DV11

FREE RUNNING ROM 1

MD-11-DZDVC-C

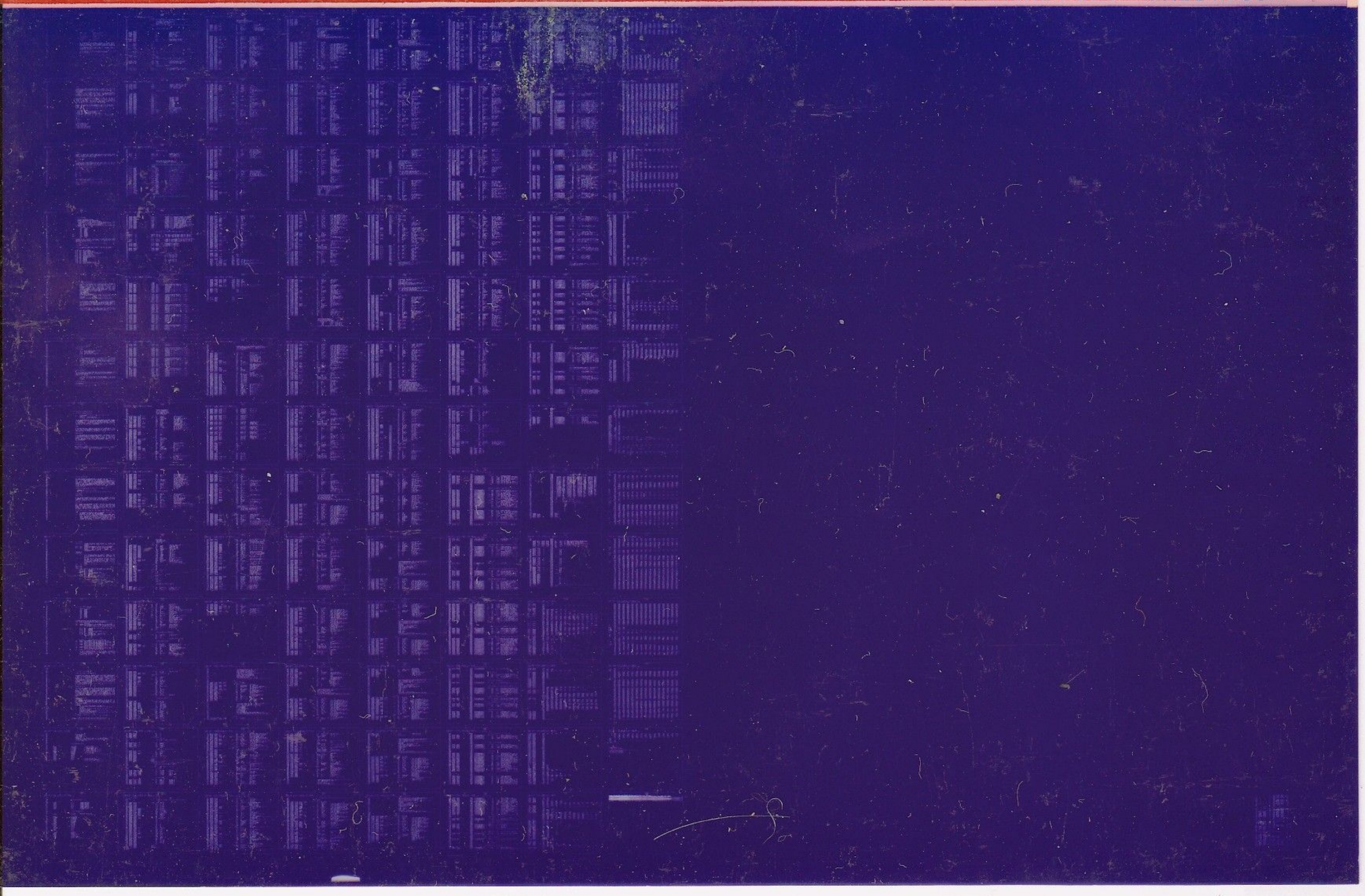
EP-DZDVC-C-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA



IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDVC-C-D

PRODUCT NAME: "FREE RUNNING" ROM TEST PART 1

DATE RELEASED: 21-APRIL-1976

MAINTAINER: DIAGNOSTICS

AUTHOR: JOHN EGOLF

1. ABSTRACT

THE FUNCTION OF THE DV11 DIAGNOSTICS ARE TO VERIFY THAT THE OPTION OPERATES ACCORDING TO SPECIFICATIONS. THE DIAGNOSTICS VERFIY THAT THERE ARE NO MALFUNCTIONS AND THE ALL OPERATIONS OF THE DV11 ARE CORRECT IN ITS ENVIROMENT.
PARAMETERS MAY BE SET TO ALERT DIAGNOSTICS AS TO THE DV11 CONFIGURATION BY USING THE "TRIAL" PROGRAM (DZDVE SA:210). ALL QUESTIONS SHOULD BE ANSWERED AND THEN EACH DIAGNOSTIC WILL "OVERLAY" THESE PARAMETERS WHICH ARE STORED IN THE "STATUS TABLE" (SEE SECTION 8.4A). THE ALTERNATIVE TO "TRIAL" PROGRAM IS "AUTO SIZING" (SEE SECTION 8.5).

DZDVC FIRST ACTLALLY READS THE DV11 ROMS DATA IN TEST 1 BY THE USE OF DVSCR02 AND A "BRANCH 'A'" TO ADDRESS 0 OF THE ROM. ALL DATA IS COMPARED TO CORE IMAGE. FROM TEST 2 ON; THE DV11 MICRO PROCESSOR IS ALLOWED TO "FREE RUN" (MAY THE DV11 HAVE MERCY ON YOU) AND DV11 FUNCTIONS ARE NOW TESTED. THINGS LIKE CURRENT ADDRESS INCREMENTING, BYTE COUNT INCREMENTING, SILO ENTRIES, ETC. ARE NOW CHECKED.

CURRENTLY THERE ARE SIX OFF LINE DIAGNOSTICS THAT ARE TO BE RUN IN SEQUENCE TO INSURE THAT IF AN ERROR SHOULD OCCUR IT WILL BE DETECTED AT AN EARLY STAGE AND INSURING THAT DIAGNOSIS OF ERROR WILL BE IMMEDIATE TO PROBLEM
NOTE: ADDITIONAL DIAGNOSTICS MAY BE ADDED IN THE FUTURE.

THE SIX DIAGNOSTICS ARE:
1. DZDVA [REV] BASIS R/W TEST AND ROM INSTRUCTION EXERCISER..
2. DZDVB [REV] STATIC LINE CARD TESTS.
3. DZDVC [REV] 'FREE RUNNING' ROM TESTS PART 1.
4. DZDVD [REV] 'FREE RUNNING' ROM TESTS PART 2.
5. DZDVE [REV] MODEM CONTROL AND CABLE TESTS PLUS MANUAL PARAMETER INPUT. [TRIAL PROGRAM]
6. DZDVF [REV] ASYNCHRONOUS LINE CARD TESTS.

2. REQUIREMENTS

2.1 EQUIPMENT

ANY PDP11 FAMILY CPU (WITH MINIMUM 8K MEMORY)
ASR 33 (OR EQUILIVALENT)
DV11-AA MUX CNTRL UNIT
AT LEAST ONE OF THE FOLLOWING
DV11-BA 8 LINE SYNC MODULES
DV11-BB 8 LINE ASYNC MODULES
DV11-BC 4 SYNC LINES, 4 ASYNC LINES

2.2 STORAGE

PROGRAM WILL USE ALL 8K OF MEMORY EXCEPT WHERE ABL AND BOOTSTRAP LOADER RESIDE. LOCATION 1500 THRU 1736 ARE ESPECIALLY TO BE NOTED AND TO BE UNTOUCHED BY OPERATOR AFTER DV11 TRIAL PROGRAM HAS BEEN EXECUTED; OR AFTER THE 'AUTO SIZING' HAS BEEN DONE.

3. LOADING PROCEEDURE

3.1 METHOD

ALL PROGRAMS ARE IN ABSOLUTE FORMAT AND ARE LOADED USING THE ABSOLUTE LOADER. NOTE: IF THE DIAGNOSTICS ARE ON A MEDIA SUCH AS DISK ,MAGTAPE,DECTAPE, OR CASSETTE; FOLLOW INSTRUCTIONS FOR THE MONITOR WHICH HAS BEEN PROVIDED ON THAT SPECIFIC MEDIA.

ABSOLUTE LOADER STARTING ADDRESS *500

MEMORY	* SIZE
4K	17
8K	37
12K	57
16K	77
20K	117
24K	137
28K	157

3.1.1 PLACE ADDRESS OF ABS LOADER INTO SWITCH REGISTER.
(ALSO PLACE 'HALT' SW UP)

3.1.2 DEPRESS 'LOAD ADDRESS' KEY ON CONSOLE AND RELEASE.

3.1.3 DEPRESS 'START KEY' ON CONSOLE AND RELEASE (PROGRAM SHOULD NOW BE LOADING INTO CPU)

4. STARTING PROCEEDURE

A. SET SWITCH REGISTER TO 000200
B. DEPRESS 'LOAD ADDRESS' KEY AND RELEASE
C. SET SWR TO ZERO FOR 'AUTO SIZING' OR LEAVE
LEAVE SWR BIT 7=1 TO USE EXISTING PARAMETERS SET UP BY DV11 TRIAL PROGRAM OR A PREVIOUSLY RUN DV11 DIAGNOSTIC THAT USED THE 'AUTO SIZING'.(SECTION 7.2 AND 8.4.8.5 MAY BE HELPFUL,
D. DEPRESS 'START KEY' AND RELEASE THE PROGRAM WILL TYPE MAINDEC NAME AND PROGRAM NAME (IF THIS WAS THE FIRST START UP OF THE PROGRAM) AND ALSO THE FOLLOWING:

```
'MAP OF DV11 STATUS'
1500    175000
1502    000300
1504    000226
1506    000062
1510    000226
1512    000062
1514    000226
1516    000062
1520    000226
1522    000062
```

THE ABOVE IS ONLY AN EXAMPLE! THIS WOULD INDICATE THE STATUS TABLE STARTING AT ADD. 1500 IN THE PROGRAM. THE STATUS TABLE MUST BE VERIFIED BY THE USER IF AUTO SIZING IS DONE. FOR INFORMATION OF STATUS TABLE SEE SECTION 8.4 FOR HELP.

THE PROGRAM WILL TYPE 'R' AND PROCEED TO RUN THE DIAGNOSTIC

4.1 CONTROL SWITCH SETTINGS

NOTE: IF THERE IS NO REAL SWR (177570): SWR MAY BE MODIFIED AT LOC:176 OR BY HITTING CONTROL "G" (^G) ON CONSOLE TERMINAL.

```
SW 15   SET:  HALT ON ERROR
SW 14   SET:  LOOP ON CURRENT TEST
SW 13   SET:  INHIBIT ERROR PRINT OUT
SW 12   SET:  INHIBIT **ALL** TYPE OUT-BELL ON ERROR.
SW 11   SET:  INHIBIT ITERATIONS.  (QUICK PASS)
SW 10   SET:  ESCAPE TO NEXT TEST
SW 09   SET:  LOOP WITH CURRENT DATA
SW 08   SET:  CATCH ERROR AND LOOP ON IT
SW 07   SET:  USE PREVIOUS STATUS TABLE.  CLR-DO AUTO SIZE.
SW 06   SET:  RESERVED
SW 05   SET:  RESERVED
SW 04   SET:  RESERVED
SW 03   SET:  RESERVED
SW 02   SET:  LOCK ON SELECTED TEST
SW 01   SET:  RESTART PROGRAM AT SELECTED TEST
SW 00   SET:  RESELECT DV11'S DESIRED ACTIVE.
```

4.1.2 SWITCH REGISTER RESTRICTIONS

SW 00 RESELECT DV11'S DESIRED ACTIVE. PLEASE NOTE THAT A MESSAGE IS TYPED OUT FOR SETTING THE SWITCH REGISTER EQUAL TO DV11'S ACTIVE. THIS MEANS IF THE SYSTEM HAS FOUR DV11S; BITS 00,01,02,03 WILL BE SET IN LOC 'DVACTV' FROM THE SWITCH REGISTER. USING THIS SWITCH(SW00) ALTERS THAT LOCATION;THEREFORE IF FOUR DV11S ARE IN THE SYSTEM ***DO NOT*** SET SWITCHS GREATER THAN SW 03 IN THE UP POSITION. THIS WOULD BE A FATAL ERROR. DO NOT SELECT MORE ACTIVE DV11S THAN HAS BEEN GIVEN INFORMATION ABOUT IN TRIAL PROGRAM.

METHOD: A: LOAD ADDRESS 200
B: START WITH SW 00=1
C: PROGRAM WILL TYPE MESSAGE
D: SET THE BINARY NUMBER OF DV11S DESIRED ACTIVE EXAMPLE: 1=1 DV11; 3=2 DV11; 7=3 DV11; 17=4 DV11 37=5 DV11 ETC. PRESS CONTINUE.
E: NUMBER (IF VALID) WILL BE IN DATA LIGHTS (EXCLUDING 11/05)
F: SET WITH ANY OTHER SWITCH SETTINGS DESIRED. PRESS CONTINUE.

SW 01 RESTART PROGRAM AT SELECTED TEST IT IS STRONGLY SUGGESTED THAT AT LEAST ONE PASS HASS BEEN MADE BEFORE TRYING TO SELECT A TEST THAT IS NOT IN THE ORDER OF SEQUENCE THE REASON BEING IS THAT THE PROGRAM HAS TO CLEAR AREAS AND SET UP PARAMETERS. ALSO WHEN A TEST IS SELECTED ALWAYS START AT THE VERY BEGINNING OF THAT TEST.

SW 09 LOOP ON CURRENT DATA: THIS SWITCH WILL ONLY WORK IF CALL 'SCOP1' IS IN THAT TEST. THE REASON BEING THAT MOST TESTS DEAL WITH BLOCKS OF DIFFERENT DATA TO BE SENT OR RECEIVED ALL AT ONCE THUS IN BLOCK DATA; ONE PATTERN CANN'T BE SINGLED OUT.

4.1.3 SWITCH REGISTER PRIORITYS

ERROR SWITCHES

1. SW 12 DELETE PRINT OUT/BELL ON ERROR.
2. SW 13 DELETE ERROR PRINTOUT.
3. SW 15 HALT ON THE ERROR.
4. SW 08 GOTO BEGINNING OF THE TEST(ON ERROR).
5. SW 10 GOTO NEXT TEST(ON ERROR).

SCOPE SWITCHES

1. SW 09 (IF ENABLED BY 'SCOP1') ON AN ERROR; IF AN '*' IS PRINTED IN FRONT OF THE TEST NO. (EX. *TEST NO. 10) SW09 IS INCORPORATED IN THAT TEST AND THEREFORE SW09 IS *USUALLY* THE BEST SWITCH FOR THE SCOPE LOOP (SW14=0, SW10=0, SW09=1, SW08=0). IF SW09 IS NOT ENABELED; AND THERE IS A *HARD* ERROR (CONSTANT); SW08 IS BEST.
   (SW14=1,0, SW10=0, SW09=0, SW08=1). FOR INTERMITTENT ERRORS; SW14=1 WILL LOOP ON TEST REGUARDLESS OF ERROR OR NOT ERROR.
   (SW14=1, SW10=0, SW09=0, SW08=1;0)
2. SW 14
3. SW 11

4.2 STARTING ADDRESS

STARTING ADDRESS IS AT 000200 THERE ARE NO OTHER STARTING ADDRESSES FOR THE DV11 DIAGNOSTICS PREVIOUSLY MENTIONED EXCEPT FOR DIDVE WHICH IS: 000200 FOR THE MODEM CONTROL AND CABLE TESTS AND 000213 FOR THE MANUAL PARAMETER INPUT PROGRAM.

NOTE: IF ADDRESS 000042 IS NON-ZERO THE PROGRAM ASSUMES IT IS UNDER ACT11 OR XXDP CONTROL AND WILL ACT ACCORDINGLY AFTER *ALL* AVAILABLE DV11'S ARE TESTED THE PROGRAM WILL RETURN TO 'XXDP' OR 'ACT-11'.

5. OPERATING PROCEDURE

WHEN PROGRAM IS INITIALLY STARTED MESSAGES AS DESCRIBED IN SECTION FOUR WILL BE PRINTED.

AND PROGRAM WILL BEGIN RUNNING THE DIAGNOSTIC

5.2 PROGRAM AND/OR OPERATOR ACTION

THE TYPICAL APPROACH SHOULD BE

1. HALT ON ERROR (VIA SW 15=1) WHEN EVER AN ERROR OCCURS.
2. CLEAR SW 15.
3. SET SW 14: (LOOP ON THIS TEST)
4. SET SW 13: (INHIBIT ERROR PRINT OUT)

THE TEST NUMBER AND PC WILL BE TYPED OUT AND POSSIBLY AN ERROR MESSAGE (THIS DEPENDS ON THE TEST) TO GIVE THE OPERATOR AN IDEA AS TO THE SOURCE OF THE PROBLEM. IF IT IS NECESSARY TO KNOW MORE INFORMATION CONCERNING THE ERROR REPORT; LOOK IN THE LISTING FOR THAT TEST NUMBER WHICH WAS TYPED OUT AND THEN NOTE THE PC OF THE ERROR REPORT THIS WAY THE EXACT FUNCTIONING OF THE TEST CAN BE INTERPEDITED.

6. ERRORS

AS DESCRIBED PREVIOUSLY THERE WILL ALWAYS BE A TEST NUMBER AND PC TYPED OUT AT THE TIME OF AN ERROR (PROVIDING SW 13=0 AND SW 12=0). IN MOST CASES ADDITIONAL INFORMATION WILL BE SUPPLIED TO THE THE ERROR MESSAGE WHICH IS TO GIVE THE OPERATOR AN INDICATION OF THE ERROR.

6.2 ERROR RECOVERY

IF FOR SOME REASON THE DV11 SHOULD 'HANG THE BUS' (GAIN CONTROL OF BUS SO THAT CONSOLE MANUAL FUNCTIONS ARE INHIBITED) AN INIT OR POWER DOWN/UP IS NECESSARY FOR OPERATOR TO REGAIN CONTROL OF CPU. IF THIS SHOULD HAPPEN; LOOK IN LOCATION 'TSTNO' (ADDRESS 1224)FOR THE NUMBER OF THE TEST THAT WAS RUNNING AT THE TIME OF THE CATASTROPHIC ERROR. IN THIS WAY THE OPERATOR WILL HAVE AN IDEA AS TO WHAT THE DV11 WAS DOING AT THE TIME OF THE ERROR.

7. RESTRICTIONS

7.1 STARTING RESTRICTIONS

SEE SECTION 4. (PLEASE)
STATUS TABLE SHOULD BE VERIFIED REGUARDLESS OF HOW PROGRAM WAS STARTED. ALSO IT IS IMPORTANT TO USE THIS LISTING ALONG WITH THE INFORMATION PRINTED ON THE TTY TO COMPLETLY ISOLATE PROBLEMS.

7.2 OPERATING RESTRICTIONS

DV11 TRIAL PROGRAM MUST BE RUN PRIOR TO THE FIRST AND ONLY THE FIRST RUNNING OF ANY DV11 DIAGNOSTIC IF "AUTO SIZING" IS NOT USED.
NOTE: IF NO PROGRAM OTHER THAN A DV11 DIAGNOSTIC WAS LOADED AFTER DV11 TRIAL OR IF CORE MEMORY HAS NOT BEEN CHANGED;OR IF THERE IS NO DV11 CONFIGURATION CHANGES; THE DV11 TRIAL PROGRAM NEED NEVER BE RUN AGAIN. HOWEVER IF ANY OF THE ABOVE HAVE BEEN VIOLATED THE DV11 TRIAL PROGRAM MUST BE RUN AGAIN BEFORE RUNNING THE DIAGNOSTICS NOTE: AN ALTERNATIVE TO THE ABOVE IS ATTEMPTING THE 'AUTO SIZING' WHEN PROGRAM IS INITIALLY STARTED WITH SW07=0.

7.3 HARDWARE CONFIGURATION RESTRICTIONS (SYNC LINE CARDS ONLY)

1. HARDWARE MUST BE SET TO FULL DUPLEX
2. PARITY OFF.
3. ALL LINES OF A PARTICULAR LINE CARD MUST BE CONFIGURED THE SAME.

8. MISCELLANEOUS

8.1 EXECUTION TIME

ALL DV11 DEVICE DIAGNOSTICS WILL GIVE AN 'END PASS' MESSAGE (PROVIDING NO ERRORS AND SW12=0) WITHIN 4 MINS. THIS IS ASSUMING SW11=1 (DELETE ITERATIONS) IS SET TO GIVE THE FASTEST POSSIBLE EXECUTION. THE ACTUAL EXECUTION TIME DEPENDS GREATLY ON THE PDP11 CPU CONFIGURATION.

8.2 PASS COMPLETE

NOTE: *EVERY* TIME THE PROGRAM IS STARTED; THE TESTS WILL RUN AS IF SW11 (DELETE ITERATIONS) WAS UP (=1). THIS IS TO 'VERIFY NO *HARD* ERRORS' AS SOON AS POSSIBLE. THEREFORE THE FIRST PASS -EACH TIME PROGRAM IS STARTED- WILL BE A 'QUICK PASS' UNTILL ALL DV11'S IN SYSTEM ARE TESTED. WHEN THE DIAGNOSTIC HAS COMPLETED A PASS THE FOLLOWING IS AN EXAMPLE OF THE PRINT OUT TO BE EXPECTED.

END PASS DZDVC-C CSR: 175000 VEC: 300 PASSES: 000001 ERRORS: 000000

NOTE: THE NUMBERS FOR CSR AND VEC ARE NOT NECESSARILY THE VALUES FOR THE DEVICE. THEY ARE ONLY FOR THIS EXAMPLE.
NOTE: DZDVE (MODEM AND CABLE TEST) END PASS MESSAGE IS A LARGE "END" TYPED OUT ON TTY. PLEASE NOTE THAT EACH CHARACTER PRINTED IS ACTUALLY AND "END PASS" INDICATION. THIS WAS USED IN PLACE OF "BELL" BECAUSE IF SW12=1 AND AN ERROR OCCURED THE BELL MAY BE MISTAKEN FOR END PASS. THE PASS EXECUTION IS SO FAST THAT THE STANDARD END PASS WAS TOO LENGHTLY. THEREFORE EACH CHAR IS AN "END PASS AND THE ENTIRE "END" IS NOT REQUIRED FOR ACCEPTANCE.

8.4 KEY LOCATIONS

RETURN (1212)	CONTAINS THE ADDRESS WHERE PROGRAM WILL RETURN WHEN ITERATION COUNT IS REACHED OR IF LOOP ON TEST IS ASSERTED.
NEXT (1214)	CONTAINS THE ADDRESS OF THE NEXT TEST TO BE PEFORMED.
TSTNO (1224)	CONTAINS THE NUMBER OF THE TEST NOW BEING PEFORMED.
RUN (1302)	THE BIT IN 'RUN' ALWAYS POINTS ONE PAST THE DV11 CURRENTLY BEING TESTED. EXAMPLE: (RUN) 1302/0000000001000000 MEANS THAT DV11 NO.05 IS THE DV11 NOW RUNNING.
DVCR00-DVCR17 DVST00-DVST17 (1500)-(1736)	THESE LOCATIONS CONTAIN THE INFORMATION NEEDED TO TEST UP TO 8 (DECIMAL) DV11S SEQUENTIALY. THEY CONTAIN THE CSR,VECTOR AND STATUS CONCERNING THE CONFIGURATION OF EACH DV11.
DVACTV (1276)	EACH BIT SET IN THIS LOCATION INDICATES THAT THE ASSOCIATED DV11 WILL BE TESTED IN TURN. EXAMPLE: (DVACTV) 1276/0000000000011111 MEANS THAT DV11 NO. 00,01,02,03,04 WILL BE TESTED. EXAMPLE: (DVACTV) 1276/0000000000010001 MEANS THAT DV11 NO. 00,04 WILL BE TESTED.
DVSCR (1356.	CONTAINS THE RECEIVER CSR OF THE CURRENT DV11 UNDER TEST.
L00.03 (1412)	
L04.07 (1414)	
L08.11 (1416)	
L12.15 (1420)	CONTAINS THE STATUS OF THE CURRENT DV11 UNDER TEST.

BIT 15 SET: LINE CARD *NOT INSTALLED (AND WONT BE TESTED)
BIT 14 SET: RESERVED
BIT 13 SET: RESERVED
BIT 12 SET: ONE SYNC, =0: TWO SYNCS.
BIT 11 SET: ASYNC LINE CARD, =0 SYNC LINE CARD
BIT 10 SET: RESERVED
BIT 09 SET: BITS PER CHAR. (USED WITH BIT8)
BIT 08 SET: BITS PER CHAR. (USED WITH BIT9)

BIT09	BIT08	BITS PER CHAR.
0	0	8
0	1	7
1	0	6
1	1	5

BIT 07-00 SYNC "A" FOR SPECIFIED LINE CARD. BITS 07-00 MUST BE ALL ZEROS FOR TESTING ASYNC LINE CARDS.

8.4A MORE ON THAT 'STATUS TABLE' (1500-1736)

```
        'MAP OF DV11 STATUS'
1500     175000
1502     000300
1504     000226
1506     000062
1510     000226
1512     000062
1514     004000
1516     000000
1520     004000
1522     000000
```

THE ABOVE INFORMATION WILL BE REPEATED FOR EACH OF UP TO 8 DV11'S IN THE SYSTEM:THESE WILL FOLLOW UNDER THIS TABLE). EXPLANATION:

1500 175000 THIS IS THE SYSTEM CONTROL REGISTER FOR THE 1ST DV11 IN THE SYSTEM.

1502 000300 THIS IS VECTOR 'A' FOR THE FIRST DV11 IN THE SYSTEM.

1504 000226 THIS REPRESENTS 'SYNC A' AND THE SOFTWARE STATUS FOR THE 1ST LINE CARD IN THE 1ST DV11. THE BITS ARE AS FOLLOWS:

BIT 15 SET: LINE CARD *NOT INSTALLED (AND WONT BE TESTED)
BIT 14 SET: RESERVED
BIT 13 SET: RESERVED
BIT 12 SET: ONE SYNC, =0: TWO SYNCS.
BIT 11 SET: ASYNC LINE CARD, =0 SYNC LINE CARD.
BIT 10 SET: RESERVED
BIT 09 SET: BITS PER CHAR. (USED WITH BIT8)
BIT 08 SET: BITS PER CHAR. (USED WITH BIT9)

BIT09	BIT08	BITS PER CHAR.
0	0	8
0	1	7
1	0	6
1	1	5

BIT 07-00 SYNC 'A' FOR SPECIFIED LINE CARD.

1506 000062 THIS REPRESENTS 'SYNC B' FOR THE 1ST LINE CARD.

1510 000226 THIS IS 'SYNC A' AND LINE STATUS FOR THE 2ND LINE CARD. (FOR BITS DEFINATION SEE EXPLANATION FOR LINE CARD 1).

1512 000062 THIS IS 'SYNC B' FOR THE SECOND LINE CARD.

1514 000226 THIS IS 'SYNC A' AND LINE STATUS FOR THE 3RD LINE CARD. (FOR BITS DEFINATION SEE EXPLANATION FOR LINE CARD 1).

1516 000062 THIS IS 'SYNC B' FOR LINE CARD NO. 3.

1520 000226 THIS IS 'SYNC A' AND LINE STATUS FOR THE 4TH LINE CARD. (FOR BITS DEFINATION SEE EXPLANATION FOR LINE CARD 1).

1522 000062 THIS IS SYNC B FOR THE 4TH LINE CARD.

THE ABOVE IS REPEATED FOR EACH DV11 IN THE SYSTEM. THE TABLE IS FILLED BY AUTO SIZING OR BY THE MANUAL PARAMETER INPUT PROGRAM AS DESCRIBED PREVIOUSLY. ALSO IF DESIRED BY USER; THE LOCATIONS MAY BE ALTERED BY HAND (TOGGLED IN) TO SUIT THE SPECIFIC CONFIGURATION.

8.5 *** METHOD OF AUTO SIZING ***

8.5.1 FINDING THE CONTROL STATUS REGISTER.

THE PROGRAM WILL START AT ADDRESS 175000 AND START 'REFERENCEING' ADDRESS. IF A NON-EX MEMORY TRAP OCCURES; THE POINTER (HOLDING 175000) IS UPDATED BY 10 AND THE ABOVE IS REPEATED UNTILL ADDRESS 175400 IS REACHED. IF A 'SLAVE SYNC RESPONSE' WAS ISSUED BY THE DV11 (OR ANY OTHER DEVICE) (NO NXM TRAP)(AND IT (SEL0) WAS=0) ; POINTER PLUS 12 (SEL12) IS TESTED TO CONTAIN 177777 (MUST BE EXACTLY 177777); IF A TRAP IS ENCOUNTERED OR IF SEL12 DOES NOT CONTAIN 177777 THE ABOVE UPDATING IS PERFORMED. IF SEL12 WAS EQUAL TO 177777 THE POINTER IS STORED AWAY AND THE ROUTINE CONTINUES AS ABOVE:
NOTE: IF THE PROGRAM DOES NOT FIND YOUR DV11; SOMETHING IS WRONG AND AUTO SIZING SHOULD NOT BE DONE.

8.5.2 FINDING THE VECTOR

THE VECTOR AREA (ADDRESS 300-776) IS FILLED WITH THE INSTRUCTION IOT AND '.+2' (NEXT ADDRESS). BIT7 AND BIT6 (RX INTERUPT AND RX INTERUPT IE) ARE SET INTO DVSCR REGISTER; A DELAY IS MADE AND IF NO INTERUPT OCCURES (BECAUSE OF A BAD DV11) THE PROGRAM ASSUMES VECTOR ADDRESS 300 AND THE PROBLEM SHOULD BE FIXED IN THE DIAGNOSTIC. ONCE THE PROBLEM IS FIXED; THE PROGRAM SHOULD BE RE-SETUP AGAIN TO GET CORRECT VECTOR. IF AN INTERUPT OCCURED; THE ADDRESS TO WHICH THE DV11 INTERUPTED TO IS PICKED UP AND REPORTED AS THE VECTOR. NOTE: IF THE VECTOR REPORTED IS NOT THE VECTOR SET UP BY YOU; THERE IS A PROBLEM AND AUTO SIZING SHOULD NOT BE DONE.

8.5.3 PARAMETER ASSUMPTIONS.

SINCE TOO MUCH HARDWARE WOULD NEED TO BE TURNED ON TO SIZE THE REST OF THE PARAMETERS; THE PROGRAM MUST ASSUME THE REMAINING VARIATIONS. THE RESULT IF NOT TO YOUR SPECIFIC CONFIGURATION MAY BE ALTERED BY HANG (TOGGLE IN) IS DESIRED. IN THIS WAY 95% OF THE PARAMETER SETUP WAS DONE BY THE PROGRAM AND 5% BY YOU.
THEREFORE:

1) ALL LINE CARDS(4) ARE ASSUMED TO BE INSTALLED.
   SET BITIS OF STATUS MAP OF ANY (APPROIATE) LINE CARDS MISSING
2) TWO SYNCS.
   SET BIT12 IF YOU HAVE A 4 LINE GROUP SET FOR 1 SYNC.
3) EIGHT BITS PER CHAR.
   ADJUST BITS 9 AND BIT 8 IN STATUS MAP FOR YOUR CORRECT CONFIG.
4) SYNCHRONOUS LINE CARDS INSTALLED
   SET BIT11 OF STATUS MAP FOR ASYNC LINE CARD AND ZERO SYNC CHARS.
5) SYNC "A=226 AND SYNC "B"=062

IN ALL ADJUSTMENTS PLEASE REFER TO SECTION 8.4A FOR GREATER DEATAIL.
$

```
;*MAINDEC-11-DZDVC-C/(377)/ROM DATA CHECK AND "FREE RUNNING" TESTS
;*COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
;*-----------------------------------------------------------------
```

```
529
530                                             ;STARTING PROCEDURE
531                                             ;LOAD PROGRAM
532                                             ;LOAD ADDRESS 000200
533                                             ;PRESS START
534                                             ;PROGRAM WILL TYPE "MAINDEC-11-DZDVC-C/<377>/ROM DATA CHECK AND "FREE RU
535                                             ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
536                                             ;AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
537                                             ;AND THEN RESUME TESTING
538
539
540                                     ;SWITCH REGISTER OPTIONS
541                                     ;----------------------
542
543             100000                  SW15=100000             ;=1,HALT ON ERROR
544             040000                  SW14=40000              ;=1,LOOP ON CURRENT TEST
545             020000                  SW13=20000              ;=1,INHIBIT ERROR TYPEOUT
546             010000                  SW12=10000              ;=1,DELETE TYPEOUT/BELL ON ERROR.
547             004000                  SW11=4000               ;=1,INHIBIT ITERATIONS
548             002000                  SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
549             001000                  SW09=1000               ;=1,LOOP WITH CURRENT DATA
550             000400                  SW08=400                ;=1,LOOP ON ERROR
551             000200                  SW07=200                ;=1, DO "AUTO SIZING" ON INITAL START UP.
552             000100                  SW06=100
553             000040                  SW05=40
554             000020                  SW04=20
555             000010                  SW03=10
556             000004                  SW02=4                  ;LOCK ON TEST SELECT
557             000002                  SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
558             000001                  SW00=1                  ;RESELECT DV11 DESIRED ACTIVE
559                                                             ;NOTE: THIS MUST NOT EXCEED ORIGINAL COUNT
```

```
                                        ;REGISTER DEFINITIONS
                                        ;--------------------

000000                                  R0=%0           ;GENERAL REGISTER
000001                                  R1=%1           ;GENERAL REGISTER
000002                                  R2=%2           ;GENERAL REGISTER
000003                                  R3=%3           ;GENERAL REGISTER
000004                                  R4=%4           ;GENERAL REGISTER
000005                                  R5=%5           ;GENERAL REGISTER
000006                                  SP=%6           ;PROCESSOR STACK POINTER
000007                                  PC=%7           ;PROGRAM COUNTER

                                        ;LOCATION EQUIVALENCIES
                                        ;----------------------

177776                                  PS=177776       ;PROCESSOR STATUS WORD
001200                                  STACK=1200      ;START OF PROCESSOR STACK

100000                                  BIT15=100000
040000                                  BIT14=40000
020000                                  BIT13=20000
010000                                  BIT12=10000
004000                                  BIT11=4000
002000                                  BIT10=2000
001000                                  BIT9=1000
000400                                  BIT8=400
000200                                  BIT7=200
000100                                  BIT6=100
000040                                  BIT5=40
000020                                  BIT4=20
000010                                  BIT3=10
000004                                  BIT2=4
000002                                  BIT1=2
000001                                  BIT0=1
                                        ;---------------------------------

010000                                  ALU=BIT12
020000                                  RAM=BIT13
030000                                  XFR=BIT13+BIT12
040000                                  NPR=BIT14
050000                                  S.C=BIT14+BIT12
060000                                  BCC=BIT14+BIT13
070000                                  BRB=BIT14+BIT13+BIT12
                                        ;---------------------------------
```

```
                                        ;;*****************************************************************
                                        ;-----------------------------------------------------------------
                                        ;TRAP CATCHER FOR ILLEGAL INTERRUPTS
                                        ;THE STANDARD "TRAP CATCHER" IS PLACED
                                        ;BETWEEN ADDRESS 0 TO ADDRESS 776.
                                        ;IT LOOKS LIKE "PC+2 HALT".
                                        ;-----------------------------------------------------------------
                                        ;;*****************************************************************

                 000000                 .=0
                                        ;STANDARD INTERRUPT VECTORS
                                        ;-------------------------

                 000024                 .=24
 000024  000000                         .PFAIL                  ;POWER FAIL HANDLER
 000026  000340                         340                     ;SERVICE AT LEVEL 7
 000030  000000                         .ERR                    ;ERROR HANDLER
 000032  000340                         340                     ;SERVICE AT LEVEL 7
 000034  003760                         .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
 000036  000340                         340                     ;SERVICE AT LEVEL 7
                 000040                 .=40
 000040  000000                         .BLKW 1                 ;SAVE FOR ACT-11 OR DDP2
 000042  000000                         .BLKW 1                 ;RETURN ADDRESS IF UNDER ACT-11 OR DDP2
 000044  000001                         .BLKW 1                 ;SAVE FOR ACT-11 OR DDP2
 000046  002560                         LOGICAL                 ;FOR USE WITH ACT-11 OR DDP2

                 000174                 .=174
 000174  000000                 LIGHT:  0
                 000176                 .=176
 000176  000000                 SSWR:   0

                 000200                 .=200
 000200  000137  001742                 JMP     .START          ;GO TO START OF PROGRAM


                 001000                 .=1000
 001000  006377  040515  047111 MTITLE: .ASCIZ  <377><12>'MAINDEC-11-DZDVC-C <377> ROM DATA CHECK AND "FREE RUN

                 001200                 .=1200
 001200                         LIGHTS:
 001200  177570                         177570
 001202  177570                 SWR:    177570
                                        ;INDIRECT POINTERS TO TELETYPE VECTORS AND REGISTERS
                                        ;--------------------------------------------------

 001204  177560                 TKCSR:  177560                  ;TELETYPE KEYBOARD CONTROL REGISTER
 001206  177562                 TKDBR:  177562                  ;TELETYPE KEYBOARD DATA BUFFER
 001210  177564                 TPCSR:  177564                  ;TELEPRINTER CONTROL REGISTER
 001212  177566                 TPDBR:  177566                  ;TELEPRINTER DATA BUFFER

                                        ;PROGRAM CONTROL PARAMETERS
                                        ;-------------------------

 001214  000000                 RETURN: 0                       ;SCOPE ADDRESS FOR LOOP ON TEST
 001216  000000                 NEXT:   0                       ;ADDRESS OF NEXT TEST TO BE EXECUTED
 001220  000000                 LOOP:   0                       ;ADDRESS FOR LOOK ON CURRENT DATA
```

```
001232  000003          ICOUNT:  3                      ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE
001234  000000          LPCNT:   0                      ;NUMBER OF ITEREATIONS COMPLETED
001236  000000          TSTNO:   0                      ;NUMBER OF TEST IN PROGRESS
001240  000000          PASCNT:  0                      ;NUMBER OF PASSES COMPLETED
001242  000000          ERRCNT:  0                      ;TOTAL NUMBER OF ERRORS
001244  000000          LSTERR:  0                      ;PC OF LAST ERROR CALL

                                ;PROGRAM VARIABLES
                                ;-----------------

001246  000000          STAT:    0                      ;DV STATUS WORD STORAGE
001250  000000          SYNCX:   0
001252  000000          CLKX:
001254  000000          MASKX:
001256  000000          TEMP1:   0                      ;TEMPORARY STORAGE
001260  000000          TEMP2:   0                      ;TEMPORARY STORAGE
001262  000000          TEMP3:   0                      ;TEMPORARY STORAGE
001264  000000          TEMP4:   0                      ;TEMPORARY STORAGE
001266  000000          TEMP5:   0                      ;TEMPORARY STORAGE
001270  000000          SAVR0:   0                      ;R0 STORAGE
001272  000000          SAVR1:   0                      ;R1 STORAGE
001274  000000          SAVR2:   0                      ;R2 STORAGE
001276  000000          SAVR3:   0                      ;R3 STORAGE
001300  000001          SAVR4:   0                      ;R4 STORAGE
001301  000001          SAVR5:   0                      ;R5 STORAGE
001302  000001          SAVSP:   0                      ;STACK POINTER STORAGE
001303  000001          SAVPC:   0                      ;PROGRAM COUNTER STORAGE
001304  000001          DVACTV:  .BLKB  1               ;DV11'S SELECTED ACTIVE.
001306                  DVNUM:   .BLKB  1               ;OCTAL NUMBER OF DV11'S.
001306  001322          SAVACT:  .BLKB  1               ;ORIGINAL ACTV.  DEVICES.
                        SAVNUM:  .BLKB  1               ;WORKABLE NUMBER.
                        RUN:     .BLKB  1               ;POINTER ONE PAST RUNNING DEVICE.
                        .EVEN
001306  001322          STREAM:  DV.MAP                 ;TABLE POINTER.
```

```
                                        ;PROGRAM CONTROL FLAGS
                                        ;---------------------

001310  000          INIFLG: .BYTE   0              ;PROGRAM INITIALIZATION FLAG
001311  000          ERRFLG: .BYTE   0              ;ERROR OCCURED FLAG
001312  000          LOKFLG: .BYTE   0              ;LOCK ON CURRENT TEST FLAG
001313  000          QV.FLG: .BYTE   0              ;QUICK VERIFY FLAG.
                                                    ;ON FIRST PASS OF EACH D/11 ITERATIONS WILL BE S
                     .EVEN
000000               $Y=0

                        ;DEFINITIONS FOR TRAP SUBROUTINE CALLS
                        ;POINTERS TO SUBROUTINES CAN BE FOUND
                        ;IN THE TABLE IMMEDIATLY FOLLOWING THE DEFINITIONS

                     ;;*****************************************************************
                     ;-----------------------------------------------------------------
001314               .TRPTAB:
104400               SCOPE=TRAP+0               ;CALL TO SCOPE LOOP AND ITERATION HANDLER
001314  002634               .SCOPE
104401               SCOP1=TRAP+1               ;CALL TO LOOP ON CURRENT DATA HANDLER
001316  003020               .SCOP1
104402               TYPE=TRAP+2                ;CALL TO TELETYPE OUTPUT ROUTINE
001320  003044               .TYPE
104403               INSTR=TRAP+3               ;CALL TO ASCII STRING INPUT ROUTINE
001322  003120               .INSTR
104404               INSTER=TRAP+4              ;CALL TO INPUT ERROR HANDLER
001324  003224               .INSTER
104405               PARAM=TRAP+5               ;CALL TO NUMERICAL DATA INPUT ROUTINE
001326  003244               .PARAM
104406               SAVOS=TRAP+6               ;CALL TO REGISTER SAVE ROUTINE
001330  003444               .SAVOS
104407               RESOS=TRAP+7               ;CALL TO REGISTER RESTORE ROUTINE
001332  003504               .RESOS
104410               CONVRT=TRAP+10             ;CALL TO DATA OUTPUT ROUTINE
001334  003536               .CONVRT
104411               CNVRT=TRAP+11              ;CALL TO DATA OUTPUT ROUNTINE WITHOUT CR/LF.
001336  003542               .CNVRT
104412               MSTCLR=TRAP+12             ;CALL TO ISUE A MASTER CLEAR
001340  004556               .MSTCLR
104413               RAMCLR=TRAP+13             ;CALL TO CLEAR THE RAMS
001342  004516               .RAMCLR
104414               DELAY=TRAP+14              ;CALL TO VARIABLE DELAY COUNTER
001344  004476               .DELAY
104415               ROMCLK=TRAP+15             ;CALL TO CLOCK ROM ONCE
001346  004566               .ROMCLK
104416               DATACLK=TRAP+16            ;CALL TO CLK DATA
001350  004576               .DATACLK

                     ;-----------------------------------------------------------------
                     ;;*****************************************************************
```

```
                                                ;DV11 VECTOR AND REGISTER INDIRECT POINTERS

001352  000000    DVRVEC: 0               ;POINTER TO DV11 RECEIVER INTERRUPT VECTOR
001354  000000    DVRLVL: 0               ;POINTER TO DV11 RECEIVER INTERRUPT SERVICE PS
001356  000000    DVTVEC: 0               ;POINTER TO DV11 TRANSMITTER INTERRUPT VECTOR
001360  000000    DVTLVL: 0               ;POINTER TO DV11 TRANSMITTER INTERRUPT SERVICE PS
001362  000000    DVSCR:  0               ;POINTER TO DV11 SYSTEM CONTROL REGISTER
001364  000000    DVSCRH: 0               ;POINTER TO DV11 SYSTEM CONTROL REGISTER HIGH BYTE.
001366  000000    DVRIC:  0               ;POINTER TO DV11 NEXT RECEIVED CHARACTER REGISTER
001370  000000    DVLCR:  0               ;POINTER TO DV11 LINE PRAMETER REGISTER
001372  000000    DVSRS:  0               ;POINTER TO DV11 SECONDARY REGISTER SELECT REGISTER
001374  000000    DVSRSH: 0               ;POINTER TO DV11 SECONDARY REGISTER SELECT HIGH BYTE.
001376  000000    DVSRA:  0               ;POINTER TO DV11 SECONDARY REGISTER ACCESS REGISTER
001400  000000    DVSFR:  0               ;POINTER TO DV11 SPECIAL FUNCTIONS REGISTER
001402  000000    DVNSR:  0               ;POINTER TO DV11 NPR STATUS REGISTER
001404  000000    RESV16: 0               ;POINTER TO RESERVED REGISTER.


                                                ;DV11 CONTROL INDICATORS FOR CURRENT DV11 UNDER TEST
                                                ;--------------------------- --------------------

001406      000     MASK.A: .BYTE 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 00-03
001407      000     MASK.B: .BYTE 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 04-07
001410      000     MASK.C: .BYTE 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 08-11
001411      000     MASK.D: .BYTE 000       ;LAST CHAR TO TEST AND PARITY MASK FOR LINES 12-15

001412      010     CLK.A:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 00-03
001413      010     CLK.B:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 04-07
001414      010     CLK.C:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 08-11
001415      010     CLK.D:  .BYTE 8.        ;NUMBER OF CLOCKS NEEDED FOR ONE CHAR FOR LINES 12-15

001416  000000    L00.03: 000000          ;PARAMETERS FOR LINES 00-03
001420  000000    L04.07: 000000          ;PARAMETERS FOR LINES 04-07
001422  000000    L08.11: 000000          ;PARAMETERS FOR LINES 08-11
001424  000000    L12.15: 000000          ;PARAMETERS FOR LINES 12-15

001426  000000    SYNC2A: 000000          ;SYNC 2
001430  000000    SYNC2B: 000000          ;
001432  000000    SYNC2C: 000000          ;
001434  000000    SYNC2D: 000000          ;


                                    ;SUMMARY
                                    ;-------
;       MASK.X              040     5 BITS PER CHAR.
;                           100     6 BITS PER CHAR.
;                           200     7 BITS PER CHAR.
;                           000     8 BITS PER CHAR.

;       CLK.X               005     5 BITS PER CHAR.
;                           006     6 BITS PER CHAR.
;                           007     7 BITS PER CHAR.
;                           010     8 BITS PER CHAR.
```

F02


```
                                                ;DV11 STATUS TABLE AND ADDRESS ASSIGNMENTS
                                                ;-----------------------------------------

                    001500                      .=1500
001500              DV.MAP:
001500  000001      DVCR00: .BLKW 1             ;CONTROL STATUS REGISTER FOR DV11 NUMBER 00
001502  000001      DVTR00: .BLKW 1             ;VECTOR "A" FOR DV11 NUMBER 00
001504  000001      DV00.A: .BLKW 1             ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 00
001506  000001      SYNA00: .BLKW 1             ;SYNC TWO
001510  000001      DV00.B: .BLKW 1             ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 00
001512  000001      SYNB00: .BLKW 1             ;SYNC TWO
001514  000001      DV00.C: .BLKW 1             ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 00
001516  000001      SYNC00: .BLKW 1             ;SYNC TWO
001520  000001      DV00.D: .BLKW 1             ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 00
001522  000001      SYND00: .BLKW 1             ;SYNC TWO

001524  000001      DVCR01: .BLKW 1             ;CONTROL STATUS REGISTER FOR DV11 NUMBER 01
001526  000001      DVTR01: .BLKW 1             ;VECTOR "A" FOR DV11 NUMBER 01
001530  000001      DV01.A: .BLKW 1             ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 01
001532  000001      SYNA01: .BLKW 1             ;SYNC TWO
001534  000001      DV01.B: .BLKW 1             ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 01
001536  000001      SYNB01: .BLKW 1             ;SYNC TWO
001540  000001      DV01.C: .BLKW 1             ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 01
001542  000001      SYNC01: .BLKW 1             ;SYNC TWO
001544  000001      DV01.D: .BLKW 1             ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 01
001546  000001      SYND01: .BLKW 1             ;SYNC TWO

001550  000001      DVCR02: .BLKW 1             ;CONTROL STATUS REGISTER FOR DV11 NUMBER 02
001552  000001      DVTR02: .BLKW 1             ;VECTOR "A" FOR DV11 NUMBER 02
001554  000001      DV02.A: .BLKW 1             ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 02
001556  000001      SYNA02: .BLKW 1             ;SYNC TWO
001560  000001      DV02.B: .BLKW 1             ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 02
001562  000001      SYNB02: .BLKW 1             ;SYNC TWO
001564  000001      DV02.C: .BLKW 1             ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 02
001566  000001      SYNC02: .BLKW 1             ;SYNC TWO
001570  000001      DV02.D: .BLKW 1             ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 02
001572  000001      SYND02: .BLKW 1             ;SYNC TWO

001574  000001      DVCR03: .BLKW 1             ;CONTROL STATUS REGISTER FOR DV11 NUMBER 03
001576  000001      DVTR03: .BLKW 1             ;VECTOR "A" FOR DV11 NUMBER 03
001600  000001      DV03.A: .BLKW 1             ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 03
001602  000001      SYNA03: .BLKW 1             ;SYNC TWO
001604  000001      DV03.B: .BLKW 1             ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 03
001606  000001      SYNB03: .BLKW 1             ;SYNC TWO
001610  000001      DV03.C: .BLKW 1             ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 03
001612  000001      SYNC03: .BLKW 1             ;SYNC TWO
001614  000001      DV03.D: .BLKW 1             ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 03
001616  000001      SYND03: .BLKW 1             ;SYNC TWO

001620  000001      DVCR04: .BLKW 1             ;CONTROL STATUS REGISTER FOR DV11 NUMBER 04
001622  000001      DVTR04: .BLKW 1             ;VECTOR "A" FOR DV11 NUMBER 04
001624  000001      DV04.A: .BLKW 1             ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 04
001626  000001      SYNA04: .BLKW 1             ;SYNC TWO
001630  000001      DV04.B: .BLKW 1             ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 04
001632  000001      SYNB04: .BLKW 1             ;SYNC TWO
001634  000001      DV04.C: .BLKW 1             ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 04
```

```
856  001636  000001                  SYNC04: .BLKW 1         ;SYNC TWO
857  001640  000001                  DV04.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 04
858  001642  000001                  SYND04: .BLKW 1         ;SYNC TWO
859
860  001644  000001                  DVCR05: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 05
861  001646  000001                  DVTR05: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 05
862  001650  000001                  DV05.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 05
863  001652  000001                  SYNA05: .BLKW 1         ;SYNC TWO
864  001654  000001                  DV05.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 05
865  001656  000001                  SYNB05: .BLKW 1         ;SYNC TWO
866  001660  000001                  DV05.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 05
867  001662  000001                  SYNC05: .BLKW 1         ;SYNC TWO
868  001664  000001                  DV05.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 05
869  001666  000001                  SYND05: .BLKW 1         ;SYNC TWO
870
871  001670  000001                  DVCR06: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 06
872  001672  000001                  DVTR06: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 06
873  001674  000001                  DV06.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 06
874  001676  000001                  SYNA06: .BLKW 1         ;SYNC TWO
875  001700  000001                  DV05.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 06
876  001702  000001                  SYNB06: .BLKW 1         ;SYNC TWO
877  001704  000001                  DV06.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 06
878  001706  000001                  SYNC06: .BLKW 1         ;SYNC TWO
879  001710  000001                  DV06.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 06
880  001712  000001                  SYND06: .BLKW 1         ;SYNC TWO
881
882  001714  000001                  DVCR07: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 07
883  001716  000001                  DVTR07: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 07
884  001720  000001                  DV07.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 07
885  001722  000001                  SYNA07: .BLKW 1         ;SYNC TWO
886  001724  000001                  DV07.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 07
887  001726  000001                  SYNB07: .BLKW 1         ;SYNC TWO
888  001730  000001                  DV07.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 07
889  001732  000001                  SYNC07: .BLKW 1         ;SYNC TWO
890  001734  000001                  DV07.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 07
891  001736  000001                  SYND07: .BLKW 1         ;SYNC TWO
892
893  001740  000000                  DV.END: 000000
```

```
                                                    ;PROGRAM INITIALIZATION
                                                    ;LOCK OUT INTERRUPTS
                                                    ;SET UP PROCESSOR STACK
                                                    ;SET UP POWER FAIL VECTOR
                                                    ;CLEAR PROGRAM CONTROL FLAGS AND COUNTS
                                                    ;TYPE TITLE MESSAGE

001742  012737  000340  177776        .START: MOV     #340,PS                 ;LOCK OUT INTERRUPTS
001750  012706  001200                MOV     #STACK,SP               ;SET UP STACK
001754  012737  004402  000024        MOV     #.PFAIL,@#24            ;SET UP POWER FAIL VECTOR
001762  113737  001301  001303        MOVB    DVNUM,SAVNUM            ;SAVE NUMBER OF DEVICES IN SYSTEM.
001770  005037  001230                CLR     PASCNT                  ;CLEAR PASS COUNT
001774  105037  001311                CLRB    ERRFLG                  ;CLEAR ERROR FLAG
002000  105037  001313                CLRB    QV.FLG                  ;ZERO QUICK VERIFY FLAG
002004  012737  001500  001306        MOV     #DV.MAP,CREAM           ;GET MAP POINTER.
002012  112737  000001  001304        MOVB    #1,RUN                  ;POINT POINTER TO FIRST DEVICE.
002020  005037  001232                CLR     ERRCNT                  ;CLEAR ERROR COUNT
002024  005037  001234                CLR     LSTERR                  ;CLEAR LAST ERROR POINTER
002030  012737  000001  001235        MOV     #1,TSTNO                ;SET UP FOR TEST 1
002036  012737  001742  001214        MOV     #.START,RETURN          ;SET UP FOR POWER FAIL BEFORE
                                                                        ;TESTING STARTS
002044  105737  001310                TSTB    INIFLG                  ;HAS INITIALIZATION BEEN PERFORMED
002050  001063                        BNE     1$                      ;BR IF YES
002052  013746  000004                MOV     4,-(SP)                 ;
002056  013746  000006                MOV     6,-(SP)                 ;
002062  005037  000006                CLR     6                       ;
002066  012737  002104  000004        MOV     #80$,4                  ;
002074  005777  177102                TST     @SWR                    ;
002100  000240                        NOP                             ;
002102  000407                        BR      81$                     ;
002104  022626                80$:    CMP     (SP)+,(SP)+             ;
002106  012737  000174  001200        MOV     #LIGHT,LIGHTS           ;
002114  012737  000176  001202        MOV     #SSWR,SWR               ;
002122  012637  000006                81$:    MOV     (SP)+,6         ;
002126  012637  000004                MOV     (SP)+,4                 ;
002132  104402  001000                TYPE    ,MTITLE                 ;TYPE TITLE MESSAGE
002136  105137  001310                COMB    INIFLG                  ;IF NOT SET FLAG AND DO
002142  105777  177034                TSTB    @SWR                    ;BIT7=1??
002146  100402                        BMI     16$                     ;BR IF NO AUTO SIZE
002150  004737  006624                JSR     PC,CSRMAP               ;GO DO THE AUTO SIZE
002154  104402  005461                16$:    TYPE    ,XHEAD          ;TYPE HEADER
002160  012737  001500  001246        MOV     #DV.MAP,TEMP1           ;SET POINTER
002166  017737  177054  001250        5$:     MOV     @TEMP1,TEMP2    ;SET DATA
002174  022737  177777  001250        CMP     #177777,TEMP2           ;ALL DONE?
002202  001406                        BEQ     1$                      ;BR IF YES
002204  104410                        CONVRT                          ;
002206  005506                        XSTATQ                          ;
002210  062737  000002  001246        ADD     #2,TEMP1                ;UPDATE POINTER
002216  000763                        BR      5$
002220  005737  000042                1$:     TST     @#42            ;IS PROGRAM RUNNING UNDER MONITOR
002224  001030                        BNE     3$                      ;BR IF YES
002226  032777  000001  176746        BIT     #SW00,@SWR              ;SELECT SPECIFIC DEVICES??
002234  001424                        BEQ     3$                      ;BR IF NO.
002236  104402  005402                TYPE    ,MNEW                   ;TYPE THE MESSAGE.
002242  005000                        CLR     R0                      ;ZERO DATA LIGHTS
```

```
                                                HALT                    ;WAIT FOR USER TO TELL WHAT DEVICES TO R
002244  000000                                  CMPB    @SWR,SAVACT     ;IS THE NUMBER VALID?
002246  127737  176730  001302                  BLOS    2$              ;BR IF NUMBER IS OK.
002254  101404                                  TYPE    .MERR3          ;TELL USER OF INVALID NUMBER.
002256  104402  005243                          HALT                    ;STOP EVERY THING.
002262  000000                                  BR      .-2             ;RESTART THE PROGRAM AGAIN.
002264  000776                                  MOVB    @SWR,DVACTV     ;GET NEW DEVICE PATTERN
002266  117737  176710  001300          2$:     MOVB    DVACTV,R0       ;SHOW THE USER WHAT HE SELECTED.
002274  113700  001300                          BIC     #^C<377>,R0     ;USE ONLY LOW BYTE.
002300  042700  177400                          HALT                    ;CONTINUE DYNAMIC SWITCHES.
002304  000000                          3$:     MOV     #300,R0         ;PREPARE TO CLEAR THE FLOATING
002306  012700  000300                          MOV     #302,R1         ;VECTOR AREA. 300-776
002312  012701  000302                  4$:     MOV     R1,(R0)+        ;START PUTTING "PC+2 - HALT"
002316  010120                                  CLR     (R1)+           ;IN VECTOR AREA.
002320  005021                                  CMP     (R0)+,(R1)+     ;POP POINTERS
002322  022021                                  CMP     #1000,R0        ;ALL DONE??
002324  022700  001000                          BNE     4$              ;BR IF NO.
002330  001372
                                                ;TEST START AND RESTART
                                                ;----------------------

002332  012737  000340  177776          .BEGIN: MOV     #340,PS         ;LOCK OUT INTERRUPTS
002340  012706  001200                          MOV     #STACK,SP       ;SET UP STACK
002344  005737  000042                          TST     @#42            ;IS PROGRAM UNDER MONITOR CONTROL
002350  001023                                  BNE     3$              ;BR IF YES
002352  032777  000004  176622                  BIT     #BIT2,@SWR      ;CHECK FOR LOCK ON TEST
002360  001411                                  BEQ     1$              ;BR IF NO LOCK DESIRED.
002362  104402  005301                          TYPE    ,MLOCK          ;TYPE LOCK SELECTED.
002366  012737  000240  002702                  MOV     #NOP,TTST       ;ADJUST SCOPE ROUTINE.
002374  012737  000240  002704                  MOV     #NOP,TTST+2     ;SET UP TO LOCK
002402  000406                                  BR      2$              ;CONTINUE ALONG.
002404  013737  003014  002702          1$:     MOV     BRW,TTST        ;PREPARE NORMAL SCOPE ROUTINE
002412  013737  003016  002704                  MOV     BRX,TTST+2      ;LOCK NOT SELECTED, SET UP FOR NORMAL SC
002420                                  2$:
002420  012737  005666  001214          3$:     MOV     #CYCLE,RETURN   ;START AT "CYCLE" FIND WHICH DEVICE TO T
002426  104402  005171                  4$:     TYPE    ,MR             ;TYPE R
002432  000177  176556                          JMP     @RETURN         ;START TESTING
```

```
                                                        ;END OF PASS
                                                        ;TYPE NAME OF TEST
                                                        ;UPDATE PASS COUNT
                                                        ;CHECK FOR EXIT TO ACT-11
                                                        ;RESTART TEST

 002436  000005                               .EOP:    RESET                    ;MAKE THE  WORLD CLEAN AGAIN.
 002440  005037  001234                               CLR     LSTERR           ;CLEAR LAST ERROR PC
 002444  105037  001311                               CLRB    ERRFLG           ;CLEAR ERROR FLAG
 002450  005237  001230                               INC     PASCNT           ;UPDATE PASS COUNT
 002454  013777  001230  176516                       MOV     PASCNT,@LIGHTS   ;DISPLAY PASS COUNT
 002462  104402  005145                               TYPE    ,MEPASS          ;TYPE END PASS
 002466  104402  005330                               TYPE    ,MCSRX           ;TYPE CSR
 002472  104411  002604                               CNVRT   ,XCSR            ;SHOW IT
 002476  104402  005336                               TYPE    ,MVECX           ;TYPE VECTOR
 002502  104411  002612                               CNVRT   ,XVEC            ;SHOW IT
 002506  104402  005344                               TYPE    ,MPASSX          ;TYPE PASSES
 002512  104411  002620                               CNVRT   ,XPASS           ;SHOW IT
 002516  104402  005355                               TYPE    ,MERRX           ;TYPE ERRORS
 002522  104411  002626                               CNVRT   ,XERR            ;SHOW IT
 002526  105337  001303                               DECB    SAVNUM           ;ARE ALL DEVICES TESTED?
 002532  001017                                       BNE     RESTRT           ;BR IF NO.
 002534  112737  000377  001313                       MOVB    #377,QV.FLG      ;SET THE QUICK VERIFY FLAG.
 002542  113737  001301  001303                       MOVB    DVNUM,SAVNUM     ;RESTORE THE COUNT
 002550  013701  000042                               MOV     @#42,R1          ;CHECK FOR ACT-11 OR DDP
 002554  001406                                       BEQ     RESTRT           ;IF NOT, CONTINUE TESTING
 002556  000005                                       RESET                    ;STOP THE SHOW--CLEAR THE WORLD
 002560                                  LOGICAL:
 002560  004711                                       JSR     PC,(R1)
 002562  000240                                       NOP
 002564  000240                                       NOP
 002566  000240                                       NOP
 002570  000240                                       NOP
 002572  012737  005666  001214           RESTRT:     MOV     #CYCLE,RETURN
 002600  000137  005666                               JMP     CYCLE
 002604  000001                           XCSR:       1
 002606     006     002                               .BYTE   6,2
 002610  001362                                       DVSCR
 002612  000001                           XVEC:       1
 002614     003     002                               .BYTE   3,2
 002616  001352                                       DVRVEC
 002620  000001                           XPASS:      1
 002622     006     002                               .BYTE   6,2
 002624  001230                                       PASCNT
 002626  000001                           XERR:       1
 002630     006     002                               .BYTE   6,2
 002632  001232                                       ERRCNT

                                                     ;SCOPE LOOP AND INTERATION HANDLER
                                                     ;--------------------------------

 002634                                  .SCOPE:
 002634  022737  177570  001202                       CMP     #177570,SWR      ;IS THERE A REAL SWR'
 002642  001411                                       BEQ     64$              ;BR IF YES
 002644  017746  176336                               MOV     @TKDBR,-(SP)     ;SAVE KEYBOARD CHAR
 002650  042716  000200                               BIC     #BIT7,(SP)       ;CLEAR PARITY BIT
```

```
1043  002654  122726  000007                          CMPB    #7,(SP)+        ;WAS IT CNTRL 'G' ?
1044  002660  001002                                  BNE     .+6             ;BR IF NO.
1045  002662  004737  004640                          JSR     PC,SERV.G       ;SERVICE "CNTRL 'G'".
1046  002666  005037  001234              64$:        CLR     LSTERR          ;CLEAR LAST ERROR PC.
1047  002672  010016                                  MOV     R0,(SP)         ;SAVE R0 ON THE STACK
1048  002674  032777  040000  176300                  BIT     #BIT14,@SWR     ;"LOOP ON THIS TEST"?
1049  002702  001407                      TTST:       BEQ     1$              ;BR IF NO.  (IF LOCK SW01=1; THIS LOC =240)
1050  002704  000437                                  BR      3$              ;GOTO 3$    (IF LOCK SW01=1; THIS LOC =240)
1051  002706  105777  176272                          TSTB    @TKCSR          ;KEYBOAD DONE?
1052  002712  100034                                  BPL     3$              ;BR IF NO. (LOCK: HIT KEY TO GOTO NEXT TEST)
1053  002714  017700  176266                          MOV     @TKDBR,R0       ;CLEAR DONE BIT
1054  002720  000415                                  BR      2$              ;CONTINUE
1055  002722  032777  004000  176252          1$:     BIT     #SW11,@SWR      ;DELETE ITERATION?  (QUICK PASS)
1056  002730  001011                                  BNE     2$              ;BR IF YES
1057  002732  105737  001313                          TSTB    QV.FLG          ;HAVE PASSES BEECOMPLETED?
1058  002736  001406                                  BEQ     2$              ;BR IF QUICK PASS.
1059  002740  005237  001224                          INC     LPCNT           ;UPDATE ITERATION COUNTER
1060  002744  023737  001224  001222                  CMP     LPCNT,ICOUNT    ;ARE ALL ITERATIONS DONE??
1061  002752  001014                                  BNE     3$              ;BR IF NOT YET
1062  002754  105037  001311              2$:         CLRB    ERRFLG          ;PREPARE FOR NEW TEST
1063  002760  005037  001224                          CLR     LPCNT           ;START ICOUNTER AT 0
1064  002764  005037  001220                          CLR     LOCK
1065  002770  012737  000024  001222                  MOV     #20.,ICOUNT     ;RESET ITERATIONS
1066  002776  013737  001216  001214                  MOV     NEXT,RETURN     ;GET NEXT TEST
1067  003004  011600                      3$:         MOV     (SP),R0         ;POP R0 OFF OF THE STACK
1068  003006  022626                                  POP2SP                  ;FAKE AN "RTI"
1069  003010  000177  176200                          JMP     @RETURN         ;GO DO THE TEST
1070  003014  001407                      BRW:        1407
1071  003016  000437                      BRX:        437
1072
1073                                                  ;CHECK FOR FREEZE ON CURRENT DATA
1074                                                  ;--------------------------------
1075
1076  003020  032777  001000  176154      .SCOP1:     BIT     #SW09,@SWR      ;IS SW09=1(SET)?
1077  003026  001405                                  BEQ     1$              ;BR IF NOT SET.
1078  003030  005737  001220                          TST     LOCK
1079  003034  001402                                  BEQ     1$
1080  003036  013716  001220                          MOV     LOCK,(SP)       ;GOTO THE ADDRESS IN LOCK.
1081  003042  000002                      1$:         RTI                     ;GO BACK.
1082
1083                                                  ;TELETYPE OUTPUT ROUTINE
1084                                                  ;-----------------------
1085
1086  003044  010546                      .TYPE:      MOV     R5,-(SP)        ;SAVE R5 ON THE STACK.
1087  003046  017605  000002                          MOV     @2(SP),R5       ;GET ADDRESS OF MESSAGE.
1088  003052  062766  000002  000002                  ADD     #2,2(SP)        ;POP OVER ADDRESS.
1089  003060  032777  010000  176114          1$:     BIT     #SW12,@SWR      ;INHIBIT ALL PRINT OUT??
1090  003066  001012                                  BNE     3$              ;BR IF NO PRINT OUT WANTED (SW12=1)
1091  003070  105715                                  TSTB    (R5)            ;IS NUMBER MINUS? (MSB=1(BIT7))
1092  003072  100002                                  BPL     2$              ;BR IF NUMBER IS PLUS
1093  003074  104402  005104                          TYPE    ,MCRLF          ;TYPE A CR/LF!
1094  003100  105777  176104              2$:         TSTB    @TPCSR          ;TTY READY?
1095  003104  100375                                  BPL     2$              ;BR IF NO.
1096  003106  112577  176100                          MOVB    (R5)+,@TPDBR    ;PRINT CURRENT CHAR.
1097  003112  001362                                  BNE     1$              ;IF NOT ZERO KEEP PRINTING!
1098  003114  012605                      3$:         MOV     (SP)+,R5        ;END OF OUTPUT. RESTORE R5
```

```
1099  003116  000002                          RTI                     ;GO HOME
1100                                          ;-----------------------
1101
1102  003120  010346                  .INSTR: MOV     R3,-(SP)        ;SAVE R3 ON STACK
1103  003122  010446                          MOV     R4,-(SP)        ;SAVE R4 ON STACK
1104  003124  017637  000004  003142          MOV     @4(SP),.MSG
1105  003132  062766  000002  000004          ADD     #2,4(SP)
1106  003140  104402                  .INST1: TYPE
1107  003142  000000                  .MSG:   0
1108  003144  012704  005520                  MOV     #INBUF,R4
1109  003150  012703  000007                  MOV     #7,R3
1110  003154  105777  176024          1$:     TSTB    @TKCSR
1111  003160  100375                          BPL     1$
1112  003162  117714  176020                  MOVB    @TKDBR,(R4)
1113  003166  142714  000200                  BICB    #200,(R4)
1114  003172  122427  000015                  CMPB    (R4)+,#15
1115  003176  001417                          BEQ     INSTR2
1116  003200  105777  176004          2$:     TSTB    @TPCSR
1117  003204  100375                          BPL     2$
1118  003206  017777  175774  175776          MOV     @TKDBR,@TPDBR
1119  003214  005303                          DEC     R3
1120  003216  001356                          BNE     1$
1121  003220  012604                          MOV     (SP)+,R4
1122  003222  012603                          MOV     (SP)+,R3
1123  003224  104402  005100          .INSTE: TYPE    ,MQM
1124  003230  010346                          MOV     R3,-(SP)
1125  003232  010446                          MOV     R4,-(SP)
1126  003234  000741                          BR      .INST1
1127  003236  012604                  INSTR2: MOV     (SP)+,R4        ;RESTORE R4
1128  003240  012603                          MOV     (SP)+,R3        ;RESTORE R3
1129  003242  000002                          RTI
1130
1131                                          ;CONVERT ASCII STRING TO OCTAL
1132                                          ;-----------------------------
1133
1134  003244  010546                  .PARAM: MOV     R5,-(SP)
1135  003246  010446                          MOV     R4,-(SP)
1136  003250  016605  000004                  MOV     4(SP),R5
1137  003254  012537  003434                  MOV     (R5)+,LOLIM
1138  003260  012537  003436                  MOV     (R5)+,HILIM
1139  003264  012537  003440                  MOV     (R5)+,DEVADR
1140  003270  112537  003442                  MOVB    (R5)+,LOBITS
1141  003274  112537  003443                  MOVB    (R5)+,ADRCNT
1142  003300  010566  000004                  MOV     R5,4(SP)
1143  003304  005005                  PARAM1: CLR     R5
1144  003306  012704  005520                  MOV     #INBUF,R4
1145  003312  122714  000015                  CMPB    #15,(R4)
1146  003316  001420                          BEQ     PARERR
1147  003320  121427  000060          1$:     CMPB    (R4),#60
1148  003324  002415                          BLT     PARERR
1149  003326  121427  000067                  CMPB    (R4),#67
1150  003332  003012                          BGT     PARERR
1151  003334  142714  000060                  BICB    #60,(R4)
1152  003340  152405                          BISB    (R4)+,R5
1153  003342  122714  000015                  CMPB    #15,(R4)
1154  003346  001406                          BEQ     LIMITS
```

```
1155  003350  006305                          ASL     R5
1156  003352  006305                          ASL     R5
1157  003354  006305                          ASL     R5
1158  003356  000760                          BR      1$
1159  003360  104404                  PARERR: INSTER
1160  003362  000750                          BR      PARAM1
1161
1162                                          ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
1163                                          ;--------------------------------------
1164
1165  003364  020537  003436          LIMITS: CMP     R5,HILIM
1166  003370  101373                          BHI     PARERR
1167  003372  020537  003434                  CMP     R5,LOLIM
1168  003376  103770                          BLO     PARERR
1169  003400  133705  003442                  BITB    LOBITS,R5
1170  003404  001365                          BNE     PARERR
1171
1172                                          ;STORE NUMBER AT SPECIFIED ADDRESS
1173
1174  003406  013704  003440                  MOV     DEVADR,R4
1175  003412  010524                  1$:     MOV     R5,(R4)+
1176  003414  062705  000002                  ADD     #2,R5
1177  003420  105337  003443                  DECB    ADRCNT
1178  003424  001372                          BNE     1$
1179  003426  012604                          MOV     (SP)+,R4
1180  003430  012605                          MOV     (SP)+,R5
1181  003432  000002                          RTI
1182  003434  000000                  LOLIM:  0
1183  003436  000000                  HILIM:  0
1184  003440  000000                  DEVADR: 0
1185  003442  000000                  LOBITS: 0
1186          003443                  ADRCNT=LOBITS+1
1187
1188                                          ;SAVE PC OF TEST THAT FAILED AND R0-R5
1189                                          ;------------------------------------
1190
1191  003444  016637  000004  001276  .SAV05: MOV     4(SP),SAVPC     ;SAVE R7 (PC)
1192
1193                                          ;SAVE R0-R5
1194
1195  003452  010537  001272          SV05:   MOV     R5,SAVR5        ;SAVE R5
1196  003456  010437  001270                  MOV     R4,SAVR4        ;SAVE R4
1197  003462  010337  001266                  MOV     R3,SAVR3        ;SAVE R3
1198  003466  010237  001264                  MOV     R2,SAVR2        ;SAVE R2
1199  003472  010137  001262                  MOV     R1,SAVR1        ;SAVE R1
1200  003476  010037  001260                  MOV     R0,SAVR0        ;SAVE R0
1201  003502  000002                          RTI                     ;LEAVE.
1202
1203                                          ;RESTORE R0-R5
1204
1205  003504  013700  001260          .RES05: MOV     SAVR0,R0        ;RESTORE R0
1206  003510  013701  001262                  MOV     SAVR1,R1        ;RESTORE R1
1207  003514  013702  001264                  MOV     SAVR2,R2        ;RESTORE R2
1208  003520  013703  001266                  MOV     SAVR3,R3        ;RESTORE R3
1209  003524  013704  001270                  MOV     SAVR4,R4        ;RESTORE R4
1210  003530  013705  001272                  MOV     SAVR5,R5        ;RESTORE R5
```

```
1211  003534  000002                                    RTI                     ;LEAVE
1212
1213                                                    ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
1214                                                    ;-----------------------------------------------------
1215
1216  003536  104402  005104                  .CONVR:   TYPE    ,MCRLF
1217  003542  010046                          .CNVRT:   MOV     R0,-(SP)
1218  003544  010146                                    MOV     R1,-(SP)
1219  003546  010346                                    MOV     R3,-(SP)
1220  003550  010446                                    MOV     R4,-(SP)
1221  003552  010546                                    MOV     R5,-(SP)
1222  003554  017601  000012                           MOV     @12(SP),R1
1223  003560  062766  000002  000012                   ADD     #2,12(SP)
1224  003566  012137  003742                           MOV     (R1)+,WRDCNT
1225  003572  112137  003744                  1$:      MOVB    (R1)+,CHRCNT
1226  003576  112137  003745                           MOVB    (R1)+,SPACNT
1227  003602  013137  003746                           MOV     @(R1)+,BINWRD
1228  003606  013704  003746                  2$:      MOV     BINWRD,R4
1229  003612  113705  003744                           MOVB    CHRCNT,R5
1230  003616  012700  005562                           MOV     #TEMP,R0
1231  003622  010403                          3$:      MOV     R4,R3
1232  003624  042703  177770                           BIC     #177770,R3
1233  003630  062703  000060                           ADD     #060,R3
1234  003634  110320                                    MOVB    R3,(R0)+
1235  003636  000241                                    CLC
1236  003640  006004                                    ROR     R4
1237  003642  000241                                    CLC
1238  003644  006004                                    ROR     R4
1239  003646  000241                                    CLC
1240  003650  006004                                    ROR     R4
1241  003652  005305                                    DEC     R5
1242  003654  001362                                    BNE     3$
1243  003656  012703  005624                           MOV     #MDATA,R3
1244  003662  114023                          4$:      MOVB    -(R0),(R3)+
1245  003664  105337  003744                           DECB    CHRCNT
1246  003670  001374                                    BNE     4$
1247  003672  105737  003745                           TSTB    SPACNT
1248  003676  001405                                    BEQ     6$
1249  003700  112723  000040                  5$:      MOVB    #040,(R3)+
1250  003704  105337  003745                           DECB    SPACNT
1251  003710  001373                                    BNE     5$
1252  003712  105013                          6$:      CLRB    (R3)
1253  003714  104402  005624                           TYPE    ,MDATA
1254  003720  005337  003742                           DEC     WRDCNT
1255  003724  001322                                    BNE     1$
1256  003726  012605                                    MOV     (SP)+,R5
1257  003730  012604                                    MOV     (SP)+,R4
1258  003732  012603                                    MOV     (SP)+,R3
1259  003734  012601                                    MOV     (SP)+,R1
1260  003736  012600                                    MOV     (SP)+,R0
1261  003740  000002                                    RTI
1262  003742  000000                          WRDCNT:   0
1263  003744  000000                          CHRCNT:   0
1264          003745                          SPACNT=CHRCNT+1
1265  003746  000000                          BINWRD:   0
1266
```

```
                                        ;TRAP DISPATCH SEQUENCE
                                        ;ARGUMENT OF TRAP IS EXTRACTED
                                        ;AND USED AS OFFSET TO OBTAIN POINTER
                                        ;TO SELECTED SUBROUTINE

                                .TRPSR: MOV     (SP)+,-(SP)             ;GET PC OF RETURN
                                        MOV     #2,-(SP)                ;PC OF TRAP
                                        BIC     #1,(SP)                 ;GET TRAP
                                .TRPCK: MOV     (SP),-(SP)              ;MULTIPLY TRAP ARG BY 2
                                        BIC     #177001,(SP)            ;CLEAR UNWANTED BITS
                                        ADD     #.TRPTAB,(SP)           ;POINTER TO SUBROUTINE ADDRESS
                                        MOV     @(SP)+,(SP)             ;SUBROUTINE ADDRESS
                                        JMP     @(SP)+                  ;GO TO SUBROUTINE

                                        ;ERROR HANDLER
                                        ;-------------

                                .ERR:
004002                                  CMP     #177570,SWR             ;IS THERE A REAL SWR?
004002  022737  177570  001203          BEQ     64$                     ;BR IF YES
004010  001411                          MOV     @TKDBR,-(SP)            ;SAVE KEYBOARD CHAR
004012  017746  175170                  BIC     #BIT7,(SP)              ;CLEAR PARITY BIT
004016  042716  000200                  CMPB    #7,(SP)+                ;WAS IT CNTRL 'G' ?
004022  122726  000007                  BNE     .+6                     ;BR IF NO.
004026  001002                          JSR     PC,$SRV.G               ;SERVICE ^CNTRL 'G'^.
004030  004737  004640                  BIT     #SW12,@SWR              ;BELL ON ERROR?
004034  032777  010000  175140  64$:    BEQ     XBX                     ;BR IF NO BELL
004042  001406                          TSTB    @TPCSR                  ;TTY READY.
004044  105777  175140                  BPL     XBX                     ;DON'T WAIT IF TTY NOT READY.
004050  100003                          MOVB    #207,@TPDBR             ;PUSH A BELL AT THE TTY.
004052  112777  000207  175132
004060  032777  020000  175114  XBX:    BIT     #SW13,@SWR              ;DELETE ERROR PRINT OUT?
004066  001105                          BNE     HALTS                   ;BR IF NO PRINT OUT WANTED.
004070  021637  001234                  CMP     (SP),LSTERR             ;WAS THIS ERROR FOUND LAST TIME?
004074  001404                          BEQ     1$                      ;BR IF YES
004076  011637  001234                  MOV     (SP),LSTERR             ;RECORD BEING HERE
004102  105037  001311                  CLRB    ERRFLG                  ;PREPARE HEADER
004106  004406                  1$:     SAVOS                           ;SAVE ALL PROC REGISTERS
004110  011605                          MOV     (SP),R5                 ;GET THE PC OF ERROR
004112  162705  000002                  SUB     #2,R5                   ;GET ADDRESS OF TRAP CALL
004116  011504                          MOV     (R5),R4                 ;GET MLT INSTRUCTION
004120  006304                          ASL     R4                      ;MULT BY TWO
004122  061504                          ADD     (R5),R4                 ;DOUBLE IT
004124  006304                          ASL     R4                      ;MULT AGAIN
004126  042704  177001                  BIC     #177001,R4              ;CLEAR JUNK
004132  062704  026234                  ADD     #.ERRTAB,R4             ;GET POINTER
004136  012437  004252                  MOV     (R4)+,ERRMSG            ;GET ERROR MESSAGE
004142  012437  004254                  MOV     (R4)+,DATAHD            ;GET DATA HEADRER
004146  011437  004276                  MOV     (R4),DATABP             ;GET DATA TABLE
004152  105737  001311                  TSTB    ERRFLG                  ;TYPE HEADREER
004156  001403                          BEQ     TYPMSG                  ;BR IF YES
004160  005737  004276                  TST     DATABP                  ;DOES DATA TABLE EXIST?
004164  001040                          BNE     TYPDAT                  ;BR IF YES.
004166  104402  005104          TYPMSG: TYPE    ,MCRLF
004172  104402  005104                  TYPE    ,MCRLF
004176  005737  001220                  TST     LOCK
```

```
1323  004202  001402                        BEQ     1$
1324  004204  104402  005400                TYPE    .MASTER
1325  004210  104402  005366        1$:     TYPE    .MTSTN
1326  004214  104411  004374                CNVRT   .XTSTN          ;SHOW IT
1327  004220  104402  005454                TYPE    .MERRPC         ;TYPE PC.
1328  004224  104411  004366                CNVRT   ,ERTABO         ;SHOW IT
1329  004230  104402  005104                TYPE    ,MCRLF          ;GIVE A CR/LF
1330  004234  112737  177777  001311        MOVB    #-1,ERRFLG      ;NO MORE HEADER UNLESS NO DATA TABLE.
1331  004242  005737  004252                TST     ERRMSG          ;IS THERE AN ERROR MESSAGE?
1332  004246  001403                        BEQ     WRKD.FM         ;BR IF NO.
1333  004250  104402                        TYPE                    ;TYPE
1334  004252  000000                ERRMSG: 0                       ;    ERROR MESSAGE
1335  004254                        WRKD.FM:                        ;
1336  004254  005737  004264                TST     DATAHD          ;DATA HEADER?
1337  004260  001402                        BEQ     TYPDAT          ;BR IF NO
1338  004262  104402                        TYPE                    ;TYPE
1339  004264  000000                DATAHD: 0                       ;    DATA HEADER
1340  004266  005737  004276        TYPDAT: TST     DATABP          ;DATA TABLE?
1341  004272  001402                        BEQ     RESREG          ;BR IF NO.
1342  004274  104410                        CONVRT                  ;SHOW
1343  004276  000000                DATABP: 0                       ;    DATA TABLE
1344  004300  104407                RESREG: RESTORE                 ;RESTORE PROC REGISTERS
1345  004302  005777  174674        HALTS:  TST     @SWR            ;HALT ON ERROR?
1346  004306  100005                        BPL     EXITER          ;BR IF NO HALT ON ERROR
1347  004310  010046                        PUSHRO                  ;SAVE RO
1348  004312  016600  000002                MOV     2(SP),RO        ;SHOW ERROR PC IN DATA LIGHTS
1349  004316  000000                        HALT                    ;HALT
1350  004320  012600                        POPRO                   ;GET RO
1351  004322  005237  001232        EXITER: INC     ERRCNT          ;UPDATE ERROR COUNT
1352  004326  032777  000400  174646        BIT     #SW08,@SWR      ;GOTO TOP OF TEST?
1353  004334  001007                        BNE     1$              ;BR IF YES
1354  004336  032777  002000  174636        BIT     #SW10,@SWR      ;GOTO NEXT TEST?
1355  004344  001407                        BEQ     2$              ;BR IF NO
1356  004346  013737  001216  001214        MOV     NEXT,RETURN     ;SET FOR NEXT TEST
1357  004354  012706  001200        1$:     MOV     #STACK,SP       ;RESET SP
1358  004360  000177  174630                JMP     @RETURN         ;GOTO SPECIFIED TEST
1359  004364  000002                2$:     RTI                     ;RETURN
1360  004366  000001                ERTABO: 1
1361  004370     006     002                .BYTE   6,2
1362  004372  001276                        SAVPC
1363  004374  000001                XTSTN:  1
1364  004376     003     002                .BYTE   3,2
1365  004400  001226                        TSTNO
1366                                        ;ENTER HERE ON POWER FAILURE
1367                                        ;---------------------------
1368
1369
1370  004402                        .PFAIL:
1371  004402  012737  004414  000024        MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
1372  004410  000000                        HALT                            ;HALT ON POWER DOWN NORMAL
1373  004412  000777                        BR      .
1374
1375                                        ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
1376
1377  004414                        RESTAR:
1378  004414  012737  004402  000024        MOV     #.-PFAIL,24     ;SET UP FOR POWER FAILURE
```

```
 1379  004422  012706  001200                    MOV     #STACK,SP       ;RESET THE STACK POINTER
 1380  004426  005037  005562                    CLR     TEMP            ;READY FOR TIMMER
 1381  004432  005237  005562                    INC     TEMP            ;PLUS ONE TO THE TIMER!
 1382  004436  001375                            BNE     .-4             ;BR IF MORE TO GO
 1383  004440  104402  005107                    TYPE    ,MPFAIL         ;TYPE THE MESSAGE
 1384  004444  104411  004470                    CNVRT   @PFTAB          ;TELL WHAT TEST TO RETURN TO.
 1385  004450  105037  001311                    CLRB    ERRFLG          ;START CLEAN
 1386  004454  005037  001234                    CLR     LSTERR          ;"""""""""""""
 1387  004460  104412                            MSTCLR                  ;START CLEAN-UP OF DEVICE
 1388  004462  104413                            RAMCLR                  ;CLEAR IT ALL!
 1389  004464  000177  174524                    JMP     @RETURN         ;START DOING THAT TEST AGAIN.
 1390  004470  000001                    PFTAB:  1
 1391  004472     003     002             .BYTE   3,2
 1392  004474  001226                            TSTNO
 1393  004476  010046                    .DELAY: MOV     R0,-(SP)
 1394  004500  013700  004514                    MOV     1$,R0
 1395  004504  005300                            DEC     R0
 1396  004506  001376                            BNE     .-2
 1397  004510  012600                            MOV     (SP)+,R0
 1398  004512  000002                            RTI
 1399  004514  000036                    1$:     30.
 1400
 1401  004516                            .RAMCLR:
 1402  004516  012777  004000  174636            MOV     #MPRESET,@DVSCR ;ISSUE A MASTER CLEAR
 1403  004524  010146                            MOV     R1,-(SP)        ;SAVE R1 ON THE STACK
 1404  004526  010446                            MOV     R4,-(SP)        ;SAVE R4 ON THE STACK
 1405  004530  013701  001372                    MOV     DVSRS,R1        ;GET SECONDARY SEL. REG.
 1406  004534  013704  001376                    MOV     DVSRA,R4        ;GET SECONDARY REGISTER ACCESS REG.
 1407  004540  005014                    1$:     CLR     (R4)            ;ZERO THE SECONDARY REGISTER.
 1408  004542  062711  170361                    ADD     #^C<BIT11+BIT10+BIT9+BIT8+BIT3+BIT2+BIT1+BIT0>+BIT0,(R1
 1409  004546  001374                            BNE     1$
 1410  004550  012604                            MOV     (SP)+,R4        ;RESTORE R4
 1411  004552  012601                            MOV     (SP)+,R1        ;RESTORE R1
 1412  004554  000002                            RTI
 1413
 1414  004556                            .MSTCLR:
 1415  004556  012777  004000  174576            MOV     #MRESET,@DVSCR  ;ISSUE MASTER CLEAR.
 1416  004564  000002                            RTI
 1417
 1418  004566                            .ROMCLK:
 1419  004566  052777  000002  174566            BIS     #BIT1,@DVSCR
 1420  004574  000002                            RTI
 1421
 1422  004576                            .DATACLK:
 1423  004576  010046                            MOV     R0,-(SP)
 1424  004600  005000                            CLR     R0
 1425  004602  052777  000400  174560            BIS     #BIT8,@DVLCR
 1426  004610  017737  174554  004636    1$:     MOV     @DVLCR,3$
 1427  004616  106037  004637                    RORB    3$+1
 1428  004622  103003                            BCC     2$
 1429  004624  005200                            INC     R0
 1430  004626  001370                            BNE     1$
 1431  004630  104000                            HLT     0
 1432  004632  012600                    2$:     MOV     (SP)+,R0
 1433  004634  000002                            RTI
 1434  004636  000001                    3$:     .BLKW   1
```

```
1435
1436  004640  032777  004000  174336          SERV.G: BIT     #4000,@TKCSR    ;RX BUSY?
1437  004646  001374                                  BNE     SERV.G          ;BR IF YES
1438  004650  017737  174326  005072                  MOV     @SWR,90$        ;SAVE (SWR).
1439  004656  013777  005072  174316          1$:     MOV     90$,@SWR        ;
1440  004664  104402  005052                          TYPE    .89$            ;
1441  004670  104411  005064                          CNVRT   .89$            ;
1442  004674  104402  005074                          TYPE    .91$            ;
1443  004700  105777  174300                          TSTB    @TKCSR          ;WAIT FOR DONE.
1444  004704  100375                                  BPL     .-4             ;
1445  004706  017746  174274                          MOV     @TKDBR,-(SP)    ;
1446  004712  042716  000200                          BIC     #BIT7,(SP)      ;
1447  004716  122726  000015                          CMPB    #15,(SP)+       ;
1448  004722  001450                                  BEQ     5$              ;
1449  004724  005077  174252                          CLR     @SWR            ;
1450  004730  105777  174254          2$:     TSTB    @TPCSR          ;
1451  004734  100375                                  BPL     .-4             ;
1452  004736  016677  177776  174246                  MOV     -2(SP),@TPDBR   ;
1453  004744  000241                                  CLC                     ;
1454  004746  006177  174230                          ROL     @SWR            ;
1455  004752  006177  174224                          ROL     @SWR            ;
1456  004756  006177  174220                          ROL     @SWR            ;
1457  004762  103735                                  BCS     1$              ;ERROR
1458  004764  026627  177776  000060                  CMP     -2(SP),#60      ;
1459  004772  002731                                  BLT     1$              ;
1460  004774  026627  177776  000067                  CMP     -2(SP),#67      ;
1461  005002  003325                                  BGT     1$              ;
1462  005004  042766  177770  177776                  BIC     #10(7),-2(SP)   ;
1463  005012  056677  177776  174162                  BIS     -2(SP),@SWR     ;
1464  005020  105777  174160                          TSTB    @TKCSR          ;
1465  005024  100375                                  BPL     .-4             ;
1466  005026  017746  174154                          MOV     @TKCBR,-(SP)
1467  005032  042716  000200                          BIC     #BIT7,(SP)
1468  005036  122726  000015                          CMPB    #15,(SP)+
1469  005042  001332                                  BNE     2$              ;
1470  005044  104402  005104          5$:     TYPE    ,MCRLF          ;
1471  005050  000207                                  RTS     PC
1472
1473  005052  020377  051450  052127          99$:    .ASCIZ  <377>" (SWR)= "
1474  005060  036451  000057
1475                                          .EVEN
1476  005064  000001                          88$:    1
1477  005066     006     000                          .BYTE   6,0
1478  005070  005072                                  90$
1479  005072  000000                          90$:    .WORD   0
1480  005074  036457  000057                  91$:    .ASCIZ  "=/"
1481                                          .EVEN
1482  005100  020040  000077                  MQM:    .ASCIZ  /  ?/
1483  005104  005015     000                  MCRLF:  .ASCIZ  <15><12>
1484  005107     377  053520  020122          MPFAIL: .ASCIZ  <377>/PWR FAILED. RESTART AT TEST /
1485  005145     377  047105  020104          MEPASS: .ASCIZ  <377>/END PASS DZDVC-C
1486  005171     377  000122                  MR:     .ASCIZ  <377>/R/
1487  005174  050377  047522  051107          MERR2:  .ASCIZ  <377>/PROGRAM INDICATES NO DEVICES PRESENT./
1488  005243     377  047111  052523          MERR3:  .ASCIZ  <377>/INSUFFICIENT DATA!/
1489  005267     377  042524  052123          MTSTPC: .ASCIZ  <377>/TEST PC-/
1490  005331     377  047514  046503          MLOCK:  .ASCIZ  <377>/LOCK ON SELECTED TEST
```

```
[illegible]  005330  051503  035122  000040          MCSRX:  .ASCIZ  /CSR: /
[illegible]  005336  042526  035103  000040          MVECX:  .ASCIZ  /VEC: /
[illegible]  005344  040520  051523  051505          MPASSX: .ASCIZ  /PASSES: /
[illegible]  005355     105  051122  051117          MERRX:  .ASCIZ  /ERROPS: /
[illegible]  005366  042524  032123  047040          MTSTN:  .ASCIZ  /TEST NO: /
[illegible]  005400  000052                          MASTEK: .ASCIZ  /*/
[illegible]  005402  051727  052105  051440          MNEW:   .ASCIZ  <377>/SET SWITCH REG TO DV11'S DESIRED ACTIVE./
[illegible]  005454  041520  020072     000          MERRPC: .ASCIZ  /PC: /
[illegible]  005461     377  040515  020120          XHEAD:  .ASCIZ  <377>/MAP OF DV11 STATUS/<377
                                                     .EVEN
[illegible]  005506  000002                          XSTATQ: 2
[illegible]  005510     006     003                          .BYTE   5.3
[illegible]  005512  001246                                  TEMP1
[illegible]  005514     006     002                          .BYTE   5.2
[illegible]  005516  001250                                  TEMP2
                                                     .EVEN

                                                             ;BUFFERS FOR INPUT-OUTPUT

[illegible]  005520  000000                          INBUF:  0
                     000562                          .=.+40
[illegible]  005562  000000                          TEMP:   0
                     005624                          .=.+40
[illegible]  005624  000000                          MORTP:  0
                     005666                          .=.+40
```

```
1497
1498                                                    ;
1499                                                    ;ROUTINE USED TO "CYCLE" THROUGH UP TO EIGHT DV11'S
1500                                                    ;THIS ROUTINE SETS UP THE CONTROL ADDRESS FOR THE DIAGNOSTIC
1501                                                    ;AND RUNS THE SPECIFIED DV11'S.   THIS ROUTINE *MUST*
1502                                                    ;BE RUN FIRST BEFORE ENTERING THE DIAGNOSTIC FOR THE
1503                                                    ;SETUP NECESSARY.
1504                                                    ;
1505 005666 105737 001300              CYCLE:  TSTB    DVACTV          ;ARE ANY DV11'S TO BE TESTED?
1506 005672 001004                             BNE     1$              ;BR IF OK.
1507 005674 104402 005174                      TYPE    .MERR2          ;NO DV11'S SELECTED!!
1508 005700 000000                             HALT                    ;STOP THE SHOW.
1509 005702 000776                             BR      .-2             ;DISQUALIFY CONT. SW.
1510 005704 133737 001304 001300       1$:     BITB    RUN,DVACTV      ;IS THIS ONE "ACTIVE"
1511 005712 001020                             BNE     2$              ;BR IF GOOD ONE FOUND.
1512 005714 000241                             CLC                     ;CLEAR PROC. CARRY BIT.
1513 005716 106137 001304                      ROLB    RUN             ;UPDATE POINTER
1514 005722 105537 001304                      ADCB    RUN             ;CATCH CARRY FROM RUN
1515 005726 062737 000024 001306               ADD     #24,CREAM       ;UPDATE ADDRESS POINTER.
1516 005734 022737 001740 001306               CMP     #DV.END,CREAM
1517 005742 001360                             BNE     1$              ;KEEP GOING; NOT ALL TESTED FOR.
1518 005744 012737 001500 001306               MOV     #DV.MAP,CREAM   ;RESET ADDRESS POINTER.
1519 005752 000754                             BR      1$              ;KEEP LOOKING FOR ACTIVE DV11
1520 005754 000241                     2$:     CLC                     ;CLEAR PROC. CARRY.
1521 005756 106137 001304                      ROLB    RUN             ;UPDATE POINTER.
1522 005762 105537 001304                      ADCB    RUN             ;CATCH CARRY.
1523 005766 013700 001306                      MOV     CREAM,R0        ;GET ADDRESS POINTER.
1524 005772 062737 000024 001306               ADD     #24,CREAM       ;UPDATE.
1525 006000 022737 001740 001306               CMP     #DV.END,CREAM
1526                                                                    ;ALL DONE?
1527 006006 001003                             BNE     3$              ;BR IF NO.
1528 006010 012737 001500 001306               MOV     #DV.MAP,CREAM   ;RESTORE POINTER.
1529 006016 012037 001362              3$:     MOV     (R0)+,DVSCR     ;LOAD SYSTEM CTRL. REG
1530 006022 012037 001352                      MOV     (R0)+,DVRVEC    ;LOAD VECTOR
1531 006026 012037 001416                      MOV     (R0)+,L00.03    ;GET LINE PARAMETERS. 00-03
1532 006032 012037 001426                      MOV     (R0)+,SYNC2A    ;
1533 006036 012037 001420                      MOV     (R0)+,L04.07    ;                     04-07
1534 006042 012037 001430                      MOV     (R0)+,SYNC2B    ;
1535 006046 012037 001422                      MOV     (R0)+,L08.11    ;                     08-11
1536 006052 012037 001432                      MOV     (R0)+,SYNC2C    ;
1537 006056 012037 001424                      MOV     (R0)+,L12.15    ;                     12-15
1538 006062 012037 001434                      MOV     (R0)+,SYNC2D    ;
1539 006066 012700 000002                      MOV     #2,R0           ;SAVE CORE THIS WAY!
1540 006072 013737 001362 001364               MOV     DVSCR,DVSCRH    ;GET SYS CTRL. REG HIGH BYTE.
1541 006100 005237 001364                      INC     DVSCRH          ;GOT IT.
1542 006104 013737 001364 001366               MOV     DVSCRH,DVRIC    ;GET NXT REC. CHAR REG.
1543 006112 005237 001366                      INC     DVRIC           ;GOT IT
1544 006116 013737 001366 001370               MOV     DVRIC,DVLCR     ;GET LN. PAR.REG.
1545 006124 060037 001370                      ADD     R0,DVLCR        ;GOT IT
1546 006130 013737 001370 001372               MOV     DVLCR,DVSRS     ;GET SEC. REG. SEL. REG.
1547 006136 060037 001372                      ADD     R0,DVSRS        ;GOT IT
1548 006142 013737 001372 001374               MOV     DVSRS,DVSRSH    ;GET HIGH BYTE.
1549 006150 005237 001374                      INC     DVSRSH          ;GOT IT
1550 006154 013737 001374 001376               MOV     DVSRSH,DVSRA    ;SEC. REG. ACCESS.
1551 006162 005237 001376                      INC     DVSRA           ;GOT IT
```

```
1553  006166  013737  001376  001400          MOV     DVSRA,DVSFR     ;SPEC. FUN. REG.
1554  006174  060037  001400                  ADD     R0,DVSFR        ;
1555  006200  013737  001400  001402          MOV     DVSFR,DVNSR     ;NPR STAT. REG.
1556  006206  060037  001402                  ADD     R0,DVNSR        ;
1557  006212  013737  001402  001404          MOV     DVNSR,RESV16    ;RESERVED REG
1558  006220  060037  001404                  ADD     R0,RESV16       ;
1559
1560  006224  013737  001352  001354          MOV     DVRVEC,DVRLVL   ;PTY LVL
1561  006232  060037  001354                  ADD     R0,DVRLVL       ;
1562  006236  013737  001354  001356          MOV     DVRLVL,DVTVEC   ;TX VEC
1563  006244  060037  001356                  ADD     R0,DVTVEC       ;
1564  006250  013737  001356  001360          MOV     DVTVEC,DVTLVL   ;TX LVL
1565  006256  060037  001360                  ADD     R0,DVTLVL
1566
1567  006262  012700  001416                  MOV     #L00.03,R0      ;LOAD STAUS 00-03
1568  006266  012701  001406                  MOV     #MASK.A,R1      ;PREPARE MASK.
1569  006272  012702  001412                  MOV     #CLK.A,R2       ;PREPARE CLOCKS
1570  006276  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1571
1572  006302  012700  001420                  MOV     #L04.07,R0      ;LOAD STAUS 00-03
1573  006306  012701  001407                  MOV     #MASK.B,R1      ;PREPARE MASK.
1574  006312  012702  001413                  MOV     #CLK.B,R2       ;PREPARE CLOCKS
1575  006316  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1576
1577  006322  012700  001422                  MOV     #L08.11,R0      ;LOAD STAUS 00-03
1578  006326  012701  001410                  MOV     #MASK.C,R1      ;PREPARE MASK.
1579  006332  012702  001414                  MOV     #CLK.C,R2       ;PREPARE CLOCKS
1580  006336  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1581
1582  006342  012700  001424                  MOV     #L12.15,R0      ;LOAD STAUS 00-03
1583  006346  012701  001411                  MOV     #MASK.D,R1      ;PREPARE MASK.
1584  006352  012702  001415                  MOV     #CLK.D,R2       ;PREPARE CLOCKS
1585  006356  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1586  006362  032777  000002  172612          BIT     #SW01,@SWR
1587  006370  001445                          BEQ     7$
1588  006372                          4$:
1589  006372  005737  000042                  TST     @#42
1590  006376  001042                          BNE     7$
1591  006400  104402  005104                  TYPE    .MCRLF
1592  006404  104403                          INSTR
1593  006406  005366                          MTSTN
1594  006410  104405                          PARAM
1595  006412  000001                          1
1596  006414  001000                          1000
1597  006416  001226                          TSTNO
1598  006420     000                  .BYTE   0
1599  006421     001                  .BYTE   1
1600  006422  012700  007256                  MOV     #TST1,R0
1601  006426  022710                  5$:     CMP     (PC)+,(R0)
1602  006430  012737                          MOV     (PC)+,@(PC)+
1603  006432  001015                          BNE     6$
1604  006434  023760  001226  000002          CMP     TSTNO,2(R0)
1605  006442  001011                          BNE     6$
1606  006444  022760  001226  000004          CMP     #TSTNO,4(R0)
1607  006452  001005                          BNE     6$
1608  006454  010037  001214                  MOV     R0,RETURN
```

```
1609  006460  104402  005104                          TYPE    ,MCRLF
1610  006464  000412                                  BR      8$
1611  006466  005720                          6$:     TST     (R0)+
1612  006470  020027  017056                          CMP     R0,#TLAST+10
1613  006474  001354                                  BNE     5$
1614  006476  104402  005100                          TYPE    ,MQM
1615  006502  000733                                  BR      4$
1616  006504  012737  007256  001214          7$:     MOV     #TST1,RETURN    ;PREPARE RETURN ADDRESS
1617  006512  000177  172476                  8$:     JMP     @RETURN         ;GO START TESTING.
1618
1619  006516  011003                          FIX.00: MOV     (R0),R3         ;GET PARAMETERS.
1620  006520  042703  176377                          BIC     #^C<1400>,R3    ;CLEAR JUNK.
1621  006524  005703                                  TST     R3              ;TEST FOR EIGHT BITS.
1622  006526  001004                                  BNE     1$              ;BR IF NOT 8 BITS.
1623  006530  105011                                  CLRB    (R1)            ;SET
1624  006532  112712  000010                          MOVB    #8.,(R2)        ;
1625  006536  000424                                  BR      4$
1626  006540  022703  000400                  1$:     CMP     #400,R3         ;CHECK FOR SEVEN BITS.
1627  006544  001005                                  BNE     2$              ;BR IF NOT 7 BITS.
1628  006546  112711  000200                          MOVB    #200,(R1)       ;
1629  006552  112712  000007                          MOVB    #7,(R2)         ;
1630  006556  000414                                  BR      4$
1631  006560  022703  001000                  2$:     CMP     #1000,R3        ;CHECK FOR SIX BITS.
1632  006564  001005                                  BNE     3$              ;BR IF NOT SIX BITS.
1633  006566  112711  000300                          MOVB    #300,(R1)       ;
1634  006572  112712  000006                          MOVB    #6,(R2)         ;
1635  006576  000404                                  BR      4$
1636  006600  112711  000340                  3$:     MOVB    #340,(R1)       ;IF NONE OF THE ABOVE; MUST BE 5 BITS.
1637  006604  112712  000005                          MOVB    #5,(R2)         ;
1638  006610  032710  040000                  4$:     BIT     #PARBIT,(R0)    ;PARITY ENABLED?
1639  006614  001401                                  BEQ     5$              ;IF =0; THEN NO PARITY.
1640  006616  105212                                  INCB    (R2)            ;PLUS ONE TO THE CLOCK!
1641  006620  000207                          5$:     RTS     PC              ;
1642
1643                                                  ;*ROUTINE USED TO "AUTO SIZE" THE DV11
1644                                                  ;*CSR AND VECTOR.
1645                                                  ;*NOTE: THE CSR MAY BE ANY WHERE IN THE FLOATING
1646                                                  ;*      ADDRESS RANGE (175000:175400)
1647                                                  ;*      AND THE VECTOR MAY BE ANY WHERE IN THE
1648                                                  ;*      FLOATING VECTOR RANGE (300:770)
1649                                                  ;*
1650
1651  006622                                  AUTO.SIZE:
1652  006622  000005                                  RESET                   ;INSURE A BUS INIT.
1653  006624  012702  001500                  CSRMAP: MOV     #DV.MAP,R2      ;LOAD MAP POINTER.
1654  006630  005022                          1$:     CLR     (R2)+           ;ZERO ENTIRE MAP
1655  006632  022702  001740                          CMP     #DV.END,R2      ;ALL DONE?
1656  006636  001374                                  BNE     1$              ;BR IF NO
1657  006640  105037  001301                          CLRB    DVNUM           ;SET OCTAL NUMBER OF DV11'S TO 0
1658  006644  012702  001500                          MOV     #DV.MAP,R2
1659  006650  012701  175000                          MOV     #175000,R1      ;SET FOR FIRST ADDRESS TO BE TESTED
1660  006654  012737  007074  000004                  MOV     #6$,@#4         ;SET FOR NON-EXISTANT DEVICE TIME OUT
1661  006662  005711                          2$:     TST     (R1)            ;IF DV11 DVSCR S/B 0
1662  006664  001037                                  BNE     3$              ;IF NO DEV ; TRAP TO 4. IF NO BIT 8 THEN NO DV11
1663  006666  022761  177777  000012                  CMP     #177777,12(R1)  ;IF DV11 THEN DVSFR S/B ALL 1'S ON INIT!
1664  006674  001033                                  BNE     3$              ;BR IF NOT DV11
```

```
1665  006676  005761  000016                          TST     16(R1)          ;IF DV11 THEN RESV16 S/B ALL 0'S
1666  006702  001030                                  BNE     3$              ;BR IF NOT DV11
1667                                          ;AT THIS POINT IT IS ASSUMED THAT R1 HOLDS A DV11 CSR ADDRESS.
1668  006704  010122                          MOV     R1,(R2)+        ;STORE CSR IN CORE TABLE.
1669  006706  005722                          TST     (R2)+           ;POP OVER VECTOR STORE AREA
1670  006710  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 1 STAT AND SYNC
1671  006714  052722  000062                  BIS     #62,(R2)+       ;
1672  006720  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 2 STAT AND SYNC
1673  006724  052722  000062                  BIS     #62,(R2)+       ;
1674  006730  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 3 STAT AND SYNC
1675  006734  052722  000062                  BIS     #62,(R2)+       ;
1676  006740  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 4 STAT AND SYNC
1677  006744  052722  000062                  BIS     #62,(R2)+       ;
1678  006750  105237  001301                  INCB    DVNUM           ;UPDATE DEVICE COUNTER
1679  006754  122737  000010  001301          CMPB    #10,DVNUM       ;ARE MAX. NO. OF DEV FOUND?
1680  006762  001405                          BEQ     100$            ;YES DON'T LOOK FOR ANY MORE.
1681  006764  062701  000010          3$:     ADD     #10,R1          ;UPDATE CSR POINTER ADDRESS
1682  006770  022701  175400                  CMP     #175400,R1
1683  006774  001332                          BNE     2$              ;BR IF MORE ADDRESS TO CHECK.
1684  006776  012722  177777          100$:   MOV     #177777,(R2)+   ;TERMINATER.
1685  007002  105037  001300                  CLRB    DVACTV
1686  007006  105737  001301                  TSTB    DVNUM           ;WERE ANY DV11'S FOUND AT ALL?
1687  007012  001423                          BEQ     5$              ;ERROR AUTO SIZER FOUND NO DV11'S IN THIS SYS.
1688  007014  113701  001301                  MOVB    DVNUM,R1
1689  007020  110137  001303                  MOVB    R1,SAVNUM       ;SAVE NUMBER OF DEVICES
1690  007024  000241          4$:     CLC
1691  007026  106137  001300                  ROLB    DVACTV          ;GENERATE ACTIVE REGISTER OF DEVICES.
1692  007032  105237  001300                  INCB    DVACTV          ;SET THE BIT
1693  007036  005301                          DEC     R1
1694  007040  001371                          BNE     4$              ;BR IF MORE TO GENERATE
1695  007042  012737  000006  000004          MOV     #6,@#4          ;RESTORE TRAP VECTOR
1696  007050  113737  001300  001302          MOVB    DVACTV,SAVACT   ;SAVE ACTIVE REGISTER
1697  007056  000137  007102                  JMP     VECMAP          ;GO FIND THE VECTOR NOW.
1698  007062  104402  005174          5$:     TYPE    ,MERR2          ;NOTIFY OPR THAT NO DV11'S FOUND.
1699  007066  005000                          CLR     R0              ;MAKE DATA LIGHTS ZERO
1700  007070  000000                          HALT                    ;STOP THE SHOW
1701  007072  000776                          BR      .-2             ;DISABLE CONT. SW.
1702  007074  012716  006754          6$:     MOV     #3$,(SP)        ;ENTERED BY NON-EXISTANT TIME-OUT.
1703  007100  000002                          RTI                     ;RETURN TO MAINSTREAM
1704
1705  007102  012737  000340  000022  VECMAP: MOV     #340,@#22       ;SET IOT TRAP PRIO TO 7
1706  007110  012737  007232  000020          MOV     #4$,@#20        ;SET IOT TRAP VECTOR
1707  007116  012702  001500                  MOV     #DV.MAP,R2      ;SET SOFTWARE POINTER
1708  007122  012700  000300                  MOV     #300,R0         ;FLOATING VECTORS START HERE.
1709  007126  012701  000302                  MOV     #302,R1         ;PC OF IOT INSTR.
1710  007132  010120          1$:     MOV     R1,(R0)+        ;START FILLING VECTOR AREA
1711  007134  012721  000004                  MOV     #4,(R1)+        ;WITH .+2; IOT
1712  007140  022021                          CMP     (R0)+,(R1)+     ;ADD 2 TO R0 +R1
1713  007142  020127  001000                  CMP     R1,#1000
1714  007146  101771                          BLOS    1$              ;BR IF MORE TO FILL
1715  007150  113737  001300  001246          MOVB    DVACTV,TEMP1    ;STORE TEMPORALLY
1716  007156  006037  001246          2$:     ROR     TEMP1           ;BRING OUT A BIT
1717  007162  103034                          BCC     5$              ;BR IF ALL DONE
1718  007164  005037  177776                  CLR     PS              ;ZERO CPU PRIO
1719  007170  012772  001300  000000          MOV     #BIT9+BIT7+BIT6,@(R2)
1720  007176  005000                          CLR     R0              ;ATTEMPT TO FORCE AN INTERUPT
```

```
1721  007200  005200                          INC     R0              ;STALL
1722  007202  001376                          BNE     .-2             ;       FOR TIME TO INTERUPT
1723  007204  052762  000300  000002          BIS     #300,2(R2)      ;NO INTERUPT ASSUME 300 AND FIX DV11 LATER
1724  007212  042772  176777  000000    3$:   BIC     #^C<BIT9>,@(R2)
1725  007220  005072  000000                  CLR     @(R2)
1726  007224  062702  000024                  ADD     #24,R2          ;POP SOFTWARE POINTER
1727  007230  000752                          BR      2$              ;KEEP GOING
1728  007232  051662  000002            4$:   BIS     (SP),2(R2)      ;GET VECTOR ADDRESS
1729  007236  042762  000007  000002        BIC     #7,2(R2)        ;CLEAR JUNK
1730  007244  022626                          CMP     (SP)+,(SP)+     ;POP IOT JUNK OFF STACK
1731  007246  012716  007212                  MOV     #3$,(SP)        ;SET FOR RETURN
1732  007252  000002                          RTI
1733  007254  000207            5$:   RTS     PC              ;ALL DONE WITH "AUTO SIZING"
1734
```

```
1735
1736                                        ;************************ TEST 1 ******************************
1737                                        ;*TEST TO VERIFY DV11 ROM DATA.
1738                                        ;*THIS PROGRAM WILL PLACE ROM ADDRESS
1739                                        ;*ZERO INTO THE DVSFR AND BIT 8
1740                                        ;*(TEST POINT 1(+3 VOLTS(TRUE))) AND BRANCH "A".
1741                                        ;*A ROM CYCLE WILL BE ISSUED AND
1742                                        ;*BRANCH "NEVER" WILL BE ASSERTED.
1743                                        ;*THE DATA WILL BE COMPARED AGAINST THE
1744                                        ;*CORE IMAGE FOR VALIDITY
1745                                        ;*AND THE ROM WILL BE CYCLED TO THE END.
1746                                        ;;*************************************************************
1747
1748                                    ;  TEST 1
1749                                    ;--------------
1750  007256  012737  000001  001226    TST1:   MOV     #1,TSTNO
1751  007264  012737  007372  001216            MOV     #TST2,NEXT
1752  007272  104412                            MSTCLR                  ;INIT DV11
1753  007274  012777  000010  172060            MOV     #BIT3,@DVSCR    ;SET "SOURCE SEL"
1754  007302  012777  000400  172070            MOV     #BIT8,@DVSFR    ;SET BRA +3 VOLT (SURE TRUE)
1755  007310  012700  023750                    MOV     #ROMDATA,R0     ;GET SOFTWARE POINTER FOR COMPARISON
1756  007314  005002                            CLR     R2              ;ZERO ROM ADDRESS IMAGE
1757  007316  012777  000002  172036            MOV     #BIT1,@DVSCR    ;SET ROM CLK
1758  007324  052777  000004  172030            BIS     #BIT2,@DVSCR    ;SET BRANCH DSABLE
1759  007332  011005                    1$:     MOV     (R0),R5         ;GET SOFTWARE ROM IMAGE
1760  007334  017704  172040                    MOV     @DVSFR,R4       ;GET ACTUAL ROM DATA
1761  007340  020504                            CMP     R5,R4           ;ROM DATA CORRECT
1762  007342  001401                            BEQ     2$              ;BR IF YES
1763  007344  104001                            HLT     1               ;ROM DATA COMPARISOM ERROR
1764  007346  005720                    2$:     TST     (R0)+           ;UPDATE SOFTWARE POINTER
1765  007350  022700  025536                    CMP     #ENDROM,R0      ;ALL DATA DONE
1766  007354  001405                            BEQ     3$              ;BR IF YES
1767  007356  052777  000002  171776            BIS     #BIT1,@DVSCR    ;CLK ROM
1768  007364  005202                            INC     R2              ;UPDATE ROM IMAGE ADDRESS
1769  007366  000761                            BR      1$              ;CONT TESTING
1770  007370  104400                    3$:     SCOPE                   ;SCOPE THIS TEST
1771
1772
1773                                        ;************************ TEST 2 ******************************
1774                                        ;*TEST OF DV11 MICRO PROCESSOR "FREE RUNNING"
1775                                        ;*TEST TO XMIT ONE CHAR (ALL 0'S) MAKING SURE THAT
1776                                        ;*BIT 15 OF DVSCR =1
1777                                        ;*TX WC PRIMARY =0
1778                                        ;*TX BA PRIMARY =#TXBAP+1
1779                                        ;*LINE STATE =BIT7
1780                                        ;*TX BA SECONDARY =0
1781                                        ;*TX WC SECONDARY =0
1782                                        ;*NPR STATUS REG 00:03 =LINE DESIRED PLUS BIT 8 SET
1783                                        ;*THIS TEST IS EXERCISED FOR BOTH ASYNC AND SYNC LINE CARDS.
1784                                        ;;*************************************************************
1785
1786                                    ;  TEST 2
1787                                    ;--------------
1788  007372  012737  000002  001226    TST2:   MOV     #2,TSTNO
1789  007400  012737  010160  001216            MOV     #TST3,NEXT
1790  007406  012700  000000                    MOV     #0,,R0          ;PLACE LINE NUMBER INTO R0
```

```
1791  007412  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1792  007420  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1793  007422  004737  007510                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1794  007426  012700  000004              100$: MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1795  007432  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1796  007440  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1797  007442  004737  007510                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1798  007446  012700  000010              101$: MOV     #8.,R0          ;LOAD LINE NUMBER
1799  007452  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1800  007460  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1801  007462  004737  007510                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1802  007466  012700  000014              102$: MOV     #12.,R0         ;LOAD LINE NO.
1803  007472  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1804  007500  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1805  007502  004737  007510                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1806  007506  104400                      103$: SCOPE                   ;SCOPE THIS TEST.
1807  007510                               105$:                        ;TEST ENTRANCE.
1808  007510  104413                          RAMCLR                  ;CLEAR ALL DV11 SECONDARY REGISTERS
1809  007512  012705  022350                  MOV     #TXTAB,R5       ;CLEAR
1810  007516  005001                          CLR     R1                      TX
1811  007520  105025                      21$: CLRB    (R5)+                   CONTROL
1812  007522  105201                          INCB    R1                      TABLE
1813  007524  001375                          BNE     21$
1814  007526  012702  000004                  MOV     #4,R2           ;SET FOR 4 LINE GROUP
1815  007532  012737  007540  001220          MOV     #1$,LOCK        ;SET FOR SW09=1
1816  007540  110077  171626                1$: MOVB    R0,@DVSRS       ;LOAD LINE NUMBER
1817  007544  004537  021026                  PERFORM ,SETREG
1818  007550     000     010                  .BYTE   000,010         ;BUS ADDRESS, CONTROL TABLE
1819  007552  021350                          TXBAP                   ;
1820  007554  022350                          TXTAB                   ;
1821  007556  005037  021350                  CLR     TXBAP           ;SET TX DATA TO 0
1822  007562  004537  021026                  PERFORM ,SETREG         ;
1823  007566     013     001                  .BYTE   013,001 ;LINE STATE, PRINCIPLE BYTE COUNT
1824  007570  000004                          BIT2                    ;LINE STATE CONTENTS
1825  007572  177777                          -1                      ;ONE CHAR
1826  007574  032737  004000  001236          BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
1827  007602  001407                          BEQ     60$             ;#BR IF NO.
1828  007604  004537  021072                  PERFORM ,LOAD.MODE      ;#
1829  007610  015000                          <BIT12+BIT11>+BIT9      ;#8 BITS/PER/CHAR
1830  007612  004537  021072                  PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
1831  007616  072000                          <BIT14+BIT13+BIT12>+BIT10
1832                                                                  ;#9600 BAUD.
1833  007620  000403                          BR      61$             ;#CONTINUE TEST.
1834  007622  004537  021072              60$: PERFORM ,LOAD.MODE      ;LOAD MODE
1835  007626  014000                          BIT12+BIT11             ;
1836  007630  005277  171526              61$: INC     @DVSCR          ;SET MICRO-PROCESSOR GO!
1837  007634  005005                          CLR     R5              ;WAIT FOR 15=1
1838  007636  005777  171520               2$: TST     @DVSCR          ;DVSCR 15=1?
1839  007642  100403                          BMI     3$              ;BR IF YES
1840  007644  104414                          DELAY                   ;WASTE TIME
1841  007646  005205                          INC     R5              ;DELAY
1842  007650  001372                          BNE     2$              ;ALL DONE?
1843  007652  013701  001362               3$: MOV     DVSCR,R1        ;SET POINTER
1844  007656  011104                          MOV     (R1),R4         ;READ SCR
1845  007660  012705  100001                  MOV     #BIT15+BIT0,R5  ;SET EXPECTED
1846  007664  020504                          CMP     R5,R4           ;RESULTS GOOD?
```

```
1847  007666  001401                                    BEQ     4$              ;BR IF OK
1848  007670  104002                                    HLT     2               ;DVSCR WRONG
1849  007672  005777  171470                  4$:       TST     @DVRIC          ;MAKE SURE NO RIC ENTRIES
1850  007676  001405                                    BEQ     5$              ;BR IF OK
1851  007700  013701  001366                            MOV     DVRIC,R1        ;SET POINTER
1852  007704  011104                                    MOV     (R1),R4         ;SAVE FOR TYPE OUT
1853  007706  005005                                    CLR     R5              ;MAKE EXPECTED=0
1854  007710  104002                                    HLT     2               ;REPORT RIC NOT=0
1855  007712  010005                  5$:       MOV     R0,R5           ;LOAD LINE NUMBER
1856  007714  052705  100400                            BIS     #BIT15+BIT8,R5  ;SILO ENTRY PRINCIPLE BYTE COUNT=0
1857  007720  012703  000020                            MOV     #16.,R3         ;SET FOR 16 REGISTERS
1858  007724  013701  001402                            MOV     DVNSR,R1        ;SET POINTER
1859  007730  011104                                    MOV     (R1),R4         ;READ NSR
1860  007732  020504                                    CMP     R5,R4           ;OK?
1861  007734  001401                                    BEQ     6$              ;BR IF OK
1862  007736  104002                                    HLT     2               ;DVNSR NOT CORRECT
1863  007740  013701  001362                  6$:       MOV     DVSCR,R1        ;SET POINTER
1864  007744  011104                                    MOV     (R1),R4         ;READ DVSCR [DID BITS CLEAR]
1865  007746  100005                                    BPL     7$              ;BR IF 15=0
1866  007750  012705  000001                            MOV     #BIT0,R5        ;SET EXPECTED RESULTS
1867  007754  104002                                    HLT     2               ;BIT 15 OF DVSCR NOT CLEARED BY READING NSR
1868  007756  005777  171420                            TST     @DVNSR          ;REFERENCE NSR
1869  007762  005303                  7$:       DEC     R3              ;ALL REGISTERS DONE?
1870  007764  001365                                    BNE     6$              ;BR IF NO
1871  007766  042777  000001  171366                    BIC     #BIT0,@DVSCR    ;CLEAR MICRO-PROCESSOR GO
1872  007774  012705  021351                            MOV     #TXBAP+1,R5     ;SET EXPECTED
1873  010000  010077  171366                            MOV     R0,@DVSRS       ;LOAD LINE NUMBER
1874  010004  017704  171366                            MOV     @DVSRA,R4       ;READ BUS ADDRESS
1875  010010  117701  171360                            MOVB    @DVSRSH,R1      ;GET SEC REG.
1876  010014  020504                                    CMP     R5,R4           ;GOOD?
1877  010016  001401                                    BEQ     8$              ;BR IF GOOD
1878  010020  104003                                    HLT     3               ;BUS ADDRESS NOT INCREMENTED CORRECTLY
1879  010022  105277  171346                  8$:       INCB    @DVSRSH         ;EXAMINE
1880  010026  005201                                    INC     R1              ;
1881  010030  017704  171342                            MOV     @DVSRA,R4       ;BYTE COUNT REGISTER
1882  010034  001402                                    BEQ     10$             ;IS IT=0?
1883  010036  005005                                    CLR     R5              ;SET EXPECTED
1884  010040  104003                                    HLT     3               ;BYTE COUNT NOT=0!
1885  010042  010077  171324                  10$:      MOV     R0,@DVSRS       ;LOAD LINE NUMBER
1886  010046  112777  000002  171320                    MOVB    #002,@DVSRSH    ;GET OPPOSITE BUS ADDRESS
1887  010054  005005                                    CLR     R5              ;SET EXPECTED
1888  010056  117701  171312                            MOVB    @DVSRSH,R1      ;GET SEC REG.
1889  010062  017704  171310                            MOV     @DVSRA,R4       ;READ RESULT
1890  010066  001401                                    BEQ     11$             ;BRANCH IF=0
1891  010070  104003                                    HLT     3               ;OPPOSITE BUS ADDRESS SHOULDN'T BE ALTERED.
1892  010072  105277  171276                  11$:      INCB    @DVSRSH         ;GET OPPOSITE BYTE COUNT
1893  010076  005201                                    INC     R1              ;SET SEC POINTER
1894  010100  017704  171272                            MOV     @DVSRA,R4       ;READ
1895  010104  001401                                    BEQ     12$             ;BRANCH IF=0
1896  010106  104003                                    HLT     3               ;OPPOSITE BYTE CNT S/B=0
1897  010110  012705  000200                  12$:      MOV     #BIT7,R5                ;SET EXPECTED RESULTS
1898  010114  112777  000013  171252                    MOVB    #13.,@DVSRSH    ;SEL LINE STATE
1899  010122  112701  000013                            MOVB    #13,R1          ;SET SEC REG POINTER
1900  010126  017704  171244                            MOV     @DVSRA,R4       ;READ LINE STATE
1901  010132  020504                                    CMP     R5,R4           ;Ok
1902  010134  001401                                    BEQ     13$             ;OK!
```

```
        010136  104003                                HLT     3               ;LINE STATE INCORRECT
        [illegible]                           13$:    MSTCLR                  ;INIT DV11
        [illegible]                                   SCOP1                   ;SW09=1?
        [illegible]                                   INC     R0              ;UPDATE LINE POINTER
        [illegible]                                   DEC     R2              ;4 LINES DONE YET?
        [illegible]                                   BEQ     .+6             ;BR IF YES
        [illegible]     007540                        JMP     1$              ;JMP IF NO
        [illegible]                                   RTS     PC              ;RETURN FOR NEXT GROUP

                                                      ;************************ TEST 3 ********************************
                                                      ;*TEST OF DV11 MICRO PROCESSOR "FREE RUNNING"
                                                      ;*TEST TO XMIT ONE CHAR (ALL 0'S) MAKING SURE THAT
                                                      ;*BIT 15 OF DVSCR =1
                                                      ;*TX WC PRIMARY =0
                                                      ;*TX BA PRIMARY =0
                                                      ;*LINE STATE =0
                                                      ;*TX BA SECONDARY =TXBAS+1
                                                      ;*TX WC SECONDARY =0
                                                      ;*NPR STATUS REG 00:03=LINE SELECTED +BIT9+BIT8=1
                                                      ;*THIS TEST IS EXERCISED FOR BOTH ASYNC AND SYNC LINE CARDS.
                                                      ;;*****************************************************************

                                              ;       TEST 3
                                              ;---------------
010160  012737  000003  001236        TST3:   MOV     #3,TSTNO
010166  012737  010754  001215                MOV     #TST4,NEXT
010174  012700  000000                        MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
010200  013737  001416  001236                MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
010206  100402                                BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
010210  004737  010276                        JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
010214  012700  000004                100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
010220  013737  001420  001236                MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
010226  100402                                BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
010230  004737  010276                        JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
010234  012700  000010                101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
010240  013737  001422  001236                MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
010246  100402                                BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
010250  004737  010276                        JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
010254  012700  000014                102$:   MOV     #12.,R0         ;LOAD LINE NO.
010260  013737  001424  001236                MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
010266  100402                                BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
010270  004737  010276                        JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
010274  104400                        103$:   SCOPE                   ;SCOPE THIS TEST.
010276                                105$:                           ;TEST ENTRANCE.
010276  104413                                RAMCLR                  ;CLEAR ALL DV11 SECONDARY REGISTERS
010300  012705  022350                        MOV     #TXTAB,R5       ;CLEAR
010304  005001                                CLR     R1              ;       TX
010306  105025                        21$:    CLRB    (R5)+           ;       CONTROL
010310  105201                                INCB    R1              ;       TABLE
010312  001375                                BNE     21$             ;
010314  012702  000004                        MOV     #4,R2           ;SET FOR 4 LINE GROUP
010320  012737  010326  001220                MOV     #1$,LOCK        ;SET FOR SW09=1
010326  110077  171040                1$:     MOVB    R0,@DVSRS       ;LOAD LINE NUMBER
010332  004537  021026                        PERFORM SETREG          ;
010336     002     010                        .BYTE   002,010         ;BUS ADDRESS, CONTROL TABLE
```

```
 1959  010340  021750                                  TXBAS                   ;
 1960  010342  022350                                  TXTAB                   ;
 1961  010344  005037  021750                          CLR     TXBAS           ;SET TX DATA TO 0
 1962  010350  004537  021026                          PERFORM ,SETREG         ;
 1963  010354     013     003                          .BYTE   013,003 ;LINE STATE, ALTERNATE BYTE COUNT
 1964  010356  000204                                  BIT7+BIT2               ;LINE STATE CONTENTS
 1965  010360  177777                                  -1                      ;ONE CHAR
 1966  010362  032737  004000  001236                  BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
 1967  010370  001407                                  BEQ     60$             ;#BR IF NO.
 1968  010372  004537  021072                          PERFORM ,LOAD.MODE      ;#
 1969  010376  015000                                  (BIT12+BIT11)+BIT9      ;#8 BITS/PER/CHAR
 1970  010400  004537  021072                          PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
 1971  010404  072000                                  (BIT14+BIT13+BIT12)+BIT10
 1972                                                                          ;#9600 BAUD.
 1973  010406  000403                                  BR      61$             ;#CONTINUE TEST.
 1974  010410  004537  021072          60$:            PERFORM ,LOAD.MODE      ;LOAD MODE
 1975  010414  014000                                  BIT12+BIT11             ;
 1976  010416  005277  170740          61$:            INC     @DVSCR          ;SET MICRO-PROCESSOR GO!
 1977  010422  005005                                  CLR     R5              ;WAIT FOR 15=1
 1978  010424  005777  170732          2$:             TST     @DVSCR          ;DVSCR 15=1?
 1979  010430  100403                                  BMI     3$              ;BR IF YES
 1980  010432  104414                                  DELAY                   ;WASTE TIME
 1981  010434  005205                                  INC     R5              ;DELAY
 1982  010436  001372                                  BNE     2$              ;ALL DONE?
 1983  010440  013701  001362          3$:             MOV     DVSCR,R1        ;SET POINTER
 1984  010444  011104                                  MOV     (R1),R4         ;READ SCR
 1985  010446  012705  100001                          MOV     #BIT15+BIT0,R5  ;SET EXPECTED
 1986  010452  020504                                  CMP     R5,R4           ;RESULTS GOOD?
 1987  010454  001401                                  BEQ     4$              ;BR IF OK
 1988  010456  104002                                  HLT     2               ;DVSCR WRONG
 1989  010460  005777  170702          4$:             TST     @DVRIC          ;MAKE SURE NO RIC ENTRIES
 1990  010464  001405                                  BEQ     5$              ;BR IF OK
 1991  010466  013701  001366                          MOV     DVRIC,R1        ;SET POINTER
 1992  010472  011104                                  MOV     (R1),R4         ;SAVE FOR TYPE OUT
 1993  010474  005005                                  CLR     R5              ;MAKE EXPECTED=0
 1994  010476  104002                                  HLT     2               ;REPORT RIC NOT=0
 1995  010500  010005                          5$:     MOV     R0,R5           ;LOAD LINE NUMBER
 1996  010502  052705  101400                          BIS     #BIT15+BIT9+BIT8,R5     ;SILO ENTRY ALTERNATE BYTE COUNT=3
 1997  010506  012703  000020                          MOV     #16.,R3         ;SET FOR 16 REGISTERS
 1998  010512  013701  001402                          MOV     DVNSR,R1        ;SET POINTER
 1999  010516  011104                                  MOV     (R1),R4         ;READ NSR
 2000  010520  020504                                  CMP     R5,R4           ;OK?
 2001  010522  001401                                  BEQ     6$              ;BR IF OK
 2002  010524  104002                                  HLT     2               ;DVNSR NOT CORRECT
 2003  010526  013701  001362          6$:             MOV     DVSCR,R1        ;SET POINTER
 2004  010532  011104                                  MOV     (R1),R4         ;READ DVSCR (DID BITS CLEAR)
 2005  010534  100005                                  BPL     7$              ;BR IF 15=0
 2006  010536  012705  000001                          MOV     #BIT0,R5        ;SET EXPECTED RESULTS
 2007  010542  104002                                  HLT     2               ;BIT 15 OF DVSCR NOT CLEARED BY READING NSR
 2008  010544  005777  170632                          TST     @DVNSR          ;REFERENCE NSR
 2009  010550  005303                          7$:     DEC     R3              ;ALL REGISTERS DONE?
 2010  010552  001365                                  BNE     6$              ;BR IF NO
 2011  010554  042777  000001  170600                  BIC     #BIT0,@DVSCR    ;CLEAR MICRO-PROCESSOR GO
 2012  010562  012705  021751                          MOV     #TXBA$+1,R5     ;SET EXPECTED
 2013  010566  010077  170600                          MOV     R0,@DVSR$       ;LOAD LINE NUMBER
 2014  010572  112777  000002  170574                  MOVB    #2,@DVSRSH      ;SEL ALTERNATE BUS ADDRESS
```

DO4


```
2015  010600  017704  170572          MOV     @DVSRA,R4       ;READ BUS ADDRESS
2016  010604  117701  170564          MOVB    @DVSRSH,R1      ;GET SEC REG.
2017  010610  020504                  CMP     R5,R4           ;GOOD?
2018  010612  001401                  BEQ     8$              ;BR IF GOOD
2019  010614  104003                  HLT     3               ;BUS ADDRESS NOT IMPEMENTED CORRECTLY
2020  010616  105277  170552    8$:   INCB    @DVSRSH         ;EXAMINE
2021  010622  005201                  INC     R1              ;
2022  010624  017704  170546          MOV     @DVSRA,R4       ;BYTE COUNT REGISTER
2023  010630  001402                  BEQ     10$             ;IS IT=0?
2024  010632  005005                  CLR     R5              ;SET EXPECTED
2025  010634  104003                  HLT     3               ;BYTE COUNT NOT=0!
2026  010636  010077  170530    10$:  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2027  010642  112777  000000  170524  MOVB    #000,@DVSRSH    ;GET OPPOSITE BUS ADDRESS
2028  010650  005005                  CLR     R5              ;SET EXPECTED
2029  010652  117701  170516          MOVB    @DVSRSH,R1      ;GET SEC REG.
2030  010656  017704  170514          MOV     @DVSRA,R4       ;READ RESULT
2031  010662  001401                  BEQ     11$             ;BRANCH IF=0
2032  010664  104003                  HLT     3               ;OPPOSITE BUS ADDRESS SHOULDN'T BE ALTERED.
2033  010666  105277  170502    11$:  INCB    @DVSRSH         ;GET OPPOSITE BYTE COUNT
2034  010672  005201                  INC     R1              ;SET SEC POINTER
2035  010674  017704  170476          MOV     @DVSRA,R4       ;READ
2036  010700  001401                  BEQ     12$             ;BRANCH IF=0
2037  010702  104003                  HLT     3               ;OPPOSITE BYTE CNT S.B=0
2038  010704  012705  000000    12$:  MOV     #0,R5           ;SET EXPECTED RESULTS
2039  010710  112777  000013  170456  MOVB    #13,@DVSRSH     ;SEL LINE STATE
2040  010716  112701  000013          MOVB    #13,R1          ;SET SEC REG POINTER
2041  010722  017704  170450          MOV     @DVSRA,R4       ;READ LINE STATE
2042  010726  020504                  CMP     R5,R4           ;OK
2043  010730  001401                  BEQ     13$             ;OK!
2044  010732  104003                  HLT     3               ;LINE STATE INCORRECT
2045  010734  104412    13$:  MSTCLR                          ;INIT DV11
2046  010736  104401                  SCOPI                   ;SW09=1?
2047  010740  005200                  INC     R0              ;UPDATE LINE POINTER
2048  010742  005302                  DEC     R2              ;4 LINES DONE YET?
2049  010744  001402                  BEQ     .+6             ;BR IF YES
2050  010746  000137  010326          JMP     1$              ;JMP IF NO
2051  010752  000207                  RTS     PC              ;RETURN FOR NEXT GROUP
2052
2053
2054                                  ;************************ TEST 4 ******************************
2055                                  ;*TEST OF TRANSMITTER NON-EXISTANT MEMORY FLAGING.
2056                                  ;*THIS TEST VERIFIES THAT THE MICO-PROCESSOR
2057                                  ;*CAN FLAG A NON-EXISTANT TX BUS ADDRESS.
2058                                  ;*EXPECTED:     (THIS IS FOR LINES  )
2059                                  ;*DVSCR         BIT15+BIT0
2060                                  ;*DVNSR         03:00=LINE  11:08=0  BIT15=1
2061                                  ;*TXBAP         177320
2062                                  ;*TXWCP         177777
2063                                  ;*DVLINE STATE  BIT4=1
2064                                  ;*THIS TEST IS DONE FOR SYNC AND ASYNC LINE CARDS.
2065                                  ;;**********************************************************
2066
2067                                  ;  TEST 4
2068                                  ;--------------
2069  010754  012737  000004  001226  TST4:  MOV     #4,TSTNO
2070  010762  012737  011434  001216          MOV     #TST5,NEXT
```

E04


```
002000  012700  000000          MOV     #0,,R0          ;PLACE LINE NUMBER INTO R0
002004  013737  001416  001236  MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
002012  100402                  BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
002014  004737  011072          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
002020  012700  000004  100$:   MOV     #4,,R0          ;PLACE LINE NUMBER INTO R0
002024  013737  001420  001236  MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
002032  100402                  BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
002034  004737  011072          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
002040  012700  000010  101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
002044  013737  001422  001236  MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
002052  100402                  BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
002054  004737  011072          JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
002060  012700  000014  102$:   MOV     #12.,R0         ;LOAD LINE NO.
002064  013737  001424  001236  MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
002072  100402                  BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
002074  004737  011072          JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
002100  104400          103$:   SCOPE                   ;SCOPE THIS TEST.
002072                  105$:                           ;TEST ENTRANCE.
002072  104413                  RAMCLR                  ;CLEAR ALL SEC. REGISTERS
002074  012737  011106  001230  MOV     #1$,LOCK        ;SET IF SW09=1
002102  012702  000004          MOV     #4,R2           ;SET FOR A 4 LINE GROUP
002106  010077  170260  1$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
002112  052777  000060  170242  BIS     #BIT5+BIT4,@DVSCR
002120  004537  021026          PERFORM ,SETREG         ;SET EA BITS
002124     000     010          .BYTE   000,010         ;GO LOAD BUS ADDRESS AND CNTL TABLE
002126  177320                  177320                  ;THIS IS ADDRESS OF 2ND KE11
002130  022350                  TXTAB                   ;
002132  042777  000060  170222  BIC     #BIT5+BIT4,@DVSCR
002140  004537  021026          PERFORM ,SETREG         ;CLEAR EA BITS FOR CERTAIN
002144     013     001          .BYTE   013,001         ;LOAD LINE STATE AND PRINCIPLE BYTE CNT.
002146  000004                  BIT2                    ;TX GO
002150  177777                  -1                      ;ONE CHAR
002152  032737  004000  001236  BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
002160  001407                  BEQ     60$             ;#BR IF NO.
002162  004537  021072          PERFORM ,LOAD.MODE      ;#
002166  015000                  (BIT12+BIT11)+BIT9      ;#8 BITS/PER/CHAR
002170  004537  021072          PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
002174  072000                  (BIT14+BIT13+BIT12)+BIT10
                                                        ;#9600 BAUD.
002176  000403                  BR      61$             ;#CONTINUE TEST.
002200  004537  021072  60$:    PERFORM ,LOAD.MODE      ;LOAD
002204  014000                  BIT12+BIT11             ;MODE
002206  005277  170150  61$:    INC     @DVSCR          ;SET MICRO PROCESSOR GO
002212  005005                  CLR     R5              ;PREPARE TIMER
002214  005777  170142  2$:     TST     @DVSCR          ;SILO ENTRY?
002220  100403                  BMI     3$              ;BR IF DONE
002222  104414                  DELAY                   ;WASTE TIME
002224  005205                  INC     R5              ;DELAY
002226  001372                  BNE     2$              ;
002230  013701  001362  3$:     MOV     DVSCR,R1        ;SET POINTER
002234  011104                  MOV     (R1),R4         ;READ SCR
002236  012705  100001          MOV     #BIT15+BIT0,R5  ;SET EXPECTED RESULTS
002242  020504                  CMP     R5,R4           ;DVSCR OK?
002244  001401                  BEQ     4$              ;
002246  104002                  HLT     2               ;DVSCR INCORRECT
002250  042777  000000  170104  4$:     BIC     #BIT0,@DVSCR    ;CLEAR MICRO CPU GO
```

```
 2127  011256  010005                        MOV     R0,R5           ;LOAD LINE NO.
 2128  011260  052705  100000                BIS     #BIT15,R5       ;SET SILO ENTRY
 2129  011264  013701  001432                MOV     DVNSR,R1        ;SET POINTER
 2130  011270  011104                        MOV     (R1),R4         ;READ NSR
 2131  011272  020504                        CMP     R5,R4           ;NSR OK?
 2132  011274  001401                        BEQ     5$              ;
 2133  011276  104002                        HLT     2               ;DVNSR INCORRECT
 2134  011300  005005                5$:     CLR     R5              ;SET EXPCTED RESULTS TO 0
 2135  011302  013701  001362                MOV     DVSCR,R1        ;SET POINTER
 2136  011306  011104                        MOV     (R1),R4         ;READ DVSCR
 2137  011310  001401                        BEQ     6$              ;IS IT=0?
 2138  011312  104002                        HLT     2               ;MOST LIKELY BIT IS IS NOT=0
 2139  011314  010077  170052        6$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
 2140  011320  017704  170052                MOV     @DVSRA,R4       ;READ TX BUS ADDRESS PRINCIP_E
 2141  011324  012705  177320                MOV     #177320,R5      ;LOAD EXPECTED
 2142  011330  005001                        CLR     R1              ;SET SEC REG POINTER
 2143  011332  020504                        CMP     R5,R4           ;DID IT INCREMENT?
 2144  011334  001401                        BEQ     7$              ;BR IF NO
 2145  011336  104003                        HLT     3               ;BUS ADDRESS INCORRECT
 2146  011340  012705  177777        7$:     MOV     #-1,R5          ;SET EXPECTED BYTE COUNT
 2147  011344  105277  170024                INCB    @DVSRSH         ;SEL PRINCIPLE BYTE COUN~
 2148  011350  005201                        INC     R1              ;SET SEC REG POINTER
 2149  011352  017704  170020                MOV     @DVSRA,R4       ;READ IT
 2150  011356  020504                        CMP     R5,R4           ;DID IT GO TO ZERO?
 2151  011360  001401                        BEQ     9$              ;BR IF NO
 2152  011362  104003                        HLT     3               ;BYTE CNT S/B=-1
 2153  011364  012705  000020        9$:     MOV     #BIT4,R5        ;LINE STATE S/B TX NXM
 2154  011370  012701  000013                MOV     #13,R1          ;SET SEC REG POINTER
 2155  011374  112777  000013  167772        MOVB    #13,@DVSRSH     ;SEL LINE STATE
 2156  011402  017704  167770                MOV     @DVSRA,R4       ;READ IT
 2157  011406  020504                        CMP     R5,R4           ;OK
 2158  011410  001401                        BEQ     10$             ;
 2159  011412  104003                        HLT     3               ;TX NUM S/B ONLY THING SET IN _INE STATE
 2160  011414  104412                10$:    MSTCLR                  ;INIT DV11
 2161  011416  104401                        SCOPI                   ;LOCK ON LINE?
 2162  011420  005200                        INC     R0              ;UPDATE LINE POINTER
 2163  011422  005302                        DEC     R2              ;4 LINE GROUP DONE
 2164  011424  001402                        BEQ     .+6             ;BR IF YES
 2165  011426  000137  011106                JMP     1$              ;JMP IF NO
 2166  011432  000207                        RTS     PC              ;EXIT FOR NEXT GROUP
 2167
 2168
 2169                                        ;************************ TEST 5 *******************************
 2170                                        ;*TEST TO FORCE ALL POSSIBLE
 2171                                        ;*TRANSMITTER NXM ERRORS PAYING
 2172                                        ;*ATTENTION TO THE DVNSR ENTRY.
 2173                                        ;*   EXERCISED          EXPECTED DVNSR  (BIT15 ALWAYS=1)
 2174                                        ;*      TXBAP           11:08=0  03:00=LINE NO.
 2175                                        ;*      TXBAS           BIT9=1   03:00=LINE NO.
 2176                                        ;*      TXTAB           BIT11=1  03:00=LINE NO.
 2177                                        ;*THIS TEST IS DONE FOR SYNC AND ASYNC _INE CARDS.
 2178                                        ;;***************************************************************
 2179
 2180                                        ;  TEST 5
 2181                                        ;--------------
 2182  011434  012737  000005  001226        TST5:   MOV     #5,TSTNO
```

```
 2193  011442  012737  012244  001216                MOV     #TST6,NEXT
 2194  011450  012700  000000                        MOV     #0,R0           ;PLACE LINE NUMBER INTO R0
 2195  011454  013737  001416  001236                MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2196  011462  100402                                BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 2197  011464  004737  011552                        JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 2198  011470  012700  000004              100$:     MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 2199  011474  013737  001420  001236                MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2200  011502  100402                                BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 2201  011504  004737  011552                        JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 2202  011510  012700  000010              101$:     MOV     #8.,R0          ;LOAD LINE NUMBER
 2203  011514  013737  001422  001236                MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2204  011522  100402                                BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 2205  011524  004737  011552                        JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 2206  011530  012700  000014              102$:     MOV     #12.,R0         ;LOAD LINE NO.
 2207  011534  013737  001424  001236                MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 2208  011542  100402                                BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 2209  011544  004737  011552                        JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 2210  011550  104400                      103$:     SCOPE                   ;SCOPE THIS TEST.
 2211  011552                               105$:                             ;TEST ENTRANCE.
 2212  011552  104413                                RAMCLR                  ;CLEAR ALL SEC REGISTERS
 2213  011554  012737  011566  001220                MOV     #1$,LOCK        ;SET FOR RETURN IF SW09=1
 2214  011562  012702  000004                        MOV     #4,R2           ;SET FOR 4 LINE GROUP
 2215  011566  104412                        1$:     MSTCLR                  ;INIT DV11
 2216  011570  010077  167576                        MOV     R0,@DVSRS       ;LOAD LINE NUMBER
 2217  011574  052777  000060  167560                BIS     #BIT5+BIT4,@DVSCR
 2218  011602  004537  021026                        PERFORM ,SETREG         ;SET EA BITS FOR NXM
 2219  011606     000     010                        .BYTE   000,010         ;PRINCIPLE BA, CNTRL TABLE
 2220  011610  177320                                177320                  ;NXM (2ND KE11)
 2221  011612  022350                                TXTAB                   ;TXTABLE
 2222  011614  004537  021026                        PERFORM ,SETREG         ;
 2223  011620     013     001                        .BYTE   013,001         ;LINE STATE, PRINCIPLE BYTE CNT
 2224  011622  000004                                BIT2                    ;TX GO
 2225  011624  177777                                -1                      ;ONE CHAR
 2226  011626  032737  004000  001236                BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
 2227  011634  001407                                BEQ     60$             ;#BR IF NO.
 2228  011636  004537  021072                        PERFORM ,LOAD.MODE      ;#
 2229  011642  015000                                (BIT12+BIT11)+BIT9      ;#8 BITS/PER/CHAR
 2230  011644  004537  021072                        PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
 2231  011650  072000                                (BIT14+BIT13+BIT12)+BIT10
 2232                                                                        ;#9600 BAUD.
 2233  011652  000400                                BR      60$             ;#CONTINUE TEST.
 2234  011654  005277  167502              60$:      INC     @DVSCR          ;SET MICRO CPU GO
 2235  011660  005005                                CLR     R5              ;WAIT FOR
 2236  011662  005777  167474              2$:       TST     @DVSCR          ;SILO ENTRY
 2237  011666  100404                                BMI     3$              ;BR IF SET
 2238  011670  104414                                DELAY                   ;WASTE TIME
 2239  011672  005205                                INC     R5              ;DELAY
 2240  011674  001372                                BNE     2$              ;BR IF NOT THROUGH
 2241  011676  104000                                HLT     0               ;NO SILO ENTRY
 2242  011700  013701  001402              3$:       MOV     DVNSR,R1        ;SET POINTER
 2243  011704  011104                                MOV     (R1),R4         ;READ NSR
 2244  011706  010005                                MOV     R0,R5           ;LOAD LINE NO.
 2245  011710  052705  100000                        BIS     #BIT15,R5       ;SET SILO ENTRY
 2246  011714  020504                                CMP     R5,R4           ;OK?
 2247  011716  001401                                BEQ     4$              ;BR IF OK
 2248  011720  104002                                HLT     2               ;TX PRINCIPLE NXM NOT IN NSR
```

```
011722  104401                      4$:   SCOP1                       ;LOCK?
011724  012737  011732  001220            MOV     #5$,LOCK            ;PREPARE LOOP
011732  104412                      5$:   MSTCLR                      ;INIT DV11
011734  010077  167432                    MOV     R0,@DVSRS           ;LOAD LINE NO.
011740  052777  000060  167414            BIS     #BIT5+BIT4,@DVSCR
011746  004537  021026                    PERFORM ,SETREG             ;SET EA BITS
011752      002     010                    .BYTE   002,010             ;ALTERNATE BA, CNTRL TABLE
011754  177320                          177320                        ;NXM (2ND KE11)
011756  022350                          TXTAB                         ;
011760  004537  021026                    PERFORM ,SETREG             ;
011764      013     003                    .BYTE   013,003             ;LINE STATE, ALTERNATE BYTE CNT
011766  000204                          BIT7+BIT2                     ;USE ALTERNATE TABLE, TXGO
011770  177777                          -1                            ;ONE CHAR
011772  032737  004000  001236            BIT     #ASYNC,STAT         ;#IS THIS ASYNC LINE CARD?
012000  001407                          BEQ     61$                   ;#BR IF NO.
012002  004537  021072                    PERFORM ,LOAD.MODE          ;#
012006  015000                          (BIT12+BIT11)+BIT9            ;#8 BITS/PER/CHAR
012010  004537  021072                    PERFORM ,LOAD.MODE          ;#SET BAUD RATE REGISTER.
012014  072000                          (BIT14+BIT13+BIT12)+BIT10
                                                                      ;#9600 BAUD.
012016  000400                          BR      61$                   ;#CONTINUE TEST.
012020  005277  167336          61$:    INC     @DVSCR                ;SET MICRO CPU GO
012024  005005                          CLR     R5                    ;DELAY
012026  005777  167330          6$:     TST     @DVSCR                ;WAITING
012032  100404                          BMI     7$                    ;FOR
012034  104414                          DELAY                         ;
012036  005205                          INC     R5                    ;SILO
012040  001372                          BNE     6$                    ;ENTRY
012042  104000                          HLT     0                     ;NO SILO ENTRY
012044  017704  167332          7$:     MOV     @DVNSR,R4             ;READ NSR
012050  010005                          MOV     R0,R5                 ;LOAD LINE NO
012052  052705  101000                    BIS     #BIT15+BIT9,R5        ;SET SILO ENTRY, ALTERNATE NXM
012056  020504                          CMP     R5,R4                 ;NSR OK?
012060  001401                          BEQ     8$                    ;
012062  104002                          HLT     2                     ;ALTERNATE BA NXM FAILED
012064  104401                    8$:   SCOP1                         ;SW09=1?
012066  012737  012074  001220            MOV     #9$,LOCK            ;SET SCOPE LOOP
012074  104412                      9$:   MSTCLR                      ;INIT DV11
012076  010077  167270                    MOV     R0,@DVSRS           ;LOAD LINE NUMBER
012102  052777  000060  167252            BIS     #BIT5+BIT4,@DVSCR
012110  004537  021026                    PERFORM ,SETREG             ;SET EA BITS
012114      010     000                    .BYTE   010,000             ;TX CNTRL TABLE, PRINCIPLE BA
012116  177320                          177320                        ;NXM (2ND KE11)
012120  021350                          TXBAP                         ;BA
012122  004537  021026                    PERFORM ,SETREG             ;
012126      013     001                    .BYTE   013,001             ;LINE STATE, PRINCIPLE BC
012130  000004                          BIT2                          ;TX GO
012132  177777                          -1                            ;ONE XFR
012134  032737  004000  001236            BIT     #ASYNC,STAT         ;#IS THIS ASYNC LINE CARD?
012142  001407                          BEQ     62$                   ;#BR IF NO.
012144  004537  021072                    PERFORM ,LOAD.MODE          ;#
012150  015000                          (BIT12+BIT11)+BIT9            ;#8 BITS/PER/CHAR
012152  004537  021072                    PERFORM ,LOAD.MODE          ;#SET BAUD RATE REGISTER.
012156  072000                          (BIT14+BIT13+BIT12)+BIT10
                                                                      ;#9600 BAUD.
012160  000400                          BR      62$                   ;#CONTINUE TEST.
```

```
2295  012162  005277  167174          62$:    INC     @DVSCR          ;SET MICRO CPU GO
2296  012166  005005                          CLR     R5              ;DELAY
2297  012170  005777  167166          10$:    TST     @DVSCR          ;FOR
2298  012174  100404                          BMI     11$             ;SILO
2299  012176  104414                          DELAY                   ;
2300  012200  005205                          INC     R5              ;ENTRY
2301  012202  001372                          BNE     10$             ;
2302  012204  104000                          HLT     0               ;NO SILO ENTRY
2303  012206  017704  167170          11$:    MOV     @DVNSR,R4       ;READ NSR
2304  012212  010005                          MOV     R0,R5           ;LOAD LINE NUMBER
2305  012214  052705  104000                  BIS     #BIT15+BIT11,R5 ;SET SILO ENTRY, CNTRL TABLE NXM
2306  012220  020504                          CMP     R5,R4           ;NSR OK?
2307  012222  001401                          BEQ     12$             ;YES
2308  012224  104002                          HLT     2               ;CNTRL TABLE NXM FAILED
2309  012226  104401          12$:    SCOPE                   ;SW09=1
2310  012230  005200                          INC     R0              ;UPDATE LINE NO.
2311  012232  005302                          DEC     R2              ;4 LINE GROUP DONE
2312  012234  001402                          BEQ     .+6             ;BR IF YES
2313  012236  000137  011566                  JMP     1$              ;JMP IF NO.
2314  012242  000207                          RTS     PC              ;EXIT FOR NEXT GROUP OF LINES
2315
2316                                  ;*********************** TEST 6 ****************************
2317                                  ;*TEST OF TRANSMITTER BCC OPERATIONS
2318                                  ;*TEST THAT THE CHAR "25" WILL
2319                                  ;*BE INCLUDED INTO THE BCC
2320                                  ;*THE POLY USED WILL BE LRC8
2321                                  ;*THIS TEST IS EXERCISED FOR BOTH ASYNC AND SYNC LINE CARDS.
2322                                  ;;***********************************************************
2323
2324                          ;  TEST 6
2325                          ;-------------
2326
2327  012244  012737  000006  001226  TST6:   MOV     #6,TSTNO
2328  012252  012737  012610  001216          MOV     #TST7,NEXT
2329  012260  012700  000000                  MOV     #0,,R0          ;PLACE LINE NUMBER INTO RC
2330  012264  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2331  012272  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2332  012274  004737  012362                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2333  012300  012700  000004          100$:   MOV     #4,,R0          ;PLACE LINE NUMBER INTO R0
2334  012304  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2335  012312  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2336  012314  004737  012362                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2337  012320  012700  000010          101$:   MOV     #8,,R0          ;LOAD LINE NUMBER
2338  012324  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2339  012332  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2340  012334  004737  012362                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2341  012340  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
2342  012344  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2343  012352  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2344  012354  004737  012362                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2345  012360  104400          103$:   SCOPE                   ;SCOPE THIS TEST.
2346  012362                  105$:                           ;TEST ENTRANCE.
2347  012362  104413                          RAMCLR                  ;CLEAR ALL SEC REGISTERS
2348  012364  012737  000200  021020          MOV     #LRC8,XPOLY     ;SET SOFTWARE POLYNOMINAL
2349  012372  012705  000025                  MOV     #25,R5          ;SET DATA CHAR
2350  012376  110537  021350                  MOVB    R5,@XBAP        ;LOAD DATA
```

```
 2351  012402  112765  000010  022350                MOVB    #BIT3,TXTAB(R5) ;SET CNTRL BYTE
 2352  012410  012702  000004                        MOV     #4,R2           ;SET FOR 4 LINE GROUP
 2353  012414  010077  166752              1$:       MOV     R0,@DVSRS       ;LOAD LINE NO.
 2354  012420  004537  021026                        PERFORM ,SETREG         ;
 2355  012424     000     001                        .BYTE   000,001         ;PRINCIPLE BA, BC
 2356  012426  021350                                TXBAP                   ;
 2357  012430  177777                                -1                      ;
 2358  012432  004537  021026                        PERFORM ,SETREG         ;
 2359  012436     013     010                        .BYTE   013,010         ;LINE STATE, CNTRL TABLE
 2360  012440  000004                                BIT2                    ;TXGO
 2361  012442  022350                                TXTAB                   ;TABLE
 2362  012444  004537  021026                        PERFORM ,SETREG         ;
 2363  012450     006     012                        .BYTE   006,012         ;TX BCC REG, LINE PROTOCOL
 2364  012452  000037                                37                      ;BCC
 2365  012454  000000                                0                       ;POLYNOMINAL SELECT
 2366  012456  032737  004000  001236                BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
 2367  012464  001407                                BEQ     60$             ;#BR IF NO.
 2368  012466  004537  021072                        PERFORM ,LOAD.MODE      ;#
 2369  012472  015000                                (BIT12+BIT11)+BIT9      ;#8 BITS/PER/CHAR
 2370  012474  004537  021072                        PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
 2371  012500  072000                                (BIT14+BIT13+BIT12)+BIT10
 2372                                                                        ;#9600 BAUD.
 2373  012502  000403                                BR      61$             ;#CONTINUE TEST.
 2374  012504  004537  021072              60$:      PERFORM ,LOAD.MODE      ;LOAD
 2375  012510  014000                                BIT12+BIT11             ;MODE
 2376  012512  005277  166644              61$:      INC     @DVSCR          ;SET MICRO CODE GO
 2377  012516  005005                                CLR     R5              ;DELAY
 2378  012520  005777  166636              2$:       TST     @DVSCR          ;FOR
 2379  012524  100403                                BMI     3$              ;SILO
 2380  012526  104414                                DELAY                   ;WASTE TIME
 2381  012530  005205                                INC     R5              ;ENTRY
 2382  012532  001372                                BNE     2$              ;
 2383  012534  112777  000006  166632        3$:     MOVB    #6,@DVSRSH      ;SEL BCC REGISTER
 2384  012542  017704  166630                        MOV     @DVSRA,R4       ;READ DVSRA [BCC REG]
 2385  012546  004537  020646                        JSR     R5,SIMBCC       ;GO GET SOFTWARE BCC RESULT
 2386  012552  000010                                8.                      ;SHIFTS
 2387  012554  000025                                25                      ;DATA
 2388  012556  000037                                37                      ;PREVIOUS BCC RESULTS
 2389  012560  013705  021024                        MOV     CALBCC,R5       ;READ SOFTWARE BCC RESULTS
 2390  012564  117701  166604                        MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
 2391  012570  020504                                CMP     R5,R4           ;SOFTWARE=HRDWARE?
 2392  012572  001401                                BEQ     4$              ;BR IF YES
 2393  012574  104003                                HLT     3               ;HRDWARE BCC WRONG
 2394  012576  104412                        4$:     MSTCLR                  ;INIT DV11
 2395  012600  005200                                INC     R0              ;UPDATE LINE NO.
 2396  012602  005302                                DEC     R2              ;4 LINE GROUP DONE?
 2397  012604  001303                                BNE     1$              ;BR IF NO
 2398  012606  000207                                RTS     PC              ;EXIT FOR NEXT 4 LINE GROUP
 2399
 2400
 2401                                                ;************************ TEST 7 *******************************
 2402                                                ;*TEST OF TRANSMITTER BCC OPERATIONS
 2403                                                ;*TEST THAT THE CHAR "25" WILL
 2404                                                ;*BE INCLUDED INTO THE BCC
 2405                                                ;*THE POLY USED WILL BE CRC16
 2406                                                ;*THIS TEST IS EXERCISED FOR BOTH ASYNC AND SYNC LINE CARDS.
```

```
2407                                                    ;;*****************************************************************************
2408
2409                                                    ;  TEST 7
2410                                                    ;--------------
2411  012610  012737  000007  001226        TST7:   MOV     #7,TSTNO
2412  012616  012737  013154  001216                MOV     #TST7.0,NEXT
2413  012624  012700  000000                        MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2414  012630  013737  001416  001236                MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2415  012636  100402                                BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2416  012640  004737  012726                        JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2417  012644  012700  000004        100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2418  012650  013737  001420  001236                MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2419  012656  100402                                BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2420  012660  004737  012726                        JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2421  012664  012700  000010        101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2422  012670  013737  001422  001236                MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2423  012676  100402                                BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2424  012700  004737  012726                        JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2425  012704  012700  000014        102$:   MOV     #12.,R0         ;LOAD LINE NO.
2426  012710  013737  001424  001236                MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2427  012716  100402                                BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2428  012720  004737  012726                        JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2429  012724  104400                103$:   SCOPE                   ;SCOPE THIS TEST.
2430  012726                        105$:                           ;TEST ENTRANCE.
2431  012726  104413                                RAMCLR                  ;CLEAR ALL SEC REGISTERS
2432  012730  012737  120001  021020                MOV     #CRC16,XPOLY    ;SET SOFTWARE POLYNOMINAL
2433  012736  012705  000025                        MOV     #25,R5          ;SET DATA CHAR
2434  012742  110537  021350                        MOVB    R5,TXBAP        ;LOAD DATA
2435  012746  112765  000010  022350                MOVB    #BIT3,TXTAB(R5) ;SET CNTRL BYTE
2436  012754  012702  000004                        MOV     #4,R2           ;SET FOR 4 LINE GROUP
2437  012760  010077  166406        1$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
2438  012764  004537  021026                        PERFORM ,SETREG         ;
2439  012770     000     001                        .BYTE   000,001         ;PRINCIPLE BA, BC
2440  012772  021350                                TXBAP                   ;
2441  012774  177777                                -1                      ;
2442  012776  004537  021C26                        PERFORM ,SETREG         ;
2443  013002     013     010                        .BYTE   013,010         ;LINE STATE, CNTRL TABLE
2444  013004  000004                                BIT2                    ;TXGO
2445  013006  022350                                TXTAB                   ;TABLE
2446  013010  004537  021026                        PERFORM ,SETREG         ;
2447  013014     006     012                        .BYTE   006,012         ;TX BCC REG, LINE PROTOCOL
2448  013016  000037                                37                      ;BCC
2449  013020  000010                                BIT3                    ;POLYNOMINAL SELECT
2450  013022  032737  004000  001236                BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
2451  013030  001407                                BEQ     60$             ;#BR IF NO.
2452  013032  004537  021072                        PERFORM ,LOAD.MODE      ;#
2453  013036  015000                                (BIT12+BIT11)+BIT9      ;#8 BITS/PER/CHAR
2454  013040  004537  021072                        PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
2455  013044  072000                                (BIT14+BIT13+BIT12)+BIT10
2456                                                                        ;#9600 BAUD.
2457  013046  000403                                BR      61$             ;#CONTINUE TEST.
2458  013050  004537  021072        60$:    PERFORM ,LOAD.MODE      ;LOAD
2459  013054  014000                                BIT12+BIT11             ;MODE
2460  013056  005277  166300        61$:    INC     @DVSCR          ;SET MICRO CODE GO
2461  013062  005005                                CLR     R5              ;DELAY
2462  013064  005777  166272        2$:     TST     @DVSCR          ;FOR
```

```
2463  013070  100403                                  BMI     3$              ;SILO
2464  013072  104414                                  DELAY                   ;WASTE TIME
2465  013074  005205                                  INC     R5              ;ENTRY
2466  013076  001372                                  BNE     2$              ;
2467  013100  112777  000006  166265          3$:     MOVB    #6,@DVSRSH      ;SEL BCC REGISTER
2468  013106  017704  166264                          MOV     @DVSRA,R4       ;READ DVSRA (BCC REG)
2469  013112  004537  020646                          JSR     R5,SIMBCC       ;GO GET SOFTWARE BCC RESULT
2470  013116  000010                                  8.                      ;SHIFTS
2471  013120  000025                                  25                      ;DATA
2472  013122  000037                                  37                      ;PREVIOUS BCC RESULTS
2473  013124  013705  021024                          MOV     CALBCC,R5       ;READ SOFTWARE BCC RESULTS
2474  013130  117701  166240                          MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
2475  013134  020504                                  CMP     R5,R4           ;SOFTWARE=HRDWARE?
2476  013136  001401                                  BEQ     4$              ;BR IF YES
2477  013140  104003                                  HLT     3               ;HRDWARE BCC WRONG
2478  013142  104412                          4$:     MSTCLR                  ;INIT DV11
2479  013144  005200                                  INC     R0              ;UPDATE LINE NO.
2480  013146  005302                                  DEC     R2              ;4 LINE GROUP DONE?
2481  013150  001303                                  BNE     1$              ;BR IF NO
2482  013152  000207                                  RTS     PC              ;EXIT FOR NEXT 4 LINE GROUP
2483
2484
2485                                                  ;********************** TEST 10 ******************************
2486                                                  ;*TEST OF TRANSMITTER BCC OPERATIONS
2487                                                  ;*TEST THAT THE CHAR "25" WILL
2488                                                  ;*BE INCLUDED INTO THE BCC
2489                                                  ;*THE POLY USED WILL BE CRC.CCITT
2490                                                  ;*THIS TEST IS EXERCISED FOR BOTH ASYNC AND SYNC LINE CARDS.
2491                                                  ;**************************************************************
2492
2493                                          ;  TEST 10
2494                                          ;--------------
2495  013154  012737  000010  001226          TST10:  MOV     #10,TSTNO
2496  013162  012737  013520  001216                  MOV     #TST11,NEXT
2497  013170  012700  000000                          MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2498  013174  013737  001416  001236                  MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2499  013202  100402                                  BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2500  013204  004737  013272                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2501  013210  012700  000004                  100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2502  013214  013737  001420  001236                  MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2503  013222  100402                                  BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2504  013224  004737  013272                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2505  013230  012700  000010                  101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2506  013234  013737  001422  001236                  MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2507  013242  100402                                  BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2508  013244  004737  013272                          JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2509  013250  012700  000014                  102$:   MOV     #12.,R0         ;LOAD LINE NO.
2510  013254  013737  001424  001236                  MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2511  013262  100402                                  BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2512  013264  004737  013272                          JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2513  013270  104400                          103$:   SCOPE                   ;SCOPE THIS TEST.
2514  013272                                  105$:                           ;TEST ENTRANCE.
2515  013272  104413                                  RAMCLR                  ;CLEAR ALL SEC REGISTERS
2516  013274  012737  102010  021020                  MOV     #CRC.CCITT,XPOLY        ;SET SOFTWARE POLYNOMINAL
2517  013302  012705  000025                          MOV     #25,R5          ;SET DATA CHAR
2518  013306  110537  021350                          MOVB    R5,TXBAP        ;LOAD DATA
```

```
2519  013312  112765  000010  022350                  MOVB    #BIT3,TXTAB(R5) ;SET CNTRL BYTE
2520  013320  012702  000004                          MOV     #4,R2           ;SET FOR 4 LINE GROUP
2521  013324  010077  166042                  1$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
2522  013330  004537  021026                          PERFORM ,SETREG         ;
2523  013334     000     001                          .BYTE   000,001         ;PRINCIPLE BA, BC
2524  013336  021350                                  TXBAP                   ;
2525  013340  177777                                  -1                      ;
2526  013342  004537  021026                          PERFORM ,SETREG         ;
2527  013346     013     010                          .BYTE   013,010         ;LINE STATE, CNTRL TABLE
2528  013350  000004                                  BIT2                    ;TXGO
2529  013352  022350                                  TXTAB                   ;TABLE
2530  013354  004537  021026                          PERFORM ,SETREG         ;
2531  013360     006     012                          .BYTE   006,012         ;TX BCC REG, LINE PROTOCOL
2532  013362  000037                                  37                      ;BCC
2533  013364  000030                                  BIT4+BIT3                       ;POLYNOMINAL SELECT
2534  013366  032737  004000  001236                  BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
2535  013374  001407                                  BEQ     60$             ;#BR IF NO.
2536  013376  004537  021072                          PERFORM ,LOAD.MODE      ;#
2537  013402  015000                                  (BIT12+BIT11)+BIT9      ;#8 BITS/PER/CHAR
2538  013404  004537  021072                          PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
2539  013410  072000                                  (BIT14+BIT13+BIT12)+BIT10
2540                                                                          ;#9600 BAUD.
2541  013412  000403                                  BR      61$             ;#CONTINUE TEST.
2542  013414  004537  021072                  60$:    PERFORM ,LOAD.MODE      ;LOAD
2543  013420  014000                                  BIT12+BIT11             ;MODE
2544  013422  005277  165734                  61$:    INC     @DVSCR          ;SET MICRO CODE GO
2545  013426  005005                                  CLR     R5              ;DELAY
2546  013430  005777  165726                  2$:     TST     @DVSCR          ;FOR
2547  013434  100403                                  BMI     3$              ;SILO
2548  013436  104414                                  DELAY                   ;WASTE TIME
2549  013440  005205                                  INC     R5              ;ENTRY
2550  013442  001372                                  BNE     2$              ;
2551  013444  112777  000006  165722          3$:     MOVB    #6,@DVSRSH      ;SEL BCC REGISTER
2552  013452  017704  165720                          MOV     @DVSRA,R4       ;READ DVSRA [BCC REG]
2553  013456  004537  020646                          JSR     R5,SIMBCC       ;GO GET SOFTWARE BCC RESULT
2554  013462  000010                                  8.                      ;SHIFTS
2555  013464  000025                                  25                      ;DATA
2556  013466  000037                                  37                      ;PREVIOUS BCC RESULTS
2557  013470  013705  021024                          MOV     CALBCC,R5       ;READ SOFTWARE BCC RESULTS
2558  013474  117701  165674                          MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
2559  013500  020504                                  CMP     R5,R4           ;SOFTWARE=HRDWARE?
2560  013502  001401                                  BEQ     4$              ;BR IF YES
2561  013504  104003                                  HLT     3               ;HRDWARE BCC WRONG
2562  013506  104412                          4$:     MSTCLR                  ;INIT DV11
2563  013510  005200                                  INC     R0              ;UPDATE LINE NO.
2564  013512  005302                                  DEC     R2              ;4 LINE GROUP DONE?
2565  013514  001303                                  BNE     1$              ;BR IF NO
2566  013516  000207                                  RTS     PC              ;EXIT FOR NEXT 4 LINE GROUP
2567
2568
2569                                                  ;************************ TEST 11 *******************************
2570                                                  ;*TEST OF TRANSMITTER BCC OPERATIONS
2571                                                  ;*TEST THAT THE CHAR "25" WILL
2572                                                  ;*WILL SEND THE BCC.
2573                                                  ;*THE POLY USED WILL BE CRC.CCITT
2574                                                  ;*THIS TEST IS EXERCISED FOR BOTH ASYNC AND SYNC LINE CARDS.
```

```
2575                                                    ;:**********************************************************************
2576
2577                                                    ;  TEST 11
2578                                                    ;--------------
2579  013520  012737  000011  001226                    TST11:  MOV     #11,TSTNO
2580  013526  012737  014064  001216                            MOV     #TST12,NEXT
2581  013534  012700  000000                                    MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2582  013540  013737  001416  001236                            MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2583  013546  100402                                            BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2584  013550  004737  013636                                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2585  013554  012700  000004                    100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2586  013560  013737  001420  001236                            MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2587  013566  100402                                            BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2588  013570  004737  013636                                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2589  013574  012700  000010                    101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2590  013600  013737  001422  001236                            MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2591  013606  100402                                            BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2592  013610  004737  013636                                    JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2593  013614  012700  000014                    102$:   MOV     #12.,R0         ;LOAD LINE NO.
2594  013620  013737  001424  001236                            MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2595  013626  100402                                            BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2596  013630  004737  013636                                    JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2597  013634  104400                    103$:   SCOPE                           ;SCOPE THIS TEST.
2598  013636                            105$:                                   ;TEST ENTRANCE.
2599  013636  104413                                            RAMCLR                  ;CLEAR ALL SEC REGISTERS
2600  013640  012737  102010  021020                            MOV     #CRC.CCITT,XPOLY        ;SET SOFTWARE POLYNOMINAL
2601  013646  012705  000025                                    MOV     #25,R5          ;SET DATA CHAR
2602  013652  110537  021350                                    MOVB    R5,TXBAP        ;LOAD DATA
2603  013656  112765  000004  022350                            MOVB    #BIT2,TXTAB(R5) ;SET CNTRL BYTE
2604  013664  012702  000004                                    MOV     #4,R2           ;SET FOR 4 LINE GROUP
2605  013670  010077  165476                    1$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
2606  013674  004537  021026                            PERFORM ,SETREG         ;
2607  013700     000     001                            .BYTE   000,001         ;PRINCIPLE BA, BC
2608  013702  021350                                    TXBAP                   ;
2609  013704  177777                                    -1                      ;
2610  013706  004537  021026                            PERFORM ,SETREG         ;
2611  013712     013     010                            .BYTE   013,010         ;LINE STATE, CNTRL TABLE
2612  013714  000004                                    BIT2                    ;TXGO
2613  013716  022350                                    TXTAB                   ;TABLE
2614  013720  004537  021026                            PERFORM ,SETREG         ;
2615  013724     006     012                            .BYTE   006,012         ;TX BCC REG, LINE PROTOCOL
2616  013726  000037                                    37                      ;BCC
2617  013730  000030                                    BIT4+BIT3                       ;POLYNOMINAL SELECT
2618  013732  032737  004000  001236                    BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
2619  013740  001407                                    BEQ     60$             ;#BR IF NO.
2620  013742  004537  021072                            PERFORM ,LOAD.MODE      ;#
2621  013746  015000                                    <BIT12+BIT11>+BIT9      ;#8 BITS/PER/CHAR
2622  013750  004537  021072                            PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
2623  013754  072000                                    <BIT14+BIT13+BIT12>+BIT10
2624                                                                            ;#9600 BAUD.
2625  013756  000403                                    BR      61$             ;#CONTINUE TEST.
2626  013760  004537  021072                    60$:    PERFORM ,LOAD.MODE      ;LOAD
2627  013764  014000                                    BIT12+BIT11             ;MODE
2628  013766  005277  165370                    61$:    INC     @DVSCR          ;SET MICRO CODE GO
2629  013772  005005                                    CLR     R5              ;DELAY
2630  013774  005777  165362                    2$:     TST     @DVSCR          ;FOR
```

```
                                                        BMI     3$              ;EILO
                                                        DELAY                   ;WASTE TIME
                                                        INC     R5              ;ENTRY
                                                        BNE     2$              ;
014410  112705  000005  165356                  3$:     MOVB    #5,@DVSRSH      ;SEL BCC REGISTER
014416  012703  002000                                  MOV     #2000,R3        ;SET DELAY TIME
014422  104414                                  5$:     DELAY                   ;WASTE TIME
014424  017704  165346                                  MOV     @DVSRA,R4       ;READ BCC REG
014430  001402                                          BEQ     6$              ;BR IF=0
014432  005303                                          DEC     R3              ;DELAY DONE?
014434  001372                                          BNE     5$              ;NO!
014436                                          6$:
014436  005005                                          CLR     R5              ;SET EXPECTED=0
014440  117701  165330                                  MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
014444  020504                                          CMP     R5,R4           ;SOFTWARE=HRDWARE?
014446  001401                                          BEQ     4$              ;BR IF YES
014450  104003                                          HLT     3               ;HRDWARE BCC WRONG
014452  104412                                  4$:     MSTCLR                  ;INIT DV11
014454  005200                                          INC     R0              ;UPDATE LINE NO.
014456  005302                                          DEC     R2              ;4 LINE GROUP DONE?
014460  001303                                          BNE     1$              ;BR IF NO
014462  000207                                          RTS     PC              ;EXIT FOR NEXT 4 LINE GROUP

                                                ;************************ TEST 12 ******************************
                                                ;*TEST TO FORCE NPR STATUS OVERFLOW.
                                                ;*TEST TO XMIT 65.10  TRANSFERS NOT
                                                ;*SERVICING THE FLAGS AND MAKING SURE THAT BIT10
                                                ;*OF THE DVSCR SETS ON THE 65TH ENTRY ONLY.
                                                ;*THIS TEST IS EXERCISED FOR BOTH ASYNC AND SYNC LINE CARDS.
                                                ;;*************************************************************

                                                ;  TEST 12
                                                ;--------------

014464  012737  000012  001226          TST12:  MOV     #12,TSTNO
014472  012737  014506  001216                  MOV     #TST13,NEXT
014100  012700  000000                          MOV     #0,,R0          ;PLACE LINE NUMBER INTO R0
014104  013737  001416  001236                  MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
014112  100402                                  BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
014114  004737  014202                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
014120  012700  000004                  100$:   MOV     #4,,R0          ;PLACE LINE NUMBER INTO R0
014124  013737  001420  001236                  MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
014132  100402                                  BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
014134  004737  014202                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
014140  012700  000010                  101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
014144  013737  001422  001236                  MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
014152  100402                                  BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
014154  004737  014202                          JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
014160  012700  000014                  102$:   MOV     #12.,R0         ;LOAD LINE NO.
014164  013737  001424  001236                  MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
014172  100402                                  BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
014174  004737  014202                          JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
014200  104400                  103$:   SCOPE                           ;SCOPE THIS TEST.
014202                          105$:                                   ;TEST ENTRANCE.
014202  104413                                  RAMCLR                  ;CLEAR ALL SEC. REGISTERS
014204  012702  000004                          MOV     #4,R2           ;SET FOR 4 LINE GROUP
014210  010077  165156                  1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
```

```
                                                PERFORM ,SETREG         ;
                                                .BYTE   000,001         ;TX BA P, TX BC P
                                                TXBAP                   ;
                                                -1                      ;
                                                PERFORM ,SETREG         ;
                                                .BYTE   010,013         ;TX CNTRL TABLE, LINE STATE
                                                TXTAB                   ;
                                                BIT2                    ;TX GO
                                                CLR     TXTAB           ;CLEAR CNTRL BYTE
                                                CLR     TXBAP           ;CLEAR TX DATA
                                                BIT     #ASYNC,STAT     ;#IS THIS ASYNC LINE CARD?
                                                BEQ     60$             ;#BR IF NO.
                                                PERFORM ,LOAD.MODE      ;#
                                                (BIT12+BIT11)+BIT9      ;#8 BITS/PER/CHAR
                                                PERFORM ,LOAD.MODE      ;#SET BAUD RATE REGISTER.
                                                (BIT14+BIT13+BIT12)+BIT10
                                                                        ;#9600 BAUD.
                                                BR      61$             ;#CONTINUE TEST.
                                        60$:    PERFORM ,LOAD.MODE      ;LOAD
                                                BIT12+BIT11             ;MODE
                                        61$:    INC     @DVSCR          ;SET MICRO CPU GO
                                                MOV     #63.,R3         ;SET FOR 63 ENTRIES INTO SILO
                                        2$:     MOVB    #13,@DVSRSH     ;SEL LINE STATE
                                                BIT     #BIT2,@DVSRA    ;IS TX GO SET?
                                                BEQ     3$              ;BR IF NO
                                                DELAY                   ;GIVE UCPU TIME
                                                                        ;TO ACCESS REG
                                                BR      2$              ;LOOK AGAIN
                                        3$:     BIT     #BIT10,@DVSCR   ;DID SILO FULL SET?
                                                BEQ     .+4             ;BR IF NO
                                                HLT     0               ;SILO FULL SET TOO SOON
                                                DEC     R3              ;63 ENTRIES MADE?
                                                BEQ     4$              ;BR IF NO
                                                PERFORM ,SETREG         ;TXBAP, TXBCP
                                                .BYTE   000,001         ;RELOAD BA
                                                TXBAP                   ;       BC
                                                -1                      ;
                                                MOVB    #13,@DVSRSH     ;SEL LINE STATE
                                                MOV     #BIT2,@DVSRA    ;CLEAR "USE SEC. TABLE" SET TX GO
                                                BR      2$              ;CONTINUE
                                        4$:     PERFORM ,SETREG         ;
                                                .BYTE   001,013         ;TX BC, LINE STATE
                                                -1                      ;SET FOR 64TH ENTRY
                                                BIT2                    ;TX GO
                                        6$:     MOVB    #13,@DVSRSH     ;SEL LINE STATE
                                                BIT     #BIT2,@DVSRA    ;TX GO SET?
                                                BEQ     5$              ;BR IF NO
                                                DELAY                   ;GIVE UCPU TIME
                                                                        ;TO ACCESS REG
                                                BR      6$              ;CONTINUE
                                        5$:     MOV     #10,R4          ;
                                                DELAY
                                                DEC     R4
                                                BNE     .-4
                                                MOV     DVSCR,R1        ;SET POINTER
                                                MOV     (R1),R4         ;READ DVSCR
```

```
014460  010405                          MOV     R4,R5           ;PLACE IMAGE IN GOOD
014462  052705  002000                  BIS     #BIT10,R5       ;SET NPR STATUS OVERFLOW
014466  030504                          CMP     R5,R4           ;WAS DVSCR OK?
014470  001401                          BEQ     1$              ;BR IF YES
014472  104002                          HLT     2               ;BIT10 OF DVSCR S/B=1  BILO OVERFLOW.
014474  104412                  1$:     MSTCLR                  ;CLEAR DV11
014476  005200                          INC     R0              ;UPDATE LINE NUMBER
014500  005302                          DEC     R2              ;ALL 4 LINES DONE?
014502  001345                          BNE     1$              ;BR IF NO
014504  000207                          RTS     PC              ;EXIT FOR NEXT 4 LINE GROUP

                                02000
                                        ;************************ TEST 13 ************************************
                                02200   ;*TEST TO TURN ON ALL
                                02300   ;*AVAILABLE TRANSMITTERS AT THE SAME TIME.
                                02400   ;*30(8) CHARS WILL BE XMITTED.
                                02500   ;*EXPECTED:
                                02600   ;*1 NUMBER OF LINES PRESENT X; ENTRIES INTO DVNSR
                                02700   ;*2)ALL TXBA = TXBA+30+LINE NO.
                                03000   ;*3)ALL TXWC = 0
                                03000   ;*4)ALL LINE STATE REGS =BIT7
                                03000   ;*5)ALL DVNSR = BIT15+BIT8+LINE NO.
                                03100   ;*
                                03200   ;*NOTE: ONLY PRIMARY REGISTERS ARE USED.
                                03300   ;*NOTE:TURN AROUND CONNECTOR MUST BE INSTALLED
                                03400   ;*       AND THE 'EIA' GATES ARE USED.
                                03450   ;*THIS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
                                        ;;*********************************************************************
                                03600
                                        ;  TEST 13
                                        ;--------------
014506  012737  000013  001226  TST13:  MOV     #13,TSTNO
014514  012737  015034  001216          MOV     #TST14,NEXT
014522  104413                          RAMCLR                  ;ZERO ALL SEC REGISTERS
014524  005001                          CLR     R1              ;SET COUNTER TO ZERO
014526  012700  000004                  MOV     #4,R0           ;SET 4 LINE COUNT
014532  005737  001416                  TST     L00.03          ;LINE CARD 1 EXIST?
014536  100401                          BMI     1$              ;BR IF NO
014540  060001                          ADD     R0,R1           ;UPDATE COUNT
014542  005737  001420          1$:     TST     L04.07          ;LINE CARD 2 EXIST?
014546  100401                          BMI     2$              ;BR IF NO
014550  060001                          ADD     R0,R1           ;
014552  005737  001422          2$:     TST     L08.11          ;LINE CARD 3 EXIST?
014556  100401                          BMI     3$              ;BR IF NO
014560  060001                          ADD     R0,R1           ;
014562  005737  001424          3$:     TST     L12.15          ;LINE CARD 4 EXIST?
014566  100401                          BMI     4$              ;BR IF NO
014570  060001                          ADD     R0,R1           ;
014572  005000                  4$:     CLR     R0              ;SET POINTER TO 0
014574  010137  025540                  MOV     R1,COUNT        ;STORE LINE CARD COUNT
014600  012702  021350          03900   MOV     #TXBAP,R2       ;SET TX BUFFER POINTER
014604  010237  014632          04000   MOV     R2,6$           ;LOAD BA
014610  105012                  04100   CLRB    (R2)            ;SET DATA TO ZERO
014612  010077  164554          04200   MOV     R0,@DVSRS       ;LOAD LINE NUMBER
014616  000240                  04300   NOP     ;PERFORM  LOAD.MODE
014620  000240                  04400   NOP     ;BIT12+BIT11
```

```
 2799  014622  000240                 04500            NOP     ;
 2900  014624  004537  021026         04600            PERFORM SETREG          ;
 2901  014630     000     001         04700            .BYTE   000,001         ;TX BA P, TX BC P
 2902  014632  000001                 04800    6$:     .BLKW   1               ;TXBA
 2903  014634  177750                 04900            -30                     ;TXBC
 2904  014636  004537  021026         05000            PERFORM SETREG          ;
 2905  014642     010     013         05100            .BYTE   010,013         ;TX CNTRL TAB, LINE STATE
 2906  014644  022350                 05200            TXTAB                   ;
 2907  014646  000004                 05300            BIT2                    ;TX GO
 2908  014650  005202                 05400            INC     R2              ;UPDATE BA POINTER
 2909  014652  005200                 05500            INC     R0              ;UPDATE LINE NO.
 2910  014654  022700  000020         05600            CMP     #16.,R0         ;ALL LINES DONE?
 2911  014660  001351                 05700            BNE     5$              ;BR IF NO
 2912  014662  005005                 05800            CLR     R5              ;
 2913  014664  005037  022350         05900            CLR     TXTAB           ;CLEAR CNTRL TABLE (1.0)
 2914  014670  012700  021350         06000            MOV     #TXBAP,R0       ;PRINCIPLE POINTER
 2915  014674  012704  021750         06100            MOV     #TXBAS,R4       ;ALTERNATE POINTER
 2916  014700  005020                 06200            CLR     (R0)+           ;CLEAR
 2917  014702  005024                 06300            CLR     (R4)+           ;TX BUFFERS
 2918  014704  022700  021420         06400            CMP     #TXBAP+50,R0    ;ALL DONE
 2919  014710  001373                 06500            BNE     .-10            ;BR IF NO
 2820  014712  005277  164444         06600    60$:    INC     @DVSCR          ;SET UCPU GO
 2821  014716  012702  021242         06700    7$:     MOV     #REGBUF,R2      ;SET BUFFER POINTER IN R2
 2822  014722  005777  164434         06800    8$:     TST     @DVSCR          ;SILO ENTRY?
 2823  014726  100005                 06900            BPL     9$              ;BR IF NO
 2824  014730  017722  164446         07000            MOV     @DVNSR,(R2)+    ;STORE SILO ENTRY AWAY
 2825  014734  005337  025540         07100            DEC     COUNT           ;ALL ENTRIES MADE?
 2826  014740  001404                 07200            BEQ     10$             ;BR IF YES
 2827  014742  104414                 07300    9$:     DELAY                   ;WASTE TIME
 2828  014744  005205                 07400            INC     R5              ;COUNT WAIT
 2829  014746  001365                 07500            BNE     8$              ;FOR SILO ENTRY
 2830  014750  104000                 07600            HLT     0               ;NO SILO ENTRY! (DVSCR15=1)
 2831  014752                         07700    10$:
 2832  014752  005737  001416         07800            TST     L00.03          ;LINE CARD 1 EXIST?
 2833  014756  100403                 07900            BMI     27$             ;BR IF NO
 2834  014760  004537  020054         08000            JSR     R5,LINT1                ;GOSUB
 2835  014764  000000                 08100            0.                      ;LINES 00-03
 2836  014766  005737  001420         08200    27$:    TST     L04.07          ;LINE CARD 2 EXIST?
 2837  014772  100403                 08300            BMI     28$             ;BR IF NO
 2838  014774  004537  020054         08400            JSR     R5,LINT1        ;GOSUB
 2839  015000  000004                 08500            4.                      ;LINES 04-07
 2940  015002  005737  001422         08600    28$:    TST     L08.11          ;LINE CARD 3 EXIST?
 2941  015006  100403                 08700            BMI     29$             ;BR IF NO
 2842  015010  004537  020054         08800            JSR     R5,LINT1        ;GOSUB
 2843  015014  000010                 08900            8.                      ;LINES 08-11
 2844  015016  005737  001424         09000    29$:    TST     L12.15          ;LINE CARD 4 EXIST?
 2845  015022  100403                 09100            BMI     30$             ;BR IF NO
 2846  015024  004537  020054         09200            JSR     R5,LINT1        ;GOSUB
 2847  015030  000014                 09300            12.                     ;LINES 12-15
 2848  015032  104400                 09400    30$:    SCOPE                   ;SCOPE TEST
 2849                                 09500
 2850                                 09600            ;************************ TEST 14 *******************************
 2851                                 09700            ;*TEST TO TURN ON ALL
 2852                                 09800            ;*AVAILABLE TRANSMITTERS AT THE SAME TIME.
 2853                                 09900            ;*30(8) CHARS WILL BE XMITTED.
 2854                                 10000            ;*EXPECTED:
```

```
                                                09100           ;*1)NUMBER OF LINES PRESENT X2 ENTRIES INTO DVNSR
                                                09200           ;*2)ALL TXBAP = TXBPA+30+LINE NO.
                                                09300           ;*3)ALL TXWCP = 0
                                                09400           ;*4)ALL LINE STATE REGS =0
                                                09500           ;*5)PRIMARY ENTRY IN DVNSR = BIT15+BIT8+LINE NO.
                                                09600           ;*6)SECONDARY ENTRY IN DVNSR =BIT15+BIT9+BIT8+LINE NO.
                                                09700           ;*7)ALL TX BAS = TXBAS+30+LINE NO.
                                                09800           ;*8) ALL TXWCS =0
                                                09900           ;*
                                                10000           ;*NOTE: PRIMARY REGISTERS ARE USED FIRST; AND THEN THE SECONDARY.
                                                10100           ;*NOTE:TURN AROUND CONNECTOR MUST BE INSTALLED
                                                10200           ;*      AND THE 'EIA' GATES ARE USED.
                                                10205           ;*THIS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
                                                                ;;*****************************************************************
                                                10400
                                                        ;  TEST 14
                                                        ;--------------
015034  012737  000014  001226                  TST14:  MOV     #14,TSTNO
015042  012737  015400  001216                          MOV     #TST15,NEXT
015050  104413                                          RAMCLR                  ;ZERO ALL SEC REGISTERS
015052  005001                                          CLR     R1              ;SET COUNTER TO ZERO
015054  012700  000004                                  MOV     #4,R0           ;SET 4 LINE COUNT
015060  005737  001416                                  TST     L00.03          ;LINE CARD 1 EXIST?
015064  100401                                          BMI     1$              ;BR IF NO
015066  060001                                          ADD     R0,R1           ;UPDATE COUNT
015070  005737  001420                          1$:     TST     L04.07          ;LINE CARD 2 EXIST?
015074  100401                                          BMI     2$              ;BR IF NO
015076  060001                                          ADD     R0,R1           ;
015100  005737  001422                          2$:     TST     L08.11          ;LINE CARD 3 EXIST?
015104  100401                                          BMI     3$              ;BR IF NO
015106  060001                                          ADD     R0,R1           ;
015110  005737  001424                          3$:     TST     L12.15          ;LINE CARD 4 EXIST?
015114  100401                                          BMI     4$              ;BR IF NO
015116  060001                                          ADD     R0,R1           ;
015120  005000                                  4$:     CLR     R0              ;SET POINTER TO 0
015122  010137  025540                                  MOV     R1,COUNT        ;STORE LINE CARD COUNT
015126  012737  021360  015172  10700                   MOV     #TXBAP,6$       ;SET TX BA POINTER
015134  012737  021750  015216  10800                   MOV     #TXBAS,11$      ;SET ALTERNATE POINTER
015142  105077  000024          10900           5$:     CLRB    @6$             ;ZERO DATA
015146  105077  000044          11000                   CLRB    @11$            ;ZERO DATA
015152  010077  164214          11100                   MOV     R0,@DVSRS       ;LOAD LINE NUMBER
015156  000240                  11200           60$:    NOP     ;PERFORM  ,LOAD.MODE
015160  000240                  11300                   NOP     ;BIT12+BIT11
015162  000240                  11400           61$:    NOP     ;
015164  004537  021026          11500                   PERFORM ,SETREG         ;
015170     000     001          11600                   .BYTE   000,001         ;TX BA P, TX BC F
015172  000001                  11700           6$:     .BLKW   1               ;TXBA
015174  177750                  11800                   -30                     ;TX BC
015176  004537  021026          11900                   PERFORM ,SETREG         ;
015202     010     013          12000                   .BYTE   010,013         ;TX CNT TAB, LINE STATE
015204  022350                  12100                   TXTAB                   ;
015206  000004                  12200                   BIT2                    ;TX GO
015210  004537  021026          12300                   PERFORM ,SETREG         ;
015214     002     003          12400                   .BYTE   002,003         ;TX BA (ALT), TX BC (ALT)
015216  000001                  12500           11$:    .BLKW 1                 ;TX BA
015220  177750                  12600                   -30                     ;TX BC
```

```
2911  015222  005237  015216          12700          INC     11$             ;UPDATE TX BA POINTER (ALT.
2912  015226  005237  015172          12800          INC     6$              ;UPDATE TX BA POINTER (P.
2913  015232  005200                  12900          INC     R0              ;UPDATE LINE POINTER
2914  015234  022700  000020          13000          CMP     #16.,R0         ;ALL LINES DONE?
2915  015240  001340                  13100          BNE     5$              ;BR IF NO
2916  015242  005005                  13200          CLR     R5              ;
2917  015244  063737  025540  025540  13300          ADD     COUNT,COUNT     ;DOUBLE THE COUNT (P+A)
2918  015252  005037  022350          13400          CLR     TXTAB           ;CLEAR CNTRL TABLE (1.0)
2919  015256  005277  164100          13500          INC     @DVSCR          ;SET UCPU GO
2920  015262  012702  021242                  7$:    MOV     #REGBUF,R2      ;SET BUFFER POINTER IN R2
2921  015266  005777  164070          8$:    TST     @DVSCR          ;SILO ENTRY?
2922  015272  100005                         BPL     9$              ;BR IF NO
2923  015274  017722  164102                 MOV     @DVNSR,(R2)+    ;STORE SILO ENTRY AWAY
2924  015300  005337  025540                 DEC     COUNT           ;ALL ENTRIES MADE?
2925  015304  001404                         BEQ     10$             ;BR IF YES
2926  015306  104414                  9$:    DELAY                   ;WASTE TIME
2927  015310  005205                         INC     R5              ;COUNT WAIT
2928  015312  001365                         BNE     8$              ;FOR SILO ENTRY
2929  015314  104000                         HLT     0               ;NO SILO ENTRY! (DVSCR15=1)
2930  015316                          10$:
2931  015316  005737  001416          13700          TST     L00.03          ;LINE CARD 1 EXIST?
2932  015322  100403                  13800          BMI     27$             ;BR IF NO
2933  015324  004537  020316          13900          JSR     R5,_INT2        ;GOSUB
2934  015330  000000                  14000          0.                      ;LINES 00-03
2935  015332  005737  001420          14100  27$:    TST     L04.07          ;LINE CARD 2 EXIST?
2936  015336  100403                  14200          BMI     28$             ;BR IF NO
2937  015340  004537  020316          14300          JSR     R5,_INT2        ;GOSUB
2938  015344  000004                  14400          4.                      ;LINES 04-07
2939  015346  005737  001422          14500  28$:    TST     L08.11          ;LINE CARD 5 EXIST?
2940  015352  100403                  14600          BMI     29$             ;BR IF NO
2941  015354  004537  020316          14700          JSR     R5,_INT2        ;GOSUB
2942  015360  000010                  14800          8.                      ;LINES 08-11
2943  015362  005737  001424          14900  29$:    TST     L12.15          ;LINE CARD 4 EXIST?
2944  015366  100403                  15000          BMI     30$             ;BR IF NO
2945  015370  004537  020316          15100          JSR     R5,_INT2        ;GOSUB
2946  015374  000014                  15200          12.                     ;LINES 12-15
2947  015376  104400                  15300  30$:    SCOPE                   ;SCOPE TEST
2948                                  15400
2949                                          ;************************ TEST 15 *******************************
2950                                  15600          ;*TEST TO TURN ON ALL TRANSMITTER
2951                                  15700          ;*BCC'S (USING LRC 8).
2952                                  15800          ;*ALL TRANSMITTER PRIMARY REGISTERS
2953                                  15900          ;*WILL XMIT 1 CHARACTER (20(8)+LINE NO.)
2954                                  16000          ;*AT THIS TIME ALL BCC REGISTERS WILL BE
2955                                  16100          ;*CHECKED FOR CORRECT DATA. (20(8)+LINE NO.)
2956                                  16200          ;*NOTE: BCC INSTRUCTION HAS BEEN CHECKED AND
2957                                  16300          ;*LRC 8 ALSO BUT IT IS NOT PROVEN
2958                                  16400          ;*THAT THE CHAR HAS BEEN "TAKEN FROM" CORE BY THE MICRO-PROCESSOR YET.
2959                                  16500          ;*NOTE:TURN AROUND CONNECTOR MUST BE INSTALLED
2960                                  16600          ;*      AND THE 'EIA' GATES ARE USED.
2961                                  16605          ;*THIS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
2962                                          ;;*************************************************************
2963                                  16800
2964                                          ;  TEST 15
2965                                          ;--------------
2966  015400  012737  000015  001226  TST15:  MOV     #15,TSTNO
```

```
2957  015406  012737  015176  001216                    MOV     #TST16,NEXT
2958  015414  104413                                    RAMCLR                  ;ZERO ALL SEC REGISTERS
2959  015416  005001                                    CLR     R1              ;SET COUNTER TO ZERO
2960  015420  012700  000004                            MOV     #4,R0           ;SET 4 LINE COUNT
2961  015424  005737  001416                            TST     L00.03          ;LINE CARD 1 EXIST?
2962  015430  100401                                    BMI     1$              ;BR IF NO
2963  015432  060001                                    ADD     R0,R1           ;UPDATE COUNT
2964  015434  005737  001420                    1$:     TST     L04.07          ;LINE CARD 2 EXIST?
2965  015440  100401                                    BMI     2$              ;BR IF NO
2966  015442  060001                                    ADD     R0,R1           ;
2967  015444  005737  001422                    2$:     TST     L08.11          ;LINE CARD 3 EXIST?
2968  015450  100401                                    BMI     3$              ;BR IF NO
2969  015452  060001                                    ADD     R0,R1           ;
2970  015454  005737  001424                    3$:     TST     L12.15          ;LINE CARD 4 EXIST?
2971  015460  100401                                    BMI     4$              ;BR IF NO
2972  015462  060001                                    ADD     R0,R1           ;
2973  015464  005000                            4$:     CLR     R0              ;SET POINTER TO 0
2974  015466  010137  025540                            MOV     R1,COUNT        ;STORE LINE CARD COUNT
2975  015472  012700  022350                    17100   MOV     #TXTAB,R0       ;SET CNTRL TABLE POINTER
2976  015476  012703  000020                    17200   MOV     #16.,R3         ;SET FOR 16 LINES
2977  015502  062700  000020                    17300   ADD     #20,R0          ;SET DATA CHAR
2978  015506  112720  000010                    17400   5$:     MOVB    #BIT3,(R0)+     ;"INC BCC"
2979  015512  005303                            17500           DEC     R3              ;16 X DONE?
2980  015514  001374                            17600           BNE     5$              ;BR IF NO
2981  015516  012702  000020                    17700           MOV     #20,R2          ;SET DATA
2982  015522  012700  021350                    17800           MOV     #TXBAP,R0       ;SET TX BUFFER POINTER
2983  015526  012703  000020                    17900           MOV     #16.,R3         ;16 LINES
2984  015532  110220                            18000   6$:     MOVB    R2,(R0)+        ;LOAD DATA
2985  015534  005202                            18100           INC     R2              ;UPDATE DATA
2986  015536  005303                            18200           DEC     R3              ;ALL DONE
2987  015540  001374                            18300           BNE     6$              ;BR IF NO
2988  015542  012737  021350  015572            18400           MOV     #TXBAP,8$       ;SET POINTER
2989  015550  005000                            18500           CLR     R0              ;SET TO LINE 0.
3000  015552  010077  163614                    18600   7$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
3001  015556  000240                            18700   60$:    NOP     ;PERFORM ,LOAD.MODE
3002  015560  000240                            18800           NOP     ;BIT12+BIT11
3003  015562  000240                            18900   61$:    NOP     ;
3004  015564  004537  021026                    19000           PERFORM ,SETREG         ;
3005  015570     000     001                    19100           .BYTE   000,001         ;TX BA P, TX BC P
3006  015572  000001                            19200   8$:     .BLKW   1               ;TX BA
3007  015574  177777                            19300           -1                      ;TX BC
3008  015576  004537  021026                    19400           PERFORM ,SETREG         ;
3009  015602     010     013                    19500           .BYTE   010,013         ;TX CNTRL TAB, LINE STATE
3010  015604  022350                            19600           TXTAB                   ;
3011  015606  000004                            19700           BIT2                    ;TX GO
3012  015610  005237  015572                    19800           INC     8$              ;UPDATE BA POINTER
3013  015614  005200                            19900           INC     R0              ;UPDATE LINE POINTER
3014  015616  022700  000020                    20000           CMP     #16.,R0         ;ALL DONE?
3015  015622  001353                            20100           BNE     7$              ;BR IF NO
3016  015624  005005                            20200           CLR     R5              ;
3017  015626  005277  163530                    20300           INC     @DVSCR          ;SET UCPU GO
3018  015632  005777  163544                    20400   9$:     TST     @DVNSR          ;SILO ENTRY?
3019  015636  100003                            20500           BPL     11$             ;BR IF NO
3020  015640  005337  025540                    20600           DEC     COUNT           ;ALL DONE?
3021  015644  001404                            20700           BEQ     10$             ;BR IF NO
3022  015646  104414                            20800   11$:    DELAY                   ;WASTE TIME
```

```
3023  015650  005205                  20900          INC     R5              ;UPCOUNT DELAY
3024  015652  001367                  21000          BNE     9$              ;BR
3025  015654  104000                  21100          HLT     0               ;DVNSR15 NOT=1 (NO SILO ENTRY)
3026  015656  042777  000001  163476  21200  10$:    BIC     #BIT0,@DVSCR    ;CLEAR UCPU GO
3027
3028  015664  005737  001416                         TST     L00.03          ;DOES LINE CARD EXIST?
3029  015670  100426                                 BMI     66$             ;BR IF NO
3030  015672  012700  000000                         MOV     #0.,R0          ;LOAD LINE POINTER
3031  015676  012702  000004                         MOV     #4,R2           ;SET FOR 4 LINE GROUP
3032  015702  010077  163464                 64$:    MOV     R0,@DVSRS       ;LOAD LINE NO.
3033  015706  112777  000006  163460                 MOVB    #6,@DVSRSH      ;SEL TX BCC REG
3034  015714  017704  163456                         MOV     @DVSRA,R4       ;READ IT
3035  015720  012705  000020                         MOV     #20,R5          ;SET EXPECTED=20
3036  015724  060005                                 ADD     R0,R5           ;ADD LINE NO TO EXPECTED
3037  015726  117701  163442                         MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
3038  015732  020504                                 CMP     R5,R4           ;TX BCC OK?
3039  015734  001401                                 BEQ     65$             ;BR IF YES
3040  015736  104003                                 HLT     3               ;TX BCC ERROR
3041  015740  005200                         65$:    INC     R0              ;UPDATE LINE POINTER
3042  015742  005302                                 DEC     R2              ;4 LINE DONE?
3043  015744  001356                                 BNE     64$             ;BR IF NO
3044  015746                                 66$:
3045
3046  015746  005737  001420                         TST     L04.07          ;DOES LINE CARD EXIST?
3047  015752  100426                                 BMI     69$             ;BR IF NO
3048  015754  012700  000004                         MOV     #4.,R0          ;LOAD LINE POINTER
3049  015760  012702  000004                         MOV     #4,R2           ;SET FOR 4 LINE GROUP
3050  015764  010077  163402                 67$:    MOV     R0,@DVSRS       ;LOAD LINE NO.
3051  015770  112777  000006  163376                 MOVB    #6,@DVSRSH      ;SEL TX BCC REG
3052  015776  017704  163374                         MOV     @DVSRA,R4       ;READ IT
3053  016002  012705  000020                         MOV     #20,R5          ;SET EXPECTED=20
3054  016006  060005                                 ADD     R0,R5           ;ADD LINE NO TO EXPECTED
3055  016010  117701  163360                         MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
3056  016014  020504                                 CMP     R5,R4           ;TX BCC OK?
3057  016016  001401                                 BEQ     68$             ;BR IF YES
3058  016020  104003                                 HLT     3               ;TX BCC ERROR
3059  016022  005200                         68$:    INC     R0              ;UPDATE LINE POINTER
3060  016024  005302                                 DEC     R2              ;4 LINE DONE?
3061  016026  001356                                 BNE     67$             ;BR IF NO
3062  016030                                 69$:
3063
3064  016030  005737  001422                         TST     L08.11          ;DOES LINE CARD EXIST?
3065  016034  100426                                 BMI     72$             ;BR IF NO
3066  016036  012700  000010                         MOV     #8.,R0          ;LOAD LINE POINTER
3067  016042  012702  000004                         MOV     #4,R2           ;SET FOR 4 LINE GROUP
3068  016046  010077  163320                 70$:    MOV     R0,@DVSRS       ;LOAD LINE NO.
3069  016052  112777  000006  163314                 MOVB    #6,@DVSRSH      ;SEL TX BCC REG
3070  016060  017704  163312                         MOV     @DVSRA,R4       ;READ IT
3071  016064  012705  000020                         MOV     #20,R5          ;SET EXPECTED=20
3072  016070  060005                                 ADD     R0,R5           ;ADD LINE NO TO EXPECTED
3073  016072  117701  163276                         MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
3074  016076  020504                                 CMP     R5,R4           ;TX BCC OK?
3075  016100  001401                                 BEQ     71$             ;BR IF YES
3076  016102  104003                                 HLT     3               ;TX BCC ERROR
3077  016104  005200                         71$:    INC     R0              ;UPDATE LINE POINTER
3078  016106  005302                                 DEC     R2              ;4 LINE DONE?
```

```
3079  016110  001356                                    BNE     70$             ;BR IF NO
3080  016112                                    72$:
3081
3082  016112  005737  001424                            TST     L12.15          ;DOES LINE CARD EXIST?
3083  016116  100426                                    BMI     75$             ;BR IF NO
3084  016120  012700  000014                            MOV     #12.,R0         ;LOAD LINE POINTER
3085  016124  012702  000004                            MOV     #4,R2           ;SET FOR 4 LINE GROUP
3086  016130  010077  163236                    73$:    MOV     R0,@DVSRS       ;LOAD LINE NO.
3087  016134  112777  000006  163232                    MOVB    #6,@DVSRSH      ;SEL TX BCC REG
3088  016142  017704  163230                            MOV     @DVSRA,R4       ;READ IT
3089  016146  012705  000020                            MOV     #20,R5          ;SET EXPECTED=20
3090  016152  060005                                    ADD     R0,R5           ;ADD LINE NO TO EXPECTED
3091  016154  117701  163214                            MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
3092  016160  020504                                    CMP     R5,R4           ;TX BCC OK?
3093  016162  001401                                    BEQ     74$             ;BR IF YES
3094  016164  104003                                    HLT     3               ;TX BCC ERROR
3095  016166  005200                            74$:    INC     R0              ;UPDATE LINE POINTER
3096  016170  005302                                    DEC     R2              ;4 LINE DONE?
3097  016172  001356                                    BNE     73$             ;BR IF NO
3098  016174                                    75$:
3099  016174  104400                 21700              SCOPE
3100                                  21800
3101                                  21900
3102
3103                                                    ;************************ TEST 16 *******************************
3104                                                    ;*TEST OF RECEIVER WITH MICRO-PROCESSOR FREE RUNNING.
3105                                                    ;*TEST OF RECEIVER BYTE COUNT =0
3106                                                    ;*TEST OF ***"ERROR CODES"*** AND RECEIVER ENTRIES INTO DARIC
3107                                                    ;*REGISTER.
3108                                                    ;*EXPECTED:     (THIS IS FOR LINES $B )
3109                                                    ;*DVSCR BIT15=1 BIT7=1 BIT0=1
3110                                                    ;*DVRIC 11:08=LINE NUMBER BIT15+32
3111                                                    ;*DVNSR 03:00=LINE NUMBER BIT15=1 BIT8=1
3112                                                    ;*TXBAP =TXBAP+1
3113                                                    ;*TXWCP =0
3114                                                    ;*RXBA  =RXBA +0
3115                                                    ;*RXWC  =0
3116                                                    ;*LINE STATE     =BIT7=1 BIT0=1
3117                                                    ;*IS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
3118                                                    ;;***********************************************************
3119
3120                                            ;  TEST 16
3121                                            ;--------------
3122  016176  012737  000016  001226            TST16:  MOV     #16,TSTNO
3123  016204  012737  016350  001216                    MOV     #TST17,NEXT
3124  016212  012700  000000                            MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
3125  016216  013737  001416  001236                    MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
3126  016224  100402                                    BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3127  016226  004737  016314                            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3128  016232  012700  000004                    100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3129  016236  013737  001420  001236                    MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3130  016244  100402                                    BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
3131  016246  004737  016314                            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3132  016252  012700  000010                    101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
3133  016256  013737  001422  001236                    MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
3134  016264  100402                                    BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
```

```
3135  016266  004737  016314                          JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
3136  016272  012700  000014                  102$:   MOV     #12.,R0         ;LOAD LINE NO.
3137  016276  013737  001424  001236                  MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
3138  016304  100402                                  BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
3139  016306  004737  016314                          JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
3140  016312  104400                          103$:   SCOPE                   ;SCOPE THIS TEST.
3141  016314                                  105$:                           ;TEST ENTRANCE.
3142  016314  012737  022750  025544                  MOV     #RXBA,EXPRBA    ;EXPECTED RX BA
3143  016322  012737  000000  025546                  MOV     #0,EXPRWC       ;EXPECTED RX BC
3144  016330  004537  017220                          JSR     R5,RXT01        ;GOTO SUBROUTINE
3145  016334  023350                                  RXTAB                   ;LOADED INTO RX CNTR TABLE REG
3146  016336  000000                                  0                       ;LOADED INTO CORE TABLE
3147  016340  022750                                  RXBA                    ;LOADED INTO RX BA
3148  016342  000000                                  0                       ;LOADED INTO RX BC
3149  016344  100032                                  BIT15+32                ;RIC ENTRY
3150  016346  000207                                  RTS     PC              ;EXIT
3151
3152
3153                                                  ;************************ TEST 17 ******************************
3154                                                  ;*TEST OF RECEIVER WITH MICRO-PROCESSOR FREE RUNNING.
3155                                                  ;*TEST OF RECEIVER BYTE COUNT WARINING
3156                                                  ;*TEST OF ***"ERROR CODES"*** AND RECEIVER ENTRIES INTO DVRIC
3157                                                  ;*REGISTER.
3158                                                  ;*EXPECTED:     (THIS IS FOR LINES $B )
3159                                                  ;*DVSCR BIT15=1 BIT7=1 BIT0=1
3160                                                  ;*DVRIC 11:08=LINE NUMBER BIT14+32
3161                                                  ;*DVNSR 03:00=LINE NUMBER BIT15=1 BIT8=1
3162                                                  ;*TXBAP =TXBAP+1
3163                                                  ;*TXWCP =0
3164                                                  ;*RXBA  =RXBA+1
3165                                                  ;*RXWC  =0
3166                                                  ;*LINE STATE    =BIT7=1 BIT0=1
3167                                                  ;*IS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
3168                                                  ;:**************************************************************
3169
3170                                          ;  TEST 17
3171                                          ;--------------
3172  016350  012737  000017  001226          TST17:  MOV     #17,TSTNO
3173  016356  012737  016522  001216                  MOV     #TST20,NEXT
3174  016364  012700  000000                          MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
3175  016370  013737  001416  001236                  MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
3176  016376  100402                                  BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3177  016400  004737  016466                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3178  016404  012700  000004                  100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3179  016410  013737  001420  001236                  MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3180  016416  100402                                  BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
3181  016420  004737  016466                          JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3182  016424  012700  000010                  101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
3183  016430  013737  001422  001235                  MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
3184  016436  100402                                  BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
3185  016440  004737  016466                          JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
3186  016444  012700  000014                  102$:   MOV     #12.,R0         ;LOAD LINE NO.
3187  016450  013737  001424  001236                  MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
3188  016456  100402                                  BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
3189  016460  004737  016466                          JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
3190  016464  104400                          103$:   SCOPE                   ;SCOPE THIS TEST.
```

```
 3191  016466                                  105$:                           ;TEST ENTRANCE.
 3192  016466  012737  022751  025544                MOV     #RXBA+1,EXPRBA    ;EXPECTED RX BA
 3193  016474  012737  000000  025546                MOV     #0,EXPRWC         ;EXPECTED RX BC
 3194  016502  004537  017220                        JSR     R5,RXT01          ;GOTO SUBROUTINE
 3195  016506  023350                                RXTAB                     ;LOADED INTO RX CNTR TABLE REG
 3196  016510  000000                                0                         ;LOADED INTO CORE TABLE
 3197  016512  022750                                RXBA                      ;LOADED INTO RX BA
 3198  016514  177777                                -1                        ;LOADED INTO RX BC
 3199  016516  040032                                BIT14+32                        ;RIC ENTRY
 3200  016520  000207                                RTS     PC                ;EXIT
 3201
 3202
 3203                                                ;********************** TEST 20 ****************************
 3204                                                ;*TEST OF RECEIVER WITH MICRO-PROCESSOR FREE RUNNING.
 3205                                                ;*TEST OF RECEIVER "RXBA NXM"
 3206                                                ;*TEST OF ***"ERROR CODES"*** AND RECEIVER ENTRIES INTO DVRIC
 3207                                                ;*REGISTER.
 3208                                                ;*EXPECTED:    (THIS IS FOR LINES $B )
 3209                                                ;*DVSCR BIT15=1 BIT7=1 BIT0=1
 3210                                                ;*DVRIC 11:08=LINE NUMBER BIT15+BIT14+32
 3211                                                ;*DVNSR 03:00=LINE NUMBER BIT15=1 BIT8=1
 3212                                                ;*TXBAP =TXBAP+1
 3213                                                ;*TXWCP =0
 3214                                                ;*RXBA  =177320 +0
 3215                                                ;*RXWC  =-3
 3216                                                ;*LINE STATE    =BIT7=1 BIT0=1
 3217                                                ;*IS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
 3218                                                ;:**************************************************************
 3219
 3220                                            ;  TEST 20
 3221                                            ;--------------
 3222  016522  012737  000020  001226  TST20:    MOV     #20,TSTNO
 3223  016530  012737  016674  001216            MOV     #TST21,NEXT
 3224  016536  012700  000000                    MOV     #0.,R0            ;PLACE LINE NUMBER INTO R0
 3225  016542  013737  001416  001236            MOV     L00.03,STAT       ;LOAD LINE CARD STATUS INTO STAT
 3226  016550  100402                            BMI     100$              ;BR IF LINE CARD NOT TO BE TESTED
 3227  016552  004737  016640                    JSR     PC,105$           ;GO DO THE TEST FOR LINE CARD 1
 3228  016556  012700  000004  100$:    MOV     #4.,R0            ;PLACE LINE NUMBER INTO R0
 3229  016562  013737  001420  001236            MOV     L04.07,STAT       ;LOAD LINE CARD STATUS INTO STAT
 3230  016570  100402                            BMI     101$              ;BR IF LINE CARD NOT TO BE TESTED
 3231  016572  004737  016640                    JSR     PC,105$           ;GO DO THE TEST FOR LINE CARD 2
 3232  016576  012700  000010  101$:    MOV     #8.,R0            ;LOAD LINE NUMBER
 3233  016602  013737  001422  001236            MOV     L08.11,STAT       ;LOAD LINE CARD STATUS INTO STAT
 3234  016610  100402                            BMI     102$              ;BR IF LINE CARD NOT TO BE TESTED
 3235  016612  004737  016640                    JSR     PC,105$           ;DO THE TEST FOR LINE CARD 3
 3236  016616  012700  000014  102$:    MOV     #12.,R0           ;LOAD LINE NO.
 3237  016622  013737  001424  001236            MOV     L12.15,STAT       ;LOAD LINE CARD STATUS
 3238  016630  100402                            BMI     103$              ;BR IF LINE CARD NOT TO BE TESTED
 3239  016632  004737  016640                    JSR     PC,105$           ;DO THE TESTS FOR LINE CARD 4
 3240  016636  104400  103$:    SCOPE                                     ;SCOPE THIS TEST.
 3241  016640  105$:                                                      ;TEST ENTRANCE.
 3242  016640  012737  177320  025544                MOV     #177320,EXPRBA    ;EXPECTED RX BA
 3243  016646  012737  177775  025546                MOV     #-3,EXPRWC        ;EXPECTED RX BC
 3244  016654  004537  017220                        JSR     R5,RXT01          ;GOTO SUBROUTINE
 3245  016660  023350                                RXTAB                     ;LOADED INTO RX CNTR TABLE REG
 3246  016662  000000                                0                         ;LOADED INTO CORE TABLE
```

```
  3247  016664  177320                                    177320                  ;LOADED INTO RX BA
  3248  016666  177775                                    -3                      ;LOADED INTO RX BC
  3249  016670  140032                                    BIT15+BIT14+32                  ;RIC ENTRY
  3250  016672  000207                                    RTS     PC              ;EXIT
  3251
  3252
  3253                                                    ;************************ TEST 21 *******************************
  3254                                                    ;*TEST OF RECEIVER WITH MICRO-PROCESSOR FREE RUNNING.
  3255                                                    ;*TEST IF RECEIVER "RXTAB NXM"
  3256                                                    ;*TEST OF ***"ERROR CODES"*** AND RECEIVER ENTRIES INTO DVRIC
  3257                                                    ;*REGISTER.
  3258                                                    ;*EXPECTED:    (THIS IS FOR LINES $B )
  3259                                                    ;*DVSCR BIT15=1 BIT7=1 BIT0=1
  3260                                                    ;*DVRIC 11:08=LINE NUMBER BIT15+BIT14+BIT12+32
  3261                                                    ;*DVNSR 03:00=LINE NUMBER BIT15=1 BIT8=1
  3262                                                    ;*TXBAP =TXBAP+1
  3263                                                    ;*TXWCP =0
  3264                                                    ;*RXBA  =RXBA +0
  3265                                                    ;*RXWC  =-3
  3266                                                    ;*LINE STATE    =BIT7=1 BIT0=1
  3267                                                    ;*IS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
  3268                                                    ;;***************************************************************
  3269
  3270                                            ;  TEST 21
  3271                                            ;--------------
  3272  016674  012737  000021  001226           TST21:  MOV     #21,TSTNO
  3273  016702  012737  017046  001216                   MOV     #TST22,NEXT
  3274  016710  012700  000000                           MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
  3275  016714  013737  001416  001236                   MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
  3276  016722  100402                                   BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
  3277  016724  004737  017012                           JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
  3278  016730  012700  000004                   100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
  3279  016734  013737  001420  001236                   MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
  3280  016742  100402                                   BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
  3281  016744  004737  017012                           JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
  3282  016750  012700  000010                   101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
  3283  016754  013737  001422  001236                   MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
  3284  016762  100402                                   BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
  3285  016764  004737  017012                           JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
  3286  016770  012700  000014                   102$:   MOV     #12.,R0         ;LOAD LINE NO.
  3287  016774  013737  001424  001236                   MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
  3288  017002  100402                                   BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
  3289  017004  004737  017012                           JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
  3290  017010  104400                           103$:   SCOPE                   ;SCOPE THIS TEST.
  3291  017012                                   105$:                           ;TEST ENTRANCE.
  3292  017012  012737  022750  025544                   MOV     #RXBA,EXPRBA    ;EXPECTED RX BA
  3293  017020  012737  177775  025546                   MOV     #-3,EXPRWC      ;EXPECTED RX BC
  3294  017026  004537  017220                           JSR     R5,RXT01        ;GOTO SUBROUTINE
  3295  017032  177266                                   177320-32                       ;LOADED INTO RX CNTR TABLE REG
  3296  017034  000000                                   0                       ;LOADED INTO CORE TABLE
  3297  017036  022750                                   RXBA                    ;LOADED INTO RX BA
  3298  017040  177775                                   -3                      ;LOADED INTO RX BC
  3299  017042  150032                                   BIT15+BIT14+BIT12+32                    ;RIC ENTRY
  3300  017044  000207                                   RTS     PC              ;EXIT
  3301
  3302
```

```
3303                                        ;************************ TEST 22 ******************************
3304                                        ;*TEST OF RECEIVER WITH MICRO-PROCESSOR FREE RUNNING.
3305                                        ;*TEST OF RECEIVER "SPECIAL CHARACTER"
3306                                        ;*TEST OF ***"ERROR CODES"*** AND RECEIVER ENTRIES INTO DVRIC
3307                                        ;*REGISTER.
3308                                        ;*EXPECTED:    (THIS IS FOR LINES $B )
3309                                        ;*DVSCR BIT15=1 BIT7=1 BIT0=1
3310                                        ;*DVRIC 11:08=LINE NUMBER 32
3311                                        ;*DVNSR 03:00=LINE NUMBER BIT15=1 BIT8=1
3312                                        ;*TXBAP =TXBAP+1
3313                                        ;*TXWCP =0
3314                                        ;*RXBA  =RXBA +0
3315                                        ;*RXWC  =-3
3316                                        ;*LINE STATE    =BIT7=1 BIT0=1
3317                                        ;*IS TEST IS DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
3318                                        ;:*****************************************************************
3319
3320                                    ;  TEST 22
3321                                    ;--------------
3322  017046  012737  000022  001226   TST22:  MOV     #22,TSTNO
3323  017054  012737  002436  001216           MOV     #.EOP,NEXT
3324  017062  012700  000000                   MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
3325  017066  013737  001416  001236           MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
3326  017074  100402                           BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3327  017076  004737  017164                   JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3328  017102  012700  000004   100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3329  017106  013737  001420  001236           MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3330  017114  100402                           BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
3331  017116  004737  017164                   JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3332  017122  012700  000010   101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
3333  017126  013737  001422  001236           MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
3334  017134  100402                           BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
3335  017136  004737  017164                   JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
3336  017142  012700  000014   102$:   MOV     #12.,R0         ;LOAD LINE NO.
3337  017146  013737  001424  001236           MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
3338  017154  100402                           BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
3339  017156  004737  017164                   JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
3340  017162  104400   103$:   SCOPE                   ;SCOPE THIS TEST.
3341  017164          105$:                   ;TEST ENTRANCE.
3342  017164  012737  022750  025544           MOV     #RXBA,EXPRBA    ;EXPECTED RX BA
3343  017172  012737  177775  025546           MOV     #-3,EXPRWC      ;EXPECTED RX BC
3344  017200  004537  017220                   JSR     R5,RXT01        ;GOTO SUBROUTINE
3345  017204  023350                           RXTAB                   ;LOADED INTO RX CNTR TABLE REG
3346  017206  000001                           BIT0                    ;LOADED INTO CORE TABLE
3347  017210  022750                           RXBA                    ;LOADED INTO RX BA
3348  017212  177775                           -3                      ;LOADED INTO RX BC
3349  017214  000032                           32                      ;RIC ENTRY
3350  017216  000207                           RTS     PC              ;EXIT
3351
3352
```

```
                                                00103           ;*SUBROUTINE.
                                                00153           ;*SUBROUTINE USED BY PREVIOUS TESTS.
                                                00175           ;*
                                                00187   RXTJ1:
017220                                          00200           MOV     (R5)+,68$       ;LOAD RX CNTRL TABLE REG.
017224  012537  017416                          00300           MOV     (R5)+,112$      ;DATA TO CORE TABLE
017230  012537  020050                          00400           MOV     (R5)+,56$       ;RX BA
017234  012537  017366                          00500           MOV     (R5)+,67$       ;RX BC
017240  012537  020052                          00600           MOV     (R5)+,115$      ;EXPECTED DVRIC ENTRY
017244  010546                                  00700           MOV     R5,-(SP)        ;SAVE R5 ON STACK
017246  012705  023350                          00800           MOV     #RXTAB,R5       ;GET RX TAB POINTER
017252  013746  020050                          00900           MOV     112$,-(SP)      ;GET CNTRL FUNCTION
017256                                          01000   65$:
017256  111625                                  01100           MOVB    (SP),(R5)+      ;LOAD
017260  022705  023750                          01200           CMP     #RXTAB+400,R5   ;CNTR FUNCTION
017264  001374                                  01300           BNE     65$             ;BR IF NOT DONE
017266  005726                                  01400           TST     (SP)+           ;POP FUNCTION OFF STACK
017270  012737  000004  020046                  01500           MOV     #4,111$         ;SET FOR 4 LINE GROUP
017276  104413                                  01600   64$:    RAMCLR                  ;CLEAR ALL SEC REGISTERS
017300  110077  162066                          01700           MOVB    R0,@DVSRS       ;LOAD LINE NUMBER
017304  032737  004000  001236                  01800           BIT     #ASYNC.STAT     ;*IS THIS AN ASYNC LINE CARD?
017312  001406                                  01900           BEQ     50$             ;*BR IF NOT ASYNC.
017314  004537  021226                          02000           PERFORM ,SETREG         ;*ADJUST REGISTERS FOR ASYNC
017320     000     001                          02100           .BYTE   000,001         ;*TXBAP AND TXBC
017322  021350                                  02200           TXBAP                   ;*BUS ADDRESS
017324  177777                                  02300           -1                      ;*SET TO TRANS. ONE CHAR
017326  000405                                  02400           BR      51$             ;*CONTINUE TEST
017330  004537  021226                          02500   50$:    PERFORM ,SETREG         ;
017334     000     001                          02600           .BYTE   000,001         ;TX BA P, TX BC P
017336  021346                                  02700           SYNC                    ;
017340  177775                                  02800           -3                      ;
017342  022737  177320  017366                  02900   51$:    CMP     #177320,66$     ;IS THIS NON-EX MEMORY TEST
017350  001003                                  03000           BNE     15$             ;BR IF NO
017352  052777  000060  162002                  03100           BIS     #BIT5+BIT4,@DVSCR       ;SET "EA" BITS
017360  004537  021226                          03200   15$:    PERFORM ,SETREG         ;
017364     004     005                          03300           .BYTE   004,005         ;RX BA, RX BC
017366  000000                                  03400   66$:    0                       ;BA
017370  000000                                  03500   67$:    0                       ;BC
017372  022737  177266  017416                  03600           CMP     #(177320-32),68$        ;IS THIS NON-EX MEMORY TEST
017400  001003                                  03700           BNE     16$             ;BR IF NO
017402  052777  000060  161752                  03800           BIS     #BIT5+BIT4,@DVSCR       ;SET "EA" BITS
017410  004537  021026                          03900   16$:    PERFORM ,SETREG         ;
017414     011     010                          04000           .BYTE   011,010         ;RX CNTRL TAB, TX CNTRL TAB
017416  000000                                  04100   68$:    0                       ;
017420  022350                                  04200           TXTAB                   ;
017422  004537  021026                          04300           PERFORM ,SETREG         ;
017426     013     012                          04400           .BYTE   013,012         ;LINE STATE, LINE PROTOCOL PARAM REG
017430  000004                                  04500           BIT2                    ;TX GO
017432  000001                                  04600           BIT0                    ;IDLE MARK
017434  032737  004000  001236                                  BIT     #ASYNC.STAT     ;*IS THIS ASYNC LINE CARD?
017442  001412                                  BEQ     60$             ;*BR IF NO.
017444  004537  021072                          PERFORM ,LOAD.MODE      ;*LOAD PARAMETERS.
017450  020000                                  BIT13                   ;*RECEIVER ENABLE
017452  004537  021072                          PERFORM ,LOAD.MODE      ;*
017456  015000                                  (BIT12+BIT11)+BIT9      ;*8 BITS PER CHAR
017460  004537  021072                          PERFORM ,LOAD.MODE      ;*SET BAUD RATE REGISTER.
```

```
                                                                    (BIT14+BIT13+BIT12)+BIT10
                                                                                        ;#9600 BAUD.
                                                        BR      61$                 ;#CONTINUE TEST.
                                        04800   60$:    PERFORM .LOAD.MODE          ;LOAD
                                        04900           BIT13+BIT12+BIT11           ;MODE+RX ENABLE
                                        05000   61$:    MOV     #32,TXBAP           ;SET DATA CHAR
                                        05100           CLR     TXTAB+32            ;SET CNTRL BYTE TO ZERO
                                        05200           PERFORM .SETSYNC            ;ADJUST SYNC CHAR'S
                                        05300   2$:     INC     @DVSCR              ;SET UCPU GO
                                        05400           CLR     R5                  ;
                                        05500   3$:     TSTB    @DVSCR              ;DVSCR07=1?
                                        05600           BMI     4$                  ;BR IF YES
                                        05700           DELAY                       ;WASTE TIME
                                        05800           INC     R5                  ;DELAY
                                        05900           BNE     3$                  ;
                                        06000           MOVB    #13,@DVSRSH         ;SEL LINE STATE.
                                        06100           BIT     #BIT0,@DVSRA        ;IS RX ACTIVE SET?
                                        06200           BNE     .+4                 ;BR IF RX ACTIVE SET.
                                        06300           HLT     0                   ;RECEIVER ACTIVE IS NOT SET!
                                        06400           TSTB    @DVSCR              ;RE-VERIFY DVSCR07.
                                        06500           BMI     .+4                 ;BR IF BIT7 IS SET.
                                        06600           HLT     0                   ;DVSCR07 NEVER SET (=1)!
                                        06700   4$:     MOV     @DVSCR,R4           ;READ DVSCR
                                        06800           MOV     #BIT15+BIT7+BIT0,R5 ;
                                        06900           CMP     R5,R4               ;NPR STATUS ENTRY, RX INTR, UCPU GO
                                        07000           BEQ     5$                  ;BR IF OK
                                        07100           HLT     2                   ;DVSCR INCORRECT
                                        07200   5$:     MOV     @DVRIC,R4           ;GET DVRIC
                                        07300           MOV     DVRIC,R1            ;SET POINTER
                                        07400           MOV     R0,R5               ;GET LINE NUMBER
                                        07500           SWAB    R5                  ;PLACE IN HIGH BYTE
                                        07600           BIS     1155$,R5            ;SET OTHER BITS EXPECTED
                                        07700           CMP     R5,R4               ;DVRIC OK?
                                        07800           BEQ     6$                  ;
                                        07900           HLT     2                   ;DVRIC INCORRECT
                                        08000   6$:     MOV     @DVNSR,R4           ;READ DVNSR
                                        08100           MOV     R0,R5               ;LOAD LINE NO.
                                        08200           BIS     #BIT15+BIT8,R5      ;SILO ENTRY, PRINCIPLE BC=0
                                        08300           CMP     R5,R4               ;DVNSR OK
                                        08400           BEQ     7$                  ;
                                        08500           MOV     DVNSR,R1            ;
                                        08600           HLT     2                   ;DVNSR INCORRECT
                                        08700   7$:     MOV     R0,@DVSRS           ;LOAD LINE NO.
                                        08800           MOV     @DVSRA,R4           ;READ TX BA?
                                        08900           MOV     #TXBAP+1,R5         ;SET EXPECTED
                                        09000           CLR     R1                  ;SET SEC REG POINTER
                                        09100           CMP     R5,R4               ;OK?
                                        09200           BEQ     8$                  ;
                                        09300           HLT     3                   ;TX BA P INCORRECT
                                        09400   8$:     INCB    @DVSRSH             ;SEL TX BC?
                                        09500           INC     R1                  ;SET SEC REG POINTER
                                        09600           MOV     @DVSRA,R4           ;READ IT
                                        09700           BEQ     9$                  ;BR IF IT=0
                                        09800           CLR     R5                  ;
                                        09900           HLT     3                   ;TX BC P NOT=0
                                        10000   9$:     MOVB    #4,@DVSRSH          ;RX BA
```

```
 3465  017730  112701  000004            10100          MOVB    #4,R1           ;SET SEC REG POINTER
 3466  017734  017704  161436            10200          MOV     @DVSRA,R4       ;READ IT
 3467  017740  013705  025544            10300          MOV     EXPRBA,R5       ;GET EXPECTED BA
 3468  017744  020504                    10400          CMP     R5,R4           ;OK
 3469  017746  001401                    10500          BEQ     10$             ;
 3470  017750  104003                    10600          HLT     3               ;RX BA INCORRECT
 3471  017752  105277  161416            10700    10$:  INCB    @DVSRSH         ;RX BC
 3472  017756  005201                    10800          INC     R1              ;SET SEC REG POINTER
 3473  017760  017704  161412            10900          MOV     @DVSRA,R4       ;
 3474  017764  013705  025546            11000          MOV     EXPRWC,R5       ;SET EXPECTED BC
 3475  017770  020504                    11100          CMP     R5,R4           ;
 3476  017772  001401                    11200          BEQ     11$             ;RX BC INCORRECT
 3477  017774  104003                    11300          HLT     3               ;LINE STATE
 3478  017776  112777  000013  161370    11400    11$:  MOVB    #13,@DVSRSH     ;
 3479  020004  012701  000013            11500          MOV     #13,R1          ;SET SEC REG POINTER
 3480  020010  017704  161366            11600          MOV     @DVSRA,R4       ;
 3481  020014  012705  000201            11700          MOV     #BIT7+BIT0,R5   ;USE SEC. TABLE, RX ACTIVE
 3482  020020  020504                    11800          CMP     R5,R4           ;
 3483  020022  001401                    11900          BEQ     12$             ;
 3484  020024  104003                    12000          HLT     3               ;LINE STATE INCORRECT
 3485  020026  005200                    12100    12$:  INC     R0              ;UPDATE LINE NO.
 3486  020030  005337  020046            12200          DEC     111$            ;4 LINE GROUP DONE?
 3487  020034  001402                    12300          BEQ     100$            ;BR IF YES
 3488  020036  000137  017276            12400          JMP     64$             ;DO MORE
 3489  020042  012605                    12500    100$: MOV     (SP)+,R5        ;RESTORE R5
 3490  020044  000205                    12600          RTS     R5              ;EXIT
 3491  020046  000000                    12700    111$: 0
 3492  020050  000000                    12800    112$: 0                       ;DATA TO CORE TABLE
 3493  020052  000000                    12900    115$: 0                       ;DVRIC
 3494                                                  ;*SUBROUTINE.
 3495                                                  ;*SUBROUTINE USED BY PREVIOUS TEST.
 3496                                                  ;*
 3497  020054                                  _INT1:
 3498
 3499  020054  012500                                MOV     (R5)+,R0        ;GET FIRST LINE IN 4 LINE GROUP
 3500  020056  010037  020314                        MOV     R0,73$          ;SAVE LINE NUMBER
 3501  020062  010546                                MOV     R5,-(SP)        ;SAVE R5 ON STACK
 3502  020064  052700  100400                        BIS     #BIT15+BIT8,R0  ;SILO ENTRY + TX PRINCIPLE BC=0
 3503  020070  012703  000004                        MOV     #4,R3           ;4 LINE GROUP
 3504  020074  012702  021242            64$:  MOV     #REGBUF,R2      ;SET SOFTWARE BUFFER
 3505  020100  012701  000020                        MOV     #16.,R1         ;SET FOR 16 ENTRIES
 3506  020104  012704  000001                        MOV     #1,R4           ;
 3507  020110  042700  001000            65$:  BIC     #BIT9,R0        ;MAKE SURE R0 ISN'T SEC: BC=0.
 3508  020114  020022                                CMP     R0,(R2)+        ;IS ENTRY IN SOFTWARE TABLE?
 3509  020116  001403                                BEQ     66$             ;BR IF YES
 3510  020120  005301                                DEC     R1              ;ALL ENTRIES DONE?
 3511  020122  001372                                BNE     65$             ;BR IF NO
 3512  020124  104000                                HLT     0               ;ENTRY (R0) WAS NOT IN TABLE
 3513  020126  005701                        66$:  TST     R1              ;HAS COUNT GONE TO ZERO?
 3514  020130  001403                                BEQ     .+10            ;BR IF YES
 3515  020132  012762  177777  177776              MOV     #-1,-2(R2)      ;RUB OUT SILO ENTRY SO IT ISN'T FOUND AGAIN
 3516  020140  012701  000020                        MOV     #16.,R1         ;RESET COUNT TO 16.
 3517  020144  012702  021242                        MOV     #REGBUF,R2      ;SET SOFTWARE BUFFER POINTER
 3518  020150  005304                                DEC     R4              ;
 3519  020152  001356                                BNE     65$             ;
 3520  020154  005200                                INC     R0              ;
```

E06


```
 3521  020156  005303                        DEC     R3              ;
 3522  020160  001345                        BNE     64$             ;
 3523  020162  012703  000004                MOV     #4,R3           ;SET FOR 4 LINES
 3524  020166  013720  020314                MOV     73$,R0          ;RESTORE LINE NUMBER
 3525  020172  010046  161174        67$:    MOV     R0,@DVSRS       ;LOAD LINE NUMBER
 3526  020176  105077  161172                CLRB    @DVSRSH         ;SET TX BA PRINCIPLE
 3527  020202  005001                        CLR     R1              ;SET SEC REG POINTER
 3528  020204  017704  161166                MOV     @DVSRA,R4       ;READ IT
 3529  020210  012705  021350                MOV     #TXBAP,R5       ;SET BASE BUS ADDRESS
 3530  020214  060005                        ADD     R0,R5           ;ADD LINE NO. TO BA
 3531  020216  062705  000030                ADD     #30,R5          ;ADD ADDITIONAL 30(8  FOR XFR
 3532  020222  020504                        CMP     R5,R4           ;DID BA INCREMENT CORRECTLY?
 3533  020224  001401                        BEQ     68$             ;BR IF YES
 3534  020226  104003                        HLT     3               ;TX BA NOT CORRECT
 3535  020230  005005        68$:    CLR     R5              ;EXPECT BC TO BE=0
 3536  020232  105277  161136                INCB    @DVSRSH         ;SET TX P BC
 3537  020236  005201                        INC     R1              ;SET SEC REG POINTER
 3538  020240  017704  161132                MOV     @DVSRA,R4       ;READ BC
 3539  020244  001401                        BEQ     69$             ;BR IF BC=0
 3540  020246  104003                        HLT     3               ;TX P BC NOT=0
 3541  020250  112777  000013  161116        MOVB    #13,@DVSRSH     ;SEL LINE STATE
 3542  020256  012701  000013                MOV     #13,R1          ;SET SEC REG POINTER
 3543  020262  017704  161110                MOV     @DVSRA,R4       ;READ
 3544  020266  012705  000200                MOV     #BIT7,R5        ;EXPECT "USE SECONDARY TABLES"
 3545  020272  020504                        CMP     R5,R4           ;LINE STATE OK?
 3546  020274  001401                        BEQ     70$             ;BR IF YES
 3547  020276  104003                        HLT     3               ;LINE STATE INCORRECT
 3548  020300                        70$:
 3549  020300  005202                        INC     R2              ;
 3550  020302  005200                        INC     R0              ;UPDATE LINE POINTER
 3551  020304  005303                        DEC     R3              ;ALL LINES DONE?
 3552  020306  001331                        BNE     67$             ;BR IF NO
 3553  020310  012605                        MOV     (SP)+,R5        ;RESTORE R5
 3554  020312  000205                        RTS     R5              ;EXIT
 3555  020314  000001        73$:    .BLKW 1
 3556                                        ;*SUBROUTINE.
 3557                                        ;*SUBROUTINE USED BY PREVIOUS TEST.
 3558                                        ;*
 3559  020316                        LINTE:
 3560
 3561  020316  012500                        MOV     (R5)+,R0        ;GET FIRST LINE IN 4 LINE GROUP
 3562  020320  010037  020644                MOV     R0,73$          ;SAVE LINE NUMBER
 3563  020324  010546                        MOV     R5,-(SP)        ;SAVE R5 ON STACK
 3564  020326  052700  100400                BIS     #BIT15+BIT8,R0  ;SILO ENTRY + TX PRINCIPLE BC=0
 3565  020332  012703  000004                MOV     #4,R3           ;4 LINE GROUP
 3566  020336  012702  021242        64$:    MOV     #REGBUF,R2      ;SET SOFTWARE BUFFER
 3567  020342  012701  000040                MOV     #32.,R1         ;SET FOR 32 ENTRIES
 3568  020346  012704  000002                MOV     #2,R4
 3569  020352  042700  001000        65$:    BIC     #BIT9,R0        ;MAKE SURE R0 ISN'T SEC. BC=0.
 3570  020356  020022                        CMP     R0,(R2)+        ;IS ENTRY IN SOFTWARE TABLE?
 3571  020360  001410                        BEQ     66$.            ;BR IF YES
 3572  020362  052700  001000                BIS     #BIT9,R0        ;SET SEC BC=0
 3573  020366  020062  177776                CMP     R0,-2(R2)       ;IS ENTRY IN TABLE?
 3574  020372  001403                        BEQ     66$             ;BR IF YES
 3575  020374  005301                        DEC     R1              ;ALL ENTRIES DONE?
 3576  020376  001365                        BNE     65$             ;BR IF NO
```

```
3577  020400  104000                          HLT     0               ;ENTRY (R0) WAS NOT IN TABLE
3578  020402  005701                  66$:    TST     R1              ;HAS COUNT GONE TO ZERO?
3579  020404  001403                          BEQ     .+10            ;BR IF YES
3580  020406  012762  177777  177776          MOV     #-1,-2(R2)      ;RUB OUT SILO ENTRY SO IT ISN'T FOUND AGAIN
3581  020414  012701  000040                  MOV     #32.,R1         ;RESET COUNT TO 32.
3582  020420  012702  021242                  MOV     #REGBUF,R2      ;SET SOFTWARE BUFFER POINTER
3583  020424  005304                          DEC     R4              ;
3584  020426  001351                          BNE     65$             ;
3585  020430  005200                          INC     R0              ;
3586  020432  005303                          DEC     R3              ;
3587  020434  001340                          BNE     64$             ;
3588  020436  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINES
3589  020442  013700  020644                  MOV     73$,R0          ;RESTORE LINE NUMBER
3590  020446  010077  160720          67$:    MOV     R0,@DVSRS       ;LOAD LINE NUMBER
3591  020452  105077  160716                  CLRB    @DVSRSH         ;SET TX BA PRINCIPLE
3592  020456  005001                          CLR     R1              ;SET SEC REG POINTER
3593  020460  017704  160712                  MOV     @DVSRA,R4       ;READ IT
3594  020464  012705  021350                  MOV     #TXBAP,R5       ;SET BASE BUS ADDRESS
3595  020470  060005                          ADD     R0,R5           ;ADD LINE NO. TO BA
3596  020472  062705  000030                  ADD     #30,R5          ;ADD ADDITIONAL 30(8) FOR XFR
3597  020476  020504                          CMP     R5,R4           ;DID BA INCREMENT CORRECTLY?
3598  020500  001401                          BEQ     68$             ;BR IF YES
3599  020502  104003                          HLT     3               ;TX BA NOT CORRECT
3600  020504  005005                  68$:    CLR     R5              ;EXPECT BC TO BE=0
3601  020506  105277  160662                  INCB    @DVSRSH         ;SET TX P BC
3602  020512  005201                          INC     R1              ;SET SEC REG POINTER
3603  020514  017704  160656                  MOV     @DVSRA,R4       ;READ BC
3604  020520  001401                          BEQ     69$             ;BR IF BC=0
3605  020522  104003                          HLT     3               ;TX P BC NOT=0
3606  020524  112777  000013  160642  69$:    MOVB    #13,@DVSRSH     ;SEL LINE STATE
3607  020532  012701  000013                  MOV     #13,R1          ;SET SEC REG POINTER
3608  020536  017704  160634                  MOV     @DVSRA,R4       ;READ
3609  020542  005005                          CLR     R5              ;EXPECT LINE STATE=0
3610  020544  020504                          CMP     R5,R4           ;LINE STATE OK?
3611  020546  001401                          BEQ     70$             ;BR IF YES
3612  020550  104003                          HLT     3               ;LINE STATE INCORRECT
3613  020552                          70$:
3614  020552  112777  000002  160614          MOVB    #002,@DVSRSH    ;SEL TX ALTERNATE BA
3615  020560  017704  160612                  MOV     @DVSRA,R4       ;READ
3616  020564  012705  021750                  MOV     #TXBAS,R5       ;SET BASE ADDRESS
3617  020570  060005                          ADD     R0,R5           ;ADD LINE NO.
3618  020572  062705  000030                  ADD     #30,R5          ;+30 CHAR XMITTED
3619  020576  117701  160572                  MOVB    @DVSRSH,R1      ;SET SEC REG POINTER
3620  020602  020504                          CMP     R5,R4           ;TX BA (ALT) OK?
3621  020604  001401                          BEQ     71$             ;BR IF YES
3622  020606  104003                          HLT     3               ;TX BA (ALT) WRONG!
3623  020610  105277  160560          71$:    INCB    @DVSRSH         ;SEL TX BC (ALT)
3624  020614  005201                          INC     R1              ;SET SEC REG POINTER
3625  020616  017704  160554                  MOV     @DVSRA,R4       ;IS IT=0?
3626  020622  001402                          BEQ     72$             ;BR IF YES
3627  020624  005005                          CLR     R5              ;SET EXPECTED TO 0
3628  020626  104003                          HLT     3               ;TX BC ALTERNATE NOT=0
3629  020630                          72$:
3630  020630  005202                          INC     R2              ;
3631  020632  005200                          INC     R0              ;UPDATE LINE POINTER
3632  020634  005303                          DEC     R3              ;ALL LINES DONE?
```

```
 3633  020636  001303                                BNE     67$             ;BR IF NO
 3634  020640  012605                                MOV     (SP)+,R5        ;RESTORE R5
 3635  020642  000205                                RTS     R5              ;EXIT
 3636  020644  000001                        73$:    .BLKW 1
 3637  020646  010046                        SIMSCC: MOV     R0,-(SP)
 3638  020650  010146                                MOV     R1,-(SP)
 3639  020652  010246                                MOV     R2,-(SP)
 3640  020654  012537  001246                        MOV     (R5)+,TEMP1
 3641  020660  012537  001250                        MOV     (R5)+,TEMP2
 3642  020664  012537  001252                        MOV     (R5)+,TEMP3
 3643  020670  005037  021022                1$:     CLR     BCCFBK
 3644  020674  013700  001252                        MOV     TEMP3,R0
 3645  020700  006037  001250                        ROR     TEMP2
 3646  020704  005500                                ADC     R0
 3647  020706  032700  000001                        BIT     #BIT0,R0
 3648  020712  001402                                BEQ     2$
 3649  020714  005137  021022                        COM     BCCFBK
 3650  020720  013700  021020                3$:     MOV     XPOLY,R0
 3651  020724  005100                                COM     R0
 3652  020726  040037  021022                        BIC     R0,BCCFBK
 3653  020732  000241                                CLC
 3654  020734  006037  001252                        ROR     TEMP3
 3655  020740  013700  021022                        MOV     BCCFBK,R0
 3656  020744  013701  001252                        MOV     TEMP3,R1
 3657  020750  010102                                MOV     R1,R2
 3658  020752  040100                                BIC     R1,R0
 3659  020754  043702  021022                        BIC     BCCFBK,R2
 3660  020760  050200                                BIS     R2,R0
 3661  020762  043737  021020  001252                BIC     XPOLY,TEMP3
 3662  020770  050037  001252                        BIS     R0,TEMP3
 3663  020774  005337  001246                        DEC     TEMP1
 3664  021000  001333                                BNE     1$
 3665  021002  013737  001252  021024                MOV     TEMP3,CALBCC
 3666  021010  012602                                MOV     (SP)+,R2
 3667  021012  012601                                MOV     (SP)+,R1
 3668  021014  012600                                MOV     (SP)+,R0
 3669  021016  000205                                RTS     R5
 3670  021020  000000                        XPOLY:  0
 3671  021022  000000                        BCCFBK: 0
 3672  021024  000000                        CALBCC: 0
 3673                  000200                LRC8=200
 3674                  120001                CRC16=120001
 3675                  102010                CRC.CCITT=102010
 3676
 3677
 3678  021026  010046                13300   SETREG: MOV     R0,-(SP)
 3679  021030  010146                13400           MOV     R1,-(SP)
 3680  021032  112500                13500           MOVB    (R5)+,R0
 3681  021034  112501                13600           MOVB    (R5)+,R1
 3682  021036  110077  160332        13700           MOVB    R0,@DVSRSH
 3683  021042  012577  160330        13800           MOV     (R5)+,@DVSRA
 3684  021046  042777  000060  160306 13900          BIC     #BIT5+BIT4,@DVSCR
 3685  021054  110177  160314        14000           MOVB    R1,@DVSRSH
 3686  021060  012577  160312        14100           MOV     (R5)+,@DVSRA
 3687  021064  012601                14200           MOV     (SP)+,R1
 3688  021066  012600                14300           MOV     (SP)+,R0
```

```
 3689  021070  000205                          14400           EXIT
 3690                                          14500
 3691  021072                                  14600   LOAD.MODE:
 3692  021072  012577  160272                  14700           MOV     (R5)+,@DVLCR
 3693  021076  052777  100000  160264          14800           BIS     #BIT15,@DVLCR
 3694  021104  010046                          14900           MOV     R0,-(SP)
 3695  021106  005000                          15000           CLR     R0
 3696  021110  005777  160254                  15100   1$:     TST     @DVLCR
 3697  021114  100004                          15200           BPL     2$
 3698  021116  104414                          15300           DELAY
 3699  021120  005200                          15305           INC     R0
 3700  021122  001372                          15400           BNE     1$
 3701  021124  104000                          15500           HLT     0                       ;BIT 15 FAILED TO CLEAR
 3702  021126  012600                          15600   2$:     MOV     (SP)+,R0
 3703  021130  000205                          15700           EXIT
 3704
 3705  021132                                          SETSCAN:
 3706  021132  010346                                          MOV     R3,-(SP)
 3707  021134  052777  000010  160220                        BIS     #BIT3,@DVSCR
 3708  021142  012503                                        MOV     (R5)+,R3
 3709  021144  001414                                        BEQ     2$
 3710  021146  012777  050102  160224          1$:     MOV     #BIT14+BIT12+BIT6+BIT1,@DVSFR
 3711  021154  104415                                        RCMCLK
 3712  021156  005201                                        INC     R1
 3713  021160  012777  050102  160212                        MOV     #BIT14+BIT12+BIT6+BIT1,@DVSFR
 3714  021166  104415                                        RCMCLK
 3715  021170  005201                                        INC     R1
 3716  021172  005303                                        DEC     R3
 3717  021174  001364                                        BNE     1$
 3718  021176  012603                                2$:     MOV     (SP)+,R3
 3719  021200  010100                                        MOV     R1,R0
 3720  021202  000241                                        CLC
 3721  021204  006000                                        ROR     R0
 3722  021206  000205                                        EXIT
 3723  021210                                          SETSYNC:
 3724  021210  113737  001236  021346                        MOVB    STAT.SYNC       ;SET SYNC FOR THIS LINE.
 3725  021216  113737  021346  021347                        MOVB    SYNC,SYNC+1     ;PLACE SYNC IN HIGH BYTE
 3726  021224  032737  010000  001236                        BIT     #TWOSYN,STAT    ;ONE SYNC OR TWO?
 3727  021232  001402                                        BEQ     1$              ;BR IF JUMPERED FOR TWO.
 3728  021234  105037  021346                              CLRB    SYNC            ;SET FIRST SYNC TO NON-SYNC
 3729  021240  000205                                1$:     EXIT
 3730                                  15000
 3731  021242  000042                 16100   REGBUF: .BLKW   34.
 3732  021346  000001                 16200   SYNC:   .BLKW   1
 3733  021350  000400                 16300   TXBAP:  .BLKB   400
 3734  021750  000400                 16400   TXBAS:  .BLKB   400
 3735  022350  000400                 16500   TXTAB:  .BLKB   400
 3736  022750  000400                 16600   RXBA:   .BLKB   400
 3737  023350  000400                 16700   RXTAB:  .BLKB   400
 3738  023750                         16800   ROMDATA:
```

```
 3739                                  17500                    ;:IDLE LOOP
 3740                                  17600                    ;
 3741  023750  050102                  17700  ↑B<0101000001000010>  ;ILOOP S/C     6,2      ;INCREMENT SCANNER
 3742  023752  030124                  17800  ↑B<0011000001010100>  ;       XFR     5,4      ;MOVE MASTER SCAN TO RAM ADDRESS
 3743  023754  001344                  17900  ↑B<0000001011100100>  ;       BRA     2,TSERV  ;TEST FOR TRANSMIT FLAG WAITING.
 3744  023756  002012                  18000  ↑B<0000010000001010>  ;       BRA     4,RSERV  ;TEST FOR RECEIVER FLAG WAITING.
 3745  023760  001503                  18100  ↑B<0000001101000011>  ;       BRA     3,SSERV  ;TEST FOR RECEIVED CHARACTER WA
 3746  023762  020013                  19200  ↑B<0010000000001011>  ;ILOP2 RAM     0,0,13   ;OBTAIN LINE STATE
 3747  023764  070456                  18300  ↑B<0111000100101110>  ;       BRB     1,RSYNC  ;TEST RAM OUTPUT 01 (RESYNC), IF
 3748  023766  071072                  18400  ↑B<0111001000111010>  ;       BRB     2,TMARK  ;TEST RAM OUTPUT 02 (XMIT GO), I
 3749  023770  002451                  18500  ↑B<0000010100101001>  ;ILOP5 BRA     5,ISERV  ;TEST FOR CHARACTER DISPATCH PRO
 3750  023772  000400                  19600  ↑B<0000000100000000>  ;       BRA     1,ILOOP  ;TEST FOR SURE TRUE, IF YES BRAN
 3751                                  18700                    ;
 3752                                  19800                    ;
 3753                                  18900                    ;
 3754                                  19000                    ;:RECEIVE FLAG SERVICE  (LINE STATE IS IN RAM OUTPUT)
 3755                                  19100                    ;
 3756  023774  020013                  19200  ↑B<0010000000001011>  ;RSERV RAM     0,0,13   ;OBTAIN LINE STATE
 3757  023776  070036                  19300  ↑B<0111000000011110>  ;       BRB     0,TESTX  ;TEST RAM OUTPUT 00 (RECEIVER AC
 3758  024000  076417                  19400  ↑B<0111110100001111>  ;       BRB     15,S/ACT         ;TEST MATCH DETECT, IF Y
 3759  024002  050106                  19500  ↑B<0101000001000110>  ;       S/C     6,6      ;SET RESYNC PULSE,
 3760  024004  000400                  19600  ↑B<0000000100000000>  ;       BRA     1,ILOOP  ;TEST FOR SURE TRUE, IF YES BRAN
 3761  024006  020012                  19700  ↑B<0010000000001010>  ;S/ACT RAM     0,0,12   ;OBTAIN DLE/PROTOCOL
 3762  024010  070430                  19800  ↑B<0111000100011000>  ;       BRB     1,SACT2  ;TEST RAM 01 (STRIP LEADING SYNC
 3763  024012  020013                  19900  ↑B<0010000000001011>  ;SACT1 RAM     0,0,13   ;OBTAIN LINE STATE
 3764  024014  050047                  20000  ↑B<0101000000100111>  ;       S/C     5,7      ;SET RAM OUTPUT 00 (RECEIVER ACT
 3765  024016  020673                  20100  ↑B<0010000110111011>  ;       RAM     1,13,13  ;WRITE NEW LINE STATE
 3766  024020  077021                  20200  ↑B<0111111000010001>  ;       BRB     16,SACT1         ;TEST FOR WRITE INHIBIT.
 3767  024022  050023                  20300  ↑B<0101000000010011>  ;CLRRF S/C     4,3      ;SET RECEIVE DATA ENABLE
 3768  024024  050022                  20400  ↑B<0101000000010010>  ;       S/C     4,2      ;CLEAR RECEIVE DATA ENABLE AND D
 3769  024026  000400                  20500  ↑B<0000000100000000>  ;       BRA     1,ILOOP  ;TEST FOR SURE TRUE, IF YES BRAN
 3770  024030  020013                  20600  ↑B<0010000000001011>  ;SACT2 RAM     0,0,13   ;OBTAIN LINE STATE
 3771  024032  050047                  20700  ↑B<0101000000100111>  ;       S/C     5,7      ;SET RAM OUTPUT 00 (RECEIVER ACT
 3772  024034  050206                  20800  ↑B<0101000010000110>  ;       S/C     7,6      ;SET RAM OUTPUT 06 (STRIP SYNC O
 3773  024036  020673                  20900  ↑B<0010000110111011>  ;       RAM     1,13,13  ;WRITE NEW LINE STATE
 3774  024040  077030                  21000  ↑B<0111111000011000>  ;       BRB     16,SACT2         ;TEST FOR WRITE INHIBIT.
 3775  024042  000425                  21100  ↑B<0000000100010101>  ;       BRA     1,CLRRF  ;TEST FOR SURE TRUE, IF YES BRAN
 3776                                  21200                    ;
 3777                                  21300                    ;
 3778  024044  020013                  21400  ↑B<0010000000001011>  ;TESTX RAM     0,0,13   ;OBTAIN LINE STATE
 3779  024046  073041                  21500  ↑B<0111011000100001>  ;       BRB     6,TMD    ;TEST RAM OUTPUT 06 (STRIP SYNC
 3780  024050  000445                  21600  ↑B<0000000100100101>  ;       BRA     1,S/RDE  ;TEST FOR SURE TRUE, IF YES BRAN
 3781  024052  076425                  21700  ↑B<0111110100010101>  ;TMD   BRB     15,CLRRF         ;TEST MATCH DETECT, IF Y
 3782  024054  050202                  21800  ↑B<0101000010000010>  ;       S/C     7,2      ;CLEAR RAM OUTPUT 06 (STRIP SYNC
 3783  024056  020673                  21900  ↑B<0010000110111011>  ;       RAM     1,13,13  ;WRITE NEW LINE STATE
 3784  024060  077012                  22000  ↑B<0111111000001010>  ;       BRB     16,RSERV         ;TEST FOR WRITE INHIBIT.
 3785  024062  050023                  22100  ↑B<0101000000010011>  ;S/RDE S/C     4,3      ;SET RECEIVE DATA ENABLE
 3786  024064  050021                  22200  ↑B<0101000000010001>  ;       S/C     4,1      ;SET SILO IN
 3787  024066  050022                  22300  ↑B<0101000000010010>  ;       S/C     4,2      ;CLEAR RECEIVE DATA ENABLE AND D
 3788  024070  000400                  22400  ↑B<0000000100000000>  ;       BRA     1,ILOOP  ;TEST FOR SURE TRUE, IF YES BRAN
 3789                                  22500                    ;
 3790                                  22600                    ;
 3791                                  22700                    ;;RECEIVE INTERRUPT RESPONSE SERVICE
 3792                                  22800                    ;
 3793  024072  030301                  22900  ↑B<0011000011000001>  ;ISERV XFR     ,14,1    ;MOVE SILO OUT TO A REGISTER
 3794  024074  010037                  23000  ↑B<0001000000011111>  ;       ALU     37       ;LET ALU RESULT = A REGISTER
```

```
 3795  024076  030144          23100  ↑B<0011000001100100>      ;       XFR     6,4     ;MOVE ALU RESULT 08-11 TO RAM AD
 3796  024100  050016          23200  ↑B<0101000000001110>      ;       S/C     3,5     ;CLEAR SCR08
 3797  024102  000542          23300  ↑B<0000000101100010>      ;       BRA     1,CTEST ;TEST FOR SURE TRUE, IF YES BRAN
 3798                          23400                            ;
 3799                          23500                            ;
 3800                          23600                            ;
 3801                          23700                            ;:RESYNCHRONIZE (MASTER SCAN IS IN RAM AR, LINE STATE I
 3802                          23800                            ;
 3803  024104  020013          23900  ↑B<0010000000001011>      ;RSYNC  RAM     0,0,13  ;OBTAIN LINE STATE
 3804  024106  050043          24000  ↑B<0101000000100011>      ;       S/C     5,3     ;CLEAR RAM OUTPUT 00 (RECEIVER A
 3805  024110  050041          24100  ↑B<0101000000100001>      ;       S/C     5,1     ;CLEAR RAM OUTPUT 01 (RESYNCHRON
 3806  024112  020673          24200  ↑B<0010000110111011>      ;       RAM     1,13,13 ;WRITE NEW LINE STATE
 3807  024114  077056          24300  ↑B<0111111000101110>      ;       BRB     16,RSYNC        ;TEST FOR WRITE INHIBIT,
 3808  024116  020016          24400  ↑B<0010000000001110>      ;PSI    RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 3809  024120  050204          24500  ↑B<0101000010000100>      ;       S/C     7,4     ;SET RAM OUTPUT 07
 3810  024122  020676          24600  ↑B<0010000110111110>      ;       RAM     1,13,16 ;WRITE NEW LINE PROTOCOL
 3811  024124  077063          24700  ↑B<0111111000110011>      ;       BRB     16,PSI  ;TEST FOR WRITE INHIBIT, IF YES
 3812  024126  050106          24800  ↑B<0101000001000110>      ;       S/C     6,6     ;SET RESYNC PULSE
 3813  024130  050021          24900  ↑B<0101000000010001>      ;       S/C     4,1     ;SET SILO IN
 3814  024132  000405          25000  ↑B<0000000100000101>      ;       BRA     1,ILOP2 ;TEST FOR SURE TRUE, IF YES BRAN
 3815                          25100                            ;
 3816                          25200                            ;
 3817                          25300                            ;
 3818                          25400                            ;
 3819                          25500                            ;:CLEAR TMARK
 3820                          25600                            ;
 3821  024134  050101          25700  ↑B<0101000001000001>      ;TMARK  S/C     6,1     ;CLEAR TMARK
 3822  024136  000410          25800  ↑B<0000000100001000>      ;       BRA     1,ILOP5 ;TEST FOR SURE TRUE, IF YES BRAN
 3823                          25900                            ;
 3824                          26000                            ;
 3825                          26100                            ;
 3826                          26200                            ;
 3827                          26300                            ;:TEST FOR RESYNC FLAG (LINE NUMBER IN RAM AR, CHARACTE
 3828                          26400                            ;
 3829  024140  000076          26500  ↑B<0000000000111110>      ;TFRF   BRA     0,CRAM7 ;TEST BIT IS OF ALU RESULT, IF Y
 3830  024142  000603          26600  ↑B<0000000110000011>      ;       BRA     1,DISC  ;TEST FOR SURE TRUE, IF YES BRAN
 3831  024144  020016          26700  ↑B<0010000000001110>      ;CRAM7  RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 3832  024146  050200          26800  ↑B<0101000010000000>      ;       S/C     7,0     ;CLEAR RAM OUTPUT 07 (RESYNCH FL
 3833  024150  020676          26900  ↑B<0010000110111110>      ;       RAM     1,13,16 ;WRITE NEW LINE PROTOCOL
 3834  024152  077076          27000  ↑B<0111111000111110>      ;       BRB     16,CRAM7        ;TEST FOR WRITE INHIBIT,
 3835  024154  000603          27100  ↑B<0000000110000011>      ;       BRA     1,DISC  ;TEST FOR SURE TRUE, IF YES BRAN
 3836                          27200                            ;
 3837                          27300                            ;
 3838                          27400                            ;:RECEIVED CHARACTER SILO SERVICE
 3839                          27500                            ;
 3840  024156  002451          27600  ↑B<0000010100101001>      ;SSERV  BRA     5,ISERV ;TEST FOR SCR08 (COULD HAVE SET
 3841  024160  030301          27700  ↑B<0011000011000001>      ;       XFR     14,1    ;MOVE SILO OUT TO A REGISTER
 3842  024162  010037          27800  ↑B<0001000000011111>      ;       ALU     37      ;LET ALU RESULT = A REGISTER
 3843  024164  030144          27900  ↑B<0011000001100100>      ;       XFR     6,4     ;MOVE ALU RESULT 08-11 TO RAM AD
 3844  024166  020013          28000  ↑B<0010000000001011>      ;       RAM     0,0,13  ;OBTAIN LINE STATE
 3845  024170  070603          28100  ↑B<0111000110000011>      ;       BRB     1,DISC  ;TEST RAM OUTPUT 01 (RESYNC), IF
 3846  024172  020016          28200  ↑B<0010000000001110>      ;       RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 3847  024174  073474          28300  ↑B<0111011100111100>      ;       BRB     7,TFRF  ;TEST RAM OUTPUT 07, IF YES BRAN
 3848  024176  104261          28400  ↑B<1000100010110001>      ;       BRA     10,POER ;TEST BITS  13, 12 OF ALU RESULT
 3849  024200  073272          28500  ↑B<0111011010111010>      ;       BRB     6,TBC2  ;TEST RAM OUTPUT 06, IF YES BRAN
 3850  024202  072645          28600  ↑B<0111010110100101>      ;       BRB     5,TBC1  ;TEST RAM OUTPUT 05, IF YES BRAN
```

```
 3851  024204  020005          28700  ↑B<0010000000000101>      ;        RAM     0,0,5   ;OBTAIN RECEIVER BYTE COUNT
 3852  024206  076214          28800  ↑B<0111110010001100>      ;        BRB     14,CRBCO        ;TEST RAM OUTPUT 0-14=0,
 3853  024210  020012          28900  ↑B<0010000000001010>      ;        RAM     0,0,12  ;OBTAIN TRANSMITTER DLE/LINE PRO
 3854  024212  072576          29000  ↑B<0111010101111110>      ;        BRB     5,CDCMR ;TEST RAM OUTPUT 05, IF YES BRAN
 3855  024214  020015          29100  ↑B<0010000000001101>      ;        RAM     0,0,15  ;OBTAIN RECEIVER MODE BITS
 3856  024216  030242          29200  ↑B<0011000010100010>      ;ZETA    XFR     ,12,2   ;MOVE RAM OUTPUT DATA TRANSLATED
 3857  024220  050027          29300  ↑B<0101000000010111>      ;        S/C     4,7     ;CLEAR ALU RESULT UPPER BYTE
 3858  024222  030361          29400  ↑B<0011000011110001>      ;        XFR     ,17,1   ;MOVE ALU RESULT TO A REGISTER
 3859  024224  010026          29500  ↑B<0001000000010110>      ;        ALU     26      ;LET ALU RESULTS = A PLUS B
 3860  024226  030362          29600  ↑B<0011000011110010>      ;        XFR     ,7,2    ;MOVE ALU RESULTS TO B REGISTER
 3861  024230  020011          29700  ↑B<0010000000001001>      ;        RAM     0,0,11  ;OBTAIN RECEIVER CONTROL TABLE B
 3862  024232  030261          29800  ↑B<0011000010110001>      ;        XFR     ,13,1   ;MOVE RAM OUTPUT DATA TO A REGIS
 3863  024234  010026          29900  ↑B<0001000000010110>      ;        ALU     26      ;LET ALU RESULTS = A PLUS B (EFF
 3864  024236  030363          30000  ↑B<0011000011110011>      ;        XFR     I,17,3  ;MOVE ALU RESULTS TO NPR ADDRESS
 3865  024240  040000          30100  ↑B<0100000000000000>      ;        NPR             ;DO NPR TO GET CONTROL BYTE
 3866  024242  074535          30200  ↑B<0111100101011101>      ;RBLS1   BRB     11,RBUS1        ;TEST REQUEST BUS, IF YE
 3867  024244  003332          30300  ↑B<0000011011011010>      ;        BRA     6,RNXMC ;TEST NXM, IF YES, BRANCH TO REC
 3868  024246  075327          30400  ↑B<0111101011010111>      ;        BRB     12,RMPEC        ;TEST MEM PAR ERR, IF YE
 3869  024250  020017          30500  ↑B<0010000000001111>      ;        RAM     0,0,17  ;OBTAIN CONTROL BYTE STORAGE REG
 3870  024252  020637          30600  ↑B<0010000110011111>      ;        RAM     1,11,17 ;MOVE DATI REGISTER TO RAM AND W
 3871                          30700                             ;
 3872                          30800                             ;
 3873                          30900                             ;
 3874                          31000                             ;
 3875                          31100                             ;;CONTROL BYTE TESTS BEGIN (CHARACTER IS IN SILO OUT, C
 3876                          31200                             ;
 3877  024254  020017          31300  ↑B<0010000000001111>      ;CTEST RAM       0,0,17  ;OBTAIN CONTROL BYTE STORAGE REG
 3878  024256  030262          31400  ↑B<0011000010110010>      ;        XFR     ,13,2   ;MOVE RAM OUTPUT TO B REGISTER
 3879  024260  010005          31500  ↑B<0001000000000101>      ;        ALU     5       ;LET ALU RESULT = B REGISTER
 3880  024262  020575          31600  ↑B<0010000101111101>      ;        RAM     1,7,15  ;MOVE ALU RESULTS TRANSLATED 5-7
 3881  024264  105253          31700  ↑B<1000101010101011>      ;        BRA     12,CBINT        ;TEST BIT 0 OF ALU RESUL
 3882  024266  006233          31800  ↑B<0000110010011011>      ;        BRA     14,EBCC ;TEST BIT 02 OF ALU RESULT, IF Y
 3883  024270  006605          31900  ↑B<0000110110000101>      ;EPSIL   BRA     15,RBCC ;TEST BIT 03 OF ALU RESULT, IF Y
 3884                          32000                             ;
 3885                          32100                             ;
 3886                          32200                             ;;RETURN FROM RECV BCC  (CONTROL BYTE IS STILL IN ALU R
 3887                          32300                             ;
 3888  024272  007203          32400  ↑B<0000111010000011>      ;RRBCC   BRA     16,DISC ;TEST BIT 4 OF ALU RESULT, IF YE
 3889  024274  020004          32500  ↑B<0010000000000100>      ;        RAM     0,0,4   ;OBTAIN RECEIVER CURRENT ADDRESS
 3890  024276  031663          32600  ↑B<0011001110110011>      ;        XFR     0,13,3  ;MOVE RAM OUTPUT TO NPR ADDRESS
 3891  024300  030305          32700  ↑B<0011000011000101>      ;        XFR     ,14,5   ;MOVE SILO OUT TO DATO REGISTER
 3892  024302  040000          32800  ↑B<0100000000000000>      ;        NPR             ;DO NPR TO STORE RECEIVED CHARAC
 3893  024304  074556          32900  ↑B<0111100101101110>      ;RBUS2   BRB     11,RBUS2        ;TEST REQUEST BUS, IF YE
 3894  024306  003340          33000  ↑B<0000011011100000>      ;        BRA     6,RNXM  ;TEST NXM, IF YES, BRANCH TO REC
 3895  024310  020005          33100  ↑B<0010000000000101>      ;ORBC    RAM     0,0,5   ;OBTAIN RECEIVER BYTE COUNT
 3896  024312  030261          33200  ↑B<0011000010110001>      ;        XFR     ,13,1   ;MOVE RAM OUTPUT TO REGISTER A
 3897  024314  010077          33300  ↑B<0001000000111111>      ;        ALU     77      ;LET ALU RESULTS = A+1
 3898  024316  020765          33400  ↑B<0010000111110101>      ;        RAM     1,17,5  ;MOVE ALU RESULTS TO RAM INPUT ;
 3899  024320  077160          33500  ↑B<0111111001110000>      ;        BRB     16,ORBC ;TEST FOR WRITE INHIBIT, IF YES
 3900  024322  020004          33600  ↑B<0010000000000100>      ;ORCA    RAM     0,0,4   ;OBTAIN RECEIVER CURRENT ADDRESS
 3901  024324  030261          33700  ↑B<0011000010110001>      ;        XFR     ,13,1   ;MOVE RAM OUTPUT DATA TO REGISTE
 3902  024326  010077          33800  ↑B<0001000000111111>      ;        ALU     77      ;LET ALU RESULTS = A+1
 3903  024330  020764          33900  ↑B<0010000111110100>      ;        RAM     1,17,4  ;MOVE ALU RESULTS TO RAM INPUT ;
 3904  024332  077165          34000  ↑B<0111111001110101>      ;        BRB     16,ORCA ;TEST FOR WRITE INHIBIT, IF YES
 3905  024334  020005          34100  ↑B<0010000000000101>      ;        RAM     0,0,5   ;OBTAIN RECEIVER BYTE COUNT
 3906  024336  076217          34200  ↑B<0111110010001111>      ;        BRB     14,NBCO ;TEST RAM OUTPUT 0-14=0, IF YES,
```

```
 3907  024340  050020          34300  !B<0101000000010000>     ;       S/C     4,0      ;SET SILO OUT
 3908  024342  000400          34400  !B<0000000100000000>     ;       BRA     1,ILOOP  ;TEST FOR SURE TRUE, IF YES BRAN
 3909                          34500                           ;
 3910                          34600                           ;
 3911                          34700                           ;;DDCMP RECEPTION
 3912                          34800                           ;
 3913  024344  020015          34900  !B<0010000000001101>     ;DDCMR RAM      0,0,15   ;OBTAIN RECEIVER MODE BITS
 3914  024346  076201          35000  !B<0111110010000001>     ;       BRB     14,DDCM2         ;TEST RAM OUTPUT 0-14=0,
 3915  024350  000523          35100  !B<0000000101010011>     ;       BRA     1,ZETA   ;TEST FOR SURE TRUE, IF YES BRAN
 3916  024352  010014          35200  !B<0001000000001100>     ;DDCM2 ALU      14       ;LET ALU RESULT = 0
 3917  024354  000605          35300  !B<0000000110000101>     ;       BRA     1,RBCC   ;TEST FOR SURE TRUE, IF YES BRAN
 3918                          35400                           ;
 3919                          35500                           ;
 3920                          35600                           ;;DISCARD RECEIVED CHARACTER
 3921                          35700                           ;
 3922  024356  050020          35800  !B<0101000000010000>     ;DISC   S/C     4,0      ;SET SILO OUT
 3923  024360  000400          35900  !B<0000000100000000>     ;       BRA     1,ILOOP  ;TEST FOR SURE TRUE, IF YES BRAN
 3924                          36000                           ;
 3925                          36100                           ;
 3926                          36200                           ;;CALCULATE RECV BCC (ASSUME RECEIVED CHARACTER IN SILO
 3927                          36300                           ;
 3928  024362  030301          36400  !B<0011000011000001>     ;RBCC   XFR     ,14,1    ;MOVE SILO OUT REGISTER TO A REG
 3929  024364  020007          36500  !B<0010000000000111>     ;       RAM     0,0,7    ;OBTAIN RECV BCC CALCULATED TO D
 3930  024366  030262          36600  !B<0011000010110010>     ;       XFR     ,13,2    ;MOVE RAM OUTPUT DATA TO B REGIS
 3931  024370  020012          36700  !B<0010000000001010>     ;       RAM     0,0,12   ;OBTAIN TRANSMITTER DLE/LINE PRO
 3932  024372  060000          36800  !B<0110000000000000>     ;       BCC              ;PERFORM SPECIFIED BCC CALCULATI
 3933  024374  020747          36900  !B<0010000111100111>     ;       RAM     1,16,7   ;MOVE BCC TO RAM INPUT AND WRITE
 3934  024376  000551          37000  !B<0000000101101001>     ;       BRA     1,RRBCC  ;TEST FOR SURE TRUE, IF YES, BRA
 3935                          37100                           ;
 3936                          37200                           ;
 3937                          37300                           ;;CHARACTER RECEIVED WHILE RECV BC=0
 3938                          37400                           ;
 3939  024400  030306          37500  !B<0011000011000110>     ;CRBCO  XFR     ,14,6    ;MOVE SILO OUT REGISTER TO RICR
 3940  024402  050010          37600  !B<0101000000001000>     ;       S/C     3,0      ;SET RICR 15 (TO INDICATE RECEPT
 3941  024404  100662          37700  !B<1000000110110010>     ;       BRA     1,CNACB  ;TEST FOR SURE TRUE, IF YES BRAN
 3942                          37800                           ;
 3943                          37900                           ;;NEXT CHARACTER WILL HAVE BC=0  (SILO OUT HAS BEEN SET
 3944                          38000                           ;
 3945  024406  030326          38100  !B<0011000011010110>     ;NBCO   XFR     ,15,6    ;MOVE NPR DATO REGISTER TO RICR
 3946  024410  050011          38200  !B<0101000000001001>     ;       S/C     3,1      ;SET RICR 14 (TO INDICATE RECEPT
 3947  024412  075623          38300  !B<0111101110010011>     ;       BRB     13,MCBCX         ;TEST RAM OUTPUT 15, IF
 3948  024414  100662          38400  !B<1000000110110010>     ;       BRA     1,CNACB  ;TEST FOR SURE TRUE, IF YES BRAN
 3949                          38500                           ;
 3950                          38600                           ;
 3951                          38700                           ;;MODE CHANGE AND BCC EXPECT
 3952                          38800                           ;
 3953  024416  020013          38900  !B<0010000000001011>     ;MCBCX  RAM     0,0,13   ;OBTAIN LINE STATE
 3954  024420  030262          39000  !B<0011000010110010>     ;       XFR     ,13,2    ;MOVE RAM OUTPUT TO B REGISTER
 3955  024422  030202          39100  !B<0011000010000010>     ;       XFR     ,10,2    ;MOVE B REGISTER 8-15 TO B REGIS
 3956  024424  010005          39200  !B<0001000000000101>     ;       ALU     5        ;LET ALU RESULT = B REGISTER
 3957  024426  020575          39300  !B<0010000101111101>     ;       RAM     1,7,15   ;MOVE ALU RESULTS TRANSLATED TO
 3958  024430  020777          39400  !B<0010000111111111>     ;       RAM     1,17,17  ;WRITE CONTROL BYTE STORAGE FROM
 3959  024432  006240          39500  !B<0000110010100000>     ;       BRA     14,EBCN  ;TEST ALU RESULT 02, IF YES BRAN
 3960  024434  100662          39600  !B<1000000110110010>     ;       BRA     1,CNACB  ;TEST FOR SURE TRUE, IF YES BRAN
 3961                          39700                           ;
 3962                          39800                           ;
```

```
 3963                                  39900                    ;;EXPECT BCC NEXT BECAUSE OF CONTROL BYTE
 3964                                  40000                    ;
 3965  024436  020016                  40100  ↑B<0010000000001110>  ;EBCC  RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 3966  024440  050205                  40200  ↑B<0101000010000101>  ;       S/C     7,5     ;SET RAM OUTPUT 05 (EXPECT BCC 1
 3967  024442  020676                  40300  ↑B<0010000110111110>  ;       RAM     1,13,16 ;WRITE LINE PROTOCOL FROM RAM OU
 3969  024444  077233                  40400  ↑B<0111111010011011>  ;       BRB     16,EBCC ;TEST FOR WRITE INHIBIT, IF YES
 3969  024446  000550                  40500  ↑B<0000000101101000>  ;       BRA     1,EPSIL ;TEST FOR SURE TRUE, IF YES BRAN
 3970                                  40600                    ;
 3971                                  40700                    ;
 3972                                  40800                    ;
 3973                                  40900                    ;;EXPECT BCC NEXT BECAUSE OF BC = 0
 3974                                  41000                    ;
 3975  024450  020016                  41100  ↑B<0010000000001110>  ;EBCN  RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 3976  024452  050205                  41200  ↑B<0101000010000101>  ;       S/C     7,5     ;SET RAM OUTPUT 05 (EXPECT BCC1
 3977  024454  020676                  41300  ↑B<0010000110111110>  ;       RAM     1,13,16 ;WRITE LINE PROTOCOL FROM RAM OU
 3978  024456  077240                  41400  ↑B<0111111010100000>  ;       BR3     16,EBCN ;TEST FOR WRITE INHIBIT, IF YES
 3979  024460  000603                  41500  ↑B<0000000110000011>  ;       BRA     1,DISC  ;TEST FOR SURE TRUE, IF YES BRAN
 3980                                  41600                    ;
 3981                                  41700                    ;
 3982                                  41800                    ;;THIS IS BCC 1
 3983                                  41900                    ;
 3984  024462  020016                  42000  ↑B<0010000000001110>  ;TBC1  RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 3985  024464  050201                  42100  ↑B<0101000010000001>  ;       S/C     7,1     ;CLEAR RAM OUTPUT 05 (EXPECT BCC
 3986  024466  050206                  42200  ↑B<0101000010000110>  ;       S/C     7,6     ;SET RAM OUTPUT 06 (EXPECT BCC 2
 3987  024470  020676                  42300  ↑B<0010000110111110>  ;       RAM     1,13,16 ;WRITE LINE PROTOCOL FROM RAM
 3988  024472  077245                  42400  ↑B<0111111010100101>  ;       BRB     16,TBC1 ;TEST FOR WRITE INHIBIT, IF YES
 3989  024474  030301                  42500  ↑B<0011000011000001>  ;       XFR     ,14,1   ;MOVE SILO OUT REGISTER TO A REG
 3990  024476  020007                  42600  ↑B<0010000000000111>  ;       RAM     0,0,7   ;OBTAIN RECV BCC CALCULATED TO D
 3991  024500  030262                  42700  ↑B<0011000010110010>  ;       XFR     ,13,2   ;MOVE RAM OUTPUT DATA TO B REGIS
 3992  024502  020012                  42800  ↑B<0010000000001010>  ;       RAM     0,0,12  ;OBTAIN TRANSMITTER DLE/LINE PRO
 3993  024504  060000                  42900  ↑B<0110000000000000>  ;       BCC             ;PERFORM SPECIFIED BCC CALCULATI
 3994  024506  020747                  43000  ↑B<0010000111100111>  ;       RAM     1,16,7  ;MOVE BCC TO RAM INPUT AND WRITE
 3995  024510  020012                  43100  ↑B<0010000000001010>  ;       RAM     0,0,12  ;OBTAIN TRANSMITTER DLE/LINE PRO
 3996  024512  071670                  43200  ↑B<0111001110111000>  ;       BRB     3,TBC1X ;TEST RAM OUTPUT 03
 3997  024514  072270                  43300  ↑B<0111010010111000>  ;       BRB     4,TBC1X ;TEST RAM OUTPUT 04
 3998  024516  020016                  43400  ↑B<0010000000001110>  ;MRTHA RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 3999  024520  050202                  43500  ↑B<0101000010000010>  ;       S/C     7,2     ;CLEAR RAM OUTPUT 06 (EXPECT BCC
 4000  024522  020676                  43600  ↑B<0010000110111110>  ;       RAM     1,13,16 ;WRITE LINE PROTOCOL
 4001  024524  077263                  43700  ↑B<0111111010110011>  ;       BRB     16,MRTHA        ;TEST FOR WRITE INHIBIT,
 4002  024526  000704                  43800  ↑B<0000000111000100>  ;       BRA     1,BCCOK ;TEST FOR SURE TRUE, IF YES BRAN
 4003  024530  050020                  43900  ↑B<0101000000010000>  ;TBC1X S/C     4,0     ;SET SILO OUT
 4004  024532  000400                  44000  ↑B<0000000100000000>  ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
 4005                                  44100                    ;
 4006                                  44200                    ;
 4007                                  44300                    ;;THIS IS BCC 2
 4008                                  44400                    ;
 4009  024534  020016                  44500  ↑B<0010000000001110>  ;TBC2  RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 4010  024536  050202                  44600  ↑B<0101000010000010>  ;       S/C     7,2     ;CLEAR RAM OUTPUT 06
 4011  024540  020676                  44700  ↑B<0010000110111110>  ;       RAM     1,13,16 ;WRITE LINE PROTOCOL
 4012  024542  077272                  44800  ↑B<0111111010111010>  ;       BRB     16,TBC2 ;TEST FOR WRITE INHIBIT, IF YES
 4013  024544  030301                  44900  ↑B<0011000011000001>  ;       XFR     ,14,1   ;MOVE SILO OUT REGISTER TO A REG
 4014  024546  020007                  45000  ↑B<0010000000000111>  ;       RAM     0,0,7   ;OBTAIN RECV BCC CALCULATED TO D
 4015  024550  030262                  45100  ↑B<0011000010110010>  ;       XFR     ,13,2   ;MOVE RAM OUTPUT DATA TO B REGIS
 4016  024552  020012                  45200  ↑B<0010000000001010>  ;       RAM     0,0,12  ;OBTAIN TRANSMITTER DLE/LINE PRO
 4017  024554  060000                  45300  ↑B<0110000000000000>  ;       BCC             ;PERFORM SPECIFIED BCC CALCULATI
 4018  024556  020747                  45400  ↑B<0010000111100111>  ;       RAM     1,16,7  ;MOVE BCC TO RAM INPUT AND WRITE
```

```
 4019  024560  010034          45500   ↑B<0001000000011100>     ;BCCCK ALU     34       ;LET ALU RESULT = MINUS 1
 4020  024562  050027          45600   ↑B<0101000000010111>     ;      S/C     4,7      ;CLEAR ALU RESULT UPPER BYTE
 4021  024564  030361          45700   ↑B<0011000011110001>     ;      XFR     ,17,1    ;MOVE ALU RESULT TO A REGISTER
 4022  024566  030302          45800   ↑B<0011000011000010>     ;      XFR     ,14,2    ;MOVE SILO OUT TO B REGISTER
 4023  024570  010015          45900   ↑B<0001000000001101>     ;      ALU     15       ;LET ALU RESULT = AND OF A COMPL
 4024  024572  030362          46000   ↑B<0011000011110010>     ;      XFR     ,17,2    ;MOVE ALU RESULT TO B REGISTER
 4025  024574  030341          46100   ↑B<0011000011100001>     ;      XFR     ,16,1    ;MOVE BCC TO A REGISTER
 4026  024576  010037          46200   ↑B<0001000000011111>     ;      ALU     37       ;LET ALU RESULT = A
 4027  024600  050027          46300   ↑B<0101000000010111>     ;      S/C     4,7      ;CLEAR ALU RESULT UPPER BYTE
 4028  024602  030361          46400   ↑B<0011000011110001>     ;      XFR     ,17,1    ;MOVE ALU RESULT TO A REGISTER
 4029  024604  010036          46500   ↑B<0001000000011110>     ;      ALU     36       ;LET ALU RESULT= A OR B
 4030  024606  030361          46600   ↑B<0011000011110001>     ;      XFR     ,17,1    ;MOVE ALU RESULT TO A REGISTER
 4031  024610  030342          46700   ↑B<0011000011100010>     ;      XFR     ,16,2    ;MOVE BCC TO B REGISTER
 4032  024612  030202          46800   ↑B<0011000010000010>     ;      XFR     ,10,2    ;MOVE B REGISTER 8-15 TO B REGIS
 4033  024614  010036          46900   ↑B<0001000000011110>     ;      ALU     36       ;LET ALU RESULT = A OR B
 4034  024616  030366          47000   ↑B<0011000011110110>     ;      XFR     ,17,6    ;MOVE ALU RESULT TO RICR REGISTE
 4035  024620  050015          47100   ↑B<0101000000001101>     ;      S/C     3,5      ;SET RICR 12
 4036  024622  050011          47200   ↑B<0101000000001001>     ;      S/C     3,1      ;SET RICR 14
 4037  024624  100662          47300   ↑B<1000000110110010>     ;      BRA     1,CNACB  ;TEST FOR SURE TRUE, IF YES BRAN
 4038                          47400                            ;
 4039                          47500                            ;;RECEIVER MPE / CONTROL BYTE
 4040                          47600                            ;
 4041  024626  030306          47700   ↑B<0011000011000110>     ;RMPEC XFR     ,14,6    ;MOVE SILO OUT TO RICR REGISTER
 4042  024630  050014          47800   ↑B<0101000000001100>     ;      S/C     3,4      ;SET RICR 13
 4043  024632  000733          47900   ↑B<0000000111011011>     ;      BRA     1,GAMMA  ;TEST FOR SURE TRUE, IF YES BRAN
 4044                          48000                            ;
 4045                          48100                            ;;RECEIVER NXM / CONTROL BYTE
 4046                          48200                            ;
 4047  024634  030306          48300   ↑B<0011000011000110>     ;RNXMC XFR     ,14,6    ;MOVE SILO OUT TO RICR REGISTER
 4048  024636  050015          48400   ↑B<0101000000001101>     ;GAMMA S/C     3,5      ;SET RICR 12
 4049  024640  050011          48500   ↑B<0101000000001001>     ;BETA  S/C     3,1      ;SET RICR 14
 4050  024642  050010          48600   ↑B<0101000000001000>     ;      S/C     3,0      ;SET RICR 15
 4051  024644  050017          48700   ↑B<0101000000001111>     ;      S/C     3,7      ;CLEAR NXM
 4052  024646  100662          48800   ↑B<1000000110110010>     ;      BRA     1,CNACB  ;TEST FOR SURE TRUE, IF YES BRAN
 4053                          48900                            ;
 4054                          49000                            ;
 4055                          49100                            ;;RECEIVER NXM (WE GOT HERE FROM RECEIVED CHARACTER SIL
 4056                          49200                            ;
 4057  024650  030306          49300   ↑B<0011000011000110>     ;RNXM  XFR     ,14,6    ;MOVE SILO OUT TO RICR REGISTER
 4058  024652  000734          49400   ↑B<0000000111011100>     ;      BRA     1,BETA   ;TEST FOR SURE TRUE, IF YES BRAN
 4059                          49500                            ;
 4060                          49600                            ;
 4061                          49700                            ;;NPR SILO OVERFLOW
 4062                          49800                            ;
 4063  024654  050012          49900   ↑B<0101000000001010>     ;NPRSO S/C     3,2      ;SET SCR 10 INDICATING NPR SILO
 4064  024656  000400          50000   ↑B<0000000100000000>     ;      BRA     1,ILOOP  ;TEST FOR SURE TRUE, IF YES BRAN
 4065                          50100                            ;
 4066                          50200                            ;
 4067                          50300                            ;;TRANSMIT SERVICE
 4068                          50400                            ;
 4069                          50500                            ;
 4070                          50600                            ;;CHECK FOR BCC TRANSMISSION
 4071                          50700                            ;
 4072  024660  020016          50800   ↑B<0010000000001110>     ;TSERV RAM     0,0,16   ;OBTAIN LINE PROTOCOL
 4073  024662  030261          50900   ↑B<0011000010110001>     ;      XFR     ,13,1    ;MOVE RAM OUTPUT DATA TO A REGIS
 4074  024664  010037          51000   ↑B<0001000000011111>     ;      ALU     37       ;LET ALU RESULT=A REGISTER
```

```
024666 105226   ↑B<1000101010010110>   ;        BRA    12,SBC1   ;TEST BIT 0 OF ALU RESULT, IF YE
024670 105642   ↑B<1000101110100010>   ;        BRA    13,SBC2   ;TEST BIT 1 OF ALU RESULT, IF YE
024672 003742   ↑B<0000011111100010>   ;        BRA    7,NPRSO   ;TEST FOR NPR SILO NOT AVAILABLE
                                       ;
                                       ;
                                       ;;PRINCIPAL/ALTERNATE SELECTION
                                       ;
024674 020013   ↑B<0010000000001011>   ;        RAM    0,0,13    ;OBTAIN LINE STATE
024676 071355   ↑B<0111001011101101>   ;        BRB    2,SIGMA   ;TEST BIT 02 OF RAM, IF YES, BRA
024700 100544   ↑B<1000000101100100>   ;        BRA    1,ITYPE   ;TEST FOR SURE TRUE, IF YES BRAN
024702 073762   ↑B<0111011111110010>   ;SIGMA   BRB    7,USCA    ;TEST RAM OUTPUT 07, IF YES BRAN
024704 020001   ↑B<0010000000000001>   ;        RAM    0,0,1     ;OBTAIN PRINCIPAL BC (GE TEST)
024706 176100   ↑B<1111110001000000>   ;        BRB    14,XPBCO          ;TEST RAM OUTPUT 0-14=0,
024710 020000   ↑B<0010000000000000>   ;        RAM    0,0,0     ;OBTAIN PRINCIPAL CURRENT ADDRES
024712 000765   ↑B<0000000111110101>   ;        BRA    1,OXCB    ;TEST FOR SURE TRUE, IF YES BRAN
                                       ;
                                       ;
                                       ;;USE ALTERNATE CA
                                       ;
024714 020003   ↑B<0010000000000011>   ;USCA    RAM    0,0,3     ;OBTAIN ALTERNATE BC. (GE TEST)
024716 176120   ↑B<1111110001010000>   ;        BRB    14,XSBCO          ;TEST RAM OUTPUT 0-14=0,
024720 020002   ↑B<0010000000000010>   ;        RAM    0,0,2     ;OBTAIN ALTERNATE CURRENT ADDRES
                                       ;;OBTAIN XMIT CONTROL BYTE
                                       ;
024722 030263   ↑B<0011000010110011>   ;OXCB    XFR    I,13,3    ;MOVE DATA FROM RAM OUTPUT TO NP
024724 040000   ↑B<0100000000000000>   ;        NPR              ;DO NPR TO GET CHARACTER
024726 074767   ↑B<0111100111110111>   ;RBUS3   BRB    11,RBUS3          ;TEST REQUEST BUS, IF YE
024730 103061   ↑B<1000011000110001>   ;        BRA    6,↑NXMC   ;TEST NXM, IF YES, BRANCH TO TRA
024732 175071   ↑B<1111101000111001>   ;        BRB    12,↑MPEC          ;TEST MEM PAR ERR, IF YE
024734 030221   ↑B<0011000010010001>   ;        XFR    ,11,1     ;MOVE DATA FROM CDC/DATI REGISTE
024736 010037   ↑B<0001000000011111>   ;        ALU    37        ;LET ALU RESULT = A REGISTER
024740 030365   ↑B<0011000011110101>   ;        XFR    ,17,5     ;MOVE DATA FROM ALU RESULT TO CA
024742 020012   ↑B<0010000000001010>   ;        RAM    0,0,12    ;OBTAIN TRANSMITTER DLE/LINE PRO
024744 173214   ↑B<1111011010001100>   ;        BRB    6,DDCMX   ;TEST RAM OUTPUT 06, IF YES BRAN
024746 020014   ↑B<0010000000001100>   ;        RAM    0,0,14    ;OBTAIN MODE BITS
024750 030242   ↑B<0011000010100010>   ;PI      XFR    ,12,2     ;MOVE RAM OUTPUT DATA TRANSLATED
024752 010026   ↑B<0001000000010110>   ;        ALU    26        ;LET ALU RESULTS = A PLUS B
024754 030362   ↑B<0011000011110010>   ;        XFR    ,17,2     ;MOVE ALU RESULTS TO B REGISTER
024756 020010   ↑B<0010000000001000>   ;        RAM    0,0,10    ;OBTAIN CONTROL TABLE BASE ADDRE
024760 030261   ↑B<0011000010110001>   ;        XFR    ,13,1     ;MOVE RAM OUTPUT TO A REGISTER
024762 010026   ↑B<0001000000010110>   ;        ALU    26        ;LET ALU RESULT = A PLUS B ((CHA
024764 030363   ↑B<0011000011110011>   ;        XFR    I,17,3    ;MOVE ALU RESULTS TO NPR ADDRESS
024766 040000   ↑B<0100000000000000>   ;        NPR              ;DO NPR TO GET CONTROL BYTE
024770 174410   ↑B<1111100100001000>   ;RBUS4   BRB    11,RBUS4          ;TEST REQUEST BUS, IF YE
024772 103061   ↑B<1000011000110001>   ;        BRA    6,↑NXMC   ;TEST NXM, IF YES, BRANCH TO TRA
024774 175071   ↑B<1111101000111001>   ;        BRB    12,↑MPEC          ;TEST MEM PAR ERR, IF YE
024776 030222   ↑B<0011000010010010>   ;        XFR    ,11,2     ;MOVE DATI REGISTER TO B REGISTE
025000 010005   ↑B<0001000000000101>   ;        ALU    5         ;LET ALU RESULT = B REGISTER
025002 105563   ↑B<1000101101110011>   ;        BRA    13,SDLE   ;TEST BIT 1 OF ALU RESULT, IF YE
                                       ;
                                       ;
                                       ;
                                       ;
                                       ;;RETURN FROM DLE SENDING
                                       ;
025004 020574   ↑B<0010000101111100>   ;RDLE    RAM    1,7,14    ;MOVE ALU RESULT TRANSLATED 0-7
```

```
 4131  025006  106202          56700  ↑B<1000110010030010>      ;       BRA     14,SSBN ;TEST BIT 2 OF ALU RESULT, IF YE
 4132                          56800                            ;
 4133                          56900                            ;
 4134                          57000                            ;;RETURN FROM SSBCCNXT
 4135                          57100                            ;
 4136  025010  106617          57200  ↑B<1000110110001111>      ;RSSBN BRA     15,XBCC :TEST BIT 3 OF ALU RESULT, IF YE
 4137                          57300                            ;
 4138                          57400                            ;
 4139                          57500                            ;;RETURN FROM XMIT BCC
 4140                          57600                            ;
 4141  025012  174151          57700  ↑B<1111100001101001>      ;RXBCC BRB     10,SIDLE        ;TEST FOR DNA FLAG, IF Y
 4142  025014  030323          57800  ↑B<0011C000011010000>      ;ALPHA XFR      15,0    ;MOVE DATO REGISTER TO TRANSMITT
 4143  025016  020013          57900  ↑B<0010000000001011>      ;       RAM     0,0,13  ;OBTAIN LINE STATE
 4144  025020  173443          58000  ↑B<1111011100100011>      ;       BRB     7,USBC  ;TEST BIT 7 OF RAM OUTPUT, IF YE
 4145                          58100                            ;
 4146                          58200                            ;
 4147                          58300                            ;;USE PRINCIPAL BC
 4148                          58400                            ;
 4149  025022  020001          58500  ↑B<0010000000000001>      ;LPBC   RAM     0,0,1   ;OBTAIN PRINCIPAL BYTE COUNT
 4150  025024  176100          59600  ↑B<1111110001000000>      ;       BRB     14,XPBCO        ;TEST RAM 0-14=0
 4151  025026  030261          58700  ↑B<0011000010110001>      ;       XFR     ,13.1   ;MOVE RAM OUTPUT TO A REGISTER
 4152  025030  010077          58800  ↑B<0001000000111111>      ;       ALU     77      ;LET ALU RESULTS = A PLUS 1
 4153  025032  020761          58900  ↑B<0010000111110001>      ;       RAM     1,17,1  ;MOVE ALU RESULT TO RAM INPUT AN
 4154  025034  177025          59000  ↑B<1111111000010101>      ;       BRB     16,UPBC ;TEST FOR WRITE INHIBIT, IF YES
 4155  025036  020000          59100  ↑B<0010000000000000>      ;OPCA   RAM     0,0,0   ;OBTAIN PRINCIPAL CURRENT ACDRES
 4156  025040  030261          59200  ↑B<0011000010110001>      ;       XFR     ,13.1   ;MOVE RAM OUTPUT TO A REGISTER
 4157  025042  010077          59300  ↑B<0001000000111111>      ;       ALU     77      ;LET ALU RESULT = A PLUS 1
 4158  025044  020760          59400  ↑B<0010000111110000>      ;       RAM     1,17,0  ;MOVE ALU RESULT TO RAM INPUT AN
 4159  025046  177033          59500  ↑B<1111111000011011>      ;       BRB     16,OPCA ;TEST FOR WRITE INHIBIT, IF YES
 4160  025050  020001          59600  ↑B<0010000000000001>      ;       RAM     0,0,1   ;OBTAIN PRINCIPAL BYTE COUNT
 4161  025052  176100          59700  ↑B<1111110001000000>      ;       BRB     14,XPBCO        ;TEST RAM 0-14-0, IF YES
 4162  025054  000400          59800  ↑B<0000000100000000>      ;       BRA     1,ILOOP ;TEST FOR SURE TRUE AND BRANCH T
 4163                          59900                            ;
 4164                          60000                            ;
 4165                          60100                            ;;USE ALTERNATE BC
 4166                          60200                            ;
 4167  025056  020003          60300  ↑B<0010000000000011>      ;USBC   RAM     0,0,3   ;OBTAIN ALTERNATE BYTE COUNT
 4168  025060  176120          60400  ↑B<1111110001010000>      ;       BRB     14,XSBCO        ;TEST RAM 0-14=0
 4169  025062  030261          60500  ↑B<0011000010110001>      ;       XFR     ,13.1   ;MOVE RAM OUTPUT TO A REGISTER
 4170  025064  010077          60600  ↑B<0001000000111111>      ;       ALU     77      ;LET ALU RESULTS = A PLUS 1
 4171  025066  020763          60700  ↑B<0010000111110011>      ;       RAM     1,17,3  ;MOVE ALU RESULT TO RAM INPUT AN
 4172  025070  177043          60800  ↑B<1111111000100011>      ;       BRB     16,USBC ;TEST FOR WRITE INHIBIT, IF YES
 4173  025072  020002          60900  ↑B<0010000000000010>      ;OSCA   RAM     0,0,2   ;OBTAIN ALTERNATE CURRENT ACDRES
 4174  025074  030261          61000  ↑B<0011000010110001>      ;       XFR     ,13,1   ;MOVE RAM OUTPUT TO A REGISTER
 4175  025076  010077          61100  ↑B<0001000000111111>      ;       ALU     77      ;LET ALU RESULT = A PLUS 1
 4176  025100  020762          61200  ↑B<0010000111110010>      ;       RAM     1,17,2  ;MOVE ALU RESULT TO RAM INPUT AN
 4177  025102  177051          61300  ↑B<1111111000101001>      ;       BRB     16,OSCA ;TEST FOR WRITE INHIBIT, IF YES
 4178  025104  020003          61400  ↑B<0010000000000011>      ;       RAM     0,0,3   ;OBTAIN ALTERNATE BYTE COUNT
 4179  025106  176120          61500  ↑B<1111110001010000>      ;       BRB     14,XSBCO        ;TEST RAM 0-14-0, IF YES
 4180  025110  000400          61600  ↑B<0000000100000000>      ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
 4181                          61700                            ;
 4182                          61800                            ;
 4183                          61900                            ;;TRANSMIT NXM CHARACTER  (CURRENT ADDRESS REGISTER ADD
 4184                          62000                            ;
 4185                          62100                            ;
 4186                          62200                            ;;TRANSMIT NXM CONTROL BYTE  (CONTROL TABLE BASE ADDRES
```

```
 4187                                   62300
 4188  025112  030007                   62400   ↑B<0011000000000111>     ;TMXMC XFR      0,7     ;MOVE  TO NPR STATUS REPORT REG.
 4189  025114  020013                   62500   ↑B<0010000000001011>     ;IOTA  RAM      0,0,13  ;OBTAIN LINE STATE
 4190  025116  050207                   62600   ↑B<0101000010000111>     ;       S/C     7,7     ;SET RAM 04 (TRANSMITTER NXM)
 4191  025120  050017                   62700   ↑B<0101000000001111>     ;       S/C     3,7     ;CLEAR NXM
 4192  025122  050042                   62800   ↑B<0101000000100010>     ;       S/C     5,2     ;CLEAR RAM 02 (TRANSMITTER GO)
 4193  025124  020673                   62900   ↑B<0010000110111011>     ;       RAM     1,13,13 ;WRITE NEW LINE STATE
 4194  025126  177065                   63000   ↑B<1111111000110010>     ;       BRB     16,IOTA ;TEST FOR WRITE INHIBIT. IF YES
 4195  025130  000400                   63100   ↑B<0000000100000000>     ;       BRA     1,ILOOP ;TEST FOR SURE TRUE. IF YES BRAN
 4196                                   63200
 4197                                   63300
 4198                                   63400                            ;;TRANSMIT MPE/CHARACTER   (CURRENT ADDRESS REGISTER AD
 4199                                   63500
 4200                                   63600
 4201                                   63700
 4202                                   63800                            ;;TRANSMIT MPE/CONTROL BYTE  (CONTROL TABLE BASE ADDRES
 4203                                   63900
 4204  025132  030007                   64000   ↑B<0011000000000111>     ;TMPEC XFR      0,7     ;MOV  TO NPR STATUS REPORT REG.
 4205  025134  020013                   64100   ↑B<0010000000001011>     ;OMEGA RAM      0,0,13  ;OBTAIN LINE STATE
 4206  025136  050205                   64200   ↑B<0101000010000101>     ;       S/C     7,5     ;SET RAM 05 (TRANSMIT MPE)
 4207  025140  050042                   64300   ↑B<0101000000100010>     ;       S/C     5,2     ;CLEAR RAM 02 (TRANSMITTER GO)
 4208  025142  020673                   64400   ↑B<0010000110111011>     ;       RAM     1,13,13 ;WRITE NEW LINE STATE
 4209  025144  177072                   64500   ↑B<1111111000111010>     ;       BRB     16,OMEGA        ;TEST FOR WRITE INHIBIT.
 4210  025146  000400                   64600   ↑B<0000000100000000>     ;       BRA     1,ILOOP ;TEST FOR SURE TRUE. IF YES. BRA
 4211                                   64700
 4212                                   64800
 4213                                   64900                            ;;XMIT PBCO
 4214                                   65000
 4215  025150  020013                   65100   ↑B<0010000000001011>     ;XPBCC RAM      0,0,13  ;OBTAIN LINE STATE
 4216  025152  050204                   65200   ↑B<0101000010000100>     ;       S/C     7,4     ;SET RAM OUTPUT BIT 7 (GO TO ALT
 4217  025154  020673                   65300   ↑B<0010000110111011>     ;       RAM     1,13,13 ;MOVE RAM OUTPUT TO RAM (AND WRI
 4218  025156  177100                   65400   ↑B<1111111001000000>     ;       BRB     16,XPBCC        ;TEST FOR WRITE INHIBT.
 4219  025160  020001                   65500   ↑B<0010000000000001>     ;       RAM     0,0,1   ;OBTAIN PRINCIPAL BYTE COUNT
 4220  025162  030007                   65600   ↑B<0011000000000111>     ;       XFR     0,7     ;MAKE NPR SILO ENTRY
 4221  025164  175510                   65700   ↑B<1111101101001000>     ;       BRB     13,RED  ;TEST RAM OUTPUT BIT 15. IF YES
 4222  025166  100527                   65800   ↑B<1000000101010111>     ;       BRA     1,CBCO  ;TEST FOR SURE TRUE. IF YES BRAN
 4223  025170  020401                   65900   ↑B<0010000100000001>     ;RED    RAM     1,0,1   ;ZERO PRINCIPAL BYTE COUNT
 4224  025172  020016                   66000   ↑B<0010000000001110>     ;CTMB   RAM     0,0,16  ;OBTAIN  LINE PROTOCOL
 4225  025174  030262                   66100   ↑B<0011000010110010>     ;       XFR     ,13,2   ;MOVE RAM OUTPUT TO REGISTER 8
 4226  025176  030202                   66200   ↑B<0011000010000010>     ;       XFR     ,10,2   ;MOVE REGISTER 89-15 TO REGISTER
 4227  025200  010005                   66300   ↑B<0001000000000101>     ;       ALU     5       ;LET ALU RESULT = B
 4228  025202  020574                   66400   ↑B<0010000101111100>     ;       RAM     1,7,14  ;MOVE ALU RESULTS TRANSLATED  TO
 4229  025204  106207                   66500   ↑B<1000110010000111>     ;       BRA     14,BCOSB        ;TEST  ALU RESULT 02, IF
 4230  025206  100527                   66600   ↑B<1000000101010111>     ;       BRA     1,CBCO  ;TEST FOR SURE TRUE. IF YES BRAN
 4231                                   66700
 4232                                   66800                            ;;XMIT SBCO
 4233                                   66900
 4234  025210  020013                   67000   ↑B<0010000000001011>     ;XSBCO RAM      0,0,13  ;OBTAIN LINE STATE
 4235  025212  050200                   67100   ↑B<0101000010000000>     ;       S/C     7,0     ;CLEAR RAM OUTPUT BIT 7 (GO TO P
 4236  025214  020673                   67200   ↑B<0010000110111011>     ;       RAM     1,13,13 ;MOVE RAM OUTPUT TO RAM (AND WRI
 4237  025216  177120                   67300   ↑B<1111111001010000>     ;       BRB     16,XSBCO        ;TEST FOR WRITE INHIBIT.
 4238  025220  020003                   67400   ↑B<0010000000000011>     ;       RAM     0,0,3   ;OBTAIN ALTERNATE BYTE COUNT
 4239  025222  030007                   67500   ↑B<0011000000000111>     ;       XFR     0,7     ;MAKE NPR SILO ENTRY
 4240  025224  175535                   67600   ↑B<1111101101011101>     ;       BRB     13,ESS  ;TEST RAM OUTPUT 15. IF YES. BRA
 4241                                   67700
 4242                                   67800                            ;
```

```
                                67900                           ;;CHECK FOR BOTH BC=0
                                68000                           ;
 025226  020001                 68100  ↑B<0010000000000001>     ;CBCO  RAM     0,0,1   ;OBTAIN PRINCIPAL BYTE COUNT
 025230  176132                 68200  ↑B<1111110001011010>     ;       BRB     14,DELTA        ;TEST RAM OUTPUT 0-14-0,
 025232  000400                 68300  ↑B<0000000100000000>     ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES, BRA
 025234  020003                 68400  ↑B<0010000000000011>     ;DELTA RAM     0,0,3   ;OBTAIN ALTERNATE BYTE COUNT
 025236  176137                 68500  ↑B<1111110001011111>     ;       BRB     14,C/GO ;TEST RAM OUTPUT 0-14-0, IF YES,
 025240  000400                 68600  ↑B<0000000100000000>     ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
                                68700                           ;
                                68800                           ;
                                68900                           ;;CLEAR ALTERNATE BYTE COUNT
                                69000                           ;
 025242  020403                 69100  ↑B<0010000100000011>     ;ESS   RAM     1,0,3   ;ZERO ALTERNATE BYTE COUNT
 025244  100511                 69200  ↑B<1000000101001001>     ;       BRA     1,0↑MB  ;TEST FOR SURE TRUE, BRANCH TO 0
                                69300                           ;
                                69400                           ;
                                69500                           ;;CLEAR GO
                                69600                           ;
 025246  020013                 69700  ↑B<0010000000001011>     ;C/GO  RAM     0,0,13  ;OBTAIN LINE STATE
 025250  050042                 69800  ↑B<0101000000100010>     ;       S/C     5,2     ;CLEAR RAM 02 (TRANSMITTER GO)
 025252  020673                 69900  ↑B<0010000110111011>     ;       RAM     1,13,13 ;WRITE NEW LINE STATE
 025254  177137                 70000  ↑B<1111111001011111>     ;       BRB     16,C/GO ;TEST FOR WRITE INHIBIT, IF YES
 025256  000400                 70100  ↑B<0000000100000000>     ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
                                70200                           ;
                                70300                           ;;SELECT TYPE OF IDLE
                                70400                           ;
 025260  020012                 70500  ↑B<0010000000001010>     ;ITYPE RAM     0,0,12  ;OBTAIN TRANSMITTER DLE/PROTOCOL
 025262  170147                 70600  ↑B<1111000001100111>     ;       BRB     0,BCOCG ;TEST RAM OUTPUT 00, IF YES BRAN
 025264  000400                 70700  ↑B<0000000100000000>     ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
 025266  050105                 70800  ↑B<0101000001000101>     ;BCOCG S/C     6,5     ;SET TMARK
 025270  000400                 70900  ↑B<0000000100000000>     ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
                                71000                           ;
                                71100                           ;
                                71200                           ;;SENT IDLE  (LINE STATE IS IN RAM OUTPUT)
                                71300                           ;
 025272  020013                 71400  ↑B<0010000000001011>     ;SIDLE RAM     0,0,13  ;OBTAIN LINE STATE
 025274  050044                 71500  ↑B<0101000000100100>     ;       S/C     5,4     ;SET RAM 03 (TRANSMITTER UNDERRU
 025276  020673                 71600  ↑B<0010000110111011>     ;       RAM     1,13,13 ;MOVE RAM OUTPUT TO RAM (AND WRI
 025300  177151                 71700  ↑B<1111111001101001>     ;       BRB     16,SIDLE        ;TEST FOR WRITE INHIBIT,
 025302  100422                 71800  ↑B<1000000100010010>     ;       BRA     1,ALPHA ;TEST FOR SURE TRUE, IF YES, BRA
                                71900                           ;
                                72000                           ;
                                72100                           ;;SENT IDLE/DLE
                                72200                           ;
 025304  020013                 72300  ↑B<0010000000001011>     ;MU    RAM     0,0,13  ;OBTAIN LINE STATE
 025306  050044                 72400  ↑B<0101000000100100>     ;       S/C     5,4     ;SET 03 (UNDERRUN)
 025310  020673                 72500  ↑B<0010000110111011>     ;       RAM     1,13,13 ;WRITE LINE STATE
 025312  177156                 72600  ↑B<1111111001101110>     ;       BRB     16,MU   ;TEST FOR INHIBIT
 025314  100573                 72700  ↑B<1000000101111011>     ;       BRA     1,MU    ;GO BACK TO SEND IDLE
                                72800                           ;
                                72900                           ;
                                73000                           ;;SEND DLE FIRST
                                73100                           ;
                                73200                           ;;(WE GOT HERE FROM TRANSMIT SERVICE.  THE CONTROL BYTE
                                73300                           ;;IN ALU RESULT AND B REGISTER.  MASTER SCAN POSITION I
                                73400                           ;;RAM ADDRESS REGISTER 0-3).
```

```
4299
4300  025316  020016          73500
4301  025320  171175  73600   ↑B<0010000000001110>  ;SDLE  RAM   0,0,16  ;OBTAIN LINE PROTOCOL
4302  025322  050046  73700   ↑B<1111001001111101>  ;      BRB   2,CRAM2 ;TEST RAM OUTPUT 02, IF YES BRAN
4303  025324  020676  73800   ↑B<0101000000100110>  ;      S/C   5,6     ;SET RAM 02
4304  025326  177163  73900   ↑B<0010000110111110>  ;      RAM   1,13,16 ;MOVE RAM OUTPUT TO RAM INPUT AN
4305  025330  020012  74000   ↑B<1111111001100011>  ;      BRB   16,SDLE ;TEST FOR WRITE INHIBIT, IF YES
4306  025332  030262  74100   ↑B<0010000000001010>  ;      RAM   0,0,12  ;OBTAIN TRANSMITTER DLE / LINE P
4307  025334  174156  74200   ↑B<0011000C10110010>  ;      XFR    13,2   ;MOVE RAM OUTPUT TO REGISTER B
4308  025336  030203  74300   ↑B<1111110C01101110>  ;      BRB   10,MU   ;TEST FOR DNA FLAG, IF YES BRANC
4309  025340  000400  74400   ↑B<0011000010000000>  ;NU    XFR    10,0   ;MOVE REGISTER B 15-8 TRANSMITTE
4310                  74500   ↑B<0000000100000000>  ;      BRA   1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
4311                  74600                         ;
4312                  74700                         ;
4313                  74800                         ;:CLEAR RAM 02 (RAM 02 IS DLE SENDING IN PROGRESS
4314  025342  020016  74900                         ;
4315  025344  050042  75000   ↑B<0010000C00001110>  ;CRAM2 RAM   0,0,16  ;OBTAIN LINE PROTOCOL
4316  025346  020676  75100   ↑B<0101000000100010>  ;      S/C   5,2     ;CLEAR RAM OUTPUT 02
4317  025350  177175  75200   ↑B<0010000110111110>  ;      RAM   1,13,16 ;MOVE RAM OUTPUT TO RAM INPUT DA
4318  025352  100416  75300   ↑B<1111111001111101>  ;      BRB   16,CRAM2        ;TEST FOR WRITE INHIBIT,
4319                  75400   ↑B<1000000100001110>  ;      BRA   1,RDLE  ;TEST FOR SURE TRUE, IF YES BRAN
4320                  75500                         ;
4321                  75600                         ;
4322                  75700                         ;:SET SEND BCC NEXT
4323                  75800                         ;
4324  025354  020016  75900   ↑B<0010000000001110>  ;SSBN  RAM   0,0,16  ;OBTAIN LINE PROTOCOL
4325  025356  050047  76000   ↑B<0101000000100111>  ;      S/C   5,7     ;SET RAM OUTPUT BIT 0
4326  025360  020675  76100   ↑B<0010000110111110>  ;      RAM   1,13,15 ;MOVE RAM OUTPUT TO RAM INPUT AN
4327  025362  177202  76200   ↑B<1111111010000010>  ;      BRB   16,SSBN ;TEST FOR WRITE INHIBIT, IF YES
4328  025364  100423  76300   ↑B<1000000100010000>  ;      BRA   1,RSSBN ;TEST FOR SURE TRUE, IF YES BRAN
4329                  76400                         ;
4330                  76500                         ;
4331                  76600                         ;:BCC SEND BCC
4332  025366  020016  76700                         ;
4333  025370  050047  76800   ↑B<0010000000001110>  ;BCCSB RAM   0,0,16  ;OBTAIN LINE PROTOCOL
4334  025372  020676  76900   ↑B<0101000000100111>  ;      S/C   5,7     ;SET RAM OUTPUT BIT 0
4335  025374  177207  77000   ↑B<0010000110111110>  ;      RAM   1,13,16 ;MOVE RAM OUTPUT TO RAM INPUT AN
4336  025376  100527  77100   ↑B<1111111010000111>  ;      BRB   16,BCCSB        ;TEST FOR WRITE INHIBIT,
4337                  77200   ↑B<1000000101010111>  ;      BRA   1,CBCC  ;TEST FOR SURE TRUE, IF YES BRAN
4338                  77300                         ;
4339                  77400                         ;
4340                  77500                         ;:DDCMP TRANSMIT
4341  025400  020014  77600                         ;
4342  025402  176217  77700   ↑B<0010000000001100>  ;DDCMX RAM   0,0,14  ;OBTAIN TRANSMITTER MODE BITS
4343  025404  100400  77800   ↑B<1111110010001111>  ;      BRB   14,XBCC ;TEST RAM 0-14=0, IF YES BRANCH
4344                  77900   ↑B<1000000100000000>  ;      BRA   1,PI    ;TEST FOR SURE TRUE, IF YES BRAN
4345                  78000                         ;
4346                  78100                         ;
4347                  78200                         ;:CALCULATE TRANSMITTER BCC
4348  025406  030321  78300                         ;
4349  025410  020006  78400   ↑B<0011000011010001>  ;XBCC  XFR    15,1   ;MOVE DATO REGISTER TO A REGISTE
4350  025412  030262  78500   ↑B<0010000000000110>  ;      RAM   0,0,6   ;OBTAIN TRANSMITTER BCC CALCULAT
4351  025414  020012  78600   ↑B<0011000010110010>  ;      XFR    13,2   ;MOVE RAM DATA TO REGISTER B
4352  025416  060000  78700   ↑B<0010000000001010>  ;      RAM   0,0,12  ;OBTAIN TRANSMITTER DLE/LINE PRO
4353  025420  020746  78800   ↑B<0110000000000000>  ;      BCC           ;PERFORM SPECIFIED BCC CALCULATI
4354  025422  100421  78900   ↑B<0010000111100110>  ;      RAM   1,16,6  ;MOVE BCC TO RAM INPUT AND WRITE
                      79000   ↑B<1000000100010001>  ;      BRA   1,RXBCC ;TEST FOR SURE TRUE, IF YES, BRA
```

```
 4355                                   79100
 4356                                   79200
 4357                                   79300   ;:SEND BCC 1
 4358                                   79400   ;
 4359                                   79500   ;:(WE GOT HERE FROM TRANSMIT SERVICE.  THE MASTER SCAN
 4360                                   79600   ;:IS IN THE RAM ADDRESS REGISTER 0-3.  THE LINE PROTOCO
 4361                                   79700   ;:THE A REGISTER AND THE ALU RESULT REGISTER.)
 4362                                   79800   ;
 4363                                   79900   ;
 4364  025424  020016                   80000   ↑B<0010000000001110>  ;SBC1  RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 4365  025426  030261                   80100   ↑B<0011000010110001>  ;       XFR     ,13,1   ;MOVE RAM OUTPUT TO A REGISTER
 4366  025430  000077                   80200   ↑B<0001000000111111>  ;       ALU     77      ;LET ALU RESULT = A PLUS 1
 4367  025432  020776                   80300   ↑B<0010000111111110>  ;       RAM     1,17,16 ;MOVE ALU RESULT TO RAM INPUT DA
 4368  025434  177226                   80400   ↑B<1111111010010110>  ;       BRB     16,SBC1 ;TEST FOR WRITE INHIBIT, IF YES
 4369  025436  020006                   80500   ↑B<0010000000000110>  ;       RAM     0,0,6   ;OBTAIN TRANSMITTER BCC
 4370  025440  030260                   80600   ↑B<0011000010110000>  ;       XFR     ,13,0   ;MOVE RAM OUTPUT DATA TO TRANSMI
 4371  025442  020012                   80700   ↑B<0010000000001010>  ;       RAM     0,0,12  ;OBTAIN TRANSMITTER DLE/LINE PRO
 4372  025444  171641                   80800   ↑B<1111001110100001>  ;       BRB     3,GOIDL ;TEST RAM OUTPUT 03, IF YES BRAN
 4373  025446  172241                   80900   ↑B<1111010010100001>  ;       BRB     4,GOIDL ;TEST RAM OUTPUT 04, IF YES BRAN
 4374  025450  100645                   81000   ↑B<1000000110100101>  ;       BRA     1,C/LU1 ;TEST FOR SURE TRUE, IF YES BRAN
 4375  025452  000400                   81100   ↑B<0000000100000000>  ;GOIDL BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
 4376                                   91200   ;
 4377                                   81300   ;
 4378                                   81400   ;:SEND BCC2
 4379                                   81500   ;
 4380                                   81600   ;:(WE GOT HERE FROM TRANSMIT SERVICE.  THE MASTER SCAN
 4381                                   81700   ;:IS IN THE RAM ADDRESS REGISTER 0-3.  THE LINE PROTOCO
 4382                                   81800   ;:THE A REGISTER AND IN THE ALU RESULT REGISTER.)
 4383                                   81900   ;
 4384  025454  020006                   82000   ↑B<0010000000000110>  ;SBC2  RAM     0,0,6   ;OBTAIN TRANSMITTER BCC
 4385  025456  030262                   82100   ↑B<0011000010110010>  ;       XFR     ,13,2   ;MOVE RAM OUTPUT DATA TO REGISTE
 4386  025460  030200                   82200   ↑B<0011000010000000>  ;       XFR     ,10,0   ;MOVE REGISTER B 8-15/0-7 TO TRA
 4387                                   82300   ;
 4388                                   82400   ;
 4389  025462  020406                   82500   ↑B<0010000100000110>  ;C/LU1 RAM     1,0,6   ;MOVE ZERO TO RAM INPUT DATA AND
 4390  025464  020016                   82600   ↑B<0010000000001110>  ;       RAM     0,0,16  ;OBTAIN LINE PROTOCOL
 4391  025466  050041                   82700   ↑B<0101000000100001>  ;       S/C     5,1     ;CLEAR RAM BIT 01 (SEND BCC 2)
 4392  025470  020676                   82800   ↑B<0010000110111110>  ;       RAM     1,13,16 ;MOVE RAM INPUT TO RAM INPUT DAT
 4393  025472  177245                   82900   ↑B<1111111010100101>  ;       BRB     16,C/LU1        ;TEST FOR WRITE INHIBIT,
 4394  025474  000400                   83000   ↑B<0000000100000000>  ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
 4395                                   83100   ;
 4396                                   83200   ;
 4397                                   83300   ;
 4398                                   83400   ;
 4399                                   83500   ;:RECEIVED ERRORS
 4400                                   83600   ;
 4401                                   83700   ;
 4402                                   83800   ;:CONTROL BYTE INTERRUPT
 4403                                   83900   ;
 4404  025476  020017                   84000   ↑B<0010000000001111>  ;CBINT RAM     0,0,17  ;READ CONTROL BYTE HOLDING REGIS
 4405  025500  050043                   84100   ↑B<0101000000100011>  ;       S/C     5,3     ;CLEAR RAM 00 (GENERATE INTERRUP
 4406  025502  020677                   84200   ↑B<0010000110111111>  ;       RAM     1,13,17 ;WRITE CONTROL BYTE HOLDING REGI
 4407  025504  030306                   84300   ↑B<0011000011000110>  ;       XFR     ,14,6   ;MOVE SILO OUT TO RICR REGISTER
 4408  025506  050013                   84400   ↑B<0101000000001011>  ;       S/C     3,3     ;SET SCROT (RECEIVER INTERRUPT)
 4409  025510  000400                   84500   ↑B<0000000100000000>  ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
 4410                                   84600   ;
```

H07


```
 4411                                    94700
 4412                                    94800                ;
 4413                                    94900                ;PARITY AND OVERRUN ERRORS
 4414  025512  030336                    95000  ↑B<0011000011000110>  ;PCER  XFR     .14.6   ;MOVE SILO OUT TO RICR REGISTER
 4415                                    95100                ;
 4416                                    95200                ;
 4417                                    95300                ;
 4418                                    95400                ;CREATE NULL ACTION CONTROL BYTE (MODE IS PRESERVED)
 4419                                    95500                ;
 4420  025514  020017                    95600  ↑B<0010000000001111>  ;CNACB RAM     0,0,17  ;READ CONTROL BYTE HOLDING REGIS
 4421  025516  050043                    95700  ↑B<0101000000100011>  ;       S/C     5,3     ;CLEAR RAM 00
 4422  025520  050041                    95800  ↑B<0101000000100001>  ;       S/C     5,1     ;CLEAR RAM 01
 4423  025522  050042                    95900  ↑B<0101000000100010>  ;       S/C     5,2     ;CLEAR RAM 02
 4424  025524  050040                    96000  ↑B<0101000000100000>  ;       S/C     5,0     ;CLEAR RAM 03
 4425  025526  050207                    96100  ↑B<0101000010000111>  ;       S/C     7,7     ;SET RAM OUTPUT 04 (DISCARD)
 4426  025530  020677                    96200  ↑B<0010000110111111>  ;       RAM     1,13,17 ;WRITE CONTROL BYTE HOLDING REGI
 4427  025532  050013                    96300  ↑B<0101000000001011>  ;       S/C     3,3     ;SET SCR C7 (RECEIVER INTERRUPT)
 4428  025534  000400                    96400  ↑B<0000000100000000>  ;       BRA     1,ILOOP ;TEST FOR SURE TRUE, IF YES BRAN
 4429                                    96500                ;
 4430                                    96600                ;
 4431                                    96700                ;
 4432                                    96800                ;END
 4433
```

```
                                00100   ;                                SYMBOL TABLE
                                00200   ;                    [IN ORDER AS THEY APPEAR IN PROGRAM]
                                00300   ;
                                00400   ;  000000000  ILOOP   000000101  ILOP2   000001000  ILOP5   000001010  RSERV
                                00500   ;  000001111  S/ACT   000010001  SACT1   000010101  CLRRF   000011000  SACT2
                                00600   ;  000011110  TESTX   000100001  TMD     000100101  S/RDE   000101001  ISERV
                                00700   ;  000101110  RSYNC   000110011  PSI     000111010  TMARK   000111100  TFRF
                                00800   ;  000111110  CRAM7   001000011  SSERV   001010011  ZETA    001011101  RBUS1
                                00900   ;  001100010  CTEST   001101000  EPSIL   001101001  RRBCC   001101110  RBUS2
                                01000   ;  001110000  ORBC    001110101  ORCA    001111110  DDCMR   010000001  DDCM2
                                01100   ;  010000011  DISC    010000101  RBCC    010001100  CRBCO   010001111  NBCO
                                01200   ;  010010011  MCBCX   010011011  EBCC    010100000  EBCN    010100101  TBC1
                                01300   ;  010110011  MRTHA   010111000  TBC1X   010111010  TBC2    011000100  BCOCK
                                01400   ;  011010111  RMPEC   011011010  RNXMC   011011011  GAMMA   011011100  BETA
                                01500   ;  011100000  RNXM    011100010  NPRSO   011100100  TSERV   011101101  SIGMA
                                01600   ;  011110010  USCA    011110101  OXCB    011110111  RBUS3   100000000  PI
                                01700   ;  100001000  RBUS4   100001110  RDLE    100010000  RSSBN   100010001  RXBCC
                                01800   ;  100010010  ALPHA   100010101  UPBC    100011011  OPCA    100100011  USBC
                                01900   ;  100101001  OSCA    100110001  TNXMC   100110010  IOTA    100111001  TMPEC
                                02000   ;  100111010  OMEGA   101000000  XPBCO   101001000  RED     101001001  OTMB
                                02100   ;  101010000  XSBCO   101010111  CBCO    101011010  DELTA   101011101  ESS
                                02200   ;  101011111  C/GO    101100100  ITYPE   101100111  BCOCG   101101001  SIDLE
                                02300   ;  101101110  MU      101110011  SDLE    101111011  NU      101111101  CRAM2
                                02400   ;  110000010  SSBN    110000111  BCOSB   110001100  DDCMX   110001111  XBCC
                                02500   ;  110010110  SBC1    110100001  GOIDL   110100010  SBC2    110100101  C/LUI
                                02600   ;  110101011  CBINT   110110001  POER    110110010  CNACB
                                02700   ;
                                02800   ;                NUMBER OF TAGS FOUND:   91
                                02900   ;
                                03000   ;                LAST ROM ADDRESS USED: 110111010
                                03100   ;
                                03200   ;                INSTRUCTION      NUMBER OF TIMES USED.
                                03300   ;                -----------      ---------------------
                                03400   ;                  BRA                 78
                                03500   ;                  ALU                 25
                                03600   ;                  RAM                 124
                                03700   ;                  XFR                 67
                                03800   ;                  NPR                 4
                                03900   ;                  S/C                 65
                                04000   ;                  BCC                 4
                                04100   ;                  BRB                 76
                                04200   ;
                                04300   ENDROM:
                                04900   NPRLOC: 0
                                05000   COUNT:  0
                                05100   DATA:   0
                                05200   EXPRBA: 0
                                05300   EXPRWC: 0
```

```
4483                                                00100
4484                                                00200
4485  025550  051377  040505  020104  00600  EM1:    .ASCIZ  <377>/READ OF ROM PRODUCED A COMPARE ERROR/
      025616  021377  051106  042505  00700  EM2:    .ASCIZ  <377>/"FREE RUNNING" PRIMARY REGISTER ERROR./
      025666  042777  051122  051117  00800  EM2A:   .ASCIZ  <377>/ERROR! SEE LISTING FOR DETAILS./
      025727     377  043042  042522  00900  EM3:    .ASCIZ  <377>/"FREE RUNNING" SECONDARY REGISTER ERROR/
      026000  051377  046517  040440  01000  DH1:    .ASCIZ  <377>/ROM ADD  EXPECTED  FOUND/
      026032  042777  050130  041505  01100  DH2:    .ASCIZ  <377>/EXPECTED  FOUND   LINE(8) PRIMARY REG/
      026101     377  054105  042520  01200  DH2A:   .ASCIZ  <377>/EXPECTED  FOUND   LINE(8) SECONDARY REG/
                                      01300  .EVEN
      026152  000004                  01700  DT1:    4
4486  026154     006      004         01800          .BYTE   6,4
4487  026156  001272                  01900          SAVR5
4488  026160     006      002         02000          .BYTE   6,2
4489  026162  001270                  02100          SAVR4
4490  026164     002      006         02200          .BYTE   2,6
4491  026166  001260                  02300          SAVR0
4492  026170     002      001         02400          .BYTE   2,1
4493  026172  001262                  02500          SAVR1
4494                                  02600
4495  026174  000004                  02700  DT2:    4
4496  026176     006      004         02800          .BYTE   6,4
4497  026200  001272                  02900          SAVR5
4498  026202     006      002         03000          .BYTE   6,2
4499  026204  001270                  03100          SAVR4
4500  026206     002      006         03200          .BYTE   2,6
4501  026210  001260                  03300          SAVR0
4502  026212     006      001         03400          .BYTE   6,1
4503  026214  001262                  03500          SAVR1
4504                                  03600
4505  026216  000003                  03700  DT1A:   3
4506  026220     003      006         03800          .BYTE   3,6
4507  026222  001264                  03900          SAVR2
4508  026224     006      004         04000          .BYTE   6,4
4509  026226  001272                  04100          SAVR5
4510  026230     006      001         04200          .BYTE   6,1
4511  026232  001270                  04300          SAVR4
4512  026234                          04400  .ERRTAB:
4513  026234  025666                  04500          EM2A
4514  026236  000000                  04600          0
4515  026240  000000                  04700          0
4516  026242  025550                  04800          EM1
4517  026244  026000                  04900          DH1     ;HALT 1
4518  026246  026216                  05000          DT1A
4519  026250  025616                  05100          EM2
4520  026252  026032                  05200          DH2     ;HALT 2
4521  026254  026174                  05300          DT2
4522  026256  025727                  05400          EM3
4523  026260  026101                  05500          DH2A    ;HALT 3
4524  026262  026152                  05600          DT1
4525                                  05700  ;;*****************************************************************************
4526  026264                          05800  CORMAX:
4527          000001                  06300  .END
```

```
ADRCNT= 003443          1141*  1177*  1186*
ALU   = 010000           597*
ASYNC = 004000           605*  1826   1966   2103   2216   2252   2287   2366   2450   2534   2618   2697   3373
                        3402
AUTO.S  006622          1651*
BCC   = 060000           602*
BCCFBK  021022          3643*  3649*  3652*  3655   3659   3671*
BINWRD  003746          1227*  1228   1265*
BIT0  = 000001           595*  1408   1845   1866   1871   1985   2006   2011   2122   2126   3026   3346   3401
                        3425   3432   3481   3647
BIT1  = 000002           594*  1408   1419   1757   1767   3710   3713
BIT10 = 002000           585*  1408   1831   1971   2108   2221   2257   2292   2371   2455   2539   2623   2702
                        2715   2744   3409
BIT11 = 004000           584*  1408   1829   1835   1969   1975   2106   2112   2219   2255   2290   2305   2369
                        2375   2453   2459   2537   2543   2621   2627   2700   2706   3407   3413
BIT12 = 010000           583*   597    599    601    603   1829   1831   1835   1969   1971   1975   2106   2108
                        2112   2219   2221   2255   2257   2290   2292   2369   2371   2375   2453   2455   2459
                        2537   2539   2543   2621   2623   2627   2700   2702   2706   3299   3407   3409   3413
                        3710   3713
BIT13 = 020000           582*   598    599    602    603   1831   1971   2108   2221   2257   2292   2371   2455
                        2539   2623   2702   3405   3409   3413
BIT14 = 040000           581*   600    601    602    603   1048   1831   1971   2108   2221   2257   2292   2371
                        2455   2539   2623   2702   3199   3249   3299   3409   3710   3713
BIT15 = 100000           580*  1845   1856   1985   1996   2122   2128   2235   2270   2305   3149   3249   3299
                        3432   3446   3502   3564   3693
BIT2  = 000004           593*   975   1408   1758   1824   1964   2101   2214   2250   2285   2360   2444   2528
                        2603   2612   2694   2710   2725   2730   2732   2807   2906   3011   3400
BIT3  = 000010           592*  1408   1753   2351   2435   2449   2519   2533   2617   2988   3707
BIT4  = 000020           591*  2093   2098   2153   2207   2243   2278   2533   2617   3386   3393   3684
BIT5  = 000040           590*  2093   2098   2207   2243   2278   3386   3393   3684
BIT6  = 000100           589*  1719   3710   3713
BIT7  = 000200           588*  1042   1289   1446   1467   1719   1897   1964   2250   3432   3481   3544
BIT8  = 000400           587*  1408   1425   1754   1856   1996   3446   3502   3564
BIT9  = 001000           586*  1408   1719   1724   1829   1969   1996   2106   2219   2255   2270   2290   2369
                        2453   2537   2621   2700   3407   3507   3569   3572
BRB   = 070000           603*
BRW     003014           981   1070*
BRX     003016           982   1071*
CALBCC  021024          2389   2473   2557   3665*  3672*
CHRCNT  003744          1225*  1229   1245*  1263*  1264
CLKX    001242           674*
CLK.A   001412           774*  1569
CLK.B   001413           775*  1574
CLK.C   001414           776*  1579
CLK.D   001415           777*  1584
CNVRT = 104411           733*  1000   1002   1004   1006   1326   1328   1384   1441
CONVRT= 104410           731*   940   1342
CORMAX  026264          4526*  4527
COUNT   025540          2792*  2825*  2890*  2917*  2924*  2984*  3020*  4478*
CRC.CC= 102010          2516   2600   3675*
CRC16 = 120001          2432   3674*
CREAM   001306           695*   909*  1516*  1517   1519*  1524   1525*  1526   1529*
CSRMAP  006624           934   1653*
CYCLE   005666           984   1020   1021   1506*
DATA    025542          4479*
DATABP  004276          1315*  1318   1340   1343*
```

DZDVCC.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
DATACL= 104416          743#
DATAHD  004264         1314*  1336   1339#
DELAY = 104414          739*  1840   1980   2117   2228   2264   2299   2380   2464   2548   2632   2637   2712
                       2734   2738   2827   2926   3022   3421   3698
DEVADR  003440         1139*  1174   1184#
DH1     026000         4485*  4517
DH2     026032         4485*  4520
DH2A    026101         4485*  4523
DT1     026152         4485*  4524
DT1A    026216         4505*  4518
DT2     026174         4495*  4521
DVACTV  001300          689*   956*   957   1506   1511   1685*  1691*  1692*  1696   1715
DVCR00  001500          805#
DVCR01  001524          816#
DVCR02  001550          827#
DVCR03  001574          838#
DVCR04  001620          849#
DVCR05  001644          860#
DVCR06  001670          871#
DVCR07  001714          882#
DVLCR   001370          757#  1425*  1426   1545*  1546*  1547   3692*  3693*  3696
DVNSR   001402          762#  1555*  1556*  1557   1858   1868   1998   2008   2129   2232   2268   2303   2924
                       2923   3018   3444   3449
DVNUM   001301          690#   905   1010   1657*  1678*  1679   1686   1688
DVRIC   001366          756#  1543*  1544*  1545   1849   1851   1989   1991   3436   3437
DVRLVL  001354          751#  1560*  1561*  1562
DVRVEC  001352          750#  1027   1531*  1560
DVSCR   001362          754#  1024   1402*  1415*  1419*  1530*  1541   1753*  1757*  1758*  1767*  1836*  1838
                       1843   1863   1871*  1976*  1978   1983   2003   2011*  2093*  2098*  2113*  2115   2120
                       2126*  2135   2207*  2224*  2226   2243*  2260*  2262   2278*  2295*  2297   2376*  2379
                       2460*  2462   2544*  2546   2628*  2630   2707*  2715   2741   2820*  2822   2919*  2921
                       3017*  3026*  3386*  3393*  3417*  3419   3428   3431   3684*  3707*
DVSCRH  001364          755#  1541*  1542*  1543
DVSFR   001400          761#  1553*  1554*  1555   1754*  1760   3710*  3713*
DVSRA   001376          760#  1406   1551*  1552*  1553   1874   1881   1889   1994   1900   2015   2022   2030
                       2035   2041   2140   2149   2156   2384   2468   2552   2638   2710   2725*  2732   3034
                       3052   3070   3088   3425   3452   3460   3466   3473   3490   3528   3538   3543   3593
                       3603   3608   3615   3625   3683*  3686*
DVSRS   001372          758#  1405   1547*  1548*  1549   1816*  1873*  1885*  1956*  2013*  2026*  2092*  2139*
                       2206*  2242*  2277*  2353*  2437*  2521*  2605*  2686*  2796*  2895*  3000*  3032*  3050*
                       3068*  3086*  3372*  3451*  3525*  3590*
DVSRSH  001374          759#  1549*  1550*  1551   1875   1879*  1886*  1888   1992*  1898*  2014*  2016   2020*
                       2027*  2029   2033*  2039*  2147*  2155*  2383*  2390   2467*  2474   2551*  2558   2635*
                       2644   2709*  2724*  2731*  3033*  3037   3051*  3055   3069*  3073   3087*  3091   3424*
                       3458*  3464*  3471*  3478*  3526*  3536*  3541*  3591*  3601*  3606*  3614*  3619   3623*
                       3682*  3685*
DVTLVL  001360          753#  1564*  1565*
DVTR00  001502          806#
DVTR01  001526          817#
DVTR02  001552          828#
DVTR03  001576          839#
DVTR04  001622          850#
DVTR05  001646          861#
DVTR06  001672          872#
DVTR07  001716          883#
DVTVEC  001356          752#  1562*  1563*  1564
```

```
DV.END  001740            993#   1517   1526   1655
DV.MAP  001500            695     804#   909    936   1519   1529   1653   1658   1707
DV00.A  001504            807#
DV00.B  001510            809#
DV00.C  001514            811#
DV00.D  001520            813#
DV01.A  001530            818#
DV01.B  001534            820#
DV01.C  001540            822#
DV01.D  001544            824#
DV02.A  001554            829#
DV02.B  001560            831#
DV02.C  001564            833#
DV02.D  001570            835#
DV03.A  001600            840#
DV03.B  001604            842#
DV03.C  001610            844#
DV03.D  001614            846#
DV04.A  001624            851#
DV04.B  001630            853#
DV04.C  001634            855#
DV04.D  001640            857#
DV05.A  001650            862#
DV05.B  001654            864#
DV05.C  001660            866#
DV05.D  001664            868#
DV06.A  001674            873#
DV06.B  001700            875#
DV06.C  001704            877#
DV06.D  001710            879#
DV07.A  001720            884#
DV07.B  001724            886#
DV07.C  001730            888#
DV07.D  001734            890#
EM1     025550           4485#   4516
EM2     025616           4485#   4519
EM2A    025666           4485#   4513
EM3     025727           4485#   4522
ENDROM  025536           1765    4476#
ERRCNT  001232            666#    911*   1033   1351*
ERRFLG  001311            701#    907*    995*  1062*  1303*  1316   1330*  1385*
ERRMSG  004252           1313*   1331   1334#
ERTABO  004366           1328    1360#
EXIT  = 000205            605#   3689   3703   3722   3729
EXITER  004322           1346    1351#
EXPRBA  025544           3142*   3192*  3242*  3292*  3342*  3467   4480#
EXPRWC  025546           3143*   3193*  3243*  3293*  3343*  3474   4481#
FIX.00  006516           1570    1575   1580   1585   1619#
HALTS   004302           1299    1345#
HILIM   003436           1138*   1165   1183#
ICOUNT  001222            662#   1060   1065*
INBUF   005520           1108    1144   1491#
INIFLG  001310            700#    916    931*
INSTER= 104404            723#   1159
INSTR = 104403            721#   1592
INSTR2  003236           1115    1127#
```

```
LIGHT   000174            634*   926
LIGHTS  001200            645*   926*   997*
LIMITS  003364           1154   1165*
LINT1   020054           2834   2838   2842   2846   3497*
LINT2   020216           2933   2937   2941   2945   3559*
LOAD.M  021072           1828   1830   1834   1968   1970   1974   2105   2107   2111   2218   2220   2254   2256
                         2289   2291   2368   2370   2374   2452   2454   2458   2536   2538   2542   2620   2622
                         2626   2699   2701   2705   3404   3406   3408   3412   3691*
LOBITS  003442           1140*  1169   1185*  1186
LOCK    001220            661*  1064*  1078   1080   1322   1815*  1955*  2090*  2203*  2240*  2275*
LOGICA  002560            631   1014*
LOKFLG  001312            702*
LOLIM   003434           1137*  1167   1182*
LPCNT   001224            663*  1059*  1060   1063*
LRCB  = 000200           2348   3673*
LSTERR  001234            667*   912*   994*  1046*  1300   1302*  1386*
LOO.03  001416            779*  1532*  1567   1791   1931   2072   2185   2330   2414   2498   2582   2657   2779
                         2932   2877   2931   2971   3028   3125   3175   3225   3275   3325
LO4.07  001420            780*  1534*  1572   1795   1935   2076   2189   2334   2418   2502   2586   2671   2782
                         2836   2880   2935   2974   3046   3129   3179   3229   3279   3329
LO8.11  001422            781*  1536*  1577   1799   1939   2080   2193   2338   2422   2506   2590   2675   2785
                         2840   2883   2939   2977   3064   3133   3183   3233   3283   3333
L12.15  001424            782*  1538*  1582   1803   1943   2084   2197   2342   2426   2510   2594   2679   2788
                         2844   2886   2943   2980   3082   3137   3187   3237   3287   3337
MASKX   001244            675*
MASK.A  001406            769*  1568
MASK.B  001407            770*  1573
MASK.C  001410            771*  1578
MASK.D  001411            772*  1583
MASTEK  005400           1324   1482*
MCRLF   005104           1093   1216   1320   1321   1329   1470   1482*  1591   1609
MCSRX   005330            999   1482*
MDATA   005624           1243   1253   1495*
MEPASS  005145            998   1482*
MERRPC  005454           1327   1482*
MERRX   005355           1005   1482*
MERR2   005174           1482*  1508   1698
MERR3   005243            953   1482*
MLOCK   005301            977   1482*
MNEW    005402            948   1482*
MPASSX  005344           1003   1482*
MPFAIL  005107           1383   1482*
MQM     005100           1123   1482*  1614
MR      005171            985   1482*
MRESET= 004000            605*  1402   1415
MSTCLR= 104412            735*  1387   1752   1904   2045   2160   2205   2241   2276   2394   2478   2562   2648
                         2748
MTITLE  001000            643*   930
MTSTN   005366           1325   1482*  1593
MTSTPC  005267           1482*
MVECX   005336           1001   1482*
NEXT    001216            660*  1066   1356   1751*  1789*  1929*  2070*  2183*  2328*  2412*  2496*  2580*  2665*
                         2775*  2873*  2967*  3123*  3173*  3223*  3273*  3323*
NOLIST= ****** U          525   3739   4477
NPR   = 040000            600*
NPRLOC  025536           4477*
```

```
PARAM = 104405          725#  1594
PARAM1  003324         1143#  1160
PARBIT= 040000          605#  1639
PARERR  003360         1146   1149   1150   1159#  1166   1168   1170
PASCNT  001330          565#   906*   996*   997   1030
PC     =.000007         572#   934*  1015*  1045*  1292*  1471*  1570*  1575*  1580*  1595*  1601   1602*  1641*
                       1733*  1793*  1797*  1801*  1805*  1910*  1933*  1937*  1941*  1945*  2031*  2074*  2078*
                       2082*  2086*  2166*  2187*  2191*  2195*  2199*  2314*  2332*  2336*  2340*  2344*  2388*
                       2416*  2420*  2424*  2428*  2482*  2500*  2504*  2508*  2512*  2566*  2584*  2588*  2632*
                       2596*  2652*  2669*  2673*  2677*  2681*  2752*  3127*  3131*  3135*  3139*  3150*  3177*
                       3181*  3185*  3189*  3200*  3227*  3231*  3235*  3239*  3250*  3277*  3281*  3285*  3289*
                       3300*  3327*  3331*  3335*  3339*  3350*
PERFCR= 004537          605#  1817   1922   1828   1830   1834   1957   1962   1968   1970   1974   2094   2099
                       2105   2107   2111   2208   2212   2218   2220   2244   2248   2254   2256   2279   2283
                       2289   2291   2354   2359   2362   2368   2370   2374   2438   2442   2446   2452   2454
                       2458   2522   2526   2530   2536   2539   2542   2606   2610   2614   2620   2622   2626
                       2687   2691   2699   2701   2705   2720   2727   2800   2804   2899   2903   2907   3004
                       3008   3375   3380   3387   3394   3398   3404   3406   3408   3412   3416
PCTAB   004470         1384   1390#
PCPRO = 012600          579#  1350
PCP1SP= 005726          579#
PCP2SP= 022626          579#  1068
PS     = 177776         577#   902*   971*  1719*
PUSHRO= 010046          579#  1347
PUSH1S= 005746          579#
PUSH2S= 024646          579#
QV.FLG  001313          703#   909*  1009*  1057
RAM    = 020000          598#
RAMCLR= 104413          737#  1388   1908   1948   2089   2202   2347   2431   2515   2599   2684   2776   2974
                       2968   3371
REGBUF  021242         2821   2920   3504   3517   3566   3582   3731#
RESREG  004300         1341   1344#
RESTAR  004414         1371   1377#
RESTRT  002572         1008   1012   1020#
RESV16  001404          763#  1557*  1558*
RESOS = 104407          729#  1344
RETURN  001214          659#   914*   984*   986   1020*  1066*  1069   1356*  1359   1389   1608*  1616*  1617
ROMCLK= 104415          741#  3711   3714
ROMDAT  023750         1755   3738#
RUN     001304          693#   910*  1511   1514*  1515*  1522*  1523*
RXBA    022750         3142   3147   3192   3197   3292   3297   3342   3347   3736#
RXTAB   023350         3145   3195   3245   3345   3363   3367   3737#
RXTS1   017220         3144   3194   3244   3294   3344   3356#
RC     =.000000          565#   949*   957*   958*   960*   962*   964    965   1047   1053*  1067*  1200   1205*
                       1217   1230*  1234*  1244   1260*  1348*  1393   1394*  1395*  1397*  1423   1424*  1429*
                       1432*  1524*  1530   1531   1532   1533   1534   1535   1536   1537   1538   1539   1540*
                       1546   1548   1554   1556   1558   1561   1563   1565   1567*  1572*  1577*  1582*  1600*
                       1601   1604   1606   1608   1611   1612   1619   1638   1699*  1708*  1710*  1712   1730*
                       1721*  1755*  1759   1764   1765   1790*  1794*  1798*  1902*  1816   1855   1873   1995
                       1906*  1930*  1934*  1938*  1942*  1956   1995   2013   2026   2047*  2071*  2075*  2079*
                       2083*  2092   2127   2139   2162*  2184*  2188*  2192*  2196*  2206   2234   2242   2269
                       2277   2304   2310*  2329*  2333*  2337*  2341*  2353   2395*  2413*  2417*  2421*  2425*
                       2437   2479*  2497*  2501*  2505*  2509*  2521   2563*  2581*  2585*  2589*  2593*  2605
                       2649*  2666*  2670*  2674*  2678*  2686   2749*  2778*  2781   2784   2787   2790   2791*
                       2796   2809*  2810   2814*  2816*  2818   2876*  2879   2882   2985   2988   2989*  2995
                       2913*  2914   2970*  2973   2976   2979   2982   2993*  2995*  2997*  2998*  2992*  2994*
```

```
                    3490*  3499   3501   3529*  3530*  3531*  3532   3535*  3544*  3545   3553*  3554*  2561
                    3563   3594*  3595*  3596*  3597   3600*  3609*  3610   3616*  3617*  3618*  3620   3627*
                    3634*  3635*  3640   3641   3642   3669*  3680   3681   3683   3686   3692   3708
SAVACT  001302       691*   951   1696*
SAVNUM  001303       692*   905*  1007*  1010*  1689*
SAVPC   001276       698*  1191*  1362
SAVR0   001260       681*  1200*  1205   4491   4501
SAVR1   001262       682*  1199*  1206   4493   4503
SAVR2   001264       683*  1198*  1207   4507
SAVR3   001266       684*  1197*  1208
SAVR4   001270       685*  1196*  1209   4489   4499   4511
SAVR5   001272       686*  1195*  1210   4487   4497   4509
SAVSP   001274       687*
SAVUS = 104406       727*  1304
SCOPE = 104400       715*  1770   1806   1946   2087   2200   2345   2429   2513   2597   2682   2848   2947
                    3099   3140   3190   3240   3290   3340
SCOPE1 = 104401      717*  1905   2046   2161   2239   2274   2309
SERR.. 004640       1045   1292   1436*  1437
SETRES  021026      1817   1822   1957   1962   2094   2099   2208   2212   2244   2248   2279   2283   2354
                    2358   2362   2438   2442   2446   2522   2526   2530   2606   2610   2614   2697   2691
                    2720   2727   2800   2804   2899   2903   2907   3004   3008   3375   3380   3387   3394
                    3398   3678*
SETSCA  021132      3705*
SETSYN  021210      3416   3723*
SIMBOL  000646      2385   2469   2553   3637*
SP     =000006       571*   903*   918*   919*   925    928    929    972*  1041*  1042*  1043   1047*  1067
                    1080*  1086*  1087   1088*  1098   1102*  1103*  1104   1105*  1121   1122   1124*  1125*
                    1127   1128   1134*  1135*  1136   1142*  1179   1180   1191   1217*  1218*  1219*  1220*
                    1221*  1222   1223*  1256   1257   1258   1259   1260   1273*  1274*  1275*  1276*  1277*
                    1278*  1279*  1280   1288*  1289*  1290   1300   1302   1305   1348   1357*  1379*  1393*
                    1397   1403*  1404*  1410   1411   1423*  1432   1445*  1446*  1447   1452   1458   1460
                    1462*  1463   1466*  1467*  1468   1702*  1728   1730   1731*  3362*  3364*  3366   3369
                    3489   3501*  3553   3563*  3634   3637*  3638*  3639*  3666   3667   3669   3678*  3679*
                    3687   3688   3694*  3702   3706*  3718
SPACNT= 003745      1226*  1247   1250*  1264*
SSWR    000176       636*   927
STACK = 001200       578*   903    972   1357   1379
STAT    001236       672*  1791*  1795*  1799*  1803*  1826   1931*  1935*  1939*  1943*  1966   2072*  2076*
                    2080*  2084*  2103   2185*  2189*  2193*  2197*  2216   2252   2287   2330*  2334*  2333*
                    2342*  2366   2414*  2418*  2422*  2426*  2450   2498*  2502*  2506*  2510*  2534   2582*
                    2586*  2590*  2594*  2618   2667*  2671*  2675*  2679*  2697   3125*  3129*  3133*  3137*
                    3175*  3179*  3183*  3187*  3225*  3229*  3233*  3237*  3275*  3279*  3283*  3287*  3325*
                    3329*  3333*  3337*  3373   3402   3724   3726
SVC5    003452      1195*
SWR     001202       647*   922    927*   932    946    951    956    975   1039   1048   1055   1076   1093
                    1286   1293   1298   1345   1352   1354   1438   1439*  1449*  1454*  1455*  1456*  1463*
                    1586
SW00 = 000001        558*   946
SW01 = 000002        557*  1586
SW02 = 000004        556*
SW03 = 000010        555*
SW04 = 000020        554*
SW05 = 000040        553*
SW06 = 000100        552*
SW07 = 000200        551*
SW08 = 000400        550*  1352
```

```
SW09   = 001000             549*   1076
SW10   = 002000             548*   1354
SW11   = 004000             547*   1055
SW12   = 010000             546*   1089    1293
SW13   = 020000             545*   1298
SW14   = 040000             544*
SW15   = 100000             543*
SYNA00  001506              808*
SYNA01  001532              819*
SYNA02  001556              830*
SYNA03  001602              841*
SYNA04  001626              852*
SYNA05  001652              863*
SYNA06  001676              874*
SYNA07  001722              885*
SYNB00  001512              810*
SYNB01  001536              821*
SYNB02  001562              832*
SYNB03  001606              843*
SYNB04  001632              854*
SYNB05  001656              865*
SYNB06  001702              876*
SYNB07  001726              887*
SYNC    021346             3382    3724*   3725*   3729*   3732*
SYNCX   001240              673*
SYNC00  001516              812*
SYNC01  001542              823*
SYNC02  001566              834*
SYNC03  001612              845*
SYNC04  001636              856*
SYNC05  001662              867*
SYNC06  001706              878*
SYNC07  001732              889*
SYNC2A  001426              784*   1533*
SYNC2B  001430              785*   1535*
SYNC2C  001432              786*   1537*
SYNC2D  001434              787*   1539*
SYND00  001522              814*
SYND01  001546              825*
SYND02  001572              836*
SYND03  001616              847*
SYND04  001642              859*
SYND05  001666              869*
SYND06  001712              880*
SYND07  001736              891*
S.C    = 050000             601*
TEMP    005562             1230    1380*   1381*   1493*
TEMP1   001246              676*    936*    937     942*   1484    1715*   1716*   3640*   3663*
TEMP2   001250              677*    937*    938     1486   3641*   3645*
TEMP3   001252              678*   3642*   3644    3654*  3656    3661*   3662*   3665
TEMP4   001254              679*
TEMP5   001256              680*
TKCSR   001204              651*   1051    1110    1436   1443    1464
TKCBR   001206              652*   1041    1053    1112   1118    1288    1445    1466
T_AST  = 017046             1612   4482*
TPCSR   001210              653*   1094    1116    1295   1450
```

```
TPDBR   001212            654#  1096*  1118*  1297*  1452*
TRPOK   003762           1276#
TSTNO   001226            664#   913*  1365   1392   1597   1604   1606   1750*  1788*  1838*  2068*  2182*  2327*
                         2411*  2495*  2579*  2664*  2774*  2872*  2966*  3122*  3172*  3222*  3272*  3322*
TST1    007256           1600   1616   1750#
TST10   013154           2412   2495#
TST11   013520           2496   2579#
TST12   014064           2580   2664#
TST13   014506           2665   2774#
TST14   015034           2775   2872#
TST15   015400           2873   2966#
TST16   016176           2967   3122#
TST17   016350           3123   3172#
TST2    007372           1751   1788#
TST20   016522           3173   3222#
TST21   016674           3223   3272#
TST22   017046           3273   3322#  4482
TST23 = ****** U         3323
TST3    010160           1789   1928#
TST4    010754           1929   2068#
TST5    011434           2070   2182#
TST6    012244           2183   2327#
TST7    012610           2328   2411#
TTST    002702            978*   979*   981*   982*  1049#
TWOSYN= 010000            605#  3726
TXBAF   021350           1819   1821*  1872   2282   2350*  2356   2434*  2440   2518*  2524   2602*  2608   2688
                         2696*  2722   2793   2814   2818   2891   2992   2998   3377   3414*  3453   3529   3594
                         3733#
TXBAS   021750           1959   1961*  2012   2815   2892   3616   3734#
TXTAB   022350           1809   1820   1949   1960   2097   2211   2247   2351*  2361   2435*  2445   2519*  2529
                         2603*  2613   2693   2695*  2806   2813*  2905   2919*  2985   3010   3397   3415*  3735*
TYPCAT  004266           1319   1337   1340#
TYPE  = 104402            719#   930    935    948    953    977    985     38    999   1001   1003   1005   1093
                         1106   1123   1216   1253   1320   1321   1324   1325   1327   1329   1333   1338   1393
                         1440   1442   1470   1508   1591   1609   1614   1698
TYPMSG  004166           1317   1320#
VECMAF  007102           1697   1705#
WRDCNT  003742           1224*  1254*  1262#
WPKD.F  004254           1332   1335#
XBX     004060           1294   1296   1298#
XCSR    002604           1000   1022#
XERR    002626           1006   1031#
XFR   = 030000            599#
XHEAD   005461            935   1482#
XPASS   002620           1004   1028#
XPOLY   021020           2348*  2432*  2516*  2600*  3650   3661   3670#
XSTATS  005536            941   1482#
XTSTN   004374           1326   1363#
XVEC    002612           1002   1025#
$CRAP = 177777              1#  1736#  1746#  1773#  1784#  1913#  1924#  2054#  2065#  2169#  2178#  2317#  2323#
                         2401#  2407#  2485#  2491#  2569#  2575#  2654#  2660#  2755#  2770#  2850#  2868#  2949#
                         2962#  3103#  3118#  3153#  3168#  3203#  3218#  3253#  3268#  3303#  3318#
$E    = 000024              1#  1751   1752#  1789   1790#  1929   1930#  2070   2071#  2183   2194#  2328   2329#
                         2412   2413#  2496   2497#  2580   2581#  2665   2666#  2775   2776#  2873   2874#  2967
                         2968#  3123   3124#  3173   3174#  3223   3224#  3273   3274#  3323   3324#
$N    = 000022              1#  1736   1748   1752#  1773   1786   1790#  1913   1926   1930#  2054   2067   2071#
```

```
                          2169    2180    2184*   2317    2325    2329*   2401    2409    2413*   2485    2493    2437*   2559
                          2577    2581*   2654    2662    2566*   2755    2772    2776*   2560    2870    2874*   3343    3364
                          2968*   3103    3120    3124*   3153    3170    3174*   3203    3220    3224*   3253    3270    3274*
                          3303    3320    3324*   4482*
$Y      = 000017            1*    706*    715     717*    719*    721*    723*    725*    727*    729*    731*    733*    735*
                          737*    739*    741*    743*    745*
.       = 026264          616*    617     620*    627*    628*    629*    630*    633*    635*    638*    642*    644*    689*
                          690*    691*    692*    693*    694*    803*    805*    806*    807*    808*    809*    810*    811*
                          812*    813*    814*    816*    817*    819*    819*    820*    821*    822*    823*    824*    825*
                          827*    828*    829*    830*    831*    832*    833*    834*    835*    836*    838*    839*    840*
                          941*    842*    843*    844*    845*    946*    847*    849*    850*    851*    852*    853*    854*
                          855*    856*    857*    858*    860*    861*    862*    863*    864*    865*    866*    867*    868*
                          869*    871*    872*    873*    874*    875*    876*    877*    878*    879*    880*    882*    883*
                          884*    895*    896*    987*    888*    889*    990*    891*    955     1044    1291    1373    1382
                         1396    1434*   1444    1451    1465    1492*   1494*   1496*   1510    1701    1722    1902    2048
                         2164    2312    2716    2740    2802*   2819    2901*   2909*   3006*   3426    3429    3514    3555*
                         3579    3636*   3731*   3732*   3733*   3734*   3735*   3736*   3737*
.BEGIN  002332           971*
.CNVRT  003542           734     1217*
.CONVR  003536           732     1216*
.DATAC  004576           744     1422*
.DELAY  004476           740     1393*
.EOP    002436           993*    3323
.ERRTA  026234          1312     4512*
.HLT    004002           623     1285*
.INSTE  003224           724     1123*
.INSTR  003120           722     1102*
.INST1  003140          1106*    1126
.MSG    003142          1104*    1107*
.MSTCL  004556           736     1414*
.PARAM  003244           726     1134*
.PFAIL  004402           521      904    1370*   1378
.RAMCL  004516           738     1401*
.RESOS  003504           730     1205*
.ROMCL  004566           742     1418*
.SAVOS  003444           728     1191*
.SCOPE  002634           716     1038*
.SCOP1  003020           718     1076*
.START  001742           639      902*    914
.TRPSR  003750           625     1273*
.TRP1P  004014           717*    1278
.TYPE   003044           720     1085*
```

```
$VEND          1#    987
$VFRNT         1#    525
HLT          579#   1431    1763    1849    1854    1862    1967    1978    1884    1891    1896    1903    1988    1994    2002
            2007    2019    2025    2033    2037    2044    2125    2133    2138    2145    2152    2159    2231    2238    2257
            2273    2302    2308    2393    2477    2561    2647    2717    2747    2830    2929    3025    3040    3058    3078
            3094    3427    3430    3435    3443    3450    3457    3463    3470    3477    3484    3512    3534    3540    3547
            3577    3599    3605    3612    3622    3629    3701
LINE71       525#   3494    3556
LINE72       525#   3037    3045    3063    3081
$ALLOCN      525#   2776    2874    2968
$BUFFE         1#   1498
$CKIS          1#
$CKISO         1#
$CLR..         1#
$CYCLE         1#   1497
$EOP           1#    987
$FINI          1#   4482
$GETFL         1#
$GETPA         1#   1592
$HEADE         1#    525
$LC11        525#   1772    1912
$LC12        525#   2053
$LC14        525#   2158
$LC15        525#   2316    2400    2484    2568
$LC16        525#   2654
$MSG           1#   1482
$PFAIL         1#   1365
$RAMCL         1#   1393
$ROMCI       525#   1735
$RXSHI         1#
$RXTI        525#   3102    3152    3202    3252    3302
$SCOPE         1#   1034
$SETAS       525#   1826    1966    2103    2216    2252    2287    2366    2450    2534    2618    2697    3402
$SETLI         1#   1786    1926    2067    2180    2325    2409    2493    2577    2662    3120    3170    3220    3270    3320
$SETSC         1#   3704
$SETSY         1#   3723
$SET.T         1#
$SILOI         1#
$SIMBC         1#   3637
$TRPDE         1#    715     717     719     721     723     725     727     729     731     733     735     737     739     741
              743
$TSTN          1#   1748    1786    1926    2067    2180    2325    2409    2493    2577    2662    2772    2870    2964    3120
             3170   3220    3270    3320
$TXSHI         1#
$VARIA         1#    641
$WAITX       525#   2821    2920
$XZ            1#   1736    1746    1773    1784    1913    1924    2054    2065    2169    2178    2317    2323    2401    2407
             2485   2491    2569    2575    2654    2660    2755    2770    2850    2868    2949    2962    3103    3118    3153
             3168   3203    3218    3253    3268    3303    3318
```

J08


```
        2695  2696  2777  2791  2812  2813  2816  2817  2875  2889  2916  2918  2969  2983  2999
        3016  3415  3418  3454  3462  3527  3535  3592  3600  3609  3627  3643  3695
CLRB     907   908   995  1062  1252  1303  1385  1623  1657  1685  1811  1951  2795  2893  2894
        3526  3591  3728
CMP      925   938   964   965  1039  1060  1165  1167  1286  1300  1458  1460  1517  1526  1601
        1604  1606  1612  1626  1631  1655  1663  1682  1712  1713  1730  1751  1765  1846  1860
        1875  1901  1986  2000  2017  2042  2123  2131  2143  2150  2157  2236  2271  2306  2391
        2475  2559  2645  2745  2810  2818  2914  3014  3038  3056  3074  3092  3367  3384  3391
        3433  3441  3447  3455  3468  3475  3482  3508  3532  3545  3570  3573  3597  3610  3620
CMPB     951  1043  1114  1145  1147  1149  1153  1290  1447  1468  1679
COM     3649  3651
COMB     931
DEC     1119  1241  1254  1395  1693  1869  1907  2009  2048  2163  2311  2396  2480  2564  2640
        2650  2718  2739  2750  2825  2924  2989  2996  3020  3042  3060  3078  3096  3496  3510
        3518  3521  3551  3575  3583  3586  3632  3663  3716
DECB    1007  1177  1245  1250
EMT      579
HALT     617   950   954   959  1349  1372  1509  1700
INC      996  1059  1351  1381  1429  1542  1544  1550  1552  1721  1768  1836  1841  1880  1893
        1906  1976  1981  2021  2034  2047  2113  2118  2148  2162  2224  2229  2260  2255  2295
        2300  2310  2376  2381  2395  2460  2465  2479  2544  2549  2563  2628  2633  2649  2707
        2749  2808  2809  2820  2828  2911  2912  2913  2919  2927  2995  3012  3013  3017  3023
        3041  3059  3077  3095  3417  3422  3459  3472  3485  3520  3537  3549  3550  3585  3602
        3624  3630  3631  3699  3712  3715
INCB    1540  1678  1692  1812  1879  1892  1952  2020  2033  2147  3458  3471  3536  3601  3623
JMP      639   986  1021  1069  1280  1358  1389  1617  1697  1909  2050  2165  2313  3488
JSR      934  1015  1045  1292  1570  1575  1580  1585  1793  1797  1801  1805  1933  1937  1941
        1945  2074  2078  2082  2086  2187  2191  2195  2199  2332  2335  2340  2344  2385  2416
        2420  2424  2428  2469  2500  2504  2508  2512  2553  2584  2588  2592  2596  2669  2673
        2677  2681  2834  2838  2842  2846  2933  2937  2941  2945  3127  3131  3135  3139  3144
        3177  3181  3185  3189  3194  3227  3231  3235  3239  3244  3277  3281  3285  3289  3294
        3327  3331  3335  3339  3344
MOV      902   903   904   909   913   914   918   919   921   926   927   928   929   936   937
         960   961   962   971   972   978   979   981   982   984   997  1011  1020  1041  1047
        1053  1065  1066  1067  1080  1086  1087  1098  1102  1103  1104  1108  1109  1118  1121
        1122  1124  1125  1127  1128  1134  1135  1136  1137  1138  1139  1142  1144  1174  1175
        1179  1180  1191  1195  1196  1197  1198  1199  1200  1205  1206  1207  1208  1209  1210
        1217  1218  1219  1220  1221  1222  1224  1227  1228  1230  1231  1243  1256  1257  1258
        1259  1260  1273  1275  1279  1288  1302  1305  1307  1313  1314  1315  1348  1356  1357
        1371  1378  1379  1393  1394  1397  1402  1403  1404  1405  1406  1410  1411  1415  1423
        1426  1432  1438  1439  1445  1452  1466  1519  1524  1529  1530  1531  1532  1533  1534
        1535  1536  1537  1538  1539  1540  1541  1543  1545  1547  1549  1551  1553  1555  1557
        1560  1562  1564  1567  1568  1569  1572  1573  1574  1577  1578  1579  1582  1583  1584
        1600  1602  1608  1616  1619  1653  1658  1659  1660  1668  1684  1695  1702  1705  1706
        1707  1708  1709  1710  1711  1719  1731  1750  1751  1753  1754  1755  1757  1759  1760
        1788  1789  1790  1791  1794  1795  1798  1799  1802  1803  1809  1814  1815  1843  1844
        1845  1851  1852  1855  1857  1858  1859  1863  1864  1866  1872  1873  1874  1881  1895
        1889  1894  1897  1900  1928  1929  1930  1931  1934  1935  1938  1939  1942  1943  1949
        1954  1955  1983  1984  1985  1991  1992  1995  1997  1998  1999  2003  2004  2006  2012
        2013  2015  2022  2026  2030  2035  2038  2041  2069  2070  2071  2072  2075  2076  2079
        2080  2083  2084  2090  2091  2092  2120  2121  2122  2127  2129  2130  2135  2136  2139
        2140  2141  2146  2149  2153  2154  2156  2182  2183  2184  2185  2188  2189  2192  2193
        2196  2197  2203  2204  2206  2232  2233  2234  2240  2242  2268  2269  2275  2277  2303
        2304  2327  2328  2329  2330  2333  2334  2337  2338  2341  2342  2348  2349  2352  2353
        2384  2389  2411  2412  2413  2414  2417  2419  2421  2422  2425  2426  2432  2433  2436
        2437  2468  2473  2495  2496  2497  2498  2501  2502  2505  2506  2509  2510  2516  2517
```

K08


```
        2520  2521  2552  2557  2579  2580  2581  2582  2585  2586  2589  2590  2593  2594  2600
        2601  2604  2605  2636  2638  2664  2665  2666  2667  2670  2671  2674  2675  2678  2679
        2685  2686  2708  2725  2737  2741  2742  2743  2774  2775  2778  2792  2793  2794  2796
        2814  2815  2821  2824  2872  2873  2876  2890  2891  2892  2895  2920  2923  2966  2967
        2970  2984  2985  2986  2991  2992  2993  2998  3000  3030  3031  3032  3034  3035  3048
        3049  3050  3052  3053  3066  3067  3068  3070  3071  3084  3085  3086  3088  3089  3122
        3123  3124  3125  3128  3129  3132  3133  3136  3137  3142  3143  3172  3173  3174  3175
        3178  3179  3182  3183  3186  3187  3192  3193  3222  3223  3224  3225  3228  3229  3232
        3233  3236  3237  3242  3243  3272  3273  3274  3275  3278  3279  3282  3283  3286  3287
        3292  3293  3322  3323  3324  3325  3328  3329  3332  3333  3336  3337  3342  3343  3357
        3358  3359  3360  3361  3362  3363  3364  3370  3414  3431  3432  3436  3437  3438  3444
        3445  3449  3451  3452  3453  3460  3466  3467  3473  3474  3479  3480  3481  3489  3499
        3500  3501  3503  3504  3505  3506  3515  3516  3517  3523  3524  3525  3528  3529  3538
        3542  3543  3544  3553  3561  3562  3563  3565  3566  3567  3568  3580  3581  3582  3588
        3589  3590  3593  3594  3603  3607  3608  3615  3616  3625  3634  3637  3638  3639  3640
        3641  3642  3644  3650  3655  3656  3657  3665  3666  3667  3668  3678  3679  3683  3686
        3687  3688  3692  3694  3702  3706  3708  3710  3713  3718  3719
MOVB     905   910   956   957  1009  1010  1095  1112  1140  1141  1225  1226  1229  1234  1244
        1249  1297  1330  1624  1628  1629  1633  1634  1636  1637  1688  1699  1696  1715  1816
        1875  1886  1888  1898  1899  1956  2014  2016  2027  2029  2039  2040  2155  2350  2351
        2383  2390  2434  2435  2467  2474  2518  2519  2551  2558  2602  2603  2635  2644  2709
        2724  2731  2988  2994  3033  3037  3051  3055  3069  3073  3087  3091  3366  3372  3424
        3464  3465  3478  3541  3606  3614  3619  3680  3681  3682  3685  3724  3725
NOP      923   978   979  1016  1017  1018  1019  2797  2798  2799  2896  2897  2898  3001  3002
        3003
RESET    993  1013  1652
ROL     1454  1455  1456
ROLB    1514  1522  1691
ROR     1236  1238  1240  1716  3645  3654  3721
RORB    1427
RTI     1081  1099  1129  1181  1201  1211  1261  1359  1398  1412  1416  1420  1433  1703  1732
RTS     1471  1641  1733  1910  2051  2166  2314  2398  2482  2566  2652  2752  3150  3200  3250
        3300  3350  3490  3554  3635  3669
SUB     1274  1306
SWAB    3439
TRAP     715   717   719   721   723   725   727   729   731   733   735   737   739   741   743
TST      922   944   973  1078  1318  1322  1331  1336  1340  1345  1589  1611  1621  1661  1665
        1669  1764  1838  1849  1868  1978  1989  2008  2115  2226  2262  2297  2378  2462  2546
        2630  2779  2782  2785  2788  2822  2832  2836  2840  2844  2877  2880  2883  2986  2921
        2931  2935  2939  2943  2971  2974  2977  2980  3018  3028  3046  3064  3082  3369  3513
        3578  3696
TSTB     916   932  1051  1057  1091  1094  1110  1116  1247  1295  1316  1443  1450  1464  1505
        1686  3419  3428
.ASCIZ   643  1473  1480  1482  4485
.BLKB    689   690   691   692   693  3733  3734  3735  3736  3737
.BLKW    628   629   630   805   806   807   808   809   810   811   812   813   814   815   817
         818   819   820   821   822   823   824   825   827   828   829   830   831   832   833
         834   835   836   838   839   840   841   842   843   844   845   846   847   849   850
         851   852   853   854   855   856   857   858   860   861   862   863   864   865   866
         867   868   869   871   872   873   874   875   876   877   878   879   880   882   883
         884   885   886   887   888   889   890   891  1434  2802  2901  2909  3006  3555  3636
        3731  3732
.BYTE    700   701   702   703   769   770   771   772   774   775   776   777  1023  1026  1029
        1032  1361  1364  1391  1477  1483  1485  1598  1599  1818  1823  1958  1963  2095  2100
        2209  2213  2245  2249  2280  2284  2355  2359  2363  2439  2443  2447  2523  2527  2531
        2607  2611  2615  2688  2692  2721  2728  2801  2805  2900  2904  2908  3005  3009  3376
```

DZDVC0.P11 CROSS REFERENCE TABLE -- PERMANENT SYMBOLS

```
                3381    3398    3395    3399    4486    4488    4490    4492    4496    4498    4500    4502    4506    4508    4510
.ENABL           539
.END            4527
.ENDC           1735    1737    1747    1752    1774    1783    1785    1790    1828    1914    1923    1925    1930    1968    2055
                2066    2071    2105    2170    2173    2184    2218    2254    2289    2318    2324    2329    2368    2390    2402
                2408    2413    2452    2474    2486    2492    2497    2536    2558    2570    2576    2581    2620    2644    2655
                2661    2666    2699    2756    2771    2776    2851    2869    2874    2950    2963    2968    3104    3106    3115
                3116    3119    3124    3143    3144    3154    3155    3156    3165    3166    3169    3174    3193    3194    3204
                3205    3206    3215    3216    3219    3224    3243    3244    3254    3255    3256    3265    3266    3269    3274
                3293    3294    3304    3305    3306    3315    3316    3319    3324    3343    3344    3406    3549    3630    3739
                4477    4527
.EQUIV           579
.EVEN            694     705    1475    1481    1482    1487    4485
.IF              525    1736    1746    1751    1773    1777    1784    1789    1828    1913    1917    1924    1929    1968    2054
                2065    2070    2105    2169    2178    2183    2218    2254    2289    2317    2320    2323    2328    2368    2401
                2404    2407    2412    2452    2485    2488    2491    2496    2536    2569    2572    2575    2580    2620    2654
                2660    2665    2699    2755    2770    2775    2850    2868    2873    2949    2962    2967    3103    3105    3106
                3114    3115    3118    3123    3142    3143    3153    3155    3156    3164    3165    3168    3173    3192    3193
                3203    3205    3206    3214    3215    3218    3223    3242    3243    3253    3255    3256    3264    3265    3268
                3273    3292    3293    3303    3305    3314    3315    3318    3323    3342    3343    3404    3497    3559    3739
                4477    4527
.IFF             1737    1746    1751    1752    1774    1783    1784    1789    1790    1914    1917    1924    1929    1930    2055
                2065    2070    2071    2170    2178    2183    2184    2318    2320    2323    2328    2329    2384    2385    2402
                2404    2407    2412    2413    2468    2469    2486    2488    2491    2496    2497    2552    2553    2570    2572
                2575    2580    2581    2643    2644    2655    2660    2665    2666    2756    2770    2775    2776    2851    2869
                2873    2874    2950    2962    2967    2968    3104    3114    3115    3118    3123    3124    3142    3143    3154
                3165    3166    3168    3173    3174    3193    3194    3204    3214    3215    3218    3223    3224    3242    3243
                3254    3264    3265    3268    3273    3274    3292    3293    3304    3314    3315    3318    3323    3342    3343
                3498    3507    3510    3517    3544    3549    3559    3567    3572    3581    3609    3614
.IFT             525    1735    1736    1746    1773    1777    1784    1913    1917    1924    2054    2065    2169    2178    2317
                2321    2323    2384    2385    2401    2405    2407    2468    2469    2485    2489    2491    2552    2553    2569
                2572    2575    2636    2643    2654    2660    2755    2770    2850    2868    2949    2962    3103    3114    3115
                3118    3142    3143    3153    3164    3165    3168    3192    3193    3203    3214    3215    3218    3242    3243
                3253    3264    3265    3268    3292    3293    3303    3314    3315    3318    3342    3343    3497    3505    3516
                3544    3559    3567    3581    3610
.IFTF            525    2321    2405    2489    2573    3498    3507    3510    3517    3545    3560    3569    3575    3582    3610
.IIF            1751    1752    1789    1790    1791    1795    1799    1803    1823    1824    1856    1857    1874    1929    1930
                1931    1935    1939    1943    1963    1996    2014    2070    2071    2072    2076    2080    2084    2183    2184
                2185    2189    2193    2197    2328    2329    2330    2334    2338    2342    2412    2413    2414    2418    2422
                2426    2496    2497    2498    2502    2506    2510    2580    2581    2582    2586    2590    2594    2665    2666
                2667    2671    2675    2679    2775    2776    2873    2874    2967    2968    3123    3124    3125    3129    3133
                3137    3173    3174    3175    3179    3183    3187    3223    3224    3225    3229    3233    3237    3273    3274
                3275    3279    3283    3287    3323    3324    3325    3329    3333    3337
.IRP             715     717     719     721     723     725     727     729     731     733     735     737     739     741     743
                 805    1736    1748    1751    1773    1786    1789    1913    1926    1929    2054    2067    2070    2169    2180
                2183    2317    2325    2328    2401    2409    2412    2485    2493    2496    2569    2577    2580    2654    2662
                2665    2755    2772    2775    2850    2870    2873    2949    2964    2967    3103    3120    3123    3153    3170
                3173    3203    3220    3223    3253    3270    3273    3303    3320    3323    4482
.LIST               1     525     539     560     579     605     607     617     641     643     717     719     721     723     725
                 727     729     731     733     735     737     739     741     743     745     800     894     987    1035    1492
                1735    1736    1746    1752    1773    1784    1790    1913    1924    1930    2054    2065    2071    2169    2178
                2184    2317    2323    2329    2401    2407    2413    2485    2491    2497    2569    2575    2581    2654    2660
                2666    2755    2770    2776    2850    2868    2874    2949    2962    2968    3103    3118    3124    3153    3168
                3174    3203    3218    3224    3253    3268    3274    3303    3318    3324    3739    4477    4482    4485
.MACRO              1     525
.NLIST              1     525     539     560     579     605     607     617     641     643     717     719     721     723     725
```

```
          727     729     731     733     735     737     739     741     743     745     800     894     987    1035    1482
         1735    1736    1746    1752    1773    1784    1790    1913    1924    1930    2054    2065    2071    2169    2178
         2184    2317    2323    2329    2401    2407    2413    2485    2491    2497    2569    2575    2581    2654    2660
         2666    2755    2770    2776    2850    2868    2874    2949    2962    2968    3103    3118    3124    3153    3162
         3174    3203    3218    3224    3253    3268    3274    3303    3318    3324    3739    4477    4482    4485
.PAGE     560     607     696     748     800     894     987    1497    1735    3739
.REM        1     894
.REPT     617
.SBTTL    525     560     607     641     894     987    1035    1735
.TITLE    539
.WORD    1479


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *,DZDVCC.SEQ/SOL/CRF/DS:ERFZ=DZDVCC.MAC,DZDVCC.P11
 RUN-TIME: 31 50 6 SECONDS
 RUN-TIME RATIO: 242/89=2.7
 CORE USED:  23K  (45 PAGES)
```

Spooler runtime 19 Seconds, 83 KCS, 519 disk reads, 3 disk writes, 103 pages