	(SP)+,R2	;RESTORE R2
6558	036430					ENDSRV		
6559	036430							
6560	036430	000002						

L10021: RTI

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 163
DEVICE TRANSMIT CONTROL MSG

```

6561 .SBTTL                                DEVICE TRANSMIT CONTROL MSG
6562 ++
6563 : FUNCTIONAL DESCRIPTION:
6564 : THIS ROUTINE DOES THE FOLLOWING
6565 : QUES A RX SPACE AT RHDMSG+1
6566 : QUES A TX MSG FROM HDMSG+1
6567 : CHECKS FOR A TIMER EXPIRED
6568 : IF EXPIRED RETURN TO CALLER
6569 : ELSE CHECK FOR A TX MSG COMPLETED
6570 : IF TX COMPLETED CHECK FOR RX COMPLETED
6571 : ELSE RECHECK TIMER AND TX COMPLETED UNTIL
6572 : EITHER TX COMPLETE OR TIME OUT
6573 : IF TX COMPLETE AND RX NOT COMPLETE THEN
6574 : REQUE TX MSG.
6575 : ELSE IF RX COMPLETE RETURN.
6576
6577 : INPUTS:
6578 :     TXM                - SET IN FLAG WORD
6579 :     HDMSG+2            - TYPE OF CONTROL MSG..
6580
6581 : SUBORDINATE ROUTINES USED:
6582 :     "CLRRTS"          - CLEAR REQUEST TO SEND IF HALF DUP.
6583 : CALLING SEQUENCE:
6584 :     JSR      PC,DVIN31
6585 : RETURN:
6586 :     RETURN TO CALLER IF SOMETHING RX'D OR TIMER OUT.
6587 :--
6588
6589
6590 036432 042737 000404 006602 DVIN31: BIC      #QRX!#BCC,FLAG      ;CLEAR RX COMPLETE & CRC ALREADY CHECK
6591
6592 036440 012737 002657 011510      MOV      #RHDMSG+1,RMSGPT      ;SET UP POINTER
6593 036446 013737 002654 011512      MOV      HDMC,RMSGCC      ;AND CC
6594
6595
6596 036454 052777 000560 152766      BIS      #RINTEN!RXENA!DSITEN!STRIP,@RXCSR ;TURN ON RX
6597
6598      ;SET UP TXMITTER TO SEND
6599
6600 036462 004737 036576      DVIN32: JSR      PC,CTSSR      ;SET RTS .
6601 036466 042737 001010 006602      BIC      #QTX!PAD,FLAG      ;CLEAR TX COMPT FLAG.
6602 036474 012737 002645 011500      MOV      #HDMSG+1,MSGPTR      ;MOVE THE CURRENT POINTER TO MSGPTR.
6603 036502 013737 002654 011502      MOV      HDMC,MSGCC
6604 036510 012737 000010 011504      MOV      #8.,SYNCC      ;SET UP SYNC COUNT
6605 036516 052777 000120 152732      BIS      #SEND!TINTEN,@TXCSR      ;TURN ON TX
6606
6607      ;NOW WAIT FOR TIME OUT OR TX COMPLETE
6608
6609 036524      DVIN35: BREAK
6610 036524 104422      TRAP      C$BRK
6611 036526 005737 006646      TST      TIMERS      ;IS IT TIMED OUT
6612 036532 001420      BEQ      DVIN34      ;IF YES EXIT
6613 036534 032737 000010 006602      BIT      #QTX,FLAG      ;IS TX DONE
6614 036542 001770      BEQ      DVIN35      ;IF NOT GO BACK AND CK TIME OUT
6615 036544 004737 037010      JSR      PC,CLRRTS      ;CLEAR RTS IF HALF DUPLEX
6616 036550 012737 002000 006536      MOV      #2000,TEMP1      ;WAIT FOR RX TO COMPLETE

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 164
DEVICE TRANSMIT CONTROL MSG

6617	036556	005337	006536	DVIN36: DEC	TEMP1	:BUMP COUNTER
6618	036562	001375		BNE	DVIN36	:DO IT AGAIN
6619	036564	032737	000004 006602	BIT	#QRX,FLAG	:DID WE RX ANYTHING
6620	036572	001733		BEQ	DVIN32	:IF NOT RETRANSMIT LAST
6621	036574	000207		DVIN34: RTS	PC	:RETURN TO CALLER
6622						

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 165
DEVICE RTS TO CTS DELAY

6623
6624
6625
6626
6627
6628
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

```

.SBTTL                                DEVICE RTS TO CTS DELAY
++
: FUNCTIONAL DESCRIPTION:
: CTSSR--THIS ROUTINE SETS DATA TERMINAL READY AND
: REQUEST TO SEND TO THE MODEM AND CHECKS FOR
: CLEAR TO SEND TO COME BACK.
: IF CTS DOES NOT COME BACK BEFORE TIMER EXPIRES
: AND ERROR IS REPORTED AND WE TRY AGAIN.
: THE ROUTINE IS SKIPPED IF INTERNAL LOOP IS SET.
:
: OUTPUTS:
:
: SUBORDINATE ROUTINES USED:
:   'LGDVE' - LOG DEVICE ERROR
: CALLING SEQUENCE:
:   JSR      PC,CTSSR
:--

CTSSR:  CMP      #1,MLTYP              ;IS THIS TTL LOOP
        BEQ      DVTXR9              ;BR IF YES
        ;SET DTR,RTS AND WAIT FOR CTS

DVTXR3: BIT      #FIRST,FLAG
        BNE      CTSS3              ;IF NOT FIRST TIME SKIP DELY
        MOV      #1,TEMP
CTSS4:  INC      TEMP
        BREAK

        TST      TEMP
        BNE      CTSS4              ;IF NOT ZERO GO BACK
        BIS      #FIRST,FLAG        ;SET FIRST FLAG.
CTSS3:  MOV      #1000,TIMER1        ;SET UP TIMER FOR 1000 TICKS
        TST      FHDPLX             ;FULL DUPLEX ?
        BNE      CTSS7              ;YES,BRANCH
10$:    CALL     DVMODC              ;GET MODEM STATUS
        BIT      #DCD,MODS          ;CARRIER DETECTED?
        BEQ      CTSS7              ;NO,BRANCH
        TST      TIMER1             ;TIME DONE ?
        BEQ      DVTXR4              ;YES,BRANCH
        BR       10$                ;TRY AGAIN

CTSS7:  BIS      #DTR!RTS,@RXCSR    ;SET REQUEST TO SEND AND
        ;DATA TERMINAL READY
        MOV      #1000 ,TIMER1      ;SET UP TIMER
DVTXR2: BREAK

        BIT      #CTS,@RXCSR        ;IS CLEAR TO SEND BACK
        BNE      DVTXR1              ;BR. IF CTS IS SET
        TST      TIMER1             ;ELSE TEST IF TIME EXPIRED
        BNE      DVTXR2              ;BR IF TIME NOT EXPRIED.

        ;SET ERROR FOR NO CTS

```

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 166
DEVICE RTS TO CTS DELAY

[illegible]

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 167
DEVICE CLEAR REQUEST TO SEND

DEVICE CLEAR REQUEST TO SEND

6693
6694
6695
6696
6697
6698
6699
6700
6701
6702
6703
6704
6705
6706
6707
6708
6709
6710
6711
6712

.SBTTL

..**

..FUNCTIONAL DESCRIPTION:

..THIS ROUTINE CLEARS REQUEST TO SEND IF
..IN HALF DUPLEX MODE OR MULTI-POINT.(WITH PROTOCOL)

..CALLING SEQUENCE:

..JSR PC,CLRRTS

..--

CLRRTS:

TST

MPPTP

:MULTI-POINT ?

BNE

20\$

:YES,BRANCH

TST

FHDPLX

:IS THIS FULL DUPLEX

BNE

DVTR5

:BRANCH IF YES

20\$:

MOV

@TXCSR,TEMP2

:GET RX STATUS

BIT

#TXACT,TEMP2

:ALL DATA SENT ?

BNE

20\$

:NO,WAIT

BIC

#RTS,@RXCSR

:CLEAR REQUEST TO SEND

DVTR5:

RTS

PC

:RETURN TO CALLER

037010
037010 005737 037132
037014 001003
037016 005737 006574
037022 001012
037024 017737 152426 006540
037032 032737 001000 006540
037040 001371
037042 042777 000004 152400
037050 000207

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 168
DEVICE CLEAR REQUEST TO SEND

```

6713
6714
6715      .SBTTL  DDCMP PROTOCOL MODULE
6716      ;*****
6717
6718      ; DCLT DDCMP PROTOCOL MODULE
6719
6720      .EVEN
6721      ;; LOCAL STORAGE
6722
6723      ;; TABLE OF STATISTICS AND ERRORS
6724      ;;: NOTE: KEEP THE VARIABLES TOGETHER AND IN SEQUENCE
6725      ;;: OTHERWISE THE RPT> ROUTINE WILL PRINT WRONG INFO.
6726
6727      037052  000000      PRSTAT: .WORD 0
6728
6729
6730
6731
6732
6733
6734
6735
6736
6737
6738
6739
6740      037054  000000      TMESTX: .WORD 0
6741      037056  000000      TMESRX: .WORD 0
6742
6743      037060      000      N:      .BYTE 0
6744
6745      037061      000      A:      .BYTE 0
6746
6747
6748      037062      000      T:      .BYTE 0
6749      037063      000      X:      .BYTE 0
6750
6751      037064      000      R:      .BYTE 0
6752      037065      000      TRIBN: .BYTE 0
6753
6754      ;; ERROR COUNTERS
6755      037066      000      REMTMO: .BYTE 0
6756      037067      000      GLOBCC: .BYTE 0
6757
6758      037070      000      REANAK: .BYTE 0
6759      037071      000      SELTHER: .BYTE 0
6760
6761      037072      000      RXTHER: .BYTE 0
6762      037073      000      TXTHER: .BYTE 0

```

```

;STATUS FLAGS
;BIT0 = BCCOK
;BIT1 = BCCBAD
;BIT2 = SNAK
;BIT3 = SACK
;BIT4 = SDATA
;SPARE
;BIT6 = RXD
;BIT7 = SPARE
;BIT8 = NAKRX
;BIT9 = MYDATA
;BIT10 = SSTACK
;BIT11 = SSTART
;TOTAL DATA MESSAGES TRANSMITTED(16 BIT COUNTER
;TOTAL DATA MESSAGES RECEIVED(16 BIT COUNTER)
;# OF HIGHEST SEQUENTIAL DATA MESSAGE TRANS
;: MITTED BY THIS STATION
;# OF THE HIGHEST SEQUENTIAL DATA MESSAGE
;: THAT HAS BEEN ACKNOWLEDGE TO THIS STATION
;# OF THE NEXT DATA MESSAGE TO BE TRANSMITTED
;LAST MESSAGE NUMBER TRANSMITTED
;LAST MESSAGE RECEIVED
;TRIB ADDRESS PT TO PT = 1
;REMOTE REPLY TIMEOUTS(ACKS SENT NUM=R)
;GLOBAL CRC ERRORS
;REASON FOR LAST NAK SENT
;SELECTION THRESHOLD ERROR
;RECEIVE THRESHOLD ERRORS
;TRANSMIT THRESHOLD ERRORS

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 169
DDCMP PROTOCOL MODULE

6763					
6764					
6765	037074	000	DEROUT: .BYTE 0		:DATA ERRORS OUTBOUND (NAKS RECEIVED
6766	037075	000	OUTMASK: .BYTE 0		: REASONS = 1,2,OR 3)
6767					: MASK VALUES -- BIT0 = HEADER CRC ERROR
6768					: -- BIT1 = DATA FIELD CRC ERROR
6769					: -- BIT2 = REP RESPONSE NUM<>R
6770					:
6771	037076	000	DERIN: .BYTE 0		:DATA ERRORS INBOUND (NAKS TRANSMITTED
6772	037077	000	INMASK: .BYTE 0		: REASONS = 1,2,OR 3)
6773					: MASK VALUES -- BIT0 = HEADER CRC ERROR
6774					: -- BIT1 = DATA FIELD CRC ERROR
6775					: -- BIT2 = REP RESPONSE NUM<>R
6776					:
6777	037100	000	LBUFER: .BYTE 0		:LOCAL BUFFER ERRORS (NAKS SENT
6778	037101	000	LBMASK: .BYTE 0		: REASONS = 8. OR 16.)
6779					: MASK VALUES -- BIT0 = BUFFER NOT AVAILABLE
6780					: -- BIT1 = MESSAGE TOO LONG
6781					:
6782	037102	000	RBUFER: .BYTE 0		:REMOTE BUFFER ERRORS (NAKS RECEIVED
6783	037103	000	RBMASK: .BYTE 0		: REASONS 8. OR 16.)
6784					: MASK VALUES -- BIT0 = BUFFER NOT AVAILABLE
6785					: -- BIT1 = MESSAGE TOO LONG
6786					:
6787	037104	000	RMSTER: .BYTE 0		:REMOTE STATION ERRORS (NAKS RECEIVED
6788	037105	000	RMMASK: .BYTE 0		: REASON 9. OR 17.)
6789					: MASK VALUES-- BIT0 = RECEIVER OVERRUN
6790					: BIT1 = FORMAT ERROR
6791					:
6792	037106	000	LOSTER: .BYTE 0		:LOCAL STATION ERRORS (NAKS SENT
6793	037107	000	LSMASK: .BYTE 0		: REASON 9. OR 17.)
6794					: MASK VALUES -- BIT0 = RECEIVER OVERRUN
6795					: -- BIT1 = FORMAT ERROR
6796					:
6797	037110	000000	RXTXTE: .WORD 0		:RX AND TX THRESHOLD ERRORS (OVERFLOWS)
6798	037112	000	SPARE0: .BYTE 0		
6799	037113	000	SPARE1: .BYTE 0		
6800	037114	000000	PROEND: .WORD 0		:END OF PROTOCOL COUNTERS
6801	037116	000000	IMFLAG: .WORD 0		: IMAGE OF MAIN CODE FLAG WORD
6802	037120	000000	RXPRC: .WORD 0		: -1 = MESSAGE RX'ED & 'ACK' SENT
6803	037122	000000	TXPRC: .WORD 0		: -1 = MESSAGE TX'ED & 'ACK' RECEIVED
6804	037124	000000	ASTRT: .WORD 0		: -1 = STACK SENT
6805	037126	000000	TXREADY: .WORD 0		: 1 = READY TO SEND ANOTHER MESSAGE
6806	037130	000000	PRUN: .WORD 0		: 1 = PROTOCOL RUNNING. USED IN THIS MODULE
6807	037132	000000	MPPTP: .WORD 0		: 1 = MULTI POINT NETWORK
6808	037134	000000	SELECT: .WORD 0		: 1 = THIS STATION CAN NOW TRANSMIT (HALF/DUPLEX)
6809	037136	000000	IMPRSTAT: .WORD 0		: COPY OF PROTOCOL STATUS WORD
6810	037140	000000	PRFLAG: .WORD 0		: USED TO COMMUNICATE WITH RX INTER. ROUTINE
6811	037142	000000	HDXMTP: .WORD 0		: 1 = HALF DUPLEX OR MULTI-POINT
6812	037144	000000	PRTEMP: .WORD 0		: TEMPORARY WORK LOCATION
6813	037146	000000	TURNON: .WORD 0		: 1 = RECEIVER IS ALREADY ON
6814	037150	000000	TIMEOUT: .WORD 0		: 20 = PRINT 'TX OR RX NOT COMPLETE'
6815					

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 170
DDCMP PROTOCOL MODULE

6816
6817
6818
6819
6820
6821
6822
6823
6824
6825
6826
6827
6828
6829
6830
6831
6832
6833
6834
6835
6836
6837
6838
6839
6840
6841
6842
6843
6844
6845
6846
6847
6848
6849
6850
6851
6852
6853
6854
6855
6856
6857
6858
6859
6860
6861
6862
6863
6864
6865
6866

000001
000002
000003
000010
000011
000020
000021

000004
000001
000002
000002
000001

000201
000144
000005

000001
000002
000003
000006
000007

000001
000002
000004
000010
000020
000100
000400
001000
002000
004000

:: NAK REASONS VALUES AS USED IN NAK CONTROL MESSAGES

HEADBCC = 1	:HEADER BCC ERROR
DATABCC = 2	:DATA BCC ERROR
REPRESENT = 3	:REP RESPONSE
BUFFNA = 10	:BUFFER TEMPORARILY NOT AVAILABLE
RXOVRUN = 11	:RECEIVER OVERRUN
MESLONG = 20	:MESSAGE TOO LONG
FORMERR = 21	:HEADER FORMAT ERROR

:: ADDITIONAL NAK BIT MASKS AS USED IN COUNTERS

REPMSK = BIT2	:REPLY RESPONSE
RXOVMSK = BIT0	:RECEIVER OVERRUN
FMTMSK = BIT1	:FORMAT ERROR
MTLMSK = BIT1	:MESSAGE TOO LONG
BNAMSK = BIT0	:BUFFER NOT AVAILABLE

:: MESSAGE TYPE DEFINITIONS

SOH = 201	:DATA MESSAGE
MAINT = 144	:MAINTENANCE MESSAGE
ENQ = 5	:CONTROL MESSAGE

:: SUBTYPES OF CONTROL MESSAGES

ACK = 1	:ACKNOWLEDGE MESSAGE
NAK = 2	:NEGATIVE ACKNOWLEDGE MESSAGE
REP = 3	:REPLY TO MESSAGE NUMBER
STRT = 6	:START MESSAGE
STACK = 7	:START ACKNOWLEDGE MESSAGE

:: STATUS WORD BIT DEFINITIONS

BCCOK = BIT0	:BCC CHECKED GOOD
BCCBAD = BIT1	:BCC CHECKED BAD
SACK = BIT2	:SEND ACK
SNACK = BIT3	:SEND NAK
SDATA = BIT4	:SEND DATA
RXD = BIT6	:RECEIVER DONE
NAKRX = BIT8	:NAK RECEIVED
MYDATA = BIT9	:MY DATA
SSTACK = BIT10	:SEND START ACKNOWLEDGE
SSTART = BIT11	:SEND START

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 171
DDCMP PROTOCOL MODULE

6867
6868
6869
6870
6871
6872
6873
6874
6875
6876 037152 037262
6877 037154 037311
6878 037156 037347
6879 037160 037403
6880 037162 037475
6881 037164 037561
6882 037166 037641
6883 037170 037725
6884 037172 040007
6885 037174 040071
6886 037176 040166
6887 037200 040262
6888 037202 040360
6889 037204 040456
6890 037206 040544
6891 037210 040631
6892
6893
6894
6895
6896
6897
6898
6899 037212 021642
6900 037214 021642
6901 037216 021642
6902 037220 021670
6903 037222 021670
6904 037224 021670
6905 037226 021670
6906 037230 021670
6907 037232 021670
6908 037234 021726
6909 037236 021726
6910 037240 021726
6911 037242 021726
6912 037244 021726
6913 037246 021726
6914 037250 021642
6915
6916 037252 000000
6917 037254 000000
6918 037256 000000
6919 037260 000000
6920

: THE BELOW TABLES AND ASCII MESSAGES ARE USED IN DCLT
: REPORTING OF ERROR COUNTERS. THEY MUST REMAIN IN THE
: CURRENT SEQUENCE ELSE WE'LL BE REPORTING ERRONEOUS
: DATA.
:*****

STALST: .WORD STA0A ; POINTER FOR OFFSET 0 ASCII
 .WORD STA1A ; POINTER FOR OFFSET 1 ASCII
 .WORD STA2A ; POINTER FOR OFFSET 2 ASCII
 .WORD STA3A ; POINTER FOR OFFSET 3 ASCII
 .WORD STA4A ; POINTER FOR OFFSET 4 ASCII
 .WORD STA5A ; POINTER FOR OFFSET 5 ASCII
 .WORD STA6A ; POINTER FOR OFFSET 6 ASCII
 .WORD STA7A ; POINTER FOR OFFSET 7 ASCII
 .WORD STA10A ; POINTER FOR OFFSET 10 ASCII
 .WORD STA11A ; POINTER FOR OFFSET 11 ASCII
 .WORD STA12A ; POINTER FOR OFFSET 12 ASCII
 .WORD STA13A ; POINTER FOR OFFSET 13 ASCII
 .WORD STA14A ; POINTER FOR OFFSET 14 ASCII
 .WORD STA15A ; POINTER FOR OFFSET 15 ASCII
 .WORD STA16A ; POINTER FOR OFFSET 16 ASCII
 .WORD STA17A ; POINTER FOR OFFSET 17 ASCII

;TABLE FOR PRINT ROUTINES
;PRIW: WORD ROUTINE
;PRI8B: BYTE/BYTE ROUTINE
;PRI8S: BYTE SPECIAL ROUTINE

STAIND: .WORD PRIW
 .WORD PRIW
 .WORD PRIW
 .WORD PRI8B
 .WORD PRI8B
 .WORD PRI8B
 .WORD PRI8B
 .WORD PRI8B
 .WORD PRI8B
 .WORD PRI8B
 .WORD PRI8S
 .WORD PRI8S
 .WORD PRI8S
 .WORD PRI8S
 .WORD PRI8S
 .WORD PRIW

LAST: .WORD 0 ;LAST MESSAGE TO PRINT
FIR: .WORD 0 ;FIRST MESSAGE TO PRINT
MES: .WORD 0 ;HOLDS MESSAGE
MESDATA: .WORD 0 ;DATA PART OF MESSAGE

CZDCLB DUP-11 DATA COMPI. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 172
DDCMP PROTOCOL MODULE

6921
6922
6923
6924
6925
6926

```
*****
THE BELOW ASCII MESSAGES USED IN 'RPT>' LEVEL OF DCLT
*****
.NLIST BEX
STA0A: .ASCII /%N%06%S2%ASTATUS FLAGS/
STA1A: .ASCII /%N%06%S2%ADATA MSGS. TX'MITTD/
STA2A: .ASCII /%N%06%S2%ADATA MSGS. RX'CVD/
STA3A: .ASCII /%N%03%S5%AHIGHEST MSG # TX'D%N%03%S5%AHIGHEST MSG # ACK'D/
STA4A: .ASCII /%N%03%S5%ANEXT MSG # TO TX%N%03%S5%ALAST MSG # TX'D/
STA5A: .ASCII /%N%03%S5%AHIGHEST MSG # RX'D%N%03%S5%ATRIB ADDR/
STA6A: .ASCII /%N%03%S5%AREMOTE TIME OUTS%N%03%S5%AGLOBAL CRC ERRS/
STA7A: .ASCII /%N%03%S5%ANAK REASON%N%03%S5%ASELECT THRESH. ERRS/
STA10A: .ASCII /%N%03%S5%ARX THRESH ERRS%N%03%S5%ATX THRESH. ERRS/
STA11A: .ASCII /%N%03%S5%ADATA ERRORS OUT%N%S8%AHBCC %01% BCC %01% REP %01/
STA12A: .ASCII /%N%03%S5%ADATA ERRORS IN%N%S8%AHBCC %01% BCC %01% REP %01/
STA13A: .ASCII /%N%03%S5%ALOCAL BUFFER ERRS%N%S8% NO BUFF %01% TOO BIG %01/
STA14A: .ASCII /%N%03%S5%AREMOTE BUFFER ERRS%N%S8% NO BUFF %01% TOO BIG %01/
STA15A: .ASCII /%N%03%S5%AREMOTE STA ERRS%N%S8%AQVRN %01% FORMAT %01/
STA16A: .ASCII /%N%03%S5%ALOCAL STA ERRS%N%S8%AQVRN %01% FORMAT %01/
STA17A: .ASCII /%N%06%S2%ATX & RX THRESHOLD ERRORS(OVERFLOW)/
.EVEN
```

.LIST BEX

6927
6928
6929
6930
6931
6932
6933
6934
6935
6936
6937
6938

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 173
DDCMP PROTOCOL MODULE

```

6939
6940
6941
6942
6943
6944
6945
6946
6947
6948
6949
6950
6951
6952
6953 040706 013737 006602 037116 PROTOC: MOV FLAG,IMFLAG ;SAVE COPY OF MAIN CODE 'FLAG' VARIABLE
6954 040714 012737 000001 037126 MOV #1,TXREADY ;INIT TRANSMITTER DONE FLAG
6955 040722 005037 037120 CLR RXPRC ;INIT RX PROCOTOL DONE
6956 040726 005037 037122 CLR TXPRC ;INIT TX PROCOTOL DONE
6957 040732 005037 037150 CLR TIMEOUT ;INIT PRINT TIMER
6958 040736 032737 000100 006576 BIT #PRORUN,PARAM ;PROTOCOL RUNNING ?
6959 040744 001067 BNE 7$ ;YES,BRANCH
6960
6961
6962
6963 040746 012746 000240
6964 040752 012746 044510
6965 040756 013746 011462
6966 040762 012746 000003
6967 040766 104437
6968 040770 062706 000010
6969 040774 005037 037142
6970 041000 005737 006574
6971 041004 001403
6972 041006 005737 037132
6973 041012 001403
6974 041014 012737 000001 037142 2$: MOV #1,HDXMTP ;SET HALF DUPLEX/MULTI-POINT
6975
6976 041022 012737 000036 006646 3$: MOV #30.,TIMERS ;30 SECONDS TO START
6977 041030 012737 000001 037134 MOV #1,SELECT ;INIT SELECT
6978 041036 005037 037146 CLR TURNON ;INIT YET ANOTHER FLAG
6979 041042 005037 037130 CLR PRUN ;INIT ANOTHER FLAG
6980 041046 005037 037124 CLR ASTRT ;INIT 'STACK SENT' FLAG
6981 041052 005037 037052 CLR PRSTAT ;INIT STATUS WORD
6982 041056 004737 041734 JSR PC,PROINT ;INIT PROTOCOL COUNTERS AND VARIABLES
6983 041062 005737 037132 TST MPPTP ;MULTI - POINT MODE ?
6984 041066 001005 BNE 4$ ;YES,BRANCH
6985 041070 052737 004000 037052 BIS #SSTART,PRSTAT ;TELL TX ROUTINE TO SEND 'START'
6986 041076 004737 045336 JSR PC,TXPROTO ;GO SEND IT
6987 041102 004737 042104 4$: JSR PC,RXPROTO ;GO WAIT FOR 'STACK' OR 'START'
6988 041106 032737 000100 006576 BIT #PRORUN,PARAM ;DID PROTOCOL START ?
6989 041114 001742 BEQ 3$ ;NO,TRY AGAIN
6990 041116 012737 000001 037130 MOV #1,PRUN ;THIS FLAG USED IN RXPROTO ROUTINE
6991
6992
6993 041124 012737 000003 006646 7$: MOV #3.,TIMERS ;SET UP TIMER
6994 041132 005737 037142 TST HDXMTP ;HALF DUPLEX OR MULTI - POINT?

```

: PROTOCOL ROUTINE:
: DESCRIPTION: IF THE USER SPECIFIES THE '/PROTOCOL' SWITCH THIS
: ROUTINE WILL BE CALLED. THIS ROUTINE DECIDES IF
: WE ARE TRANSMITTING AND/OR RECEIVING AND CALLS
: THE NECESSARY ROUTINES.
: THIS CODE WAS WRITTEN ONLY TO BE USED WITH DCLT.
:*****

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 174
DDCMP PROTOCOL MODULE

```

6995 041136 001076          BNE      PROHDX          ;YES,BRANCH
6996
6997
6998 041140 022737 000003 006570  ;; IF FULL DUPLEX AND ACTIVE MODE-- JUMP
6999 041146 001440          CMP      #ACT,MODTYP      ;ACTIVE MODE?
7000          BEQ      200$          ;YES, BRANCH
7001
7002          ;; PROTOCOL IS RUNNING -- LINK IS HOT SO SEND DATA
7003 041150 032737 000010 037116 10$:  BIT      #QTX,IMFLAG      ;TRANSMITTING A MESSAGE ?
7004 041156 001414          BEQ      100$          ;NO,BRANCH
7005 041160 052737 000020 037052 20$:  BIS      #SDATA,PRSTAT      ;SEND DATA FLAG
7006 041166 004737 045336          CALL     TXPROTO          ;GO SEND THE MESSAGE
7007 041172 004737 042104          CALL     RXPROTO          ;CHECK THE REPLY
7008 041176 005737 037122          TST      TXPRC          ;MESSAGE TRANSMITTED & 'ACK'ED'?
7009 041202 001766          BEQ      20$          ;NO,BRANCH
7010 041204 005237 037054          INC      TMESTX          ;BUMP 'TOTAL MESSAGES TRANSMITTED' COUNTER
7011
7012 041210 005737 037120          100$:  TST      RXPRC          ;RECEIVE PROTOCOL FINISHED ?
7013 041214 001011          BNE      110$          ;YES,BRANCH
7014 041216 032737 000004 037116      BIT      #QRX,IMFLAG      ;RECEIVING A MESSAGE ?
7015 041224 001002          BNE      105$          ;YES,BRANCH
7016 041226 000137 041674          JMP      PROTEX          ;EXIT
7017
7018
7019 041232 004737 042104          105$:  CALL     RXPROTO          ;GO PROCESS INCOMING MESSAGE
7020 041236 000764          BR       100$          ;SEE IF RECEIVE PROTOCOL COMPLETE
7021 041240 005237 037056          110$:  INC      TMESRX          ;BUMP 'TOTAL MESSAGES RECEIVED' COUNTER
7022 041244 000137 041674          JMP      PROTEX          ;EXIT
7023
7024
7025          ;; ACTIVE MODE (FULL DUPLEX AND POINT TO POINT LINKS)
7026
7027 041250 004737 042000          200$:  CALL     RXON          ;TURN ON RECEIVER
7028 041254 052737 000020 037052 210$:  BIS      #SDATA,PRSTAT      ;SEND DATA FLAG
7029 041262 004737 045336          CALL     TXPROTO          ;DO SEND DATA MESSAGE
7030 041266 004737 042104          215$:  CALL     RXPROTO          ;GO PROCESS INCOMING MESSAGE
7031 041272 005737 037122          TST      TXPRC          ;TX PROTOCOL DONE ?
7032 041276 001766          BEQ      210$          ;NO,BRANCH
7033 041300 005737 037120          TST      RXPRC          ;RX PROTOCOL DONE ?
7034 041304 001770          BEQ      215$          ;NO,BRANCH
7035 041306 005237 037056          INC      TMESRX          ;BUMP 'TOTAL MESSAGES RECEIVED'
7036 041312 005237 037054          INC      TMESTX          ;BUMP 'TOTAL MESSAGE SENT' COUNTER
7037
7038          ;; TXREADY SET IN TX INTERRUPT ROUTINE
7039 041316 005737 037126          220$:  TST      TXREADY          ;MESSAGE SENT ?
7040 041322 001775          BEQ      220$          ;NO,BRANCH
7041 041324 004737 042000          CALL     RXON          ;TURN ON RECEIVER
7042 041330 000137 041674          JMP      PROTEX          ;EXIT
7043
7044
7045          ::: THIS ROUTINE(PROHDX) IS USE IN HALF-DUPLEX PT-PT & MTP
7046
7047 041334          PROHDX:
7048 041334 005737 006574          10$:  TST      FHDPLX          ;FULL DUPLEX ?
7049 041340 001072          BNE      PROFDX          ;YES,BRANCH
7050 041342 032737 000010 037116      BIT      #QTX,IMFLAG      ;TRANSMITTING ?

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 175
DDCMP PROTOCOL MODULE

7051	041350	001424			BEQ	100\$	:NO,BRANCH
7052	041352	005737	037134	20\$:	TST	SELECT	:DO WE HAVE THE SELECT BIT ?
7053	041356	001005			BNE	30\$	:YES,BRANCH
7054	041360	004737	042000		CALL	RXON	:TURN ON RX
7055	041364	004737	042110	25\$:	CALL	RXWAIT	:TURN ON RX AND WAIT FOR SELECT BIT
7056	041370	000770			BR	20\$	:DID WE GET THE SELECT BIT ?
7057	041372	052737	000020	037052	30\$:	BIS	#SDATA,PRSTAT
7058	041400	004737	045336		CALL	TXPROTO	:SEND DATA FLAG
7059	041404	004737	042104		CALL	RXPROTO	:GO SENT IT
7060	041410	005737	037122		TST	TXPRC	:CHECK REPLY
7061	041414	001756			BEQ	20\$	:TX PROTOCOL DONE ?
7062	041416	005237	037054		INC	TMESTX	:NO,BRANCH
7063	041422	012737	000001	037122	100\$:	MOV	#1,TXPRC
7064	041430	005737	037120	103\$:	TST	RXPRC	:BUMP TOTAL MESSAGES SENT
7065	041434	001026			BNE	150\$	:SET TX PROTOCOL DONE
7066	041436	032737	000004	037116	BIT	#QRX,IMFLAG	:RX PROTOCOL DONE ?
7067	041444	001002			BNE	110\$	:YES,BRANCH
7068	041446	000137	041674		JMP	PROTEX	:RECEIVING ?
7069							:YES,BRANCH
7070	041452	005737	037134		TST	SELECT	:EXIT
7071	041456	001005		110\$:	BNE	130\$	:WAS THE BALL TOSSED BACK IN OUR COURT ?
7072	041460	004737	042000		CALL	RXON	:HAVE WE RECEIVED THE SELECT BIT YET?
7073	041464	004737	042110	115\$:	CALL	RXWAIT	:YES,BRANCH
7074	041470	000757			BR	103\$	:TURN ON RECEIVER
7075	041472	052737	000004	037052	130\$:	BIS	#SACK,PRSTAT
7076	041500	004737	045336		CALL	TXPROTO	:PROCESS DATA
7077	041504	004737	042104		CALL	RXPROTO	:TRY AGAIN
7078	041510	000747			BR	103\$	:SEND ACK TO TURN THE LINE AROUND
7079	041512	005237	037056	150\$:	INC	TMESRX	:SEND IT
7080	041516	004737	042000		CALL	RXON	:GO RECEIVE THE PENDING MESSAGE
7081	041522	000137	041674		JMP	PROTEX	:BRANCH
7082							:BUMP 'RECIEVED MESSAGE COUNTER'
7083							:TURN ON RX
7084							:EXIT
7085	041526	032737	000010	037116	PROFDX:	BIT	#QTX,IMFLAG
7086	041534	001003			BNE	10\$	:TRANSMITTING ?
7087	041536	012737	000001	037122	MOV	#1,TXPRC	:YES,BRANCH
7088	041544	005737	037120	10\$:	TST	RXPRC	:SET TRANSMIT PROTOCOL COMPLETE
7089	041550	001015			BNE	30\$	:WAS THE 1ST MESSAGE RX'ED DURING STARTUP?
7090	041552	032737	000004	037116	BIT	#QRX,IMFLAG	:YES,BRANCH
7091	041560	001004			BNE	20\$	:RECEIVING ?
7092	041562	012737	000001	037120	MOV	#1,RXPRC	:YES,BRANCH
7093	041570	000410			BR	100\$	:SET RECEIVE PROTOCOL COMPLETE
7094	041572	004737	042104	20\$:	CALL	RXPROTO	:BRANCH
7095	041576	005737	037120		TST	RXPRC	:PROCESS INCOMING MESSAGE
7096	041602	001773			BEQ	20\$	:DONE ?
7097	041604	005237	037056	30\$:	INC	TMESRX	:NO,BRANCH
7098	041610	000400			BR	100\$	:BUMP RX MESSAGE COUNT
7099							:BRANCH
7100	041612	005737	037122	100\$:	TST	TXPRC	:ANYTHING TO SEND ?
7101	041616	001024			BNE	135\$	:NO,BRANCH
7102							
7103	041620	005737	037134	120\$:	TST	SELECT	:DO WE HAVE PERMISSION TO SEND ?
7104	041624	001005			BNE	130\$	:YES,BRANCH
7105	041626	004737	042000		CALL	RXON	:TURN ON TX
7106	041632	004737	042110	125\$:	CALL	RXWAIT	:WAIT ON SELECT BIT

::THIS ROUTINE(PROFDX:) USED WITH FULL DUPLEX-MULTI POINT LINKS

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 177
DDCMP PROTOCOL MODULE

7126
7127
7128
7129
7130
7131
7132
7133
7134
7135
7136
7137
7138
7139
7140
7141
7142
7143
7144
7145
7146
7147
7148
7149
7150
7151
7152
7153
7154
7155
7156
7157
7158
7159
7160
7161
7162
7163
7164
7165
7166
7167
7168
7169
7170
7171
7172
7173
7174

041734 010146
041736 113737 037065 006534
041744 012701 037052
041750 005021
041752 020127 037114
041756 001374
041760 113737 006534 037065
041766 112737 000001 037062
041774 012601
041776 000207

```
*****
:
:      PROTOCOL INIT ROUTINE:
:
:      THIS ROUTINE WILL INITIALIZE THE ERROR COUNTERS AND MESSAGE
:      COUNTERS AS NEEDED FOR PROPER DDCMP PROTOCOL OPERATION.
:      DURING NORMAL OPERATION THIS CODE WILL BE CALLED ONCE FROM
:      PROTOCOL STARTUP ROUTINE.
:
:*****
```

```
PROINT: MOV      R1, -(SP)      ;SAVE R1
          MOV      TRIBN, TEMP   ;SAVE TRIB NUMBER
          MOV      #PRSTAT, R1   ;FIRST LOCATION TO CLEAR
10$:      CLR      (R1)+         ;CLEAR AND INCREMENT
          CMP      R1, #PROEND   ;LAST LOCATION TO CLEAR
          BNE      10$           ;NO BRANCH
20$:      MOV      TEMP, TRIBN   ;RESTORE TRIB #
          MOV      #1, T         ;FIRST MESSAGE # TO BE TRANSMITTED
          MOV      (SP)+, R1     ;RESTORE R1
          RETURN                 ;EXIT
```

```
*****
:      TURN ON RECEIVER ROUTINE:
:
:      DESCRIPTION: THIS ROUTINE SIMPLY ENABLES THE RECEIVER AND
:      INITIALIZES VARIABLES THEN RETURNS.
:
:*****
```

```
RXON:    TST      TURNON        ;RX ALREADY ON ?
          BNE      RXONEX       ;YES, BRANCH
          CLR      PRSTAT       ;INIT STATUS WORD
          CLR      RHDMSG       ;INIT 1ST WORD OF RX BUFFER
          BIC      #QRX!#BCC!#RXM, PRFLAG ;FLAGS USED IN RX INTERRUPT ROUTINE
          BIS      #MYDATA, PRSTAT ;ASSUME MESSAGE FOR ME
          MOV      #RHDMDID, RMSGPT ;BUFFER ADDRESS FOR HEADER PART ON MESSAGE
          MOV      HDMC, RMSGCC  ;INIT CHARACTER COUNT = 6
          BIT      #MOCHK, PARAM ;MODEM CHANGES WANTED ?
          BNE      20$          ;YES, BRANCH
          BIS      #RINTEN!RXENA!STRIP, @RXCSR ;TURN ON RX
          BR       25$          ;BRANCH
20$:      BIS      #RINTEN!RXENA!DSITEN!STRIP, @RXCSR ;TURN ON RX
25$:      MOV      #1, TURNON    ;RX IS ON FLAG
RXONEX:  RETURN
```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 178
DDCMP PROTOCOL MODULE

7175
7176
7177
7178
7179
7180
7181
7182
7183
7184
7185
7186
7187
7188
7189
7190
7191
7192
7193
7194
7195
7196
7197
7198
7199
7200
7201
7202
7203
7204
7205
7206
7207
7208
7209
7210
7211
7212
7213
7214
7215
7216
7217
7218
7219

042104
042104 004737 042000

042110
042110
042110 104422
042112 105737 002657
042116 001007
042120 005737 006646
042124 001371
042126 004737 043732
042132 000137 043730

042136 123727 002657 000005
042144 001003
042146 052737 000040 037140

042154 032737 000003 037052
042162 001021
042164 005737 006646
042170 001004
042172 004737 043732
042176 000137 042104

: RECEIVER PROTOCOL ROUTINE:
:
: DESCRIPTION: THIS ROUTINE WILL PROCESS AN INCOMING MESSAGE
: AND DETERMINE IF IT'S A VALID CONTROL OR DATA
: MESSAGE. IF AN ERROR IS DETECTED THE APPROPRIATE
: ERROR COUNTERS WILL BE UPDATED BY THE ERROR
: ROUTINE.
:
: SUBORDINATE ROUTINES USED: 'TXPROTO'
: 'ERROR PROCESSOR'
:*****

RXPROTO:
CALL RXON ;TURN ON RECEIVER

:: WAIT FOR FIRST CHARACTER TO APPEAR IN RX BUFFER
RXWAIT:
20\$: BREAK ;CHECK FOR ^C TRAP CSBRK

TSTB RHD MID ;FIRST CHARACTER READ ?
BNE 30\$;YES, BRANCH
TST TIMERS ;60 SECONDS ELAPSED ?
BNE 20\$;NO, BRANCH
JSR PC,ERRPRC ;CALL ERROR PROCESSOR
JMP RXPREX ;EXIT

:: IF A CONTROL MESSAGE THEN TELL RX INTR. TO PROCESS HEADER ONLY
30\$: CMPB RHD MID,#ENQ ;CONTROL MESSAGE ?
BNE 40\$;NO, BRANCH
BIS #RXM,PRFLAG ;PROCESS HEADER ONLY

:: WAIT FOR CRC TO BE CHECKED
40\$: BIT #BCCOK!BCCBAD,PRSTAT ;CRC CHECKED ?
BNE 50\$;YES, BRANCH
TST TIMERS ;60 SECONDS ELAPSED ?
BNE 45\$;NO, BRANCH
JSR PC,ERRPRC ;GO PROCESS ERROR
JMP RXPROTO ;TRY AGAIN

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 179
DDCMP PROTOCOL MODULE

```

7220
7221
7222
7223 042202 032737 000010 037052 45$:  BIT    #SNAK,PRSTAT    ;RX OVERRUN ?
7224 042210 001761                BEQ     40$              ;NO,BRANCH
7225 042212 004737 043732          JSR     PC,ERRPRC        ;GO PROCESS ERROR
7226 042216 004737 045336          JSR     PC,TXPROTO        ;GO SEND NAK
7227 042222 000137 042104          JMP     RXPROTO          ;TRY AGAIN
7228
7229
7230 042226 032737 000002 037052 50$:  BIT     #BCCBAD,PRSTAT    ;CRC ERROR ?
7231 042234 001430                BEQ     60$              ;NO,BRANCH
7232 042236 052737 000010 037052      BIS     #SNAK,PRSTAT    ;SET SNAK (SEND NAK)
7233 042244 152737 000001 037077      BISB    #HEADBCC,INMASK    ;SET THE MASK
7234 042252 112737 000001 037070      MOVB    #HEADBCC,REANAK    ;NAK REASON = 1
7235 042260 105237 037076          INCB     DERIN              ;LOG DATA ERROR INBOUND
7236 042264 001003                BNE     55$              ;BRANCH IF NOT OVERFLOW
7237 042266 112737 000377 037076      MOVB    #377,DERIN          ;LATCH COUNTER AT 255.
7238 042274 004737 043732          JSR     PC,ERRPRC        ;GO PROCESS ERROR
7239 042300 012737 000001 037134      MOV     #1,SELECT        ;IF HALF/DUPLEX, WE ASSUME S-BIT WAS SET
7240 042306 004737 045336          JSR     PC,TXPROTO        ;GO SEND NAK
7241 042312 000137 042104          JMP     RXPROTO          ;TRY AGAIN
7242
7243
7244 042316 123737 037065 002664 60$:  CMPB    TRIBN,RHDADR        ;MESSAGE- IS IT FOR ME ?
7245 042324 001422                BEQ     70$              ;MY ADDRESS ?
7246
7247
7248 042326 042737 001000 037052 62$:  BIC     #MYDATA,PRSTAT    ;MESSAGE NOT FOR ME
7249 042334 032737 000100 037052      BIT     #RXD,PRSTAT        ;RECEIVER DONE ?
7250 042342 001003                BNE     65$              ;YES,BRANCH
7251 042344 005737 006646          TST     TIMERS              ;HAVE WE DAWDLED LONG ENOUGH ?
7252 042350 001366                BNE     62$              ;NO,BRANCH
7253
7254 042352 032737 000001 037052 65$:  BIT     #BCCOK,PRSTAT    ;DATA CRC OK ?
7255 042360 001002                BNE     67$              ;YES,BRANCH
7256 042362 105237 037067          INCB     GLOBCC            ;LOG GLOBAL CRC ERROR
7257 042366 000137 042104          JMP     RXPROTO          ;GO RE-QUE BUFFER
7258
7259
7260
7261
7262 042372 105037 037072 002657 70$:  CLRB     RXTHER              ;INIT RX THRESHOLD ERROR COUNTER
7263 042376 122737 000005          CMPB    #ENQ,RHDMID        ;CONTROL MESSAGE ?
7264 042404 001402                BEQ     75$              ;YES,BRANCH
7265 042406 000137 043256          JMP     200$            ;GO PROCESS DATA MESSAGE
7266
7267
7268 042412 122737 000002 002660 75$:  CMPB    #NAK,RHDTYP        ;NAK?
7269 042420 001022                BNE     90$              ;NO,BRANCH
7270 042422 032737 000100 006576      BIT     #PRORUN,PARAM      ;PROTOCOL RUNNING ?
7271 042430 001002                BNE     80$              ;YES,BRANCH
7272 042432 000137 042104          JMP     RXPROTO          ;IGNORE THIS MESSAGE
7273 042436 052737 000400 037052 80$:  BIS     #NAKRX,PRSTAT    ;FLAG NAK RECEIVED
7274 042444 004737 043732          JSR     PC,ERRPRC        ;GO LOG NAK REASON
7275 042450 052737 000020 037052      BIS     #SDATA,PRSTAT    ;SEND DATA

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 180
DDCMP PROTOCOL MODULE

7276 042456 004737 045336
7277 042462 000137 042104

JSR PC,TXPROTO
JMP RXPROTO

;GO RE-TRANSMIT PREVIOUS MESSAGE
;GO RE-QUE RX

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 181
DDCMP PROTOCOL MODULE

```

7278
7279
7280
7281 042466 122737 000001 002660 90$: CMPB #ACK,RHDTYP ;ACK ?
7282 042474 0C1057 BNE 100$ ;NO,BRANCH
7283 042476 032737 000100 006576 BIT #PRORUN,PARAM ;PROTOCOL RUNNING ?
7284 042504 001004 BNE 93$ ;YES,BRANCH
7285 042506 052737 000100 006576 BIS #PRORUN,PARAM ;TELL THE WORLD THAT LINK HAS STARTED
7286 042514 000445 BR 97$ ;EXIT
7287 042516 123737 037062 002662 93$: CMPB T,RHDTYP ;CORRECT MESSAGE # ACKNOWLEDGED ?
7288 042524 001405 BEQ 95$ ;YES,BRANCH
7289 042526 005737 037142 TST HDXMTF ;HALF DUPLEX/MULTI -POINT ?
7290 042532 001036 BNE 97$ ;YES,BRANCH
7291 042534 000137 042104 JMP RXPROTO ;TRY AGAIN
7292 042540 105037 037073 95$: CLRB TXTHR ;INIT. TX THRESHOLD COUNTER
7293 042544 113737 037062 037060 MOVB T,N ;HIGHEST SEQUENTIAL MESSAGE # SENT
7294 042552 113737 037062 037063 MOVB T,X ;HIGHEST MESSAGE # SENT
7295 042560 113737 002662 037061 MOVB RHDREP,A ;HIGHEST MESSAGE # ACKNOWLEDGED TO THIS STATION
7296 042566 105237 037062 INCB T ;# OF NEXT DATA MESSAGE TO BE TRANSMITTED
7297 042572 012737 177777 037122 MOV #-1,TXPRC ;TRANSMIT PROTOCOL COMPLETE
7298 042600 022737 000003 006570 CMP #ACT,MODTYP ;ACTIVE MODE ?
7299 042606 001010 BNE 97$ ;NO,BRANCH
7300 042610 005737 037120 TST RXPRC ;RX PROTOCOL COMPLETE?
7301 042614 001005 BNE 97$ ;YES,BRANCH
7302 042616 005737 006574 TST FHDPLX ;HALF DUPLEX?
7303 042622 001402 BEQ 97$ ;YES,BRANCH
7304 042624 000137 042104 JMP RXPROTO ;GO PROCESS INCOMING MESSAGE
7305
7306 042630 000137 043730 97$: JMP RXPREX ;EXIT
7307
7308
7309 042634 122737 000003 002660 100$: CMPB #REP,RHDTYP ;REP ?
7310 042642 001054 BNE 150$ ;NO,BRANCH
7311
7312
7313 042644 032737 000100 006576 110$: BIT #PRORUN,PARAM ;PROTOCOL RUNNING ?
7314 042652 001002 BNE 110$ ;YES,BRANCH
7315 042654 000137 042104 JMP RXPROTO ;IGNORE MESSAGE- TRY AGAIN
7316 042660 123737 002663 037064 110$: CMPB RHDNUM,R ;HAVE WE RECEIVED THIS MESSAGE ?
7317 042666 001015 BNE 120$ ;NO, BRANCH
7318 042670 052737 000004 037052 BIS #SACK,PRSTAT ;SET SEND ACK
7319 042676 105237 037066 INCB REMTMO ;BUMP REMOTE TIME OUT COUNTER
7320 042702 001003 BNE 115$ ;BRANCH IF NOT OVERFLOW
7321 042704 112737 000377 037066 115$: MOVB #377,REMTMO ;LATCH COUNTER AT 255.
7322 042712 004737 045336 JSR PC,TXPROTO ;GO SEND ACK
7323 042716 000137 042104 JMP RXPROTO ;TRY AGAIN
7324
7325
7326 042722 052737 000010 037052 120$: BIS #SNAK,PRSTAT ;SET SEND NAK
7327 042730 112737 000003 037070 MOVB #REPRESENT,REANAK ;SET REASON FOR NAK
7328 042736 105237 037076 INCB DERIN ;BUMP DATA ERROR INBOUND
7329 042742 001003 BNE 125$ ;BRANCH IF NOT OVERFLOW
7330 042744 112737 000377 037076 125$: MOVB #377,DERIN ;LATCH AT 255.
7331 042752 152737 000004 037077 BISB #REPMASK,INMASK ;ERROR REASON IS REMOTE TIME OUT
7332 042760 004737 045336 JSR PC,ERPPRC ;PROCESS NAK
7333 042764 004737 045336 JSR PC,TXPROTO ;GO SEND NAK

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 182
DDCMP PROTOCOL MODULE

7334 042770 000137 042104

JMP RXPROTO

;TRY AGAIN

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 183
DDCMP PROTOCOL MODULE

```

7335
7336
7337
7338 042774 122737 000006 002660 150$: CMPB #STRT,RHDTYP ;START ?
7339 043002 001071 BNE 170$ ;NO,BRANCH
7340 043004 032737 000100 006576 BIT #PRORUN,PARAM ;PROTOCOL RUNNING ?
7341 043012 001007 BNE 160$ ;YES,BRANCH
7342 043014 052737 002000 037052 BIS #SSTACK,PRSTAT ;SEND START ACKNOWLEDGE
7343 043022 004737 045336 JSR PC,TXPROTO ;GO SEND STACK
7344 043026 000137 042104 JMP RXPROTO ;GO TO RX ROUTINE AND EXPECT ACK OR DATA
7345 ;: IF DMV OR DMP.
7346
7347 043032 052737 000200 006576 160$: BIS #ABORT,PARAM ;TELL MAIN CODE TO ABORT!!
7348 043040 012737 177777 037120 MOV #-1,RXPRC ;RECEIVE PROTOCOL DONE
7349 043046 012737 177777 037122 MOV #-1,TXPRC ;TRANSMIT PROTOCOL DONE
7350 043054 PRINTF #165$ ;FATAL ERROR
7351 043054 012746 043100 MOV #165$,-(SP)
7352 043060 012746 000001 MOV #1,-(SP)
7353 043064 010600 MOV SP,R0
7354 043066 104417 TRAP C$PNTF
7355 043070 062706 000004 ADD #4,SP
7356 043074 000137 043730 JMP RXPREX ;EXIT
7357
7358 043100 047045 040445 052123 .NLIST BEX
043166 .ASCIZ /%N%ASTART RECEIVED WITH PROTOCOL RUNNING--ABORTING!!/
.EVEN
.LIST BEX
7359
7360 ;: IS IT A STACK ? IF SO SEND AN 'ACK'
7361
7362 043166 122737 000007 002660 170$: CMPB #STACK,RHDTYP ;STACK ?
7363 043174 001012 BNE 180$ ;NO, BRANCH
7364 043176 052737 000004 037052 BIS #SACK,PRSTAT ;TELL TX ROUTINE TO SEND ACK
7365 043204 004737 045336 JSR PC,TXPROTO ;SEND ACK
7366 043210 052737 000100 006576 BIS #PRORUN,PARAM ;SET 'PROTOCOL RUNNING' FLAG
7367 043216 000137 043730 JMP RXPREX ;EXIT
7368
7369 ;: IF WE GOT TO HERE, WE HAVE A STRANGE PROBLEM !
7370
7371 043222 052737 000010 037052 180$: BIS #SNAK,PRSTAT ;SET SEND NAK FLAG
7372 043230 105237 037106 INCB LOSTER ;LOCAL STATION ERROR
7373 043234 152737 000021 037107 BISB #FORMERR,LSMASK ;FORMAT ERROR
7374 043242 004737 043732 JSR PC,ERRPRC ;PROCESS ERROR
7375 043246 004737 045336 JSR PC,TXPROTO ;SEND NAK
7376 043252 000137 042104 JMP RXPROTO ;TRY AGAIN
7377 ;: END OF CONTROL MESSAGE PROCESSOR
7378
7379
7380

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 184
DDCMP PROTOCOL MODULE

```

7381
7382
7383
7384
7385
7386 043256 005737 037120      200$:  TST      RXPRC      ;ALREADY PROCESSED A MESSAGE?
7387 043262 001432              BEQ      215$      ;NO,BRANCH
7388 043264 042737 001000 037052 BIC      #MYDATA,PRSTAT ;TELL RX INTERRUPT NOT TO STORE THIS
7389 043272 005737 037122      TST      TXPRC      ;TX PROTOCOL DONE ?
7390 043276 001037              BNE      220$      ;YES,BRANCH
7391      ;: SEE IF IMPLICIT ACK IMBEDDED IN THIS MESSAGE
7392 043300 123737 037062 002662 CMPB     T,RHDREP      ;RESP=MESSAGE SENT?
7393 043306 001033              BNE      220$      ;NO,BRANCH
7394 043310 113737 037062 037060 MOVB     T,N          ;HIGHEST # SENT
7395 043316 113737 037062 037063 MOVB     T,X          ;
7396 043324 113737 037062 037061 MOVB     T,A          ;HIGHEST MESSAGE ACK'ED
7397 043332 105237 037062      INCB     T          ;NEXT MESSAGE TO SEND
7398 043336 012737 177777 037122 MOV      #-1,TXPRC    ;TX PROTOCOL DONE
7399 043344 000137 043730      JMP      RXPREX    ;EXIT
7400      ;:CHECK SEQUENCE OF MESSAGE
7401 043350 105237 037064      215$:  INCB     R          ;EXPECTED #?
7402 043354 123737 037064 002663 CMPB     R,RHDNUM      ;CORRECT MESSAGE #?
7403 043362 001423              BEQ      300$      ;YES,PROCESS IT
7404 043364 105337 037064      DECB     R          ;SUBTRACT 1
7405 043370 042737 001000 037052 BIC      #MYDATA,PRSTAT ;JUST COUNT OUT MESSAGE-DON'T PUT IN BUFFER
7406 043376 032737 000100 037052 220$:  BIT      #RXD,PRSTAT ;WAIT FOR DONE
7407 043404 001003              BNE      250$      ;BRANCH
7408 043406 005737 006646      TST      TIMERS     ;TIME OUT?
7409 043412 001371              BNE      220$      ;NO,BRANCH
7410
7411      ;:SEND AN "ACK"
7412 043414 052737 000004 037052 250$:  BIS      #SACK,PRSTAT ;SEND ACK
7413 043422 004737 045336      CALL     TXPROTO  ;GO SEND IT
7414 043426 000137 042104      JMP      RXPROTO  ;TRY AGAIN
7415
7416      ;: IS DATA PART OF MESSAGE COMPLETE ?
7417 043432 032737 000100 037052 300$:  BIT      #RXD,PRSTAT ;MESSAGE COMPLETE ?
7418 043440 001021              BNE      330$      ;YES,BRANCH
7419
7420      ;: IS THE LINE DEAD ?
7421 043442 005737 006646      TST      TIMERS     ;TIMED-OUT ?
7422 043446 001004              BNE      305$      ;NO,BRANCH
7423 043450 004737 043732      JSR      PC,ERRPRC ;GO PROCESS TIMER ERROR
7424 043454 000137 042104      JMP      RXPROTO  ;TRY AGAIN
7425
7426      ;: CHECK FOR RECEIVER OVERRUN OR BUFFER PROBLEM
7427 043460 032737 000010 037052 305$:  BIT      #SNAK,PRSTAT ;DID RX INTERRUPT SET THIS ?
7428 043466 001761              BEQ      300$      ;NO,BRANCH
7429
7430      ;:RX ERROR SEND A NAK AND TRY AGAIN
7431 043470 004737 043732      JSR      PC,ERRPRC ;GO PROCESS ERROR
7432 043474 004737 045336      JSR      PC,TXPROTO ;SEND NAK
7433 043500 000137 042104      JMP      RXPROTO  ;TRY AGAIN
7434
7435      ;:CHECK FOR DATA CRC ERROR
7436 043504 032737 000001 037052 330$:  BIT      #BCCOK,PRSTAT ;DATA CRC GOOD ?

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 185
DDCMP PROTOCOL MODULE

```

7437 043512 001022          BNE      400$          :YES,BRANCH
7438
7439          :: LOG CRC ERROR AND SEND A NAK
7440 043514 052737 000010 037052      BIS      #SNAK,PRSTAT      :SET SEND NAK FLAG
7441 043522 105237 037076          INCB      DERIN          :BUMP DATA ERROR INBOUND COUNTER
7442 043526 001003          BNE      340$          :BRANCH IF NOT OVERFLOW
7443 043530 112737 000377 037076      MOVB      #377,DERIN      :LATCH AT 255.
7444 043536 152737 000002 037077 340$: BISB      #DATABCC,INMASK      :SET DATA CRC BIT
7445 043544 004737 043732          JSR      PC,ERRPRC      :GO PROCESS ERROR
7446 043550 004737 045336          JSR      PC,TXPROTO      :GO SEND NAK
7447 043554 000137 042104          JMP      RXPROTO      :TRY AGAIN
7448
7449
7450          :: WE HAVE A GOOD MESSAGE !!! SO ACKNOWLEDGE IT
7451 043560 032737 000100 006576 400$: BIT      #PRORUN,PARAM      :PROTOCOL RUNNING?
7452 043566 001007          BNE      420$          :YES,BRANCH
7453 043570 005737 037124          TST      ASTRT          :DID WE SEND A STACK?
7454 043574 001001          BNE      415$          :YES,BRANCH
7455 043576 000454          BR      RXPREX          :EXIT
7456
7457          :: NOTE: DMV/DPM WILL SEND 'START - STACK - DATA' FOR STARTUP SEQUENCE
7458 043600 052737 000100 006576 415$: BIS      #PRORUN,PARAM      :SET PROTOCOL RUNNING
7459
7460          :: CHECK FOR AN IMPLICIT 'ACK'
7461 043606 123737 037062 002662 420$: CMPB      T,RHDREP          :RESP = MESSAGE SENT ?
7462 043614 001016          BNE      450$          :NO,BRANCH
7463 043616 113737 037062 037060      MOVB      T,N          :HIGHEST SEQ MESSAGE # SENT
7464 043624 113737 037062 037063      MOVB      T,X          :HIGHEST MESSAGE SENT
7465 043632 113737 037062 037061      MOVB      T,A          :HIGHEST MESSAGE 'ACK'ED'
7466 043640 105237 037062          INCB      T          :NEXT MESSAGE # TO TRANSMIT
7467 043644 012737 177777 037122      MOV      #-1,TXPRC      :SET TRANSMIT PROTOCOL COMPLETE
7468 043652 052737 000004 037052 450$: BIS      #SACK,PRSTAT      :SET SEND ACK FLAG
7469 043660 004737 045336          JSR      PC,TXPROTO      :SEND ACK
7470 043664 012737 177777 037120      MOV      #-1,RXPRC      :RECEIVE MESSAGE PROTOCOL FINISHED
7471 043672 005737 037130          TST      PRUN          :PROTOCOL RUNNING ?
7472 043676 001414          BEQ      RXPREX          :NO,BRANCH
7473 043700 005737 037142          TST      HDXMTF          :FULL DUPLEX PT-PT?
7474 043704 001011          BNE      RXPREX          :NO,BRANCH
7475 043706 022737 000003 006570      CMP      #ACT,MODTYP      :ACTIVE MODE ?
7476 043714 001005          BNE      RXPREX          :NO,BRANCH
7477 043716 005737 037122          TST      TXPRC          :TRANSMIT PROTOCOL COMPLETE ?
7478 043722 001002          BNE      RXPREX          :YES,BRANCH
7479 043724 000137 042104          JMP      RXPROTO      :GO PROCESS MESSAGE
7480
7481 043730 000207          RXPREX: RETURN          :DONE !!
7482

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 186
DDCMP PROTOCOL MODULE

7483
7484
7485
7486
7487
7488
7489
7490
7491
7492
7493
7494
7495
7496
7497
7498
7499
7500
7501
7502
7503
7504
7505
7506
7507
7508
7509
7510
7511
7512
7513
7514
7515
7516
7517
7518
7519
7520
7521
7522
7523
7524
7525
7526
7527
7528
7529
7530
7531
7532
7533
7534
7535
7536
7537
7538

043732 005737 006646
043736 001075
043740 032737 000100 006576
043746 001034

043750 005037 006542
043754 005037 006544
043760 012737 017313 006540
043766 113737 002660 006542
043774 113737 002646 006544
044002 004737 020162
044006 005237 006506
044012 104457
044014 000012
044016 017313
044020 017636
044022 005237 006502
044026 012737 000036 006646
044034 000137 044506

044040 005237 037150
044044 022737 000024 037150
044052 001023
044054 012737 017110 006540
044062 017737 145362 006542
044070 017737 145362 006544
044076 004737 020162
044102 005237 006506
044106 104457
044110 000007

```
*****
ERROR PROCESSING ROUTINE (ERRPRC):

DESCRIPTION: THIS ROUTINE IS USED TO PROCESS INBOUND AND
              OUTBOUND ERRORS. ALSO THE 60 SECOND 'WATCHDOG'
              TIMER IS CHECKED.

              THE MAJORITY OF THE CODE IS USED IN PROCESSING
              OUTBOUND ERRORS (NAKS RECEIVED). THE NAK REASON
              TYPE IS DETERMINED AND THE APPROPRIATE ERROR
              COUNTER IS INCREMENTED. IF THE TRANSMIT THRESHOLD
              COUNTER (TXTHER) REACHES 7, IT IS CLEARED
              AND THE CUMULATIVE RECEIVE/TRANSMIT THRESHOLD
              ERROR (RXTXTE) COUNTER IS BUMPED.
*****
```

::CHECK THE WATCHDOG TIMER

```
ERRPRC: TST     TIMERS      ;60 SECONDS ELAPSED
        BNE     10$         ;NO BRANCH
        BIT     #PRORUN,PARAM ;PROTOCOL RUNNING ?
        BNE     7$         ;YES BRANCH
```

:: INFORM USER OF 'START - STACK' TIMEOUT

```
CLR     TEMP3      ;INIT IT
CLR     TEMP4      ;INIT IT
MOV     #DVEM5,TEMP2 ;'TIME OUT IN START-STACK SEQUENCE'
MOVB    RHDACC,TEMP3 ;RECEIVED DATA
MOVB    HDMCC,TEMP4  ;TRANSMITTED DATA
JSR     PC,LGDVE     ;LOG TIME OUT IN EVENT LOG
INC     ERRCNT       ;BUMP ERROR COUNT
ERRSOFT 10.,DVEM5,ERR13 ;PRINT ERROR
```

```
TRAP    CSERSOFT
.WORD   10
.WORD   DVEM5
.WORD   ERR13
```

```
INC     OPVAR       ;BUMP ERROR COUNTER
MOV     #30.,TIMERS ;RE-INIT TIMER
JMP     ERREXT      ;EXIT
```

:: INFORM USER OF 'DATA MESSAGE' TIMEOUT

```
7$: INC     TIMEOUT    ;BUMP COUNTER
    CMP     #20.,TIMEOUT ;60 SECONDS ?
    BNE     9$         ;NO BRANCH
    MOV     #DVEM2,TEMP2 ;'TIME OUT WAITING FOR RX OR TX TO COMPLETE'
    MOV     @RXCSR,TEMP3 ;RECEIVER ADDRESS
    MOV     @TXCSR,TEMP4 ;TRANSMIT ADDRESS
    JSR     PC,LGDVE     ;LOG ERROR
    INC     ERRCNT       ;BUMP ERROR COUNT
    ERRSOFT 7,DVEM2,ERR13 ;PRINT ERROR
```

```
TRAP    CSERSOFT
.WORD   7
```

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 187
DDCMP PROTOCOL MODULE

ADDRESS	DATA	INSTR	OPERAND	COMMENT	WORD	WORD	WORD
7539	044112	017110					
7540	044114	017636					
7541	044116	005037	037150	CLR	TIMEOUT		;INIT TIMEOUT
7542							
7543	044122	012737	000003	006646	9\$:	MOV	#3.,TIMERS
7544	044130	000566				BR	ERREXT
7545							;SET UP TIMER
7546							;EXIT
7547							
7548	044132	032737	000400	037052	10\$:	BIT	#NAKRX,PRSTAT
7549	044140	001542				BEQ	100\$
7550							;NAK RECEIVED?
7551							;NO,BRANCH
7552							
7553	044142	122737	000007	037073	20\$:	CMPB	#7,TXTHR
7554	044150	001403				BEQ	20\$
7555	044152	105237	037073			INCB	TXTHR
7556	044156	000404				BR	30\$
7557	044160	005237	037110			INC	RXTXTE
7558	044164	105037	037073			CLRB	TXTHR
7559							;BUMP TRANSMIT/RECEIVE THRESHOLD COUNTER
7560							;SET TRANSMIT COUNTER TO ZERO
7561							
7562							
7563							
7564							
7565							
7566							
7567							
7568							
7569							
7570							
7571							
7572							
7573							
7574							
7575							
7576							
7577							
7578							
7579							
7580							
7581							
7582							
7583							
7584							
7585							
7586							
7587							
7588							
7589							
7590							
7591							
7592							
7593							
7594							

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 188
DDCMP PROTOCOL MODULE

```

7595 044346 152737 000001 037105 47$: BISB #RXOVMSK,RMMASK ;SET MASK
7596 044354 000454 BR ERREXT ;EXIT
7597
7598
7599 044356 122737 000020 002661 50$: CMPB #MESLONG,RHDTYP+1 ;MESSAGE TOO LONG?
7600 044364 001012 BNE 55$ ;NO,BRANCH
7601 044366 105237 037102 INCB RBUFER ;LOG REMOTE STATION BUFFER ERROR
7602 044372 001003 BNE 52$ ;BRANCH IF NO OVERFLOW
7603 044374 112737 000377 037102 MOVB #377,RBUFER ;LATCH AT 255.
7604 044402 152737 000002 037103 52$: BISB #MTLMSK,RBMASK ;SET MASK
7605 044410 000436 BR ERREXT ;EXIT
7606
7607
7608 044412 122737 000021 002661 55$: CMPB #FORMERR,RHDTYP+1 ;REMOTE STATION FORMAT ERROR?
7609 044420 001012 BNE 100$ ;NO,BRANCH
7610 044422 105237 037104 INCB RMSTER ;LOG ERROR
7611 044426 001003 BNE 57$ ;BRANCH IF NO OVERFLOW
7612 044430 112737 000377 037104 MOVB #377,RMSTER ;LATCH AT 255.
7613 044436 152737 000002 037105 57$: BISB #FMTMSK,RMMASK ;SET MASK
7614 044444 000420 BR ERREXT ;EXIT
7615
7616
7617
7618
7619 044446 032737 000010 037052 100$: BIT #SNAK,PRSTAT ;SEND NAK ?
7620 044454 001414 BEQ ERREXT ;NO, BRANCH
7621
7622 044456 122737 000007 037072 CMPB #7,RXTHER ;RECEIVER THRESHOLD = 7?
7623 044464 001403 BEQ 120$ ;YES,BRANCH
7624 044466 105237 037072 INCB RXTHER ;BUMP COUNTER
7625 044472 000405 BR ERREXT ;BRANCH
7626
7627 044474 005237 037110 120$: INC RXTXTE ;BUMP CUMULATIVE COUNTER
7628 044500 105037 037072 CLRB RXTHER ;INIT RECEIVER THRESHOLD COUNTER
7629 044504 000400 BR ERREXT ;EXIT
7630
7631
7632
7633 044506 000207 ERREXT: RETURN
7634

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 189
DDCMP PROTOCOL MODULE

.SBTTL RECEIVER PROTOCOL INTERRUPT ROUTINE

++
FUNCTIONAL DESCRIPTION:
THIS ROUTINE IS USED ONLY WHEN THE '/PROTOCOL' SWITCH
IS SPECIFIED BY THE USER.

WHEN A RX INT. OCCURS THIS ROUTINE DECIDES IF IT IS A DATA SET
CHANGE OR DATA INTERRUPT. IF IT IS A DATA SET CHANGE
INTERRUPT IT PUTS THE STATUS IN 'CMODS' AND COMPARES
THAT STATUS TO THE OLD STATUS IN 'MODS'. IF THEY ARE
THE SAME THAT MEANS THE INTERRUPT WAS CAUSED BY A GLITCH
ON ONE OF THE LINES. IF THEY ARE DIFFERENT THEN A HARD
MODEM ERROR HAS OCCURED. IN ANY EVENT THE MODEM STATUS
CHANGE IS LOGGED.

IF A DATA INTERRUPT, THE ROUTINE CHECK FOR AN OVERRUN
CONDITION AND IF SET

INPUTS:
RMSGPT - ADDRESS OF RX BUFFER
RMSCC - COUNT OF DATA TO BE RXED.

SUBORDINATE ROUTINES USED:
'LOGMSC' - LOG MODEM STATUS CHANGE
'LGDVE' - LOG DEVICE ERROR

--

7665	044510				BGNSRV	PRRXI		
7666	044510						PRRXI::	
7667	044510	010246			MOV	R2,-(SP)	;SAVE R2	
7668	044512	017737	144732	011474	MOV	@RXCSR,IRXCSR	;MOV RX CSR TO IMAGE	
7669	044520	032737	000010	006576	BIT	#MOCHK,PARAM	;ANY MODEM CHANGES TO REPORT	
7670	044526	001453			BEQ	PRIN21	;IF NOT IGNORE DS CHANGE.	
7671	044530	032737	002000	037116	BIT	#INOV,IMFLAG	;IS INIT OVER	
7672	044536	001447			BEQ	PRIN21	;NO THEN IGNORE DS CHANGE.	
7673	044540	032737	004000	037116	BIT	#FIRST,IMFLAG	;FIRST TIME HERE?	
7674	044546	001443			BEQ	PRIN21	;YES,BRANCH	
7675	044550	005737	011474		TST	IRXCSR	;DATA SET CHANGE ?	
7676	044554	100040			BPL	PRIN21	;IF DATA SET CHANGE IS NOT SET BR	
7677	044556	013737	011474	011472	MOV	IRXCSR,CMODS	;MOV THE NEW MODEM STATUS IN	
7678	044564	042737	104761	011472	BIC	#104761,CMODS	;CLEAR BITS NOT RELATING TO MODEM STATUS	
7679	044572	013737	011472	006542	PRIN2: MOV	CMODS,TEMP3		
7680	044600	013737	007556	006544	MOV	MODS,TEMP4		
7681	044606	023737	006544	006542	CMP	TEMP4,TEMP3	;COMPARE OLD TO CURRENT	
7682	044614	001406			BEQ	GLINC	;INC GLITCH COUNT	
7683	044616	005237	011520		INC	MHRCNT	;INC HARD COUNT	
7684	044622	012737	016717	006540	MOV	#HRDMSG,TEMP2	;SET UP HARD MESG.	
7685	044630	000405			BR	PRIN1		
7686	044632	005237	011516		GLINC: INC	MGLCNT	;INC GLITCH COUNT	
7687	044636	012737	016671	006540	MOV	#GLMSG,TEMP2	;SET UP GLITCH	
7688	044644	004737	020336		PRIN1: JSR	PC,LOGMSC	;GO LOG MODEM STATUS CHANGE	
7689	044650	013737	011472	007556	MOV	CMODS,MODS	;MOVE CURRENT TO OLD	
7690								

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 190
RECEIVER PROTOCOL INTERRUPT ROUTINE

```

7691      ;;TEST FOR DATA
7692
7693 044656 032737 000200 011474 PRIN21: BIT    #RXDONE,IRXCSR ;RX DONE ?
7694 044664 001002          BNE    10$      ;YES,BRANCH
7695 044666 000137 045332          JMP    PRINEX ;EXIT
7696 044672 017737 144556 011476 10$:  MOV    @RXDRUF,IRXDBUF ;READ DATA
7697 044700 032737 100000 011476      BIT    #RERR,IRXDBUF ;OVERRUN ERROR ?
7698 044706 001414          BEQ    PRIN4     ;NO,BRANCH
7699
7700      ;;IF AN OVERRUN THEN LOG ERROR,SET NAK REASON,TURN OFF RX & EXIT
7701 044710 052737 000010 037052      BIS    #SNAK,PRSTAT ;TELL MAIN CODE ABOUT OVERRUN ERROR
7702 044716 105237 037106          INCB   LOSTER ;LOG LOCAL STATION ERROR
7703 044722 152737 000001 037107      BISB   #RXOVMSK,LSMASK ;SET RX OVERRUN MASK BIT
7704 044730 112737 000011 037070      MOVB   #RXOVRUN,REANAK ;SET REASON FOR SENDING NAK
7705 044736 000570          BR      PRIN8     ;GO TURN OFF RX AND EXIT
7706
7707      :::IF IN MULTI-POINT MODE AND NOT MY ADDRESS THEN JUST BUMP CHAR COUNT
7708
7709      ;;STORE AWAY DATA
7710
7711 044740 032737 001000 037052 PRIN4:  BIT    #MYDATA,PRSTAT ;STORE THIS DATA ?
7712 044746 001406          BEQ    10$      ;NO,BRANCH
7713 044750 013702 011510          MOV    RMSGPT,R2 ;SET RX MESSAGE POINTER
7714 044754 113722 011476          MOVB   IRXDBUF,(R2)+ ;STORE DATA AWAY
7715 044760 010237 011510          MOV    R2,RMSGPT ;SAVE UPDATED MESSAGE POINTER
7716
7717      ;;DECREMENT CHARACTER COUNT
7718 044764 005337 011512      10$:  DEC    RMSGCC ;ALL DATA RECEIVED ?
7719 044770 001160          BNE    PRINEX ;NO,BRANCH
7720 044772 032737 000400 0_7140      BIT    #BCC,PRFLAG ;CHECK CRC ?
7721 045000 001410          BEQ    PRIN6     ;YES,BRANCH
7722 045002 032737 010000 011476      BIT    #CRCOK,IRXDBUF ;CRC GOOD ?
7723 045010 001016          BNE    PRIN5     ;YES,BRANCH
7724 045012 052737 000002 037052      BIS    #BCCBAD,PRSTAT ;TELL MAIN CODE ABOUT CRC ERROR
7725 045020 000537          BR      PRIN8     ;DISABLE INTERRUPTS AND EXIT
7726
7727      :::IN ORDER TO CHECK CRC, WE MUST READ 2 MORE CHARACTERS(CRC)
7728 045022 052737 000400 037140 PRIN6:  BIS    #BCC,PRFLAG ;SET CRC ALREADY CHECKED FLAG
7729 045030 012737 000002 011512      MOV    #2,RMSGCC ;COUNT TWO CHARACTERS
7730 045036 012737 011514 011510      MOV    #BCCW,RMSGPT ;CRC STORAGE ADDRESS
7731 045044 000532          BR      PRINEX ;EXIT
7732
7733 045046 052737 000001 037052 PRIN5:  BIS    #BCCOK,PRSTAT ;TELL MAIN CODE CRC HAS BEEN CHECKED
7734 045054 123737 037065 002664      CMPB   TRIBN,RHDADR ;MY MESSAGE
7735 045062 001404          BEQ    5$      ;YES,BRANCH
7736 045064 042737 001000 037052      BIC    #MYDATA,PRSTAT ;DON'T STORE IT
7737 045072 000407          BR      7$      ;BRANCH
7738
7739 045074 032737 100000 002660 5$:  SELECT BIT SET ?
7740 045102 001403          BIT    #BIT15,RHDMCC ;SELECT BIT SET?
7741 045104 012737 000001 037134      BEQ    7$      ;NO,BRANCH
7742          MOV    #1,SELECT ;WE NOW HAVE THE RIGHT TO TRANSMIT,IF HALF-DUPL
7743 045112 032737 000040 037140 7$:  BIT    #RXM,PRFLAG ;READ DATA MESSAGE ?
7744 045120 001071          BNE    PRIN7     ;NO,BRANCH
7745
7746      ;;SET UP TO READ IN DATA PART OF MESSAGE

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 191
RECEIVER PROTOCOL INTERRUPT ROUTINE

```

7747 045122 042737 000003 037052      BIC      #BCCOK!BCCBAD,PRSTAT ;CLEAR FLAGS (USED IN PROTOCOL CODE)
7748 045130 052737 000040 037140      BIS      #RXM,PRFLAG      ;SET DATA MESSAGE READ FLAG
7749 045136 042737 000400 037140      BIC      #BCC,PRFLAG      ;CLEAR CRC CHECKED FLAG(USED BY THIS ROUTINE)
7750 045144 042737 140000 002660      BIC      #BIT15!BIT14,RHDMCC ;CLEAR SELECT & QS BITS
7751
7752                                     :::IS ALLOCATED BUFFER SPACE LARGE ENOUGH FOR MESSAGE?
7753 045152 023727 002660 001000      CMP      RHDMCC,#512.      ;WILL MESSAGE FIT IN ALLOCATED BUFFER?
7754 045160 003414                                     BLE      10$              ;YES,BRANCH
7755
7756                                     :::MESSAGE TOO LONG !! LOG ERROR
7757 045162 105237 037100      INCB     LBUFER      ;LOG LOCAL BUFFER ERROR
7758 045166 152737 000002 037101      BISB     #MTLMSK,LBMASK ;SET MESSAGE TOO LONG BIT
7759 045174 112737 000020 037070      MOVB     #MESLONG,REANAK ;SET REASON FOR NAK
7760 045202 152737 000010 037052      BISB     #SNAK,PRSTAT   ;SET SEND NAK FLAG
7761 045210 000443      BR          PRIN8      ;TURN OFF RX & EXIT
7762
7763                                     ::: IF A NEW BUFFER IS AVAILABLE
7764                                     :::SET BUFFER AND CHARACTER COUNT FOR MESSAGE
7765 045212 005737 037120      10$:      TST      RXPRC      ;NEW BUFFER AVAILABLE ?
7766 045216 001420      BEQ      15$              ;YES,BRANCH
7767 045220 105237 037100      INCB     LBUFER      ;LOCAL BUFFER ERROR
7768 045224 001003      BNE      12$              ;OVERFLOW?
7769 045226 012737 000377 037100      MOV      #377,LBUFER ;LATCH A 255.
7770 045234 152737 000001 037101      12$:      BISB     #BNAMSK,LBMASK ;SET MASK
7771 045242 112737 000010 037070      MOVB     #BUFFNA,REANAK ;SET NAK REASON
7772 045250 152737 000010 037052      BISB     #SNAK,PRSTAT   ;SET 'SEND NAK FLAG'
7773 045256 000412      BR          PRIN7      ;EXIT
7774
7775 045260 013737 006470 011510      15$:      MOV      DVRXA,RMSGPT ;MESSAGE BUFFER ADDRESS
7776 045266 013737 002660 011512      MOV      RHDMCC,RMSGCC ;CHARACTER COUNT OF MESSAGE
7777 045274 013737 002660 006472      MOV      RHDMCC,DVRCC  ;TELL MAIN CODE HOW LARGE MESSAGE IS
7778 045302 000413      BR          PRINEX     ;EXIT
7779
7780                                     :::MESSAGE COMPLETE
7781 045304 052737 000004 037140      PRIN7:  BIS      #QRX,PRFLAG      ;SET MESSAGE COMPLETE FLAG(USED BY MAIN CODE)
7782 045312 052737 000100 037052      BIS      #RXD,PRSTAT   ;MESSAGE COMPLETE(USED BY PROTOCOL MODULE)
7783
7784 045320 005037 037146      PRIN8:  CLR      TURNON      ;RX NOT ON
7785 045324 042777 000120 144116      BIC      #RINTEN+RXENA,QRXCSR ;TURN OFF RECEIVER
7786
7787 045332 012602      PRINEX:  MOV      (SP)+,R2      ;RESTORE R2
7788 045334      ENDSRV
7789 045334
7790 045334 000002      L10022:
7791
7792      RTI
7793

```

CZDCLR DUP-11 DATA COMM. LINK TEST
CZDCLB.F11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 192
RECEIVER PROTOCOL INTERRUPT ROUTINE

.SBTTL PROTOCOL TRANSMIT ROUTINE

FUNCTIONAL DESCRIPTION:

THIS ROUTINE IS USED TO SETUP EITHER CONTROL MESSAGES OR
DATA MESSAGES FOR TRANSMISSION.
IF THE SEND ACK(SACK) IS SET AN 'ACK' MESSAGE WILL BE SETUP
AND TRANSMITTED.
IF THE SEND NAK(SNAK) IS SET A 'NAK' MESSAGE WILL BE SETUP
AND TRANSMITTED.
ELSE A DATA MESSAGE WILL BE SETUP AND SENT.
IF THE NETWORK IS HALF-DUPLEX THEN REQUEST TO SEND(RTS) WILL
BE ASSERTED BEFORE TRANSMISSION.

7794
7795
7796
7797
7798
7799
7800
7801
7802
7803
7804
7805
7806
7807
7808
7809
7810
7811
7812
7813
7814
7815
7816
7817
7818
7819
7820
7821
7822
7823
7824
7825
7826
7827
7828
7829
7830
7831
7832
7833
7834
7835
7836
7837
7838
7839
7840
7841
7842
7843
7844
7845
7846
7847
7848
7849

045336	013737	037052	037136	TXPROT: MOV	PRSTAT,IMPRSTAT	;SAVE A COPY OF FLAGS
045344	032737	000100	006576	BIT	#PRORUN,PARAM	;PROTOCOL RUNNING ?
045352	001407			BEQ	7\$	;NO,BRANCH
045354	022737	000003	006570	CMP	#ACT,MODTYP	;ACTIVE MODE?
045362	001003			BNE	7\$	;NO,BRANCH
045364	005737	037126		5\$: TST	TXREADY	;TRANSMITTER READY FOR MESSAGE ?
045370	001775			BEQ	5\$	;NO,BRANCH
:: IF HALF DUPLEX OR MULTI-POINT LINK, WE NEED THE SELECT BIT						
:: BEFORE WE CAN SEND.						
045372	005737	037142		7\$: TST	HDXMTP	;FULL DUPLEX AND PT TO PT ?
045376	001406			BEQ	8\$	;YES,BRANCH
045400	005737	037134		6\$: TST	SELECT	;OK TO SEND ?
045404	001003			BNE	8\$	;YES,BRANCH
045406	004737	042104		CALL	RXPROTO	;GO WAIT ON SELECT BIT
045412	000772			BR	6\$	;TRY AGAIN
:: DETERMINE WHAT TO SEND						
045414	013737	037136	037052	8\$: MOV	IMPRSTAT,PRSTAT	;RESTORE ORIGINAL FLAGS
045422	113737	037065	002652	MOVB	TRIBN,HDMADR	;SET TRIB ADDRESS
045430	042737	001000	006602	BIC	#PAD,FLAG	;THIS BIT USED IN TX INTER ROUTINE
045436	005037	037126		CLR	TXREADY	;TRANSMITTER BUSY
045442	005037	037134		CLR	SELECT	;IF HALF DUPLEX/MTP MODE
045446	032737	000004	037052	BIT	#SACK,PRSTAT	;SEND ACK ?
045454	001021			BNE	10\$	;YES,BRANCH
045456	032737	000010	037052	BIT	#SNAK,PRSTAT	;SEND NAK ?
045464	001062			BNE	50\$	;YES, BRANCH
045466	032737	004000	037052	BIT	#SSTART,PRSTAT	;SEND START ?
045474	001106			BNE	60\$	;YES,BRANCH
045476	032737	002000	037052	BIT	#SSTACK,PRSTAT	;SEND START ACKNOWLEDGE ?
045504	001126			BNE	70\$	;YES,BRANCH
045506	032737	000020	037052	BIT	#SDATA,PRSTAT	;SEND DATA MESSAGE ?
045514	001151			BNE	100\$	;YES,BRANCH
045516	000000			HALT		;FATAL ERROR

:: SETUP TO SEND AN 'ACK'

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 193
PROTOCOL TRANSMIT ROUTINE

7850	045520	052737	000020	006602	10\$:	BIS	#TXM,FLAG	;SEND HEADER ONLY(USED IN TX INTER. ROUTINE)
7851	045526	112737	000005	002645		MOVB	#ENQ,HDMID	;CONTROL MESSAGE
7852	045534	012737	000001	002646		MOV	#ACK,HDMTYP	;ACK CONTROL MESSAGE
7853	045542	052737	140000	002646		BIS	#BIT15!BIT14,HDMTYP	;SET SELECT & QS FLAG
7854	045550	005737	037142			TST	HDXMTP	;HALF DUPLEX OR MULTI - POINT
7855	045554	001415				BEQ	20\$	;NO,BRANCH
7856	045556	005737	037122			TST	TXPRC	;ANY THING TO SEND ?
7857	045562	001012				BNE	20\$	;NO,BRANCH
7858	045564	032737	000100	006576		BIT	#PRORUN,PARAM	;PROTOCOL RUNNING?
7859	045572	001406				BEQ	20\$	;NO,BRANCH
7860	045574	042737	100000	002646		BIC	#BIT15,HDMTYP	;CLEAR SELECT BIT
7861	045602	012737	000001	037134		MOV	#1,SELECT	;WE HAVE SOMETHING TO SEND, SO KEEP THE LINE
7862	045610	113737	037064	002650	20\$:	MOVB	R,HDMREP	;SET RESPONSE NUMBER
7863	045616	105037	002651			CLRB	HDMNUM	;FILLER
7864	045622	042737	000004	037052		BIC	#SACK,PRSTAT	;CLEAR SEND ACK FLAG
7865	045630	000526				BR	200\$	;GO SEND IT
7866								
7867								
7868								
7869	045632	052737	000020	006602	:: SETUP TO SEND A 'NAK'	50\$:	BIS	#TXM,FLAG
7870	045640	112737	000005	002645		MOVB	#ENQ,HDMID	;TELL TX INTERRUPT TO SEND HEADER ONLY
7871	045646	012737	000002	002646		MOV	#NAK,HDMTYP	;CONTROL MESSAGE
7872	045654	113737	037070	002647		MOVB	REANAK,HDMTYP+1	;REASON FOR NAK
7873	045662	052737	140000	002646	55\$:	BIS	#BIT15!BIT14,HDMTYP	;SET SELECT & QS FLAGS
7874	045670	105037	002651			CLRB	HDMNUM	;FILLER
7875	045674	113737	037064	002650		MOVB	R,HDMREP	;LAST MESSAGE RECEIVED CORRECTLY
7876	045702	042737	000010	037052		BIC	#SNAK,PRSTAT	;CLEAR SEND NAK FLAG
7877	045710	000476				BR	200\$	;GO SEND IT
7878								
7879								
7880								
7881	045712	052737	000020	006602	:: SETUP TO SEND START MESSAGE	60\$:	BIS	#TXM,FLAG
7882	045720	112737	000005	002645		MOVB	#ENQ,HDMID	;TELL TX INT. ROUTINE TO SEND HEADER ONLY
7883	045726	012737	000006	002646		MOV	#STR,HDMTYP	;CONTROL MESSAGE
7884	045734	052737	140000	002646		BIS	#BIT15!BIT14,HDMTYP	;START MESSAGE
7885	045742	105037	002650			CLRB	HDMREP	;SET SELECT & QS FLAGS
7886	045746	105037	002651			CLRB	HDMNUM	;FILLER
7887	045752	042737	004000	037052		BIC	#SSTART,PRSTAT	;FILLER
7888	045760	000452				BR	200\$	;CLEAR SEND START FLAG
7889								;GO SEND IT
7890								
7891	045762	052737	000020	006602	:: SETUP TO SEND STACK MESSAGE	70\$:	BIS	#TXM,FLAG
7892	045770	112737	000005	002645		MOVB	#ENQ,HDMID	;TELL TX INT. TO SEND HEADER ONLY
7893	045776	012737	000007	002646		MOV	#STACK,HDMTYP	;CONTROL MESSAGE
7894	046004	052737	140000	002646		BIS	#BIT15!BIT14,HDMTYP	;START ACKNOWLEDGE MESSAGE
7895	046012	105037	002650			CLRB	HDMREP	;SET SELECT & QS FLAGS
7896	046016	105037	002651			CLRB	HDMNUM	;FILLER
7897	046022	012737	177777	037124		MOV	#-1,ASTRT	;FILLER
7898	046030	042737	002000	037052		BIC	#SSTACK,PRSTAT	;START HAS BEEN ACKNOWLEDGED
7899	046036	000423				BR	200\$	;CLEAR SEND STACK FLAG
7900								;GO SEND IT
7901								
7902								
7903	046040	042737	000020	006602	:: SETUP TO SEND DATA	100\$:	BIC	#TXM,FLAG
7904	046046	112737	000201	002645		MOVB	#SOH,HDMID	;TELL TX INTERRUPT TO SEND HEADER + DATA
7905	046054	013737	006456	002646		MOV	DVTCC,HDMCC	;DATA MESSAGE
								;CHARACTERS COUNT

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 194
PROTOCOL TRANSMIT ROUTINE

```

7906 046062 052737 140000 002646      BIS      #BIT15!BIT14,HDMCC ;SET SELECT & QS FLAGS
7907 046070 113737 037064 002650      MOV      R,HDMREP      ;LAST MESSAGE RECEIVED CORRECTLY
7908 046076 113737 037062 002651      MOV      T,HDMNUM      ;THIS MESSAGE NUMBER
7909 046104 000400                      BR        200$      ;GO SEND IT
7910
7911                                     ;; GO SET 'REQUEST TO SEND'
7912 046106 004737 036576                200$: JSR      PC,CTSSR      ;GO SET REQUEST TO SEND
7913 046112 052737 004000 037116      BIS      #FIRST,IMFLAG ;TELL THE CTSSR SUBROUTINE TO SKIP DELAY
7914
7915                                     ;; SETUP TO TRANSMIT HEADER PORTION OF MESSAGE
7916 046120 012737 002645 011500      210$: MOV      #HDMID,MSGPTR ;HEADER MESSAGE ADDRESS
7917 046126 012737 000006 011502      MOV      #6,MSGCC      ;CHARACTER COUNT OF HEADER = 6
7918 046134 012737 000020 011504      MOV      #20,SYNCC      ;NUMBER OF SYNCs TO TRANSMIT
7919
7920                                     ;; SEND THE DATA
7921 046142 005737 006574                TST      FHDPLX      ;FULL DUPLEX?
7922 046146 001004                      BNE      215$      ;YES,BRANCH
7923 046150 052777 000130 143300      BIS      #SEND!TINTEN!HDPLX,@TXCSR ;ENABLE FOR HALF DUPLEX
7924 046156 000403                      BR        217$      ;
7925
7926 046160 052777 000120 143270      215$: BIS      #SEND!#TINTEN,@TXCSR ;TURN ON TRANSMITTER
7927
7928                                     ;; IF ACTIVE MODE, TURN ON TX AND GET OUT IN A HURRY
7929                                     ;; NOTE: START UP SEQUENCE OPERATES LIKE HALF-DUPLEX
7930
7931                                     217$: TST      HDXMTPT      ;FULL DUPLEX PT-PT?
7932 046166 005737 037142                BNE      220$      ;NO,BRANCH
7933 046172 001005                      CMP      #ACT,MODTYP      ;ACTIVE MODE ?
7934 046174 022737 000003 006570      BNE      220$      ;NO,BRANCH
7935 046202 001001                      BR        TXPREX      ;EXIT
7936 046204 000406
7937
7938                                     220$: BREAK
7939 046206 104422                      TRAP      C$BRK
7940 046210 005737 037126                TST      TXREADY      ;TX FINISHED ?
7941 046214 001774                      BEQ      220$      ;NO, BRANCH
7942
7943                                     ;; IF HALF-DUPLEX OR MULTI-POINT REQUEST TO SEND WILL BE DROPPED
7944 046216 004737 037010      230$: JSR      PC,CLRRTS      ;DROP RTS IF HALF DUPLEX
7945
7946 046222 000207      TXPREX: RETURN      ;WE ARE DONE !
7947
7948                                     .EVEN
7949
7950                                     ENDTST
7951 046224
7952 046224
7953 046224 104401      L10017: TRAP      C$ETST
7954
7955
7956
7957

```

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 195
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 L10023-L\$HARD/2
L\$HARD::

.SBTTL DEVICE INDEPENDENT SECTION

GPRML DPLX,0,1,YES

.WORD T\$CODE
.WORD D?LX
.WORD 1

.SBTTL DEVICE DEPENDENT SECTION

GPRMA CSRADR,2,0,160000,177776,YES

.WORD T\$CODE
.WORD CSRADR
.WORD T\$LOLIM
.WORD T\$HILIM

GPRMA VECTOR,4,0,300,776,YES

.WORD T\$CODE
.WORD VECTOR
.WORD T\$LOLIM
.WORD T\$HILIM

GPRML RNODM,14,1,YES

.WORD T\$CODE
.WORD RNODM
.WORD 1

XFERT ENHHL

.WORD T\$CODE

GPRML PTPMLP,10,1,YES

.WORD T\$CODE
.WORD PTPMLP
.WORD 1

XFERF ENHHL

.WORD T\$CODE

GPRMD TRIBNQ,12,D,-1,1,255.,YES

.WORD T\$CODE
.WORD TRIBNQ
.WORD -1
.WORD T\$LOLIM

7958
7959
7960
7961
7962
7963
7964
7965
7966
7967
7968
7969
7970 046226
7971 046226 000030
7972 046230
7973
7974
7975
7976
7977 046230
7978 046230 000130
7979 046232 046310
7980 046234 000001
7981
7982
7983
7984
7985
7986
7987 046236
7988 046236 001031
7989 046240 046341
7990 046242 160000
7991 046244 177776
7992 046246
7993 046246 002031
7994 046250 046367
7995 046252 000300
7996 046254 000776
7997 046256
7998 046256 006130
7999 046260 046422
8000 046262 000001
8001 046264
8002 046264 012024
8003 046266
8004 046266 004130
8005 046270 046446
8006 046272 000001
8007 046274
8008 046274 006044
8009 046276
8010 046276 005052
8011 046300 046504
8012 046302 177777
8013 046304 000001

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 196
DEVICE DEPENDENT SECTION

8014 046306 000377
8015 046310
8016
8017 046310
8018
8019

ENDHWL: ENDHRD

.WORD T\$HILIM

.EVEN

L10023:

.NLIST BEX

;DEVICE INDEPENDENT QUESTIONS

046310 052506 046114 042040 DPLX: .ASCIZ /FULL DUPLEX OPERATION : /

;DEVICE DEPENDENT QUESTION

046341	104	053105	041511	CSRADR: .ASCIZ	/DEVICE CSR ADDRESS : /
046367	111	052116	051105	VECTOR: .ASCIZ	/INTERRUPT VECTOR ADDRESS: /
046422	042522	047515	042524	RNODM: .ASCIZ	/REMOTE NODE 'ITEP':/
046446	051511	052040	044510	PTPMLP: .ASCIZ	/IS THIS A MULTIPOINT NETWORK:/
046504	042101	051104	051505	TRIBNQ: .ASCIZ	/ADDRESS THIS STATION:/

.LIST BEX
.EVEN

8020
8021
8022

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 197
DEVICE DEPENDENT SECTION

8023
8024
8025
8026
8027
8028
8029
8030
8031
8032
8033
8034
8035
8036
8037
8038
8039
8040
8041
8042
8043
8044
8045
8046
8047
8048
8049
8050
8051
8052
8053
8054
8055
8056
8057
8058

046532
046532 000030

046612

046612 000000
046614 000000
046616
046616

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

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 199
CROSS REFERENCE TABLE -- USER SYMBOLS

A	=	037061	6745#	7295*	7396*	7465*
ABO	=	000026	2298#	4728		
ABORT	=	000200	2276#	4872	5616	7347
ACK	=	000001	6847#	7281	7852	
ACT	=	000003	2249#	4854	5303	5383 5636 6998 7298 7475 7815 7934
ACTATV		031312	5077	5303#		
ACTBCR		031116	5095	5265#		
ACTCHK		031526	5057	5349#		
ACTCLB		030440	5153	5167#		
ACTCLP		031640	5091	5377#		
ACTCLR		030130	5055	5108#		
ACTCOP		030736	5065	5228#		
CTCRC		031542	5086	5355#		
ACTCSE		030264	5060	5139#		
ACTCST		030356	5061	5155#		
ACTDLL		031360	5081	5317#		
ACTDME		030664	5097	5208	5211#	
ACTDMQ		030656	5098	5210#		
ACTDMS		030634	5096	5205#		
ACTDMX		030672	5212#			
ACTECH		031436	5085	5333#		
ACTEQO		031060	5069	5254#		
ACTERR		021414	3526	3516#		
ACTEXT		030214	5101	5127#		
ACTFUL		021372	3527	3560#		
ACTHLP		030150	5059	5114#		
ACTLIS		031350	5080	5314#		
ACTLLP		031650	5092	5379#		
ACTLPX		031666	5374	5376	5378	5380 5383#
ACTLXX		031730	5347	5368	5371	5384 5393#
ACTMEX		031304	5247	5263	5285	5290 5295 5298 5300#
ACTME1		031240	5274	5276	5278	5280 5282 5289#
ACTMOP		031620	5089	5373#		
ACTMOS		031550	5100	5358#		
ACTMS0		031140	5070	5273#		
ACTMS1		031146	5071	5275#		
ACTMS2		031156	5072	5277#		
ACTMS3		031166	5073	5279#		
ACTMS4		031176	5074	5281#		
ACTMS5		031206	5075	5283#		
ACTMS6		031224	5076	5286#		
ACTM2X		031406	5304	5312	5315	5318 5321 5325#
ACTNO		031426	5084	5330#		
ACTNUF		030120	5094	5105#		
ACTNUL		030126	5054	5106#		
ACTNUM		030746	5066	5231#		
ACTOPM		031040	5067	5249#		
ACTPAS		031322	5078	5306#		
ACTPRO		031556	5087	5361#		
ACTPRT		030224	5099	5130#		
ACTQFG		031562	5350	5353	5356	5359 5363#
ACTREC		031342	5079	5311#		
ACTREX		021346	3524	3551#		
ACTRHL		021302	3523	3536#		
ACTRLG		021356	3525	3555#		
ACTRLP		031660	5093	5381#		

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 200
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 201
CROSS REFERENCE TABLE -- USER SYMBOLS

BIT1 = 000002 G	2194#	2270	2303	2412	3648	6834	6835	6857										
BIT10 = 002000 G	2174#	2318	6864															
BIT11 = 004000 G	2173#	2320	2406	2418	2422	6865												
BIT12 = 010000 G	2172#	2403	2423	2430														
BIT13 = 020000 G	2171#	2401																
BIT14 = 040000 G	2170#	2405	7562	7750	7853	7873	7884	7894	7906									
BIT15 = 100000 G	2169#	2428	2429	2431	7562	7739	7750	7853	7860	7873	7884	7894	7906					
BIT2 = 000004 G	2193#	2271	2304	2404	3651	6832	6858											
BIT3 = 000010 G	2192#	2272	2305	2413	6859													
BIT4 = 000020 G	2191#	2273	2310	2414	2424	6860												
BIT5 = 000040 G	2190#	2274	2312	2415														
BIT6 = 000100 G	2189#	2275	2306	2416	2425	6861												
BIT7 = 000200 G	2188#	2276	2307	2417	2421	5429	5432											
BIT8 = 000400 G	2187#	2314	2419	2426	2432	6862												
BIT9 = 001000 G	2186#	2316	2402	2420	2427	6113	6863											
BLDBEX 023474	4030	4034#																
BLDBUF 023364	4012#	4845	4852	4987	5029	5174	5441											
BLDB1 023374	4017#																	
BLDB2 023436	4025#	4032																
BLDB3 023454	4028#	4033																
BNAMSK= 000001	6836#	7586	7770															
BOE = 000400 G	2226#																	
BUFEX 014355	3070#	4959	5000															
BUFFNA= 000010	6824#	7581	7771															
BUFLIM= 001000	2237#	2626	2627	2628	3970	3982	4101	4956	4997	5167	5437							
BYTBIT 006514	2661#	3918	5206*	5210*														
CABLE = 000002	2256#	6082																
CABLOP= 010000	2423#	6084																
CBLLUP= 000045	2374#	2974																
CCURAD 006452	2641#	4113*	4121	4125*	4841*	4849	5027	5032*	5152*									
CHECK = 000003	2340#	2905																
CLEAR = 000001	2338#	2843	4949	5108														
CLIACT 027740	4907	5046#																
CLIALN= 000007	2331#	2951																
CLIALP= 000006	2330#	2891	2901	2956														
CLIBCR 012025	3070#	5266																

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 203
CROSS REFERENCE TABLE -- USER SYMBOLS

CTSS4	036624	6652#	6656											
CTSS7	036706	6660	6663	6668#										
CURADD	006526	2670#	4017	4022	4024	4034*	4843*	4849*	4985*	4990	5027*	5032	5168*	5438*
CURCC	006520	2667#	3969	3981	3984*	3985	4018	4023	4034	4839*	4851*	5170*	5243*	5256*
		5284*	5287*	5289*	5437*									
C\$AU =	000052	1994#	4788											
C\$AUTO=	000061	1994#	4709											
C\$BRK =	000022	1994#	6056	6111	6150	6314	6610	6654	6672	7199	7939			
C\$BSEG=	000004	1994#												
C\$BSUB=	000002	1994#												
C\$CEFG=	000045	1994#												
C\$CLCK=	000062	1994#	4558	4568										
C\$CLEA=	000012	1994#	4745											
C\$CLOS=	000035	1994#												
C\$CLP1=	000006	1994#												
C\$CVEC=	000036	1994#												
C\$DCLN=	000044	1994#	4526											
C\$DODU=	000051	1994#												
C\$DRPT=	000024	1994#												
C\$DU =	000053	1994#	4766											
C\$EDIT=	000003	1994#	2058											
C\$ERDF=	000055	1994#												
C\$ERHR=	000056	1994#												
C\$ERRO=	000060	1994#												
C\$ERSF=	000054	1994#												
C\$ERSO=	000057	1994#	5749	5767	5780	5823	6067	6105	6173	6326	6444	6466	6686	7519
		7537												
C\$ESCA=	000010	1994#												
C\$ESEG=	000005	1994#												
C\$ESUB=	000003	1994#												
C\$ETST=	000001	1994#	7953											
C\$EXIT=	000032	1994#	4685	4737	4886	4946								
C\$GETB=	000026	1994#												
C\$GETW=	000027	1994#												
C\$GMAN=	000043	1994#	3452	4587	4896	5908								
C\$GPHR=	000042	1994#	4622											
C\$GPLO=	000030	1994#												
C\$GPRI=	000040	1994#												
C\$INIT=	000011	1994#	4693											
C\$INLP=	000020	1994#												
C\$MANI=	000050	1994#	4880											
C\$MEM =	000031	1994#												
C\$MSG =	000023	1994#	3086	3099	3113	3127	3147	3167						
C\$OPEN=	000034	1994#												
C\$PNTB=	000014	1994#	3082	3095	3109	3123	3143	3163						
C\$PNTF=	000017													

CZDCLB DUP-11 DATA COMM. LINK TEST
CZDCLB.P11 19-JUL-83 17:12

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 204
CROSS REFERENCE TABLE -- USER SYMBOLS

C\$RFLA= 000021	1994#										
C\$RPT = 000025	1994#	4494									
C\$SEFG= 000046	1994#										
C\$SPRI= 000041	1994#	4683	4723								
C\$VEEC= 000037	1994#	4661	4671	4678	4870	6967					
C\$TPRI= 000013	1994#										
DATABC= 000002	6822#	7444	7572	7577							
DATAWO 006556	2683#	6541*	6543*	6545							
DATCKB= 000002	2270#	4187	5349	5400	5485	5632	5681	5725	5902	5970	
DCD = 010000	2403#	2751	6662								
DCK = 000014	2291#	3352									
DCLFLG 006564	2689#	4522	4524*	4944*							
DDE = 000022	2296#	3360									
DER = 000010	2289#	3338									
DERIN 037076	6771#	7235*	7237*	7328*	7330*	7441*	7443*				
DEROUT 037074	6765#	7565*	7567*	7574*	7576*						
DEV1 007704	2806#	3743*	3754	3762*	3764*	4159	4873*	5198*			
DEV2 007706	2807#	3744*	3753	3763*	3765*	4162	4874*	5199*			
DEV3 007710	2808#	3766*	4169	4875*	5200*						
DEV4 007712	2809#	3767*	4183	4187	4191	4196	4201	4876*	5201*		
DFPTBL 002130 G	2123#										
DIAGMC= 000000	1994										
DIVN91 034772	6147#	6152									
DLE = 000020	2295#	3356									
DLL 033570	2716	5873#									
DLLCM 014130	3070#	5875									
DLLMOD= 000033	2364#	2865									
DMPE = 000053	2380#	2895									
DMPQ = 000054	2381#	2897									
DMPS = 000052	2379#	2893	4938	5207							
DMSGAD 002172	2458#	4026									
DMSGCT 002150	2445#	4025									
DOW = 000004	2250#	5317	6087								
DPLX 046310	7979	8019#									
DPV = 000000	2281#										
DSCA = 100000	2431#										
DSITEN= 000040	2415#	6248	6596	7171							
DSR = 001000	2402#	2750									
DTR = 000002	2412#	6091	6668								
DUMEX 023264	3938	3944#									
DUMPSR 023130	3909#	4940									
DUM1 023222	3919	3930#									
DUM2 023244	3929	3937#									
DUM3 023160	3918#	3942									
DUM4 023134	3910#	3941									
DUPPAR 011470	3008#	5429*	5432*	6115							
DVEMO 016747	3070#	6061	6069								
DVEM1 017026	3070#	6680	6688								
DVEM2 017110	3070#	6320	6328	7531	7539						
DVEM3 017205	3070#	6440	6446								
DVEM4 017246	3070#	6460	6468								
DVEM5 017313	3070#	6167	6175	7513	7521						
DVEM6 017377	3070#	6099	6107								
DVI = 000012	2290#	3343									
DVINEX 035130	6089	6117	6119	6121	6148	6160	6180#				
DVINIT 034306	5435	6046#	6071	6178							

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 205
CROSS REFERENCE TABLE -- USER SYMBOLS

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 206
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 207
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 20-JUL-83 13:19 PAGE 208
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 209
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 210
CROSS REFERENCE TABLE -- USER SYMBOLS

MODLOC=	000003	2222#												
MODREM=	000004	2223#												
MODS	007556	2708#	3346	6170*	6172*	6370	6379*	6625	7628	7637*				
MODTYP	006570	2656#	3309	4819*	4838	5163	5267*	5270*	5275*	5278*	5281*	5284*	5287*	5300
		5347	5406	5600	5616	5633	5912*	5972*	6051	6954	7250	7425	7762	7881
MOMSGS	007574	2724#	3825											
MOP =	000043	2224#	2337#											
MOSC =	000056	2348#	2885											
MOO	013632	2564	3035#											
MO1	013642	2565	3035#											
MO2	013653	2566	3035#											
MO3	013663	2567	3035#											
MO4	013672	2568	3035#											
MO5	013707	2569	3035#											
MO6	013714	2570	3035#											
MPPTP	037132	4614*	6665	6767#	6928	6939								
MSC =	000016	2258#	3335											
MSCMS	014764	3035#	5789											
MSGCC	011502	2984#	6108*	6260*	6267*	6503	6510*	6517*	6567*	7864*				
MSG LIM=	000017	2204#	2594	2595	2596	4069	4936	4978						
MSGPTR	011500	2983#	6107*	6259*	6265*	6506	6508*	6516*	6566*	7863*				
MSGTRN	014373	3035#	4925	4940	4966	4982								
MSGTRU	014424	3035#	3941											
MSGTYP	006516	2630#	3985	4803*	4815*	5134*	5213*	5237*	5239*	5241*	5243*	5245*	5247*	5250*
		5404*												
MSG0	002214	2411	2424	2434#										
MSG0C	002150	2411#												
MSG1	002215	2412	2425	2436#										
MSG1C	002152	2412#												
MSG2	002216	2413	2426	2438#										
MSG2C	002154	2413#												
MSG3	002217	2414	2427	2440#										
MSG3C	002156	2414#												
MSG4	002220	2415	2428	2442#										
MSG4C	002160	2415#												
MSG5	002320	2416	2429	2456#										
MSG5C	002162	2416#	4804	4816	5135	5248								
MSG6	002412	2417	2430	2470#										
MSG6C	002164	2417#	5251											
MSG8	002642	2419	2432	2496#										
MSG8C	002170	2419#												
MTLMSK=	000002	6794#	7553	7706										
MYDATA=	001000	6822#	7119	7201	7338	7355	7659	7684						
N	037060	6704#	7245*	7344*	7413*									

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 211
CROSS REFERENCE TABLE -- USER SYMBOLS

NOD104	010654	2880#
NOD105	010672	2881#
NOD106	010676	2882#
NOD107	010710	2883#
NOD11	010000	2804#
NOD110	010714	2885#
NOD111	010730	2886#
NOD112	010734	2888#
NOD113	010750	2889#
NOD114	010754	2892#
NOD115	010760	2895#
NOD116	010774	2896#
NOD117	011000	2897#
NOD12	010012	2805#
NOD120	011016	2898#
NOD121	011022	2899#
NOD122	011036	2900#
NOD123	011042	2901#
NOD124	011056	2902#
NOD125	011062	2903#
NOD126	011076	2904#
NOD127	011102	2905#
NOD13	010016	2806#
NOD130	011116	2906#
NOD131	011122	2907#
NOD132	011136	2908#
NOD133	011142	2909#
NOD134	011162	2910#
NOD135	011166	2912#
NOD136	011172	2913#
NOD137	011176	2914#
NOD14	010032	2807#
NOD140	011202	2915#
NOD141	011206	2916#
NOD142	011212	2917#
NOD143	011216	2918#
NOD144	011220	2921#
NOD145	011224	2922#
NOD146	011230	2923#
NOD147	011244	2924#
NOD15	010036	2808#
NOD150	011250	2925#
NOD151	011264	2926#
NOD152	011270	2929#
NOD153	011274	2930#
NOD154	011300	2931#
NOD155	011304	2934#
NOD156	011310	2937#
NOD157	011332	2938#
NOD16	010052	2809#
NOD160	011336	2939#
NOD161	011352	2940#
NOD162	011356	2941#
NOD163	011400	2942#
NOD164	011404	2943#
NOD165	011426	2944#

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 212
CROSS REFERENCE TABLE -- USER SYMBOLS

NOD166	011432	2947#
NOD167	011436	2948#
NOD17	010056	2810#
NOD170	011442	2949#
NOD171	011446	2954#
NOD172	021032	3460#
NOD173	021036	3461#
NOD174	021042	3462#
NOD175	021044	3463#
NOD176	021060	3464#
NOD177	021062	3465#
NOD2	007724	2797#
NOD20	010062	2811#
NOD200	021076	3466#
NOD201	021100	3467#
NOD202	021112	3468#
NOD203	021114	3469#
NOD204	021134	3470#
NOD205	021140	3471#
NOD206	021144	3472#
NOD207	021160	3473#
NOD21	010074	2812#
NOD210	021162	3474#
NOD211	021176	3475#
NOD212	021200	3476#
NOD213	021216	3477#
NOD214	021222	3478#
NOD215	021226	3479#
NOD216	021230	3480#
NOD217	021232	3481#
NOD22	010100	2813#
NOD23	010112	2814#
NOD24	010116	2815#
NOD25	010120	2819#
NOD26	010124	2820#
NOD27	010140	2821#
NOD3	007726	2798#
NOD30	010144	2822#
NOD31	010162	2823#
NOD32	010166	2824#
NOD33	010204	2825#
NOD34	010210	2826#
NOD35	010226	2827#
NOD36	010232	2828#
NOD37	010250	2829#
NOD4	007742	2799#
NOD40	010254	2830#
NOD41	010300	2831#
NOD42	010304	2832#
NOD43	010310	2833#
NOD44	010326	2834#
NOD45	010332	2835#
NOD46	010344	2836#
NOD47	010350	2840#
NOD5	007744	2800#
NOD50	010354	2841#

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 213
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 214
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 215
CROSS REFERENCE TABLE -- USER SYMBOLS

OSPOIN= 000001	1959#	1972#	2063											
OSSETU= 000000	1959#	1991	7999											
PAD = 001000	2281#	6105	6254	6496	6519	6565	7779							
PARAM 006576	2663#	3312	3351	3561	4822*	4837*	4841	5166	5299*	5299*	5309*	5327*	5330*	
	5364	5449*	5563	5577	5580	5591	5596	5645	5689	5866*	5897	5934*	5949	
	6084	6361	6914	6944	7122	7223	7235	7237*	7265	7291	7298*	7317*	7401	
	7408*	7456	7617	7760	7805									
PARCSR 011452	2962#	4600*	4601*	6079*										
PAS = 000002	2213#	5270	5300	5616	5633									
PASC = 000042	2336#	2948												
PASMOD= 000030	2326#	2824												
PCADD 006562	2653#													
PCK 014014	3035#	4151												
PCLKCT= 001600	2230#	4539												
PCLKEN= 000111	2229#	4541												
PCPM 014655	3035#													
PEC 014024	3035#	4155												
PLCK 032302	2679	5473#												
PLCK2 032302	5474#													
PLCK3 032316	5476#													
PMS 014033	3035#	4165												
PNCK 014012	3035#	4154												
PNEC 014022	3035#	4158												
PNMS 014031	3035#	4168												
PNPR 014041	3035#	4163												
PNST 014001	3035#	4150												
PNT = 001000 G	2192#													
PPR 014043	3035#	4160												
PRFLAG 037140	6770#	7118*	7165*	7663	7676*	7691	7696*	7697*	7729*					
PRI = 002000 G	2193#													
PRI88 021670	3594#	6860	6861	6862	6863	6864	6865							
PRI8S 021726	3607#	6866	6867	6868	6869	6870	6871							
PRINEX 045332	7643	7667	7679	7726	7735#									
PRIN1 044644	7633	7636#												
PRIN2 044572	7627#													
PRIN21 044656	7618	7620	7622	7624	7641#									
PRIN4 044740	7646	7659#												
PRIN5 045046	7671	7681#												
PRIN6 045022	7669	7676#												
PRIN7 045304	7692	7721	7729#											
PRIN8 045320	7653	7673	7709	7732#										
PRIW 021642	3583#	6857	6858	6859	6872									
PRI00 = 000000 G	2181#	4647												
PRI01 = 000040 G	2180#													
PRI02 = 000100 G	2179#													
PRI03 = 000140 G	2178#													
PRI04 = 000200 G	2177#													
PRI05 = 000240 G	2176#	4632	4639	4831	6919									
PRI06 = 000300 G	2175#													
PRI07 = 000340 G	2016	2174#	4687											
PRNT = 000055	2347#	2800	4898	5095										
PROEND 037114	6760#	7099												
PROFDX 041526	7005	7041#												
PROHDX 041334	6951	7003#												
PROINT 041734	6938	7095#												
PRORUN= 000100	2240#	4837	6914	6944	7223	7235	7237	7265	7291	7317	7401	7408	7456	

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 216
CROSS REFERENCE TABLE -- USER SYMBOLS

		7760	7805																
PROTEX	041674	6972	6978	6998	7024	7037	7072#												
PROTO =	000041	2335#	2880																
PROTOB=	000040	2239#	3561	4161	5325	5563	5577	5897	5949	6084									
PROTGC	040706	5579	5902*	5956*	6909#														
PRRXI	044510	6920	7614#																
PRSTAT	037052	3572	6688#	6937*	6941*	6961*	6984*	7013*	7031*	7064*	7097	7116*	7119*	7168					
		7176	7183	7185*	7201*	7202	7207	7226*	7228*	7270*	7278*	7293*	7315*	7322*					
		7338*	7355*	7356	7362*	7367	7377	7386	7390*	7418*	7497	7568	7649*	7659					
		7672*	7681*	7684*	7695*	7708*	7720*	7730*	7759	7777*	7782	7784	7786	7788					
		7790	7811*	7823*	7834*	7845*													
PRTEMP	037144	6772#																	
PRUN	037130	6766#	6935*	6946*	7421														
PSCNT	006504	2622#	4695	5387*	5811*	5814													
PST	014003	3035#	4147																
PTPMLP	046446	7952	7966#																
PTREND	006434	2597#																	
PTRTAB	006150	2594#	4072	4077	4797	4934	5120	5127	5378										
PTR13	006244	2595#	4801	4976	5104	5114	5379												
PTR23	006340	2596#	5380																
P\$ACT	003136	2582#	3428*	4249	4872*														
P\$BUFA	003132	2580#	3426*	4204	4870*														
P\$CNT	003140	2583#	4346*	4353	4415*	4422*	4427												
P\$EXIT	024364	4207	4242#	4266	4271														
P\$GDBD	003147	2587#	3412*	3431	3555*	4264*	4269*	4277	4328*	4371*	4858*	4875	5205*	5258*					
		5308*	5349*																
P\$NUUF	003146	2586#	3413*	3441	3497*	3538*	4859*	4884	5070*	5100*	5108*	5122*	5177*	5264*					
		5271*	5290*	5357*															
P\$NUM	003142	2584#	3539	3541	3543	3545	3546	4341*	4344*	4346	4347*	4349*	4353*	4355*					
		4361*	5170	5176	5197	5207	5209	5334											
P\$RADX	003144	2585#	4301*	4304*	4311*	4318	4351	4359											
P\$TREE	003134	2581#	3427*	4205	4871*														
P\$TRV	024246	3430	4203#	4874															
P\$TRS	024256	4206#	4215	4235	4240														

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 217
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 218
CROSS REFERENCE TABLE -- USER SYMBOLS

RXBUF	004150	2592#	5402											
RXC	= 000006	2253#	3299	3349										
RXCSR	011450	2961#	4597*	5823*	6017	6026	6055*	6064	6076	6170	6212*	6285	6360	6447*
		6560*	6631*	6635	6643	6672*	7124*	7126*	7481	7616	7733*			
RXD	= 000100	6820#	7202	7356	7367	7730								
RXDBUF	011454	2963#	4602*	4603*	6385	7644								
RXDONE	= 000200	2382#	6383	7641										
RXENA	= 000020	2379#	5823	6212	6447	6560	7124	7126	7733					
RXINEX	036224	6384	6397	6419	6443	6449#								
RXIN1	035702	6375	6378#											
RXIN2	035630	6369#												
RXIN21	035714	6362	6364	6366	6383#									
RXIN3	036074	6387	6422#											
RXIN4	035742	6391#												
RXIN5	036140	6401	6436#											
RXIN6	036050	6399	6416#											
RXIN7	036210	6434	6437	6445#										
RXIN8	036216	6413	6447#											
RXM	= 000040	2277#	6091	6199	6202	6436	6438	7118	7165	7691	7696			
RXMTOT	006476	2618#	5382*	5428	5501	5693								
RXON	042000	6983*	6997*	7010*	7028*	7036*	7061*	7070*	7114#	7148*				
RXONEX	042102	7115	7128#											
RXONLY	032210	2677	5426#											
RXON2	032210	5427#												
RXOVMS	= 000001	6792#	7544	7651										
RXOVRU	= 000011	6784#	7539	7652										
RXPRC	037120	6762#	6911*	6968	6989	7020	7044	7048*	7051	7252	7299*	7336	7420*	7713
RXPREX	043730	7159	7258	7307	7318	7349	7405	7422	7424	7426	7428	7431#		
RXPROT	042104	6943	6963*	6975*	6986*	7015*	7033*	7050*	7066*	7147#	7173	7180	7194	7210
		7225	7230	7243	7256	7267	7275	7286	7295	7327	7364	7374	7383	7397
		7429	7773*											
RXPTR	006440	2601#	4799*	5380*	5403	5427	5476	5502	5599	5691				
RXQ	= 000004	2252#	3295	5901										
RXTHER	037072	6722#	7215*	7571	7573*	7577*								
RXTXTE	037110	6757#	7505*	7576*										
RXWAIT	042110	7011*	7029*	7062*	7151#									
R10\$	C 1036	3460#												
R11\$	021044	3461	3462#											
R12\$	021062	3463	3464#											
R125\$	021232	3480#												
R13\$	021100	3465	3466#											
R14\$	021114	3467	3468#											
R20\$	021140	3470#												
R21\$	021162	3472	3473#											
R22\$	021200	3474	3475#											
R30\$	021230	3469	3471	3476	3477	3478	3479#							
SACK	= 000004	6817#	7031	7270	7315	7362	7418	7782	7811					
SCM	016603	3035#	3316											
SCMD	016636	3035#	3324											
SCML	016625	3035#	3320											
SDATA	= 000020	6819#	6961	6984	7013	7064	7228	7790						
SDVE	016572	3035#	3302											
SDVI	016614	3035#	3307											
SELECT	037134	6768#	6933*	7008	7026	7059	7192*	7689*	7771	7781*	7808*			
SELTHE	037071	6720#												
SEND	= 000020	2389#	6110	6273	6499	6569	7870	7873						

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 219
CROSS REFERENCE TABLE -- USER SYMBOLS

[illegible]

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 220
CROSS REFERENCE TABLE -- USER SYMBOLS

SVCINS= 000001

3035#	3289											
3035#	3284											
1959#	1978	1987	1989	1991	1993	1995	1997	1999	2001	2003	2005	2007
2009	2011	2013	2015	2017	2019	2021	2024	2027	2029	2031	2035	2035
2037	2039	2041	2043	2045	2047	2049	2051	2053	2055	2057	2059	2061
2073	2087	2088	3011	3021	3037	3054	3067	3081	3101	3120	3219	4448
4468	4484	4669	4683	4719	4741	6358	6489	7614	7919	8001#	8002	
1959#	1979	1980	1981	1982	1983	1984	1985	1986	1988	1990	1992	1994
1996	1998	2000	2002	2004	2006	2008	2010	2012	2014	2016	2018	2020
2022	2023	2025	2026	2028	2030	2032	2034	2036	2038	2040	2042	2044
2046	2048	2050	2052	2054	2056	2058	2060	2062	2072	2074	2086	3012
3014	3022	3027	3039	3040	3041	3042	3043	3044	3045	3046	3047	3048
3051	3056	3057	3058	3059	3060	3061	3064	3069	3070	3071	3072	3073
3074	3075	3078	3083	3084	3085	3086	3087	3088	3089	3092	3103	3104
3105	3106	3107	3108	3109	3112	3122	3123	3124	3125	3126	3127	3128
3129	3132	3135	3136	3245	3360	3361	3362	3363	3364	3368	3369	3370
3371	3372	3406	3407	3408	3409	3410	3417	3418	3419	3420	3421	3422
3423	3424	3434	3435	3436	3437	3438	3444	3445	3446	3447	3448	3503
3504	3505	3506	3507	3508	3550	3551	3552	3553	3554	3564	3565	3566
3567	3568	3584	3585	3586	3587	3588	3589	3595	3596	3597	3598	3599
3600	3601	3602	3603	3620	3621	3622	3623	3624	3625	3626	3627	3628
3629	3630	3631	3632	3649	3650	3651	3652	3653	3672	3673	3674	3675
3676	3682	3683	3684	3685	3686	3687	3688	3689	3690	3697	3698	3699
3700	3701	3702	3703	3711	3712	3713	3714	3715	3716	3718	3719	3720
3721	3722	3723	3724	3744	3745	3746	3747	3748	3749	3750	3751	3758
3759	3760	3761	3762	3763	3764	3765	3773	3774	3775	3776	3777	3778
3779	3780	3790	3791	3792	3793	3794	3795	3799	3800	3801	3802	3803
3819	3820	3821	3822	3823	3839	3840	3841	3842	3843	3877	3878	3879
3880	3881	3882	3887	3888	3889	3890	3891	3892	3893	3896	3897	3898
3899	3900	3901	3941	3942	3943	3944	3945	4136	4137	4138	4139	4140
4141	4142	4143	4144	4172	4173	4174	4175	4176	4177	4178	4179	4180
4181	4323	4324	4325	4326	4327	4366	4367	4368	4369	4370	4459	4491
4495	4496	4498	4500	4501	4503	4505	4506	4508	4511	4512	4514	4522
4523	4524	4526	4532	4533	4534	4536	4545	4547	4552	4553	4554	4555
4556	4557	4558	4559	4564	4565	4566	4567	4568	4586</			

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 221
CROSS REFERENCE TABLE -- USER SYMBOLS

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 222
CROSS REFERENCE TABLE -- USER SYMBOLS

TRIBN	037065	4616*	4618*	6713#	7096	7101*	7197	7682	7778										
TRIBNQ	046504	7958	7966#																
TRVACT	024366	4236	4247#	4263	4268	4273	4276	4296	4362	4385	4406	4430							
TRVALN	025160	4225	4389#																
TRVALP	025114	4224	4375#																
TRVBIF	024472	4221	4276#																
TRVBR	024462	4220	4273#																
TRVBRC	024406	4234	4254#	4274	4279	4298	4372	4387	4408	4434									
TRVDEC	024566	4227	4301#																
TRVERR	024424	4218	4263#																
TRVEXI	024444	4219	4268#																
TRVNMA	024606	4302	4305#																
TRVNOB	024416	4259#	4280	4297	4363	4386	4407												
TRVNUM	024600	4223	4304#																
TRVOCT	024600	4226	4303#																
TRVSPA	024514	4222	4282#																
TRVSTR	025246	4228	4412#																
TSOM =	000400	2391#																	
TTL =	000001	2220#	4853	6041															
TTL =	004000	2387#	6040	6043															
TTLLOP=	009044	2338#	2937																
TTOTCC	006464	2612#	4096	4795*	4920	4931	4954*	5126*											
TUPNON	037146	6773#	6934*	7114	7127*	7732*													
TXACT =	001000	2385#	6670																
TXBUF	003150	2591#	4064	4805	5129														
TXC =	000002	2251#	3290																
TXCSR	011456	2964#	4604*	4605*	6013*	6016*	6027	6040*	6043*	6048*	6065	6110*	6273*	6286					
		6499*	6569*	6644	6669	7482	7870*	7873*											
TXDBUF	011460	2965#	4606*	4607*	6493*	6498*	6509*	6512*											
TXDONE=	000200	2386#																	
TXINEX	036426	6495	6502	6511	6518	6521#													
TXIN1	036254	6492	6495#																
TXIN2	036314	6497	6503#																
TXIN3	036420	6514	6519#																
TXIN4	036356	6504	6512#																
TXM =	000020	2275#	6091	6254	6258	6513	6515	7797	7816	7828	7838	7850							
TXMTOT	006462	2611#	4084	4811*	4933*	4935	4956*	5121	5125*	5366	5451	5474	5499						
TXONLY	032242	2678	5449#																
TXON2	032250	5450#																	
TXPRC	037122	6763#	6912*	6964	6987	7016	7019*	7043*	7056	7067	7249*	7300*	7339	7348*					
		7417*	7427	7803															
TXPREX	046222	7883	7893#																
TXPROT	045336	6942	6962*	6985*	7014*	7032*	7065*	7179	7193	7229	7274	7285	7294	7316					
		7326	7363*	7382	7396	7419	7759#												
TXPTR	006442	2602#	4077*	4079*	4080	4089*	4091	4798*	4809	4934*	4948*	4949	4953*	5127*					
		5128	5378*	5450	5475	5500													
TXQ =	000000	2250#	3285																
TXREAD	037126	6500*	6765#	6910*	6995	7764	7780*	7887											
TXTHER	037073	6723#	7244*	7501	7503*	7506*													
TSARGC=	000001	1979#	1980#	1981#	1982#	1983#	1984#	3039#	3048	3056#	3061	3069#	3075	3083#					
		3089	3103#	3109	3122#	3129	3360#	3364	3368#	3372	3406#	3410	3434#	3438					
		3444#	3448	3503#	3508	3550#	3554	3564#	3568	3584#	3589	3595#	3603	3620#					
		3632	3649#	3653	3672#	3676	3682#	3690	3697#	3703	3711#	3716	3718#	3724					
		3744#	3751	3758#	3765	3773#	3780	3790#	3795	3799#	3803	3819#	3823	3839#					
		3843	3877#	3882	3887#	3893	3896#	3901	3941#	3945	4136#	4144	4172#	4181					
		4323#	4327	4366#	4370	4564#	4568	4783#	4787	4825#	4829	4878#	4882	4887#					

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 223
CROSS REFERENCE TABLE -- USER SYMBOLS

	4891	4924#	4929	4939#	4944	4965#	4970	4981#	4986	5081#	5086	5153#	5159
	5200#	5204	5230#	5234	5303#	5307	5352#	5356	5369#	5373	5839#	5843	5936#
	5940	5963#	5967	7302#	7306								
TSCODE= 005052	3420#	4555#	4864#	5875#	7925#	7935#	7940#	7945#	7949#	7951#	7955#	7957#	
TSERRN= 000007	1959#	5714#	5732#	5745#	5788#	6032#	6070#	6138#	6291#	6409#	6431#	6649#	7469#
	7487#												
TSEXCP= 000000	3420#	3425	4555#	4560	4864#	4869	5875#	5880	7935#	7939	7940#	7944	7957#
	7962												
TSFLAG= 000040	3135#	3137	4650#	4702#	4723#	4725	4745#	4747	4851#	4911#			
TSGMAN= 000000	1959#	3417#	3420	3426#	4552#	4561#	4861#	4864	4870#	5872#	5875	5881#	
TSHILI= 000377	3420#	3424	4555#	4559	4864#	4868	5875#	5879	7935#	7938	7940#	7943	7957#
	7961												
TSLAST= 000001	1959#	7999#											
TSLOLI= 000001	3420#	3423	4555#	4558	4864#	4867	5875#	5878	7935#	7937	7940#	7942	7957#
	7960												
TSLSYM= 010000	1959#	2115	3051	3064	3078	3092	3112	3132	3245	4459	4658	4674	4710
	4731	4753	6452	6524	7738	7900	7965						
TSLTNO= 000001	8002#												
TSNEST= 177777	1959#	1961#	2086#	2114#	3037#	3050#	3054#	3063#	3067#	3077#	3081#	3091#	3101#
	3111#	3120#	3131#	3219#	3244#	4448#	4458#	4468#	4475#	4484#	4657#	4669#	4673#
	4683#	4709#	4719#	4730#	4741#	4752#	4769#	6358#	6451#	6489#	6523#	7614#	7737#
	7899#	7918#	7949	7955	7963#	8003#							
TSNS0 = 000000	1961#	8003											
TSNS1 = 000004	2086#	2114	3037#	3050	3054#	3063	3067#	3077	3081#	3091	3101#	3111	3120#
	3131	3219#	3244	4448#	4458	4468#	4475	4484#	4657	4669#	4673	4683#	4709
	4719#	4730	4741#	4752	4769#	7899	7918#	7949	7955	7963			
	6358#	6451	6489#	6523	7614#	7737							
TSNS2 = 000010	1959#												
TSPTNU= 000000	1959#												
TSSAVL= 177777	1959#												
TSSEGL= 177777	1959#												
TSSUBN= 000000	1959#	4768#											
TSTAGL= 177777	1959#												
TSTAGN= 010024	1959#	2086#	3037#	3054#	3067#	3081#	3101#	3120#	3219#	4448#	4468#	4484#	4669#
	4683#	4719#	4741#	4769#	6358#	6489#	7614#	7918#					
STEMP= 000000	2074#	2075#	2114#	3050#	3063#	3077#	3091#	3111#	3131#	3135#	3136	3244#	3420#
	4458#	4475#	4555#	4650#	4651								

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 224
CROSS REFERENCE TABLE -- USER SYMBOLS

TSSINI= 010012	4484#	4650	4657										
TSSMSG= 010006	3037#	3050	3054#	3063	3067#	3077	3081#	3091	3101#	3111	3120#	3131	3135
TSSPRO= 010011	4468#												
TSSRPT= 010010	4448#	4458											
TSSSRV= 010022	3219#	3244	6358#	6451	6489#	6523	7614#	7737					
TSSSES= 010017	4769#	4851	4911	7899									
T1 026342 G	2074	4768#											
UAM = 000200 G	2190#												
UPTABL 032610	5591#												
UPTA1 032676	5597	5604#											
UPTA3 032674	5601	5603#											
UPTA4 032634	5592	5596#											
UPTEX 032746	5603	5615#											
VECTOR 046367	7941	7966#											
X 037063	6710#	7246*	7345*	7414*									
XS = 000220	1962#	2795#	2796#	2797#	2798#	2799#	2800#	2801#	2802#	2803#	2804#	2805#	2806#
	2807#	2808#	2809#	2810#	2811#	2812#	2813#	2814#	2815#	2819#	2820#	2821#	2822#
	2823#	2824#	2825#	2826#	2827#	2828#	2829#	2830#	2831#	2832#	2833#	2834#	2835#
	2836#	2840#	2841#	2842#	2843#	2844#	2849#	2850#	2851#	2852#	2853#	2856#	2857#
	2858#	2859#	2860#	2861#	2862#	2863#	2866#	2867#	2868#	2869#	2870#	2871#	2874#
	2875#	2876#	2877#	2879#	2880#	2881#	2882#	2883#	2885#	2886#	2888#	2889#	2892#
	2895#	2896#	2897#	2898#	2899#	2900#	2901#	2902#	2903#	2904#	2905#	2906#	2907#
	2908#	2909#	2910#	2912#	2913#	2914#	2915#	2916#	2917#	2918#	2921#	2922#	2923#
	2924#	2925#	2926#	2929#	2930#	2931#	2934#	2937#	2938#	2939#	2940#	2941#	2942#
	2943#	2944#	2947#	2948#	2949#	2954#	3460#	3461#	3462#	3463#	3464#	3465#	3466#
	3467#	3468#	3469#	3470#	3471#	3472#	3473#	3474#	3475#	3476#	3477#	3478#	3479#
	3480#	3481#											
XSALWA= 000000	1959#												
XSFALS= 000040	1959#	7955											
X\$OFFS= 000400	1959#	7949	7955										
X\$TRUE= 000020	1959#	7949											
\$PATCH 046532	7993#												
. = 046616	1959#	2484#	2490#	2498#	2513#	2524#	2528#	2562#	2591#	2592#	2593#	2594#	2595#
	2596#	2599#	2703#	2704#	2798#	2802#	2806#	2813#	2820#	2822#	2828#	2830#	2841#
	2843#	2850#	2852#	2868#	2870#	2874#	2876#	2888#	2895#	2897#	2899#	2901#	2903#
	2909#	2923#	2925#	2941#	3014#	3035#	3136	3463#	3465#	3469#	3474#	3476#	4651
	4703	4724	4746	4852	4912	7309#	7949	7955	7994#				

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 226
CROSS REFERENCE TABLE -- MACRO NAMES

[illegible]

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 227
CROSS REFERENCE TABLE -- MACRO NAMES

ENDMOD	1#	1959#	8002															
ENDMSG	1#	1959#	3049	3062	3076	3090	3110	3130										
ENDPRO	1#	1959#	4474															
ENDPTA	1#	1959#																
ENDRPT	1#	1959#	4457															
ENDSEG	1#	1959#																
ENDSET	1#	1959#																
ENDSFT	1#	1959#																
ENDSRV	1#	1959#	3243	6450	6522	7736												
ENDSUB	1#	1959#																
ENDSW	1#	1959#																
ENDTST	1#	1959#	7898															
EQUALS	1#	1959#	2130															
ERRDF	1#	1959#																
ERRHRD	1#	1959#																
ERROR	1#	1959#																
ERRSF	1#	1959#																
ERRSOF	1#	1959#	5712	5730	5743	5786	6030	6068	6136	6289	6407	6429	6647	7467	7485			
ERRTBL	1#	1959#																
ESCAPE	1#	1959#																
EXIT	1#	1959#	3134	4649	4701	4722	4744	4850	4910									
FEQUAL	1#	1959#																
GETBYT	1#	1959#																
GETPRI	1#	1959#																
GETWOR	1#	1959#																
GMANIA	1#	1959#																
GMANID	1#	1959#	3416	4551	4860	5871												
GMANIL	1#	1959#																
GPHARD	1#	1959#	4585															
GPRMA	1#	1959#	7934	7939														
GPRMD	1#	1959#	3417#	3420	4552#	4555	4861#	4864	5872#	5875	7956							
GPRML	1#	1959#	7924	7944	7950													
HEADER	1#	1959#	1977															
INLOOP	1#	1959#																
IOSETU	1#	1959#																
IOSTAR	1#	1959#																
KT11	1#	1959#																
LASTAD	1#	1959#	7997															
MANUAL	1#	1959#	4844															
MEMORY	1#	1959#																
MSBYTE	1#	1959#	1978#	1984	1985	1986												
MSCHEC	1#	1959#	3135#	4650#	4702#	4723#	4745#	4851#	4911#									
MSCNTO	1#	1959#	3420#	4555#	4864#	5875#	7925#	7935#	7940#	7945#	7951#	7957#						
MSCOUN	1#	1959#	3039#	3056#	3069#	3083#	3103#	3122#	3360#	3368#	3406#	3434#	3444#	3503#	3550#			
	3564#	3584#	3595#	3620#	3649#	3672#	3682#	3697#	3711#	3718#	3744#	3758#	3773#	3790#	3799#			
	3819#	3839#	3877#	3887#	3896#	3941#	4136#	4172#	4323#	4366#	4564#	4783#	4825#	4878#	4887#			
	4924#	4939#	4965#	4981#	5081#	5153#	5200#	5230#	5303#	5352#	5369#	5839#	5936#	5963#	7302#			
MSDATA	1#	1959#	1978#	1987	1989	1991	1993	1995	1997	1999	2001	2003	2005	2007	2009			
	2011	2013	2015	2017#	2019	2021	2024	2027	2029	2031	2033	2035	2037	2039	2041			
	2043	2045	2047	2049	2051	2053	2055	2057	2059	2061	3011#	3021#						
MSDECR	1#	1959#	2114#	3050#	3063#	3077#	3091#	3111#	3131#	3244#	4458#	4475#	4657#	4673#	4709#			
	4730#	4752#	6451#	6523#	7737#	7899#	7963#	8003#										
MSDEFA	1#	1959#	3420#	4555#	4864#	5875#	7925#	7935#	7940#	7945#	7951#	7957#						
MSENDE	1#	1959#	2114#	3050#	3063#	3077#	3091#	3111#	3131#	3244#	4458#	4657#	4673#	4709#	4730#			
	4752#	6451#	6523#	7737#	7899#	7963#	8003#											
MSERRI	1#	1959#	5713#	5731#	5744#	5787#	6031#	6069#	6137#	6290#	6408#	6430#	6648#	7468#	7486#			

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 228
CROSS REFERENCE TABLE -- MACRO NAMES

MSESCA	1#	1959#														
MSESCS	1#	1959#														
MSEXCP	1#	1959#	3420#	4555#	4864#	5875#	7935#	7940#	7957#							
MSEXIT	1#	1959#	3135#	4650#	4651	4702#	4703	4723#	4745#	4851#	4852	4911#	4912			
MSEXSE	1#	1959#	3135#	4650#	4702#	4723#	4745#	4851#	4911#							
MSEXTJ	1#	1959#	3135#	3136	4650#	4702#	4723#	4724	4745#	4746	4851#	4911#				
M\$GEN	1#	1959#	1978#	1987#	1989#	1991#	1993#	1995#	1997#	1999#	2001#	2003#	2005#	2007#	2009#	
	2011#	2013#	2015#	2017#	2019#	2021#	2024#	2027#	2029#	2031#	2033#	2035#	2037#	2039#	2041#	
	2043#	2045#	2047#	2049#	2051#	2053#	2055#	2057#	2059#	2061#	2073#	2087#	2088#	2114#	3011#	
	3021#	3037#	3050#	3054#	3063#	3067#	3077#	3081#	3091#	3101#	3111#	3120#	3131#	3219#	3244#	
	3425#	4448#	4458#	4468#	4484#	4560#	4657#	4669#	4673#	4683#	4709#	4719#	4730#	4741#	4752#	
	4768#	4869#	5880#	6358#	6451#	6489#	6523#	7614#	7737#	7899#	7919#	7964#	8001#			
M\$GENB	1#	1959#	3417#	3418	4552#	4553	4861#	4862	5872#	5873						
M\$GETS	1#	1959#	2114#	3050#	3063#	3077#	3091#	3111#	3131#	3244#	4458#	4475#	4657#	4673#	4709#	
	4730#	4752#	6451#	6523#	7737#	7899#	7949#	7955#	7963#	8003#						
M\$GETT	1#	1959#	3135#	4650#	4702#	4723#	4745#	4851#	4911#	7949#	7955#					
M\$GNGB	1#	1959#	1961#	1978#	1987#	1989#	1991#	1993#	1995#	1997#	1999#	2001#	2003#	2005#	2007#	
	2009#	2011#	2013#	2015#	2017#	2019#	2021#	2024#	2027#	2029#	2031#	2033#	2035#	2037#	2039#	
	2041#	2043#	2045#	2047#	2049#	2051#	2053#	2055#	2057#	2059#	2061#	2072#	2073	2086#	2087	
	2088	3011#	3021#	3037#	3054#	3067#	3081#	3101#	3120#	3219#	4448#	4468#	4484#	4669#	4683#	
	4719#	4741#	6358#	6489#	7614#	7918#	7919	7998#	8001							
M\$GNIN	1#	1959#	1978#	1979	1980	1981	1982	1983	1984#	1985#	1986#	1987#	1988	1989#	1990	
	1991#	1992	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	2049#	2050	
	2051#	2052	2053#	2054	2055#	2056	2057#	2058	2059#	2060	2061#	2062	2072#	2074#	2086#	
	3011#	3012	3014	3021#	3022	3027	3039#	3040	3041#	3042	3043#	3044#	3045#	3046	3047#	

CZDCLA DUP-11 DATA COMM. LINK TEST
CZDCLA.P11 19-MAR-82 18:19

MACY11 30A(1052) 23-MAR-82 16:47 PAGE 231
CROSS REFERENCE TABLE -- MACRO NAMES

	2925	2926	2929	2930	2931	2934	2937	2938	2939	2940	2941	2942	2943	2944	2947
	2948	2949	2954	3460	3461	3462	3463	3464	3465	3466	3467	3468	3469	3470	3471
	3472	3473	3474	3475	3476	3477	3478	3479	3480	3481					
OPEN	1#	1959#													
POINTE	1#	1959#	1971												
PRINTB	1#	1959#	3038	3055	3068	3082	3102	3121							
PRINTF	1#	1959#	3359	3367	3405	3433	3443	3502	3563	3876	3886	3895	3940	4322	4365
	4563	4782	4824	4877	4886	4923	4938	4964	4980	5080	5152	5199	5229	5302	5351
	5368	5838	5935	5962	7301										
PRINTS	1#	1959#	3549	3583	3594	3619	3648	3671	3681	3696	3710	3717	3743	3757	3772
	3789	3798	3818	3838	4135	4171									
PRINTX	1#	1959#													
READBU	1#	1959#	4544												
READEF	1#	1959#	4494	4499	4504	4510									
RFLAGS	1#	1959#													
SETPRI	1#	1959#	4646	4686											
SETVEC	1#	1959#	4621	4631	4638	4830	6918								
SLASH	1#	1959#													
STARS	1#	1959#													
SVC	1#	1959#													
XFER	1#	1959#	3135#	4650#	4702#	4723#	4745#	4851#	4911#						
XFERF	1#	1959#	7954												
XFERT	1#	1959#	7948												

. ABS. 046616 000

ERRORS DETECTED: 0

CZDCLA,CZDCLA.LST/CRF/SOL=SVC34R.MLB,CZDCLA.P11

RUN-TIME: 27 34 4 SECONDS

RUN-TIME RATIO: 91/66=1.3

CORE USED: 22K (43 PAGES)