DV11

STATIC LINE CARD TEST

MD-11-DZDVB-B

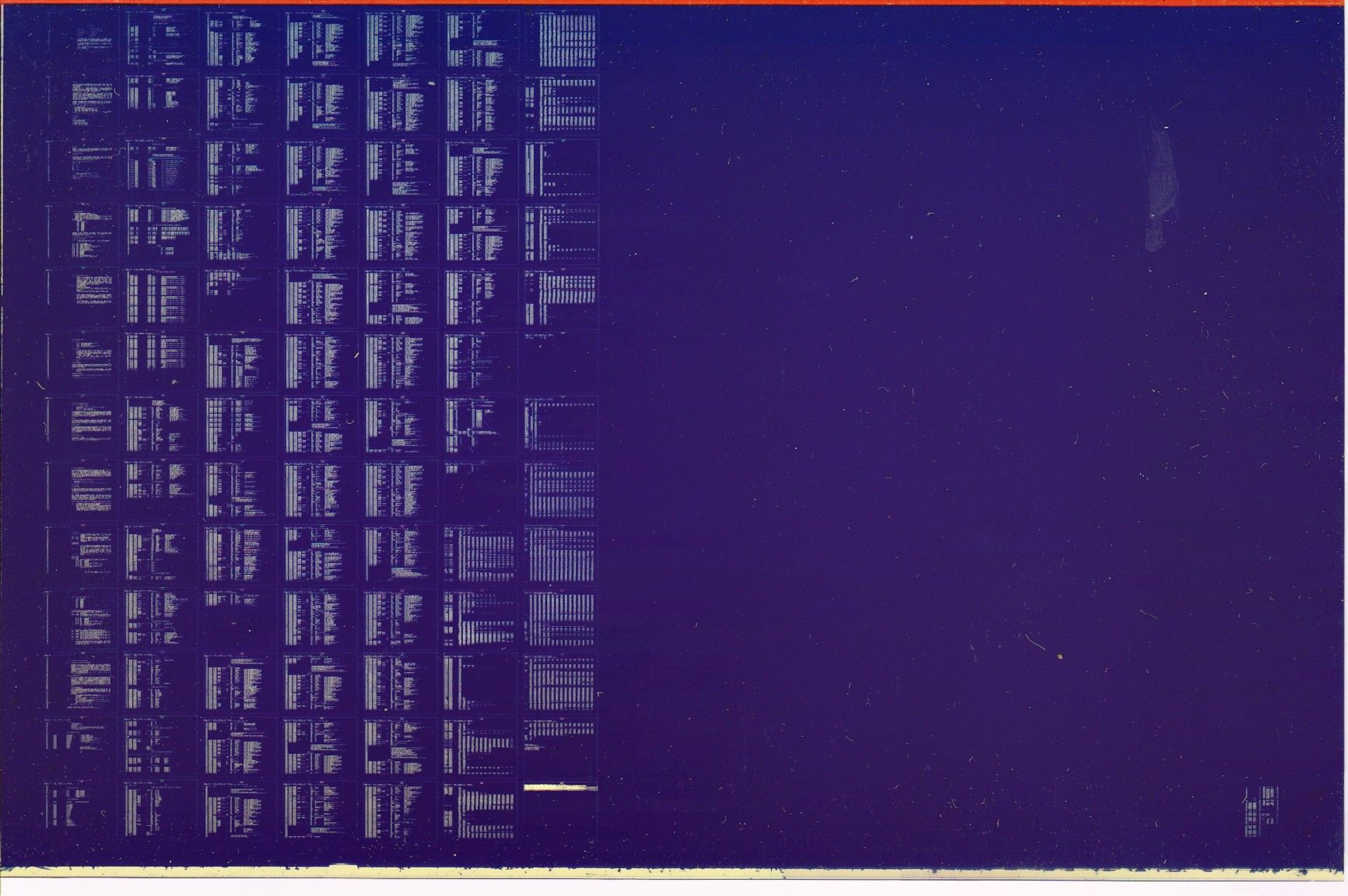
EP-DZDVB-B-DL-A

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA



IDENTIFICATION

PRODUCT CODE: MAINDEC-11-DZDVB-B-D

PRODUCT NAME: STATIC LINE CARD TESTS

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 ENVIRONMENT.
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).

DZDVB EXERCISES ALL EXISTING LINE CARDS IN A STATIC STATE (MICRO PROCESSOR IS NEVER TURNED ON). TRANSMITTER AND RECEIVER FLAGS,TRANSMITTER AND RECEIVER DATA,RECEIVER SYNCING AND CHAR SILO ARE TESTED. MOST TESTS EXERCISE A "GROUP" OF 4 LINES AT A TIME (00-03,04-07,08-11,12-15). FOR EASE OF TROBLESHOOTING; ONLY ONE LINE CARD MAY BE INSTALLED AND BY ALERTING THE DIAGNOSTIC AS TO WHICH LINE CARDS ARE PHYSICALLY REMOVED (SEE SECTION 8.4A) PROGRAM WILL RUN ANY COMBINATION OF LINE CARDS.

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 READ 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 INTERMITTEMT 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 DZDVE WHICH IS: 000200 FOR THE MODEM CONTROL AND CABLE TESTS AND 000210 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 POSSIBILY 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 DZDVB-B 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.

3.4 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 UNDEP 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 BIT15 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 CARDS.
5) SYNC "A"=226 AND SYNC "B"=062

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

```
;*MAINDEC-11-DZDVB-B*(377)*STATIC LINE CARD TESTS
;*COPYRIGHT 1972, DIGITAL EQUIPMENT CORP., MAYNARD, MASS. 01754
;*-----------------------------------------------------------------
```

```
                                        ;STARTING PROCEDURE
                                        ;LOAD PROGRAM
                                        ;LOAD ADDRESS 000200
                                        ;PRESS START
                                        ;PROGRAM WILL TYPE "MAINDEC-11-DZDVB-B/<377>/STATIC LINE CARD TESTS "
                                        ;PROGRAM WILL TYPE "R" TO INDICATE THAT TESTING HAS STARTED
                                        ;AT THE END OF A PASS, PROGRAM WILL TYPE PASS COMPLETE MESSAGE
                                        ;AND THEN RESUME TESTING


                                ;SWITCH REGISTER OPTIONS
                                ;----------------------

  100000                        SW15=100000             ;=1,HALT ON ERROR
  040000                        SW14=40000              ;=1,LOOP ON CURRENT TEST
  020000                        SW13=20000              ;=1,INHIBIT ERROR TYPEOUT
  010000                        SW12=10000              ;=1,DELETE TYPEOUT/BELL ON ERROR.
  004000                        SW11=4000               ;=1,INHIBIT ITERATIONS
  002000                        SW10=2000               ;=1,ESCAPE TO NEXT TEST ON ERROR
  001000                        SW09=1000               ;=1,LOOP WITH CURRENT DATA
  000400                        SW08=400                ;=1,LOOP ON ERROR
  000200                        SW07=200                ;=1, DO "AUTO SIZING" ON INITAL START UP.
  000100                        SW06=100
  000040                        SW05=40
  000020                        SW04=20
  000010                        SW03=10
  000004                        SW02=4                  ;LOCK ON TEST SELECT
  000002                        SW01=2                  ;RESTART PROGRAM AT SELECTED TEST
  000001                        SW00=1                  ;RESELECT DV11 DESIRED ACTIVE
                                                        ;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
                                        .PFAIL                  ;POWER FAIL HANDLER
                                        340                     ;SERVICE AT LEVEL 7
                                        .+2                     ;ERROR HANDLER
                                        340                     ;SERVICE AT LEVEL 7
                                        .TRPSRV                 ;GENERAL HANDLER DISPATCH SERVICE
                                        340                     ;SERVICE AT LEVEL 7
                                        .=40
                                        .BLKW 1                 ;SAVE FOR ACT-11 OR DDP2
                                        .BLKW 1                 ;RETURN ADDRESS IF UNDER ACT-11 OR DDP2
                                        .BLKW 1                 ;SAVE FOR ACT-11 OR DDP2
                                        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  005377  040515  047111  MTITLE: .ASCIZ  <377><12><MAINDEC-11-D2DVB-B <377> STATIC LINE CARD TESTS  <377><

                 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  LOCK:   0               ;ADDRESS FOR LOCK ON CURRENT DATA
```

```
                                ICOUNT: 3                       ;NUMBER OF ITERATIONS THAT CURRENT TEST WILL BE EXECUTED
                                LPCNT:  0                       ;NUMBER OF ITEREATIONS COMPLETED
                                TSTNO:  0                       ;NUMBER OF TEST IN PROGRESS
                                PASCNT: 0                       ;NUMBER OF PASSES COMPLETED
                                ERRCNT: 0                       ;TOTAL NUMBER OF ERRORS
                                LSTERR: 0                       ;PC OF LAST ERROR CALL

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

                                STAT:   0                       ;DV STATUS WORD STORAGE
                                SYNCX:  0
                                CLKX:   0
                                MASKX:  0
                                TEMP1:  0                       ;TEMPORARY STORAGE
                                TEMP2:  0                       ;TEMPORARY STORAGE
                                TEMP3:  0                       ;TEMPORARY STORAGE
                                TEMP4:  0                       ;TEMPORARY STORAGE
                                TEMP5:  0                       ;TEMPORARY STORAGE
                                SAVR0:  0                       ;R0 STORAGE
                                SAVR1:  0                       ;R1 STOPAGE
                                SAVR2:  0                       ;R2 STORAGE
                                SAVR3:  0                       ;R3 STORAGE
                                SAVR4:  0                       ;R4 STORAGE
                                SAVR5:  0                       ;R5 STORAGE
                                SAVSP:  0                       ;STACK POINTER STORAGE
                                SAVPC:  0                       ;PROGRAM COUNTER STORAGE
                                DVACTV: .BLKB 1                 ;DV11'S SELECTED ACTIVE.
                                DVNUM:  .BLKB 1                 ;OCTAL NUMBER OF DV11'S.
                                SAVACT: .BLKB 1                 ;ORIGINAL ACTV.  DEVICES.
                                SAVNUM: .BLKB 1                 ;WORKABLE NUMBER.
                                RUN:    .BLKB 1                 ;POINTER ONE PAST RUNNING DEVICE.
                                .EVEN
                                CREAM:  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 DV11 ITERATIONS WILL BE SUPPRESSE

                 .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  SAVO5=TRAP+6            ;CALL TO REGISTER SAVE ROUTINE
001330  003444          .SAVO5
        104407  RESO5=TRAP+7            ;CALL TO REGISTER RESTORE ROUTINE
001332  003504          .RESO5
        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.


                                        ;DV1. 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     5 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
```

```
   858  001636  000001          SYNC04: .BLKW 1         ;SYNC TWO
   859  001640  000001          DV04.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 04
   860  001642  000001          SYND04: .BLKW 1         ;SYNC TWO
   861
   862  001644  000001          DVCR05: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 05
   863  001646  000001          DVTR05: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 05
   864  001650  000001          DV05.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 05
   865  001652  000001          SYNA05: .BLKW 1         ;SYNC TWO
   866  001654  000001          DV05.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 05
   867  001656  000001          SYNB05: .BLKW 1         ;SYNC TWO
   868  001660  000001          DV05.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 05
   869  001662  000001          SYNC05: .BLKW 1         ;SYNC TWO
   870  001664  000001          DV05.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 05
   871  001666  000001          SYND05: .BLKW 1         ;SYNC TWO
   872
   873  001670  000001          DVCR06: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 06
   874  001672  000001          DVTR06: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 06
   875  001674  000001          DV06.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 06
   876  001676  000001          SYNA06: .BLKW 1         ;SYNC TWO
   877  001700  000001          DV06.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 06
   878  001702  000001          SYNB06: .BLKW 1         ;SYNC TWO
   879  001704  000001          DV06.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 06
   880  001706  000001          SYNC06: .BLKW 1         ;SYNC TWO
   881  001710  000001          DV06.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 06
   882  001712  000001          SYND06: .BLKW 1         ;SYNC TWO
   883
   884  001714  000001          DVCR07: .BLKW 1         ;CONTROL STATUS REGISTER FOR DV11 NUMBER 07
   885  001716  000001          DVTR07: .BLKW 1         ;VECTOR "A" FOR DV11 NUMBER 07
   886  001720  000001          DV07.A: .BLKW 1         ;PARAMETER FOR LINES 00-03 FOR DV11 NUMBER 07
   887  001722  000001          SYNA07: .BLKW 1         ;SYNC TWO
   888  001724  000001          DV07.B: .BLKW 1         ;PARAMETER FOR LINES 04-07 FOR DV11 NUMBER 07
   889  001726  000001          SYNB07: .BLKW 1         ;SYNC TWO
   890  001730  000001          DV07.C: .BLKW 1         ;PARAMETER FOR LINES 08-11 FOR DV11 NUMBER 07
   891  001732  000001          SYNC07: .BLKW 1         ;SYNC TWO
   892  001734  000001          DV07.D: .BLKW 1         ;PARAMETER FOR LINES 12-15 FOR DV11 NUMBER 07
   893  001736  000001          SYND07: .BLKW 1         ;SYNC TWO
   894
   895  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  001226          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  005432                  TYPE    ,MNEW                   ;TYPE THE MESSAGE.
002242  005000                          CLR     R0                      ;ZERO DATA LIGHTS
```

```
   952  002244  000000                          HALT                    ;WAIT FOR USER TO TELL WHAT DEVICES TO RUN
   953  002246  127737  176730  001302          CMPB    @SWR,SAVACT     ;IS THE NUMBER VALID?
   954  002254  101404                          BLOS    2$              ;BR IF NUMBER IS OK.
   955  002256  104402  005243                  TYPE    ,MERR3          ;TELL USER OF INVALID NUMBER.
   956  002262  000000                          HALT                    ;STOP EVERY THING.
   957  002264  000776                          BR      .-2             ;RESTART THE PROGRAM AGAIN.
   958  002266  117737  176710  001300  2$:     MOVB    @SWR,DVACTV     ;GET NEW DEVICE PATTERN
   959  002274  113700  001300                  MOVB    DVACTV,R0       ;SHOW THE USER WHAT HE SELECTED.
   960  002300  042700  177400                  BIC     #^C<377>,R0     ;USE ONLY LOW BYTE.
   961  002304  000000                          HALT                    ;CONTINUE DYNAMIC SWITCHES.
   962  002306  012700  000300          3$:     MOV     #300,R0         ;PREPARE TO CLEAR THE FLOATING
   963  002312  012701  000302                  MOV     #302,R1         ;VECTOR AREA. 300-776
   964  002316  010120                  4$:     MOV     R1,(R0)+        ;START PUTTING "PC+2 - HALT"
   965  002320  005021                          CLR     (R1)+           ;IN VECTOR AREA.
   966  002322  022021                          CMP     (R0)+,(R1)+     ;POP POINTERS
   967  002324  022700  001000                  CMP     #1000,R0        ;ALL DONE??
   968  002330  001372                          BNE     4$              ;BR IF NO.
   969
   970                                          ;TEST START AND RESTART
   971                                          ;---------------------
   972
   973  002332  012737  000340  177776  .BEGIN: MOV     #340,PS         ;LOCK OUT INTERRUPTS
   974  002340  012706  001200                  MOV     #STACK,SP       ;SET UP STACK
   975  002344  005737  000042                  TST     @#42            ;IS PROGRAM UNDER MONITOR CONTROL
   976  002350  001023                          BNE     3$              ;BR IF YES
   977  002352  032777  000004  176622          BIT     #BIT2,@SWR      ;CHECK FOR LOCK ON TEST
   978  002360  001411                          BEQ     1$              ;BR IF NO LOCK DESIRED.
   979  002362  104402  005301                  TYPE    ,MLOCK          ;TYPE LOCK SELECTED.
   980  002366  012737  000240  002702          MOV     #NOP,TTST       ;ADJUST SCOPE ROUTINE.
   981  002374  012737  000240  002704          MOV     #NOP,TTST+2     ;SET UP TO LOCK
   982  002402  000406                          BR      2$              ;CONTINUE ALONG.
   983  002404  013737  003014  002702  1$:     MOV     BRW,TTST        ;PREPARE NORMAL SCOPE ROUTINE
   984  002412  013737  003016  002704          MOV     BRX,TTST+2      ;LOCK NOT SELECTED. SET UP FOR NORMAL SCOPE LOOP
   985  002420                           2$:
   986  002420  012737  005666  001214  3$:     MOV     #CYCLE,RETURN   ;START AT "CYCLE" FIND WHICH DEVICE TO TEST
   987  002426  104402  005171          4$:     TYPE    ,MR             ;TYPE R
   988  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  176515              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
```

```
 1045  002654  122726  000007                    CMPB    #7,(SP)+        ;WAS IT CNTRL 'G' ?
 1046  002660  001002                            BNE     .+6             ;BR IF NO.
 1047  002662  004737  004640                    JSR     PC,SERV.G       ;SERVICE "CNTRL 'G'".
 1048  002666  005037  001234          64$:      CLR     LSTERR          ;CLEAR LAST ERROR PC.
 1049  002672  010016                            MOV     R0,(SP)         ;SAVE R0 ON THE STACK
 1050  002674  032777  040000  176300            BIT     #BIT14,@SWR     ;"LOOP ON THIS TEST"?
 1051  002702  001407                  TTST:     BEQ     1$              ;BR IF NO.  (IF LOCK SW01=1; THIS LOC =240)
 1052  002704  000437                            BR      3$              ;GOTO 3$    (IF LOCK SW01=1; THIS LOC =240)
 1053  002706  105777  176272                    TSTB    @TKCSR          ;KEYBOAD DONE?
 1054  002712  100034                            BPL     3$              ;BR IF NO. (LOCK: HIT KEY TO GOTO NEXT TEST)
 1055  002714  017700  176266                    MOV     @TKDBR,R0       ;CLEAR DONE BIT
 1056  002720  000415                            BR      2$              ;CONTINUE
 1057  002722  032777  004000  176252  1$:       BIT     #SW11,@SWR      ;DELETE ITERATION?  (QUICK PASS)
 1058  002730  001011                            BNE     2$              ;BR IF YES
 1059  002732  105737  001313                    TSTB    QV.FLG          ;HAVE PASSES BEECOMPLETED?
 1060  002736  001406                            BEQ     2$              ;BR IF QUICK PASS.
 1061  002740  005237  001224                    INC     LPCNT           ;UPDATE ITERATION COUNTER
 1062  002744  023737  001224  001222            CMP     LPCNT,ICOUNT    ;ARE ALL ITERATIONS DONE??
 1063  002752  001014                            BNE     3$              ;BR IF NOT YET
 1064  002754  105037  001311          2$:       CLRB    ERRFLG          ;PREPARE FOR NEW TEST
 1065  002760  005037  001224                    CLR     LPCNT           ;START ICOUNTER AT 0
 1066  002764  005037  001220                    CLR     LOCK
 1067  002770  012737  000020  001222            MOV     #20,ICOUNT      ;RESET ITERATIONS
 1068  002776  013737  001216  001214            MOV     NEXT,RETURN     ;GET NEXT TEST
 1069  003004  011600                  3$:       MOV     (SP),R0         ;POP R0 OFF OF THE STACK
 1070  003006  022626                            POP2SP                  ;FAKE AN "RTI"
 1071  003010  000177  176200                    JMP     @RETURN         ;GO DO THE TEST
 1072  003014  001407                  BRW:      1407
 1073  003016  000437                  BRX:      437
 1074
 1075                                            ;CHECK FOR FREEZE ON CURRENT DATA
 1076                                            ;--------------------------------
 1077
 1078  003020  032777  001000  176154  .SCOP1:   BIT     #SW09,@SWR      ;IS SW09=1(SET)?
 1079  003026  001405                            BEQ     1$              ;BR IF NOT SET.
 1080  003030  005737  001220                    TST     LOCK
 1081  003034  001402                            BEQ     1$
 1082  003036  013716  001220                    MOV     LOCK,(SP)       ;GOTO THE ADDRESS IN LOCK.
 1083  003042  000002                  1$:       RTI                     ;GO BACK.
 1084
 1085                                            ;TELETYPE OUTPUT ROUTINE
 1086                                            ;-----------------------
 1087
 1088  003044  010546                  .TYPE:    MOV     R5,-(SP)        ;SAVE R5 ON THE STACK.
 1089  003046  017605  000002                    MOV     @2(SP),R5       ;GET ADDRESS OF MESSAGE.
 1090  003052  062766  000002  000002            ADD     #2,2(SP)        ;POP OVER ADDRESS.
 1091  003060  032777  010000  176114  1$:       BIT     #SW12,@SWR      ;INHIBIT ALL PRINT OUT??
 1092  003066  001012                            BNE     3$              ;BR IF NO PRINT OUT WANTED (SW12=1)
 1093  003070  105715                            TSTB    (R5)            ;IS NUMBER MINUS? (MSB=1(BIT7))
 1094  003072  100002                            BPL     2$              ;BR IF NUMBER IS PLUS
 1095  003074  104402  005104                    TYPE    ,MCRLF          ;TYPE A CR/LF!
 1096  003100  105777  176104          2$:       TSTB    @TPCSR          ;TTY READY?
 1097  003104  100375                            BPL     2$              ;BR IF NO.
 1098  003106  112577  176100                    MOVB    (R5)+,@TPDBR    ;PRINT CURRENT CHAR.
 1099  003112  001362                            BNE     1$              ;IF NOT ZERO KEEP PRINTING!
 1100  003114  012605                  3$:       MOV     (SP)+,R5        ;END OF OUTPUT. RESTORE R5
```

```
1101  003116  000002                          RTI                     ;GO HOME
1102                                          ;-----------------------
1103
1104  003120  010346                  .INSTR: MOV     R3,-(SP)        ;SAVE R3 ON STACK
1105  003122  010446                          MOV     R4,-(SP)        ;SAVE R4 ON STACK
1106  003124  017637  000004  003142          MOV     @4(SP),.MSG
1107  003132  062766  000002  000004          ADD     #2,4(SP)
1108  003140  104402                  .INST1: TYPE
1109  003142  000000                  .MSG:   0
1110  003144  012704  005520                  MOV     #INBUF,R4
1111  003150  012703  000007                  MOV     #7,R3
1112  003154  105777  176024          1$:     TSTB    @TKCSR
1113  003160  100375                          BPL     1$
1114  003162  117714  176020                  MOVB    @TKDBR,(R4)
1115  003166  142714  000200                  BICB    #200,(R4)
1116  003172  122427  000015                  CMPB    (R4)+,#15
1117  003176  001417                          BEQ     INSTR2
1118  003200  105777  176004          2$:     TSTB    @TPCSR
1119  003204  100375                          BPL     2$
1120  003206  017777  175774  175776          MOV     @TKDBR,@TPDBR
1121  003214  005303                          DEC     R3
1122  003216  001356                          BNE     1$
1123  003220  012604                          MOV     (SP)+,R4
1124  003222  012603                          MOV     (SP)+,R3
1125  003224  104402  005100          .INSTE: TYPE    ,MQM
1126  003230  010346                          MOV     R3,-(SP)
1127  003232  010446                          MOV     R4,-(SP)
1128  003234  000741                          BR      .INST1
1129  003236  012604                  INSTR2: MOV     (SP)+,R4        ;RESTORE R4
1130  003240  012603                          MOV     (SP)+,R3        ;RESTORE R3
1131  003242  000002                          RTI
1132
1133                                          ;CONVERT ASCII STRING TO OCTAL
1134                                          ;----------------------------
1135
1136  003244  010546                  .PARAM: MOV     R5,-(SP)
1137  003246  010446                          MOV     R4,-(SP)
1138  003250  016605  000004                  MOV     4(SP),R5
1139  003254  012537  003434                  MOV     (R5)+,LOLIM
1140  003260  012537  003436                  MOV     (R5)+,HILIM
1141  003264  012537  003440                  MOV     (R5)+,DEVADR
1142  003270  112537  003442                  MOVB    (R5)+,LOBITS
1143  003274  112537  003443                  MOVB    (R5)+,ADRCNT
1144  003300  010566  000004                  MOV     R5,4(SP)
1145  003304  005005                  PARAM1: CLR     R5
1146  003306  012704  005520                  MOV     #INBUF,R4
1147  003312  122714  000015                  CMPB    #15,(R4)
1148  003316  001420                          BEQ     PARERR
1149  003320  121427  000060          1$:     CMPB    (R4),#60
1150  003324  002415                          BLT     PARERR
1151  003326  121427  000067                  CMPB    (R4),#67
1152  003332  003012                          BGT     PARERR
1153  003334  142714  000060                  BICB    #60,(R4)
1154  003340  152405                          BISB    (R4)+,R5
1155  003342  122714  000015                  CMPB    #15,(R4)
1156  003346  001406                          BEQ     LIMITS
```

```
1157  003350  006305                          ASL     R5
1158  003352  006305                          ASL     R5
1159  003354  006305                          ASL     R5
1160  003356  000760                          BR      1$
1161  003360  104404                PARERR:   INSTER
1162  003362  000750                          BR      PARAM1
1163
1164                                          ;TEST TO SEE IF NUMBER IS WITHIN LIMITS
1165                                          ;--------------------------------------
1166
1167  003364  020537  003436        LIMITS:   CMP     R5,HILIM
1168  003370  101373                          BHI     PARERR
1169  003372  020537  003434                  CMP     R5,LOLIM
1170  003376  103770                          BLO     PARERR
1171  003400  133705  003442                  BITB    LOBITS,R5
1172  003404  001365                          BNE     PARERR
1173
1174                                          ;STORE NUMBER AT SPECIFIED ADDRESS
1175
1176  003406  013704  003440                  MOV     DEVADR,R4
1177  003412  010524                1$:       MOV     R5,(R4)+
1178  003414  062705  000002                  ADD     #2,R5
1179  003420  105337  003443                  DECB    ADRCNT
1180  003424  001372                          BNE     1$
1181  003426  012604                          MOV     (SP)+,R4
1182  003430  012605                          MOV     (SP)+,R5
1183  003432  000002                          RTI
1184  003434  000000                LOLIM:    0
1185  003436  000000                HILIM:    0
1186  003440  000000                DEVADR:   0
1187  003442  000000                LOBITS:   0
1188          003443                ADRCNT=LOBITS+1
1189
1190                                          ;SAVE PC OF TEST THAT FAILED AND R0-R5
1191                                          ;-------------------------------------
1192
1193  003444  016637  000004  001276  .SAV05:  MOV     4(SP),SAVPC     ;SAVE R7 (PC)
1194
1195                                          ;SAVE R0-R5
1196
1197  003452  010537  001272        SV05:     MOV     R5,SAVR5        ;SAVE R5
1198  003456  010437  001270                  MOV     R4,SAVR4        ;SAVE R4
1199  003462  010337  001266                  MOV     R3,SAVR3        ;SAVE R3
1200  003466  010237  001264                  MOV     R2,SAVR2        ;SAVE R2
1201  003472  010137  001262                  MOV     R1,SAVR1        ;SAVE R1
1202  003476  010037  001260                  MOV     R0,SAVR0        ;SAVE R0
1203  003502  000002                          RTI                     ;LEAVE.
1204
1205                                          ;RESTORE R0-R5
1206
1207  003504  013700  001260        .RES05:   MOV     SAVR0,R0        ;RESTORE R0
1208  003510  013701  001262                  MOV     SAVR1,R1        ;RESTORE R1
1209  003514  013702  001264                  MOV     SAVR2,R2        ;RESTORE R2
1210  003520  013703  001266                  MOV     SAVR3,R3        ;RESTORE R3
1211  003524  013704  001270                  MOV     SAVR4,R4        ;RESTORE R4
1212  003530  013705  001272                  MOV     SAVR5,R5        ;RESTORE R5
```

```
1213  003534  000002                          RTI                     ;LEAVE
1214
1215                                          ;CONVERT OCTAL NUMBER TO ASCII AND OUTPUT TO TELEPRINTER
1216                                          ;-----------------------------------------------------
1217
1218  003536  104402  005104          .CONVR: TYPE    ,MCRLF
1219  003542  010046                  .CNVRT: MOV     R0,-(SP)
1220  003544  010146                          MOV     R1,-(SP)
1221  003546  010346                          MOV     R3,-(SP)
1222  003550  010446                          MOV     R4,-(SP)
1223  003552  010546                          MOV     R5,-(SP)
1224  003554  017601  000012                  MOV     @12(SP),R1
1225  003560  062766  000002  000012          ADD     #2,12(SP)
1226  003566  012137  003742                  MOV     (R1)+,WRDCNT
1227  003572  112137  003744          1$:     MOVB    (R1)+,CHRCNT
1228  003576  112137  003745                  MOVB    (R1)+,SPACNT
1229  003602  013137  003746                  MOV     @(R1)+,BINWRD
1230  003606  013704  003746          2$:     MOV     BINWRD,R4
1231  003612  113705  003744                  MOVB    CHRCNT,R5
1232  003616  012700  005562                  MOV     #TEMP,R0
1233  003622  010403                  3$:     MOV     R4,R3
1234  003624  042703  177770                  BIC     #177770,R3
1235  003630  062703  000060                  ADD     #060,R3
1236  003634  110320                          MOVB    R3,(R0)+
1237  003636  000241                          CLC
1238  003640  006004                          ROR     R4
1239  003642  000241                          CLC
1240  003644  006004                          ROR     R4
1241  003646  000241                          CLC
1242  003650  006004                          ROR     R4
1243  003652  005305                          DEC     R5
1244  003654  001362                          BNE     3$
1245  003656  012703  005624                  MOV     #MDATA,R3
1246  003662  114023                  4$:     MOVB    -(R0),(R3)+
1247  003664  105337  003744                  DECB    CHRCNT
1248  003670  001374                          BNE     4$
1249  003672  105737  003745                  TSTB    SPACNT
1250  003676  001405                          BEQ     6$
1251  003700  112723  000040          5$:     MOVB    #040,(R3)+
1252  003704  105337  003745                  DECB    SPACNT
1253  003710  001373                          BNE     5$
1254  003712  105013                  6$:     CLRB    (R3)
1255  003714  104402  005624                  TYPE    ,MDATA
1256  003720  005337  003742                  DEC     WRDCNT
1257  003724  001322                          BNE     1$
1258  003726  012605                          MOV     (SP)+,R5
1259  003730  012604                          MOV     (SP)+,R4
1260  003732  012603                          MOV     (SP)+,R3
1261  003734  012601                          MOV     (SP)+,R1
1262  003736  012600                          MOV     (SP)+,R0
1263  003740  000002                          RTI
1264  003742  000000          WRDCNT: 0
1255  003744  000000          CHRCNT: 0
1266          003745          SPACNT=CHRCNT+1
1267  003746  000000          BINWRD: 0
1268
```

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

003750  011646                .TRPSR: MOV     (SP),-(SP)          ;GET PC OF RETURN
003752  162716  000002                SUB     #2,(SP)             ;=PC OF TRAP
003756  017616  000000                MOV     @(SP),(SP)          ;GET TRP
003762  006316                TRPOK:  ASL     (SP)                ;MULTIPLY TRAP ARG BY 2
003764  042716  177001                BIC     #177001,(SP)        ;CLEAR UNWANTED BITS
003770  062716  001314                ADD     #.TRPTAB,(SP)       ;POINTER TO SUBROUTINE ADDRESS
003774  017616  000000                MOV     @(SP),(SP)          ;SUBROUTINE ADDRESS
004000  000136                        JMP     @(SP)+              ;GO TO SUBROUTINE

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

004002                          .HLT:
004002  022737  177570  001202        CMP     #177570,SWR         ;IS THERE A REAL SWR?
004010  001411                        BEQ     64$                 ;BR IF YES
004012  017746  175170                MOV     @TKDBR,-(SP)        ;SAVE KEYBOARD CHAR
004016  042716  000200                BIC     #BIT7,(SP)          ;CLEAR PARITY BIT
004022  122726  000007                CMPB    #7,(SP)+            ;WAS IT CNTRL 'G' ?
004026  001002                        BNE     .+6                 ;BR IF NO.
004030  004737  004640                JSR     PC,SERV.G           ;SERVICE "CNTRL 'G'".
004034  032777  010000  175140  64$:  BIT     #SW12,@SWR          ;BELL ON ERROR?
004042  001406                        BEQ     XBX                 ;BR IF NO BELL
004044  105777  175140                TSTB    @TPCSR              ;TTY READY.
004050  100003                        BPL     XBX                 ;DON'T WAIT IF TTY NOT READY.
004052  112777  000207  175132        MOVB    #207,@TPDBR         ;PUSH A BELL AT THE TTY.
004060  032777  020000  175114  XBX:  BIT     #SW13,@SWR          ;DELETE ERROR PRINT OUT?
004066  001105                        BNE     HALTS               ;BR IF NO PRINT OUT WANTED.
004070  021637  001234                CMP     (SP),LSTERR         ;WAS THIS ERROR FOUND LAST TIME?
004074  001404                        BEQ     1$                  ;BR IF YES
004076  011637  001234                MOV     (SP),LSTERR         ;RECORD BEING HERE
004102  105037  001311                CLRB    ERRFLG              ;PREPARE HEADER
004106  104406                  1$:   SAVDS                       ;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 HLT 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  023056                ADD     #.ERRTAB,R4         ;GET POINTER
004136  012437  004252                MOV     (R4)+,ERRMSG        ;GET ERROR MESSAGE
004142  012437  004264                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     LOC#
```

```
1325  004202  001402                        BEQ     1$
1326  004204  104402  005400                TYPE    .MASTEK
1327  004210  104402  005366        1$:     TYPE    .MTSTN
1328  004214  104411  004374                CNVRT   .XTSTN          ;SHOW IT
1329  004220  104402  005454                TYPE    .MERRPC         ;TYPE PC.
1330  004224  104411  004366                CNVRT   .ERTABO         ;SHOW IT
1331  004230  104402  005104                TYPE    .MCRLF          ;GIVE A CR/LF
1332  004234  112737  177777  001311        MOVB    #-1,ERRFLG      ;NO MORE HEADER UNLESS NO DATA TABLE.
1333  004242  005737  004252                TST     ERRMSG          ;IS THERE AN ERROR MESSAGE?
1334  004246  001402                        BEQ     WRK0.FM         ;BR IF NO.
1335  004250  104402                        TYPE                    ;TYPE
1336  004252  000000        ERRMSG: 0                               ;    ERROR MESSAGE
1337  004254                WRK0.FM:                                ;
1338  004254  005737  004264                TST     DATAHD          ;DATA HEADER?
1339  004260  001402                        BEQ     TYPDAT          ;BR IF NO
1340  004262  104402                        TYPE                    ;TYPE
1341  004264  000000        DATAHD: 0                               ;    DATA HEADER
1342  004266  005737  004276        TYPDAT: TST     DATABP          ;DATA TABLE?
1343  004272  001402                        BEQ     RESREG          ;BR IF NO.
1344  004274  104410                        CONVRT                  ;SHOW
1345  004276  000000        DATABP: 0                               ;    DATA TABLE
1346  004300  104407        RESREG: RESOS                           ;RESTORE PROC REGISTERS
1347  004302  005777  174674  HALTS:  TST     @SWR            ;HALT ON ERROR?
1348  004306  100005                        BPL     EXITER          ;BR IF NO HALT ON ERROR
1349  004310  010046                        PUSHRO                  ;SAVE R0
1350  004312  016600  000002                MOV     2(SP),R0        ;SHOW ERROR PC IN DATA LIGHTS
1351  004316  000000                        HALT                    ;HALT
1352  004320  012600                        POPRO                   ;GET R0
1353  004322  005237  001232        EXITER: INC     ERRCNT          ;UPDATE ERROR COUNT
1354  004326  032777  000400  174646        BIT     #SW08,@SWR      ;GOTO TOP OF TEST?
1355  004334  001007                        BNE     1$              ;BR IF YES
1356  004336  032777  002000  174636        BIT     #SW10,@SWR      ;GOTO NEXT TEST?
1357  004344  001407                        BEQ     2$              ;BR IF NO
1358  004346  013737  001216  001214        MOV     NEXT,RETURN     ;SET FOR NEXT TEST
1359  004354  012706  001200        1$:     MOV     #STACK,SP       ;RESET SP
1360  004360  000177  174630                JMP     @RETURN         ;GOTO SPECIFIED TEST
1361  004364  000002        2$:     RTI                             ;RETURN
1362  004366  000001        ERTABO: 1
1363  004370     006     002                .BYTE   6,2
1364  004372  001276                        SAVPC
1365  004374  000001        XTSTN:  1
1366  004376     003     002                .BYTE   3,2
1367  004400  001226                        TSTNO
1368                                        ;ENTER HERE ON POWER FAILURE
1369                                        ;--------------------------
1370
1371
1372  004402                .PFAIL:
1373  004402  012737  004414  000024        MOV     #RESTART,24             ;SET UP FOR POWER UP TRAP
1374  004410  000000                        HALT                            ;HALT ON POWER DOWN NORMAL
1375  004412  000777                        BR      .
1376
1377                                        ;PROCESSOR WILL TRAP HERE WHEN POWER IS RESTORED
1378
1379  004414                RESTAR:
1380  004414  012737  004402  000024        MOV     #.PFAIL,24      ;SET UP FOR POWER FAILURE
```

```
 1381  004422  012706  001200                 MOV     #STACK,SP       ;RESET THE STACK POINTER
 1382  004426  005037  005562                 CLR     TEMP            ;READY FOR TIMMER
 1383  004432  005237  005562                 INC     TEMP            ;PLUS ONE TO THE TIMER!
 1384  004436  001375                         BNE     .-4             ;BR IF MORE TO GO
 1385  004440  104402  005107                 TYPE    ,MPFAIL         ;TYPE THE MESSAGE
 1386  004444  104411  004470                 CNVRT   ,PFTAB          ;TELL WHAT TEST TO RETURN TO.
 1387  004450  105037  001311                 CLRB    ERRFLG          ;START CLEAN
 1388  004454  005037  001234                 CLR     LSTERR          ;
 1389  004460  104412                         MSTCLR                  ;START CLEAN UP OF DEVICE
 1390  004462  104413                         RAMCLR                  ;CLEAR IT ALL!
 1391  004464  000177  174524                 JMP     @RETURN         ;START DOING THAT TEST AGAIN.
 1392  004470  000001                 PFTAB:  1
 1393  004472     003     002          .BYTE   3,2
 1394  004474  001226                         TSTNO
 1395  004476  010046                 .DELAY: MOV     R0,-(SP)
 1396  004500  013700  004514                 MOV     1$,R0
 1397  004504  005300                         DEC     R0
 1398  004506  001376                         BNE     .-2
 1399  004510  012600                         MOV     (SP)+,R0
 1400  004512  000002                         RTI
 1401  004514  000036                 1$:     30.
 1402
 1403  004516                         .RAMCLR:
 1404  004516  012777  004000  174536         MOV     #MRESET,@DVSCR  ;ISSUE A MASTER CLEAR
 1405  004524  010146                         MOV     R1,-(SP)        ;SAVE R1 ON THE STACK
 1406  004526  010446                         MOV     R4,-(SP)        ;SAVE R4 ON THE STACK
 1407  004530  013701  001372                 MOV     DVSRS,R1        ;GET SECONDARY SEL. REG.
 1408  004534  013704  001376                 MOV     DVSRA,R4        ;GET SECONDARY REGISTER ACCESS REG.
 1409  004540  005014                 1$:     CLR     (R4)            ;ZERO THE SECONDARY REGISTER.
 1410  004542  062711  170361                 ADD     #^C<BIT11+BIT10+BIT9+BIT8+BIT3+BIT2+BIT1+BIT0>+BIT0,(R1
 1411  004546  001374                         BNE     1$
 1412  004550  012604                         MOV     (SP)+,R4        ;RESTORE R4
 1413  004552  012601                         MOV     (SP)+,R1        ;RESTORE R1
 1414  004554  000002                         RTI
 1415
 1416  004556                         .MSTCLR:
 1417  004556  012777  004000  174576         MOV     #MRESET,@DVSCR  ;ISSUE MASTER CLEAR.
 1418  004564  000002                         RTI
 1419
 1420  004566                         .ROMCLK:
 1421  004566  052777  000002  174566         BIS     #BIT1,@DVSCR
 1422  004574  000002                         RTI
 1423
 1424  004576                         .DATACLK:
 1425  004576  010046                         MOV     R0,-(SP)
 1426  004600  005000                         CLR     R0
 1427  004602  052777  000400  174560         BIS     #BIT8,@DVLCR
 1428  004610  017737  174554  004636 1$:     MOV     @DVLCR,3$
 1429  004616  106037  004637                 RORB    3$+1
 1430  004622  103003                         BCC     2$
 1431  004624  005200                         INC     R0
 1432  004626  001370                         BNE     1$
 1433  004630  104000                         HLT     0
 1434  004632  012600                 2$:     MOV     (SP)+,R0
 1435  004634  000002                         RTI
 1436  004636  000001                 3$:     .BLKW   1
```

```
 1437
 1438  004640  032777  004000  174336  SERV.G: BIT     #4000,@TKCSR    ;RX BUSY?
 1439  004646  001374                          BNE     SERV.G          ;BR IF YES
 1440  004650  017737  174326  005072          MOV     @SWR,90$        ;SAVE (SWR).
 1441  004656  013777  005052  174316  1$:     MOV     90$,@SWR        ;
 1442  004664  104402  005052                  TYPE    .89$            ;
 1443  004670  104411  005064                  CNVRT   .88$            ;
 1444  004674  104402  005074                  TYPE    .91$            ;
 1445  004700  105777  174300                  TSTB    @TKCSR          ;WAIT FOR DONE.
 1446  004704  100375                          BPL     .-4             ;
 1447  004706  017746  174274                  MOV     @TKDBR,-(SP)    ;
 1448  004712  042716  000200                  BIC     #BIT7,(SP)      ;
 1449  004716  122726  000015                  CMPB    #15.,(SP)+      ;
 1450  004722  001450                          BEQ     5$              ;
 1451  004724  005077  174252                  CLR     @SWR            ;
 1452  004730  105777  174254          2$:     TSTB    @TPCSR          ;
 1453  004734  100375                          BPL     .-4             ;
 1454  004736  016677  177776  174246          MOV     -2(SP),@TPDBR   ;
 1455  004744  000241                          CLC                     ;
 1456  004746  006177  174230                  ROL     @SWR            ;
 1457  004752  006177  174224                  ROL     @SWR            ;
 1458  004756  006177  174220                  ROL     @SWR            ;
 1459  004762  103735                          BCS     1$              ;ERROR
 1460  004764  026627  177776  000060          CMP     -2(SP),#60      ;
 1461  004772  002731                          BLT     1$              ;
 1462  004774  026627  177776  000067          CMP     -2(SP),#67      ;
 1463  005002  003325                          BGT     1$              ;
 1464  005004  042766  177770  177776          BIC     #10(7),-2(SP)   ;
 1465  005012  056677  177776  174162          BIS     -2(SP),@SWR     ;
 1466  005020  105777  174160                  TSTB    @TKCSR          ;
 1467  005024  100375                          BPL     .-4             ;
 1468  005026  017746  174154                  MOV     @TKDBR,-(SP)
 1469  005032  042716  000200                  BIC     #BIT7,(SP)
 1470  005036  122726  000015                  CMPB    #15.,(SP)+
 1471  005042  001332                          BNE     2$              ;
 1472  005044  104402  005104          5$:     TYPE    .MCRLF          ;
 1473  005050  000207                          RTS     PC
 1474
 1475  005052  020377  051450  051127  89$:     .ASCIZ  <377>? (SWR)=/?
 1476  005060  036451  000057
 1477                                  .EVEN
 1478  005064  000001                  88$:     !
 1479  005066     006     000                   .BYTE   6,0
 1480  005070  005072                          90$
 1481  005072  000000                  90$:     .WORD   0
 1482  005074  036457  000057          91$:     .ASCIZ  ?/=/?
 1483                                  .EVEN
 1484  005100  020040  000077          MQM:     .ASCIZ  /  ?/
  (2)  005104  005015     000          MCRLF:   .ASCIZ   <15><12>
  (2)  005107     377  053520  020122  MPFAIL:  .ASCIZ  <377>/PWR FAILED. RESTART AT TEST /
  (2)  005145     377  047105  020104  MEPASS:  .ASCIZ  <377>/END PASS DZDVB-B  /
  (2)  005171     377  000122          MR:      .ASCIZ  <377>/R/
  (2)  005174  050377  047522  051107  MERR2:   .ASCIZ  <377>/PROGRAM INDICATES NO DEVICES PRESENT./
  (2)  005243     377  047111  052523  MERR3:   .ASCIZ  <377>/INSUFFICIENT DATA!/
  (2)  005267     377  042524  052123  MTSTPC:  .ASCIZ  <377>/TEST PC-/
  (2)  005321     377  047514  045503  MLOCK:   .ASCIZ  <377>/LOCK ON SELECTED TEST
```

F03

```
DZDVS   MACY11 27(732)  17-SEP-76  11:14  PAGE 33
DZD\BB.P11      GENERAL UTILITIES (TYPE OUT,ERROR,SCOPE,ETC.)

    (2)  005330  051503  035122  000040  MCSRX:  .ASCIZ   /CSR: /
    (2)  005336  042526  035103  000040  MVECX:  .ASCIZ  /VEC: /
    (2)  005344  040520  051523  051505  MPASSX: .ASCIZ  /PASSES: /
    (3)  005355     105  051122  05111   MERRX:  .ASCIZ  /ERRORS: /
    (3)  005366  042524  053123  047040  MTSTN:  .ASCIZ  /TEST NO: /
    (2)  005400  000052                  MASTEK: .ASCIZ  /*/
    (2)  005402  051777  052105  051440  MNEW:   .ASCIZ  <377>/SET SWITCH REG TO DV11'S DESIRED ACTIVE./
    (3)  005454  041520  020072     000  MERRPC: .ASCIZ  /PC: /
    (3)  005461     377  040515  020120  XHEAD:  .ASCIZ  <377>/MAP OF DV11 STATUS/<377>
    (3)                                  .EVEN
    (2)  005506  000002                  XSTATQ: 2
 1485    005510     006     003                  .BYTE   6,3
 1486    005512  001246                          TEMP1
 1487    005514     006     002                  .BYTE   6,2
 1488    005516  001250                          TEMP2
 1489                                 .EVEN
 1490
 1491                                         ;BUFFERS FOR INPUT-OUTPUT
 1492
 1493    005520  000000                  INBUF:  0
 1494            005562                  .=.+40
 1495    005562  000000                  TEMP:   0
 1496            005624                  .=.+40
 1497    005624  000000                  MDATA:  0
 1498            005666                  .=.+40
```

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

```
1555  006166  013737  001376  001400          MOV     DVSRA,DVSFR     ;SPEC. FUN. REG.
1556  006174  060037  001400                  ADD     R0,DVSFR        ;
1557  006200  013737  001400  001402          MOV     DVSFR,DVNSR     ;NPR STAT. REG.
1558  006206  060037  001402                  ADD     R0,DVNSR        ;
1559  006212  013737  001402  001404          MOV     DVNSR,RESV16    ;RESERVED REG
1560  006220  060037  001404                  ADD     R0,RESV16       ;
1561
1562  006224  013737  001352  001354          MOV     DVRVEC,DVRLVL   ;PTY LVL
1563  006232  060037  001354                  ADD     R0,DVRLVL       ;
1564  006236  013737  001354  001356          MOV     DVRLVL,DVTVEC   ;TX VEC
1565  006244  060037  001356                  ADD     R0,DVTVEC       ;
1566  006250  013737  001356  001360          MOV     DVTVEC,DVTLVL   ;TX LVL
1567  006256  060037  001360                  ADD     R0,DVTLVL
1568
1569  006262  012700  001416                  MOV     #L00.03,R0      ;LOAD STAUS 00-03
1570  006266  012701  001406                  MOV     #MASK.A,R1      ;PREPARE MASK.
1571  006272  012702  001412                  MOV     #CLK.A,R2       ;PREPARE CLOCKS
1572  006276  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGLRATION.
1573
1574  006302  012700  001420                  MOV     #L04.07,R0      ;LOAD STAUS 00-03
1575  006306  012701  001407                  MOV     #MASK.B,R1      ;PREPARE MASK.
1576  006312  012702  001413                  MOV     #CLK.B,R2       ;PREPARE CLOCKS
1577  006316  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1578
1579  006322  012700  001422                  MOV     #L08.11,R0      ;LOAD STAUS 00-03
1580  006326  012701  001410                  MOV     #MASK.C,R1      ;PREPARE MASK.
1581  006332  012702  001414                  MOV     #CLK.C,R2       ;PREPARE CLOCKS
1582  006336  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1583
1584  006342  012700  001424                  MOV     #L12.15,R0      ;LOAD STAUS 00-03
1585  006346  012701  001411                  MOV     #MASK.D,R1      ;PREPARE MASK.
1586  006352  012702  001415                  MOV     #CLK.D,R2       ;PREPARE CLOCKS
1587  006356  004737  006516                  JSR     PC,FIX.00       ;GO AND CALCULATE CONFIGURATION.
1588  006362  032777  000002  172612          BIT     #SW01,@SWR
1589  006370  001445                          BEQ     7$
1590  006372                          4$:
1591  006372  005737  000042                  TST     @#42
1592  006376  001042                          BNE     7$
1593  006400  104402  005104                  TYPE    .MCRLF
1594  006404  104403                          INSTR
1595  006406  005366                          MTSTN
1596  006410  104405                          PARAM
1597  006412  000001                          1
1598  006414  001000                          1000
1599  006416  001226                          TSTNO
1600  006420     000                  .BYTE   0
1601  006421     001                  .BYTE   1
1602  006422  012700  007256                  MOV     #TST1,R0
1603  006426  022710                  5$:     CMP     (PC)+,(R0)
1604  006430  012737                          MOV     (PC)+,@(PC)+
1605  006432  001015                          BNE     6$
1606  006434  023760  001226  000002          CMP     TSTNO,2(R0)
1607  006442  001011                          BNE     6$
1608  006444  022760  001226  000004          CMP     #TSTNO,4(R0)
1609  006452  001005                          BNE     6$
1610  006454  010037  001214                  MOV     R0,RETURN
```

```
 1611  006460  104402  005104                    TYPE    ,MCRLF
 1612  006464  000412                            BR      9$
 1613  006466  005720                    6$:     TST     (R0)+
 1614  006470  020027  021754                    CMP     R0,#TLAST+10
 1615  006474  001354                            BNE     5$
 1616  006476  104402  005100                    TYPE    ,MQM
 1617  006502  000733                            BR      4$
 1618  006504  012737  007256  001214    7$:     MOV     #TST1,RETURN    ;PREPARE RETURN ADDRESS
 1619  006512  000177  172476            9$:     JMP     @RETURN         ;GO START TESTING.
 1620
 1621  006516  011003                    FIX.00: MOV     (R0),R3         ;GET PARAMETERS.
 1622  006520  042703  176377                    BIC     #1400,R3        ;CLEAR JUNK.
 1623  006524  005703                            TST     R3              ;TEST FOR EIGHT BITS.
 1624  006526  001004                            BNE     1$              ;BR IF NOT 8 BITS.
 1625  006530  105011                            CLRB    (R1)            ;SET
 1626  006532  112712  000010                    MOVB    #8.,(R2)        ;
 1627  006536  000424                            BR      4$
 1628  006540  022703  000400            1$:     CMP     #400,R3         ;CHECK FOR SEVEN BITS.
 1629  006544  001005                            BNE     2$              ;BR IF NOT 7 BITS.
 1630  006546  112711  000200                    MOVB    #200,(R1)       ;
 1631  006552  112712  000007                    MOVB    #7,(R2)         ;
 1632  006556  000414                            BR      4$
 1633  006560  022703  001000            2$:     CMP     #1000,R3        ;CHECK FOR SIX BITS.
 1634  006564  001005                            BNE     3$              ;BR IF NOT SIX BITS.
 1635  006566  112711  000300                    MOVB    #300,(R1)       ;
 1636  006572  112712  000006                    MOVB    #6,(R2)         ;
 1637  006576  000404                            BR      4$
 1638  006600  112711  000340            3$:     MOVB    #340,(R1)       ;IF NONE OF THE ABOVE; MUST BE 5 BITS.
 1639  006604  112712  000005                    MOVB    #5,(R2)         ;
 1640  006610  032710  040000            4$:     BIT     #PARBIT,(R0)    ;PARITY ENABLED?
 1641  006614  001401                            BEQ     5$              ;IF =0; THEN NO PARITY.
 1642  006616  105212                            INCB    (R2)            ;PLUS ONE TO THE CLOCK!
 1643  006620  000207                    5$:     RTS     PC              ;
 1644
 1645                                            ;*ROUTINE USED TO "AUTO SIZE" THE DV11
 1646                                            ;*CSR AND VECTOR.
 1647                                            ;*NOTE: THE CSR MAY BE ANY WHERE IN THE FLOATING
 1648                                            ;*      ADDRESS RANGE (175000:175400)
 1649                                            ;*      AND THE VECTOR MAY BE ANY WHERE IN THE
 1650                                            ;*      FLOATING VECTOR RANGE (300:770)
 1651                                            ;*
 1652
 1653  006622                            AUTO.SIZE:
 1654  006622  000005                            RESET                   ;INSURE A BUS INIT.
 1655  006624  012702  001500            CSRMAP: MOV     #DV.MAP,R2      ;LOAD MAP POINTER.
 1656  006630  005022                    1$:     CLR     (R2)+           ;ZERO ENTIRE MAP
 1657  006632  022702  001740                    CMP     #DV.END,R2      ;ALL DONE?
 1658  006636  001374                            BNE     1$              ;BR IF NO
 1659  006640  105037  001301                    CLRB    DVNUM           ;SET OCTAL NUMBER OF DV11'S TO 0
 1660  006644  012702  001500                    MOV     #DV.MAP,R2
 1661  006650  012701  175000                    MOV     #175000,R1      ;SET FOR FIRST ADDRESS TO BE TESTED
 1662  006654  012737  007074  000004          MOV     #6$,@#4         ;SET FOR NON-EXISTANT DEVICE TIME OUT
 1663  006662  005711                    2$:     TST     (R1)            ;IF DV11 DVSCR S/B 0
 1664  006664  001037                            BNE     3$              ;IF NO DEV ; TRAP TO 4. IF NO BIT 8 THEN NO DV11
 1665  006666  022761  177777  000012          CMP     #177777,12 R1)  ;IF DV11 THEN DVSFR S B ALL 1'S ON INIT!
 1666  006674  001033                            BNE     3$              ;BR IF NOT DV11
```

```
1667  006676  005761  000016                  TST     16(R1)          ;IF DV11 THEN RESV16 S/B ALL 0'S
1668  006702  001030                          BNE     3$              ;BR IF NOT DV11
1669                                  ;AT THIS POINT IT IS ASSUMED THAT R1 HOLDS A DV11 CSR ADDRESS.
1670  006704  010122                          MOV     R1,(R2)+        ;STORE CSR IN CORE TABLE.
1671  006706  005722                          TST     (R2)+           ;POP OVER VECTOR STORE AREA
1672  006710  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 1 STAT AND SYNC
1673  006714  052722  000062                  BIS     #62,(R2)+       ;
1674  006720  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 2 STAT AND SYNC
1675  006724  052722  000062                  BIS     #62,(R2)+       ;
1676  006730  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 3 STAT AND SYNC
1677  006734  052722  000062                  BIS     #62,(R2)+       ;
1678  006740  052722  000226                  BIS     #226,(R2)+      ;SET LINE CARD 4 STAT AND SYNC
1679  006744  052722  000062                  BIS     #62,(R2)+       ;
1680  006750  105237  001301                  INCB    DVNUM           ;UPDATE DEVICE COUNTER
1681  006754  122737  000010  001301          CMPB    #10,DVNUM       ;ARE MAX. NO. OF DEV FOUND?
1682  006762  001405                          BEQ     100$            ;YES DON'T LOOK FOR ANY MORE.
1683  006764  062701  000010          3$:     ADD     #10,R1          ;UPDATE CSR POINTER ADDRESS
1684  006770  022701  175400                  CMP     #175400,R1
1685  006774  001332                          BNE     2$              ;BR IF MORE ADDRESS TO CHECK.
1686  006776  012722  177777          100$:   MOV     #177777,(R2)+   ;TERMINATER.
1687  007002  105037  001300                  CLRB    DVACTV
1688  007006  105737  001301                  TSTB    DVNUM           ;WERE ANY DV11'S FOUND AT ALL?
1689  007012  001423                          BEQ     5$              ;ERROR AUTO SIZER FOUND NO DV11'S IN THIS SYS.
1690  007014  113701  001301                  MOVB    DVNUM,R1
1691  007020  110137  001303                  MOVB    R1,SAVNUM       ;SAVE NUMBER OF DEVICES
1692  007024  000241          4$:     CLC
1693  007026  106137  001300                  ROLB    DVACTV          ;GENERATE ACTIVE REGISTER OF DEVICES.
1694  007032  105237  001300                  INCB    DVACTV          ;SET THE BIT
1695  007036  005301                          DEC     R1
1696  007040  001371                          BNE     4$              ;BR IF MORE TO GENERATE
1697  007042  012737  000006  000004          MOV     #6,@#4          ;RESTORE TRAP VECTOR
1698  007050  113737  001300  001302          MOVB    DVACTV,SAVACT   ;SAVE ACTIVE REGISTER
1699  007056  000137  007102                  JMP     VECMAP          ;GO FIND THE VECTOR NOW.
1700  007062  104402  005174          5$:     TYPE    ,MERR2          ;NOTIFY OPR THAT NO DV11'S FOUND.
1701  007066  005000                          CLR     R0              ;MAKE DATA LIGHTS ZERO
1702  007070  000000                          HALT                    ;STOP THE SHOW
1703  007072  000776                          BR      .-2             ;DISABLE CONT. SW.
1704  007074  012716  006764          6$:     MOV     #3$,(SP)        ;ENTERED BY NON-EXISTANT TIME-OUT.
1705  007100  000002                          RTI                     ;RETURN TO MAINSTREAM
1706
1707  007102  012737  000340  000022  VECMAP: MOV     #340,@#22       ;SET IOT TRAP PRIO TO 7
1708  007110  012737  007232  000020          MOV     #4$,@#20        ;SET IOT TRAP VECTOR
1709  007116  012702  001500                  MOV     #DV.MAP,R2      ;SET SOFTWARE POINTER
1710  007122  012700  000300                  MOV     #300,R0         ;FLOATING VECTORS START HERE.
1711  007126  012701  000302                  MOV     #302,R1         ;PC OF IOT INSTR.
1712  007132  010120          1$:     MOV     R1,(R0)+        ;START FILLING VECTOR AREA
1713  007134  012721  000004                  MOV     #4,(R1)+        ;WITH .+2; IOT
1714  007140  022021                          CMP     (R0)+,(R1)+     ;ADD 2 TO R0 +R1
1715  007142  020127  001000                  CMP     R1,#1000
1716  007146  101771                          BLOS    1$              ;BR IF MORE TO FILL
1717  007150  113737  001300  001246          MOVB    DVACTV,TEMP1    ;STORE TEMPORALLY
1718  007156  006037  001246          2$:     ROR     TEMP1           ;BRING OUT A BIT
1719  007162  103034                          BCC     5$              ;BR IF ALL DONE
1720  007164  005037  177776                  CLR     PS              ;ZERO CPU PRIO
1721  007170  012772  001300  000000          MOV     #BIT9+BIT7+BIT6,@(R2)
1722  007176  005000                          CLR     R0              ;ATTEMPT TO FORCE AN INTERUPT
```

```
1723  007200  005200                          INC     R0              ;STALL
1724  007202  001376                          BNE     .-2             ;       FOR TIME TO INTERUPT
1725  007204  052762  000300  000002          BIS     #300,2(R2)      ;NO INTERUPT ASSUME 300 AND FIX DV11 LATER
1726  007212  042772  176777  000000  3$:     BIC     #^C<BIT9>,@(R2)
1727  007220  005072  000000                  CLR     @(R2)
1728  007224  062702  000024                  ADD     #24,R2          ;POP SOFTWARE POINTER
1729  007230  000752                          BR      2$              ;KEEP GOING
1730  007232  051662  000002          4$:     BIS     (SP),2(R2)      ;GET VECTOR ADDRESS
1731  007236  042762  000007  000002          BIC     #7,2(R2)        ;CLEAR JUNK
1732  007244  022626                          CMP     (SP)+,(SP)+     ;POP IOT JUNK OFF STACK
1733  007246  012716  007212                  MOV     #3$,(SP)        ;SET FOR RETURN
1734  007252  000002                          RTI
1735  007254  000207                  5$:     RTS     PC              ;ALL DONE WITH "AUTO SIZING"
1736
```

```
                                        ;********************** TEST 1 ******************************
                                        ;*TEST THAT "TRANSMITTER FLAG WAITING"
                                        ;*IS TRUE AND THAT "RECV FLAG WAITING" IS
                                        ;*FALSE  AFTER AN INIT.
                                        ;*THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
                                        ;:************************************************************

                                ;  TEST 1
                                ;--------------
007256  012737  000001  001226  TST1:   MOV     #1,TSTNO
007264  012737  007540  001216          MOV     #TST2,NEXT
007272  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
007276  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
007304  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
007306  004737  007374                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
007312  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
007316  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
007324  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
007326  004737  007374                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
007332  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
007336  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
007344  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
007346  004737  007374                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
007352  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
007356  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
007364  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
007366  004737  007374                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
007372  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
007374                          105$:                           ;TEST ENTRANCE.
007374  104413                          RAMCLR                  ;CLEAR ALL DV11 SEC. REGS.
007376  010037  007410                  MOV     R0,65$          ;STORE LINE NO. POINTER.
007402  005001                          CLR     R1              ;ZERO MSCANNER POINTER
007404  004537  022470          1$:     PERFORM ,SETSCAN        ;POSITION SCANNER TO LINE NUMBER.
007410  000001                  55$:    .BLKW 1                 ;INITAL LINE NUMBER HERE.
007412  012703  000004          2$:     MOV     #4,R3           ;SET TO DO 4 LINES AT A TIME
007416  012705  000003          3$:     MOV     #BIT1+BIT0,R5   ;SET EXPECTED RESULTS IN R5
007422  012702  002000                  MOV     #BIT10,R2       ;BR-A "RX FLAG WAITING"?
007426  010277  171746                  MOV     R2,@DVSFR       ;LOAD DV11 INSTRUCTION
007432  017704  171732                  MOV     @DVLCR,R4       ;READ BR TEST POINTS
007436  020504                          CMP     R5,R4           ;TEST POINTS OK?
007440  001401                          BEQ     4$              ;BR IF YES
007442  104001                          HLT     1               ;EXPECT DVLCR BIT1+BIT0=1
007444  012777  050102  171726  4$:     MOV     #S.C+BIT6+BIT1,@DVSFR
007452  104415                          ROMCLK                  ;S/C "ADVANCE SCANNER"
007454  005201                          INC     R1              ;UPDATE MSCAN POINTER
007456  010100                          MOV     R1,R0           ;PREPARE TO SET LINE POINTER
007460  000241                          CLC                     ;TO CORRECT POSITION
007462  006000                          ROR     R0              ;
007464  012702  001000                  MOV     #BIT9,R2        ;BR-A "TX FLAG WAITING"?
007470  010277  171704                  MOV     R2,@DVSFR       ;LOAD DV11 INSTRUCTION
007474  017704  171670                  MOV     @DVLCR,R4       ;READ BR TEST POINT
007500  012705  000002                  MOV     #BIT1,R5        ;SET EXPECTED RESULTS
007504  020504                          CMP     R5,R4           ;TX FLAG WAITING TRUE?
007506  001401                          BEQ     5$              ;BR IF LCR BIT1=1 AND BIT0=0
007510  104001                          HLT     1               ;ERROR.
```

```
 1793  007512  012777  050102  171660  5$:     MOV     #S.C+BIT6+BIT1,@DVSFR
 1794  007520  104415                          ROMCLK                  ;S/C "ADVANCE SCANNER"
 1795  007522  005201                          INC     R1              ;UPDATE MSCAN POINTER
 1796  007524  010100                          MOV     R1,R0           ;UPDATE LINE POINTER
 1797  007526  000241                          CLC                     ;
 1798  007530  006000                          ROR     R0              ;
 1799  007532  005303                          DEC     R3              ;ARE ALL 4 LINES TESTED?
 1800  007534  001330                          BNE     3$              ;BR IF NO!
 1801  007536  000207                          RTS     PC              ;CHECK NEXT SET OF LINES.
 1802
 1803
 1804                                          ;************************ TEST 2 *******************************
 1805                                          ;*TEST THAT "MATCH DETECT" IS
 1806                                          ;*FALSE AFTER AN INIT.
 1807                                          ;*THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
 1808                                          ;;***********************************************************
 1809
 1810                                  ;  TEST 2
 1811                                  ;--------------
 1812  007540  012737  000002  001226  TST2:   MOV     #2,TSTNO
 1813  007546  012737  007742  001216          MOV     #TST3,NEXT
 1814  007554  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 1815  007560  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 1816  007566  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 1817  007570  004737  007656                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 1818  007574  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 1819  007600  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 1820  007606  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 1821  007610  004737  007656                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 1822  007614  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 1823  007620  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 1824  007626  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 1825  007630  004737  007656                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 1826  007634  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 1827  007640  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 1828  007646  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 1829  007650  004737  007656                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 1830  007654  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
 1831  007656                          105$:                           ;TEST ENTRANCE.
 1832  007656  010037  007672                  MOV     R0,65$          ;SET LINE POINTER
 1833  007662  104412                          MSTCLR                  ;RESET THE DV11
 1834  007664  005001                          CLR     R1              ;ZERO MSCANNER POINTER
 1835  007666  004537  022470          1$:     PERFORM ,SETSCAN        ;SET MSCAN TO CORRECT LINE
 1836  007672  000001                  65$:    .BLKW 1                 ;INITAL LINE POINTER PLACED HERE.
 1837  007674  012703  000004          2$:     MOV     #4,R3           ;SET FOR A FOUR LINE GROUP.
 1838  007700  012705  000003          3$:     MOV     #BIT1+BIT0,R5   ;SET EXPECTED RESULTS.
 1839  007704  012702  076400          4$:     MOV     #BRB+BIT11+BIT10+BIT8,R2
 1840  007710  010277  171464                  MOV     R2,@DVSFR       ;BR-B "MATCH DET"?
 1841  007714  017704  171450                  MOV     @DVLCR,R4       ;READ DVLCR INTO R4
 1842  007720  020504                          CMP     R5,R4           ;MATCH DET FALSE??
 1843  007722  001401                          BEQ     5$              ;BR IF YES
 1844  007724  104001                          HLT     1               ;LCR BIT1=1+BIT0=1 EXPECTED.
 1845  007726  004537  022470          5$:     PERFORM ,SETSCAN        ;UPDATE MSCAN POINTER TO NEXT LINE.
 1846  007732  000001                          1                       ;1 LINE
 1847  007734  005303                          DEC     R3              ;ALL FOUR LINES DONE YET?
 1848  007736  001362                          BNE     4$              ;BR IF NO
```

```
1849  007740  000207                          RTS     PC              ;CHECK NEXT SET OF LINES
1850
1851
1852                                          ;************************ TEST 3 ******************************
1853                                          ;*TEST THAT MAINT BIT WINDOW IS CLEARED
1854                                          ;* AFTER AN INIT.
1855                                          ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
1856                                          ;:**************************************************************
1857
1858
1859                                  ;  TEST 3
1860                                  ;--------------
1861  007742  012737  000003  001226  TST3:   MOV     #3,TSTNO
1862  007750  012737  010150  001216          MOV     #TST4,NEXT
1863  007756  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
1864  007762  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
1865  007770  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
1866  007772  004737  010060                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
1867  007776  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
1868  010002  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
1869  010010  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
1870  010012  004737  010060                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
1871  010016  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
1872  010022  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
1873  010030  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
1874  010032  004737  010060                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
1875  010036  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
1876  010042  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
1877  010050  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
1878  010052  004737  010060                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
1879  010056  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
1880  010060                          105$:                           ;TEST ENTRANCE.
1881  010060  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
1882  010066  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
1883  010070  000207                          RTS     PC              ;EXIT TEST
1884  010072  104412                          MSTCLR                  ;RESET DV11
1885  010074  005002                          CLR     R2              ;ZERO SFR IMAGE
1886  010076  017705  171266                  MOV     @DVLCR,R5       ;READ THE DVLCR INTO R5
1887  010102  042705  000200                  BIC     #BIT7,R5        ;CLEAR MAINT BIT WINDOW EXPECTED
1888  010106  012703  000004                  MOV     #4,R3           ;SET TO DO 4 LINES.
1889  010112  010077  171254          1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
1890  010116  017704  171246                  MOV     @DVLCR,R4       ;READ DVLCR RESULTS INTO R4
1891  010122  042705  000060                  BIC     #BIT5+BIT4,R5   ;CLEAR EXTENDED ADDRESS BITS
1892  010126  042704  000060                  BIC     #BIT5+BIT4,R4   ;""
1893  010132  020504                          CMP     R5,R4           ;OK?
1894  010.34  001401                          BEQ     2$              ;
1895  010136  104001                          HLT     1               ;BIT7 INCORRECT
1896  010140  005200                  2$:     INC     R0              ;UPDATE LINE POINTER
1897  010142  005303                          DEC     R3              ;ALL LINES DONE?
1898  010144  001362                          BNE     1$              ;BR IF NO
1899  010146  000207                          RTS     PC              ;RETURN FOR NEXT SET OF LINES.
1900
1901
1902                                          ;************************ TEST 4 ******************************
1903                                          ;*TEST THAT THE BIT WINDOW WILL
1904                                          ;*STAY CLEARED WHEN MAINT INTERNAL
```

```
                                                ;*MODE IS SELECTED BUT COND. STROBE IS
                                                ;*NOT ASSERETED.
                                                ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                                                ;;**********************************************************************


                                        ;  TEST 4
                                        ;--------------
010150  012737  000004  001226  TST4:   MOV     #4,TSTNO
010156  012737  010364  001216          MOV     #TST5,NEXT
010164  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
010170  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
010176  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
010200  004737  010266                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
010204  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
010210  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
010216  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
010220  004737  010266                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
010224  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
010230  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
010236  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
010240  004737  010266                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
010244  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
010250  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
010256  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
010260  004737  010266                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
010264  104400          103$:   SCOPE                           ;SCOPE THIS TEST.
010266          105$:                   ;TEST ENTRANCE.
010266  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
010274  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
010276  000207                          RTS     PC              ;EXIT TEST
010300  104412                          MSTCLR                  ;RESET DV11
010302  005002                          CLR     R2              ;ZERO SFR IMAGE
010304  012777  004000  171056          MOV     #BIT11,@DVLCR   ;SET INTERNAL MAINT MODE
010312  017705  171052                  MOV     @DVLCR,R5       ;READ THE DVLCR INTO R5
010316  042705  000200                  BIC     #BIT7,R5        ;CLEAR MAINT BIT WINDOW EXPECTED
010322  012703  000004                  MOV     #4,R3           ;SET TO DO 4 LINES.
010326  010077  171040          1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
010332  017704  171032                  MOV     @DVLCR,R4       ;READ DVLCR RESULTS INTO R4
010336  042705  000060                  BIC     #BIT5+BIT4,R5   ;CLEAR EXTENDED ADDRESS BITS
010342  042704  000060                  BIC     #BIT5+BIT4,R4   ;""
010346  020504                          CMP     R5,R4           ;OK?
010350  001401                          BEQ     2$              ;
010352  104001                          HLT     1               ;BIT INCORRECT
010354  005200          2$:     INC     R0              ;UPDATE LINE POINTER
010356  005303                  DEC     R3              ;ALL LINES DONE?
010360  001362                  BNE     1$              ;BR IF NO
010362  000207                  RTS     PC              ;RETURN FOR NEXT SET OF LINES.


                                                ;************************ TEST 5 *******************************
                                                ;*TEST THAT THE BIT WINDOW WILL
                                                ;*SET WHEN MAINT INTERNAL MODE IS SELECTED
                                                ;*AND COND. STOBE IS ASSERTED.
                                                ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                                                ;;**********************************************************************
```

```
                                                ;  TEST 5
                                                ;--------------
010264  012737  000005  001226  TST5:   MOV     #5,TSTNO
010272  012737  010612  001216          MOV     #TST6,NEXT
010300  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
010404  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
010412  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
010414  004737  010502                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
010420  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
010424  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
010432  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
010434  004737  010502                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
010440  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
010444  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
010452  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
010454  004737  010502                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
010460  012700  000014          103$:   MOV     #12.,R0         ;LOAD LINE NO.
010464  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
010472  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
010474  004737  010502                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
010500  104400          103$:   SCOPE                           ;SCOPE THIS TEST.
010502          105$:                                           ;TEST ENTRANCE.
010502  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
010510  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
010512  000207                          RTS     PC              ;EXIT TEST
010514  104412                          MSTCLR                  ;RESET DV11
010516  005002                          CLR     R2              ;ZERO SFR IMAGE
010520  012777  004000  170642          MOV     #BIT11,@DVLCR   ;SET INTERNAL MAINT MODE
010526  017705  170636                  MOV     @DVLCR,R5       ;READ THE DVLCR INTO R5
010532  052705  000200                  BIS     #BIT7,R5        ;SET MAINT BIT WINDOW EXP RESULTS
010536  012703  000004                  MOV     #4,R3           ;SET TO DO 4 LINES.
010542  010077  170624          1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
010546  052777  100000  170614          BIS     #BIT15,@DVLCR   ;SET STROBE
010554  004737  022406                  JSR     PC,CKBIT15      ;GO WAIT FOR BIT15 TO =0
010560  017704  170604                  MOV     @DVLCR,R4       ;READ DVLCR RESULTS INTO R4
010564  042705  000060                  BIC     #BIT5+BIT4,R5   ;CLEAR EXTENDED ADDRESS BITS
010570  042704  000060                  BIC     #BIT5+BIT4,R4   ;""
010574  020504                          CMP     R5,R4           ;OK?
010576  001401                          BEQ     2$              ;
010600  104001                          HLT     1               ;BIT7 INCORRECT
010602  005200          2$:     INC     R0                      ;UPDATE LINE POINTER
010604  005303                  DEC     R3                      ;ALL LINES DONE?
010606  001355                  BNE     1$                      ;BR IF NO
010610  000207                  RTS     PC                      ;RETURN FOR NEXT SET OF LINES.


                        ;********************** TEST 6 ******************************
                        ;*TEST THAT THE BIT WINDOW WILL BE CLEARED
                        ;*WHEN MAINT INTERNAL MODE IS SELECTED AND TX DSABLE
                        ;*IS ASSERTED.
                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                        ;;************************************************************
```

```
2017                                        ;  TEST 6
2018                                        ;--------------
2019  010612  012737  000006  001226  TST6:   MOV     #6,TSTNO
2020  010620  012737  011040  001216          MOV     #TST7,NEXT
2021  010626  012700  000000                  MOV     #0,,R0          ;PLACE LINE NUMBER INTO R0
2022  010632  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2023  010640  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2024  010642  004737  010730                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2025  010646  012700  000004          100$:   MOV     #4,,R0          ;PLACE LINE NUMBER INTO R0
2026  010652  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2027  010660  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2028  010662  004737  010730                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2029  010666  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2030  010672  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2031  010700  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2032  010702  004737  010730                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2033  010706  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
2034  010712  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2035  010720  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2036  010722  004737  010730                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2037  010726  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
2038  010730                          105$:                           ;TEST ENTRANCE.
2039  010730  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2040  010736  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
2041  010740  000207                          RTS     PC              ;EXIT TEST
2042  010742  104412                          MSTCLR                  ;RESET DV11
2043  010744  005002                          CLR     R2              ;ZERO SFR IMAGE
2044  010746  012777  005000  170414          MOV     #BIT11+BIT9,@DVLCR      ;SET INTER MAINT MODE FOR SYSTEM TESTING
2045  010754  017705  170410                  MOV     @DVLCR,R5       ;READ THE DVLCR INTO R5
2046  010760  042705  000200                  BIC     #BIT7,R5        ;CLEAR MAINT BIT WINDOW EXPECTED
2047  010764  012703  000004                  MOV     #4,R3           ;SET TO DO 4 LINES.
2048  010770  010077  170376          1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2049  010774  052777  100000  170366          BIS     #BIT15,@DVLCR   ;SET STROBE
2050  011002  004737  022406                  JSR     PC,CKBIT15      ;GO WAIT FOR BIT15 TO =0
2051  011006  017704  170356                  MOV     @DVLCR,R4       ;READ DVLCR RESULTS INTO R4
2052  011012  042705  000060                  BIC     #BIT5+BIT4,R5   ;CLEAR EXTENDED ADDRESS BITS
2053  011016  042704  000060                  BIC     #BIT5+BIT4,R4   ;""
2054  011022  020504                          CMP     R5,R4           ;OK?
2055  011024  001401                          BEQ     2$              ;
2056  011026  104001                          HLT     1               ;BIT7 INCORRECT
2057  011030  005200                  2$:     INC     R0              ;UPDATE LINE POINTER
2058  011032  005303                          DEC     R3              ;ALL LINES DONE?
2059  011034  001355                          BNE     1$              ;BR IF NO
2060  011036  000207                          RTS     PC              ;RETURN FOR NEXT SET OF LINES.
2061
2062
2063                                         ;********************** TEST 7 ******************************
2064                                         ;*TEST THAT "MAINT DATA" WILL SHOW
2065                                         ;*UP IN "MAINT BIT WINDOW".
2066                                         ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2067                                         ;;***************************************************************
2068
2069                                         ;  TEST 7
2070                                         ;--------------
2071  011040  012737  000007  001226  TST7:   MOV     #7,TSTNO
2072  011046  012737  011344  001216          MOV     #TST10,NEXT
```

E04


```
2073  011054  012700  000000            MOV     #0,,R0          ;PLACE LINE NUMBER INTO R0
2074  011060  013737  001416  001236    MOV     L00.03.STAT     ;LOAD LINE CARD STATUS INTO STAT
2075  011066  100402                    BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2076  011070  004737  011156            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2077  011074  012700  000004    100$:   MOV     #4,,R0          ;PLACE LINE NUMBER INTO R0
2078  011100  013737  001420  001236    MOV     L04.07.STAT     ;LOAD LINE CARD STATUS INTO STAT
2079  011106  100402                    BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2080  011110  004737  011156            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2081  011114  012700  000010    101$:   MOV     #8,,R0          ;LOAD LINE NUMBER
2082  011120  013737  001422  001236    MOV     L08.11.STAT     ;LOAD LINE CARD STATUS INTO STAT
2083  011126  100402                    BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2084  011130  004737  011156            JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2085  011134  012700  000014    102$:   MOV     #12.,R0         ;LOAD LINE NO.
2086  011140  013737  001424  001236    MOV     L12.15.STAT     ;LOAD LINE CARD STATUS
2087  011146  100402                    BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2088  011150  004737  011156            JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2089  011154  104400            103$:   SCOPE                   ;SCOPE THIS TEST.
2090  011156                    105$:   ;TEST ENTRANCE.
2091  011156  032737  004000  001236    BIT     #ASYNC.STAT     ;IS THIS A SYNC LINE CARD?
2092  011164  001401                    BEQ     .+4             ;BR IF SYNC LINE CARD.
2093  011166  000207                    RTS     PC              ;EXIT TEST
2094  011170  104412                    MSTCLR                  ;RESET DV11
2095  011172  005002                    CLR     R2              ;CLEAR DVSFR IMAGE
2096  011174  012703  000004            MOV     #4,R3           ;SET TO DO 4 LINES
2097  011200  010077  170166    1$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2098  011204  004537  022266            PERFORM ,LOAD.MODE      ;LOAD THE MODE
2099  011210  005000                    BIT11+BIT9              ;INT WAIT MODE AND TX ENABLE
2100  011212  017705  170152            MOV     @DVLCR,R5       ;READ LSR
2101  011216  010504                    MOV     R5,R4           ;
2102  011220  042705  000200            BIC     #BIT7,R5        ;CLEAR WAIT BIT WINDOW RESULT
2103  011224  020504                    CMP     R5,R4           ;WAS BIT WINDOW =TO 0
2104  011226  001401                    BEQ     .+4             ;BR IF YES
2105  011230  104001                    HLT     1               ;BIT7 OF LCR S/B=0
2106  011232  012737  000012  001250    MOV     #10.,TEMP2      ;SET FOR 10 BITS
2107  011240  052705  040200    2$:     BIS     #BIT14+BIT7,R5  ;SET MAINT DATA AND MAINT BIT WINDOW
2108  011244  052777  140000  170116    BIS     #BIT15+BIT14,@DVLCR
2109  011252  004737  022406            JSR     PC,CKBIT15      ;STROBE MAINT DATA. WAIT BIT15=0
2110  011256  017704  170106            MOV     @DVLCR,R4       ;READ THE LCR
2111  011262  020504                    CMP     R5,R4           ;BIT14+BIT7=1?
2112  011264  001401                    BEQ     3$              ;YES
2113  011266  104001                    HLT     1               ;MAINT DATA DID NOT SHOW UP IN WINDOW
2114  011270  042705  040200    3$:     BIC     #BIT14+BIT7,R5  ;CLEAR DATA AND WINDOW
2115  011274  042777  040000  170066    BIC     #BIT14,@DVLCR   ;CLEAR WAIT DATA
2116  011302  052777  100000  170060    BIS     #BIT15,@DVLCR   ;SET STROBE ON DV11
2117  011310  004737  022406            JSR     PC,CKBIT15      ;WAIT 15=0
2118  011314  017704  170050            MOV     @DVLCR,R4       ;READ DVLCR
2119  011320  020504                    CMP     R5,R4           ;WINDOW =0?
2120  011322  001401                    BEQ     4$              ;BR IF YES
2121  011324  104001                    HLT     1               ;BIT7 S/B=0
2122  011326  005337  001250    4$:     DEC     TEMP2           ;10 BITS DONE?
2123  011332  001342                    BNE     2$              ;BR IF NO
2124  011334  005200                    INC     R0              ;UPDATE LINE POINTER
2125  011336  005303                    DEC     R3              ;4 LINE GROUP DONE?
2126  011340  001317                    BNE     1$              ;BR IF NO
2127  011342  000207                    RTS     PC              ;RETURN FOR NEXT GROUP
2128
```

```
                                        ;************************ TEST 10 ********************************
                                        ;*TEST TO XMIT A BINARY COUNT PATTERN
                                        ;*THUR THE USE OF THE BIT WINDOW.
                                        ;*ONLY ONE LINE AT A TIME WILL BE EXERCISED.
                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                                        ;;*************************************************************

                                ;  TEST 10
                                ;--------------
011344  012737  000010  001226  TST10:  MOV     #10,TSTNO
011352  012737  012224  001216          MOV     #TST11,NEXT
011360  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
011364  113737  001412  001242          MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
011372  113737  001406  001244          MOVB    MASK.A,MASKX    ;PLACE "MASK"FOR CHARS INTO MASKX
011400  013737  001416  001236          MOV     LC0.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
011406  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
011410  004737  011542                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
011414  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
011420  113737  001413  001242          MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
011426  113737  001407  001244          MOVB    MASK.B,MASKX    ;GET MASK
011434  013737  001420  001236          MOV     LC4.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
011442  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
011444  004737  011542                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
011450  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
011454  113737  001414  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
011462  113737  001410  001244          MOVB    MASK.C,MASKX    ;GET MASK
011470  013737  001422  001236          MOV     LC8.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
011476  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
011500  004737  011542                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
011504  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
011510  113737  001415  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
011516  113737  001411  001244          MOVB    MASK.D,MASKX    ;GET MASKK
011524  013737  001424  001236          MOV     LC12.15,STAT    ;LOAD LINE CARD STATUS
011532  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
011534  004737  011542                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
011540  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
011542                          105$:                           ;TEST ENTRANCE.
011542  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
011550  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
011552  000207                          RTS     PC              ;EXIT TEST
011554  010037  011570                  MOV     R0,65$          ;SET LINE NO. POINTER
011560  104412                          MSTCLR                  ;CLEAR DV11
011562  005001                          CLR     R1              ;ZERO MSCANNER POINTER
011564  004537  022470          1$:     PERFORM .SETSCAN        ;ADJUST SCANNER FOR PROPER LINE
011570  000001                  65$:    .BLKW 1                 ;LINE NUMBER POINTER.
011572  012703  000004          2$:     MOV     #4,R3           ;SET FOR 4 LINES EXERCISED
011576  005005                  3$:     CLR     R5              ;SET DATA POINTER TO 0
011600  012777  050102  167572          MOV     #S.C+BIT6+BIT1,@DVSFR
011606  104415                          ROMCLK                  ;CLOCK SCANNER BY ONE
011610  005201                          INC     R1              ;ADD +1 TO SCANNER POINTER
011612  010077  167554                  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
011616  004537  022266                  PERFORM ,LOAD.MODE      ;LOAD MODE
011622  004000                          BIT11
011624  004537  022560          7$:     PERFORM .CLR.TMARK      ;CLEAR TMARK BIT.
011630  012777  001000  167542          MOV     #BIT9,@DVSFR    ;DO A BR "A" TEST FOR TX FLAG
```

```
2195  011636  005005                          CLR     R5              ;SET EXPECTED DATA TO 0
2196  011640  032777  000001  167522          BIT     #BIT0,@DVLCR    ;IF FLAG TRUE?
2197  011646  001401                          BEQ     .+4             ;BR IF YES
2198  011650  104000                          HLT                     ;TX FLAG NO TRUE(LOW(LPRO=0))
2199  011652  005077  167514                  CLR     @DVSRS          ;ZERO LINE TO LINE 0
2200  011656  010577  167514                  MOV     R5,@DVSRA       ;LOAD DATA INTO DVSRA
2201  011662  012777  020000  167510          MOV     #BIT13,@DVSFR   ;EXECUTE A "ROM READ" INTSTR
2202  011670  104415                          ROMCLK                  ;CLOCK.
2203  011672  012777  030260  167500          MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
2204  011700  104415                          ROMCLK                  ;DO A DATA XFER FROM RAM OUTPUT TO TX BUFFER
2205  011702  104416                          DATACLK                 ;ISSUE A MAINT CLK.
2206  011704  012777  001000  167466          MOV     #BIT9,@DVSFR    ;DO A "BR A" TEST FOR TX FLAG
2207  011712  032777  000001  167450          BIT     #BIT0,@DVLCR    ;IS FLAG FALSE?
2208  011720  001001                          BNE     .+4             ;BR IF YES
2209  011722  104000                          HLT                     ;TX FLAG NOT FALSE(HIGH(LPRO=1))
2210  011724  012737  011732  001220          MOV     #4$,LOCK        ;SET IF SW09=1 GOTO 4$
2211  011732  113702  001242          4$:     MOVB    CLKX,R2         ;SET REQUIRED SHIFTS
2212  011736  005037  022622                  CLR     DATA            ;CLEAR STUFFER LOCATION
2213  011742  010077  167424                  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2214  011746  104416                  5$:     DATACLK                 ;ISSUE MAINT CLK
2215  011750  004537  022246                  PERFORM .TXSHIFT        ;WORK THE TRANSMITTER
2216  011754  005302                          DEC     R2              ;ALL SHIFTS DONE?
2217  011756  022702  000001                  CMP     #1,R2           ;IS THE BUFFER ALMOST EMPTY?
2218  011762  001030                          BNE     8$              ;BR IF NO
2219  011764  005077  167402                  CLR     @DVSRS          ;ZERO LINE NUMBER
2220  011770  032777  001000  167204          BIT     #BIT9,@SWR      ;LOCK ON DATA?
2221  011776  001001                          BNE     .+4             ;BR IF YES!!
2222  012000  005205                          INC     R5              ;UPDATE DATA.
2223  012002  010577  167370                  MOV     R5,@DVSRA       ;LOAD DATA INTO DVSRA
2224  012006  012777  020000  167364          MOV     #BIT13,@DVSFR   ;DO A ROM READ
2225  012014  104415                          ROMCLK                  ;CLK
2226  012016  012777  030260  167354          MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
2227  012024  104415                          ROMCLK                  ;DO A  DATA XFER TO TX BUFF
2228  012026  010077  167340                  MOV     R0,@DVSRS       ;RESELECT LINE NUMBER
2229  012032  032777  001000  167142          BIT     #BIT9,@SWR      ;LOCK ON DATA?
2230  012040  001001                          BNE     .+4             ;BR IF YES!!
2231  012042  005305                          DEC     R5              ;READJUST DATA CHAR.
2232  012044  005702                  8$:     TST     R2              ;ALL SHIFTS DONE?
2233  012046  001337                          BNE     5$              ;BR IF NO
2234  012050  022737  000010  001242          CMP     #8.,CLKX        ;IS LINE CARD SET TO 8 BITS?
2235  012056  001414                          BEQ     15$             ;BR IF YES
2236  012060  013737  001242  001246          MOV     CLKX,TEMP1      ;SAVE NUMBER OF SHIFTS DONE.
2237  012066  000241                  16$:    CLC                     ;CLEAR CARRY
2238  012070  006037  022622                  ROR     DATA            ;RIGHT JUSTIFY TX RESULTS.
2239  012074  005237  001246                  INC     TEMP1   ;ALL DONE?
2240  012100  022737  000010  001246          CMP     #8.,TEMP1       ;?
2241  012106  001367                          BNE     16$             ;BR IF NO
2242  012110                          15$:
2243  012110  013704  022622                  MOV     DATA,R4         ;READ IMAGE CHAR FROM TX
2244  012114  143704  001244                  BICB    MASKX,R4        ;STRIP PARITY IF IT EXISTS.
2245  012120  020504                          CMP     R5,R4           ;ARE DATA CHARS THE SAME?
2246  012122  001401                          BEQ     .+4             ;BR IF GOOD DATA FROM TX
2247  012124  104003                          HLT     3               ;TX DATA COMPARE ERROR
2248  012126  104401                          SCOP1                   ;LOCK ON DATA?
2249  012130  105205                          INCB    R5              ;UPDATE DATA CHAR.
2250  012132  001403                          BEQ     6$              ;BR IF 8BIT CODE DONE.
```

```
2241  012134  133705  001244                BITB    MASKX,R5        ;IF <8BIT SEE IF ALL DONE.
2242  012140  001674                        BEQ     4$              ;BR IF NOT ALL DONE
2243  012142  004537  022546          6$:   PERFORM .SET.TMARK      ;SET TMARK BIT
2244                                        ;*VERIFY THAT SETTING TMARK BIT PUTS LINE AT MARK.
2245                                        ;*
2246  012146  113702  001242                MOVB    CLKX,R2         ;SET COUNTER
2247  012152  010077  167214                MOV     R0,@DVSRS       ;SET LINE
2248  012156  104416                  9$:   DATACLK                 ;CLOCK
2249  012160  005302                        DEC     R2              ;FLUSH LAST CHARACTER.
2250  012162  001375                        BNE     9$              ;CHAR FLUSHED?
2251  012164  012702  000024                MOV     #20.,R2         ;LOOK AT 20. BITS.
2252  012170  104416                  10$:  DATACLK                 ;MAINT CLK
2253  012172  032777  000200  167170       BIT     #BIT7,@DVLCR    ;BIT WINDOW
2254  012200  001001                        BNE     11$             ;SET (MARK)
2255  012202  104000                        HLT     0               ;TX BIT WINDOW NOT SET (MARK)
2256  012204  005302                  11$:  DEC     R2              ;ALL BITS LOOKED AT?
2257  012206  001370                        BNE     10$             ;BR IF NO
2258  012210  004537  022470                PERFORM .SETSCAN        ;ADVANCE SCANNER TO NEXT LINE
2259  012214  000001                        1                       ;ONE LINE ADVANCE
2260  012216  005303                        DEC     R3              ;ALL LINES(4) DONE?
2261  012220  001201                        BNE     7$              ;BR IF NO
2262  012222  000207                        RTS     PC              ;GET NEXT GROUP OF 4 LINES.
2263
2264
2265                                        ;************************ TEST 11 ******************************
2266                                        ;*TEST TO CHECK THE IDLE CHARACTER
2267                                        ;*FOR EACH LINE OF THE TRANSMITTER.
2268                                        ;*THIS TEST USES "SYNCA".
2269                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2270                                        ;;*************************************************************
2271
2272                                        ;  TEST 11
2273                                        ;--------------
2274  012224  012737  000011  001226  TST11:  MOV     #11,TSTNO
2275  012232  012737  012740  001216          MOV     #TST12,NEXT
2276  012240  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2277  012244  113737  001412  001242          MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
2278  012252  113737  001406  001244          MOVB    MASK.A,MASKX    ;PLACE "MASK"FOR CHARS INTO MASKX
2279  012260  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2280  012266  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2281  012270  004737  012422                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2282  012274  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2283  012300  113737  001413  001242          MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
2284  012306  113737  001407  001244          MOVB    MASK.B,MASKX    ;GET MASK
2285  012314  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2286  012322  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2287  012324  004737  012422                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2288  012330  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2289  012334  113737  001414  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2290  012342  113737  001410  001244          MOVB    MASK.C,MASKX    ;GET MASK
2291  012350  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2292  012356  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2293  012360  004737  012422                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2294  012364  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
2295  012370  113737  001415  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
2296  012376  113737  001411  001244          MOVB    MASK.D,MASKX    ;GET MASKK
```

```
 2297  012404  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 2298  012412  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 2299  012414  004737  012422                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 2300  012420  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
 2301  012422                          105$:                           ;TEST ENTRANCE.
 2302  012422  032737  004000  001236          BIT     #ASYNC.STAT     ;IS THIS A SYNC LINE CARD?
 2303  012430  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
 2304  012432  000207                          RTS     PC              ;EXIT TEST
 2305  012434  010037  012450                  MOV     R0,65$          ;LOAD LINE NO. POINTER
 2306  012440  104412                          MSTCLR                  ;RESET THE DV11
 2307  012442  005001                          CLR     R1              ;ZERO MSCANNER POINTER
 2308  012444  004537  022470          1$:     PERFORM .SETSCAN        ;SET MSCANNER TO LINES TESTED
 2309  012450  000001          65$:    .BLKW 1                         ;INITAL LINE VALUE
 2310  012452  012703  000004          2$:     MOV     #4,R3           ;SET TO DO 4 LINE GROUP
 2311  012456  005005                  3$:     CLR     R5              ;ZERO
 2312  012460  012777  050102  166712          MOV     #S.C+BIT6+BIT1,@DVSFR
 2313  012466  104415                          ROMCLK                  ;SET/CLEAR "ADVANCE MSCANNER"
 2314  012470  005201                          INC     R1              ;UPDATE MSCANNER POINTER
 2315  012472  010077  166674          6$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER INTO DV11
 2316  012476  004537  022560                  PERFORM ,CLR.TMARK      ;CLR TMARK BIT.
 2317  012502  004537  022266                  PERFORM .LOAD.MODE      ;LOAD THE MODE
 2318  012506  004000                          BIT11                   ;INT MAINT MODE
 2319  012510  005077  166662                  CLR     @DVSRA          ;ZERO DATA FOR XFR
 2320  012514  012777  020000  166656          MOV     #BIT13,@DVSFR   ;DO A RAM READ INSTR.
 2321  012522  104415                          ROMCLK                  ;
 2322  012524  012777  030260  166646          MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
 2323  012532  104415                          ROMCLK                  ;DATA XFR  TXBUFFER+RAM OUTPUT
 2324  012534  104416                          DATACLK                 ;ISSUE MAIT CLOCK PULSE
 2325  012536  012737  012570  001220          MOV     #4$,LOCK        ;SET FOR SCOPI
 2326  012544  113702  001242                  MOVB    CLKX,R2         ;NUMBER OF CLOCK PULSES NEEDED
 2327  012550  104416                          DATACLK                 ;MAINT CLOCK PULSE
 2328  012552  005302                          DEC     R2              ;ALL CLOCKS DONE?
 2329  012554  001375                          BNE     .-4             ;NO . DO MORE
 2330  012556  113705  001236                  MOVB    STAT,R5         ;GET SYNC (IDLE) CHAR.
 2331  012562  012737  000005  001250          MOV     #5,TEMP2        ;SET FOR 5 CHARS
 2332  012570  113702  001242          4$:     MOVB    CLKX,R2         ;GET CLOCKS NEEDED
 2333  012574  005037  022622                  CLR     DATA            ;ZERO STORAGE AREA
 2334  012600  010077  166566                  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
 2335  012604  104416                  5$:     DATACLK                 ;ISSUE MAINT CLK PULSE
 2336  012606  004537  022246                  PERFORM ,TXSHIFT        ;CLOCK THE TRANSMITTER
 2337  012612  005302                          DEC     R2              ;MORE SHIFTS REQUIRED?
 2338  012614  001373                          BNE     5$              ;BR IF YES
 2339  012616  022737  000010  001242          CMP     #8.,CLKX        ;IS LINE CARD SET TO 8 BITS?
 2340  012624  001414                          BEQ     15$             ;BR IF YES
 2341  012626  013737  001242  001246          MOV     CLKX,TEMP1      ;SAVE NUMBER OF SHIFTS DONE.
 2342  012634  000241                  16$:    CLC                     ;CLEAR CARRY
 2343  012636  006037  022622                  ROR     DATA            ;RIGHT JUSTIFY TX RESULTS.
 2344  012642  005237  001246                  INC     TEMP1   ;ALL DONE?
 2345  012646  022737  000010  001246          CMP     #8.,TEMP1       ;?
 2346  012654  001367                          BNE     16$             ;BR IF NO
 2347  012656                          15$:
 2348  012656  013704  022622                  MOV     DATA,R4         ;SAVE DATA SHIFTED OUT OF TX.
 2349  012662  143704  001244                  BICB    MASKX,R4        ;CLEAR UNWANTED BITS.
 2350  012666  042705  177400                  BIC     #^C<377>,R5     ;CLEAR SIGN EXTEND.
 2351  012672  143705  001244                  BICB    MASKX,R5        ;CLEAR UNUSED BITS
 2352  012676  042704  177400                  BIC     #^C<377>,R4     ;CLEAR SIGN EXTEND.
```

```
2353  012702  020504                        CMP     R5,R4           ;EXPECTED = FOUND ??
2354  012704  001401                        BEQ     .+4             ;BR IF OK
2355  012706  104003                        HLT     3               ;IDLE CHAR NOT WHAT EXPECTED.
2356  012710  005337  001250                DEC     TEMP2           ;ALL IDLE CHARS DONE?
2357  012714  001325                        BNE     4$              ;BR IF NO
2358  012716  104401                        SCOP1                   ;LOCK (SW09=1)?
2359  012720  004537  022546                PERFORM ,SET.TMARK      ;SET TMARK BIT
2360  012724  004537  022470                PERFORM ,SETSCAN        ;UPDATE SCANNER TO NEXT LINE
2361  012730  000001                        1                       ;
2362  012732  005303                        DEC     R3              ;ALL LINES DONE
2363  012734  001256                        BNE     6$              ;BR IF NO
2364  012736  000207                        RTS     PC              ;EXIT FOR NEXT GROUP OF LINES.
2365
2366
2367                                        ;********************** TEST 12 ****************************
2368                                        ;*TEST TO CHECK THE IDLE CHARACTER
2369                                        ;*FOR EACH LINE OF THE TRANSMITTER.
2370                                        ;*THIS TEST USES "SYNC8".
2371                                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2372                                        ;:*************************************************************
2373
2374                                ;  TEST 12
2375                                ;--------------
2376  012740  012737  000012  001226 TST12:  MOV     #12,TSTNO
2377  012746  012737  013504  001216         MOV     #TST13,NEXT
2378  012754  012700  000000                MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2379  012760  113737  001412  001242        MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
2380  012766  113737  001406  001244        MOVB    MASK.A,MASKX    ;PLACE "MASK"FOR CHARS INTO MASKX
2381  012774  013737  001426  001240        MOV     SYNC2A,SYNCX    ;
2382  013002  013737  001416  001236        MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2383  013010  100402                        BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2384  013012  004737  013166                JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2385  013016  012700  000004         100$:  MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2386  013022  113737  001413  001242        MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
2387  013030  113737  001407  001244        MOVB    MASK.B,MASKX    ;GET MASK
2388  013036  013737  001430  001240        MOV     SYNC2B,SYNCX    ;
2389  013044  013737  001420  001236        MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2390  013052  100402                        BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2391  013054  004737  013166                JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2392  013060  012700  000010         101$:  MOV     #8.,R0          ;LOAD LINE NUMBER
2393  013064  113737  001414  001242        MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2394  013072  113737  001410  001244        MOVB    MASK.C,MASKX    ;GET MASK
2395  013100  013737  001432  001240        MOV     SYNC2C,SYNCX    ;
2396  013106  013737  001422  001236        MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2397  013114  100402                        BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2398  013116  004737  013166                JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2399  013122  012700  000014         102$:  MOV     #12.,R0         ;LOAD LINE NO.
2400  013126  113737  001415  001242        MOVB    CLK.D,CLKX      ;GET SHIFTS
2401  013134  113737  001411  001244        MOVB    MASK.D,MASKX    ;GET MASKK
2402  013142  013737  001434  001240        MOV     SYNC2D,SYNCX    ;
2403  013150  013737  001424  001236        MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2404  013156  100402                        BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2405  013160  004737  013166                JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2406  013164  104400                 103$:  SCOPE                   ;SCOPE THIS TEST.
2407  013166                         105$:                          ;TEST ENTRANCE.
2408  013166  032737  004000  001236        BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
```

```
2409  013174  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
2410  013176  000207                          RTS     PC              ;EXIT TEST
2411  013200  010037  013214                  MOV     R0,65$          ;LOAD LINE NO. POINTER
2412  013204  104412                          MSTCLR                  ;RESET THE DV11
2413  013206  005001                          CLR     R1              ;ZERO MSCANNER POINTER
2414  013210  004537  022470          1$:     PERFORM .SETSCAN        ;SET MSCANNER TO LINES TESTED
2415  013214  000001                  65$:    .BLKW 1                 ;INITAL LINE VALUE
2416  013216  012703  000004          2$:     MOV     #4,R3           ;SET TO DO 4 LINE GROUP
2417  013222  005005                  3$:     CLR     R5              ;ZERO
2418  013224  012777  050102  166146          MOV     #S.C+BIT6+BIT1,@DVSFR
2419  013232  104415                          ROMCLK                  ;SET/CLEAR "ADVANCE MSCANNER"
2420  013234  005201                          INC     R1              ;UPDATE MSCANNER POINTER
2421  013236  010077  166130          6$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER INTO DV11
2422  013242  004537  022560                  PERFORM ,CLR.TMARK      ;CLR TMARK BIT.
2423  013246  004537  022266                  PERFORM ,LOAD.MODE      ;LOAD THE MODE
2424  013252  006000                          BIT11+BIT10             ;INT MAINT MODE AND SECOND SYNC
2425  013254  005077  166116                  CLR     @DVSRA          ;ZERO DATA FOR XFR
2426  013260  012777  020000  166112          MOV     #BIT13,@DVSFR   ;DO A RAM READ INSTR.
2427  013266  104415                          ROMCLK                  ;
2428  013270  012777  030260  166102          MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
2429  013276  104415                          ROMCLK                  ;DATA XFR  TXBUFFER←RAM OUTPUT
2430  013300  104416                          DATACLK                 ;ISSUE MAIT CLOCK PULSE
2431  013302  012737  013334  001220          MOV     #4$,LOCK        ;SET FOR SC' 1
2432  013310  113702  001242                  MOVB    CLKX,R2         ;NUMBER OF CLOCK PULSES NEEDED
2433  013314  104416                          DATACLK                 ;MAINT CLOCK PULSE
2434  013316  005302                          DEC     R2              ;ALL CLOCKS DONE?
2435  013320  001375                          BNE     .-4             ;NO , DO MORE
2436  013322  113705  001240                  MOVB    SYNCX,R5        ;GET SYNC (IDLE CHAR).
2437  013326  012737  000005  001250          MOV     #5,TEMP2        ;SET FOR 5 CHARS
2438  013334  113702  001242          4$:     MOVB    CLKX,R2         ;GET CLOCKS NEEDED
2439  013340  005037  022622                  CLR     DATA            ;ZERO STORAGE AREA
2440  013344  010077  166022                  MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2441  013350  104416                  5$:     DATACLK                 ;ISSUE MAINT CLK PULSE
2442  013352  004537  022246                  PERFORM ,TXSHIFT        ;CLOCK THE TRANSMITTER
2443  013356  005302                          DEC     R2              ;MORE SHIFTS REQUIRED?   .
2444  013360  001373                          BNE     5$              ;BR IF YES
2445  013362  022737  000010  001242          CMP     #8.,CLKX        ;IS LINE CARD SET TO 8 BITS?
2446  013370  001414                          BEQ     15$             ;BR IF YES
2447  013372  013737  001242  001246          MOV     CLKX,TEMP1      ;SAVE NUMBER OF SHIFTS DONE.
2448  013400  000241                  16$:    CLC                     ;CLEAR CARRY
2449  013402  006037  022622                  ROR     DATA            ;RIGHT JUSTIFY TX RESULTS.
2450  013406  005237  001246                  INC     TEMP1   ;ALL DONE?
2451  013412  022737  000010  001246          CMP     #8.,TEMP1       ;?
2452  013420  001367                          BNE     16$             ;BR IF NO
2453  013422                          15$:
2454  013422  013704  022622                  MOV     DATA,R4         ;SAVE DATA SHIFTED OUT OF TX
2455  013426  143704  001244                  BICB    MASKX,R4        ;CLEAR UNWANTED BITS.
2456  013432  042705  177400                  BIC     #↑C<377>,R5     ;CLEAR SIGN EXTEND.
2457  013436  143705  001244                  BICB    MASKX,R5        ;CLEAR UNUSED BITS
2458  013442  042704  177400                  BIC     #↑C<377>,R4     ;CLEAR SIGN EXTEND.
2459  013446  020504                          CMP     R5,R4           ;EXPECTED = FOUND ??
2460  013450  001401                          BEQ     .+4             ;BR IF OK
2461  013452  104003                          HLT     3               ;IDLE CHAR NOT WHAT EXPECTED.
2462  013454  005337  001250                  DEC     TEMP2           ;ALL IDLE CHARS DONE?
2463  013460  001325                          BNE     4$              ;BR IF NO
2464  013462  104401                          SCOP1                   ;LOCK (SW09=1)?
```

```
 2465  013464  004537  022546                PERFORM .SET.TMARK      ;SET TMARK BIT
 2466  013470  004537  022470                PERFORM .SETSCAN        ;UPDATE SCANNER TO NEXT LINE
 2467  013474  000001                         1                       ;
 2468  013476  005303                         DEC     R3              ;ALL LINES DONE
 2469  013500  001256                         BNE     6$              ;BR IF NO
 2470  013502  000207                         RTS     PC              ;EXIT FOR NEXT GROUP OF LINES.
 2471
 2472
 2473                                         ;************************ TEST 13 ******************************
 2474                                         ;*THIS TEST CHECKS "RECEIVE CHAR SILO" TO BE
 2475                                         ;*ALL ZERO'S WHEN "DATA ENABLE" IS NOT SET.
 2476                                         ;*EXPECTED DATA SHOULD BE LINE NUMBER ONLY
 2477                                         ;*DATA 0'S AND ERROR FLAGS 0.
 2478                                         ;*THIS TEST WILL BE DONE FOR BOTH ASYNC AND SYNC LINE CARDS.
 2479                                         ;:**************************************************************
 2480
 2481                                 ;  TEST 13
 2482                                 ;--------------
 2483  013504  012737  000013  001226 TST13:  MOV     #13,TSTNO
 2484  013512  012737  014012  001216         MOV     #TST14,NEXT
 2485  013520  012700  000000                 MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 2486  013524  013737  001416  001236         MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2487  013532  100402                         BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 2488  013534  004737  013622                 JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 2489  013540  012700  000004         100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 2490  013544  013737  001420  001236         MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2491  013552  100402                         BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 2492  013554  004737  013622                 JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 2493  013560  012700  000010         101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 2494  013564  013737  001422  001236         MC      L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2495  013572  100402                         BMI     [illegible]$    ;BR IF LINE CARD NOT TO BE TESTED
 2496  013574  004737  013622                 JSR     [illegible]05$  ;DO THE TEST FOR LINE CARD 3
 2497  013600  012700  000014         102$:   MOV     #1[illegible]   ;LOAD LINE NO.
 2498  013604  013737  001424  001236         MOV     L12.15,[illegible]AT ;LOAD LINE CARD STATUS
 2499  013612  100402                         BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 2500  013614  004737  013622                 JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 2501  013620  104400                 103$:   SCOPE                   ;SCOPE THIS TEST.
 2502  013622                         105$:                           ;TEST ENTRANCE.
 2503  013622  010037  013642                 MOV     R0,65$          ;STORE LINE NO. POINTER
 2504  013626  012703  000004                 MOV     #4,R3           ;SET FOR 4 LINE GROUP
 2505  013632  104412                 1$:     MSTCLR                  ;RESET DV11
 2506  013634  005001                         CLR     R1              ;ZERO MSCANNER POINTER
 2507  013636  004537  022470                 PERFORM ,SETSCAN         ;ADJUST SCANNER
 2508  013642  000001                 65$:    .BLKW 1                 ;TO CORRECT LINE NO.
 2509  013644  010005                         MOV     R0,R5           ;PLACE LINE NUMBER INTO R5
 2510  013646  000305                         SWAB    R5              ;PLACE LINE NO. IN HIGH BYTE
 2511  013650  105005                         CLRB    R5              ;CLEAR LOW BYTE OF EXPECTED
 2512  013652                         3$:
 2513  013652  012777  050021  165520         MOV     #S.C+BIT4+BIT0,@DVSFR
 2514  013660  104415                         ROMCLK                  ;SET/CLEAR SILO IN
 2515  013662  005002                         CLR     R2              ;
 2516  013664  012777  001400  165506         MOV     #BIT9+BIT8,@DVSFR
 2517  013672  032777  000001  165470 4$:     BIT     #BIT0,@DVLCR    ;"RECV CHAR WAITING TRUE"
 2518  013700  001403                         BEQ     5$              ;BR IF YES
 2519  013702  005202                         INC     R2              ;DELAY IF NOT READY
 2520  013704  001372                         BNE     4$              ;END OF DELAY?
```

```
2521  013706  104000                          HLT     0               ;"RECV CHAR WAITING" NOT TRUE
2522  013710  012777  030306  165462  5$:     MOV     #XFR+BIT7+BIT6+BIT2+BIT1,@DVSFR
2523  013716  017702  165456                  MOV     @DVSFR,R2       ;XFR RICR←SILO OUT
2524  013722  104415                          ROMCLK                  ;DATA/XFER  RICR←SILO OUT
2525  013724  017704  165436                  MOV     @DVRIC,R4       ;READ RIC
2526  013730  020504                          CMP     R5,R4           ;EXPECTED OK?
2527  013732  001401                          BEQ     .+4             ;
2528  013734  104001                          HLT     1               ;
2529  013736  062705  000400                  ADD     #400,R5         ;UPDATE LINE NO. (POINTER)
2530  013742  005002                          CLR     R2              ;SFR IMAGE
2531  013744  012777  050020  165426          MOV     #S.C+BIT4,@DVSFR
2532  013752  104415                          ROMCLK                  ;S/C "SET SILO OUT"
2533  013754  012777  001400  165416          MOV     #BIT9+BIT8,@DVSFR
2534  013762  032777  000001  165400  6$:     BIT     #BIT0,@DVLCR    ;"RECV CHAR WAITING"
2535  013770  001003                          BNE     7$              ;FALSE?
2536  013772  005202                          INC     R2              ;DELAY WAITING....
2537  013774  001372                          BNE     6$              ;DELAY DONE?
2538  013776  104000                          HLT     0               ;
2539  014000  005237  013642          7$:     INC     65$             ;UPDATE MSCANNER POINTER(LINE)
2540  014004  005303                          DEC     R3              ;GROUP OF 4 LINES DONE.
2541  014006  001311                          BNE     1$              ;BR IF YES
2542  014010  000207                          RTS     PC              ;EXIT FOR NEXT GROUP OF LINES
2543
2544
2545                                          ;************************ TEST 14 *******************************
2546                                          ;*THIS TEST CHECKS "RECEIVER CHAR SILO"
2547                                          ;*WHEN "DATA ENABLE IS SET" EXPECTED DATA S/B
2548                                          ;*ALL 1'S FOR RX DATA, LINE NUMBER CORRECT,
2549                                          ;*AND ERROR FLAGS =0.
2550                                          ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2551                                          ;:************************************************************
2552
2553                                  ;  TEST 14
2554                                  ;--------------
2555  014012  012737  000014  001226  TST14:  MOV     #14,TSTNO
2556  014020  012737  014344  001216          MOV     #TST15,NEXT
2557  014026  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2558  014032  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2559  014040  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2560  014042  004737  014130                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2561  014046  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2562  014052  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2563  014060  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2564  014062  004737  014130                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2565  014066  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2566  014072  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2567  014100  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2568  014102  004737  014130                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2569  014106  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
2570  014112  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2571  014120  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2572  014122  004737  014130                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2573  014126  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
2574  014130                          105$:                           ;TEST ENTRANCE.
2575  014130  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2576  014136  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
```

```
2577  014140  000207                          RTS     PC              ;EXIT TEST
2578  014142  010037  014162                  MOV     R0,65$          ;STORE LINE NO. POINTER
2579  014146  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINE GROUP
2580  014152  104412                  1$:     MSTCLR                  ;RESET DV11
2581  014154  005001                          CLR     R1              ;ZERO MSCANNER POINTER
2582  014156  004537  022470                  PERFORM ,SETSCAN         ;ADJUST SCANNER
2583  014162  000001                  65$:    .BLKW 1                 ;TO CORRECT LINE NO.
2584  014164  010005                          MOV     R0,R5           ;PLACE LINE NUMBER INTO R5
2585  014166  000305                          SWAB    R5              ;PLACE LINE NO. IN HIGH BYTE
2586  014170  052705  000377                  BIS     #377,R5         ;SET LOW BYTE TO ALL 1'S
2587  014174                           3$:
2588  014174  012777  050023  165176          MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
2589  014202  104415                          ROMCLK                  ;S/C "SET RECV DATA ENABLE"
2590  014204  012777  050021  165166          MOV     #S.C+BIT4+BIT0,@DVSFR
2591  014212  104415                          ROMCLK                  ;SET/CLEAR SILO IN
2592  014214  005002                          CLR     R2              ;
2593  014216  012777  001400  165154          MOV     #BIT9+BIT8,@DVSFR
2594  014224  032777  000001  165136  4$:     BIT     #BIT0,@DVLCR    ;"RECV CHAR WAITING TRUE"
2595  014232  001403                          BEQ     5$              ;BR IF YES
2596  014234  005202                          INC     R2              ;DELAY IF NOT READY
2597  014236  001372                          BNE     4$              ;END OF DELAY?
2598  014240  104000                          HLT     0               ;"RECV CHAR WAITING" NOT TRUE
2599  014242  012777  030306  165130  5$:     MOV     #XFR+BIT7+BIT6+BIT2+BIT1,@DVSFR
2600  014250  017702  165124                  MOV     @DVSFR,R2       ;XFR RICR-SILO OUT
2601  014254  104415                          ROMCLK                  ;DATA/XFER  RICR-SILO OUT
2602  014256  017704  165104                  MOV     @DVRIC,R4       ;READ RIC
2603  014262  020504                          CMP     R5,R4           ;EXPECTED OK?
2604  014264  001401                          BEQ     .+4             ;
2605  014266  104001                          HLT     1               ;
2606  014270  062705  000400                  ADD     #400,R5         ;UPDATE LINE NO. (POINTER)
2607  014274  005002                          CLR     R2              ;SFR IMAGE
2608  014276  012777  050020  165074          MOV     #S.C+BIT4,@DVSFR
2609  014304  104415                          ROMCLK                  ;S/C "SET SILO OUT"
2610  014306  012777  001400  165064          MOV     #BIT9+BIT8,@DVSFR
2611  014314  032777  000001  165046  6$:     BIT     #BIT0,@DVLCR    ;"RECV CHAR WAITING"
2612  014322  001003                          BNE     7$              ;FALSE?
2613  014324  005202                          INC     R2              ;DELAY WAITING....
2614  014326  001372                          BNE     6$              ;DELAY DONE?
2615  014330  104000                          HLT     0               ;
2616  014332  005237  014162          7$:     INC     65$             ;UPDATE MSCANNER POINTER(LINE)
2617  014336  005303                          DEC     R3              ;GROUP OF 4 LINES DONE.
2618  014340  001304                          BNE     1$              ;BR IF YES
2619  014342  000207                          RTS     PC              ;EXIT FOR NEXT GROUP OF LINES
2620
2621
2622                                  ;********************** TEST 15 ******************************
2623                                  ;*TEST THAT EACH RECEIVER WILL SET
2624                                  ;*"MATCH DETECT" WHEN THE FIRST SYNC
2625                                  ;*CHARACTER IS PUMPED INTO IT.
2626                                  ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
2627                                  ;:************************************************************
2628
2629                                  ;  TEST 15
2630                                  ;--------------
2631  014344  012737  000015  001226  TST15:  MOV     #15,TSTNO
2632  014352  012737  014642  001216          MOV     #TST16,NEXT
```

```
 2633  014360  012700  000000            MOV     #0,,R0          ;PLACE LINE NUMBER INTO R0
 2634  014364  113737  001412  001242    MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
 2635  014372  013737  001416  001236    MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2636  014400  100402                    BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 2637  014402  004737  014512            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 2638  014406  012700  000004    100$:   MOV     #4,,R0          ;PLACE LINE NUMBER INTO R0
 2639  014412  113737  001413  001242    MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
 2640  014420  013737  001420  001236    MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2641  014426  100402                    BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 2642  014430  004737  014512            JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 2643  014434  012700  000010    101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 2644  014440  113737  001414  001242    MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
 2645  014446  013737  001422  001236    MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2646  014454  100402                    BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 2647  014456  004737  014512            JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 2648  014462  012700  000014    102$:   MOV     #12.,R0         ;LOAD LINE NO.
 2649  014466  113737  001415  001242    MOVB    CLK.D,CLKX      ;GET SHIFTS
 2650  014474  013737  001424  001236    MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 2651  014502  100402                    BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 2652  014504  004737  014512            JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 2653  014510  104400            103$:   SCOPE                   ;SCOPE THIS TEST.
 2654  014512                    105$:   ;TEST ENTRANCE.
 2655  014512  032737  004000  001236    BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
 2656  014520  001401                    BEQ     .+4             ;BR IF SYNC LINE CARD.
 2657  014522  000207                    RTS     PC              ;EXIT TEST
 2658  014524  012703  000004            MOV     #4,R3
 2659  014530  010037  014544            MOV     R0,65$          ;SET LINE NO. POINTER
 2660  014534  104412            1$:     MSTCLR                  ;RESET DV11
 2661  014536  005001                    CLR     R1              ;ZERO MSCANNER POINTER
 2662  014540  004537  022470            PERFORM .SETSCAN
 2663  014544  000001            65$:    .BLKW 1                 ;SET MSCANNER TO CORRECT LINE.
 2664  014546  010077  164620    3$:     MOV     R0,@DVSRS       ;LOAD LINE NO.
 2665  014552  004537  022266            PERFORM .LOAD.MODE      ;LOAD THE MODE
 2666  014556  025000                    BIT13+BIT11+BIT9        ;RECV ENABLE,INT MAINT,TX DSABLE
 2667  014560  113737  001236  022622    MOVB    STAT,DATA       ;GET "SYNC" CHAR.
 2668  014566  104416                    DATACLK                 ;PRIME DV11
 2669  014570  004537  022326            PERFORM .RXSHIFT        ;SHIFT DATA INTO RECEIVER
 2670  014574  001242                    CLKX                    ;NO. OF SHIFTS GIVEN
 2671  014576  012777  076400  164574    MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
 2672  014604  017704  164560            MOV     @DVLCR,R4       ;BRB "MATCH DET"
 2673  014610  010405                    MOV     R4,R5           ;
 2674  014612  052705  000001            BIS     #BIT0,R5        ;
 2675  014616  042705  000002            BIC     #BIT1,R5        ;
 2676  014622  020504                    CMP     R5,R4           ;MATCH DET TRUE??
 2677  014624  001401                    BEQ     4$              ;BR IF YES
 2678  014626  104001                    HLT     1               ;
 2679  014630  005237  014544    4$:     INC     65$             ;UPDATE TO NEXT LINE.
 2680  014634  005303                    DEC     R3              ;4 LINE GROUP DONE?
 2681  014636  001336                    BNE     1$              ;BR IF NO
 2682  014640  000207                    RTS     PC              ;OBTAIN NEXT 4 LINE GROUP
 2683
 2684
 2685                                    ;************************ TEST 16 *********************************
 2686                                    ;*TEST TO VERIFY THAT IF THE DV11 RECEIVER
 2687                                    ;*IS SET FOR ONE SYNC CHAR;
 2688                                    ;*"MATCH DET" *AND* "CHAR FLAG" ARE
```

```
                                                ;*SET AFTER ONE SYNC IS PUSHED INTO THE RECEIVER
                                                ;*                HOWEVER..
                                                ;*IF THE DV11 RECEIVER IS SET FOR
                                                ;*TWO SYNC CHARS....
                                                ;*VERIFY THAT "MATCH DET" SETS ON THE FIRST SYNC
                                                ;*AND VERIFY THAT "MATCH DET" *AND* "CHAR FLAG"
                                                ;*ARE SET ON THE SECOND SYNC.
                                                ;*THIS TEST USES "SYNC A".
                                                ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                                                ;;*************************************************

                                        ;  TEST 16
                                        ;--------------
014642  012737  000016  001226  TST16:  MOV     #16,TSTNO
014650  012737  015276  001216          MOV     #TST17,NEXT
014656  012700  000000                 MOV     #0,,R0          ;PLACE LINE NUMBER INTO R0
014662  113737  001412  001242         MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
014670  013737  001416  001236         MOV     LC0.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
014676  100402                       BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
014700  004737  015010                JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
014704  012700  000004         100$:  MOV     #4,,R0          ;PLACE LINE NUMBER INTO R0
014710  113737  001413  001242         MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
014716  013737  001420  001236         MOV     LC4.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
014724  100402                       BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
014726  004737  015010                JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
014732  012700  000010         101$:  MOV     #8.,R0          ;LOAD LINE NUMBER
014736  113737  001414  001242         MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
014744  013737  001422  001236         MOV     LC8.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
014752  100402                       BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
014754  004737  015010                JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
014760  012700  000014         102$:  MOV     #12.,R0         ;LOAD LINE NO.
014764  113737  001415  001242         MOVB    CLK.D,CLKX      ;GET SHIFTS
014772  013737  001424  001236         MOV     LC12.15,STAT    ;LOAD LINE CARD STATUS
015000  100402                       BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
015002  004737  015010                JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
015006  104400                 103$:  SCOPE                   ;SCOPE THIS TEST.
015010                         105$:                          ;TEST ENTRANCE.
015010  032737  004000  001236         BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
015016  001401                       BEQ     .+4             ;BR IF SYNC LINE CARD.
015020  000207                       RTS     PC              ;EXIT TEST
015022  012703  000004                MOV     #4,R3           ;SET FOR 4 LINES
015026  010037  015042                MOV     R0,65$          ;PLACE LINE NO. POINTER
015032  104412                 1$:    MSTCLR                  ;INIT DV11
015034  005001                       CLR     R1              ;ZERO MSCANNER POINTER
015036  004537  022470                PERFORM ,SETSCAN        ;SET SCANNER TO LINE DESIRED
015042  000001                 65$:   .BLKW 1                 ;INITIAL LINE NUMBER.
015044  010077  164322           3$:   MOV     R0,@DVSRS       ;LOAD LINE NUMBER
015050  004537  022266                PERFORM ,LOAD.MODE      ;LOAD
015054  025000                       BIT13+BIT11+BIT9        ;MODE AND RX ENABLE&NO TX DSABLE
015056  113737  001236  022622         MOVB    STAT,DATA       ;PLACE SYNC CHAR IN DATA
015064  104416                       DATACLK                 ;INIT DATA CLOCK.
015066  004537  022326                PERFORM ,RXSHIFT        ;SHIFT DATA INTO RX
015072  001242                       CLKX                    ;NUMBER OF SHIFTS NEEDED
015074  012777  076400  164276         MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
                                                ;SET BR "B" AND MATCH DET.
015102  017704  164262                MOV     @DVLCR,R4       ;SAVE LPR IN R4
```

```
                015106  010405                  MOV     R4,R5           ;SET FOR COMPARE
                015110  052705  000001          BIS     #BIT0,R5        ;BR "A" FALSE
                015114  042705  000002          BIC     #BIT1,R5        ;BR "B" TRUE
                015120  020504                  CMP     R5,R4
                015122  001401                  BEQ     .+4             ;BR IF LPR OK.
                015124  104001                  HLT     1               ;EXPECT B TRUE; A FALSE
                015126  012777  002000  164244  MOV     #BIT10,@DVSFR   ;SET BR "A" AND RX CHAR FLAG.
                015134  017704  164230          MOV     @DVLCR,R4       ;SAVE LPR IN R4
                015140  010405                  MOV     R4,R5           ;SET FOR COMPARE
                015142  032737  010000  001236  BIT     #TWOSYN,STAT    ;SET FOR ONE SYNC OR TWO?
                015150  001036                  BNE     4$              ;BR IF SET FOR ONE SYNC
                015152  052705  000003          BIS     #BIT1+BIT0,R5
                015156  020504                  CMP     R5,R4
                015160  001401                  BEQ     .+4
                015162  104001                  HLT     1
                015164  113737  001236  022622  MOVB    STAT,DATA
                015172  004537  022326          PERFORM .RXSHIFT
                015176  001242                  CLKX
                015200  012777  076400  164172  MOV     #BR8+BIT11+BIT10+BIT8,@DVSFR
                                                                        ;SET BR "B" AND MATCH DET.
                015206  017704  164156          MOV     @DVLCR,R4       ;SAVE LPR IN R4
                015212  010405                  MOV     R4,R5           ;SET FOR COMPARE
                015214  052705  000001          BIS     #BIT0,R5        ;BR "A" FALSE
                015220  042705  000002          BIC     #BIT1,R5        ;BR "B" TRUE
                015224  020504                  CMP     R5,R4
                015226  001401                  BEQ     .+4             ;BR IF LPR OK.
                015230  104001                  HLT     1               ;EXPECT B TRUE; A FALSE
                015232  012777  002000  164140  MOV     #BIT10,@DVSFR   ;SET BR "A" AND RX CHAR FLAG.
                015240  017704  164124          MOV     @DVLCR,R4       ;SAVE LPR IN R4
                015244  010405                  MOV     R4,R5           ;SET FOR COMPARE
                015246  052705  000002  4$:     BIS     #BIT1,R5
                015252  042705  000001          BIC     #BIT0,R5
                015256  020504                  CMP     R5,R4
                015260  001401                  BEQ     .+4
                015262  104001                  HLT     1
                015264  005237  015042          INC     65$             ;UPDATE LINE NUMBER
                015270  005303                  DEC     R3
                015272  001257                  BNE     1$
                015274  000207                  RTS     PC


                        ;************************ TEST 17 ********************************
                        ;*TEST TO VERIFY THAT IF THE DV11 RECEIVER
                        ;*IS SET FOR ONE SYNC CHAR;
                        ;*"MATCH DET" *AND* "CHAR FLAG" ARE
                        ;*SET AFTER ONE SYNC IS PUSHED INTO THE RECEIVER
                        ;*              HOWEVER...
                        ;*IF THE DV11 RECEIVER IS SET FOR
                        ;*TWO SYNC CHARS....
                        ;*VERIFY THAT "MATCH DET" SETS ON THE FIRST SYNC
                        ;*AND VERIFY THAT "MATCH DET" *AND* "CHAR FLAG"
                        ;*ARE SET ON THE SECOND SYNC.
                        ;*THIS TEST USES "SYNC B".
                        ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
                        ;;*************************************************************
```

```
2801                                        ;  TEST 17
2802                                        ;--------------
2803  015276  012737  000017  001226  TST17:  MOV     #17,TSTNO
2804  015304  012737  015762  001216           MOV     #TST20,NEXT
2805  015312  012700  000000                   MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
2806  015316  113737  001412  001242           MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
2807  015324  013737  001426  001240           MOV     SYNC2A,SYNCX    ;
2808  015332  013737  001416  001236           MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
2809  015340  100402                           BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
2810  015342  004737  015474                   JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
2811  015346  012700  000004           100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
2812  015352  113737  001413  001242           MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
2813  015360  013737  001430  001240           MOV     SYNC2B,SYNCX    ;
2814  015366  013737  001420  001236           MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
2815  015374  100402                           BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
2816  015376  004737  015474                   JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
2817  015402  012700  000010           101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
2818  015406  113737  001414  001242           MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
2819  015414  013737  001432  001240           MOV     SYNC2C,SYNCX    ;
2820  015422  013737  001422  001236           MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
2821  015430  100402                           BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
2822  015432  004737  015474                   JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
2823  015436  012700  000014           102$:   MOV     #12.,R0         ;LOAD LINE NO.
2824  015442  113737  001415  001242           MOVB    CLK.D,CLKX      ;GET SHIFTS
2825  015450  013737  001434  001240           MOV     SYNC2D,SYNCX    ;
2826  015456  013737  001424  001236           MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
2827  015464  100402                           BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
2828  015466  004737  015474                   JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
2829  015472  104400           103$:   SCOPE                           ;SCOPE THIS TEST.
2830  015474                   105$:                                   ;TEST ENTRANCE.
2831  015474  032737  004000  001236           BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
2832  015502  001401                           BEQ     .+4             ;BR IF SYNC LINE CARD.
2833  015504  000207                           RTS     PC              ;EXIT TEST
2834  015506  012703  000004                   MOV     #4,R3           ;SET FOR 4 LINES
2835  015512  010037  015526                   MOV     R0,65$          ;PLACE LINE NO. POINTER
2836  015516  104412           1$:     MSTCLR                          ;INIT DV11
2837  015520  005001                           CLR     R1              ;ZERO MSCANNER POINTER
2838  015522  004537  022470                   PERFORM .SETSCAN         ;SET SCANNER TO LINE DESIRED
2839  015526  000001           65$:    .BLKW 1                         ;INITIAL LINE NUMBER.
2840  015530  010077  163636           3$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
2841  015534  004537  022266                   PERFORM .LOAD.MODE       ;LOAD
2842  015540  027000                           BIT13+BIT11+BIT10+BIT9  ;MODE, RX ENABL, TX DSABL, SYNC2
2843  015542  013737  001240  022622           MOV     SYNCX,DATA      ;PLACE SYNC 2 IN  DATA
2844  015550  104416                           DATACLK                 ;INIT DATA CLOCK.
2845  015552  004537  022326                   PERFORM .RXSHIFT         ;SHIFT DATA INTO RX
2846  015556  001242                           CLKX                    ;NUMBER OF SHIFTS NEEDED
2847  015560  012777  076400  163612           MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
2848                                                                   ;SET BR "B" AND MATCH DET.
2849  015566  017704  163576                   MOV     @DVLCR,R4       ;SAVE LPR IN R4
2850  015572  010405                           MOV     R4,R5           ;SET FOR COMPARE
2851  015574  052705  000001                   BIS     #BIT0,R5        ;BR "A" FALSE
2852  015600  042705  000002                   BIC     #BIT1,R5        ;BR "B" TRUE
2853  015604  020504                           CMP     R5,R4
2854  015606  001401                           BEQ     .+4             ;BR IF LPR Ok.
2855  015610  104001                           HLT     1               ;EXPECT B TRUE; A FALSE
2856  015612  012777  002000  163560           MOV     #BIT10,@DVSFR   ;SET BR "A" AND RX CHAR FLAG.
```

```
 2857  015620  017704  163544                  MOV     @DVLCR,R4       ;SAVE LPR IN R4
 2858  015624  010405                          MOV     R4,R5           ;SET FOR COMPARE
 2859  015626  032737  010000  001236          BIT     #TWOSYN,STAT    ;SET FOR ONE SYNC OR TWO
 2860  015634  001036                          BNE     4$              ;BR IF SET FOR ONE SYNC
 2861  015636  052705  000003                  BIS     #BIT1+BIT0,R5
 2862  015642  020504                          CMP     R5,R4
 2863  015644  001401                          BEQ     .+4
 2864  015646  104001                          HLT     1
 2865  015650  013737  001240  022622          MOV     SYNCX,DATA
 2866  015656  004537  022326                  PERFORM .RXSHIFT
 2867  015662  001242                          CLKX
 2868  015664  012777  076400  163506          MOV     #BRB+BIT11+BIT10+BIT8,@DVSFR
 2869                                                                  ;SET BR "B" AND MATCH DE .
 2870  015672  017704  163472                  MOV     @DVLCR,R4       ;SAVE LPR IN R4
 2871  015676  010405                          MOV     R4,R5           ;SET FOR COMPARE
 2872  015700  052705  000001                  BIS     #BIT0,R5        ;BR "A" FALSE
 2873  015704  042705  000002                  BIC     #BIT1,R5        ;BR "B" TRUE
 2874  015710  020504                          CMP     R5,R4
 2875  015712  001401                          BEQ     .+4             ;BR IF LPR OK.
 2876  015714  104001                          HLT     1               ;EXPECT B TRUE; A FALSE
 2877  015716  012777  002000  163454          MOV     #BIT10,@DVSFR   ;SET BR "A" AND RX CHAR FLAG.
 2878  015724  017704  163440                  MOV     @DVLCR,R4       ;SAVE LPR IN R4
 2879  015730  010405                          MOV     R4,R5           ;SET FOR COMPARE
 2880  015732  052705  000002          4$:     BIS     #BIT1,R5
 2881  015736  042705  000001                  BIC     #BIT0,R5
 2882  015742  020504                          CMP     R5,R4
 2883  015744  001401                          BEQ     .+4
 2884  015746  104001                          HLT     1
 2885  015750  005237  015526                  INC     6S$             ;UPDATE LINE NUMBER
 2886  015754  005303                          DEC     R3
 2887  015756  001257                          BNE     1$
 2888  015760  000207                          RTS     PC
 2889
 2890
 2891                                          ;********************** TEST 20 ****************************
 2892                                          ;*TEST TO FORCE RECEIVER OVERRUN.
 2893                                          ;*THIS TEST WILL PUSH INTO THE RECEIVER
 2894                                          ;*TWO FULL CHARS (SYNCS) AND ONE MORE CHAR MINUS
 2895                                          ;*ONE BIT. THE PROGRAM WILL VERIFY NO OVERRUN EXISTS
 2896                                          ;*THEN THE LAST BITS WILL BE PUSHED IN VERIFING
 2897                                          ;*THAT THE OVERRUN WAS GENERATED.
 2898                                          ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
 2899                                          ;;**************************************************************
 2900
 2901                                  ;  TEST 20
 2902                                  ;-------------
 2903  015762  012737  000020  001226  TST20:  MOV     #20,TSTNO
 2904  015770  012737  016600  001216          MOV     #TST21,NEXT
 2905  015776  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 2906  016002  113737  001412  001242          MOVB    CLK.A,CLKX      ;PLACE "SHIFTS PER CHAR" IN CLKX
 2907  016010  113737  001406  001244          MOVB    MASK.A,MASKX    ;PLACE "MASK"FOR CHARS INTO MASKX
 2908  016016  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2909  016024  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 2910  016026  004737  016160                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 2911  016032  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 2912  016036  113737  001413  001242          MOVB    CLK.B,CLKX      ;PLACE "SHIFTS PER CHAR" IN CLKX
```

```
 2913  016044  113737  001407  001244          MOVB    MASK.B,MASKX    ;GET MASK
 2914  016052  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2915  016060  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 2916  016062  004737  016160                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 2917  016066  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 2918  016072  113737  001414  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
 2919  016100  113737  001410  001244          MOVB    MASK.C,MASKX    ;GET MASK
 2920  016106  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 2921  016114  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 2922  016116  004737  016160                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 2923  016122  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 2924  016126  113737  001415  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
 2925  016134  113737  001411  001244          MOVB    MASK.D,MASKX    ;GET MASKK
 2926  016142  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 2927  016150  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 2928  016152  004737  016160                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 2929  016156  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
 2930  016160                          105$:                           ;TEST ENTRANCE.
 2931  016160  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
 2932  016166  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
 2933  016170  000207                          RTS     PC              ;EXIT TEST
 2934  016172  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINE GROUP
 2935  016176  010037  016212                  MOV     R0,65$          ;SET LINE POINTER
 2936  016202  104412                  1$:     MSTCLR                  ;RESET DV11
 2937  016204  005001                          CLR     R1              ;ZERO MSCANNER POINTER
 2938  016206  004537  022470                  PERFORM .SETSCAN         ;ADJUST MSCANNER
 2939  016212  000001                  65$:    .BLKW 1                 ;LINE POINTER
 2940  016214  010077  163152          3$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
 2941  016220  012777  125000  163142          MOV     #BIT15+BIT13+BIT11+BIT9,@DVLCR
 2942  016226  004737  022406                  JSR     PC,CKBIT15
 2943  016232  113737  001236  022622          MOVB    STAT,DATA       ;GET SYNC CHAR
 2944  016240  104416                          DATACLK                 ;INIT DV11 BY ONE CLOCK
 2945  016242  113737  001242  016576          MOVB    CLKX,10$        ;GET NUMBER OF SHIFTS PER CHAR.
 2946  016250  004537  022326                  PERFORM ,RXSHIFT        ;CLOCK RX
 2947  016254  016576                          10$                     ;NUMBER OF SHIFTS
 2948  016256  113737  001236  022622          MOVB    STAT,DATA       ;GET ANOTHER SYNC
 2949  016264  004537  022326                  PERFORM ,RXSHIFT        ;SHIFT RX
 2950  016270  016576                          10$                     ;NUMBER OF SHIFTS
 2951  016272  113737  001236  022622          MOVB    STAT,DATA       ;SYNC CHAR
 2952  016300  162737  000001  016576          SUB     #1,10$          ;SET NUMBER OF SHIFTS -1
 2953  016306  004537  022326                  PERFORM ,RXSHIFT        ;SHIFT RX
 2954  016312  016576                          10$                     ;SHIFTS
 2955  016314  012777  050023  163056          MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
 2956  016322  104415                          ROMCLK                  ;S/C "SET RECV DATA ENABLE"
 2957  016324  012777  050021  163046          MOV     #S.C+BIT4+BIT0,@DVSFR
 2958  016332  104415                          ROMCLK                  ;SET/CLEAR SILO IN
 2959  016334  012777  001400  163036          MOV     #BIT9+BIT8,@DVSFR
 2960  016342  032777  000001  163020  4$:     BIT     #BIT0,@DVLCR    ;RCV CHAR WAITING??
 2961  016350  001374                          BNE     4$              ;BR IF YES
 2962  016352  012702  030306                  MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
 2963  016356  010277  163016                  MOV     R2,@DVSFR       ;XFR RIC+SILO OUT
 2964  016362  104415                          ROMCLK                  ;DATA/XFER  RICR+SILO OUT
 2965  016364  017704  162776                  MOV     @DVRIC,R4       ;READ DVRIC REG
 2966  016370  010405                          MOV     R4,R5           ;
 2967  016372  042705  020000                  BIC     #BIT13,R5       ;
 2968  016376  020504                          CMP     R5,R4           ;OVERRUN??
```

```
 2969  016400  001401                          BEQ     .+4             ;BR IF NO
 2970  016402  104001                          HLT     1               ;OVERRUN OCCURED TO SOON.
 2971  016404  004537  022456                  PERFORM ,SILO.OUT       ;SILO OUT
 2972  016410  113737  001236  022623          MOVB    STAT,DATA
 2973  016416  113704  001242                  MOVB    CLKX,R4
 2974  016422  005304                          DEC     R4
 2975  016424  000241                  66$:    CLC
 2976  016426  106037  022622                  RORB    DATA
 2977  016432  105304                          DECB    R4
 2978  016434  001373                          BNE     66$
 2979  016436  012737  000001  016576          MOV     #1,10$
 2980  016444  004537  022326                  PERFORM ,RXSHIFT
 2981  016450  016576                          10$
 2982  016452  012777  050021  162720          MOV     #S.C+BIT4+BIT0,@DVSFR
 2983  016460  104415                          ROMCLK                  ;SET/CLEAR SILO IN
 2984  016462  012777  001400  162710          MOV     #BIT9+BIT8,@DVSFR
 2985  016470  032777  000001  162672  5$:     BIT     #BIT0,@DVLCR    ;REC\ CHAR WAITING
 2986  016476  001374                          BNE     5$              ;
 2987  016500  010005                          MOV     R0,R5           ;GET LINE NUMBER
 2988  016502  000305                          SWAB    R5              ;PUT LINE NUMBER INTO HIGH BYTE
 2989  016504  153705  001236                  BISB    STAT,R5         ;PLACE SYNC INTO EXPECTED
 2990  016510  143705  001244                  BICB    MASKX,R5        ;CLEAR UNUSED BITS.
 2991  016514  052705  020000                  BIS     #BIT13,R5       ;SET OVERRUN
 2992  016520  012702  030306                  MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
 2993  016524  010277  162650                  MOV     R2,@DVSFR
 2994  016530  104415                          ROMCLK                  ;DATA/XFER  RICR+SILO OUT
 2995  016532  017704  162630                  MOV     @DVRIC,R4       ;READ DVRIC
 2996  016536  032737  040000  001236          BIT     #PARBIT,STAT    ;PARITY?
 2997  016544  001402                          BEQ     6$              ;BR IF NO
 2998  016546  042704  010000                  BIC     #BIT12,R4       ;CLEAR PARITY ERROR IF IT EXISTS
 2999  016552  020504                  6$:     CMP     R5,R4           ;OVERRUN SET?
 3000  016554  001401                          BEQ     .+4             ;BR IF YES
 3001  016556  104001                          HLT     1               ;LINE,CHAR,AND OVERRUN EXPECTED.
 3002  016560  004537  022456                  PERFORM ,SILO.OUT       ;SILO OUT
 3003  016564  005237  016212                  INC     65$             ;UPDATE LINE POINTER
 3004  016570  005303                          DEC     R3              ;4 LINE GROUP DONE?
 3005  016572  001203                          BNE     1$              ;BR IF NO
 3006  016574  000207                          RTS     PC              ;RETURN FOR NEXT 4 LINE GROUP
 3007  016576  000001                  10$:    .BLKW 1
 3008
 3009
 3010                                          ;************************ TEST 21 *******************************
 3011                                          ;*TEST OF RECEIVER DATA .
 3012                                          ;*THIS TEST RUNS A BINARY COUNT PATTERN THROUGH
 3013                                          ;*THE RECEIVER OF EACH LINE
 3014                                          ;*THROUGH THE USE OF MAINT. DATA BIT.
 3015                                          ;*THE TX IS NEVER ENABLED.
 3016                                          ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
 3017                                          ;;*****************************************************************
 3018
 3019                                  ;  TEST 21
 3020                                  ;--------------
 3021  016600  012737  000021  001226  TST21:  MOV     #21,TSTNO
 3022  016606  012737  017346  001216          MOV     #TST22,NEXT
 3023  016614  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 3024  016620  113737  001412  001242          MOVB    CLK.A,CLKX      ;PLACE "SHIFTS PER CHAR" IN CLKX
```

```
3025  016626  113737  001406  001244          MOVB    MASK.A,MASKX    ;PLACE "MASK"FOR CHARS INTO MASKX
3026  016634  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
3027  016642  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3028  016644  004737  016776                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3029  016650  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3030  016654  113737  001413  001242          MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
3031  016662  113737  001407  001244          MOVB    MASK.B,MASKX    ;GET MASK
3032  016670  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3033  016676  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
3034  016700  004737  016776                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3035  016704  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
3036  016710  113737  001414  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
3037  016716  113737  001410  001244          MOVB    MASK.C,MASKX    ;GET MASK
3038  016724  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
3039  016732  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
3040  016734  004737  016776                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
3041  016740  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
3042  016744  113737  001415  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
3043  016752  113737  001411  001244          MOVB    MASK.D,MASKX    ;GET MASKK
3044  016760  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
3045  016766  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
3046  016770  004737  016776                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
3047  016774  104400          103$:   SCOPE                           ;SCOPE THIS TEST.
3048  016776          105$:                   ;TEST ENTRANCE.
3049  016776  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
3050  017004  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
3051  017006  000207                          RTS     PC              ;EXIT TEST
3052  017010  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINE GROUP.
3053  017014  010037  017030                  MOV     R0,65$          ;PLACE LINE POINTER
3054  017020  104412          1$:     MSTCLR                          ;CLEAR THE DV11
3055  017022  005001                          CLR     R1              ;ZERO MSCANNER POINTER
3056  017024  004537  022470                  PERFORM .SETSCAN         ;SET SCANNER
3057  017030  000001          65$:    .BLKW 1                         ;POSITION MSCAN TO LINE NO.
3058  017032  010077  162334          3$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
3059  017036  012777  125000  162324          MOV     #BIT15+BIT13+BIT11+BIT9,@DVLCR
3060  017044  004737  022406                  JSR     PC,CKBIT15      ;GO WAIT FOR BIT15 TO=0
3061  017050  113737  001236  022622          MOVB    STAT,DATA       ;LOAD SYNC CHAR
3062  017056  104416                          DATACLK                 ;GIVE AN INITIAL CLOCK
3063  017060  004537  022326                  PERFORM .RXSHIFT         ;STROBE CHAR INTO RX.
3064  017064  001242                          CLKX                    ;PICK UP NO. OF CLOCKS.
3065  017066  032737  010000  001236          BIT     #TWOSYN,STAT    ;TWO SYNCS REQUIRED??
3066  017074  001006                          BNE     4$              ;BR IF ONLY ONE SYNC..
3067  017076  113737  001236  022622          MOVB    STAT,DATA       ;GIVE ANOTHER SYNC TO THE RX
3068  017104  004537  022326                  PERFORM .RXSHIFT         ;STROBE IT IN
3069  017110  001242                          CLKX                    ;SHIFTS REQUIRED
3070  017112  010005          4$:     MOV     R0,R5                   ;LOAD LINE NUMBER INTO "EXPECTED"
3071  017114  000305                          SWAB    R5              ;PLACE IT INTO HIGH BYTE
3072  017116  105005                          CLRB    R5              ;ZERO LOW BYTE
3073  017120  012737  017174  001220          MOV     #5$,LOCK        ;SET IF SW09=1; GOTO 5$
3074  017126  012777  050023  162244          MOV     #5.C+BIT4+BIT1+BIT0,@DVSFR
3075  017134  104415                          ROMCLK                  ;CLOCK "DATA ENABLE"
3076  017136  004537  022434                  PERFORM .SILO.IN         ;READ RX BUFFER INTO SILO
3077  017142  005002                          CLR     R2              ;SET FOR DELAY
3078  017144  012777  001400  162226          MOV     #BIT9+BIT8,@DVSFR
3079  017152  032777  000001  162210  10$:    BIT     #BIT0,@DVLCR    ;IS "RX CHAR WAITING" TRUE?
3080  017160  001403                          BEQ     9$              ;BR IF TRUE..
```

```
 3081  017162  005202                          INC     R2              ;DELAY.....
 3082  017164  001372                          BNE     10$             ;BR IF DELAY NOTDONE
 3083  017166  104000                          HLT     0               ;RX CHAR WAITING NOT TRUE!
 3084  017170  004537  022456          9$:     PERFORM ,SILO.OUT       ;REMOVE CHAR FROM SILO
 3085  017174  010537  022622          5$:     MOV     R5,DATA         ;PLACE CHAR INTO SOFTWARE LOC.
 3086  017200  105037  022623                  CLRB    DATA+1          ;ZERO LINE NUMBER.
 3087  017204  004537  022326                  PERFORM ,RXSHIFT        ;PLACE CHAR INTO RX BUFFER.
 3088  017210  001242                          CLKX                    ;CLOCKS.
 3089  017212  012777  050023  162160          MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
 3090  017220  104415                          ROMCLK                  ;SET RX DATA ENABLE
 3091  017222  004537  022434                  PERFORM ,SILO.IN        ;READ FROM RX BUFFER INTO SILO
 3092  017226  005002                          CLR     R2              ;SET DELAY
 3093  017230  012777  001400  162142          MOV     #BIT9+BIT8,@DVSFR
 3094  017236  032777  000001  162124  6$:     BIT     #BIT0,@DVLCR    ;WAIT FOR RX CHAR WAITING
 3095  017244  001403                          BEQ     7$              ;BR IF TRUE
 3096  017246  005202                          INC     R2              ;UPDATE DELAY
 3097  017250  001372                          BNE     6$              ;GOBACK
 3098  017252  104000                          HLT     0               ;RX CHAR WAITING NOT TRUE
 3099  017254  012702  030306          7$:     MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
 3100  017260  010277  162114                  MOV     R2,@DVSFR       ;DO DATA XFER FROM SILO TO DVRIC
 3101  017264  104415                          ROMCLK                  ;CLOCK
 3102  017266  017704  162074                  MOV     @DVRIC,R4       ;LOAD DVRIC TO "FOUND" LOC.
 3103  017272  032737  040000  001236          BIT     #PARBIT,STAT    ;PARITY ON??
 3104  017300  001402                          BEQ     16$             ;BR IF PARITY NOT ON.
 3105  017302  042704  010000                  BIC     #BIT12,R4       ;CLEAR PARITY ERROR (DON'T WORRY ABOUT PARITY NOW!)
 3106  017306                          16$:
 3107  017306  020504                          CMP     R5,R4           ;RX DATA AND LINE NUMBER OK??
 3108  017310  001401                          BEQ     .+4             ;BR IF EXPECTED =FOUND.
 3109  017312  104002                          HLT     2               ;RX DATA ERROR
 3110  017314  004537  022456                  PERFORM ,SILO.OUT       ;REMOVE RX DATA FROM SILO
 3111  017320  104401                          SCOP1                   ;SW09=1?
 3112  017322  105205                          INCB    R5              ;UPDATE DATA
 3113  017324  001403                          BEQ     8$              ;BR IF ALL DATA DONE
 3114  017326  133705  001244                  BITB    MASKX,R5        ;IF <8BITS CHECK END OF DATA.
 3115  017332  001720                          BEQ     5$              ;BR IF MORE TO GO
 3116  017334  005237  017C30          8$:     INC     65$             ;UPDATE TO NEXT LINE.
 3117  017340  005303                          DEC     R3              ;ALL 4 LINES DONE?
 3118  017342  001226                          BNE     1$              ;BR IF NOT ALL DONE
 3119  017344  000207                          RTS     PC              ;SCOPE THIS TEST
 3120
 3121
 3122                                          ;************************ TEST 22 *******************************
 3123                                          ;*TEST OF RECEIVER DATA .
 3124                                          ;*THIS TEST RUNS A SET PATTERN TRHOUGH
 3125                                          ;*THE RECEIVER OF EACH LINE
 3126                                          ;*THROUGH THE USE OF THE TRANSMITTER.
 3127                                          ;*THIS TEST EXERCISES ALL LINES IN GROUPS OF 4.
 3128                                          ;*NOTE: SHOULD A DATA COMPARE ERROR OCCUR; THE PROGRAM
 3129                                          ;*      REPORTS THE ERROR AS A RECEIVER DATA ERROR BASED
 3130                                          ;*      ON THE TRANSMITTER HAS PREVIOUSLY BEEN CHECKED AND ASSUMED GOOD.
 3131                                          ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
 3132                                          ;;*************************************************************
 3133
 3134                                  ;  TEST 22
 3135                                  ;--------------
 3136  017346  012737  000022  001226  TST22:  MOV     #22,TSTNO
```

```
 3137  017354  012737  020336  001216          MOV     #TST23,NEXT
 3138  017362  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 3139  017366  113737  001412  001242          MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
 3140  017374  113737  001406  001244          MOVB    MASK.A,MASKX    ;PLACE "MASK"FOR CHARS INTO MASKX
 3141  017402  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3142  017410  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 3143  017412  004737  017544                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 3144  017416  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 3145  017422  113737  001413  001242          MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
 3146  017430  113737  001407  001244          MOVB    MASK.B,MASKX    ;GET MASK
 3147  017436  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3148  017444  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 3149  017446  004737  017544                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 3150  017452  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 3151  017456  113737  001414  001242          MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
 3152  017464  113737  001410  001244          MOVB    MASK.C,MASKX    ;GET MASK
 3153  017472  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3154  017500  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 3155  017502  004737  017544                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 3156  017506  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 3157  017512  113737  001415  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
 3158  017520  113737  001411  001244          MOVB    MASK.D,MASKX    ;GET MASKK
 3159  017526  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 3160  017534  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 3161  017536  004737  017544                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 3162  017542  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
 3163  017544                          105$:                           ;TEST ENTRANCE.
 3164  017544  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
 3165  017552  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
 3166  017554  000207                          RTS     PC              ;EXIT TEST
 3167  017556  010037  017654                  MOV     R0,65$          ;PLACE LINE NO.
 3168  017562  005037  001250                  CLR     TEMP2
 3169  017566  113704  001244                  MOVB    MASKX,R4
 3170  017572  005037  001252                  CLR     TEMP3
 3171  017576  110437  001252                  MOVB    R4,TEMP3
 3172  017602  000241                          CLC
 3173  017604  006104                          ROL     R4
 3174  017606  050437  001252                  BIS     R4,TEMP3
 3175  017612  000241                          CLC
 3176  017614  006104                          ROL     R4
 3177  017616  050437  001252                  BIS     R4,TEMP3
 3178  017622  013737  001236  022572          MOV     STAT,SYNC
 3179  017630  113737  001236  022573          MOVB    STAT,SYNC+1
 3180  017636  012737  000004  001246          MOV     #4,TEMP1        ;SET FOR 4 LINES
 3181  017644  104412                  1$:     MSTCLR                  ;RESET DV11
 3182  017646  005001                          CLR     R1              ;ZERO MSCANNER POINTER
 3183  017650  004537  022470                  PERFORM ,SETSCAN         ;ADJUST SCANNER FOR PROPER LINE
 3184  017654  000001                  65$:    .BLKW 1                 ;
 3185  017656                          3$:
 3186                                                                  ;SET SOURCE SELECT
 3187  017656  010077  161510          7$:     MOV     R0,@DVSRS       ;LOAD LINE NUMBER
 3188  017662  004537  022560                  PERFORM ,CLR.TMARK       ;CLEAR TMARK BIT.
 3189  017666  004537  022266                  PERFORM ,LOAD.MODE       ;LOAD
 3190  017672  024000                          BIT13+BIT11             ;MODE AND RX ENABLE
 3191  017674  032737  010000  001236          BIT     #TWOSYN,STAT
 3192  017702  001003                          BNE     9$
```

```
3193  017704  012703  022572              MOV     #SYNC,R3
3194  017710  000402                      BR      10$
3195  017712  012703  022573      9$:     MOV     #SYNC+1,R3
3196  017716  111337  001250      10$:    MOVB    (R3),TEMP2
3197  017722  043737  001252  001250      BIC     TEMP3,TEMP2
3198  017730  005077  161436              CLR     @DVSRS          ;ZERO LINE TO LINE 0
3199  017734  013777  001250  161434      MOV     TEMP2,@DVSRA    ;LOAD DATA INTO DVSRA
3200  017742  012777  020000  161430      MOV     #BIT13,@DVSFR   ;EXECUTE A "ROM READ" INTSTR
3201  017750  104415                      ROMCLK                  ;CLOCK.
3202  017752  012777  030260  161420      MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
3203  017760  104415                      ROMCLK                  ;DO A DATA XFER FROM RAM OUTPUT TO TX BUFFER
3204  017762  104416                      DATACLK                 ;ISSUE A MAINT CLK.
3205  017764  012737  017776  001220      MOV     #4$,LOCK        ;SET IF SW09=1 GOTO 4$
3206  017772  010005                      MOV     R0,R5
3207  017774  000305                      SWAB    R5
3208  017776  113702  001242      4$:     MOVB    CLKX,R2         ;SET REQUIRED SHIFTS
3209  020002  010077  161364              MOV     R0,@DVSRS       ;LOAD LINE NUMBER
3210  020006  111337  001250              MOVB    (R3),TEMP2
3211  020012  043737  001252  001250      BIC     TEMP3,TEMP2
3212  020020  105005                      CLRB    R5
3213  020022  053705  001250              BIS     TEMP2,R5
3214  020026  104416              5$:     DATACLK                 ;ISSUE MAINT CLK
3215  020030  005302                      DEC     R2              ;ALL SHIFTS DONE?
3216  020032  022702  000001              CMP     #1,R2           ;IS THE BUFFER ALMOST EMPTY?
3217  020036  001033                      BNE     8$              ;BR IF NO
3218  020040  005077  161326              CLR     @DVSRS          ;ZERO LINE NUMBER
3219  020044  032777  001000  161130      BIT     #BIT9,@SWR      ;LOCK ON DATA?
3220  020052  001001                      BNE     .+4             ;BR IF YES!!
3221  020054  005203                      INC     R3              ;UPDATE DATA POINTER.
3222  020056  111337  001250              MOVB    (R3),TEMP2      ;STORE DATA
3223  020062  013777  001250  161306      MOV     TEMP2,@DVSRA    ;LOAD DATA INTO DVSRA
3224  020070  012777  020000  161302      MOV     #BIT13,@DVSFR   ;DO A ROM READ
3225  020076  104415                      ROMCLK                  ;CLK
3226  020100  012777  030260  161272      MOV     #XFR+BIT7+BIT5+BIT4,@DVSFR
3227  020106  104415                      ROMCLK                  ;DO A  DATA XFER TO TX BUFF
3228  020110  010077  161256              MOV     R0,@DVSRS       ;RESELECT LINE NUMBER
3229  020114  032777  001000  161060      BIT     #BIT9,@SWR      ;LOCK ON DATA?
3230  020122  001001                      BNE     .+4             ;BR IF YES!!
3231  020124  005303                      DEC     R3              ;READJUST DATA CHAR POINTER.
3232  020126  005702              8$:     TST     R2              ;ALL SHIFTS DONE?
3233  020130  001336                      BNE     5$              ;BR IF NO
3234  020132  022703  022572              CMP     #SYNC,R3
3235  020136  001465                      BEQ     50$
3236  020140  022703  022573              CMP     #SYNC+1,R3
3237  020144  001462                      BEQ     50$
3238  020146  012777  050023  161224      MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
3239  020154  104415                      ROMCLK                  ;SET RX DATA ENABLE
3240  020156  004537  022434              PERFORM ,SILO.IN        ;READ FROM RX BUFFER INTO SILO
3241  020162  005002                      CLR     R2              ;SET DELAY
3242  020164  012777  001400  161206      MOV     #BIT9+BIT8,@DVSFR
3243  020172  032777  000001  161170  26$:BIT     #BIT0,@DVLCR    ;WAIT FOR RX CHAR WAITING
3244  020200  001403                      BEQ     27$             ;BR IF TRUE
3245  020202  005202                      INC     R2              ;UPDATE DELAY
3246  020204  001372                      BNE     26$             ;GOBACK
3247  020206  104000                      HLT     0               ;RX CHAR WAITING NOT TRUE
3248  020210  012702  030306      27$:    MOV     #XFR+BIT7+BIT6+BIT2+BIT1,R2
```

```
3249  020214  010277  161160                    MOV     R2,@DVSFR       ;DO DATA XFER FROM SILO TO DVRIC
3250  020220  104415                            ROMCLK                  ;CLOCK
3251  020222  017704  161140                    MOV     @DVRIC,R4       ;LOAD DVRIC TO "FOUND" LOC.
3252  020226  032737  040000  001236            BIT     #PARBIT,STAT    ;PARITY ON??
3253  020234  001402                            BEQ     36$             ;BR IF PARITY NOT ON.
3254  020236  042704  010000                    BIC     #BIT12,R4       ;CLEAR PARITY ERROR (DON'T WORRY ABOUT PARITY NOW!)
3255  020242  020504                  36$:      CMP     R5,R4           ;RX DATA AND LINE NUMBER OK??
3256  020244  001401                            BEQ     .+4             ;BR IF EXPECTED =FOUND.
3257  020246  104002                            HLT     2               ;RX DATA ERROR
3259  020250  004537  022456                    PERFORM ,SILO.OUT       ;REMOVE RX DATA FROM SILO
3259  020254  104401                            SCOP1                   ;LOCK ON DATA?
3260  020256  005203                  11$:      INC     R3
3261  020260  020327  022620                    CMP     R3,#ENDPAT
3262  020264  001244                            BNE     4$
3263  020266  004537  022546          6$:       PERFORM ,SET.TMARK      ;SET TMARK BIT.
3264  020272  005237  017654                    INC     65$             ;UPDATE LINE NO.
3265  020276  005337  001246                    DEC     TEMP1           ;ALL LINES(4) DONE?
3266  020302  001402                            BEQ     46$
3267  020304  000137  017644                    JMP     1$
3268  020310  000207                  46$:      RTS     PC                      ;SCOPE THESE 4 LINES!
3269  020312  012777  050023  161060  50$:      MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
3270  020320  104415                            ROMCLK
3271  020322  012777  050022  161050            MOV     #S.C+BIT4+BIT1,@DVSFR
3272  020330  104415                            ROMCLK
3273  020332  000137  020256                    JMP     11$
3274
3275
3276                                            ;********************** TEST 23 ****************************
3277                                            ;*TEST OF RECEIVER "RE-SYNC"
3278                                            ;*THIS TEST WILL SEND (BY BIT WINDOW) TWO SYNC CHARS AND
3279                                            ;*THEN VERIFY THAT RX CHAR FLAG IS TRUE.
3280                                            ;*THEN A "RE-SYNC" WILL BE ISSUED AND
3281                                            ;*TWO NON-SYNC CHARS WIIL BE SENT INTO THE RX
3282                                            ;*VERIFYING THAT THERE IS NO RX CHAR FLAG.
3283                                            ;*NEXT TWO SYNC CHARS ARE AGAIN MOVED INTO THE RX
3284                                            ;*VERIFING CHAR FLAG AND THE THE RX SOULD INDEED
3285                                            ;*          RE SYNC!
3286                                            ;*THIS TEST WILL BE DONE FOR SYNC LINE CARDS ONLY.
3287                                            ;:**********************************************************
3288
3289                                    ;  TEST 23
3290                                    ;--------------
3291  020336  012737  000023  001226    TST23:  MOV     #23,TSTNO
3292  020344  012737  021042  001216            MOV     #TST24,NEXT
3293  020352  012700  000000                    MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
3294  020356  113737  001412  001242            MOVB    CLK.A,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
3295  020364  013737  001416  001236            MOV     L00.0  STAT     ;LOAD LINE CARD STATUS INTO STAT
3296  020372  100402                            BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
3297  020374  004737  020504                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
3298  020400  012700  000004          100$:     MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
3299  020404  113737  001413  001242            MOVB    CLK.B,CLKX      ;PLACE "SHIFTS/PER/CHAR" IN CLKX
3300  020412  013737  001420  001236            MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
3301  020420  100402                            BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
3302  020422  004737  020504                    JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
3303  020426  012700  000010          101$:     MOV     #8.,R0          ;LOAD LINE NUMBER
3304  020432  113737  001414  001242            MOVB    CLK.C,CLKX      ;GET SHIFTS PER CHAR
```

```
 3305  020440  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3306  020446  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 3307  020450  004737  020504                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 3308  020454  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 3309  020460  113737  001415  001242          MOVB    CLK.D,CLKX      ;GET SHIFTS
 3310  020466  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 3311  020474  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 3312  020476  004737  020504                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 3313  020502  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
 3314  020504                          105$:                           ;TEST ENTRANCE.
 3315  020504  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS A SYNC LINE CARD?
 3316  020512  001401                          BEQ     .+4             ;BR IF SYNC LINE CARD.
 3317  020514  000207                          RTS     PC              ;EXIT TEST
 3318  020516  012703  000004                  MOV     #4,R3           ;SET FOR 4 LINE GROUP
 3319  020522  010037  020536                  MOV     R0,68$          ;SAVE LINE NO
 3320  020526  104412                  1$:     MSTCLR                  ;RESET
 3321  020530  005001                          CLR     R1              ;ZERO MSCANNER POINTER
 3322  020532  004537  022470                  PERFORM ,SETSCAN         ;SET SCANNER
 3323  020536  000001                  68$:    .BLKW 1                 ;TO RIGHT LINE
 3324  020540  012737  020546  001220          MOV     #3$,LOCK        ;SET IF SW09=1
 3325  020546  010077  160620          3$:     MOV     R0,@DVSRS       ;LOAD LINE
 3326  020552  004537  022266                  PERFORM ,LOAD.MODE       ;LOAD
 3327  020556  025000                          BIT13+BIT11+BIT9        ;MODE
 3328  020560  012702  000002                  MOV     #2,R2           ;SET COUNT
 3329  020564  104416                          DATACLK                 ;INIT DV11 SAT/SAR
 3330  020566  013737  001236  022622  4$:     MOV     STAT,DATA       ;GET SYNC
 3331  020574  004537  022326                  PERFORM ,RXSHIFT         ;SHIFT INTO RX
 3332  020600  001242                          CLKX                    ;CLOCKS
 3333  020602  005302                          DEC     R2              ;TWO CHARS YET
 3334  020604  001370                          BNE     4$              ;
 3335  020606  012702  002000                  MOV     #BIT10,R2       ;BRA TEST
 3336  020612  010277  160562                  MOV     R2,@DVSFR       ;
 3337  020616  017704  160546                  MOV     @DVLCR,R4       ;
 3338  020622  010405                          MOV     R4,R5           ;
 3339  020624  042705  000001                  BIC     #BIT0,R5        ;
 3340  020630  020504                          CMP     R5,R4           ;BRANCH TEST POINT BAD
 3341  020632  001401                          BEQ     64$             ;
 3342  020634  104001                          HLT     1               ;
 3343  020636  012777  050106  160534  64$:     MOV     #S.C+BIT6+BIT2+BIT1,@DVSFR
 3344  020644  104415                          ROMCLK                  ;S/C "RESYNC PULSE"
 3345  020646  010277  160526                  MOV     R2,@DVSFR       ;
 3346  020652  017704  160512                  MOV     @DVLCR,R4       ;
 3347  020656  010405                          MOV     R4,R5           ;
 3348  020660  052705  000001                  BIS     #BIT0,R5        ;
 3349  020664  020504                          CMP     R5,R4           ;
 3350  020666  001401                          BEQ     65$             ;
 3351  020670  104001                          HLT     1               ;RESYNC FAILED.
 3352  020672  012702  000002          65$:    MOV     #2,R2           ;
 3353  020676  013737  001236  022622  5$:     MOV     STAT,DATA       ;GET SYNC
 3354  020704  005437  022622                  NEG     DATA            ;MAKE IT A NON-SYNC
 3355  020710  004537  022326                  PERFORM ,RXSHIFT         ;SHIFT
 3356  020714  001242                          CLKX                    ;INTO RX
 3357  020716  005302                          DEC     R2              ;TWO DONE?
 3358  020720  001366                          BNE     5$              ;
 3359  020722  012702  002000                  MOV     #BIT10,R2       ;
 3360  020726  010277  160446                  MOV     R2,@DVSFR       ;
```

```
                                                MOV     @DVLCR,R4       ;
                                                MOV     R4,R5           ;
                                                BIS     #BIT6,R5        ;
                                                CMP     R5,R4           ;
                                                BEQ     66$             ;
                                                HLT     1               ;
                                        66$:    MOV     #2,R2           ;
                                        6$:     MOV     STAT,DATA       ;
                                                PERFORM ,RXSHIFT        ;
                                                CLKX                    ;
                                                DEC     R2              ;
                                                BNE     6$              ;
                                                MOV     #BIT10,R2       ;
                                                MOV     R2,@DVSFR       ;
                                                MOV     @DVLCR,R4       ;
                                                MOV     R4,R5   :       ;
                                                BIC     #BIT0,R5        ;
                                                CMP     R5,R4           ;
                                                BEQ     67$             ;
                                                HLT     1               ;
                                        67$:    SCOPI                   ;
                                                INC     68$             ;
                                                DEC     R2              ;
                                                BNE     1$              ;
                                                RTS     PC              ;EXIT


                                ;************************ TEST 24 ******************************
                                ;*TEST TO VERIFY THAT SETTING RECEIVER ENABLE
                                ;*WILL SET RX FLAG AND MATCH DETECT.
                                ;*TEST WILL ALSO VERIFY THAT CLEARING RECEIVER
                                ;*ENABLE WILL CLEAR RX FLAG AND MATCH DETECT.
                                ;*THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.

                                ;;**************************************************************

                                ;  TEST 24
                                ;--------------
021042  012737  000024  001226  TST24:  MOV     #24,TSTNO
021050  012737  021432  001216          MOV     #TST25,NEXT
021056  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
021062  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
021070  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
021072  004737  021160                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
021076  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
021102  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
021110  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
021112  004737  021160                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
021116  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
021122  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
021130  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
021132  004737  021160                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
021136  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
021142  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
021150  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
021152  004737  021160                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
```

```
3417  021156  104400                  103$:   SCOPE                   ;SCOPE THIS TEST.
3418  021160                          105$:                           ;TEST ENTRANCE.
3419  021160  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS AN ASYNC LINE CARD
3420  021166  001001                          BNE     .+4             ;BR IF ASYNC.
3421  021170  000207                          RTS     PC              ;EXIT TEST
3422  021172  012703  000004                  MOV     #4,R3           ;SET TO TEST 4 LINES.
3423  021176  104412                  1$:     MSTCLR                  ;INIT DV11
3424  021200  005001                          CLR     R1              ;INIT SCANNER POINTER.
3425  021202  012777  000010  160152          MOV     #BIT3,@DVSCR    ;SET SOURCE ENABLE
3426  021210  010037  021220                  MOV     R0,65$          ;PREPARE MASTER SCANNER.
3427  021214  004537  022470                  PERFORM ,SETSCAN        ;SET SCANNER
3428  021220  000001                  65$:    .BLKW   1               ;POSITION OF SCANNER.
3429  021222  010077  160144                  MOV     R0,@DVSRS       ;LOAD LINE NO.
3430  021226  004537  022266                  PERFORM ,LOAD.MODE      ;SET RX ENABLE.
3431  021232  020000                          BIT13                   ;
3432  021234  012702  076400                  MOV     #BRB+BIT11+BIT10+BIT8,R2
3433  021240  010277  160134                  MOV     R2,@DVSFR       ;BRB MATCH DETECT.
3434  021244  017704  160120                  MOV     @DVLCR,R4       ;READ BR POINTS.
3435  021250  010405                          MOV     R4,R5           ;
3436  021252  052705  000001                  BIS     #BIT0,R5        ;BR A FALSE.
3437  021256  042705  000002                  BIC     #BIT1,R5        ;BR B TRUE.
3438  021262  020504                          CMP     R5,R4           ;MATCH DETECT TRUE?
3439  021264  001401                          BEQ     2$              ;BR IF YES
3440  021266  104001                          HLT     1               ;RX FLAG NOT TRUE.
3441  021270  012702  002000          2$:     MOV     #BIT10,R2       ;BRA RX FLAG.
3442  021274  010277  160100                  MOV     R2,@DVSFR       ;LOAD INSTRUCTION.
3443  021300  017704  160064                  MOV     @DVLCR,R4       ;READ BR POINTS.
3444  021304  010405                          MOV     R4,R5           ;
3445  021306  052705  000002                  BIS     #BIT1,R5        ;BR B FALSE
3446  021312  042705  000001                  BIC     #BIT0,R5        ;BR A TRUE.
3447  021316  020504                          CMP     R5,R4           ;RX FLAG TRUE?
3448  021320  001401                          BEQ     3$              ;BR IF YES
3449  021322  104001                          HLT     1               ;RX FLAG NOT TRUE.
3450  021324  004537  022266          3$:     PERFORM ,LOAD.MODE      ;CLEAR RX ENABLE.
3451  021330  000000                          0                       ;
3452  021332  012702  076400                  MOV     #BRB+BIT11+BIT10+BIT8,R2
3453  021336  010277  160036                  MOV     R2,@DVSFR       ;BRB MATCH DETECT.
3454  021342  017704  160022                  MOV     @DVLCR,R4       ;READ BR POINTS.
3455  021346  010405                          MOV     R4,R5           ;
3456  021350  052705  000001                  BIS     #BIT0,R5        ;BR A FALSE.
3457  021354  052705  000002                  BIS     #BIT1,R5        ;BR B FALSE.
3458  021360  020504                          CMP     R5,R4           ;MATCH DETECT FALSE?
3459  021362  001401                          BEQ     4$              ;BR IF YES
3460  021364  104001                          HLT     !               ;RX FLAG NOT FALSE.
3461  021366  012702  002000          4$:     MOV     #BIT10,R2       ;BRA RX FLAG.
3462  021372  010277  160002                  MOV     R2,@DVSFR       ;LOAD INSTRUCTION.
3463  021376  017704  157766                  MOV     @DVLCR,R4       ;READ BR POINTS.
3464  021402  010405                          MOV     R4,R5           ;
3465  021404  052705  000002                  BIS     #BIT1,R5        ;BR B FALSE
3466  021410  052705  000001                  BIS     #BIT0,R5        ;BR A FALSE.
3467  021414  020504                          CMP     R5,R4           ;RX FLAG FALSE?
3468  021416  001401                          BEQ     5$              ;BR IF YES
3469  021420  104001                          HLT     1               ;RX FLAG NOT FALSE.
3470  021422  005200                  5$:     INC     R0              ;UPDATE LINE NO.
3471  021424  005303                          DEC     R3              ;4 LINES DONE?
3472  021426  001263                          BNE     1$              ;BR IF NO.
```

```
 3473  021430  000207                          RTS     PC              ;EXIT TEST.
                                                ;************************ TEST 25 ******************************
                                                ;*TEST TO SET RECEIVER ENABLE.
                                                ;*SET "RX DATA ENABLE".
                                                ;*CLR "RX DATA ENABLE".
                                                ;*AND EXPECT BOTH RX FLAG AND MATCH DETECT TO BE FALSE.
                                                ;*THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.
                                                ;;**************************************************************
                                        ;  TEST 25
                                        ;-------------
 3486  021432  012737  000025  001226  TST25:  MOV     #25,TSTNO
 3487  021440  012737  021744  001216          MOV     #TST26,NEXT
 3488  021446  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 3489  021452  013737  001416  001236          MOV     L00.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3490  021460  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 3491  021462  004737  021550                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 3492  021466  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 3493  021472  013737  001420  001236          MOV     L04.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3494  021500  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 3495  021502  004737  021550                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 3496  021506  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 3497  021512  013737  001422  001236          MOV     L08.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3498  021520  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 3499  021522  004737  021550                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 3500  021526  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 3501  021532  013737  001424  001236          MOV     L12.15,STAT     ;LOAD LINE CARD STATUS
 3502  021540  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 3503  021542  004737  021550                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 3504  021546  104400          103$:           SCOPE                   ;SCOPE THIS TEST.
 3505  021550                  105$:                                   ;TEST ENTRANCE.
 3506  021550  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS AN ASYNC LINE CARD
 3507  021556  001001                          BNE     .+4             ;BR IF ASYNC.
 3508  021560  000207                          RTS     PC              ;EXIT TEST
 3509  021562  012703  000004                  MOV     #4,R3           ;SET TO TEST 4 LINES.
 3510  021566  104412          1$:             MSTCLR                  ;INIT DV11
 3511  021570  005001                          CLR     R1              ;INIT SCANNER POINTER.
 3512  021572  012777  000010  157562          MOV     #BIT3,@DVSCR    ;SET SOURCE ENABLE
 3513  021600  010037  021610                  MOV     R0,65$          ;PREPARE MASTER SCANNER.
 3514  021604  004537  022470                  PERFORM ,SETSCAN         ;SET SCANNER
 3515  021610  000001          65$:            .BLKW   1               ;POSITION OF SCANNER.
 3516  021612  010077  157554                  MOV     R0,@DVSRS       ;LOAD LINE NO.
 3517  021616  004537  022266                  PERFORM .LOAD.MODE       ;SET RX ENABLE.
 3518  021622  020000                          BIT13                   ;
 3519  021624  012777  050023  157546          MOV     #S.C+BIT4+BIT1+BIT0,@DVSFR
 3520  021632  104415                          ROMCLK                  ;SET RX DATA ENABLE.
 3521  021634  012777  050022  157536          MOV     #S.C+BIT4+BIT1,@DVSFR
 3522  021642  104415                          ROMCLK                  ;CLEAR RX DATA ENABLE.
 3523  021644  012702  076400                  MOV     #BRB+BIT11+BIT10+BIT8,R2
 3524  021650  010277  157524                  MOV     R2,@DVSFR       ;BRB MATCH DETECT.
 3525  021654  017704  157510                  MOV     @DVLCR,R4       ;READ BR POINTS.
 3526  021660  010405                          MOV     R4,R5           ;
 3527  021662  052705  000001                  BIS     #BIT0,R5        ;BR A FALSE.
 3528  021666  052705  000002                  BIS     #BIT1,R5        ;BR B FALSE.
```

```
 3529  021672  020504                          CMP     R5,R4           ;MATCH DETECT FALSE?
 3530  021674  001401                          BEQ     4$              ;BR IF YES
 3531  021676  104001                          HLT     1               ;RX FLAG NOT FALSE.
 3532  021700  012702  002000          4$:     MOV     #BIT10,R2       ;BRA RX FLAG.
 3533  021704  010277  157470                  MOV     R2,@DVSFP       ;LOAD INSTRUCTION.
 3534  021710  017704  157454                  MOV     @DVLCR,R4       ;READ BR POINTS.
 3535  021714  010405                          MOV     R4,R5           ;
 3536  021716  052705  000002                  BIS     #BIT1,R5        ;BR B FALSE
 3537  021722  052705  000001                  BIS     #BIT0,R5        ;BR A FALSE.
 3538  021726  020504                          CMP     R5,R4           ;RX FLAG FALSE?
 3539  021730  001401                          BEQ     5$              ;BR IF YES
 3540  021732  104001                          HLT     1               ;RX FLAG NOT FALSE.
 3541  021734  005200          5$:             INC     R0              ;UPDATE LINE NO.
 3542  021736  005303                          DEC     R3              ;4 LINES DONE?
 3543  021740  001312                          BNE     1$              ;BR IF NO.
 3544  021742  000207                          RTS     PC              ;EXIT TEST.
 3545
 3546
 3547                                          ;********************** TEST 26 ******************************
 3548                                          ;*TEST TO SET RECEIVER ENABLE.
 3549                                          ;*ISSUE A RESYNC  SIGNAL.
 3550                                          ;*AND EXPECT BOTH RX FLAG AND MATCH DETECT TO BE FALSE.
 3551                                          ;*THIS TEST WILL BE DONE FOR ASYNC LINE CARDS ONLY.
 3552                                          ;;************************************************************
 3553
 3554                          ;  TEST 26
 3555                          ;--------------
 3556  021744  012737  000026  001226  TST26:  MOV     #26,TSTNO
 3557  021752  012737  002436  001216          MOV     #.EOP,NEXT
 3558  021760  012700  000000                  MOV     #0.,R0          ;PLACE LINE NUMBER INTO R0
 3559  021764  013737  001416  001236          MOV     LC0.03,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3560  021772  100402                          BMI     100$            ;BR IF LINE CARD NOT TO BE TESTED
 3561  021774  004737  022062                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 1
 3562  022000  012700  000004          100$:   MOV     #4.,R0          ;PLACE LINE NUMBER INTO R0
 3563  022004  013737  001420  001236          MOV     LC4.07,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3564  022012  100402                          BMI     101$            ;BR IF LINE CARD NOT TO BE TESTED
 3565  022014  004737  022062                  JSR     PC,105$         ;GO DO THE TEST FOR LINE CARD 2
 3566  022020  012700  000010          101$:   MOV     #8.,R0          ;LOAD LINE NUMBER
 3567  022024  013737  001422  001236          MOV     LC8.11,STAT     ;LOAD LINE CARD STATUS INTO STAT
 3568  022032  100402                          BMI     102$            ;BR IF LINE CARD NOT TO BE TESTED
 3569  022034  004737  022062                  JSR     PC,105$         ;DO THE TEST FOR LINE CARD 3
 3570  022040  012700  000014          102$:   MOV     #12.,R0         ;LOAD LINE NO.
 3571  022044  013737  001424  001236          MOV     LC12.15,STAT    ;LOAD LINE CARD STATUS
 3572  022052  100402                          BMI     103$            ;BR IF LINE CARD NOT TO BE TESTED
 3573  022054  004737  022062                  JSR     PC,105$         ;DO THE TESTS FOR LINE CARD 4
 3574  022060  104400          103$:   SCOPE                           ;SCOPE THIS TEST.
 3575  022062                  105$:                                   ;TEST ENTRANCE.
 3576  022062  032737  004000  001236          BIT     #ASYNC,STAT     ;IS THIS AN ASYNC LINE CARD
 3577  022070  001001                          BNE     .+4             ;BR IF ASYNC.
 3578  022072  000207                          RTS     PC              ;EXIT TEST
 3579  022074  012703  000004                  MOV     #4,R3           ;SET TO TEST 4 LINES.
 3580  022100  104412          1$:     MSTCLR                          ;INIT DV11
 3581  022102  005001                          CLR     R1              ;INIT SCANNER POINTER.
 3582  022104  012777  000010  157250          MOV     #BIT3,@DVSCR    ;SET SOURCE ENABLE
 3583  022112  010037  022122                  MOV     R0,65$          ;PREPARE MASTER SCANNER.
 3584  022116  004537  022470                  PERFORM .SETSCAN        ;SET SCANNER
```

```
 3595  022122  000001                    55$:    .BLKW   1               ;POSITION OF SCANNER.
 3596  022124  010077  157242                    MOV     R0,@DVSRS       ;LOAD LINE NO.
 3597  022130  004537  022266                    PERFORM .LOAD.MODE      ;SET RX ENABLE.
 3598  022134  020000                            BIT13                   ;
 3599  022136  012777  050106  157234            MOV     #S.C+BIT6+BIT2+BIT1,@DVSFR
 3600  022144  104415                            ROMCLK                  ;ISSUE RESYNC.
 3601  022146  012702  076400                    MOV     #BRB+BIT11+BIT10+BIT8,R2
 3602  022152  010277  157222                    MOV     R2,@DVSFR       ;BRB MATCH DETECT.
 3603  022156  017704  157206                    MOV     @DVLCR,R4       ;READ BR POINTS.
 3604  022162  010405                            MOV     R4,R5           ;
 3605  022164  052705  000001                    BIS     #BIT0,R5        ;BR A FALSE.
 3606  022170  052705  000002                    BIS     #BIT1,R5        ;BR B FALSE.
 3607  022174  020504                            CMP     R5,R4           ;MATCH DETECT FALSE?
 3608  022176  001401                            BEQ     4$              ;BR IF YES
 3609  022200  104001                            HLT     1               ;RX FLAG NOT FALSE.
 3610  022202  012702  002000            4$:     MOV     #BIT10,R2       ;BRA RX FLAG.
 3611  022206  010277  157166                    MOV     R2,@DVSFR       ;LOAD INSTRUCTION.
 3612  022212  017704  157152                    MOV     @DVLCR,R4       ;READ BR POINTS.
 3613  022216  010405                            MOV     R4,R5           ;
 3614  022220  052705  000002                    BIS     #BIT1,R5        ;BR B FALSE
 3615  022224  052705  000001                    BIS     #BIT0,R5        ;BR A FALSE.
 3616  022230  020504                            CMP     R5,R4           ;RX FLAG FALSE?
 3617  022232  001401                            BEQ     5$              ;BR IF YES
 3618  022234  104001                            HLT     1               ;RX FLAG NOT FALSE.
 3619  022236  005200            5$:     INC     R0              ;UPDATE LINE NO.
 3620  022240  005303                    DEC     R3              ;4 LINES DONE?
 3621  022242  001316                    BNE     1$              ;BR IF NO.
 3622  022244  000207                    RTS     PC              ;EXIT TEST.
 3623
 3624  022246                    TXSHIFT:
 3625  022246  010046                            MOV     R0,-(SP)
 3626  022250  017700  157114                    MOV     @DVLCR,R0
 3627  022254  106100                            ROLB    R0
 3628  022256  106037  022622                    RORB    DATA
 3629  022262  012600                            MOV     (SP)+,R0
 3630  022264  000205                            EXIT
 3631  022266                    LOAD.MODE:
 3632  022266  012577  157076                    MOV     (R5)+,@DVLCR
 3633  022272  052777  100000  157070            BIS     #BIT15,@DVLCR
 3634  022300  010046                            MOV     R0,-(SP)
 3635  022302  005000                            CLR     R0
 3636  022304  005777  157060            1$:     TST     @DVLCR
 3637  022310  100004                            BPL     2$
 3638  022312  104414                            DELAY
 3639  022314  005200                            INC     R0
 3640  022316  001372                            BNE     1$
 3641  022320  104000                            HLT     0                       ;BIT 15 FAILED TO CLEAR
 3642  022322  012600            2$:     MOV     (SP)+,R0
 3643  022324  000205                    EXIT
 3644  022326                    RXSHIFT:
 3645  022326  010046                            MOV     R0,-(SP)
 3646  022330  010246                            MOV     R2,-(SP)
 3647  022332  113502                            MOVB    @(R5)+,R2
 3648  022334  042777  040000  157026    1$:     BIC     #BIT14,@DVLCR
 3649  022342  005000                            CLR     R0
 3650  022344  000241                            CLC
```

```
 3641  022346  006037  022522                    ROR     DATA
 3642  022352  006000                            ROR     R0
 3643  022354  006000                            ROR     R0
 3644  022356  052700  100000                    BIS     #BIT15,R0
 3645  022362  050077  157002                    BIS     R0,@DVLCR
 3646  022366  004737  022406                    JSR     PC,CKBIT15
 3647  022372  104416                            DATACLK
 3648  022374  105302                            DECB    R2
 3649  022376  001356                            BNE     1$
 3650  022400  012602                            MOV     (SP)+,R2
 3651  022402  012600                            MOV     (SP)+,R0
 3652  022404  000205                            EXIT
 3653
 3654  022406                          CKBIT15:
 3655  022406  010046                            MOV     R0,-(SP)
 3656  022410  005000                            CLR     R0
 3657  022412  005777  156752          64$:      TST     @DVLCR
 3658  022416  100004                            BPL     65$
 3659  022420  104414                            DELAY
 3660  022422  005200                            INC     R0
 3661  022424  001372                            BNE     64$
 3662  022426  104000                            HLT     0                       ;BIT 15 FAILED TO CLEAR
 3663  022430  012600                  65$:      MOV     (SP)+,R0
 3664  022432  000207                            RTS     PC
 3665  022434                          SILO.IN:
 3666  022434  012777  050021  156736            MOV     #BIT14+BIT12+BIT4+BIT0,@DVSFR
 3667  022442  104415                            ROMCLK
 3668  022444  012777  050022  156726            MOV     #BIT14+BIT12+BIT4+BIT1,@DVSFR
 3669  022452  104415                            ROMCLK
 3670  022454  000205                            EXIT
 3671
 3672  022456                          SILO.OUT:
 3673  022456  012777  050020  156714            MOV     #BIT14+BIT12+BIT4,@DVSFR
 3674  022464  104415                            ROMCLK
 3675  022466  000205                            EXIT
 3676
 3677
 3678  022470                          SETSCAN:
 3679  022470  010346                            MOV     R3,-(SP)
 3680  022472  052777  000010  156662            BIS     #BIT3,@DVSCR
 3681  022500  012503                            MOV     (R5)+,R3
 3682  022502  001414                            BEQ     2$
 3683  022504  012777  050102  156666  1$:       MOV     #BIT14+BIT12+BIT6+BIT1,@DVSFR
 3684  022512  104415                            ROMCLK
 3685  022514  005201                            INC     R1
 3686  022516  012777  050102  156654            MOV     #BIT14+BIT12+BIT6+BIT1,@DVSFR
 3687  022524  104415                            ROMCLK
 3688  022526  005201                            INC     R1
 3689  022530  005303                            DEC     R3
 3690  022532  001364                            BNE     1$
 3691  022534  012603                  2$:       MOV     (SP)+,R3
 3692  022536  010100                            MOV     R1,R0
 3693  022540  000241                            CLC
 3694  022542  006000                            ROR     R0
 3695  022544  000205                            EXIT
 3696  022546                          SET.TMARK:
```

```
 3697  022546  012777  050105  156624          MOV     #BIT14+BIT12+BIT6+BIT2+BIT0,@DVSFR
 3698  022554  104415                          ROMCLK                  ;SET/CLEAR "SET TMARK"
 3699  022556  000205                          EXIT
 3700  022560                          CLR.TMARK:
 3701  022560  012777  050101  156612          MOV     #BIT14+BIT12+BIT6+BIT0,@DVSFR
 3702  022566  104415                          ROMCLK          ;SET/CLEAR "CLEAR TMARK"
 3703  022570  000205                          EXIT
 3704
 3705  022572  000001                  SYNC:   .BLKW 1
 3706  022574     000  DATPAT: .BYTE   ↑B<00000000>    ;ALL ZERO'S
 3707  022575     377          .BYTE   ↑B<11111111>    ;ALL ONE'S
 3708  022576     125          .BYTE   ↑B<01010101>    ;ALTERNATE ONE'S
 3709  022577     252          .BYTE   ↑B<10101010>    ;ALTERNATE ZERO'S
 3710  022600     001          .BYTE   ↑B<00000001>    ;F
 3711  022601     002          .BYTE   ↑B<00000010>    ; L
 3712  022602     004          .BYTE   ↑B<00000100>    ;  O
 3713  022603     010          .BYTE   ↑B<00001000>    ;   A
 3714  022604     020          .BYTE   ↑B<00010000>    ;    T
 3715  022605     040          .BYTE   ↑B<00100000>    ;     I
 3716  022606     100          .BYTE   ↑B<01000000>    ;      N
 3717  022607     200          .BYTE   ↑B<10000000>    ;       G  ONE!
 3718  022610     177          .BYTE   ↑B<01111111>    ;F
 3719  022611     277          .BYTE   ↑B<10111111>    ; L
 3720  022612     337          .BYTE   ↑B<11011111>    ;  O
 3721  022613     357          .BYTE   ↑B<11101111>    ;   A
 3722  022614     367          .BYTE   ↑B<11110111>    ;    T
 3723  022615     373          .BYTE   ↑B<11111011>    ;     I
 3724  022616     375          .BYTE   ↑B<11111101>    ;      N
 3725  022617     376          .BYTE   ↑B<11111110>    ;       G  ZERO!
 3726  022620                  ENDPAT:
 3727  022620  000000          NPRLOC: 0
 3728  022622  000000          DATA:   0
 3729  022624  046377  047111  020105  EM1:    .ASCIZ  <377>/LINE CARD STATIC TEST/
       022653     377  042522  042503  EM2:    .ASCIZ  <377>/RECEIVER DATA COMAPRISON ERROR/
       022713     377  051124  047101  EM3:    .ASCIZ  <377>/TRANSMITTER DATA COMPARISON ERROR/
       022756  046777  052123  041523  DH1:    .ASCIZ  <377>/MSTSCAN  DVSFR  EXPECTED  FOUND  LINE:B/
                                       .EVEN
                000000                 SKIP=000000
 3730  023030  000005                  DT6:    5
 3731  023032     006     003                  .BYTE   6,3
 3732  023034  001262                          SAVR1
 3733  023036     006     001                  .BYTE   6,1
 3734  023040  001264                          SAVR2
 3735  023042     006     004                  .BYTE   6,4
 3736  023044  001272                          SAVR5
 3737  023046     006     001                  .BYTE   6,1
 3738  023050  001270                          SAVR4
 3739  023052     002     001                  .BYTE   2,1
 3740  023054  001260                          SAVR0
 3741
 3742  023056                  .ERRTAB:
 3743  023056  000000                          0
 3744  023060  000000                          0
 3745  023062  000000                          0
 3746  023064  022624                          EM1
 3747  023066  022756                          DH1     ;HALT 1
```

I06


```
 3748  023070  023030                          DT6
 3749  023072  022653                          EM2
 3750  023074  022756                          DH1     ;HALT 2
 3751  023076  023030                          DT6
 3752  023100  022713                          EM3
 3753  023102  022756                          DH1     ;HALT 3
 3754  023104  023030                          DT6
 3755  023106                          CORMAX:
 3756          000001                  .END
```

```
ADRCNT= 003443          1143*  1179*  1188*
ALU   = 010000           599*
ASYNC = 004000           607*  1891   1933   1985   2039   2091   2167   2302   2408   2575   2655   2726   2831
                        2931   3049   3164   3315   3419   3506   3576
AUTO.S  006622          1653*
BCC   = 060000           604*
BINWRD  003746          1229*  1230   1267*
BIT0  = 000001           597*  1410   1773   1838   2186   2197   2513   2517   2534   2588   2590   2594   2611
                        2674   2746   2756   2767   2776   2851   2861   2872   2881   2955   2957   2960   2982
                        2985   3074   3079   3089   3094   3238   3243   3269   3339   3348   3363   3377   3436
                        3446   3456   3466   3519   3527   3537   3595   3605   3666   3697   3701
BIT1  = 000002           596*  1410   1421   1773   1780   1789   1793   1838   2177   2312   2418   2522   2588
                        2599   2675   2747   2756   2768   2775   2852   2861   2873   2880   2955   2962   2992
                        3074   3089   3099   3238   3248   3269   3271   3343   3437   3445   3457   3465   3519
                        3521   3528   3536   3589   3596   3604   3668   3683   3686
BIT10 = 002000           587*  1410   1774   1839   2424   2671   2742   2751   2763   2772   2842   2847   2856
                        2868   2877   3335   3359   3373   3432   3441   3452   3461   3523   3532   3591   3600
BIT11 = 004000           586*  1410   1839   1938   1990   2044   2099   2182   2318   2424   2666   2671   2737
                        2742   2763   2842   2847   2868   2941   3059   3190   3327   3432   3452   3523   3531
BIT12 = 010000           585*   599    601    603    605   2998   3105   3254   3666   3668   3673   3693   3696
                        3697   3701
BIT13 = 020000           584*   600    601    604    605   2191   2214   2320   2426   2666   2737   2842   2941
                        2967   2991   3059   3190   3200   3224   3327   3431   3518   3588
BIT14 = 040000           583*   602    603    604    605   1050   2107   2108   2114   2115   3638   3666   3669
                        3673   3683   3686   3697   3701
BIT15 = 100000           582*  1995   2049   2108   2116   2941   3059   3623   3644
BIT2  = 000004           595*   977   1410   2522   2599   2962   2992   3099   3248   3343   3589   3697
BIT3  = 000010           594*  1410   3425   3512   3582   3680
BIT4  = 000020           593*  1991   1892   1944   1945   1998   1999   2052   2053   2193   2216   2322   2428
                        2513   2531   2588   2590   2608   2955   2957   2982   3074   3089   3202   3226   3238
                        3269   3271   3519   3521   3666   3668   3673
BIT5  = 000040           592*  1891   1892   1944   1945   1998   1999   2052   2053   2193   2216   2322   2428
                        3202   3226
BIT6  = 000100           591*  1721   1780   1793   2177   2312   2418   2522   2599   2962   2992   3099   3248
                        3343   3589   3683   3686   3697   3701
BIT7  = 000200           590*  1044   1291   1448   1469   1721   1887   1940   1992   2046   2102   2107   2114
                        2193   2216   2253   2322   2428   2522   2599   2962   2992   3099   3202   3226   3248
BIT8  = 000400           589*  1410   1427   1839   2516   2533   2593   2610   2671   2742   2763   2847   2968
                        2959   2984   3078   3093   3242   3432   3452   3523   3591
BIT9  = 001000           588*  1410   1721   1726   1786   2044   2099   2184   2196   2210   2219   2516   2533
                        2593   2610   2666   2737   2842   2941   2959   2984   3059   3078   3093   3219   3229
                        3242   3327
BRB   = 070000           605*  1839   2671   2742   2763   2847   2868   3432   3452   3523   3591
BRW     003014           983   1072*
BRX     003016           984   1073*
CHRCNT  003744          1227*  1231   1247*  1265*  1266
CKBIT1  022406          1996   2050   2109   2117   2942   3060   3646   3654*
CLKX    001242           676*  2142*  2148*  2154*  2160*  2201   2224   2226   2246   2277*  2283*  2289*  2295*
                        2326   2332   2339   2341   2379*  2386*  2393*  2400*  2432   2438   2445   2447   2634*
                        2639*  2644*  2649*  2670   2705*  2710*  2715*  2720*  2741   2762   2806*  2812*  2819*
                        2824*  2846   2867   2906*  2912*  2918*  2924*  2945   2973   3024*  3030*  3036*  3042*
                        3064   3069   3098   3139*  3145*  3151*  3157*  3208   3294*  3299*  3304*  3309*  3332
                        3356   3370
CLK.A   001412           776*  1571   2142   2277   2379   2634   2705   2806   2906   3024   3139   3294
CLK.B   001413           777*  1576   2148   2283   2386   2639   2710   2812   2912   3030   3145   3299
CLK.C   001414           778*  1581   2154   2289   2393   2644   2715   2818   2918   3036   3151   3304
```

DZDVBB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
CLK.D   001415   779* 1586 2160 2295 2400 2649 2720 2824 2924 3042 3157 3309
CLR.TM  022560  2183  2316 2422 3188 3700*
CNVRT = 104411   735* 1002 1004 1006 1008 1328 1330 1386 1443
CONVRT= 104410   733*  942 1344
CORMAX  023106  3755* 3756
CREAM   001306   697*  911* 1518* 1519 1521* 1526 1527* 1528 1531*
CSRMAP  006624   936  1655*
CYCLE   005666   986  1022 1023 1508*
DATA    022622  2202* 2228* 2233 2333* 2343* 2348 2439* 2449* 2454 2667* 2738* 2760* 2843*
                2865* 2943* 2948* 2951* 2972* 2976* 3061* 3067* 3085* 3086* 3330* 3353* 3354*
                3368* 3618* 3641* 3728*
DATABP  004276  1317* 1320 1342 1345*
DATACL= 104416   745* 2195 2204 2248 2252 2324 2327 2335 2430 2433 2441 2668 2739
                2844  2944 3062 3204 3214 3329 3647
DATAHD  004264  1316* 1338 1341*
DATPAT  022574  3706*
DELAY = 104414   741* 3628 3659
DEVADR  003440  1141* 1176 1186*
DH1     022756  3729* 3747 3750 3753
DT6     023030  3730* 3748 3751 3754
DVACTV  001300   691*  958*  959 1508 1513 1697* 1693* 1694* 1698 1717
DVCR00  001500   807*
DVCR01  001524   818*
DVCR02  001550   829*
DVCR03  001574   840*
DVCR04  001620   851*
DVCR05  001644   862*
DVCR06  001670   873*
DVCR07  001714   884*
DVLCR   001370   759* 1427* 1428 1547* 1548* 1549 1776 1788 1841 1886 1890 1938* 1939
                1943  1990* 1991 1995* 1997 2044* 2045 2049* 2051 2100 2108* 2110 2115*
                2116* 2118  2186 2197  2253 2517  2534 2594  2611 2672 2744  2752 2765
                2773  2849  2857 2870  2878 2941* 2960 2985  3059* 3079 3094  3243 3337
                3346  3361  3375 3434  3443 3454  3463 3525  3534  3593 3602  3616 3622*
                3623* 3626  3638* 3645* 3657
DVNSR   001402   764* 1557* 1558* 1559
DVNUM   001301   692*  907  1012 1659* 1680* 1681 1688 1690
DVRIC   001366   758* 1545* 1546* 1547 2525 2602 2965 2995 3102 3251
DVRLVL  001354   753* 1562* 1563* 1564
DVRVEC  001352   752* 1029  1533* 1562
DVSCR   001362   756* 1026  1404* 1417* 1421* 1532* 1543 3425* 3512* 3582* 3690*
DVSCRH  001364   757* 1543* 1544* 1545
DVSFR   001400   763* 1555* 1556* 1557 1775* 1780* 1787* 1793* 1840* 2177* 2184* 2191* 2193*
                2196* 2214* 2216* 2312* 2320* 2322* 2418* 2426* 2428* 2513* 2516* 2522* 2523
                2531* 2533* 2588* 2590* 2593* 2599* 2600  2608* 2610* 2671* 2742* 2751* 2763*
                2772* 2847* 2856* 2868* 2877* 2955* 2957* 2959* 2963* 2982* 2984* 2993* 3074*
                3078* 3089* 3093* 3100* 3200* 3202* 3224* 3226* 3238* 3242* 3249* 3269* 3271*
                3336* 3343* 3345* 3360* 3374* 3433* 3442* 3453* 3462* 3519* 3521* 3524* 3533*
                3589* 3592* 3601* 3666* 3668* 3673* 3683* 3686* 3697* 3701*
DVSRA   001376   762* 1408 1553* 1554* 1555 2190* 2213* 2319* 2425* 3199* 3223*
DVSRS   001372   760* 1407 1549* 1550* 1551 1889* 1942* 1994* 2048* 2097* 2180* 2189* 2203*
                2209* 2218* 2247* 2315* 2334* 2421* 2440* 2664* 2735* 2840* 2940* 3058* 3167*
                3198* 3209* 3218* 3228* 3325* 3429* 3516* 3586*
DVSRSH  001374   761* 1551* 1552* 1553
DVTLVL  001360   755* 1566* 1567*
DVTR00  001502   808*
```

DZDVBB.P11 CROSS REFERENCE TABLE -- USER SYMBOLS

```
DVTR01  001526            819#
DVTR02  001552            830#
DVTR03  001576            841#
DVTR04  001622            852#
DVTR05  001646            863#
DVTR06  001672            874#
DVTR07  001716            885#
DVTVEC  001356            754#   1564*   1565*   1566
DV.END  001740            895#   1519    1528    1657
DV.MAP  001500            697     806#    911     938    1521    1531    1655    1660    1709
DV00.A  001504            809#
DV00.B  001510            811#
DV00.C  001514            813#
DV00.D  001520            815#
DV01.A  001530            820#
DV01.B  001534            822#
DV01.C  001540            824#
DV01.D  001544            826#
DV02.A  001554            831#
DV02.B  001560            833#
DV02.C  001564            835#
DV02.D  001570            837#
DV03.A  001600            842#
DV03.B  001604            844#
DV03.C  001610            846#
DV03.D  001614            848#
DV04.A  001624            853#
DV04.B  001630            855#
DV04.C  001634            857#
DV04.D  001640            859#
DV05.A  001650            864#
DV05.B  001654            866#
DV05.C  001660            868#
DV05.D  001664            870#
DV06.A  001674            875#
DV06.B  001700            877#
DV06.C  001704            879#
DV06.D  001710            881#
DV07.A  001720            886#
DV07.B  001724            888#
DV07.C  001730            890#
DV07.D  001734            892#
EM1     022624           3729#   3746
EM2     022653           3729#   3749
EM3     022713           3729#   3752
ENDPAT  022620           3261    3726#
ERRCNT  001232            668#    913*   1035    1353*
ERRFLG  001311            703#    909*    997*   1064*   1305*   1318    1332*   1387*
ERRMSG  004252           1315*   1333    1336#
ERTABO  004366           1330    1362#
EXIT  = 000205            607#   3620    3633    3652    3670    3675    3695    3699    3703
EXITER  004322           1348    1353#
FIX.00  006516           1572    1577    1582    1587    1621#
HALTS   004302           1301    1347#
HILIM   003436           1140*   1167    1185#
ICOUNT  001222            664#   1062    1067*
```

```
INBUF   005520          1110    1146    1493#
INIFLG  001310           702#    918     933*
INSTER= 104404           725#   1161
INSTR = 104403           723#   1594
INSTR2  003236          1117    1129#
LIGHT   000174           636#    928
LIGHTS  001200           647#    928*    999*
LIMITS  003364          1156    1167#
LOAD.M  022266          2098    2181    2317    2423    2665    2736    2841    3189    3326    3430    3450    3517    3587
                        3621#
LOBITS  003442          1142*   1171    1187#   1188
LOCK    001220           663#   1066*   1080    1082    1324    2200*   2325*   2431*   3073*   3205*   3324*
LOGICA  002560           633    1016#
LOKFLG  001312           704#
LOLIM   003434          1139*   1169    1184#
LPCNT   001224           665#   1061*   1062    1065*
LSTERR  001234           669#    914*    996*   1048*   1302    1304*   1388*
LO0.03  001416           781#   1534*   1569    1750    1815    1864    1916    1968    2022    2074    2144    2279    2382
                        2486    2558    2635    2706    2808    2908    3026    3141    3295    3402    3489    3559
LO4.07  001420           782#   1536*   1574    1754    1819    1868    1920    1972    2026    2078    2150    2285    2389
                        2490    2562    2640    2711    2814    2914    3032    3147    3300    3406    3493    3563
LO8.11  001422           783#   1538*   1579    1758    1823    1872    1924    1976    2030    2082    2156    2291    2396
                        2494    2566    2645    2716    2820    2920    3038    3153    3305    3410    3497    3567
L12.15  001424           784#   1540*   1584    1762    1827    1876    1928    1980    2034    2086    2162    2297    2403
                        2498    2570    2650    2721    2826    2926    3044    3159    3310    3414    3501    3571
MASKX   001244           677#   2143*   2149*   2155*   2161*   2234       1    2278*   2284*   2290*   2296*   2349    2351
                        2380*   2387*   2394*   2401*   2455    2457      *    2913*   2919*   2925*   2990    3025*   3031*
                        3037*   3043*   3114    3140*   3146*   3152*     *    3169
MASK.A  001406           771#   1570    2143    2278    2380    2907           3140
MASK.B  001407           772#   1575    2149    2284    2387    2913           3146
MASK.C  001410           773#   1580    2155    2290    2394    2919    3037   3152
MASK.D  001411           774#   1585    2161    2296    2401    2925    3043   3158
MASTEK  005400          1326    1484#
MCRLF   005104          1095    1218    1322    1323    1331    1472    1484#   1593    1611
MCSRX   005330          1001    1484#
MDATA   005624          1245    1255    1497#
MEPASS  005145          1000    1484#
MERRPC  005454          1329    1484#
MERRX   005355          1007    1484#
MERR2   005174          1484#   1510    1700
MERR3   005243           955    1484#
MLOCK   005301           979    1484#
MNEW    005402           950    1484#
MPASSX  005344          1005    1484#
MPFAIL  005107          1385    1484#
MQM     005100          1125    1484#   1616
MR      005171           987    1484#
MRESET= 004000           607#   1404    1417
MSTCLR= 104412           737#   1389    1833    1884    1936    1988    2042    2094    2171    2306    2412    2505    2580
                        2660    2731    2836    2936    3054    3181    3320    3423    3510    3580
MTITLE  001000           645#    932
MTSTN   005366          1327    1484#   1595
MTSTPC  005267          1484#
MVECX   005336          1003    1484#
NEXT    001216           662#   1068    1358    1748*   1813*   1862*   1914*   1966*   2020*   2072*   2140*   2275*   2377*
                        2484*   2556*   2632*   2703*   2804*   2904*   3022*   3137*   3292*   3400*   3487*   3557*
```

```
NOLIST= ****** U          527
NPR   = 040000            602#
NPRLOC  022620           3727#
PARAM = 104405            727#   1596
PARAM1  003204           1145#   1162
PARBIT= 040000            607#   1640    2996    3103    3252
PARERR  003360           1148    1150    1152    1161#   1168    1170    1172
PASCNT  001230            667#    908*    998*    999    1032
PC    =%000007            574#    936*   1017*   1047*   1294*   1473*   1572*   1577*   1582*   1587*   1603    1604*   1643*
                         1735*   1752*   1756*   1760*   1764*   1801*   1817*   1821*   1825*   1829*   1849*   1866*   1870*
                         1874*   1878*   1883*   1899*   1918*   1922*   1926*   1930*   1935*   1952*   1970*   1974*   1978*
                         1982*   1987*   1996*   2006*   2024*   2028*   2032*   2036*   2041*   2050*   2060*   2076*   2080*
                         2084*   2088*   2093*   2109*   2117*   2127*   2146*   2152*   2158*   2164*   2169*   2262*   2281*
                         2287*   2293*   2299*   2304*   2364*   2384*   2391*   2398*   2405*   2410*   2470*   2488*   2492*
                         2496*   2500*   2542*   2560*   2564*   2568*   2572*   2577*   2619*   2637*   2642*   2647*   2652*
                         2657*   2682*   2708*   2713*   2718*   2723*   2728*   2783*   2810*   2816*   2822*   2828*   2833*
                         2888*   2910*   2916*   2922*   2928*   2933*   2942*   3006*   3028*   3034*   3040*   3046*   3051*
                         3060*   3119*   3143*   3149*   3155*   3161*   3166*   3268*   3297*   3302*   3307*   3312*   3317*
                         3385*   3404*   3408*   3412*   3416*   3421*   3473*   3491*   3495*   3499*   3503*   3508*   3544*
                         3561*   3565*   3569*   3573*   3578*   3612*   3646*   3664*
PERFOR= 004537            607#   1770    1835    1845    2098    2173    2181    2183    2205    2243    2258    2308    2316
                         2317    2336    2359    2360    2414    2422    2423    2442    2465    2466    2507    2582    2662
                         2665    2669    2733    2736    2740    2761    2838    2841    2845    2866    2938    2946    2949
                         2953    2971    2980    3002    3056    3063    3068    3076    3084    3087    3091    3110    3183
                         3188    3189    3240    3258    3263    3322    3326    3331    3355    3369    3427    3430    3450
                         3514    3517    3584    3587
PFTAB   004470           1386    1392#
POPRO = 012600            581#   1352
POP1SP= 005726            581#
POP2SP= 022626            581#   1070
PS    = 177776            579#    904*    973*   1720*
PUSHRO= 010046            581#   1349
PUSH1S= 005746            581#
PUSH2S= 024646            581#
QV.FLG  001313            705#    910*   1011*   1059
RAM   = 020000            600#
RAMCLR= 104413            739#   1390    1767
RESREG  004300           1343    1346#
RESTAR  004414           1373    1379#
RESTRT  002572           1010    1014    1022#
RESV16  001404            765#   1559*   1560*
RESOS = 104407            731#   1346
RETURN  001214            661#    916*    986*    988    1022*   1068*   1071    1358*   1360    1391    1610*   1618*   1619
ROMCLK= 104415            743#   1781    1794    2178    2192    2194    2215    2217    2313    2321    2323    2419    2427
                         2429    2514    2524    2532    2589    2591    2601    2609    2956    2958    2964    2983    2994
                         3075    3090    3101    3201    3203    3225    3227    3239    3250    3270    3272    3344    3520
                         3522    3590    3667    3669    3674    3684    3687    3698    3702
RUN     001304            695#    912*   1513   1516*   1517*   1524*   1525*
RXSHIF  022326           2669    2740    2761    2845    2866    2946    2949    2953    2980    3063    3068    3087    3331
                         3355    3369   3634#
RO    =%000000            567#    951*    959*    960*    962*    964*    966    967    1049   1055*   1069*   1202   1207*
                         1219   1232*   1236*   1246   1262*   1350*   1395   1396*   1397*   1399*   1425   1426*   1431*
                         1434*   1526*   1532   1533   1534   1535   1536   1537   1538   1539   1540   1541   1542*
                         1548   1550   1556   1558   1560   1563   1565   1567   1569*   1574*   1579*   1584*   1602*
                         1603.   1606   1608   1610   1613   1614   1621   1640   1701*   1710*   1712*   1714   1722*
                         1723*   1749*   1753*   1757*   1761*   1768   1783*   1785*   1796*   1798*   1814*   1818*   1822*
```

```
                1326*  1832   1863*  1867*  1871*  1875*  1899   1896*  1915*  1919*  1923*  1927*  1942
                1949*  1967*  1971*  1975*  1979*  1994   2003*  2021*  2025*  2029*  2033*  2048   2057*
                2073*  2077*  2081*  2085*  2097   2124*  2141*  2147*  2153*  2159*  2170   2180   2203
                2218   2247   2276*  2282*  2288*  2294*  2305   2315   2334   2378*  2385*  2392*  2399*
                2411   2421   2440   2485*  2489*  2493*  2497*  2503   2509   2557*  2561*  2565*  2569*
                2578   2584   2633*  2638*  2643*  2648*  2659   2664   2704*  2709*  2714*  2719*  2730
                2735   2805*  2811*  2817*  2823*  2935   2840   2905*  2911*  2917*  2923*  2935   2940
                2987   3023*  3029*  3035*  3041*  3053   3058   3070   3138*  3144*  3150*  3156*  3167
                3187   3206   3209   3228   3293*  3298*  3303*  3308*  3319   3325   3401*  3405*  3409*
                3413*  3426   3429   3470*  3488*  3492*  3496*  3500*  3513   3516   3541*  3558*  3562*
                3566*  3570*  3583   3586   3609*  3615   3616*  3617*  3619*  3624   3625*  3629*  3632*
                3635   3639*  3642*  3643*  3644*  3645   3651*  3655   3656*  3660*  3663*  3692*  3694*
R1      =:000001   568#   963*   964    965*   966    1013*  1017   1201   1208*  1220   1224*  1226   1227
                1228   1229   1261*  1405   1407*  1410*  1413*  1570*  1575*  1580*  1585*  1625*  1630*
                1635*  1638*  1661*  1663   1665   1667   1670   1683*  1684   1690*  1691   1695*  1711*
                1712   1713*  1714   1715   1769*  1782*  1783   1795*  1796   1834*  2172*  2179*  2307*
                2314*  2413*  2420*  2506*  2581*  2661*  2732*  2837*  2937*  3055*  3182*  3321*  3424*
                3511*  3581*  3685*  3688*  3692
R2      =:000002   569#   1200   1209*  1571*  1576*  1581*  1586*  1626*  1631*  1635*  1639*  1642*  1655*
                1656*  1657   1660*  1670*  1671   1672*  1673*  1674*  1675*  1676*  1677*  1678*  1679*
                1686*  1709*  1721*  1725*  1726*  1727*  1729*  1730*  1731*  1774*  1775   1786*  1787
                1839*  1840   1885*  1937*  1989*  2043*  2095*  2201*  2206*  2207   2222   2246*  2249*
                2251*  2256*  2326*  2328*  2332*  2337*  2432*  2434*  2438*  2443*  2515*  2519*  2523*
                2530*  2536*  2592*  2596*  2600*  2607*  2613*  2962*  2963   2992*  2993   3077*  3081*
                3092*  3096*  3099*  3100   3208*  3215*  3216   3232   3241*  3245*  3248*  3249   3329*
                3333*  3335*  3336   3345   3352*  3357*  3359*  3360   3367*  3371*  3373*  3374   3432*
                3433   3441*  3442   3452*  3453   3461*  3462   3523*  3524   3532*  3533   3591*  3592
                3600*  3601   3636   3637*  3648*  3650*
R3      =:000003   570#   1104   1111*  1121*  1124*  1126   1130*  1199   1210*  1221   1233*  1234*  1235*
                1236   1245*  1246*  1251*  1254*  1260*  1621*  1622*  1623   1628   1633   1772*  1799*
                1837*  1847*  1888*  1897*  1941*  1950*  1993*  2004*  2047*  2058*  2096*  2125*  2175*
                2260*  2310*  2362*  2416*  2468*  2504*  2540*  2579*  2617*  2658*  2690*  2729*  2791*
                2834*  2886*  2934*  3004*  3052*  3117*  3193*  3195*  3196   3210   3221*  3232   3321*
                3234   3236   3260*  3261   3318*  3383*  3422*  3471*  3509*  3542*  3579*  3610*  3679
                3681*  3689*  3691*
R4      =:000004   571#   1105   1110*  1114*  1115*  1116   1123*  1127   1129*  1137   1146*  1147   1149
                1151   1153*  1154   1155.  1176*  1177*  1181*  1198   1211*  1222   1230*  1233   1238*
                1240*  1242*  1259*  1309*  1310*  1311*  1312*  1313*  1314*  1315   1316   1317   1406
                1408*  1409*  1412*  1776*  1777   1788*  1790   1841*  1842   1890*  1892*  1893   1943*
                1945*  1946   1997*  1999*  2000   2051*  2053*  2054   2101*  2103   2110*  2111   2118*
                2119   2233*  2234*  2235   2348*  2349*  2352*  2353   2454*  2455*  2458*  2459   2525*
                2526   2602*  2603   2672*  2673   2676   2744*  2745   2748   2752*  2753   2757   2765*
                2766   2769   2773*  2774   2777   2849*  2850   2953   2857*  2859   2862   2870*  2871
                2874   2878*  2879   2882   2965*  2966   2968   2973*  2974*  2977*  2995*  2998*  2999
                3102*  3105*  3107   3169*  3171   3173*  3174   3176*  3177   3251*  3254*  3255   3337*
                3338   3340   3346*  3347   3349   3361*  3362   3364   3375*  3376   3378   3434*  3435
                3438   3443*  3444   3447   3454*  3455   3458   3463*  3464   3467   3525*  3526   3529
                3534*  3535   3538   3593*  3594   3597   3602*  3603   3606
R5      =:000005   572#   1088   1089*  1093   1098   1100*  1136   1138*  1139   1140   1141   1142   1143
                1144   1145*  1154*  1157*  1158*  1159*  1167   1169   1171   1177   1179*  1182*  1197
                1212*  1223   1231*  1243*  1258*  1307*  1308*  1309   1311   1773*  1777   1789*  1790
                1838*  1842   1886*  1887*  1891*  1893   1939*  1940*  1944*  1946   1991*  1992*  1999*
                2000   2045*  2046*  2052*  2054   2100*  2101   2102*  2103   2107*  2111   2114*  2119
                2176*  2185*  2190   2212*  2213   2221*  2235   2239*  2241   2311*  2330*  2350*  2351*
                2353   2417*  2436*  2456*  2457*  2459   2509*  2510*  2511*  2526   2529*  2594*  2595*
                2586*  2603   2606*  2673*  2674*  2675*  2676   2745*  2746*  2747*  2749   2753*  2755*
```

```
SW01  = 000002          559*   1588
SW02  = 000004          558*
SW03  = 000010          557*
SW04  = 000020          556*
SW05  = 000040          555*
SW06  = 000100          554*
SW07  = 000200          553*
SW08  = 000400          552*   1354
SW09  = 001000          551*   1079
SW10  = 002000          550*   1356
SW11  = 004000          549*   1057
SW12  = 010000          548*   1091    1295
SW13  = 020000          547*   1300
SW14  = 040000          546*
SW15  = 100000          545*
SYNB00  001506          810*
SYNB01  001532          821*
SYNB02  001556          832*
SYNB03  001602          843*
SYNB04  001626          854*
SYNB05  001652          865*
SYNB06  001676          876*
SYNB07  001722          887*
SYNB0D  001512          812*
SYNB0I  001536          823*
SYNB0J  001562          834*
SYNB0K  001606          845*
SYNB0L  001632          856*
SYNB0M  001656          867*
SYNB0N  001702          878*
SYNB0O  001726          889*
SYNC    022572         3178*   3179*   3193    3195    3234    3236    3705*
SYNCX   001240          675*   2381*   2389*   2395*   2402*   2436    2807*   2813*   2819*   2825*   2943    2965
SYNC00  001516          814*
SYNC01  001542          825*
SYNC02  001566          836*
SYNC03  001612          847*
SYNC04  001636          858*
SYNC05  001662          869*
SYNC06  001706          880*
SYNC07  001732          891*
SYNC2A  001426          786*   1535*   2381    2907
SYNC2B  001430          787*   1537*   2388    2913
SYNC2C  001432          788*   1539*   2395    2819
SYNC2D  001434          789*   1541*   2402    2825
SYNC0C  001522          816*
SYND01  001546          827*
SYND02  001572          838*
SYND03  001616          849*
SYND04  001642          860*
SYNC0E  001666          871*
SYNC0G  001712          882*
SYNC07  001736          893*
[illegible]  = 000000   603*   1780    1793    2177    2312    2418    2513    2531    2588    2593    2609    2955    2957
               2982     3074    3089    3238    3269    3271    3343    3519    3521    3599
TEMP    000662         1232    1382*   1383*   1495*
```

```
TEMP1   001246          678#   938*   939    944*  1486   1717*  1718*  2226*  2229*  2230   2341*  2344*  2345
                       2447*  2450*  2451   3190*  3265*
TEMP2   001250          679#   939*   940   1488   2106*  2122*  2331*  2356*  2437*  2462*  3168*  3196*  3197*
                       3199   3210*  3211*  3213   3222*  3223
TEMP3   001252          680#  3170*  3171*  3174*  3177*  3197   3211
TEMP4   001254          681#
TEMP5   001256          682#
TKBSR   001204          653#  1053   1112   1438   1445   1466
TKBBR   001206          654#  1043   1055   1114   1120   1290   1447   1468
TLAST = 021744         1614   3729#
TPBSR   001210          655#  1096   1118   1297   1452
TPBBR   001212          656#  1098*  1120*  1299*  1454*
TRBUF   003762         1278#
TSTNO   001226          666#   915*  1367   1394   1599   1606   1608   1747*  1812*  1861*  1913*  1965*  2019*
                       2071*  2139*  2274*  2376*  2483*  2555*  2631*  2702*  2803*  2903*  3021*  3136*  3291*
                       3399*  3486*  3556*
TST1    007256         1602   1618   1747#
TST10   011344         2072   2139#
TST11   012224         2140   2274#
TST12   012740         2275   2376#
TST13   013504         2377   2483#
TST14   014012         2484   2555#
TST15   014344         2556   2631#
TST16   014642         2632   2702#
TST17   015276         2703   2803#
TST2    007540         1748   1812#
TST20   015762         2804   2903#
TST21   016600         2904   3021#
TST22   017346         3022   3136#
TST23   020336         3137   3291#
TST24   021042         3292   3399#
TST25   021432         3400   3486#
TST26   021744         3487   3556#  3729
TST27 = ******         3557
TST3    007742         1813   1861#
TST4    010150         1862   1913#
TST5    010364         1914   1965#
TST6    010612         1966   2019#
TST7    011040         2020   2071#
TTST    002702          980*   981*   983*   984*  1051#
TWOSYN= 013000          607#  2754   2859   3065   3191
TXSHIF  022246         2205   2336   2442   3614#
TYPOAT  004266         1321   1339   1342#
TYPE  = 104402          721#   932    937    950    955    979    987   1000   1001   1003   1005   1007   1095
                       1108   1125   1218   1255   1322   1323   1326   1327   1329   1331   1335   1340   1385
                       1442   1444   1472   1510   1593   1611   1616   1700
TYPMSG  004166         1319   1322#
VECMAP  007102         1699   1707#
WRDCNT  003742         1226*  1256*  1264#
WRKO.F  004254         1334   1337#
XBX     004060         1296   1298   1300#
XCSR    002604         1002   1024#
XERR    002626         1008   1033#
XFF   = 030000          601#  2193   2216   2322   2428   2522   2599   2962   2992   3099   3202   3236   3249
XHEAD   003461          937   1484#
XPASS   002620         1006   1030#
```

```
XSTATQ  005506            943   1484#
XTSTN   004374           1329   1365#
XVEC    003612           1004   1027#
$CRAP = 177777              1#   1738#  1743#  1804#  1808#  1852#  1856#  1902#  1908#  1955#  1960#  2009#  2014#
                         2063#  2067#  2130#  2135#  2265#  2270#  2367#  2372#  2473#  2479#  2545#  2551#  2622#
                         2627#  2685#  2698#  2786#  2799#  2891#  2899#  3010#  3017#  3122#  3132#  3276#  3287#
                         3398#  3395#  3476#  3482#  3547#  3552#
$E      = 000030            1#   1748   1749#  1813   1814#  1862   1863#  1914   1915#  1966   1967#  2020   2021#
                         2072   2073#  2140   2141#  2275   2276#  2377   2378#  2484   2485#  2556   2557#  2632
                         2633#  2703   2704#  2804   2805#  2904   2905#  3022   3023#  3137   3138#  3292   3293#
                         3400   3401#  3487   3488#  3557   3558#
$N      = 000026            1#   1738   1745   1749#  1804   1810   1814#  1852   1859   1963#  1902   1911   1915#
                         1955   1963   1967#  2009   2017   2021#  2063   2069   2073#  2130   2137   2141#  2265
                         2272   2276#  2367   2374   2378#  2473   2481   2485#  2545   2553   2557#  2622   2629
                         2633#  2685   2700   2704#  2786   2801   2805#  2891   2901   2905#  3010   3019   3023#
                         3122   3134   3138#  3276   3289   3293#  3388   3397   3401#  3476   3484   3488#  3547
                         3554   3558#  3729#
$Y      = 000017            1#   708#   717    719#   721#   723#   725#   727#   729#   731#   733#   735#   737#
                          739#   741#   743#   745#   747#
.       = 033106          618#   619    622#   629#   630#   631#   632#   635#   637#   640#   644#   646#   691#
                          692#   693#   694#   695#   696#   805#   807#   808#   809#   810#   811#   812#   813#
                          814#   815#   816#   818#   819#   820#   821#   822#   823#   824#   825#   826#   827#
                          829#   830#   831#   832#   833#   834#   835#   836#   837#   938#   940#   841#   842#
                          843#   844#   845#   846#   847#   848#   849#   851#   852#   853#   854#   955#   856#
                          857#   858#   859#   860#   862#   863#   864#   865#   866#   867#   868#   869#   870#
                          871#   873#   874#   875#   876#   877#   878#   879#   880#   881#   882#   884#   985#
                          886#   887#   888#   889#   890#   891#   892#   893#   957    1046   1293   1375   1384
                         1398   1436#  1446   1453   1467   1494#  1496#  1498#  1512   1703   1724   1771#  1836#
                         1882   1934   1986   2040   2092   2104   2168   2174#  2187   2198   2211   2220   2236
                         2303   2309#  2329   2354   2409   2415#  2435   2460   2508#  2527   2576   2583#  2604
                         2656   2663#  2727   2734#  2749   2758   2770   2778   2832   2839#  2854   2863   2975
                         2883   2932   2939#  2969   3000   3007#  3050   3057#  3108   3165   3194#  3220   3230
                         3256   3316   3323#  3420   3428#  3507   3515#  3577   3585#  3705#
.BEGIN  002332            973#
.CNVRT  003542            736   1219#
.CONVR  003536            734   1218#
.DATAC  004576            746   1424#
.DELAY  004476            742   1395#
.EOP    002436            995#  3557
.ERRTA  023056           1314   3742#
.HLT    004002            625   1287#
.INSTE  003224            726   1125#
.INSTR  003120            724   1104#
.INST1  003140           1108#  1128
.MSG    003142           1106#  1109#
.MSTCL  004556            738   1416#
.PARAM  003244            728   1136#
.PFAIL  004402            623    906   1372#  1380
.RAMCL  004516            740   1403#
.RESOS  003504            732   1207#
.ROMCL  004566            744   1420#
.SAVOS  003444            730   1193#
.SCOPE  002634            718   1040#
.SCOP1  003020            720   1078#
.START  001742            641    904#   916
.TRPSR  003750            627   1275#
```

```
.TRPTA  001314            716#   1280
.TYPE   003044            722    1088#
```

```
DVEND        1#     999
DVFRNT       1#     527
HLT        581#    1433    1779    1792    1844    1895    1948    2002    2056    2105    2113    2121    2188    2199    2237
          2255     2355    2461    2521    2528    2538    2598    2605    2615    2678    2750    2759    2771    2779    2855
          2864     2876    2894    2970    3001    3083    3098    3109    3247    3257    3342    3351    3366    3380    3440
          3449     3460    3469    3531    3540    3599    3608    3631    3662
$BUFFE       1#    1490
$CK15        1#    3646
$CK:50       1#    3654
$CLR.T       1#    3700
$CYCLE       1#    1499
$ECP         1#     999
$FINI        1#    3729
$GETFL       1#
$GETPA       1#    1594
$HEADE       1#     527
$LC01      527#    1737
$LC02      527#    1851    1901    1954    2008
$LC02A     527#    2062
$LC03      527#    2129
$LC03A     527#
$LC04      527#    2264    2366
$LC05      527#    2472    2544
$LC06      527#    2621
$LC06A     527#    2684    2785
$LC06B     527#    2890
$LC07      527#    3009
$LC07A     527#    3121
$LC10      527#    3275
$LC11      527#    3387
$LC12      527#    3475
$LC13      527#    3546
$MSG         1#    1484
$PFAIL       1#    1368
$RAMCL       1#    1395
$RXSHI       1#    3634
$SCOPE       1#    1036
$SETLI       1#    1745    1810    1859    1911    1963    2017    2069    2137    2272    2374    2481    2553    2629    2700
          2801     2901    3019    3134    3289    3397    3484    3554
$SETSC       1#    3677
$SETSY       1#
$SET.T       1#    3696
$SILOI       1#    3665
$SIMBC       1#
$TRPDE       1#     717     719     721     723     725     727     729     731     733     735     737     739     741     743
           745
$TSTN        1#    1745    1810    1859    1911    1963    2017    2069    2137    2272    2374    2481    2553    2629    2700
          2801     2901    3019    3134    3289    3397    3484    3554
$TXSHI       1#    3614
$VARIA       1#     643
$XZ          1#    1738    1743    1804    1808    1852    1856    1902    1908    1955    1960    2009    2014    2063    2067
          2130     2135    2265    2270    2367    2372    2473    2479    2545    2551    2622    2627    2685    2698    2786
          2799     2891    2899    3010    3017    3122    3132    3276    3287    3389    3395    3476    3492    3547    3552
```

```
ADCB    1517  1525
ADD      944  1090  1107  1178  1225  1235  1280  1311  1314  1410  1513  1527  1548  1550  1556
        1558  1560  1563  1565  1567  1583  1728  2529  2606
ASL     1157  1158  1159  1278  1310  1312
BCC     1420  1719
BCS     1459
BED      941   949   978  1014  1042  1051  1060  1079  1081  1117  1148  1156  1250  1289  1296
        1303  1319  1325  1334  1339  1343  1357  1450  1589  1641  1682  1689  1778  1791  1843
        1892  1894  1934  1947  1986  2001  2040  2055  2092  2104  2112  2120  2168  2187  2225
        2236  2240  2242  2303  2340  2354  2409  2446  2460  2518  2527  2576  2595  2604  2656
        2677  2727  2749  2759  2770  2778  2832  2854  2863  2875  2883  2932  2969  2997  3000
        3050  3080  3095  3104  3108  3113  3115  3165  3235  3237  3244  3253  3256  3266  3316
        3341  3350  3365  3379  3439  3448  3459  3468  3530  3539  3598  3607  3682
BGT     1152  1463
BHI     1168
BIC      960  1044  1234  1279  1291  1313  1448  1464  1469  1622  1726  1731  1887  1891  1892
        1940  1944  1945  1998  1999  2046  2052  2053  2102  2114  2115  2350  2352  2456  2458
        2675  2747  2768  2776  2852  2873  2881  2967  2998  3105  3197  3211  3254  3339  3377
        3437  3446  3638
BICB    1115  1153  2234  2349  2351  2455  2457  2990
BIS     1421  1427  1465  1672  1673  1674  1675  1676  1677  1678  1679  1725  1730  1992  1995
        2049  2107  2108  2116  2586  2674  2746  2756  2767  2775  2851  2861  2872  2880  2991
        3174  3177  3213  3348  3363  3436  3445  3456  3457  3465  3466  3527  3528  3536  3537
        3595  3596  3604  3605  3623  3644  3645  3680
BISB    1154  2989
BIT      948   977  1050  1057  1078  1091  1295  1300  1354  1356  1438  1588  1640  1881  1933
        1985  2039  2091  2167  2186  2197  2210  2219  2253  2302  2408  2517  2534  2575  2594
        2611  2655  2726  2754  2831  2859  2931  2960  2985  2996  3049  3065  3079  3094  3103
        3164  3191  3219  3229  3243  3252  3315  3419  3506  3576
BITB    1171  1513  2241  3114
BLO     1170
BLOS     954  1716
BLT     1150  1461
BMI      935  1751  1755  1759  1763  1816  1820  1824  1828  1865  1869  1873  1877  1917  1921
        1925  1929  1969  1973  1977  1981  2023  2027  2031  2035  2075  2079  2083  2087  2145
        2151  2157  2163  2280  2286  2292  2298  2383  2390  2397  2404  2487  2491  2495  2499
        2559  2563  2567  2571  2636  2641  2646  2651  2707  2712  2717  2722  2809  2815  2821
        2827  2909  2915  2921  2927  3027  3033  3039  3045  3142  3148  3154  3160  3296  3301
        3306  3311  3403  3407  3411  3415  3490  3494  3498  3502  3560  3564  3568  3572
BNE      919   947   968   976  1010  1046  1058  1063  1092  1099  1122  1172  1180  1244  1248
        1253  1257  1293  1301  1321  1355  1384  1398  1411  1432  1439  1471  1509  1514  1520
        1530  1592  1605  1607  1609  1615  1624  1629  1634  1658  1664  1666  1668  1685  1696
        1724  1800  1848  1898  1951  2005  2059  2123  2126  2198  2208  2211  2220  2223  2231
        2250  2254  2257  2261  2329  2338  2346  2357  2363  2435  2444  2452  2463  2469  2520
        2535  2537  2541  2597  2612  2614  2618  2681  2755  2782  2960  2987  2961  2978  2996
        3005  3066  3082  3097  3118  3192  3217  3220  3230  3233  3246  3262  3334  3358  3372
        3384  3420  3472  3507  3543  3577  3611  3630  3649  3661  3690
BPL     1054  1094  1097  1113  1119  1298  1348  1446  1453  1467  3627  3658
BR       926   945   957   982  1052  1056  1128  1160  1162  1375  1512  1522  1612  1617  1627
        1632  1637  1703  1729  3194
CLC     1237  1239  1241  1455  1515  1523  1692  1784  1797  2227  2342  2448  2975  3172  3175
        3640  3693
CLR      908   913   914   922   951   965   996  1048  1065  1066  1145  1382  1388  1409  1426
        1451  1656  1701  1720  1722  1727  1769  1834  1885  1937  1999  2043  2095  2172  2176
        2185  2189  2202  2209  2307  2311  2319  2333  2413  2417  2425  2439  2506  2515  2530
        2581  2592  2607  2661  2732  2837  2937  3055  3077  3092  3168  3170  3182  3198  3218
```

```
        3241  3321  3424  3511  3581  3625  3639  3656
CLRB     909   910   997  1064  1254  1305  1387  1625  1659  1687  2511  3072  3086  3212
CMP      927   940   966   967  1041  1062  1167  1169  1288  1302  1460  1462  1519  1528  1603
        1606  1608  1614  1628  1633  1657  1665  1684  1714  1715  1732  1777  1790  1842  1893
        1946  2000  2054  2103  2111  2119  2207  2224  2230  2235  2339  2345  2353  2445  2451
        2459  2526  2603  2676  2748  2757  2769  2777  2853  2862  2874  2892  2968  2999  3107
        3216  3234  3236  3255  3261  3340  3349  3364  3378  3438  3447  3458  3467  3529  3538
        3597  3606
CMPB     953  1045  1116  1147  1149  1151  1155  1292  1449  1470  1681
COMB     933
DEC     1121  1243  1256  1397  1695  1799  1847  1897  1950  2004  2058  2122  2125  2206  2221
        2249  2256  2260  2328  2337  2356  2362  2434  2443  2462  2468  2540  2617  2680  2781
        2886  2974  3004  3117  3215  3231  3265  3333  3357  3371  3383  3471  3542  3610  3689
DECB    1009  1179  1247  1252  2977  3648
EMT      581
HALT     619   952   956   961  1351  1374  1511  1702
INC      998  1061  1353  1393  1431  1544  1546  1552  1554  1723  1782  1795  1896  1949  2003
        2057  2124  2179  2212  2229  2314  2344  2420  2450  2519  2536  2539  2596  2613  2615
        2679  2780  2885  3003  3081  3096  3116  3221  3245  3260  3264  3382  3470  3541  3609
        3629  3660  3685  3688
INCB    1642  1680  1694  2239  3112
JMP      641   988  1023  1071  1282  1360  1391  1619  1699  3267  3273
JSR      936  1017  1047  1294  1572  1577  1582  1587  1752  1756  1760  1764  1817  1821  1825
        1829  1866  1870  1874  1878  1918  1922  1926  1930  1970  1974  1978  1982  1996  2024
        2028  2032  2036  2050  2076  2080  2084  2088  2109  2117  2146  2152  2158  2164  2281
        2287  2293  2299  2384  2391  2398  2405  2488  2492  2496  2500  2560  2564  2569  2572
        2537  2642  2647  2652  2708  2713  2718  2723  2810  2816  2822  2828  2910  2916  2922
        2928  2942  3028  3034  3040  3046  3060  3143  3149  3155  3161  3297  3302  3307  3312
        3404  3408  3412  3416  3491  3495  3499  3503  3561  3565  3569  3573  3646
MOV      904   905   906   911   915   916   920   921   923   928   929   930   931   938   939
         962   963   964   973   974   980   981   983   984   986   999  1013  1022  1043  1049
        1055  1067  1068  1069  1082  1088  1089  1100  1104  1105  1106  1110  1111  1120  1123
        1124  1126  1127  1129  1130  1136  1137  1138  1139  1140  1141  1144  1146  1176  1177
        1181  1182  1193  1197  1198  1199  1200  1201  1202  1207  1208  1209  1210  1211  1212
        1219  1220  1221  1222  1223  1224  1226  1229  1230  1232  1233  1245  1258  1259  1260
        1261  1262  1275  1277  1281  1290  1304  1307  1309  1315  1316  1317  1350  1358  1359
        1373  1380  1381  1395  1396  1399  1404  1405  1406  1407  1408  1412  1413  1417  1425
        1428  1434  1440  1441  1447  1454  1468  1521  1526  1531  1532  1533  1534  1535  1536
        1537  1538  1539  1540  1541  1542  1543  1545  1547  1549  1551  1553  1555  1557  1559
        1562  1564  1566  1569  1570  1571  1574  1575  1576  1579  1580  1581  1584  1585  1586
        1602  1604  1610  1618  1621  1655  1660  1661  1662  1670  1686  1697  1704  1707  1708
        1709  1710  1711  1712  1713  1721  1733  1747  1748  1749  1750  1753  1754  1757  1758
        1761  1762  1768  1772  1773  1774  1775  1776  1780  1783  1786  1787  1798  1789  1793
        1796  1812  1813  1814  1815  1818  1819  1822  1823  1826  1827  1832  1837  1838  1839
        1840  1841  1861  1862  1863  1864  1867  1868  1871  1872  1875  1876  1886  1888  1889
        1890  1913  1914  1915  1916  1919  1920  1923  1924  1927  1928  1938  1939  1941  1942
        1943  1965  1966  1967  1968  1971  1972  1975  1976  1979  1980  1990  1991  1993  1994
        1997  2019  2020  2021  2022  2025  2026  2029  2030  2033  2034  2044  2045  2047  2049
        2051  2071  2072  2073  2074  2077  2078  2081  2082  2085  2086  2096  2097  2100  2101
        2106  2110  2118  2139  2140  2141  2144  2147  2150  2153  2156  2159  2162  2170  2175
        2177  2180  2184  2190  2191  2193  2196  2200  2203  2213  2214  2216  2218  2226  2233
        2247  2251  2274  2275  2276  2279  2282  2285  2288  2291  2294  2297  2305  2310  2312
        2315  2320  2322  2325  2331  2334  2341  2348  2376  2377  2378  2381  2382  2385  2388
        2389  2392  2395  2396  2399  2402  2403  2411  2416  2418  2421  2426  2428  2431  2437
        2440  2447  2454  2483  2484  2485  2486  2489  2490  2493  2494  2497  2498  2503  2504
        2509  2513  2516  2522  2523  2525  2531  2533  2555  2556  2557  2558  2561  2562  2565
```

```
        2566  2569  2570  2578  2579  2584  2588  2590  2593  2599  2600  2602  2608  2610  2631
        2632  2633  2635  2638  2640  2643  2645  2648  2650  2658  2659  2664  2671  2672  2673
        2702  2703  2704  2706  2709  2711  2714  2716  2719  2721  2729  2730  2735  2742  2744
        2745  2751  2752  2753  2763  2765  2766  2772  2773  2774  2803  2804  2805  2807  2808
        2911  2813  2814  2817  2819  2820  2823  2825  2826  2834  2835  2840  2843  2847  2849
        2850  2856  2857  2858  2865  2868  2870  2871  2877  2878  2879  2903  2904  2905  2908
        2911  2914  2917  2920  2923  2926  2934  2935  2940  2941  2955  2957  2959  2962  2963
        2965  2966  2979  2982  2984  2987  2992  2993  2995  3021  3022  3023  3026  3029  3032
        3035  3038  3041  3044  3052  3053  3058  3059  3070  3073  3074  3078  3085  3089  3093
        3099  3100  3102  3136  3137  3138  3141  3144  3147  3150  3153  3156  3159  3167  3178
        3180  3187  3193  3195  3199  3200  3202  3205  3206  3209  3223  3224  3226  3228  3238
        3242  3248  3249  3251  3269  3271  3291  3292  3293  3295  3298  3300  3303  3305  3308
        3310  3318  3319  3324  3325  3328  3330  3335  3336  3337  3338  3343  3345  3346  3347
        3352  3353  3359  3360  3361  3362  3367  3368  3373  3374  3375  3376  3399  3400  3401
        3402  3405  3406  3409  3410  3413  3414  3422  3425  3426  3429  3432  3433  3434  3435
        3441  3442  3443  3444  3452  3453  3454  3455  3461  3462  3463  3464  3486  3487  3488
        3489  3492  3493  3496  3497  3500  3501  3509  3512  3513  3516  3519  3521  3523  3524
        3525  3526  3532  3533  3534  3535  3556  3557  3558  3559  3562  3563  3566  3567  3570
        3571  3579  3582  3583  3586  3589  3591  3592  3593  3594  3600  3601  3602  3603  3615
        3616  3619  3622  3624  3632  3635  3636  3650  3651  3655  3663  3666  3668  3673  3679
        3681  3683  3686  3691  3692  3697  3701
MOVB     907   912   958   959  1011  1012  1098  1114  1142  1143  1227  1228  1231  1236  1246
        1251  1299  1332  1626  1630  1631  1635  1636  1638  1639  1690  1691  1698  1717  2142
        2143  2148  2149  2154  2155  2160  2161  2201  2246  2277  2278  2283  2284  2289  2290
        2295  2296  2326  2330  2332  2379  2380  2386  2387  2393  2394  2400  2401  2432  2436
        2438  2634  2639  2644  2649  2667  2705  2710  2715  2720  2738  2760  2806  2812  2818
        2824  2906  2907  2912  2913  2918  2919  2924  2925  2943  2945  2948  2951  2972  2973
        3024  3025  3030  3031  3036  3037  3042  3043  3061  3067  3139  3140  3145  3146  3151
        3152  3157  3158  3169  3171  3179  3196  3208  3210  3222  3294  3299  3304  3309  3637
NEG     3354
NOP      925   980   981  1018  1019  1020  1021
RESET    995  1015  1654
ROL     1456  1457  1458  3173  3176
ROLB    1516  1524  1693  3617
ROR     1238  1240  1242  1718  1785  1798  2228  2343  2449  3641  3642  3643  3694
RORB    1429  2976  3618
RTI     1083  1101  1131  1183  1203  1213  1263  1361  1400  1414  1418  1422  1435  1705  1734
RTS     1473  1643  1735  1801  1849  1883  1899  1935  1952  1987  2006  2041  2060  2093  2127
        2169  2262  2304  2364  2410  2470  2542  2577  2619  2657  2682  2729  2783  2933  2988
        2933  3006  3051  3119  3166  3268  3317  3385  3421  3473  3508  3544  3578  3612  3664
SUB     1276  1308  2952
SWAB    2510  2585  2988  3071  3207
TRAP     717   719   721   723   725   727   729   731   733   735   737   739   741   743   745
TST      924   946   975  1080  1320  1324  1333  1338  1342  1347  1591  1613  1623  1663  1667
        1671  2222  3232  3626  3657
TSTB     918   934  1053  1059  1093  1096  1112  1118  1249  1297  1318  1445  1452  1466  1508
        1688
.ASCIZ   645  1475  1482  1484  3729
.BLKB    691   692   693   694   695
.BLKW    630   631   632   807   808   809   810   811   812   813   814   815   816   818   819
         820   821   822   823   824   825   826   827   829   830   831   832   833   834   835
         836   837   838   840   841   842   843   844   845   846   847   848   849   851   852
         853   854   855   856   857   858   859   860   862   863   864   865   866   867   868
         869   870   871   873   874   875   876   877   878   879   880   881   882   884   885
         886   887   888   889   890   891   892   893  1436  1771  1836  2174  2309  2415  2508
        2583  2663  2734  2839  2939  3007  3057  3184  3323  3428  3515  3585  3705
```

```
.BYTE    702   703   704   705   771   772   773   774   776   777   778   779  1025  1028  1031
        1034  1363  1366  1393  1479  1485  1487  1600  1601  3706  3707  3708  3709  3710  3711
        3712  3713  3714  3715  3716  3717  3718  3719  3720  3721  3722  3723  3724  3725  3731
        3733  3735  3737  3739
.ENABL   541
.END    3756
.ENDC   1737  1739  1744  1749  1805  1809  1814  1853  1855  1857  1863  1903  1907  1909  1915
        1956  1959  1961  1967  2010  2013  2015  2021  2064  2068  2073  2131  2136  2141  2266
        2271  2276  2368  2373  2378  2474  2479  2480  2485  2503  2513  2546  2551  2552  2557
        2578  2590  2623  2628  2633  2686  2699  2704  2787  2800  2805  2892  2900  2905  3011
        3018  3023  3123  3133  3138  3277  3288  3293  3389  3396  3401  3477  3483  3488  3548
        3553  3558  3756
.EQUIV   581
.EVEN    696   707  1477  1483  1484  1489  3729
.IF      527  1738  1743  1748  1804  1808  1813  1852  1853  1855  1856  1862  1902  1903  1907
        1908  1914  1955  1956  1959  1960  1966  2009  2010  2014  2020  2063  2067  2072  2130
        2135  2140  2265  2270  2275  2367  2372  2377  2473  2474  2479  2484  2503  2513  2545
        2546  2551  2556  2575  2588  2622  2627  2632  2685  2698  2703  2786  2799  2804  2891
        2899  2904  3010  3017  3022  3122  3132  3137  3276  3287  3292  3388  3395  3400  3476
        3482  3487  3547  3552  3557  3756
.IFF     1739  1743  1748  1749  1805  1808  1813  1814  1853  1856  1862  1863  1903  1908  1914
        1915  1956  1960  1966  1967  2010  2014  2020  2021  2064  2067  2072  2073  2131  2135
        2140  2141  2266  2270  2275  2276  2368  2372  2377  2378  2474  2479  2484  2485  2546
        2551  2556  2557  2623  2627  2632  2633  2686  2698  2703  2704  2787  2799  2804  2805
        2892  2899  2904  2905  3011  3017  3022  3023  3123  3132  3137  3138  3277  3287  3292
        3293  3389  3395  3400  3401  3477  3482  3487  3488  3548  3552  3557
.IFT     527  1737  1738  1743  1804  1808  1852  1856  1902  1908  1955  1960  2009  2014  2063
        2067  2130  2135  2265  2270  2367  2372  2473  2474  2479  2545  2546  2551  2622  2627
        2685  2698  2786  2799  2891  2899  3010  3017  3122  3132  3276  3287  3388  3395  3476
        3482  3547  3552
.IFTF    527
.IIF    1748  1749  1750  1754  1758  1762  1813  1814  1815  1819  1823  1827  1862  1863  1864
        1868  1872  1876  1886  1887  1888  1890  1914  1915  1916  1920  1924  1928  1938  1939
        1940  1941  1943  1966  1967  1968  1972  1976  1980  1990  1991  1992  1995  1996  1997
        2020  2021  2022  2026  2030  2034  2044  2046  2047  2049  2050  2072  2073  2074  2078
        2082  2086  2140  2141  2142  2143  2144  2148  2149  2150  2154  2155  2156  2160  2161
        2162  2268  2269  2272  2275  2276  2277  2278  2279  2283  2284  2285  2289  2290  2291
        2295  2296  2297  2318  2319  2330  2331  2370  2374  2377  2378  2379  2380  2381  2386
        2387  2388  2393  2394  2395  2400  2401  2402  2408  2424  2436  2484  2485  2486  2490
        2494  2498  2511  2512  2556  2557  2558  2562  2566  2570  2586  2632  2633  2634  2635
        2639  2640  2644  2645  2649  2650  2696  2700  2703  2704  2705  2706  2710  2711  2715
        2716  2720  2721  2737  2738  2739  2760  2761  2797  2798  2801  2804  2805  2806  2807
        2812  2813  2818  2819  2824  2825  2831  2842  2843  2865  2904  2905  2906  2907  2908
        2912  2913  2914  2918  2919  2920  2924  2925  2926  3022  3023  3024  3025  3026  3030
        3031  3032  3036  3037  3038  3042  3043  3044  3137  3138  3139  3140  3141  3145  3146
        3147  3151  3152  3153  3157  3158  3159  3292  3293  3294  3295  3299  3300  3304  3305
        3309  3310  3400  3401  3402  3406  3410  3414  3487  3488  3489  3493  3497  3501  3557
        3558  3559  3563  3567  3571
.IRP     717   719   721   723   725   727   729   731   733   735   737   739   741   743   745
        807  1738  1745  1748  1804  1810  1813  1852  1859  1862  1902  1911  1914  1955  1963
        1966  2009  2017  2020  2063  2069  2072  2130  2137  2140  2265  2272  2275  2367  2374
        2377  2473  2481  2484  2545  2553  2556  2622  2629  2632  2685  2700  2703  2786  2801
        2804  2891  2901  2904  3010  3019  3022  3122  3134  3137  3276  3289  3292  3388  3397
        3400  3476  3484  3487  3547  3554  3557  3729
.LIST      1   527   541   562   581   607   609   619   643   645   719   721   723   725   727
        729   731   733   735   737   739   741   743   745   747   802   896   989  1037  1484
```

M07


```
          1737    1738    1743    1749    1804    1808    1814    1852    1856    1863    1902    1908    1915    1955    1960
          1967    2009    2014    2021    2063    2067    2073    2130    2135    2141    2265    2270    2276    2367    2372
          2378    2473    2479    2485    2545    2551    2557    2622    2627    2633    2685    2698    2704    2786    2799
          2805    2891    2899    2905    3010    3017    3023    3122    3132    3138    3276    3287    3293    3388    3395
          3401    3476    3482    3488    3547    3552    3558    3729
.MACRO       1     527
.NLIST       1     527     541     562     581     607     609     619     643     645     719     721     723     725     727
           729     731     733     735     737     739     741     743     745     747     802     896     989    1037    1484
          1737    1738    1743    1749    1804    1808    1814    1852    1856    1863    1902    1908    1915    1955    1960
          1967    2009    2014    2021    2063    2067    2073    2130    2135    2141    2265    2270    2276    2367    2372
          2378    2473    2479    2485    2545    2551    2557    2622    2627    2633    2685    2698    2704    2786    2799
          2805    2891    2899    2905    3010    3017    3023    3122    3132    3138    3276    3287    3293    3388    3395
          3401    3476    3482    3488    3547    3552    3558    3729
.PAGE      562     609     698     750     802     896     989    1499    1737
.REM         1     896
.REPT      619
.SBTTL     527     562     609     643     896     989    1037    1737
.TITLE     541
.WORD     1481


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *,DZDVBB/SOL/CRF=DZDVBB.MAC,DZDVBB.P11
 RUN-TIME: 31 45 7 SECONDS
 RUN-TIME RATIO: 820/84=9.7
 CORE USED:  27K  (53 PAGES)
```

Spooler runtime 16 Seconds, 69 KCS, 428 disk reads, 3 disk writes, 90 pages