VT20

DIAGNOSTIC TEST

MD-11-DBVTA-C

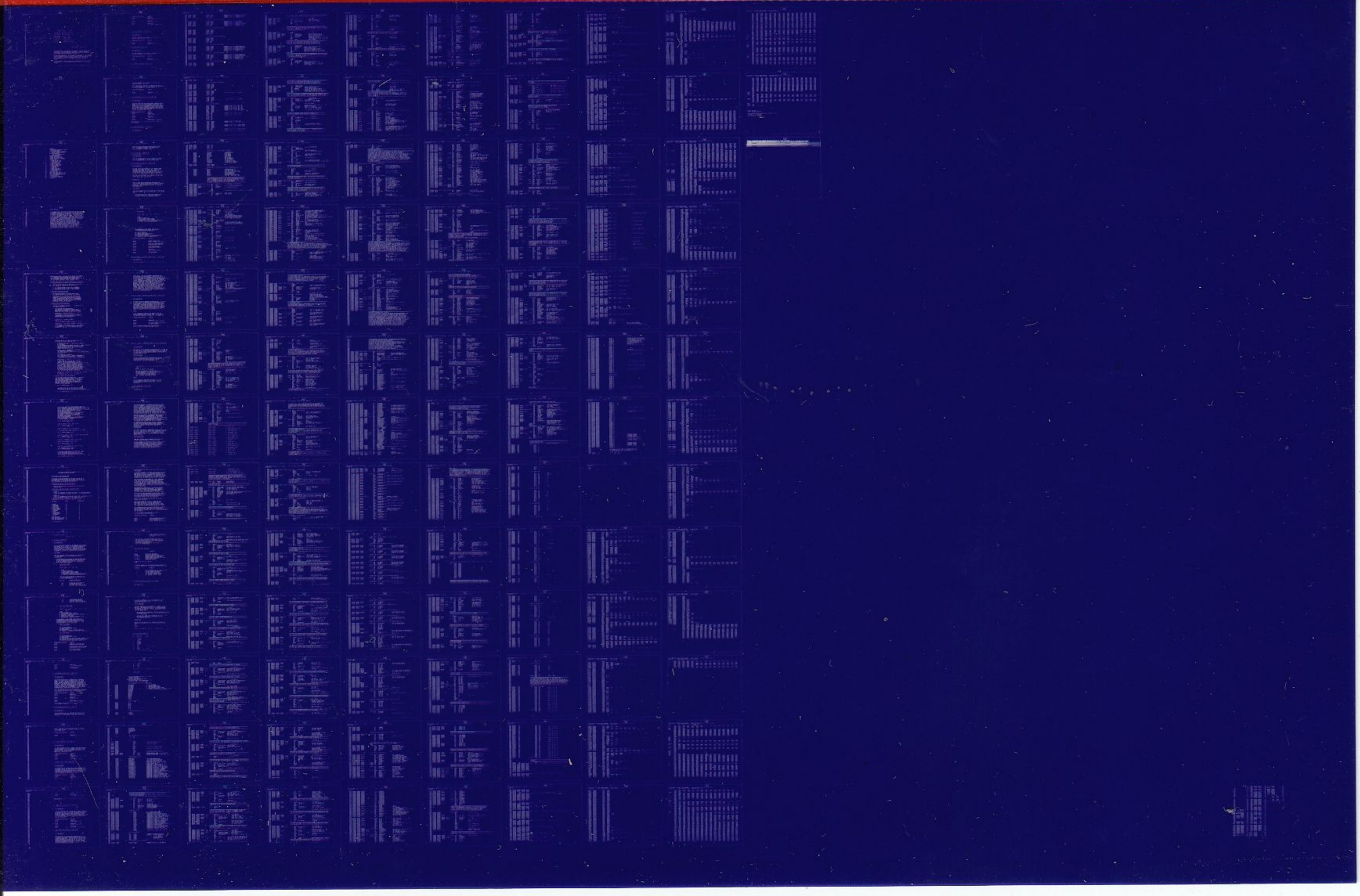
EP DBVTA C DL A

FICHE 1 OF 1

NOV 1976

digital

MADE IN USA



.REM %

IDENTIFICATION

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

PRODUCT NAME: VT20 DIAGNOSTIC TEST

DATE CREATED: JUNE 17, 1973

DATE REVISED: OCTOBER, 1975

MAINTAINER: DIAGNOSTIC GROUP

AUTHOR: ED BADGER/P. NELSON

FIRST PRINTING, JANUARY 1972

55
56

C01
TABLE OF CONTENTS
****

1. A B S T R A C T
****

THIS PROGRAM TESTS, EXERCISES AND DIAGNOSES THE VT20 SYSTEM (KEYBOARDS & DISPLAYS) IN ITS ENTIRETY. THE PROGRAM CONSISTS OF EIGHTEEN INDIVIDUALLY SELECTABLE TESTS WHICH FACILITATE IN CHECKOUT AND ACCEPTANCE OF THE VT20. THE SYSTEM IS TESTED ON A UNIT (1 KEYBOARD & 1 DISPLAY) BASES. THIS HOLDS TRUE IN ALL TESTS EXCEPT FOR TEST 21. HERE, THE VT20 IS TESTED AS A SLAVE SYSTEM IN CONFIGURATION WITH EITHER A PDP-8 OR PDP-11 HOST COMPUTER. THIS TEST REQUIRES FOR THE VT20 TO BE CABLED VIA SERIAL LINE INTERFACE TO THE HOST COMPUTER. THE HOST DIAGNOSTIC, EITHER MD-11-DZVTE(PDP-11 W/DL11'S), MD-11-DZVTG(PDP-11 W/DH11'S) OR MD-08-DIVTB(PDP-8 W/ KL8J'S) IS TO BE LOADED INTO THE HOST COMPUTER. THIS PROGRAM ACTS AS SERIAL LINE INTERFACE DIAGNOSTIC AND A DATA RECEIVE/TRANSMIT ROUTINE. REFER TO THE SPECIFIC MAINDEC FOR A FURTHER EXPLAINATION. THIS DIAGNOSTIC IS SET UP TO OPERATE 'WITH' OR 'WITHOUT' A TELETYPE. IF A TELETYPE ISN'T AVAILABLE PROGRAM 'HALTS' ARE USED TO REPORT ERRORS AND COLLECT NEEDED INFORMATION. THE PROGRAM

RESPONDS FUNCTIONALLY THE SAME WITH OR WITHOUT THE TELETYPE.
IT IS IMPERATIVE THAT THE DISPLAY TESTS BE RUN AND PROVED
FULLY OPERATIONAL BEFORE RUNNING THE KEYBOARD TESTS.
THIS IS NECESSARY SINCE THE DISPLAY IS USED IN CONJUNCTION
WITH THE 'KBD' (FUNCTIONAL & ASCII KEYBOARD) TESTS.

2. REQUIREMENTS (EQUIPMENT & MEMORY)

A. VT20 WITH EITHER 8 OR 16K OF MEMORY AND 1 TO 4 TUBES.
B. HOST COMPUTER W/ CONSOLE DEVICE (TEST 21 ONLY)

1. MD-11-DZVTE FOR PDP-11 HOST W/DL11 INTERFACE
2. MD-11-DZVTG FOR PDP-11 HOST W/ DH11 INTERFACE
3. MD-08-DIVTB FOR PDP-8 HOST W/ KL8J INTERFACE

3. LOADING PROCEDURE

A. USE STANDARD PROCEDURE FOR LOADING BINARY TAPES
IF A READ IN DEVICE (PC11 OR ASR-33) IS AVAILABLE.

B. THIS PROGRAM MAY ALSO BE BOOTED OVER FROM THE HOST
DIAGNOSTIC. THIS IS DONE BY MODIFING THE BOOTSTRAP
LOADED. SIMPLY REPLACE THE PC11 OR TTY CSR ADDRESS
IN LOCATION 37776 WITH ONE OF THE DL11 CSR ADDRESSES
(175510 OR 175620). REFER TO THE PARTICULAR MAINDEC
BEING USED FOR THE BOOT PROCEDURE.

4. STARTING PROCEDURE

A. THERE ARE TWO STARTING ADDRESSES. (1) WITH TELETYPE AND
(2) WITHOUT A TELETYPE.

1. WITH TELETYPE (OR OTHER TERMINAL).
LOAD AND START THE PROGRAM AT ADDRESS '200'.
WHEN STARTED THE PROGRAM WILL PRINT THE PROGRAM
HEADER AND THEN A SERIES OF QUESTIONS ARE ASKED.

QUESTION NO.1 "FOREIGN CHARACTER SET (Y OR N)?"

THIS ENABLES THE PROGRAM TO DISPLAY THE FOREIGN
CHARACTER SET IN THE CHARACTER DISPLAY TESTS IF THE
FOREIGN CHARACTER ROMS ARE PRESENT.

QUESTION NO.2 - "NUMBER OF TUBES?"

TYPE IN THE NUMBER OF TUBES ON SYSTEM (1,2,3 OR 4)

QUESTION NO.3 - "ARE DEFAULT ADDRESSES & VECTORS OK (Y OR N)?"

IF THE ANSWER TO THIS QUESTION WAS 'Y', THEN STANDARD
'FK' KEYBOARD AND 'VT' DEVICE ADDRESSES AND VECTOR
ADDRESSES WILL BE USED. IT WILL ALSO AUTOMATICALLY

ASSOCIATE THE 1ST DL (175610) WITH THE FIRST TUBE,
DL-175620 WITH THE 2ND TUBE AND SO ON.

IF QUESTION NO.3 WAS 'N', THEN THE NEXT QUESTIONS
WILL BE ASKED:
"FK VECTOR AND DEVICE ADDRESS FOR UNIT X" (WHERE X REPRESENTS
THE UNIT ADDRESS TO BE SET UP. RESPOND TO THIS
QUESTION BY TYPING THE FK 'VECTOR' AND 'DEVICE'
ADDRESS SEPERATED BY A COMMA.

AFTER ALL THE FK ADDRESSES HAVE BEEN SETUP,
THE PROGRAM WILL THEN PRINT:
"VT VECTOR AND DEVICE ADDRESS FOR UNIT X" - ANSWER SAME AS
ABOVE ONLY FOR VT.

THE PROGRAM WILL THEN PRINT:
"DL11 VECTOR FOR UNIT X" - WHERE X REPRESENTS THE
UNIT NUMBER OF THE ASSICIATED DL11. RESPONSE
TO THIS QUESTION BY TYPING THE DL11 VECTOR ADDRESS.

QUESTION NO.3 - "TUBE '0','1','2','3',OR '4'"

TO RUN A TEST ON A SELECTED TUBE, TYPE THE
NUMBER OF THE TUBE TO BE TESTED (0,1,2 OR 3)
AND (CR). ALSO, AN OPTION TO THIS IS TO TYPE
A '4'. THIS WILL CAUSE THE PROGRAM TO TEST ALL
AVAILABLE TUBES, ONE AT A TIME ON ANY SELECTED
TEST 1-15. THUS, AFTER ONE TUBE HAS COMPLETED
A TEST, THE NEXT SEQUENTIAL TUBE WILL AUTOMATICALLY
BE TESTED. AFTER THE LAST TUBE HAS BEEN TESTED,
THE PROGRAM WILL RE-CYCLE AND AGAIN BE TESTED.
THIS OPTION CAN ALSO BE USED TO RUN EVERY TEST
(1-15) ON EVERY TUBE (REFER TO SECTION 14.).

QUESTION NO.4 "TYPE TEST NO. TO BE EXECUTED" - RESPOND

BY TYPING THE TEST NO: '0-21' YOU WISH TO RUN.

2. WITHOUT TELETYPE (OR OTHER TERMINAL)

LOAD AND START PROGRAM AT ADDRESS '204'. THE PROGRAM
WILL THEN EXECUTE A SERIES OF HALTS, THESE HALTS
ENABLE THE USER TO ANSWER QUESTIONS TO THE PROGRAM
VIA THE SWITCH REGISTER. PDP 11/05, 11/10 HALT
HALT LOCATIONS ARE INDICATED BY THE LOCATIONS ON
BRACKETS. THESE HALTS WITH THEIR ASSOCIATED HALT
ADDRESSES ARE NOW DESCRIBED:

A. HALT AT ADDRESS 001734 (1736)

    HALT USED TO ASK YOU IF YOUR VT20 HAS A FOREIGN
    CHARACTER SET, IF SO SET THE SWITCH REGISTER EQUAL TO
    A 1 (IF NOT CLEAR THE SWITCH REGISTER) THEN PRESS CONTINUE.

B. HALT AT ADDRESS 001746 (1750)

```
                                        ------

                                    AT THIS HALT, ENTER THE OCTAL NUMBER OF TUBES YOU
                                    HAVE (1 THROUGH 4) INTO THE SWITCH REGISTER.  ALSO IF YOU
                                    WISH TO ENTER THE ADDRESSES AND VECTORS OF THE FK, VT, OR
                                    DL11'S, SET SWR (SWITCH REGISTER) BIT 15 TO A 1, OTHERWISE LEAVE
                                    SWR BIT 15=0 (CLEARED) AND DEFAULT ADDRESS.
                                    AND VECTORS WILL BE USED.
                                    IF YOU INDICATED IN HALT 'B', VIA SETTING SW15=1,
                                    THAT DEFAULT ADDRESSES ARE NOT TO BE USED,
                                    THEN HALTS 'C' THRU 'G' WILL BE EXECUTED. THIS
                                    WILL ENABLE YOU TO SELECT YOUR OWN DEVICE,
                                    VECTOR & DL11 ADDRESSES.
                                    NOTE: THIS SERIES OF HALTS WILL OCCUR SEQUENTIALLY
                                    FOR EACH TUBE THAT YOU INDICATED SO PAY HEED
                                    TO THE HALT ADDRESS - PLEASE.

                                C.  HALT AT ADDRESS 002010 (2012)
                                                    ------

                                    ENTER FK VECTOR OF UNIT N INTO THE SWR.

                                D.  HALT AT ADDRESS 002022 (2024)
                                                    ------

                                    ENTER FK ADDRESS OF UNIT N INTO THE SWR.

                                *.  AT THIS POINT HALTS C AND D REPEATED FOR ALL TUBES.

                                E.  HALT AT ADDRESS 002050 (2052)
                                                    ------

                                    ENTER THE VT VECTOR OF TUBE N - INTO THE SWITCH REGISTER.

                                F.  HALT AT ADDRESS 002060 (2062)
                                                    ------

                                    ENTER THE VT ADDRESS OF TUBE N INTO THE SWR

                                *.  HALTS E AND F REPEATED FOR ALL TUBES.

                                G.  HALT AT ADDRESS 002114 (2116)
                                                    ------

                                    ENTER DL11 VECTOR ADDRESS FOR TUBE N INTO SWR.

                                *.  HALT G REPEATED FOR ALL TUBES.
                                H.  HALT AT ADDRESS 002136 (2140)
                                                    ------

                                    AT THIS HALT ENTER INTO THE SWR THE OCTAL NUMBER
                                    OF THE TUBE TO BE TESTED (0,1,2,3 OR 4).  IF
                                    OCTAL 4 IS ENTERED IN THE SWR, ALL UNITS WILL
                                    BE AUTOMATICALLY ALTERNATED AFTER PASS COMPLETION.

                                I.  HALT AT ADDRESS 002634 (2636)
```

```
                                                ------

                                    THIS HALT IS FOR THE TEST NUMBER TO BE RUN.  ENTER
                                    THE NUMBER AND PRESS CONTINUE.


                          5.        R E S T A R T   P R O C E D U R E
                                    **********************************

                                    THE PROGRAM SHOULD BE RESTARTED AT THE SAME LOCATION THAT IT
                                    WAS ORIGINALLY STARTED UNLESS A NEW TUBE IS TO BE TESTED. IN
                                    THIS CASE TYPE A '↑A' IF A TELETYPE IS BEING USED. OTHERWISE,
                                    HALT AND RESTART THE PROGRAM AT LOCATION '210'.

                          6.        C O N S O L E   S W I T C H   S E T T I N G S
                                    ********************************************

                                    A.  REFER TO THE INDIVIDUAL TEST DESCRIPTIONS FOR APPLICABLE
                                        'SWR' SETTINGS.

                          7.        C O N T R O L   S W I T C H E S   ( T E L E T Y P E )
                                    ******************************************************

                                    A.  CONTROL <C>
                                        TYPING '↑C' (OBTAINED VIA TYPING THE 'CNTR' & 'C' KEYS SIMULTAEOUSLY)
                                        AT ANY TIME ENABLES THE PROGRAM TO RETURN TO  THE KEYBOARD MONITOR.

                                    B.  CONTROL <A>
                                        TYPING '↑A' (OBTAINED VIA TYPING THE 'CNTRL' AND 'A' KEYS SIMULTANEOUSLY)
                                        AT ANY TIME CAUSES A RESTART OF THE CURRENT TEST BEING RUN.
                          9.        P R O G R A M   T E S T   T A B L E   &   D I R E C T O R Y
                                    ************************************************************

                                    TEST NAME                TEST NO.              DESCRIPTION
                                    ---------                --------              -----------

                                    VT LOGIC                   00                     9
                                    VT CHARACTER               01                    10
                                    VT FIELD MODE              02                    11
                                    VT CSR PRESET              03                    12
                                    VT END OF LINE             04                    13
                                    VT END OF SCREEN           05                    14
                                    VT BLANK CONTROL           06                    15
                                    VT ALIGNMENT               07                    16
                                    VT FOCUS                   10                    17
                                    VT WORST CASE              11                    18
                                    VT CURSOR CONTROL          12                    19
                                    VT ODD ADDRESS             13                    20
                                    VT RUN-ALL                 14                    21

                                    ASCII KBD LOGIC            15                    22
                                    ASCII KBD SWITCH & LIGHT   16                    23
                                    ASCII KBD CHARACTER SET    17                    24
                                    ASCII KBD REPEATIBILITY    20                    25

                                    VT/KBD SYSTEM TEST         21                    26
```

9. V T L O G I C T E S T (0)

****

A. TEST DESCRIPTION

THE 'VT' LOGIC TEST CONSISTS OF '28' SUBTESTS WHICH EXERCISE AND DIAGNOSE THE 'VT' LOGIC. EACH SUBTEST IS LOOPED '100' TIMES TO TEST LOGIC RELIABILITY. THESE TESTS ARE RUN AS A UNIT WHEN THE 'VT' LOGIC TEST IS SELECTED OR MAY BE SELECTED INDIVIDUALLY BY THE FK/VT SUBTEST SELECTOR (REFER TO SECTION 27). AT THE END OF EACH SUBTEST THE 'AUDIO BEEP' IS SOUNDED TO INDICATE A NEW SUBTEST IS BEING EXECUTED.

B. LOGIC ERRORS

THERE ARE TWO ERROR REPORTING SCHEMES USED FOR REPORTING 'VT' LOGIC ERRORS, (1) WITH TELETYPE AND (2) WITH PROGRAM HALTS.

1. ON ENCOUNTERING A LOGIC ERROR (ALL DATA SWITCHES DOWN) THE FAILING SUBTEST NUMBER ERROR ADDRESS,AND CONTENTS OF THE 'VT' REGISTERS ARE TYPED OUT. THERE ARE TWO ERROR FORMATS USED, (A) NORMAL 'VT' LOGIC ERROR OR (B) 'VT' SHIFT LOGIC ERROR.

(A). NORMAL ERROR FORMAT

TST NO.	PASS	MA	CSR	MAR
A	B	C	D	E

WHERE:
A= FAILING SUBTEST NO.
B= PASS ON WHICH ERROR OCCURRED.
C= ERROR ADDRESS (MEMORY ADDRESS)
D= CONTENTS OF CONTROL & STATUS ADDRESS REGISTER
E= CONTENTS OF MAINTENANCE ADDRESS REGISTER*

*'SWR' BITS '2-0' MAY BE SET TO SELECT ANY ONE OF THE '8' ADDRESSES THAT CAN BE READ BACK VIA THE MAINTENANCE REGISTER.

'SWR' 2-0	ADDRESS DATA SELECTED
000	COMMAND AND STATUS REGISTER
001	DATA ADDRESS REGISTER
010	SHIFT REG. INPUT DATA(EVEN ROW)
011	SHIFT REG. OUTPUT DATA(EVEN ROW)

```
                                        100         CURSOR ADDRESS REGISTER
                                        101         STARTING ADDRESS REGISTER
                                        110         SHIFT REG. INPUT DATA(ODD ROW)
                                        111         SHIFT REG. OUTPUT DATA(ODD ROW)



                        (B).  SHIFT LOGIC ERROR FORMAT
                              ------------------------

                              TST NO.     MA      SHIFT    IN/OUT     EXPT'D     REC'D
                                 A        B        C         D          E          F

                              WHERE:
                              A= FAILING SUBTEST NO.
                              B= ERROR ADDRESS (MEMORY ADDRESS).
                              C= THE SHIFT IN OCTAL ON WHICH ERROR OCCURRED (1-32).
                              D= DIRECTION OF DATA SHIFT
                              E= EXPECTED DATA (SHIFTED IN FROM SHIFT TABLE).
                              F= RECEIVED DATA (CONTENTS OF 'MAR').

                    2.  ON ENCOUNTERING A LOGIC ERROR (WITH DATA SW15 UP)
                        THE PROGRAM WILL SOUND THE AUDIO BEEP AND LOAD THE
                        INTERNAL REGISTERS WITH ERROR DATA AND HALT. THESE
                        REGISTERS ARE SET UP IN ONE OF TWO WAYS: (A) NORMAL
                        LOGIC ERROR AND (B) SHIFT LOGIC ERROR.

                        (A).  NORMAL LOGIC ERROR
                              ------------------

                              R0= FAILING SUBTEST NO.
                              R1= PASS ON WHICH ERROR OCCURRED
                              R2= ERROR ADDRESS
                              R3= CONTENTS OF CONTROL & STATUS ADDRESS REG.
                              R4= CONTENTS OF MAINTENANCE ADDRESS REG *(REFER ABOVE).

                        (B).  SHIFT LOGIC ERROR
                              -----------------

                              R0= FAILING SUBTEST NO.
                              R1= PASS ON WHICH ERROR OCCURED
                              R2= ERROR ADDRESS
                              R3= THE SHIFT IN OCTAL ON WHICH ERROR OCCURRED (1-100).
                              R4= EXPECTED DATA (SHIFTED IN FROM THE SHIFT TABLE).
                              R5= ACTUAL CONTENTS OF MAINTENANCE ADDRESS REG.

                C.  CONSOLE SWITCH SETTINGS     FUNCTION
                    -----------------------     --------
                    SW11=0                      NORMAL RUN [100 PASSES/TEST]
                    SW11=1                      SUPPRESS SUBTEST ITERATIONS

                    SW12=0                      CONTINUE SHIFTING ON SHIFT ERROR
                    SW12=1                      ABORT SHIFTING ON SHIFT ERROR

                    SW13=0                      PRINT ERROR MESSAGES
                    SW13=1                      INHIBIT ERROR TYPEOUT
```

SW14=0	NORMAL RUN
SW14=1	LOOP ON CURRENT SUBTEST
SW15=0	CONTINUE ON ERROR
SW15=1	HALT ON ERROR

10. V T ' C H A R A C T E R S E T ' T E S T (1)

A. TEST DESCRIPTION

THIS TEST DISPLAYS THE 'VT' CHARACTER SET IN NUMERICAL ORDER. THE CHARACTER SET IS REPEATED UNTIL THE ENTIRE SCREEN IS FILLED. THE TEST STARTS DISPLAYING A BLANK SCREEN. AS EACH CHARACTER IS CREATED IT IS DISPLAYED FOR APPROXIMATELY ONE SECOND AND THEN THE NEXT CHARACTER IS DISPLAYED. THE DISPLAY BUFFER IS SET UP SO THAT EACH LINE (EXCEPT THE LAST LINE) IS TERMINATED WITH EITHER VISIBLE 'END OF LINE' OR VISIBLE 'END OF PARAGRAPH'. THE LAST LINE IS TERMINATED WITH 'VISIBLE END OF SCREEN'. WHEN THE ENTIRE SCREEN HAS BEEN FILLED THE TEST IS RESTARTED.

NOTE: THE CHARACTER SET MAY CONTAIN BLANK SPACES. THESE ARE CODES RESERVED FOR FUTURE SPECIAL TYPESETTING CHARACTERS.

B.

CONSOLE SWITCH SETTINGS	FUNCTION
SW11=0	NORMAL RUN
SW11=1	INHIBIT DISPLAY DELAY
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT*
SW14=0	NORMAL RUN
SW14=1	HOLD ON CURRENT CHARACTER

*THIS ADVANCE WILL ONLY TAKE PLACE AFTER THE CURRENT TEST HAS BEEN COMPLETED.

11. V T ' F I E L D C O N T R O L ' T E S T (2)

A. TEST DESCRIPTION

THIS TEST DISPLAYS THE 'VT' CHARACTER SET USING ALL FIVE 'FIELD CONTROL' MODES: NORMAL, BLINK, BOLD, BLANK AND UNDERLINE. THE TEST BREAKS THE CHARACTER SET INTO THREE

PARTS. EACH PART IS THEN DISPLAYED IN ALL '5' MODES.
THIS ENABLES THE USER TO VIEW EACH CHARACTER IN A STRAIGHT
LINE IN EVERY MODE.

B. CONSOLE SWITCH SETTING — FUNCTION

CONSOLE SWITCH SETTING	FUNCTION
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST

12. V T ' C S R P R E S E T ' T E S T (3)

****

A. TEST DESCRIPTION

THIS TEST CONSISTS OF FOUR (4) PARTS. EACH PART DISPLAYS
AN ENTIRE SCREEN OF THE 'VT' CHARACTER SET USING A 'CSR'
PRESET FUNCTION. THE SELECTED PRESET FUNCTIONS IN ORDER
ARE: BOLD, BLINK, BLANK, & UNDERLINE. AN IDENTIFYING
MESSAGE WILL PRECEDE EACH PART.

B. CONSOLE SWITCH SETTINGS — FUNCTION

CONSOLE SWITCH SETTINGS	FUNCTION
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST*
SW14=0	NORMAL RUN
SW14=1	HALT ON CURRENT PRESET

13. V T ' E N D O F L I N E ' T E S T (4)

****

A. TEST DESCRIPTION

THIS TESTS THAT THE 'VISIBLE END OF LINE' CHARACTER CAN
BE PLACED IN ANY OF THE '1024' CHARACTER POSITIONS ON THE
SCREEN. THIS IS DONE BY STARTING ALL '16' LINES WITH A
VISIBLE END OF LINE CHARACTER AND THEN INCREMENTING THESE
CHARACTERS ACROSS THE SCREEN. AS THE 'EOL' CHARACTER IS
MOVED AN ASTERISK IS DISPLAYED IN ITS PLACE.

B. CONSOLE SWITCH SETTINGS — FUNCTION

CONSOLE SWITCH SETTINGS	FUNCTION
SW11=0	NORMAL RUN
SW11=1	INHIBIT DISPLAY DELAY
SW12=0	NORMAL RUN

```
                        SW12=1                      ADVANCE TO NEXT TEST*

                        SW14=0                      NORMAL RUN
                        SW14=1                      HOLD CURRENT 'EOL' POSITION


                    * THIS ADVANCE WILL ONLY TAKE PLACE AFTER THE CURRENT TEST HAS
                      BEEN COMPLETED.




14.     V T  ' E N D  O F  S C R E E N '  T E S T (5)
        ********************************************


        A.  TEST DESCRIPTION
            ----------------

            THIS TEST TESTS THAT THE 'VISIBLE END OF SCREEN' CHARACTER
            CAN BE PLACED IN ANY OF THE '1024' CHARACTER POSITIONS ON
            THE SCREEN. THIS IS DONE BY FIRST PRE-LOADING THE DATA
            BUFFER WITH QUESTION MARKS (THESE WILL BE DISPLAYED IF
            THE EOS IS SKIPPED). THE 'EOS' CHARACTER IS THEN INCREMENTED
            THRU THIS BUFFER REPLACING THESE QUESTION MARKS BEHIND
            THE 'EOS' WITH DOTS.

        B.  CONSOLE SWITCH SETTINGS     FUNCTION
            -----------------------     --------
            SW11=0                      NORMAL RUN
            SW11=1                      INHIBIT DISPLAY DELAY

            SW12=0                      NORMAL RUN
            SW12=I                      ADVANCE TO NEXT TEST

            SW14=0                      NORMAL RUN
            SW14=1                      HOLD CURRENT 'EOS' POSITION


15.     V T  ' B L A N K  C O N T R O L '  T E S T (6)
        **********************************************


        A.  TEST DESCRIPTION
            ----------------

            THIS TEST TESTS THE 'BLANK' FIELD CONTROL LOGIC BY
            INCREMENTING THE 'BLANK' CONTROL CHARACTER THRU A BUFFER
            OF ASTERISK CHARACTERS.  THE TEST STARTS USING THE BLANK
            CONTROL CHARACTER AT THE START OF THE 'VT' BUFFER.  THE
            SCREEN SHOULD START OFF BLANK AND AS THE BLANK CONTROL
            CHARACTER IS INCREMENTED THRU THE 'VT' BUFFER ASTERISKS
            WILL START APPEARING ON THE SCREEN.  THE TEST WILL RESTART
            AFTER THE ENTIRE SCREEN HAS BEEN FILLED.
```

B. CONSOLE SWITCH SETTINGS	FUNCTION
SW11=0	NORMAL RUN
SW11=1	INHIBIT DISPLAY DELAY
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST
SW14=0	NORMAL RUN
SW14=1	HOLD CURRENT BLANK POSITION

16. V T ' A L I G N M E N T ' T E S T (7)

A. TEST DESCRIPTION

THIS TEST DISPLAYS A FULL SCREEN OF THE CHARACTER 'E' TO ENABLE A VISUAL ALIGNMENT OF THE 'VT'.

B. CONSOLE SWITCH SETTINGS

CONSOLE SWITCH SETTINGS	FUNCTION
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST

17. V T ' F O C U S ' T E S T (10)

A. TEST DESCRIPTION

THIS TEST DISPLAYS A FULL SCREEN OF ALTERNATE ASTERISKS AND ONE'S TO ENABLE A VISUAL FOCUS OF THE 'VT'.

B. CONSOLE SWITCH SETTINGS

CONSOLE SWITCH SETTINGS	FUNCTION
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST

18. V T ' W O R S T C A S E ' T E S T (11)

A. TEST DESCRIPTION

THIS TEST DISPLAYS A FULL SCREEN OF THE WORST CASE CHARACTER PATTERN '125 & 252'.

NOTE: CODE 125 WILL DISPLAY A "U" AND CODE 252 WILL DISPLAY EITHER A BLANK OR A SPECIAL TYPESETTING CHARACTER, THIS BEING DEPENDENT ON THE CHARACTER SET.

B. CONSOLE SWITCH SETTING — FUNCTION

CONSOLE SWITCH SETTING	FUNCTION
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT TEST

19. VT 'C U R S O R C O N T R O L' T E S T (12)

A. TEST DESCRIPTION

THIS TEST TESTS THAT THE CURSOR CONTROL CHARACTER CAN BE PLACED IN ANY OF THE '1024' CHARACTER POSITIONS ON THE SCREEN. THIS IS DONE BY DISPLAYING A 'CURSOR EOS' AND INCREMENTING IT THRU A PRE-LOADED BUFFER OF QUESTION MARKS. AS THE CURSOR CONTROL CHARACTER IS MOVED ALONG ASTERISKS ARE FILLED IN BEHIND THE CURSOR CHARACTER. WHEN THE TEST IS COMPLETE THE ENTIRE SCREEN SHOULD BE FILLED WITH ASTERISKS AND WITH NO QUESTION MARKS VISIBLE.

B. CONSOLE SWITCH SETTINGS — FUNCTION

CONSOLE SWITCH SETTINGS	FUNCTION
SW11=0	NORMAL RUN
SW11=1	INHIBIT DISPLAY DELAY
SW12=0	NORMAL RUN
SW12=1	ADVANCE TO NEXT*
SW14=0	NORMAL RUN
SW14=1	HOLD ON CURRENT CHARACTER

* THIS ADVANCE WILL ONLY TAKE PLACE AFTER THE CURRENT TEST HAS BEEN COMPLETED.

20. VT 'O D D A D R E S S' T E S T (13)

A. TEST DESCRIPTION

THIS TEST SIMPLY DISPLAYS A MESSAGE WHICH IS LOADED
FROM AN ODD STARTING ADDRESS. THE MESSAGE SHOULD
SAY: THIS MESSAGE IS BEING LOADED FROM AN ODD
STARTING ADDRESS. HOW DOES IT LOOK?

21. V T ' R U N A L L ' T E S T (14)

A. TEST DESCRIPTION

THIS TEST ENABLES TESTS '0-13,15' TO BE RUN IN ORDER.
IF A '4' WAS SELECTED AS THE TUBE NUMBER, THESE TESTS
WILL BE RUN SEQUENTIALLY ON ALL AVAILABLE TUBES.

22. A S C I I K B D ' L O G I C ' T E S T (15)

A. TEST DESCRIPTION

THE 'KBD' LOGIC TEST CONSISTS OF '19' SUBTESTS WHICH
EXERCISE AND DIAGNOSE THE 'KBD' LOGIC. EACH SUBTEST
IS LOOPED '100' TIMES TO TEST LOGIC RELIABILITY.
THESE TESTS ARE RUN AS A UNIT WHEN THE 'KBD' LOGIC TEST
IS SELECTED BUT THEY MAY ALSO BE SELECTED INDIVIDUALLY
BY THE 'FK/VT' SUBTEST SELECTOR (REFER TO SECTION 27).

THE 'KBD' LOGIC TEST DOES 'NOT' CHECKOUT THE LIGHT AND
SWITCH LOGIC. THIS LOGIC IS CHECKED BY THE 'KBD FUNCTION
LIGHT & SWITCH TEST' (SECTION 23).

B. RESTRICTIONS

THE 'VT' LOGIC MUST BE OPERATIONAL AS REQUESTS FOR
'KBD' KEYBOARD INPUTS ARE REQUESTED BY MESSAGES DISPLAYED
ON THE 'VT'. CARE SHOULD BE TAKEN THAT ONLY ONE KEY BE
STRUCK EACH TIME THIS REQUEST IS MADE TO AVOID FALSE
ERROR INDICATIONS.

C. LOGIC ERRORS

THERE ARE TWO ERROR REPORTING SCHEMES USED FOR REPORTING
'KBD' LOGIC ERRORS, (1) WITH TELETYPE AND (2) WITH PROGRAM
HALTS.

1. ON ENCOUNTERING A LOGIC ERROR (ALL DATA SWITCHES DOWN)
    THE FAILING SUBTEST NO., THE ERROR ADDRESS AND
    CONTENTS OF THE 'KBD' REGISTERS ARE TYPED OUT.

A. ERROR FORMAT

TST NO. PASS MA CSR DATA
A B C D E

WHERE:
A= FAILING SUBTEST NO.
B= PASS ON WHICH ERROR OCCURRED.
C= ERROR ADDRESS (MEMORY ADDRESS)
D= CONTENTS OF CONTROL & STATUS ADDRESS REGISTER
E= CONTENTS OF DATA ADDRESS REGISTER.

2. ON ENCOUNTERING A LOGIC ERROR (DATA SWI5 UP)
THE PROGRAM LOADS THE INTERNAL REGISTERS WITH
THE FOLLOWING ERROR DATA AND HALTS.

R0= FAILING SUBTEST NO.
R1= PASS ON WHICH ERROR OCCURRED.
R2= FAILING ERROR ADDRESS
R3= CONTENTS OF CONTROL & STATUS REGISTER ADDRESS
R4= CONTENTS OF THE DATA ADDRESS REGISTER.

D. CONSOLE SWITCH SETTINGS FUNCTION

CONSOLE SWITCH SETTINGS	FUNCTION
SW10=0	REQUEST KEYWARD INPUT
SW10=1	INHIBIT KEYWARD INPUT REQUEST
SW11=0	NORMAL RUN (100 PASSES/TEST)
SW11=1	SUPPRESS SUBTEST ITERATIONS
SW13=0	PRINT ERROR MESSAGES
SW13=1	SUPPRESS ERROR MESSAGES
SW14=0	NORMAL RUN
SW14=1	LOOP ON CURRENT SUBTEST
SW15=0	CONTINUE ON ERROR
SW15=1	HALT ON ERROR

23. A S C I I K B D ' L I G H T & S W I T C H ' T E S T (16)

A. TEST DESCRIPTION

THIS TEST IS USED TO CHECK OUT THE KBD FUNCTION LIGHT AND

SWITCH LOGIC. THIS IS AN OPERATOR INTERVENTION TEST
WHICH MONITORS THE FUNCTION KEYS. WHEN A FUNCTION KEY OR
KEYS ARE HELD DOWN ITS CORRESPONDING FUNCTION LIGHT IS LIT.
IF THE KEY IS HELD DOWN A SECOND TIME THE CORRESPONDING FUNCTION
LITE IS TURNED OFF. IT SHOULD BE A NOTED THAT THESE ARE
INTERVAL SWITCHES AND THE SIGNAL LEVEL WHICH INDICATES THAT
A KEY IS PRESSED WILL REMAIN TRUE FOR THE LENGTH OF TIME
THE KEY IS HELD DOWN. SINCE THERE IS NO WAY FOR THE
PROGRAM TO TELL IF A KEY IS BEING HELD DOWN OR IF IT HAS
BEEN STRUCK A SECOND TIME THE LIGHT DATA WILL BE COMPLI-
MENTED EVERY TIME THE SIGNAL IS PRESENT. TO ALLEVIATE THIS
PROBLEM THE PROGRAM ONLY READS THE SWITCHES EVERY '512'
INTERVAL TIME FLAGS. SO IF A SWITCH IS HELD DOWN IT
WILL BE TURNED ON AT THE BEGINNING OF THIS COUNT AND TURNED
OFF AT THE END OF THE COUNT IF THE SWITCH REMAINS DOWN.
TO SWITCH TO TESTING THE NEXT TUBE (IF RUNALL IS SELECTED)
TYPE ↑C ON THE TUBE BEING TESTED.

.24. A S C I I K B D 'C H A R A C T E R S E T' T E S T (17)

****

A. TEST DESCRIPTION

THIS TEST IS ALSO AN OPERATOR INTERVENTION TEST. IT IS USED
TO VERIFY THAT ALL CHARACTERS CAN BE TRANSMITTED BY THE
ASCII KEYBOARD. THE TEST REQUESTS THREE COMPLETE PASSES
(LOWER CASE, UPPER CASE AND CONTROL) TO BE TYPED IN BY
THE OPERATOR. A MESSAGE IS DISPLAYED ON THE 'VT' FOR THE
CASE TO BE INPUTED. EACH PASS MUST START IN THE UPPER LEFT
HAND CORNER OF THE KEYBOARD AND END WITH SPACE. AS EACH
CHARACTER IS TRANSMITTED FROM THE KEYBOARD IT IS VERIFIED
AGAINST A CHARACTER BUFFER AND DISPLAYED ON THE 'VT'.

B. ERRORS

IF THE TRANSMITTED CHARACTER CODE DOESN'T MATCH THE
EXPECTED CHARACTER CODE THE AUDIO 'BEEP' IS SOUNDED AND
THE RECEIVED CHARACTER IS DISPLAYED IN 'BOLD-UNDERLINED'.

C. CONSOLE SWITCH SETTINGS FUNCTION

SW14=0	NORMAL RUN
SW14=1	LOOP ON THE CURRENT CHARACTER
SW15=0	LOOP UNTIL CORRECT CODE RECEIVED
SW15=1	SKIP BAD CHARACTER

NOTE: FOR PROPER OPERATION SW15 SHOULD BE RE-SET TO '0'
AS SOON AS THE NEXT CHARACTER HAS BEEN INPUTED.

25. A S C I I K B D 'R E P E A T I B I L I T Y' T E S T (20)

A. TEST DESCRIPTION

THIS TEST IS ALSO AN OPERATOR INTERVENTION TEST. THE TEST HAS TWO FUNCTIONS. IT CAN BE USED AS A BACKUP FOR THE 'CHARACTER SET' TEST TO TEST ANYONE SELECTED CHARACTER CODE OR, SECOND, IT CAN BE USED TO TEST KEYBOARD REPEATIBILITY BETWEEN ANY TWO SELECTED CHARACTER CODES.

B. STARTING SEQUENCE

THE TEST VERIFIES UP TO TWO CHARACTERS RECEIVED FROM THE KEYBOARD AGAINST THE CHARACTER CODE(S) LOADED BY THE OPERATOR. THESE CHARACTER CODES ARE LOADED ONE OF TWO WAY. (1) VIA TELETYPE OR (2) VIA 'SWR' ON PROGRAM HALTS.

(25. CONT)

1. TYPE: NNN[,NNN] (CR) (I.E. 101,141 UPPER CASE A AND LOWER A)

   WHERE:
   NNN IS THE CODE(S) TO BE VERIFIED.

2. WHEN TEST IS SELECTED THE PROGRAM WILL HALT.
   LOAD THE CHARACTER CODE TO BE VERIFIED IN THE 'SWR'
   AND PRESS CONTINUE. A SECOND HALT WILL OCCUR.
   IF TWO CHARACTERS ARE TO BE VERIFIED
   LOAD THE CODE OF THE SECOND CHARACTER IN THE
   'SWR'. OTHERWISE JUST PRESS CONTINUE.

C. ERRORS

IF THE TRANSMITTED CHARACTER CODE DOESN'T MATCH THE EXPECTED CHARACTER CODE THE AUDIO 'BEEP' IS SOUNDED AND THE RECEIVED CHARACTER IS DISPLAYED IN 'BOLD-UNDERLINED'.

26. V T 2 0 S Y S T E M T E S T (21)

A. TEST DESCRIPTION

THIS TEST IS DESIGNED TO EXERCISE THE VT20 IN A SYSTEM
TYPE OF ENVIRONMENT. THE TEST UTILIZES THE ENTIRE VT20
SYSTEM (ALL TUBES) IN CONJUNCTION WITH A HOST COMPUTER.
THE HOST COMPUTER MAY BE EITHER A PDP-11 OR A PDP-8. THE
HOST IS TO BE CABLED VIA SERIAL LINE INTERFACE TO THE VT20
SYSTEM. ONE INTERFACE LINE IS REQUIRED FOR EACH UNIT.
THE HOST DIAGNOSTIC, EITHER MD-11-DZVTE(FOR A PDP-11
HOST W/DL11'S), MD-11-DZVTG (FOR A PDP-11 HOST W/DH11'S)
OR MD-D8-DIVTB (FOR A PDP-8 HOST W/KL8J'S) MUST BE LOADED
AND RUNNING BEFORE ATTEMPTING TO TRANSFER DATA VIA THIS TEST.

WHEN TEST 21 IS SELECTED, A MESSAGE REMINDER FOR SETTING UP
THE HOST DIAGNOSTIC WILL BE DISPLAYED. THIS MESSAGE
WILL THEN MOMENTARILY BE REPLACED WITH A TUBE STATUS
HEADER. IN THIS HEADER WILL BE DISPLAYED: NO. OF CHARACTERS
ON THE SCREEN, NO. OF CHARACTERS RECEIVED FROM THE
HOST, NO. OF BLOCKS TRANSFERRED AND NO. OF ERRORS ENCOUNTERED.
THIS MESSAGE ALSO INDICATES THAT THE TEST IS READY.

AS CHARACTERS ARE RECEIVED FROM THE ASCII KEYBOARD, THEY
THEY ARE TESTED FOR BEING PROGRAM CONTROL CHARACTERS,
AND IF NOT, THEY ARE TREATED AS DATA AND DISPLAYED. THE
PROGRAM WILL ACCEPT UP TO 384 DATA CHARACTERS OR 6
COMPLETE LINES.

B. DATA CREATION

DATA CAN BE AUTOMATICALLY GENERATED BY TYPING EITHER A (↑A)
OR A (↑W). THE (↑A) WILL ENABLE FOR A INCREMENTAL CHARACTER
SET OF THE CHARACTERS '40-177' TO BE CREATED UNTIL THE
SCREEN BUFFER IS FULL. A (↑W) WILL ENABLE FOR A WORST
CASE PATTERN OF '125-252' TO BE PRODUCED UNTIL THE SCREEN
BUFFER IS FILLED.

C. TRANSMITTING MODES

THERE ARE TWO MODES USED TO TRANSMIT DATA TO THE HOST
COMPUTER: SINGLE TRANSFER OR CONTINCUS TRANSFER.

A SINGLE TRANSFER IS INITIATED BY TYPING 'ALTMODE' AFTER
THE DATA HAS BEEN INPUTTED. ON RECIVING THE ALTMODE
THE INPUTED DATA IS TRANSFERED TO THE HOST COMPUTER.
THE DATA IS BUFFERED THERE AND TRANSMITTED BACK TO THE VT20
ON RECEIVING THE END OF PARAGRAPH. AS THE DATA IS
RECEIVED IT IS DISPLAYED BELOW THE ORIGINAL DATA.
AFTER EXAMINING THE DATA THE OPERATOR MAY TYPE '↑C' TO

CLEAR THE SCREEN AND RESTART THE TEST OR START A CONTIN-
OUS TRANSFER.

A CONTINOUS TRANSFER IS INITIATED AFTER THE DATA HAS BEEN
INPUTTED VIA TYPING A '↑T'. THIS MODE OPERATES IN THE
SAME MANNER AS A SINGLE TRANSFER EXCEPT UPON RECEVING THE
DATA BACK FROM THE HOST COMPUTER IT IS AUTOMATICALLY RE-
TRANSMITTED. TYPING A SECOND '↑T' WILL STOP THE CONTINOUS
TRANSFER AFTER THE CURRENT TRANSFER HAS BEEN COMPLETED.

THIS DATA IS SENT IN THE FORMAT OF: 4 NULL CHARACTERS
(000), A START CODE CHARACTER (377), DATA (ORIGINATED
BY THE USER), AND A END OF PARAGRAPH CHARACTER (14).
ON RECEIPT OF THE 'EOP' (END OF PARAGRAPH) CHARACTER,
THE HOST COMPUTER WILL TRANSMIT THE DATA, IN THE FORMAT
IT WAS RECEIVED, BACK TO THE VT20. ON RECEIPT OF THE START
CODE CHARACTER (377), THE VT20 DISPLAYS A UP-ARROW(↑) TO INDICATE
THE DATA HAS RETURNED.

EACH CHARACTER IS VERIFIED AS IT IS RECEIVED AGAINST
CORRESPONDING CHARACTER TRANSMITTED. IF A RECEIVED
CHARACTER DOESN'T MATCH THE CORRESPONDING TRANSMITTED
CHARACTER. IT IS DISPLAYED AS 'BOLD-UNDERLINED'.
THIS ERROR WOULD ALSO BE RECORDED IN THE ERROR HEADER.

ERROR CHECKING CAN BE INHIBITED BY SETTING DATA 'SW00' TO
A '1'(UP). ALL DATA THEN RECEIVED FROM THE HOST COMPUTER
WILL SIMPLY BE DISPLAYED. THIS SWITCH SHOULD ONLY
BE SET WHEN THE HOST COMPUTER IS ORGINATING DATA USING
THE SEND MODE (REFER TO THE HOST DIAGNOSTIC).

THERE IS ALSO ANOTHER OPTION AVAILABLE TO THE USER
RUNNING WITH MD-11-DZVTG.

SW00 & SW01 CAN BE SET. THIS WILL ENABLE THE DATA
BEING RECEIVED FROM THE HOST WILL BE DISPLAYED IN
OCTAL FORM. THE DATA WILL ALSO BE ECHOED BACK TO THE
HOST. THIS OPTION IS USED WHEN RUNNING THE HOST VERIFY
TEST. REFER TO MD-11-DZVTG FOR FULL DETAILS OF THE TEST.

IF A TRANSFER IS INITIATED AND FOR ONE REASON OR ANOTHER IT
FAILS TO RETURN BACK TO THE VT20, THE TEST WILL APP-
EAR HUNG SINCE NO KEYBOARD COMMANDS WILL BE RECOGNIZED. IN THIS
CASE AND ONLY IN THIS CASE TYPE '↑E' TO ESCAPE THE TRANSFER
MODE AND RESTART.

IF '↑E' IS TYPED IN THE MIDDLE OF A TRANSFER, IT
WILL CAUSE MULTIPLE RECEIVER ERRORS.

C. CONSOLE SWITCH SETTINGS FUNCTION

CONSOLE SWITCH SETTINGS	FUNCTION
SW15=0	*CONTINUE TRNANSFERING ON ERROR
SW15=1	HALT TRANSFERING ON ERROR
SW00=0	VERIFY RECEIVER DATA
SW00=1	**INHIBIT VERIFING RECEIVER DATA

```
                        SW00 & SW01=1          ***DISPLAY RECEIVED CHARACTER IN
                                                  OCTAL FORM AND ECHO IT BACK TO
                                                  THE HOST.


                         * SW15 APPLIES TO CONTINOUS TRANSFERS ONLY AND THE
                           HALT WILL OCCUR AT THE END OF THE CURRENT TRANSFER.
                        ** THIS SWITCH MUST BE SET IF THE HOST IS TRANSMITTING
                           DATA USING THE SEND MODE.
                       *** THIS OPTION SHOULD ONLY BE USED WHEN RUNNING THE
                           VERIFY TEST OF MD-11-DZVTA.


(26. CONT)

        D.  ASCII KRB CONTROL SWITCHES
            --------------------------

              KEY                       FUNCTION
              ---                       --------

            RUBOUT              DELETE LAST CHARACTER.
            ALTMODE             INITIATE SINGLE TRANSFER.
            ↑A (CTRL A)         GENERATE INCREMENTAL CHARACTER SET.
            ↑T (CTRL T)         ENABLE/DISABLE CONTINUOUS TRANS.
            ↑W                  GENERATE WORST CASE CHARACTER PATTERN
            ↑C (CTRL C)         CLEAR SCREEN AND RESTAPT
            ↑E (CTRL E)         *ESCAPE AND RESTART

            * USE WITH CAUTION AS NOTED ABOVE.
        E.  ERRORS
            ------

            TO INDICATE TRANSMITTER, RECEIVER AND KEYBOARD ERRORS THE
            FOLLOWING CHARACTERS WILL BE DISPLAYED USING THE BLINK MODE.

            CHARACTER                   MEANING
            ---------                   -------

                K               ASCII KBD ERROR FLAG SET (OVERRUN ERROR)
                T               ILLEGAL TRANSMITTER INTERRUPT
                R               ILLEGAL RECEIVER INTERRUPT
                O               DL11 RECEIVER 'OVERRUN' ERROR
                F               DL11 RECEIVER 'FRAMING' ERROR
                P               DL11 RECEIVER 'PARITY' ERROR


    27.     V T / K B D  S U B T E S T  S E L E C T O R (22)
            ***********************************************


            A.  TEST DESCRIPTION
                ----------------
```

THIS TEST IS DESIGNED TO ALLOW THE OPERATOR TO LOOP
ON ANY SELECTED 'KBD' OR 'VT' LOGIC SUBTEST WITHOUT
RUNNING THE ENTIRE LOGIC TEST

B. STARTING SEQUENCE

THIS TEST ACCEPTS AN OCTAL ADDRESS OF A 'SUBTEST' TO BE
EXECUTED. THE PROGRAM THEN SETS UP THE PROPER PARAMETERS
AND EXECUTES THE SELECTED 'SUBTEST'. THE SUBTEST ADDRESS
IS LOADED ONE OF TWO WAYS, (1) VIA TELETYPE OR (2) VIA
'SWR' ON PROGRAM HALT.

1. WHEN SELECTED THE PROGRAM WILL TYPE THE MESSAGE 'TEST ADDR.?'.
   THE OPERATOR SHOULD THEN TYPE IN THE 'SCOPE ADDRESS' OF THE
   TEST TO BE EXECUTED.

2. WHEN THE TEST IS SELECTED THE PROGRAM WILL HALT. LOAD
   THE ADDRESS OF THE TEST TO BE EXECUTED INTO THE
   'SWR' AND PRESS CONTINUE.

C. RESTRICTIONS

DATA "SW11" MUST BE DOWN [0] TO INHIBIT THE PROGRAM FROM
ESCAPING THE SELECTED TEST AND CONTINUING THRU THE LOGIC
TEST.

D. ERRORS

REFER TO THE RESPECTIVE 'KBD' OR 'VT' ERROR SECTION.

29. EXECUTION TIMES

TEST	PASS TIME	NOTES
0	7.5 MINS	
1	4 MINS	
2	N/A	*
3	30 SECS	
4	25 SECS	
5	6 MINS	
6	6 MINS	
7	N/A	*
10	N/A	*
11	N/A	*
12	6 MINS	
13	N/A	*
14	TOTAL OF ABOVE + TEST 15	

```
                                        15      75 SECS
                                        16      N/A
                                        17      N/A
                                        20      N/A
                                        21      N/A

                                        * IN RUN-ALL (TEST 14), THESE RUN FOR 3 SECS EACH.


                                30.     L I S T I N G
                                        *************

                                        %

                                .TITLE  VT20 SYSTEM DIAGNOSTIC TEST PROGRAM
                                :MAINDEC-11-DBVTA-C-D
                                ;COPYRIGHT SEPTEMBER 5, 1974,1975
                                :REVISED: APRIL 25, 1975
                                ;DIGITAL EQUIPMENT CORP. MAYNARD MASS. 01754
                                :PROGRAMMER: EARL L. BOUSE/ED BADGER

                                ;SWITCH REGISTER DEFINITIONS AND FUNCTIONS:

100000                          SW15=100000                     ;=1, HALT ON ERROR
040000                          SW14=40000                      ;=1, LOOP ON CURRENT TEST
020000                          SW13=20000                      ;=1, SUPPRESS ERROR TYPEOUT
010000                          SW12=10000                      ;=1, HALT SHIFTING ON SHIFT ERROR.
004000                          SW11=4000                       ;=1, SUPPRESS 'SUBPROGRAM' ITERATIONS
002000                          SW10=2000                       ;=1, INHIBIT ASCII KRB MANUAL INTERVENTION
001000                          SW09=1000
000400                          SW08=400
000200                          SW07=200
000100                          SW06=100
000040                          SW05=40
000020                          SW04=20
000010                          SW03=10
000004                          SW02=4
000002                          SW01=2
000001                          SW00=1

                                ;REGISTER DEFINITIONS

000000                          R0=%0
000001                          R1=%1
000002                          R2=%2
000003                          R3=%3
000004                          R4=%4
000005                          R5=%5
000006                          SP=%6
000007                          PC=%7

                                ;INSTRUCTIONS DEFINITIONS

000004                          FKCSR4=%4
000004                          VTCSR4=%4
000005                          VTMAR5=%5
```

```
1224            005726                  POP1SP=5726
1225            005746                  PUSH1SP=5746
1226            024646                  PUSH2SP=24646
1227            022626                  POP2SP=22626
1228            000240                  NOP=240
1229            000002                  X=2
1230            000002                  Y=2
1231
1232                                    ;LOAD TRAP CATCHER INTO LOC'S 0-1000
1233            000000                          .=0
1234                                    .REPT   200
1235                                            .+2
1236                                            4
1237                                    .ENDR
1238            000020                          .=20
1239  000020    016456                          ERTRAP                  ;ERROR TRAP REPORTER ROUTINE
1240            000024                          .=24
1241  000024    016014                          PWRFAL                  ;POWER FAIL HANDLER
1242  000026    000340                          340
1243            000030                          .=30
1244  000030    001200                          EMTSRV                  ;EMT TRAP, EMT DISPATCH SERVICE
1245  000032    000340                          340
1246  000034    017442                          FKERR                   ;TRAP TRAP, LOGIC ERROR TRAP
1247  000036    000340                          340
1248            000060                          .=60
1249  000060    014350                          XTTYIN                  ;TELEPRINTER KEYBOARD ROUTINE
1250            000200                          .=200
1251  000200    000137  001624                  JMP     INITTY          ;PROGRAM STARTING ADDR. IF TTY AVAILABLE.
1252  000204    000137  001634                  JMP     NOTTY           ;PROGRAM STARTING ADDR. IN 'NO' TTY.
1253  000210    000137  002136                  JMP     INITBA          ;PROGRAM RE-START ADDR. TO SELECT NEW UNIT.
1254
1255                                    ;TRAP EQUIVALENCE TABLE:
1256
1257            104400                  ERROR=TRAP                      ;LOGIC TEST ERROR ROUTINE
1258            104000                  DISPLAY=EMT                     ;'VT' MESSAGE DISPLAY ROUTINE
1259            104001                  SCOPE=EMT+1                     ;LOGIC TEST SCOPE SUBROUTINE
1260            104002                  SAVREG=EMT+2                    ;SUBROUTINE TO SAVE 'R0-R5' ON STACK
1261            104003                  GETREG=EMT+3                    ;SUBROUTINE TO GET 'R0-R5' FROM STACK
1262            104004                  DELAY=EMT+4                     ;3 SEC. DISPLAY DELAY
1263            104005                  EOSBUF=EMT+5                    ;SUBROUTINE TO LOAD VT BUFER W/ 'EOS'
1264            104006                  ENDTST=EMT+6                    ;SUBROUTINE TO CHECK DATA SW.'S
1265            104007                  SETLNE=EMT+7                    ;SUBROUTINE TO SET UP A LINE TO CHAR.'S
1266            104010                  LDLINE=EMT+10                   ;SUBROUTINE TO LOAD A 'VT' BUFFER LINE
1267            104011                  PRELOAD=EMT+11                  ;SUBROUTINE TO PRE-LOAD 'VT' DATA BUFFER
1268            104012                  PRINT=EMT+12                    ;SUBROUTINE TO PRINT ASCII MESSAGES.
1269            104013                  TTYIN=EMT+13                    ;SUBROUTINE TO INPUT VIA KEYBOARD
1270            104014                  PRTOCT=EMT+14                   ;SUBROUTINE TO PRINT A 6 DIGIT OCTAL NO.
1271            104015                  ASEMBL=EMT+15                   ;SUBROUTINE TO ASSEMLE CHARACTERS INTO OCTAL VALUE
1272            104016                  SPACE=EMT+16                    ;SUBROUTINE TO PRINT SPACES
1273            104017                  TSTTKS=EMT+17                   ;SUBROUTINE TO TEST FOR KEYBOARD FLAGS
1274            104020                  DELAYL=EMT+20                   ;SUBROUTINE TO SETUP A LONG DISPLAY DELAY
1275            104021                  WAITSY=EMT+21                   ;SUBROUTINE TO WAIT FOR VERTICAL SYNC
1276            104022                  NULL=EMT+22                     ;SUBROUTINE TO TRANSMIT A NULL PRINTER CHAR.
1277            104023                  THEND=EMT+23                    ;SUBROUTINE ENTERED AT END OF DISP. TEST
1278            104024                  BINDEC=EMT+24                   ;SUBROUTINE TO CONVERT 'BCD' TO DECIMAL
1279
```

```
                                ;*****************************************************************************
                                ;EMT DISPATCH SERVICE ROUTINE
                                ;ARGUMENT OF EMT IS EXTRACTED AND USED AS OFFSET TO OBTAIN POINTER
                                ;TO THE SELECTED SUBROUTINE.
                                ;*****************************************************************************

                        001200                  .=1200
001200  011646          EMTSRV: MOV     (SP),-(SP)      ;GET PC FOR TO RETURN
001202  162716  000002          SUB     #2,(SP)         ;PC OF EMT
001206  017616  000000          MOV     @(SP),(SP)      ;GET EMT
001212  005716                  TST     (SP)            ;IS EMT VALID?
001214  001001                  BNE     EMTOK
001216  000000                  HALT                    ;INVALID EMT
001220  006316          EMTOK:  ASL     (SP)            ;MULTIPLY EMT ARG BY '2'
001222  042716  177001          BIC     #177001,(SP)    ;CLEAR UNWANTED BITS
001226  062716  001240          ADD     #EMTTAB,(SP)    ;POINTER TO SUBROUTINE ADDRESS
001232  017616  000000          MOV     @(SP),(SP)      ;SUBROUTINE ADDRESS
001236  000136                  JMP     @(SP)+          ;GO TO SUBROUTINE

                                :EMT DISPATCH TABLE

001240  017246          EMTTAB: VTMES                   ;'VT' MESSAGE DISPLAY ROUTINE
001242  020266                  SCOPEC                  ;LOGIC TEST SCOPE ROUTINE
001244  016122                  XSAVRG                  ;SUBROUTINE TO SAVE 'R0-R5' ON STACK
001246  016176                  XGETRG                  ;SUBROUTINE TO GET 'R0-R5' FROM STACK
001250  015540                  XDELAY                  ;SUBROUTINE TO SETUP 3 SEC. DISPLAY DELAY
001252  015656                  XEOSBF                  ;SUBROUTINE TO LOAD VT BUFFER W/ VISIBLE 'EOS'
001254  015732                  XENDT                   ;SUBROUTINE TO CHECK DATA SW.'S
001256  017024                  XSTLNE                  ;SUBROUTINE TO SET UP A 'VT' BUFFER LINE
001260  015632                  XLDLNE                  ;SUBROUTINE TO LOAD A 'VT' BUFFER LINE
001262  016252                  XPRELD                  ;SUBROUTINE TO PRE-LOAD THE DATA BUFFER
001264  014754                  TYPMES                  ;SUBROUTINE TO PRINT ASCII MESSAGES.
001266  014350                  XTTYIN                  ;SUBROUTINE TO INPUT VIA KEYBOARD
001270  015056                  OCTPRT                  ;SUBROUTINE TO PRINT A 6 DIGIT OCTAL NO.
001272  017104                  XASEMB                  ;SUBROUTINE TO ASSEMLE CHARACTERS INTO OCTAL VALUE
001274  014276                  XSPACE                  ;SUBROUTINE TO PRINT SPACES
001276  014330                  TKSFLG                  ;SUBROUTINE TO TEST FOR KEYBOARD FLAGS
001300  015550                  XDLAYL                  ;SUBROUTINE TO SET UP LONG DISPLAY DELAY
001302  015476                  XWAITS                  ;SUBROUTINE TO WAIT FOR VERT. SYNC
001304  017360                  XNULL                   ;SUBROUTINE TO TRANSMIT A 'NULL' CHAR.
001306  015726                  XTHEND
001310  016546                  XBINDEC                 ;SUBROUTINE TO CONVERT 'BCD' TO DECIMAL

                                ;REGISTER ADDRESSES

001312  177776          PSW:    177776                  ;ADDRESS OF PROCESSOR STATUS REG.
001314  177560          TKS:    177560                  ;ADDRESS OF KEYBOARD STATUS REG.
001316  177562          TKB:    177562                  ;   "      "        "     BUFFER  "
001320  177564          TPS:    177564                  ;   "      "  PRINTER STATUS REG.
001322  177566          TPB:    177566                  ;   "      "  PRINTER BUFFER REG.
001324  177570          SWR:    177570                  ;   "      "  SWITCH REG.
001326  177571          SWR0:   177571                  ;   "      "     "        " HIGH BYTE
                        ;'FK' UNIT 0'S REGISTER ADDRESSES.

001330  170330          FKOLDB: 170330                  ;ADDRESS OF UNIT 0'S 'LIGHT DATA B'
001332  170332          FKOLDA: 170332                  ;ADDRESS OF UNIT 0'S 'LIGHT DATA A'
```

```
1336  001334  170334          FKOCSR: 170334                  ;ADDRESS OF UNIT 0'S 'CONTROL & STATUS' REG.
1337  001336  170336          FKODAT: 170336                  ;ADDRESS OF UNIT 0'S 'KEYBOARD DATA' REG.
1338  001340  000320          FKOINT: 320                     ;ADDRESS OF UNIT 0'S 'INTERRUPT VECTOR'
1339  001342  000322          FKOLVL: 322                     ;ADDRESS OF UNIT 0'S 'INTERRUPT LEVEL'
1340
1341                          ;'FK' UNIT 1'S REGISTER ADDRESSES.
1342
1343  001344  170340          FK1LDB: 170340                  ;ADDRESS OF UNIT 1'S 'LIGHT DATA B'
1344  001346  170342          FK1LDA: 170342                  ;ADDRESS OF UNIT 1'S 'LIGHT DATA A'
1345  001350  170344          FK1CSR: 170344                  ;ADDRESS OF UNIT 1'S 'CONTROL & STATUS' REG.
1346  001352  170346          FK1DAT: 170346                  ;ADDRESS OF UNIT 1'S 'KEYBOARD DATA' REG.
1347  001354  000324          FK1INT: 324                     ;ADDRESS OF UNIT 1'S 'INTERRUPT VECTOR'
1348  001356  000326          FK1LVL: 326                     ;ADDRESS OF UNIT 1'S 'INTERRUPT LEVEL'
1349
1350                          ;'FK' UNIT 2'S REGISTER ADDRESSES
1351
1352  001360  170360          FK2LDB: 170360
1353  001362  170362          FK2LDA: 170362
1354  001364  170364          FK2CSR: 170364
1355  001366  170366          FK2DAT: 170366
1356  001370  000350          FK2INT: 350
1357  001372  000352          FK2LVL: 352
1358
1359                          ;'FK' UNIT 3'S REGISTER ADDRESSES
1360
1361  001374  170370          FK3LDB: 170370
1362  001376  170372          FK3LDA: 170372
1363  001400  170374          FK3CSR: 170374
1364  001402  170376          FK3DAT: 170376
1365  001404  000354          FK3INT: 354
1366  001406  000356          FK3LVL: 356
1367
1368                          ;'VT' UNIT 0'S REGISTER ADDRESSES.
1369
1370  001410  170300          VTOCAR: 170300                  ;ADDRESS OF UNIT 0'S CURSOR ADDRESS REG.
1371  001412  170302          VTOSAR: 170302                  ;ADDRESS OF UNIT 0'S STATING ADDRESS REG.
1372  001414  170304          VTOCSR: 170304                  ;ADDRESS OF UNIT 0'S COMMAND & STATUS REG.
1373  001416  170306          VTOMAR: 170306                  ;ADDRESS OF UNIT 0'S MAINTENANCE REG.
1374  001420  000330          VTOINT: 330                     ;ADDRESS OF UNIT 0'S INTERRUPT VECTOR
1375  001422  000332          VTOLVL: 332                     ;ADDRESS OF UNIT 0'S INTERRUPT LEVEL
1376
1377                          ;'VT' UNIT 1'S REGISTER ADDRESSES.
1378
1379  001424  170310          VT1CAR: 170310                  ;ADDRESS OF UNIT 1'S CURSOR ADDRESS REG.
1380  001426  170312          VT1SAR: 170312                  ;ADDRESS OF UNIT 1'S STATING ADDRESS REG.
1381  001430  170314          VT1CSR: 170314                  ;ADDRESS OF UNIT 1'S COMMAND & STATUS REG.
1382  001432  170316          VT1MAR: 170316                  ;ADDRESS OF UNIT 1'S MAINTENANCE REG.
1383  001434  000340          VT1INT: 340                     ;ADDRESS OF UNIT 1'S INTERRUPT VECTOR
1384  001436  000342          VT1LVL: 342                     ;ADDRESS OF UNIT 1'S INTERRUPT LEVEL
1385
1386                          ;'VT' UNIT 2'S REGISTER ADDRESSES
1387
1388  001440  170320          VT2CAR: 170320
1389  001442  170322          VT2SAR: 170322
1390  001444  170324          VT2CSR: 170324
1391  001446  170326          VT2MAR: 170326
```

```
1392  001450  000360                VT2INT: 360
1393  001452  000362                VT2LVL: 362
1394
1395                                ;'VT' UNIT 3'S REGISTER ADDRESSES
1396
1397  001454  170350                VT3CAR: 170350
1398  001456  170352                VT3SAR: 170352
1399  001460  170354                VT3CSR: 170354
1400  001462  170356                VT3MAR: 170356
1401  001464  000370                VT3INT: 370
1402  001466  000372                VT3LVL: 372
1403
1404  001470  170330                FKLDB:  170330
1405  001472  170332                FKLDA:  170332
1406  001474  170334                FKCSR:  170334          ;TEMPORARY STORAGE OF FK ADDRESSES
1407  001476  170336                FKDATA: 170336
1408  001500  000320                FkINT:  320
1409  001502  000322                FKLVL:  322
1410
1411  001504  170300                VTCAR:  170300
1412  001506  170302                VTSAR:  170302
1413  001510  170304                VTCSR:  170304          ;TEMPORARY STORAGE OF VT ADDRESSES
1414  001512  170306                VTMAR:  170306
1415  001514  000330                VTINT:  330
1416  001516  000332                VTLVL:  332
1417
1418                                ;DL11 REGISTER ADDRESSES
1419
1420  001520  175610                RCSR0:  175610          ;ADDRESS OF UNIT 0'S DL11 REC. CSR
1421  001522  175620                RCSR1:  175620          ;ADDRESS OF UNIT 1'S DL11 REC. CSR
1422  001524  175630                RCSR2:  175630          ;ADDRESS OF UNIT 2'S DL11 REC. CSR
1423  001526  175640                RCSR3:  175640          ;ADDRESS OF UNIT 3'S DL11 REC. CSR
1424  001530  175612                RBUF0:  175612          ;ADDRESS OF UNIT 0'S DL11 REC. BUFFER
1425  001532  175622                RBUF1:  175622          ;ADDRESS OF UNIT 1'S DL11 REC. BUFFER
1426  001534  175632                RBUF2:  175632
1427  001536  175642                RBUF3:  175642
1428  001540  000300                RINT0:  300             ;ADDRESS OF UNIT 0'S REC. VECTOR
1429  001542  000302                RLVL0:  302
1430  001544  000310                RINT1:  310             ;ADDRESS OF UNIT 1'S REC. VECTOR
1431  001546  000312                RLVL1:  312
1432  001550  000400                RINT2:  400
1433  001552  000402                RLVL2:  402
1434  001554  000410                RINT3:  410
1435  001556  000412                RLVL3:  412
1436
1437  001560  175614                XCSR0:  175614          ;ADDRESS OF UNIT 0'S TRANS. CSR
1438  001562  175624                XCSR1:  175624          ;ADDRESS OF UNIT 1'S DL11 TRANS. CSR
1439  001564  175634                XCSR2:  175634
1440  001566  175644                XCSR3:  175644
1441  001570  175616                XBUF0:  175616          ;ADDRESS OF UNIT 0'S DL11 TRANS. BUFFER
1442  001572  175626                XBUF1:  175626          ;ADDRESS OF UNIT 1'S DL11 TRANS. BUFFER
1443  001574  175636                XBUF2:  175636
1444  001576  175646                XBUF3:  175646
1445  001600  000304                XINT0:  304             ;ADDRESS OF UNIT 0'S DL11 TRANS. VECTOR
1446  001602  000306                XLVL0:  306
1447  001604  000314                XINT1:  314             ;ADDRESS OF UNIT 1'S DL11 TRANS. VECTOR
```

```
                                XLVL1:  316
                                XINT2:  404
                                XLVL2:  406
                                XINT3:  414
                                XLVL3:  416

                                ;DEFINITIONS OF 'VT FIELD CONTROL' FUNCTIONS

                        000370  UNLINE=370                      ;UNDERLINE PRESET
                        000364  BOLD=364                        ;BOLD PRESET
                        000361  BLANK=361                       ;BLANK PRESET
                        000362  BLINK=362                       ;BLINK PRESET
                        000200  AUDIO=200                       ;AUDIO PRESET
                        000012  EOL=12                          ;END OF LINE
                        000014  EOP=14                          ;END OF PARAGRAPH
                        000031  EOS=31                          ;END OF SCREEN
                        000360  CLRFLD=360                      ;CLR FIELD CONTROL
                        000212  VISEOL=212                      ;VISIBLE END OF LINE
                        000214  VISEOP=214                      ;VISIBLE END OF PARAGRAPH
                        000231  VISEOS=231                      ;VISIBLE END OF SCREEN

001620  000000          FCSET:  000000
001622  000000          FCSET1: 000000

                                ;DEFINITIONS OF 'MAINTENANCE' FUNCTIONS ( SELECTED VIA THE 'CSR')

                        000000  CSR=000                         ;COMMAND & STATUS REG.
                        000002  DAR=2                           ;DATA ADDRESS REG.
                        000004  SRIDE=4                         ;SHIFT REG. INPUT DATA (EVEN)
                        000006  SRODE=6                         ;SHIFT REG. OUTPUT DATA (EVEN)
                        000010  CAR=10                          ;CURSOR ADDRESS REG.
                        000012  SAR=12                          ;START ADDRESS REG.
                        000014  SRIDO=14                        ;SHIFT REG. INPUT DATA (ODD)
                        000016  SRODO=16                        ;SHIFT REG. OUTPUT DATA (ODD)

                                ;*****************************************************************
                                ;TEST INITIALIZATION ROUTINE. IF THE PROGRAM IS STARTED AT LOCATION
                                ;'200' THE PROGRAM IS SET UP TO RUN VIA A TELETYPE MONITOR. IF THE PRO-
                                ;GRAM IS  STARTED AT LOCATION '204' THE PROGRAM USES PROGRAM 'HALTS'
                                ;TO COLLECT NEEDED INFORMATION AND TO REPORT ERRORS.
                                ;*****************************************************************

001624  012737  000001  024160  INITTY: MOV     #1,TTYSWH       ;TTY AVAILABLE, SET SOFTWARE SW.
001632  000402                  BR      .+6
001634  005037  024160          NOTTY:  CLR     TTYSWH          ;CLR TTY AVAILABLE SOFTWARE SWITCH.
001640  012777  000340  177444          MOV     #340,@PSW
001646  013706  024136                  MOV     STACK,SP        ;INIT STACK POINTER=1000
001652  000240                          NOP
001654  000240                          NOP
001656  000240                          NOP                     ;INITIALIZATION SWITCH TEST REMOVED TO ALLOW RE-SELECT
001660  000240                          NOP
001662  000240                          NOP
001664  012737  001702  000004  1$:     MOV     #CORSIZ,@#4     ;RESET TIMEOUT
001672  012701  020000                  MOV     #20000,R1       ;TEST CORE SIZE
001676  005721                          TST     (R1)+
001700  000776                          BR      .-2
```

```
1504  001702  162701  002000          CORSIZ: SUB     #2000,R1        ;SAVE LAST 1K FOR 'VT' TEST
1505  001706  010137  024144                  MOV     R1,MEMSIZ
1506  001712  012737  000006  000004          MOV     #6,@#4          ;RESET TIMEOUT TO 'ERTRAP'
1507  001720  012737  000004  000006          MOV     #4,@#6
1508  001726  005737  024160          INITB:  TST     TTYSWH          ;TTY AVAILABLE?
1509  001732  001106                  BNE     INITC           ;YES, INIT VIA TTY.
1510  001734  000000                  HALT                    ;SEE IF FORIEGN CHAR SET EXISTS
1511  001736  000240                  NOP                     ;BY EXAMINING THE SWITCH REGISTER
1512  001740  017737  177360  001620          MOV     @SWR,FCSET      ;IF CLEAR THEN NOT FORIEGN SET
1513  001746  000000                  HALT                    ;GET # OF TUBES FROM SWITCHES
1514  001750  000240                  NOP
1515  001752  017737  177346  024142          MOV     @SWR,   UNITNO
1516  001760  042737  177770  024142          BIC     #177770,UNITNO
1517  001766  005777  177332                  TST     @SWR            ;IF BIT 15 IN SWR IS SET THEN
1518  001772  100402                  BMI     4$              ;LET USER INPUT DEVICE ADDRESSES
1519  001774  000137  002436                  JMP     INITCB          ;DEFAULT DEVICES & VECTORS REQUESTED
1520
1521  002000  013700  024142          4$:     MOV     UNITNO,R0
1522  002004  012705  001330                  MOV     #FKOLDB,R5
1523  002010  000000          1$:     HALT                    ;HALT FOR FK VECTOR
1524  002012  017737  177306  001500          MOV     @SWR,FKINT
1525  002020  000240                  NOP
1526  002022  000000                  HALT                    ;HALT FOR FK ADDR.
1527  002024  017704  177274                  MOV     @SWR,R4
1528  002030  004737  002700                  JSR     PC,SETED
1529  002034  005300                  DEC     R0
1530  002036  001364                  BNE     1$
1531
1532  002040  012705  001410                  MOV     #VTOCAR,R5
1533  002044  013700  024142                  MOV     UNITNO,R0
1534  002050  000000          2$:     HALT                    ;HALT FOR VT VECTOR
1535  002052  017737  177246  001500          MOV     @SWR,FKINT
1536  002060  000240                  NOP
1537  002062  000000                  HALT                    ;HALT FOR VT ADDR.
1538  002064  017704  177234                  MOV     @SWR,R4
1539  002070  004737  002700                  JSR     PC,SETED
1540  002074  005300                  DEC     R0
1541  002076  001364                  BNE     2$
1542  002100  013700  024142                  MOV     UNITNO,R0
1543  002104  012702  001540                  MOV     #RINTO,R2
1544  002110  012705  001600                  MOV     #XINTO,R5
1545  002114  000000          3$:     HALT                    ;HALT FOR DL11 VECTOR
1546  002116  017703  177202                  MOV     @SWR,R3
1547  002122  012322                  MOV     (3)+,(2)+
1548  002124  012322                  MOV     (3)+,(2)+
1549  002126  012325                  MOV     (3)+,(5)+
1550  002130  012325                  MOV     (3)+,(5)+
1551  002132  005300                  DEC     R0
1552  002134  001367                  BNE     3$
1553  002136  000000          INITBA: HALT                    ;GET TUBE NO. FROM SWITCHES
1554  002140  017737  177160  024140          MOV     @SWR,UNITFG     ;SAVE TUBE NO.
1555  002146  000543                  BR      INITD
1556  002150  005737  024172          INITC:  TST     MTRSWH          ;TYPE HEADER
1557  002154  001002                  BNE     .+6             ;NO, ITS BEEN TYPED
1558  002156  104012                  PRINT
1559  002160  021023                  TITLE                   ;PRINT PROGRAM HEADER
```

```
1560  002162  104012                        PRINT                   ;SEE IF FORIEGN CHARACTER SET EXISTS
1561  002164  021445                        MFCP
1562  002166  104015                        ASEMBL
1563  002170  042703  177776                BIC     #177776,R3
1564  002174  010337  001620                MOV     R3,FCSET
1565  002200  104012                        PRINT                   ;REQUEST NO. OF TUBES
1566  002202  024006                        MED1
1567  002204  104015                        ASEMBL                  ;GET INPUT
1568  002206  010337  024142                MOV     R3,UNITNO
1569  002212  104012  021165                PRINT,  MES1A           ;TEXT 'ARE DEFAULT ADDR. & VECT. OK?'
1570  002216  104015                        ASEMBL                  ;WAIT FOR HIS ANSWER
1571  002220  005703                        TST     R3              ;WAS 'CR' TYPED?
1572  002222  001505                        BEQ     INITCB          ;YES, USE DEFAULT ADDRESSES
1573  002224  022703  000001                CMP     #1,R3           ;WAS 'Y' TYPED?
1574  002230  001502                        BEQ     INITCB          ;YES, IT SURE WAS
1575  002232  012705  001330        INITAC: MOV     #FKOLDB,R5      ;NO, LET USER DEFINE HIS OWN ADDRESSES
1576  002236  013700  024142        1$:     MOV     UNITNO,R0       ;WE HAVE ASK HIM FOR ADDRESSES
1577  002242  112737  000260  014750       MOVB    #260,MRUNN1
1578  002250  104012                2$:     PRINT                   ;FIRST GET FK ADDRESS
1579  002252  024037                        MFKP
1580  002254  104012                        PRINT                   ;PRINT UNIT NO
1581  002256  014750                        MRUNN1
1582  002260  104015                        ASEMBL
1583  002262  010337  001500                MOV     R3,FKINT
1584  002266  001770                        BEQ     2$
1585  002270  004737  002700                JSR     PC,SETED
1586  002274  105237  014750                INCB    MRUNN1
1587  002300  005300                        DEC     R0
1588  002302  001362                        BNE     2$
1589  002304  012705  001410                MOV     #VTOCAR,R5      ;GET VT VECTORS AND ADDRESS
1590  002310  013700  024142                MOV     UNITNO,R0       ;NUMBER OF TUBES
1591  002314  112737  000260  014750       MOVB    #260,MRUNN1
1592
1593  002322  104012                3$:     PRINT
1594  002324  024032                        MUTP
1595  002326  104012                        PRINT
1596  002330  024043                        MAUFK
1597  002332  104012                        PRINT
1598  002334  014750                        MRUNN1
1599  002336  104015                        ASEMBL
1600  002340  010337  001500                MOV     R3,FKINT
1601  002344  001766                        BEQ     3$
1602  002346  004737  002700                JSR     PC,SETED
1603  002352  105237  014750                INCB    MRUNN1
1604  002356  005300                        DEC     R0
1605  002360  001360                        BNE     3$
1606
1607  002362  013700  024142                MOV     UNITNO,R0       ;ASK FOR DL11 VECTORS
1608  002366  012702  001540                MOV     #RINTO,R2
1609  002372  012705  001600                MOV     #XINTO,R5
1610  002376  112737  000260  014750       MOVB    #260,MRUNN1
1611
1612  002404  104012                4$:     PRINT                   ;PRINT MESSAGE
1613  002406  024101                        MDLVA
1614  002410  104012                        PRINT
1615  002412  014750                        MRUNN1
```

```
                                        ASEMBL
                                        MOV     (3)+,(2)+
                                        MOV     (3)+,(2)+
                                        MOV     (3)+,(5)+
                                        MOV     (3)+,(5)+
                                        INCB    MRUNN1
                                        DEC     R0
                                        BNE     4$

INITCB:                                 TST     TTYSWH
                                        BEQ     INITBA
                                        PRINT
                                        MESS                        ;REQUEST UNIT NO.
                                        ASEMBL                      ;DECODE IT.
                                        MOV     R3,UNITFG           ;SAVE IT
INITD:                                  MOV     UNITFG,PIMP
                                        CLR     SWAPEM              ;CLR SWAP SW.
                                        CMP     #4,UNITFG
                                        BNE     .+12
                                        CLR     UNITFG              ;YES. SET UP UNIT '0'
                                        INC     SWAPEM              ;SET SWAP SW.
                                        JSR     PC,STUNIT           ;SET UP SELECTED UNIT ADDRESSES
                                        TST     TTYSWH              ;TTY AVAILABLE?
                                        BEQ     MONITR              ;NO
                                        PRINT                       ;YES,
                                        MES3                        ;TEXT 'TYPE IN TST NO. TO BE EXECUTED'.
                                ;*******************************************************************
                                ;TEST SELECTION ROUTINE. IF NO TTY IS AVAILABLE THE PROGRAM WILL HALT
                                ;AND WAIT FOR A TEST NO. TO BE LOADED VIA THE SW.'S. IF A TTY IS
                                ;AVAIBLE, A REQUEST IS MADE VIA THE PRINTER FOR THE SELECTED TEST
                                ;NUMBER TO BE TYPED IN.
                                ;*******************************************************************

MONITR:                                 RESET                       ;INITIALIZE ON ENTRY
                                        NULL                        ;NULL FOR PRINTER.
                                        MOV     STACK,SP            ;RESET STACK POINTER
                                        CLR     SWAPEM
                                        CMP     #4,PIMP             ;DOES HE WANT TO SWAP?
                                        BNE     1$
                                        CLR     UNITFG
                                        INC     SWAPEM
                                        JSR     PC,STUNIT
1$:                                     MOV     #INITB,AVECTR       ;SET UP '^A' RESTART ADDR.
                                        JSR     PC,CLRFKV           ;CLR 'FK' INTR ADDR TO HALT
                                        JSR     PC,CLRVTV           ;CLR 'VT' INTR ADDR TO HALT
                                        MOV     #1,MTRSWH           ;SET MONITOR SW.
                                        CLR     RUNSWH              ;CLR 'RUN ALL' SOFTWARE SW.
                                        CLR     HOLDSO
                                        CLR     DONESW
                                        TST     TTYSWH              ;TTY AVAILABLE?
                                        BNE     TTYMTR              ;YES, REQUEST TEST NO.
                                        HALT                        ;GET TEST NO. FROM DATA SW'S
                                        MOV     @SWR,R3             ;SAVE TEST NO.
                                        BR      TTYMT1              ;EXECUTE SELECTED TEST
TTYMTR:                                 PRINT
                                        CNTRLC
```

```
                                        ASEMBL                          ;DECODE IT
002650  104015                  TTYMT1: BIC     #177740,R3              ;CLR UN-WANTED BITS
002652  042703  177740                  ASL     R3
002656  006303                          CMP     R3,#44
002660  020327  000044                  BGT     .+6                     ;TYPE '?' IF NUMBER TO BIG
002664  003002                          JMP     @TESTOO(R3)             ;EXECUTE SELECTED TEST
002666  000173  002750                  PRINT
002672  104012                          QMARK                           ;TYPE '?' ON ILLEGAL NO.'S
002674  023013                          BR      MONITR+2                ;RE-START
002676  000714                  SETED:  MOV     R1,-(6)                 ;SAVE R1
002700  010146                          MOV     #4,R1
002702  012701  000004          1$:     MOV     R4,(5)+                 ;MOV ADDR TO ADDR. POINTER
002706  010425                          ADD     #2,R4
002710  062704  000002                  DEC     R1
002714  005301                          BNE     1$
002716  001373                          MOV     FKINT,(5)+              ;MOV VECTOR ADDR TO ADDR POINTER
002720  013725  001500                  ADD     #2,FKINT
002724  062737  000002  001500          MOV     FKINT,(5)+
002732  013725  001500                  MOV     (6)+,R1
002736  012601                          ADD     #2,FKINT
002740  062737  000002  001500          RTS     PC
002746  000207
                        ;*****************************************************************
                        ;'VT20' TEST TABLE
                        ;*****************************************************************

002750  003122          TESTOO: VTLOGIC                 ;VT 'LOGIC' TEST

002752  005166          TEST01: VTCHAR                  ;VT 'CHARACTER DISPLAY' TEST

002754  005226          TEST02: VTMOD1                  ;VT 'FIELD MODE' TEST '1'

002756  005322          TEST03: VTMOD2                  ;VT 'CSR PRESET' TEST

002760  005454          TEST04: VTEOL                   ;VT 'END OF LINE' TEST

002762  005574          TEST05: VTEOS                   ;VT 'END OF SCREEN' TEST

002764  005660          TEST06: VTBLANK                 ;VT 'BLANK CONTROL' TEST

002766  005740          TEST07: VTALINE                 ;VT 'ALIGNMENT' TEST

002770  005766          TEST10: VTFOCUS                 ;VT 'FOCUS' TEST

002772  006014          TEST11: VTWORST                 ;VT 'WORST CASE' TEST

002774  006042          TEST12: VTCURSR                 ;VT 'CURSOR CONTROL' TEST

002776  006136          TEST13: VTODD                   ;VT 'ODD ADDRESS' TEST.

003000  003016          TEST14: VTRUN                   ;VT 'RUN ALL' TEST

003002  006236          TEST15: FKLOGIC                 ;FK 'LOGIC' TEST

003004  007416          TEST16: FKFUN                   ;FK 'FUNCTION' TEST

003006  007576          TEST17: FKCHAR                  ;FK 'CHARACTER' TEST
```

```
1728
1729  003010  010126                TEST20: FKREPT                  ;FK 'REPEATIBILITY' TEST
1730
1731  003012  010344                TEST21: SYSTST                  ;VT 'SYSTEM EXERCISER' TEST
1732
1733  003014  014162                TEST22: TESTX                   ;FK/VT 'LOGIC SUBTEST' SELECTION
1734
1735                                ;*************************************************************************
1736                                ;SUBROUTINE  ENTERED VIA SELECTING 'VTRUN'.THIS ROUTINE SETS A
1737                                ;SOFTWARE SW. WHICH ENABLES THE VT LOGIC TEST, ALL DISPLAY TESTS, AND
1738                                ;THE ASCII KEYBOARD TEST TO BE RUN IN ORDER. THESE TESTS WILL BE RUN
1739                                ;ON BOTH UNITS IF UNIT '2' WAS SELECTED.
1740                                ;*************************************************************************
1741
1742  003016  005237  024154        VTRUN:  INC     RUNSWH          ;SET SOFTWARE SW.
1743  003022  000137  003122                JMP     VTLOGIC         ;START 1ST TEST.
1744                                ;*************************************************************************
1745                                ;VT20 LOGIC TEST (00)
1746                                ;*************************************************************************
1747
1748  003026  013704  001510        SETUPV: MOV     VTCSR,VTCSR4    ;LOAD 'R4' WITH THE 'CSR' ADDRESS
1749  003032  013705  001512                MOV     VTMAR,VTMAR5    ;LOAD 'R5' WITH THE 'MAINT' ADDR. REG.
1750  003036  005037  024450                CLR     MESPRT          ;CLR PRINT INHIBIT SW.
1751  003042  005037  024452                CLR     MESPT1          ;CLR PRINT INHIBIT SW.
1752  003046  012737  000100  020376        MOV     #100,ICOUNT
1753  003054  012737  017526  000034        MOV     #VTERR,@#34     ;LOAD THE TRAP 'TRAP' WITH LOGIC ERROR TRAP
1754  003062  012737  003142  020402        MOV     #VT1,RETURN     ;SET UP 'SCOPE' RETURN ADDRESS
1755  003070  012777  000340  176214        MOV     #340,@PSW       ;SET PROCESSOR PRIORITY TO '7'
1756  003076  012737  000001  024146        MOV     #1,TSTNUM       ;SET UP TEST NO.
1757  003104  005037  024436                CLR     SHFTSW
1758  003110  005037  024440                CLR     SHFPRT
1759  003114  005037  020400                CLR     SCOPEF
1760  003120  000207                        RTS     PC
1761
1762  003122  005737  024160        VTLOGIC:TST     TTYSWH          ;TTY AVAILABLE?
1763  003126  001403                        BEQ     .+10            ;NO
1764  003130  104012                        PRINT
1765  003132  023017                        MES20                   ;TEXT 'VT20 LOGIC TEST'
1766  003134  104022                        NULL
1767  003136  004737  003026        VTREST: JSR     PC,SETUPV       ;INITIALIZE THE LOGIC TEST
1768
1769                                ;*************************************************************************
1770                                ;TEST THAT THE 'CSR' WAS INITIALIZED CORRECTLY
1771                                ;*************************************************************************
1772
1773  003142  000240                VT1:    NOP                     ;'VT' TEST 1
1774  003144  000240                        NOP
1775  003146  012714  125212                MOV     #125212,@VTCSR4
1776  003152  000005                        RESET
1777  003154  011401                        MOV     @VTCSR4,R1      ;READ CONTENTS OF THE 'CSR'
1778  003156  042701  050000                BIC     #50000,R1       ;IGNORE 'BIT 12' AND 'BIT 14'
1779  003162  001401                        BEQ     .+4             ;RESET SHOULD HAVE CLEARED 'CSR'
1780  003164  104400                        ERROR                   ;'CSR' SHOULD = '0 OR 50000'
1781
1782                                ;*************************************************************************
1783                                ;TEST THAT THE 'MAINTENANCE' REGISTER WAS INITIALIZED TO '0'
```

```
1784                                ;*****************************************************************************
1785
1786  003166  104001  000002      VT2:    SCOPE,2                 ;'VT' TEST 2
1787  003172  000005                       RESET
1788  003174  011501                       MOV     @VTMAR5,R1      ;READ CONTENTS OF THE 'MAR'
1789  003176  042701  050000               BIC     #50000,R1       ;IGNORE 'BIT 12' AND 'BIT 14'
1790  003202  001401                       BEQ     .+4
1791  003204  104400                       ERROR                   ;RESET DIDN'T CLR THE 'MAINT' REG. (EXCEPT BIT '12' AND
1792                                ;*****************************************************************************
1793                                ;TEST FOR WRITING THE 'CSR' TO 125212
1794                                ;*****************************************************************************
1795
1796  003206  104001  000003      VT3:    SCOPE,3                 ;'VT' TEST 3
1797  003212  012714  125212               MOV     #125212,@VTCSR4 ;WRITE 'CSR' TO '125212'
1798  003216  011401                       MOV     @VTCSR4,R1      ;READ CONTENTS OF THE 'CSR'
1799  003220  042701  050000               BIC     #50000,R1       ;IGNORE 'BIT 12' AND 'BIT 14'
1800  003224  022701  125212               CMP     #125212,R1
1801  003230  001401                       BEQ     .+4
1802  003232  104400                       ERROR                   ;'CSR' SHOULD = '125212'
1803  003234  005014                       CLR     @VTCSR4
1804
1805                                ;*****************************************************************************
1806                                ;TEST FOR WRITING THE 'CSR' TO '52504'
1807                                ;*****************************************************************************
1808
1809  003236  104001  000004      VT4:    SCOPE,4                 ;'VT' TEST 4
1810  003242  012714  052504               MOV     #52504,@VTCSR4  ;WRITE 'CSR' TO '52504'
1811  003246  104021                       WAITSYNC
1812  003250  022714  052504               CMP     #52504,@VTCSR4
1813  003254  001401                       BEQ     .+4             ;**BEWARE** BIT 6 IS CLEARED ON TYPEOUT
1814  003256  104400                       ERROR                   ;'CSR' SHOULD = '52504'
1815  003260  005014                       CLR     @VTCSR4         ;WRITE 'CSR' TO '0'
1816
1817                                ;*****************************************************************************
1818                                ;TEST FOR READING THE 'CSR' VIA THE 'MAINT' REG.
1819                                ;*****************************************************************************
1820
1821  003262  104001  000005      VT5:    SCOPE,5                 ;'VT' TEST 5
1822  003266  012714  157760               MOV     #157760,@VTCSR4 ;SELECT 'MAINT CSR' VIA THE 'CSR'
1823  003272  104021                       WAITSYNC
1824  003274  022715  157760               CMP     #157760,@VTMAR5 ;MAINT SHOULD='CSR'
1825  003300  001401                       BEQ     .+4             ;**BEWARE** BIT 6 IS CLEARED ON TYPEOUT
1826  003302  104400                       ERROR                   ;MAINT REG. NOT ='CSR' (157760)
1827  003304  005014                       CLR     @VTCSR4         ;CLR 'CSR'
1828  003306  011501                       MOV     @VTMAR5,R1
1829  003310  042701  050000               BIC     #50000,R1
1830  003314  005701                       TST     R1              ;'MAR' SHOULD = 'CSR' (0)
1831  003316  001401                       BEQ     .+4
1832  003320  104400                       ERROR                   ;MAINT REG. NOT='CSR' (0)
1833
1834                                ;*****************************************************************************
1835                                ;WRITE THE 'CURSOR' ADDRESS REGISTER TO '125252'
1836                                ;*****************************************************************************
1837
1838  003322  104001  000006      VT6:    SCOPE,6                 ;'VT' TEST 6
1839  003326  012777  125252  176150       MOV     #125252,@VTCAR  ;WRITE REG. TO ALTERNATE 1'S
```

```
1840  003334  012777  000010  176146          MOV     #CAR,@VTCSR     ;SELECT 'CURSOR' ADDRESS VIA 'CSR'
1841  003342  022715  125252                  CMP     #125252,@VTMAR5 ;READ CURSOR ADDRESS VIA 'MAINT'
1842  003346  001401                          BEQ     .+4
1843  003350  104400                          ERROR                   ;CURSOR ADDRESS SHOULD=125252
1844  003352  005077  176132                  CLR     @VTCSR
1845  003356  005077  176122                  CLR     @VTCAR
1846
1847                                  ;*************************************************************************
1848                                  ;WRITE THE 'CURSOR' ADDRESS REGISTER TO '52525'
1849                                  ;*************************************************************************
1850
1851  003362  104001  000007          VT7:    SCOPE,7                 ;'VT' TEST 7
1852  003366  012777  052525  176110          MOV     #52525,@VTCAR   ;WRITE COMPLIMENT OF PART I
1853  003374  012777  000010  176106          MOV     #CAR,@VTCSR     ;SELECT 'CURSOR' ADDRESS VIA 'CSR'
1854  003402  022715  052525                  CMP     #52525,@VTMAR5  ;READ 'CAR' VIA 'MAR'
1855  003406  001401                          BEQ     .+4
1856  003410  104400                          ERROR                   ;'CAR' SHOULD=52525
1857  003412  005077  176072                  CLR     @VTCSR
1858  003416  005077  176062                  CLR     @VTCAR
1859
1860                                  ;*************************************************************************
1861                                  ;WRITE THE 'CURSOR' ADDRESS REGISTER TO '0'
1862                                  ;*************************************************************************
1863
1864  003422  104001  000010          VT10:   SCOPE,10                        ;'VT' TEST 10
1865  003426  012777  177777  176050          MOV     #-1,@VTCAR      ;WRITE REG. TO -1
1866  003434  012714  000010                  MOV     #CAR,@VTCSR4    ;SELECT CURSOR ADDRESS
1867  003440  005077  176040                  CLR     @VTCAR          ;WRITE REG. TO '0'
1868  003444  005715                          TST     @VTMAR5
1869  003446  001401                          BEQ     .+4
1870  003450  104400                          ERROR                   ;CURSOR ADDRESS SHOULD='0'
1871
1872                                  ;*************************************************************************
1873                                  ;TEST THAT THE 'CURSOR ADDRESS' REG. IS CLEARED VIA 'RESET'
1874                                  ;*************************************************************************
1875
1876  003452  104001  000011          VT11:   SCOPE,11                        ;'VT' TEST 11
1877  003456  012777  177777  176020          MOV     #-1,@VTCAR      ;WRITE REG. =-1
1878  003464  000005                          RESET                   ;ISSUE 'RESET' TO CLR REG.
1879  003466  012714  000010                  MOV     #CAR,@VTCSR4    ;SELECT 'CAR' VIA 'CSR'
1880  003472  005715                          TST     @VTMAR5         ;TEST IF 'CAR' WAS CLEARED
1881  003474  001401                          BEQ     .+4
1882  003476  104400                          ERROR                   ;RESET DIDN'T CLR 'CAR' REG.
1883                                  ;*************************************************************************
1884                                  ;TEST FOR ROTATING A '1' THRU THE 'CAR'.
1885                                  ;*************************************************************************
1886
1887  003500  104001  000012          VT12:   SCOPE,12                        ;'VT' TEST 12
1888  003504  012703  000001                  MOV     #1,R3           ;LOAD THE TEST REG. WITH '1'
1889  003510  010377  175770          LOOP12: MOV     R3,@VTCAR       ;LOAD THE 'CAR' WITH THE TEST REG.
1890  003514  012714  000010                  MOV     #CAR,@VTCSR4
1891  003520  020315                          CMP     R3,@VTMAR5
1892  003522  001401                          BEQ     .+4
1893  003524  104400                          ERROR                   ;'CAR' SHOULD = CONTAINS OF R3
1894  003526  005014                          CLR     @VTCSR4
1895  003530  005077  175750                  CLR     @VTCAR
```

```
1896  003534  006103                          ROL     R3              ;ROTATE BIT
1897  003536  001364                          BNE     LOOP12
1898                                  ;*****************************************************************
1899                                  ;TEST FOR WRITING THE 'STARTING ADDRESS' REG. TO '125252'
1900                                  ;*****************************************************************
1901
1902  003540  104001  000013         VT13:   SCOPE,13                        ;'VT' TEST 13
1903  003544  012777  024500  175734         MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1904  003552  012714  000012                MOV     #SAR,@VTCSR4    ;SELECT 'SAR' TO MAINT VIA 'CSR'
1905  003556  012777  125252  175722         MOV     #125252,@VTSAR  ;WRITE REG. TO ALTERNATE 1'S
1906  003564  104021                         WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1907  003566  022715  125252                CMP     #125252,@VTMAR5
1908  003572  001401                         BEQ     .+4
1909  003574  104400                         ERROR                   ;STARTING ADDR. REG. SHOULD=125252
1910
1911                                  ;*****************************************************************
1912                                  ;TEST FOR WRITING THE 'STARTING ADDRESS' REG. TO '52525'
1913                                  ;*****************************************************************
1914
1915  003576  104001  000014         VT14:   SCOPE,14                        ;'VT' TEST 14
1916  003602  012777  024500  175676         MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1917  003610  012714  000012                MOV     #SAR,@VTCSR4    ;SELECT 'SAR' TO MAINT VIA 'CSR'
1918  003614  012777  052525  175664         MOV     #52525,@VTSAR   ;WRITE COMPLIMENT OF PART I
1919  003622  104021                         WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1920  003624  022715  052525                CMP     #52525,@VTMAR5
1921  003630  001401                         BEQ     .+4
1922  003632  104400                         ERROR                   ;'SAR' SHOULD =52525
1923                                  ;*****************************************************************
1924                                  ;TEST FOR WRITING THE 'STARTING ADDRESS' REG. TO '0'
1925                                  ;*****************************************************************
1926
1927  003634  104001  000015         VT15:   SCOPE,15                        ;'VT' TEST 15
1928  003640  012777  024500  175640         MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1929  003646  012714  000012                MOV     #SAR,@VTCSR4    ;SELECT 'STARTING ADDR.' REG.
1930  003652  005077  175630                CLR     @VTSAR          ;WRITE REG. TO '0'
1931  003656  104021                         WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1932  003660  005715                         TST     @VTMAR5         ;CHECK FOR '0'
1933  003662  001401                         BEQ     .+4
1934  003664  104400                         ERROR                   ;STARTING ADDRESS REG. SHOULD='0'
1935                                  ;*****************************************************************
1936                                  ;TEST FOR ROTATING A '1' THUR THE STARTING ADDRESS REG.
1937                                  ;NOTE: 'R3' IS USED AS A CHECKING ADDRESS AND WILL CONTAIN THE
1938                                  ;EXPECTED DATA ON A FAILURE.
1939                                  ;*****************************************************************
1940
1941  003666  104001  000016         VT16:   SCOPE,16                        ;'VT' TEST 16
1942  003672  012777  024500  175606         MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
1943  003700  012703  000001                MOV     #1,R3           ;LOAD THE TEST REG. WITH '1'
1944  003704  012714  000012        LOOP16: MOV     #SAR,@VTCSR4    ;SELECT 'STARTING ADDR.' REG.
1945  003710  010377  175572                MOV     R3,@VTSAR       ;LOAD 'SAR  FROM 'R3'
1946  003714  104021                         WAITSYNC                ;WAIT FOR VERTICAL SYNC.
1947  003716  020315                         CMP     R3,@VTMAR5      ;CHECK IF REG. LOADED CORRECTLY.
1948  003720  001401                         BEQ     .+4
1949  003722  104400                         ERROR                   ;'SAR SHOULD = CONTAINS OF 'R3'
1950  003724  006103                         ROL     R3              ;ROTATE BIT
1951  003726  001366                         BNE     LOOP16          ;LOOP  UNTIL CARRY IS SET
```

```
                                        ;*****************************************************************
                                        ;TEST THAT 'RESET' WILL CLR THE 'STARTING ADDRESS' REG.
                                        ;*****************************************************************

003730  104001  000017          VT17:   SCOPE,17                        ;'VT' TEST 17
003734  012777  024500  175544          MOV     #VTBUFF,@VTSAR  ;LOAD LEGAL ADDR. TO PREVENT TIMEOUT
003742  012714  000012                  MOV     #SAR,@VTCSR4    ;SELECT 'STARTING ADDRESS' REG.
003746  012777  177777  175532          MOV     #-1,@VTSAR      ;WRITE REG. TO '-1'
003754  000005                          RESET                   ;RESET SHOULD CLR REG.
003756  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
003760  012714  000012                  MOV     #SAR,@VTCSR4    ;RE-SELECT THE 'SAR'
003764  005715                          TST     @VTMAR5
003766  001401      -                   BEQ     .+4
003770  104400                          ERROR                   ;RESET DIDN'T CLR 'SAR'

                                        ;*****************************************************************
                                        ;TEST THAT THE 'DAR' (DATA ADDRESS REG.) IS LOADED FROM 'SAR' (START ADDR. REG.)
                                        ;*****************************************************************

003772  104001  000020          VT20:   SCOPE,20                        ;'VT' TEST 20
003776  012777  125252  175502          MOV     #125252,@VTSAR  ;LOAD THE 'SAR'
004004  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
004006  012714  000002                  MOV     #DAR,@VTCSR4    ;SELECT MAINT. 'DAR' VIA 'CSR'
004012  022715  125252                  CMP     #125252,@VTMAR5
004016  001401                          BEQ     .+4
004020  104400                          ERROR                   ;'DAR' WASN'T LOADED FROM 'SAR'
004022  005014                          CLR     @VTCSR4

                                        ;*****************************************************************
                                        ;ISSUE RESET TO CLR TIME-OUT FROM PREVIOUS TEST
                                        ;*****************************************************************

004024  104001  000021          VT21:   SCOPE,21                        ;'VT' TEST 21
004030  000005                          RESET

                                        ;*****************************************************************
                                        ;TEST THAT THE 'DAR' (DATA ADDRESS REG.) IS LOADED FROM 'SAR' (START ADDR. REG.)
                                        ;*****************************************************************

004032  104001  000022          VT22:   SCOPE,22                        ;'VT' TEST 22
004036  012777  052525  175442          MOV     #052525,@VTSAR  ;LOAD THE 'SAR'
004044  104021                          WAITSYNC                ;WAIT FOR VERTICAL SYNC.
004046  012714  000002                  MOV     #DAR,@VTCSR4    ;SELECT MAINT. 'DAR'
004052  022715  052524                  CMP     #052524,@VTMAR5
004056  001401                          BEQ     .+4
004060  104400                          ERROR                   ;'DAR' WASN'T LOADED FROM 'SAR'
004062  005014                          CLR     @VTCSR4
                                        ;*****************************************************************
                                        ;ISSUE RESET TO CLR TIME-OUT FROM PREVIOUS TEST
                                        ;*****************************************************************

004064  104001  000023          VT23:   SCOPE,23                        ;'VT' TEST 23
004070  000005                          RESET
                                        ;*****************************************************************
                                        ;TEST FOR ROTATING A '1' THUR THE 'DAR' (DAR IS LOADED FROM THE SAR).
                                        ;*****************************************************************
```

```
2008
2009  004072  104001  000024          VT24:   SCOPE,24                          ;'VT' TEST 24
2010  004076  012703  000002                  MOV     #2,R3                   ;LOAD THE TEST REG. WITH '2'
2011  004102  010377  175400          LOOP24: MOV     R3,@VTSAR               ;LOAD THE 'SAR'
2012  004106  104021                          WAITSYNC                        ;WAIT FOR VERTICAL SYNC.
2013  004110  012714  000002                  MOV     #DAR,@VTCSR4            ;SELECT MAINT. 'DAR' VIA 'CSR'
2014  004114  020315                          CMP     R3,@VTMAR5              ;TEST 'DAR'
2015  004116  001401                          BEQ     .+4
2016  004120  104400                          ERROR                           ;'DAR' WASN'T LOADED FROM 'SAR'
2017  004122  005014                          CLR     @VTCSR4
2018  004124  006103                          ROL     R3                      ;ROTATE BIT
2019  004126  001365                          BNE     LOOP24                  ;ROTATE THRU REG.
2020
2021                                  ;*************************************************************************
2022                                  ;ISSUE RESET TO CLR TIME-OUT FROM PREVIOUS TEST
2023                                  ;*************************************************************************
2024
2025  004130  104001  000025          VT25:   SCOPE,25                        ;'VT' TEST 25
2026  004134  000005                          RESET
2027
2028                                  ;*************************************************************************
2029                                  ;TEST THAT THE 'CURSOR ADDR.' BIT IS SET IN THE 'CSR' IF THE 'CAR'
2030                                  ;AND THE 'SAR' ARE SET EQUAL.  ALSO BIT 14 CLEARS IF CAR=SAR
2031                                  ;*************************************************************************
2032
2033  004136  104001  000026          VT26:   SCOPE,26                        ;'VT' TEST 26
2034  004142  012777  020723  175336          MOV     #SHFTBF+1,@VTSAR        ;LOAD 'SAR'
2035  004150  104021                          WAITSYNC                        ;WAIT FOR VERTICAL SYNC.
2036  004152  012777  020723  175324          MOV     #SHFTBF+1,@VTCAR        ;LOAD 'CAR' = 'SAR'
2037  004160  012714  000004                  MOV     #SRIDE,@VTCSR4          ;FORCE 'NPR' TRANSFER
2038  004164  032714  010000                  BIT     #10000,@VTCSR4          ;TEST IF BIT 12 (CAR BIT) SET
2039  004170  001001                          BNE     .+4
2040  004172  104400                          ERROR                           ;'CAR' BIT IN 'CSR' DIDN'T SET W/'CAR' = 'SAR'
2041  004174  032714  040000                  BIT     #40000,@VTCSR4          ;TEST IF BIT14 CLEAR
2042  004200  001401                          BEQ     .+4                     ;SHOULD BE WHEN SAR=CAR
2043  004202  104400                          ERROR                           ;BIT 14 NOT CLEAR
2044  004204  005014                          CLR     @VTCSR4
2045                                  ;*************************************************************************
2046                                  ;TEST THAT THE 'CURSOR ADDR.' BIT IS CLEARED IF THE 'CAR' AND 'SAR'
2047                                  ;ARE NOT SET EQUAL.
2048                                  ;*************************************************************************
2049
2050  004206  104001  000027          VT27:   SCOPE,27                        ;'VT' TEST 27
2051  004212  012777  020735  175266          MOV     #SHFTBF+13,@VTSAR       ;LOAD 'SAR'
2052  004220  104021                          WAITSYNC                        ;WAIT FOR VERTICAL SYNC.
2053  004222  012777  020733  175254          MOV     #SHFTBF+11,@VTCAR       ;LOAD 'CAR' = 'SAR'
2054  004230  012714  000004                  MOV     #SRIDE,@VTCSR4  ;FORCE 'NPR' TRANSFER
2055  004234  032715  010000                  BIT     #10000,@VTMAR5          ;TEST IF BIT 12 (CAR BIT) SET
2056  004240  001401                          BEQ     .+4
2057  004242  104400                          ERROR                           ;'CAR' SET W/ 'CAR' NOT = 'SAR'
2058  004244  032777  004000  175236          BIT     #4000,@VTCSR    ;SEE IF BIT 14 IN CSR, SET SHOULD BE
2059  004252  001001                          BNE     .+4             ;WHEN CAR <> SAR WITH ECO FOR BIT 14
2060  004254  000240                          NOP                             ;NOP IF NO ECO, PUT IN 104400 IN THIS
2061                                                                          ;LOCATION IF ECO IS IN.
2062  004256  005014                          CLR     @VTCSR4
2063                                  ;*************************************************************************
```

```
                                        ;TEST THE 'EOS' (END OF SCREEN) BIT IS SET IF THE 'EOS' CHAR. IS TRANSFERRED
                                        ;*************************************************************************
                                VT30:   SCOPE,30                        ;'VT' TEST 30
                                        MOV     #SHFEND,@VTSAR          ;LOAD 'SAR'
                                        WAITSYNC                        ;WAIT FOR VERTICAL SYNC.
                                        MOV     #SRIDE,@VTCSR4          ;FORCE 'NPR' TRANSFER
                                        BIT     #20000,@VTCSR4          ;TEST IF BIT 13 (EOS)
                                        BNE     .+4
                                        ERROR                           ;'EOS' DIDN'T SET IN 'CSR'
                                        CLR     @VTCSR4

                                        ;*************************************************************************
                                        ;TEST THE 'SHIFT REG. INPUT DATA EVEN' GETS TRANSFERRED FROM THE DATA BUFFER
                                        ;TO THE 'MAR'.  A SERIES OF '64' SHIFTS ARE DONE AND THE 'MAR' IS COMPARED
                                        ;TO THE DATA BUFFER AFTER EACH SHIFT.
                                        ;*************************************************************************
004312  104001  000031          VT31:   SCOPE,31                        ;'VT' TEST 31
004316  000005                          RESET
004320  012737  000010  020376          MOV     #10,ICOUNT              ;RESET ITERATION COUNT TO '8'
004326  012737  023550  020022          MOV     #MES37,SHFMES           ;SET UP TO PRINT ON ERROR
004334  012737  000004  024434          MOV     #SRIDE,SELECT           ;SELECT 'SHIFT INPUT DATA EVEN'
004342  004737  004752                  JSR     PC,INITSF               ;INITIALIZE SHIFT TEST
004346  004737  005004                  JSR     PC,SHIFT                ;SUBROUTINE TO FORCE '64' 'NPR' XFER
004352  104400                          ERROR                           ;'SHIFT INPUT DATA EVEN' SHIFT ERROR
004354  000415                          BR      TAGVB                   ;EXIT ON ERROR
                                        ;*************************************************************************
                                        ;AT THIS POINT '64' SHIFTS HAVE BEEN MADE TO THE 'SHIFT REG. INPUT DATA
                                        ;EVEN'. THE NEXT TEST EXECUTES '63' MORE SHIFTS VALIDATING THE
                                        ;'SHIFT REG. OUTPUT DATA EVEN' INPUTTED FROM THE PREVIOUS TEST.
                                        ;*************************************************************************

004356  012737  023561  020022                  MOV     #MES38,SHFMES
004364  012737  000006  024434                  MOV     #SRODE,SELECT   ;SELECT 'SHIFT OUTPUT DATA EVEN'
004372  004737  004752                          JSR     PC,INITSF       ;INITIALIZE SHIFT TEST
004376  005722                                  TST     (R2)+           ;UPDATE BUFFER COMPARE POINTER BY '2'
004400  005200                                  INC     R0              ;SHIFT ONLY 63 TIMES FOR OUTPUT
004402  004737  005004                          JSR     PC,SHIFT        ;SUBROUTINE TO FORCE THE '63' NPR XFERS
004406  104400                                  ERROR                   ;'SHIFT OUTPUT DATA EVEN' SHIFT ERROR
004410  000240                  TAGVB:  NOP
                                        ;*************************************************************************
                                        ;TEST THE 'SHIFT REG. INPUT DATA ODD' GETS TRANSFERRED FROM THE DATA BUFFER
                                        ;TO THE 'MAR'.  A SERIES OF '64' SHIFTS ARE DONE AND THE 'MAR' IS COMPARED
                                        ;TO THE DATA BUFFER AFTER EACH SHIFT.
                                        ;*************************************************************************

004412  104001  000032          VT32:   SCOPE,32                        ;'VT' TEST 32
004416  000005                          RESET
004420  012737  023550  020022          MOV     #MES37,SHFMES
004426  012737  000014  024434          MOV     #SRIDO,SELECT   ;SELECT 'SHIFT INPUT DATA ODD'
004434  004737  004752                  JSR     PC,INITSF       ;INITIALIZE SHIFT TEST
004440  004737  005004                  JSR     PC,SHIFT        ;SUBROUTINE TO FORCE '1' 'NPR' XFER
004444  104400                          ERROR                   ;'SHIFT INPUT DATA ODD' SHIFT ERROR
004446  000415                          BR      TAGVC           ;EXIT ON ERROR
```

```
                                ;*****************************************************************************
                                ;AT THIS POINT '64' SHIFTS HAVE BEEN MADE TO THE 'SHIFT REG. INPUT DATA
                                ;ODD'. THIS NEXT TEST EXECUTES '63' MORE SHIFTS VALIDATING THE
                                ;'SHIFT REG. OUTPUT DATA ODD' INPUTTED FROM THE PREVIOUS TEST.
                                ;*****************************************************************************

004450  012737  023561  020022          MOV     #MES38,SHFMES
004456  012737  000016  024434          MOV     #SRODO,SELECT   ;SELECT 'SHIFT OUTPUT DATA ODD'
004464  004737  004752                  JSR     PC,INITSF       ;INITIALIZE SHIFT TEST
004470  005722                          TST     (R2)+           ;UPDATE BUFFER COMPARE POINTER BY '2'
004472  005200                          INC     R0              ;SHIFT ONLY 63 TIMES FOR OUTPUT
004474  004737  005004                  JSR     PC,SHIFT        ;SUBROUTINE TO FORCE THE '64' NPR XFERS
004500  104400                          ERROR                   ;'SHIFT OUTPUT DATA ODD' SHIFT ERROR
004502  000240                  TAGVC:  NOP
                                ;*****************************************************************************
                                ;TEST 'NXM' FOR CAUSING AN INTERRUPT VIA LOADING A 'NXM' ADDRESS IN THE 'SAR'
                                ;NOTE: THIS IS THE FIRST TIME THAT THE 'INTR. ENABLE BIT IS SET OR USED.
                                ;*****************************************************************************

004504  104001  000033          VT33:   SCOPE,33                        ;'VT' TEST 33
004510  012737  000100  020376          MOV     #100,ICOUNT             ;RESET ITERATION COUNT TO '100'
004516  005037  024436                  CLR     SHFTSW          ;CLR SOFTWARE SW.
004522  012777  173000  174756          MOV     #173000,@VTSAR  ;LOAD 'SAR' W/ 'NXM' ADDRESS
004530  004737  016740                  JSR     PC,LDVTVT       ;LOAD THE 'VT' VECTOR ADDRESS
                                        TAGVA
004536  104021                          WAITSYNC                ;WAIT FOR INTERRUPT
004540  104021                          WAITSYNC
004542  104400                          ERROR                   ;SETTING 'SAR' TO 'NXM' ADDR. DIDN'T ERROR
004544  013706  024136          TAGVA:  MOV     STACK,SP        ;RESET STACK POINTER
004550  004737  016774                  JSR     PC,CLRVTV       ;RESET 'VT' VECTOR ADDR. TO HALT
004554  005714                          TST     @VTCSR4         ;TEST THE 'NXM' SET THE ERROR BIT
004556  100401                          BMI     .+4
004560  104400                          ERROR                   ;'NXM' FAILED TO SET 'ERROR' BIT
004562  000005                          RESET

                                ;*****************************************************************************
                                ;TEST THAT NO INTERRUPT OCCURS WITH PROC. AT PRIORITY '5'
                                ;*****************************************************************************

004564  104001  000034          VT34:   SCOPE,34                        ;'VT' TEST 34
004570  012777  173000  174710          MOV     #173000,@VTSAR
004576  004737  016740                  JSR     PC,LDVTVT       ;LOAD 'VT' INTERRUPT VECTOR
004602  004624                          TAGVD
004604  012777  000240  174500          MOV     #240,@PSW
004612  012700  170000                  MOV     #170000,R0
004616  005200                          INC     R0
004620  001376                          BNE     .-2
004622  000403                          BR      .+10            ;'OK', NO INTERRUPT OCCURRED
004624  104400                  TAGVD:  ERROR                   ;INTERRUPTED WITH PROC @ PRIOR '5'
004626  013706  024136                  MOV     STACK,SP
004632  000005                          RESET
004634  004737  016774                  JSR     PC,CLRVTV
                                ;*****************************************************************************
                                ;TEST THAT NO 'NXM' ERRORS OCCUR IF THE 'VT' IS RUNNING AND THE
                                ;STARTING ADDRESS REG. IS CHANGED.
                                ;*****************************************************************************
```

```
2176
2177  004640  104001  000035          VT35:   SCOPE,35                          ;'VT' TEST 35
2178  004644  012701  000010                  MOV     #10,R1          ;SET UP A LOOP COUNTER
2179  004650  012700  020724                  MOV     #SHFTBF+2,R0
2180  004654  012777  020722  174624          MOV     #SHFTBF,@VTSAR
2181  004662  104021                         WAITSYNC                ;WAIT FOR VERICAL SYNC
2182  004664  004737  016740                  JSR     PC,LDVTVT       ;LOAD 'VT' INTERRUPT VECTOR ADDR.
2183  004670  004716                         TAGVF
2184  004672  005002                 TAGVE:  CLR     R2
2185  004674  005202                         INC     R2              ;EVERYTIME 'R2' OVERFLOWS THE
2186  004676  001376                         BNE     .-2             ;'SAR' IS RELOADED
2187  004700  010077  174602                 MOV     R0,@VTSAR       ;RELOAD 'SAR'
2188  004704  005720                         TST     (R0)+
2189  004706  104021                         WAITSYNC                ;WAIT FOR VERICAL SYNC
2190  004710  005301                         DEC     R1
2191  004712  001367                         BNE     TAGVE
2192  004714  000403                         BR      .+10            ;OK, NO INTERRUPTS OCCURRED
2193  004716  104400                 TAGVF:  ERROR                   ;'NXM' ERROR OCCURRED WHEN 'SAR' WAS CHANGED
2194  004720  013706  024136                 MOV     STACK,SP
2195  004724  004737  016774                 JSR     PC,CLRVTV
2196
2197                                  ;*****************************************************************
2198                                  ;'VT' LOGIC TEST COMPLETE
2199                                  ;*****************************************************************
2200
2201  004730  104001  000036          VT36:   SCOPE,36                        ;'VT' TEST 36
2202  004734  104012                         PRINT
2203  004736  021615                         MES7                    ;TEXT 'TEST COMPLETE'
2204  004740  104006                         ENDTST                  ;END OF TEST
2205  004742  000401                         BR      .+4
2206  004744  000510                         BR      VTCHAR          ;CONTINUE TO CHARACTER TEST
2207  004746  000137  003136                 JMP     VTREST          ;RESTART THE LOGIC TEST.
2208
2209                                  ;*****************************************************************
2210                                  ;SUBROUTINE TO INITIALIZE THE 'SAR', 'CAR' AND SHIFT COUNTER FOR THE
2211                                  ;'VT' LOGIC SHIFT TEST.
2212                                  ;*****************************************************************
2213
2214  004752  012777  020722  174526  INITSF: MOV     #SHFTBF,@VTSAR  ;LOAD 'STARTING ADDR'
2215  004760  012703  020722                  MOV     #SHFTBF,R3
2216  004764  104021                         WAITSYNC                ;WAIT FOR VERTICAL SYNC.
2217  004766  012702  020722                  MOV     #SHFTBF,R2
2218  004772  005000                         CLR     R0              ;CLR 'SHIFT' COUNTER
2219  004774  012737  000001  024436          MOV     #1,SHFTSW       ;SET SOFTWARE SW.
2220  005002  000207                         RTS     PC
2221                                  ;*****************************************************************
2222                                  ;SUBROUTINE TO FORCE THE 'NPR' XFER FOR THE 'VT' LOGIC SHIFT TEST.
2223                                  ;THE ADDRESS 'SELECT' CONTAINS THE SELECTED SHIFT FUNCTION (SHIFT INPUT
2224                                  ;DATA EVEN, ETC.) WHICH IS EXECUTED.
2225                                  ;*****************************************************************
2226
2227  005004  005200                 SHIFT:  INC     R0              ;INCREMENT SOFTWARE SHIFT COUNTER
2228  005006  013714  024434                  MOV     SELECT,@VTCSR4  ;FORCE 'NPR' TRANSFER
2229  005012  005723                         TST     (R3)+           ;UPDATE STARTING ADDRESS BY '2'
2230  005014  010377  174466                 MOV     R3,@VTSAR       ;SET UP TO GET NEXT BUFFER WORD.
2231  005020  104021                         WAITSYNC                ;WAIT FOR VERTICAL SYNC.
```

```
2232  005022  012201                        MOV     (R2)+,R1        ;GET XFER'D WORD FROM DATA BUFFER
2233  005024  042701  000360                BIC     #360,R1         ;CLR UNWANTED BITS FROM CONTROL WORD
2234  005030  000301                        SWAB    R1              ;SWAP BYTES TO COMPARE TO 'MAR'
2235  005032  022700  000040        SHFLST: CMP     #32.,R0         ;IS THIS THE '32' SHIFT?
2236  005036  001010                        BNE     SHF64           ;NO, CHECK FOR '64' SHIFT
2237  005040  022715  020031                CMP     #20031,@VTMAR5  ;YES,END OF SCREEN SHOULD BE SET IN 'MAR'
2238  005044  001013                        BNE     SHFBAD          ;NO. 32ND SHIFT IS BAD
2239  005046  004737  004752        SHF32:  JSR     PC,INITSF       ;RE-INITIALIZE SOFWARE POINTERS FOR '32' MORE SHIFTS
2240  005052  012700  000040                MOV     #32.,R0         ;RESET SHIFT COUNTER TO COUNT TO '64'
2241  005056  000752                        BR      SHIFT           ;DO IT.
2242  005060  022700  000100        SHF64:  CMP     #64.,R0         ;IS THIS THE LAST SHIFT
2243  005064  001006                        BNE     TSTSHF          ;NO, COMPARE DATA TO 'MAR' DIRECTLY
2244  005066  022715  020031                CMP     #20031,@VTMAR5  ;YES, 'EOS' SHOULD BE SET
2245  005072  001432                        BEQ     EXITSF          ;EXIT
2246  005074  052701  020000        SHFBAD: BIS     #20000,R1       ;SET UP 'R1' TO = EXPECTED WORD
2247  005100  000412                        BR      REPORT          ;REPORT ERROR
2248  005102  032701  000400        TSTSHF: BIT     #400,R1         ;COMPLIMENT BLANK BIT FOR COMPARE
2249  005106  001003                        BNE     .+10
2250  005110  052701  000400                BIS     #400,R1
2251  005114  000402                        BR      .+6
2252  005116  042701  000400                BIC     #400,R1
2253  005122  020115                        CMP     R1,@VTMAR5      ;DOES 'MAR' = EXPT'D WORD?
2254  005124  001727                        BEQ     SHIFT           ;YES, FORCE NEXT SHIFT
2255  005126  022700  000100        REPORT: CMP     #64.,R0         ;IS ERROR ON THE LAST SHIFT?
2256  005132  001404                        BEQ     .+12            ;YES,EXIT
2257  005134  032777  010000  174162        BIT     #SW12,@SWR      ;NO, IS SW SET TO CONTINUE SHIFTING?
2258  005142  001401                        BEQ     .+4             ;YES
2259  005144  000207                        RTS     PC              ;RETURN TO SUBTEST AND REPORT ERROR
2260  005146  104400                        ERROR                   ;PRINT ERROR DATA
2261  005150  022700  000040                CMP     #32.,R0         ;IS THIS THE '32' SHIFT
2262  005154  001734                        BEQ     SHF32           ;YES, RE-INITIALIZE BUFFER
2263  005156  000712                        BR      SHIFT           ;NO,CONTINUE SHIFTING
2264  005160  062716  000004        EXITSF: ADD     #4,(SP)         ;SET UP STACK TO SKIP OVER ERROR
2265  005164  000207                        RTS     PC
2266                                ;******************************************************************************
2267                                ;VT 'CHARACTER DISPLAY' TEST (1)
2268                                ;THIS TEST DISPLAYS ALL '160' VT CHARACTERS ONE AT A TIME UNTIL THE ENTIRE
2269                                ;'VT' SCREEN IS FILLED. EACH CHARACTER IS DISPLAYED FOR APPROXIMATELY '3' SECONDS
2270                                ;BEFORE THE NEXT CHARACTER IS DISPLAYED.  ON EACH LINE,THE '64' CHARACTER IS DIS-
2271                                ;PLAYED AS EITHER A 'VISIBLE END OF LINE OR A 'VISIBLE END OF PARAGRAPH'.
2272                                ;DATA 'SW14' MAY BE SET AT ANYTIME TO HOLD ON ANY INDIVIDUAL CHARACTER.
2273                                ;******************************************************************************
2274
2275  005166  012737  005166  020402  VTCHAR: MOV     #VTCHAR,RETURN
2276  005174  104000                        DISPLAY
2277  005176  023210                        MES24                           ;TEXT "VT CHARACTER DISPLAY TEST".
2278  005200  104005                        EOSBUF                          ;LOAD 'VT' BUFFER WITH 'EOS'.
2279  005202  005037  024156                CLR     INTSWH                  ;SUBROUTINE SOFTWARE SW.
2280  005206  000401                        BR      .+4
2281  005210  104004                        DELAY                           ;3 SEC. DELAY.
2282  005212  004737  015170                JSR     PC,INTCHR               ;CREATE & DISPLAY NEXT CHAR.
2283  005216  000240                        NOP                             ;NORMAL RETURN TO HERE
2284  005220  000773                        BR      .-10                    ;END OF LINE RETURNS HERE
2285  005222  104006                        ENDTST
2286  005224  000760                        BR      VTCHAR                  ;END OF SCREEN RETURN HERE
2287
```

```
2288                                    ;*************************************************************************
2289                                    ;VT 'FIELD CONTROL' TEST (2)
2290                                    ;THIS TEST DISPLAYS THE 'VT' CHARACTER SET USING ALL '5' FIELD CONTROL MODES.
2291                                    ;(NORMAL,BLINK,BOLD,BLANK AND UNDERLINE). THE TEST DISPLAYS CHARACTERS
2292                                    ;'40-136' ON LINES '1-5' USING ALL '5' MODES. CHARACTERS '137-240 (WITH THE
2293                                    ;EXCEPTION OF CHARACTERS 212,214 & 231) ON LINES '6-10'. CHARACTERS
2294                                    ;'241-277' ARE DISPLAYED ON THE LAST '5' LINES.
2295                                    ;*************************************************************************
2296
2297  005226  012737  005226  020402  VTMOD1: MOV     #VTMOD1,RETURN
2298  005234  104000                          DISPLAY
2299  005236  023244                          MES25                           ;TEXT 'VT FIELD CONTROL TEST'.
2300  005240  104005                          EOSBUF                          ;LOAD 'VT' BUFFER WITH 'VISIBLE EOS'.
2301  005242  012737  024500  024464          MOV     #VTBUFF,KSTOR1
2302  005250  005037  024156                  CLR     INTSWH                  ;CLR SOFTWARE SWITCH
2303  005254  012737  177775  024470          MOV     #-3,KSTOR3              ;SET UP FOR '3' SETS OF CHAR.'S
2304  005262  004737  005274                  JSR     PC,VTMD1A               ;LOAD CHAR.'S
2305  005266  104023                          THEND
2306  005270  000756                          BR      VTMOD1
2307  005272  000413                          BR      VTMOD2                  ;GO TO NEXT TEST IF SW. IS SET
2308
2309  005274  004737  015170          VTMD1A: JSR     PC,INTCHR               ;CREATE AND LOAD CHAR.
2310  005300  000775                          BR      .-4                     ;LOOP UNTIL LINE IS FULL
2311  005302  000240                          NOP                             ;END OF LINE RETURNS HERE
2312  005304  104007                          SETLNE                          ;SET UP '4' LINES USING ALL MODES
2313  005306  010037  024464                  MOV     R0,KSTOR1               ;SAVE ST. ADDR. FOR NEXT BUFFERED LINE
2314  005312  005237  024470                  INC     KSTOR3                  ;FINISHED?
2315  005316  001366                          BNE     VTMD1A                  ;NO, CREATE NEXT SET
2316  005320  000207                          RTS     PC                      ;YES, EXIT
2317                                    ;*************************************************************************
2318                                    ;VT 'CSR PRESET' TEST (3)
2319                                    ;THIS TEST DISPLAYS THE 'VT' CHARACTER SET IN ALL FIELD CONTROL MODES
2320                                    ;(BOLD,BLINK,BLANK & UNDERLINE). THE FIELD CONTROL IS SELECTED VIA THE
2321                                    ;'CSR' BEFORE EACH CHARACTER SET IS DISPLAYED.
2322                                    ;*************************************************************************
2323
2324  005322  012737  005322  020402  VTMOD2: MOV     #VTMOD2,RETURN
2325  005330  104000                          DISPLAY
2326  005332  023274                          MES26                           ;TEXT 'VT' CSR PRESET TEST'.
2327  005334  104005                          EOSBUF                          ;LOAD 'VT' BUFFER W/ 'EOS'.
2328  005336  104000                          DISPLAY
2329  005340  023572                          MES39                           ;TEXT 'BOLD PRESET TEST'
2330  005342  012777  002001  174140  VTMD2A: MOV     #2001,@VTCSR            ;SELECT 'BOLD' PRESET & GO
2331  005350  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
2332  005354  104000                          DISPLAY
2333  005356  023615                          MES40                           ;TEXT 'BLINK PRESET TEST'
2334  005360  012777  001001  174122          MOV     #1001,@VTCSR             ;SELECT 'BLINK' PRESET
2335  005366  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
2336  005372  104000                          DISPLAY
2337  005374  023641                          MES41                           ;TEXT 'BLANK PRESET TEST'
2338  005376  012777  000401  174104          MOV     #401,@VTCSR            ;SELECT 'BLANK' PRESET
2339  005404  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
2340  005410  104000                          DISPLAY
2341  005412  023665                          MES42                           ;TEXT 'UNDERLINE PRESET TEST'
2342  005414  012777  004001  174066          MOV     #4001,@VTCSR            ;SELECT 'UNDERLINE' PRESET
2343  005422  004737  005434                  JSR     PC,VTMD2B               ;LOAD CHAR.'S
```

```
2344  005426  104006                          ENDTST
2345  005430  000734                          BR      VTMOD2                  ;RE-CYCLE
2346  005432  000410                          BR      VTEOL                   ;BRANCH TO NEXT TEST IS SW. SET
2347
2348  005434  005037  024156          VTMD2B: CLR     INTSWH                  ;CLR SOFTWARE SW.
2349  005440  004737  015170                  JSR     PC,INTCHR               ;CREATE & DISPLAY CHAR.
2350  005444  000240                          NOP
2351  005446  000774                          BR      .-6                     ;LOOP UNTIL SCREEN IF FILLED
2352  005450  104020                          DELAYL                          ;6 SEC. DELAY
2353  005452  000207                          RTS     PC                      ;EXIT
2354                                  ;*************************************************************************
2355                                  ;VT 'END OF LINE' TEST (4)
2356                                  ;THIS TEST TESTS THAT THE 'END OF LINE' CHARACTER CAN BE PLACED IN
2357                                  ;ANY OF THE '1024' CHARACTER POSITIONS ON THE SCREEN. THIS IS DONE BY
2358                                  ;STARTING ALL 16 LINES WITH A 'VISIBLE END OF LINE' CHARACTER AND
2359                                  ;THEN INCREMENTING THESE CHARACTERS ACROSS THE SCREEN.
2360                                  ;*************************************************************************
2361
2362  005454  012737  005454  020402  VTEOL:  MOV     #VTEOL,RETURN
2363  005462  104000                          DISPLAY
2364  005464  023321                          MES27                           ;TEXT 'VT END OF LINE TEST'
2365  005466  104005                          EOSBUF                          ;LOAD 'VT' BUFFER & START
2366  005470  005001                          CLR     R1                      ;CLR LINE CNTR.
2367  005472  112720  000212          RESCNT: MOVB    #VISEOL,(R0)+           ;TERMINATE LINE W/ 'EOL'
2368  005476  012703  000077                  MOV     #63.,R3                 ;SAVE '63' BYTES BEFORE NEXT LINE
2369  005502  105020                          CLRB    (R0)+
2370  005504  005303                          DEC     R3
2371  005506  001375                          BNE     .-4
2372  005510  005201                          INC     R1                      ;INCREMENT LINE CNTR.
2373  005512  022701  000021                  CMP     #17.,R1                 ;DONE '16' LINES ?
2374  005516  001365                          BNE     RESCNT                  ;NO, CREATE NEXT LINE
2375  005520  104004                          DELAY                           ;DISPLAY THE 1ST 16 LINES OF 'EOL'
2376
2377                                  ;AT THIS POINT 16 LINES EACH CONTAINING 1 'EOL' AND 63 SPACES HAVE
2378                                  ;BEEN SET UP IN MEMORY. THE FOLLOWING SUBROUTINE GOES THRU ONE
2379                                  ;BY ONE REPLACING THE 'EOL' WITH AN '*' AND MOVING THE 'EOL'
2380                                  ;INTO THE NEXT SPACE UNTIL THE ENTIRE LINE HAS BEEN CHECKED.
2381
2382  005522  012705  000077                  MOV     #63.,R5                 ;SET UP FOR '63' SHIFTS
2383  005526  012704  000020          SRHEOL: MOV     #16.,R4
2384  005532  012700  024500                  MOV     #VTBUFF,R0              ;RESET BUFFER ADDR. POINTER
2385  005536  122027  000212          SHFEOL: CMPB    (R0)+,#VISEOL           ;SEARCH BUFFER FOR "EOL"
2386  005542  001375                          BNE     .-4
2387  005544  112740  000052                  MOVB    #52,-(R0)               ;REPLACE 'EOL' W/ '*'
2388  005550  105720                          TSTB    (R0)+                   ;RESET POINTER
2389  005552  112720  000212                  MOVB    #VISEOL,(R0)+           ;MOVE 'EOL' OVER '1'
2390  005556  005304                          DEC     R4                      ;DONE '16' LINES
2391  005560  001366                          BNE     SHFEOL                  ;NO, SHIFT NEXT 'EOL'
2392  005562  104004                          DELAY                           ;3 SEC. DISPLAY DELAY
2393  005564  005305                          DEC     R5                      ;DONE '63' SHIFTS
2394  005566  001357                          BNE     SRHEOL                  ;NO, LOOP AGAIN
2395  005570  104006                          ENDTST
2396  005572  000730                          BR      VTEOL                   ;YES, RESTART TEST
2397                                  ;*************************************************************************
2398                                  ;VT 'END OF SCREEN' TEST (5)
2399                                  ;THIS TEST TESTS THAT THE 'VISIBLE END OF SCEEEN' CHARACTER CAN
```

```
                                        ;BE PLACED IN ALL '1024' CHARACTER POSITIONS ON THE SCREEN. THIS
                                        ;IS DONE BY FIRST PRE-LOADING THE  BUFFER TO BE DISPLAYED WITH
                                        ;QUESTION MARKS. THE 'EOS' CHARACTER IS THEN INCREMENTED THRU THIS
                                        ;BUFFER REPLACING THE QUESTION MARKS BEHIND THE 'EOS' CHARACTER WITH
                                        ;DOTS.
                                        ;*************************************************************************

005574  012737  005574  020402  VTEOS:  MOV     #VTEOS, RETURN
005602  104000                          DISPLAY
005604  023345                          MES30                   ;TEXT 'VT END OF SCREEN TEST'
005606  104011                          PRELOAD                 ;PRE-LOAD THE BUFFER WITH '?'
005610     077                          .BYTE   77
005611     077                          .BYTE   77
005612  012701  002000                  MOV     #1024.,R1       ;SET UP CHAR. CNTR.
005616  012700  024500                  MOV     #VTBUFF,R0      ;SET UP BUFFER POINTER
005622  112710  000231                  MOVB    #VISEOS,(R0)    ;LOAD 1ST CHAR. WITH 'EOS'
005626  012777  000001  173654          MOV     #1,@VTCSR       ;ST DISPLAY
005634  104004                  SRHEOS: DELAY
005636  005301                          DEC     R1              ;DONE
005640  001405                          BEQ     ENDEOS          ;YES. EXIT
005642  012720  000056                  MOV     #56.,(R0)+      ;NO, REPLACE THE 'EOS' W/ DOT.
005646  112710  000231                  MOVB    #VISEOS,(R0)    ;MOVE 'EOS' OVER 'I' PLACE
005652  000770                          BR      SRHEOS
005654  104006                  ENDEOS: ENDTST
005656  000746                          BR      VTEOS           ;YES, RESTART TEST

                                        ;*************************************************************************
                                        ;VT 'BLANK CONTROL' TEST (6)
                                        ;THIS TEST INCREMENTS THE 'BLANK' CONTROL CHARACTER (361) THRU A
                                        ;FULL SCREEN BUFFER OF ?'S. THE SCREEN STARTS BLANK AND END'S
                                        ;UP FULL OF *'S.
                                        ;*************************************************************************

005660  012737  005660  020402  VTBLANK:MOV     #VTBLANK,RETURN
005666  104000                          DISPLAY
005670  023375                          MES31                   ;TEXT 'VT BLANKING' TEST
005672  104011                          PRELOAD                 ;PRELOAD DATA BUFFER
005674     077                          .BYTE   77              ;W/ ?'S
005675     077                          .BYTE   77
005676  112710  000361                  MOVB    #BLANK,(R0)     ;SET UP BLANK CONTROL CHAR.
005702  012701  002000                  MOV     #1024.,R1       ;SET UP CNTR
005706  012777  000001  173574          MOV     #1,@VTCSR       ;ST. DISPLAY
005714  104004                          DELAY                   ;3 SEC. DISPLAY DELAY
005716  112720  000052          MOVBLK: MOVB    #52,(R0)+       ;REPLACE, 'BLANK' CHAR. W/*
005722  112710  000361                  MOVB    #BLANK,(R0)     ;MOVE 'BLANK' CHAR. UP 'I'
005726  104004                          DELAY
005730  005301                          DEC     R1              ;MOVED 'BLANK' THUR BUFFER
005732  001371                          BNE     MOVBLK          ;NO
005734  104006                          ENDTST
005736  000750                          BR      VTBLANK         ;RESTART TEST

                                        ;*************************************************************************
                                        ;VT 'ALIGNMENT' TEST (7)
                                        ;THIS TEST DISPLAYS A FULL SCREEN OF THE CHAR. 'E' TO ENABLE A VISUAL
                                        ;ALIGNMENT OF THE 'VT'.
                                        ;*************************************************************************
```

```
2456
2457  005740  012737  005740  020402  VTALIN: MOV     #VTALIN,RETURN
2458  005746  104000                          DISPLAY
2459  005750  023425                          MES33                   ;TEXT 'VT' 'ALIGNMENT' TEST
2460  005752  104011                          PRELOAD                 ;PRELOAD 'VT' DATA BUFFER
2461  005754     105                          .BYTE   105             ;W/ E'S
2462  005755     105                          .BYTE   105
2463  005756  005277  173526                  INC     @VTCSR          ;ST. DISPLAY
2464  005762  104023                          THEND
2465
2466  005764  000765                          BR      VTALIN
2467                                ;*************************************************************************
2468                                ;VT 'FOCUS' TEST (10)
2469                                ;THIS TEST DISPLAYS A FULL SCREEN OF ALTERNATE *'S & 1'S TO ENABLE A VISUAL
2470                                ;FOCUS OF THE 'VT'.
2471                                ;*************************************************************************
2472
2473  005766  012737  005766  020402  VTFOCUS:MOV     #VTFOCUS,RETURN
2474  005774  104000                          DISPLAY
2475  005776  023451                          MES34                   ;TEXT 'VT FOCUS TEST'
2476  006000  104011                          PRELOAD                 ;LOAD 'VT' DATA BUFFER
2477  006002     052                          .BYTE   52              ;WITH *'S
2478  006003     061                          .BYTE   61              ;AND 1'S
2479  006004  005277  173500                  INC     @VTCSR          ;ST. DISPLAY
2480  005010  104023                          THEND
2481  006012  000765                          BR      VTFOCUS
2482
2483                                ;*************************************************************************
2484                                ;VT 'WORST CASE' TEST (11)
2485                                ;THIS TEST DISPLAYS A FULL SCREEN OF THE WORST  CASE PATTERN '125 & 252'.
2486                                ;*************************************************************************
2487
2488  006014  012737  006014  020402  VTWORST:MOV     #VTWORST,RETURN
2489  006022  104000                          DISPLAY
2490  006024  023471                          MES35                   ;TEXT 'VT WORST  CASE TEST'
2491  006026  104011                          PRELOAD                 ;LOAD 'VT' BUFFER W/ WORST CASE
2492  006030     125                          .BYTE   125             ;CHAR. 125
2493  006031     252                          .BYTE   252             ;CHAR. 252
2494  006032  005277  173452                  INC     @VTCSR          ;ST. DISPLAY
2495  006036  104023                          THEND
2496  006040  000765                          BR      VTWORST
2497                                ;*************************************************************************
2498                                ;VT 'CURSOR MOVEMENT' TEST (12)
2499                                ;THIS TEST TESTS THAT THE CURSOR CONTROL CHARACTER CAN BE PLACED
2500                                ;IN ANY POSITION ON THE SCREEN.THIS IS DONE BY DISPLAYING A 'CURSOR EOS'
2501                                ;BEING INCREMENTED ACROSS THE SCREEN. AS THE CURSOR IS INCREMENTED ALONG
2502                                ;ASTRICKS ARE FILLED IN BEHIND IT AND WHEN THE TEST IS COMPLETE THE
2503                                ;ENTIRE SCREEN WILL BE FILLED WITH *'S.
2504                                ;*************************************************************************
2505
2506  006042  012737  006042  020402  VTCURSR:MOV     #VTCURSR,RETURN
2507  006050  104000                          DISPLAY
2508  006052  023516                          MES36                   ;TEXT 'VT CURSOR MOVEMENT TEST'
2509  006054  104011                          PRELOAD                 ;PRE-LOAD THE BUFFER WITH '?'
2510  006056     077                          .BYTE   77
2511  006057     077                          .BYTE   77
```

```
2512  006060  012701  002000                  MOV     #1024.,R1       ;SET UP CHAR. CNTR.
2513  006064  012700  024500                  MOV     #VTBUFF,R0      ;SET UP BUFFER POINTER
2514  006070  010077  173410                  MOV     R0,@VTCAR       ;LOAD 'CURSOR' ADDRESS REG.
2515  006074  012710  000231                  MOV     #VISEOS,(R0)    ;LOAD 1ST CHAR. WITH 'EOS'
2516  006100  012777  000001  173402          MOV     #1,@VTCSR       ;ST DISPLAY
2517  006106  104004                  SHFCUR: DELAY
2518  006110  005301                          DEC     R1              ;DONE
2519  006112  001407                          BEQ     ENDCUR          ;YES, EXIT
2520  006114  012720  000056                  MOV     #56,(R0)+       ;NO, REPLACE THE 'EOS' W/ DOT.
2521  006120  010077  173360                  MOV     R0,@VTCAR       ;UPDATE 'CAR' TO STAY WITH 'EOS'
2522  006124  012710  000231                  MOV     #VISEOS,(R0)    ;MOVE 'EOS' OVER '1' PLACE
2523  006130  000766                          BR      SHFCUR
2524  006132  104006                  ENDCUR: ENDTST
2525  006134  000742                          BR      VTCURSR         ;YES, RESTART TEST
2526
2527                                  ;*************************************************************************
2528                                  ;VT 'ODD ADDRESS' TEST (13)
2529                                  ;THIS TEST SIMPLY DISPLAYS A MESSAGE WHICH IS LOADED FROM AN ODD
2530                                  ;STARTING ADDRESS.
2531                                  ;*************************************************************************
2532
2533  006136  012737  006136  020402  VTODD:  MOV     #VTODD, RETURN
2534  006144  104000                          DISPLAY
2535  006146  022104                          MES10                   ;TEXT 'VT ODD ADDRESS TEST'
2536  006150  005077  173334                  CLR     @VTCSR          ;STOP DISPLAY
2537  006154  012777  022255  173324          MOV     #MES11,@VTSAR   ;LOAD 'SAR' W/ MESSAGE FROM ODD ADDRESS.
2538  006162  005277  173322                  INC     @VTCSR          ;START & DISPLAY MESSAGE
2539  006166  104023                          THEND
2540  006170  000762                          BR      VTODD
2541  006172  000421                          BR      FKLOGIC         ;CONTINUE TO FK LOGIC TEST
2542                                  ;*************************************************************************
2543                                  ;FK11 FUNCTION KEYBOARD LOGIC TEST (15)
2544                                  ;THIS IS AN OPERATOR INTERVENTION TEST WHICH TESTS THE FUNCTION KEYBOARD LOGIC.
2545                                  ;REQUESTS ARE MADE FROM THE PROGRAM FOR INPUTS FROM THE FUNCTION KEYBOARD.
2546                                  ;*************************************************************************
2547
2548  006174  013704  001474          SETUPF: MOV     FKCSR,FKCSR4            ;MOVE 'CSR' ADDR. TO R4
2549  006200  005037  024450                  CLR     MESPRT                  ;CLR PRINT INHIBIT SW.
2550  006204  012737  000400  020376          MOV     #400,ICOUNT             ;INITIALIZE THE ITERATION COUNT
2551  006212  012737  017442  000034          MOV     #FKERR,@#34             ;LOAD 'TRAP' VECTOR FOR ERROR HANDLER
2552  006220  012737  006246  020402          MOV     #FKT1,RETURN            ;SET UP THE RETURN ADDRESS FOR SCOPE
2553  006226  012737  000001  024146          MOV     #1,TSTNUM               ;INIT TEST NO. CNTR.
2554  006234  000207                          RTS     PC
2555
2556  006236  104000                  FKLOGIC:DISPLAY
2557  006240  021245                          MES2                    ;TEXT 'FK LOGIC TEST;
2558  006242  004737  006174                  JSR     PC,SETUPF               ;INITIALIZE THE 'FK' LOGIC TEST
2559
2560                                  ;*************************************************************************
2561                                  ;ISSUE 'RESET' AND CHECK THAT THE 'CONTROL REG. IS INITIALIZED CORRECTLY
2562                                  ;*************************************************************************
2563
2564  006246  000240                  FKT1:   NOP                             ;Fk TEST '1'
2565  006250  000240                          NOP
2566  006252  000005                          RESET
2567  006254  005714                          TST     @FKCSR4                 ;'RESET' SHOULD OF CLEARED REGISTER
```

```
2568  006256  001401                  BEQ     FKT2                    ;BRANCH IF CLEARED
2569  006260  104400                  ERROR                           ;FK 'CSR' WASN'T CLEARED VIA ' RESET'
2570
2571                                  ;*************************************************************************
2572                                  ;TEST THAT THE 'ERROR', 'ASCII FLAG', & 'GO' BITS CAN BE WROTE TO '1'
2573                                  ;*************************************************************************
2574
2575  006262  104001  000002  FKT2:   SCOPE,2                         ;'FK' TEST 2
2576  006266  052714  177665          BIS     #177665,@FKCSR4         ;WRITE ALL 'CSR' BITS EXCEPT '6' TO 1'S
2577  006272  011400                  MOV     @FKCSR4,R0
2578  006274  042700  002000          BIC     #2000,R0                        ;CLR INTERVAL TIMING BIT
2579  006300  022700  100201          CMP     #100201,R0              ;TEST IF BITS '15,7,0' SET
2580  006304  001401                  BEQ     .+4
2581  006306  104400                  ERROR                           ;BITS '15,7&0' FAILED TO WRITE TO 1 IN 'CSR'
2582  006310  012714  000012          MOV     #12,@FKCSR4             ;CLR FLAGS
2583
2584                                  ;*************************************************************************
2585                                  ;TEST THAT THE  'GO' BIT CAN BE WROTE TO '0'
2586                                  ;*************************************************************************
2587
2588  006314  104001  000003  FKT3:   SCOPE,3                         ;'FK' TEST 3
2589  006320  052714  177665          BIS     #177665,@FKCSR4         ;WRITE ALL 'CSR' BITS TO '1'
2590  006324  005014                  CLR     @FKCSR4                 ;WRITE BITS TO '0'
2591  006326  032714  000001          BIT     #1,@FKCSR4              ;TEST IF 'GO' BIT CLEARED
2592  006332  001401                  BEQ     .+4
2593  006334  104400                  ERROR                           ;THE 'GO' BIT DIDN'T CLR
2594  006336  012714  000012          MOV     #12,@FKCSR4             ;CLR FLAGS
2595                                  ;*************************************************************************
2596                                  ;TEST THAT THE 'ERROR', 'ASCII FLAG' & 'GO' BITS CAN BE CLEARED VIA 'RESET'
2597                                  ;*************************************************************************
2598
2599  006342  104001  000004  FKT4:   SCOPE,4                         ;'FK' TEST 4
2600  006346  052714  177665          BIS     #177665,@FKCSR4         ;WRITE BITS TO 1
2601  006352  000005                  RESET
2602  006354  005714                  TST     @FKCSR4                 ;TEST IF 'RESET' CLEARED 'CSR'
2603  006356  001401                  BEQ     .+4
2604  006360  104400                  ERROR                           ;RESET FAILED TO CLR 'CSR'
2605
2606                                  ;*************************************************************************
2607                                  ;TEST THAT THE 'INTERVAL TIMER' FLAG DOESN'T SET IF 'GO' IS CLEARED
2608                                  ;*************************************************************************
2609
2610  006362  104001  000005  FKT5:   SCOPE,5                         ;'FK' TEST 5
2611  006366  012700  176000          MOV     #-2000,R0               ;SET UP WAIT LOOP
2612  006372  005200                  INC     R0                      ;WAIT FOR INTERVAL TIMER FLAG
2613  006374  001376                  BNE     .-2
2614  006376  032714  002000          BIT     #2000,@FKCSR4
2615  006402  001401                  BEQ     .+4
2616  006404  104400                  ERROR                           ;INTERVAL TIMER FLAG SET W/GO CLEARED.
2617  006406  012714  000012          MOV     #12,@FKCSR4             ;CLR FLAGS
2618
2619                                  ;*************************************************************************
2620                                  ;TEST THAT THE 'INTERVAL TIMER' FLAG WILL SET IF GO IS SET
2621                                  ;*************************************************************************
2622
2623  006412  104001  000006  FKT6:   SCOPE,6                         ;'FK' TEST 6
```

```
 2624  006416  012700  176000            MOV     #-2000,R0               ;SET UP WAIT LOOP
 2625  006422  052714  000001            BIS     #1,@FKCSR4              ;SET 'GO'
 2626  006426  005200                    INC     R0                      ;GIVE FLAG A CHANCE TO SET
 2627  006430  001376                    BNE     .-2
 2628  006432  032714  002000            BIT     #2000,@FKCSR4           ;TEST IF FLAG SET
 2629  006436  001001                    BNE     .+4
 2630  006440  104400                    ERROR                           ;INTERVAL TIMER FLAG DIDN'T SET W/GO SET
 2631  006442  012714  000012            MOV     #12,@FKCSR4             ;CLR FLAGS
 2632
 2633                                  ;*************************************************************************
 2634                                  ;TEST THAT THE INTERVAL TIMER FLAG CAN BE CLEARED VIA SETTING BIT 1
 2635                                  ;*************************************************************************
 2636
 2637  006446  104001  000007  FKT7:   SCOPE,7                         ;'FK' TEST 7
 2638  006452  052714  000001            BIS     #1,@FKCSR4              ;SET 'GO'
 2639  006456  032714  002000            BIT     #2000,@FKCSR4           ;WAIT FOR FLAG
 2640  006462  001775                    BEQ     .-4
 2641  006464  042714  000001            BIC     #1,@FKCSR4              ;CLR 'GO'
 2642  006470  052714  000002            BIS     #2,@FKCSR4              ;CLR INTERVAL TIMER FLAG
 2643  006474  032714  002000            BIT     #2000,@FKCSR4           ;SEE IF FLAG CLEARED
 2644  006500  001401                    BEQ     .+4
 2645  006502  104400                    ERROR                           ;SETTING BIT '1' DIDN'T CLR INTERVAL FLAG.
 2646                                  ;*************************************************************************
 2647                                  ;TEST THAT 'RESET' WILL CLR THE 'INTERVAL TIMER' FLAG.
 2648                                  ;*************************************************************************
 2649
 2650  006504  104001  000010  FK~10:  SCOPE,10                        ;'FK' TEST 10
 2651  006510  052714  000001            BIS     #1,@FKCSR4              ;SET 'GO'
 2652  006514  032714  002000            BIT     #2000,@FKCSR4           ;WAIT FOR FLAG
 2653  006520  001775                    BEQ     .-4
 2654  006522  000005                    RESET                           ;CLR FLAG W/RESET
 2655  006524  005714                    TST     @FKCSR4                 ;RESET SHOULD OF CLEARED REG.
 2656  006526  001401                    BEQ     .+4
 2657  006530  104400                    ERROR                           ;'RESET' DIDN'T CLR INTERVAL FLAG
 2658
 2659                                  ;*************************************************************************
 2660                                  ;TEST THAT THE INTERVAL TIMER FLAG CAUSES AN INTERRUPT WITH PROCESSOR PRIORITY @ 0
 2661                                  ;*************************************************************************
 2662
 2663  006532  104001  000011  FKT11:  SCOPE,11                        ;'FK' TEST 11
 2664  006536  012700  176000            MOV     #-2000,R0               ;SET UP A WAIT LOOP
 2665  006542  004737  016654            JSR     PC,LDFKVT               ;SET UP 'FK' INTR. ADDR.
 2666  006546  006560                    TAGA                            ;INTERRUPT SERVICE ADDR.
 2667  006550  005200                    INC     R0                      ;WAIT FOR INTERRUPT
 2668  006552  001376                    BNE     .-2
 2669  006554  104400                    ERROR                           ;INTERVAL TIMER FLAG DIDN'T CAUSE INTERRJPT
 2670  006556  000401                    BR      .+4
 2671  006560  022626  TAGA:   POP2SP                                   ;RESET THE STACK
 2672  006562  004737  016710            JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
 2673
 2674                                  ;*************************************************************************
 2675                                  ;TEST THAT 'NO' INTERRUPT OCCURRS IF 'GO' IS CLEARED
 2676                                  ;*************************************************************************
 2677
 2678  006566  104001  000012  FKT12:  SCOPE,12                        ;'FK' TEST 12
 2679  006572  012777  006630  172700    MOV     #TAGAB,@FKINT           ;SET UP THE INTERRUPT ADDRESS
```

```
2680  006600  012777  000340  172674          MOV     #340,@FKLVL
2681  006606  012700  176000                  MOV     #-2000,R0               ;SET UP A WAIT LOOP
2682  006612  005077  172474                  CLR     @PSW                    ;CLR PROC. PRIORITY
2683  006616  012714  000100                  MOV     #100,@FKCSR4            ;SET INTR. ENABLE
2684  006622  005200                          INC     R0
2685  006624  001376                          BNE     .-2
2686  006626  000402                          BR      .+6                     ;OK, NO INTERRUPT OCCURRED.
2687  006630  022626                  TAGAB:  POP2SP                          ;RESET THE STACK
2688  006632  104400                          ERROR                           ;INTERRUPT OCCURRED W/ 'GO' CLR
2689  006634  004737  016710                  JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
2690                                  ;*************************************************************
2691                                  ;TEST THAT 'NO' INTERRUPT OCCURS WITH PROC. @ PRIORITY '4'
2692                                  ;*************************************************************
2693
2694  006640  104001  000013          FKT13:  SCOPE,13                        ;'FK' TEST 13
2695  006644  012777  006704  172626          MOV     #TAGB,@FKINT            ;SET UP THE INTERRUPT ADDRESS
2696  006652  012777  000340  172622          MOV     #340,@FKLVL
2697  006660  012700  176000                  MOV     #-2000,R0               ;SET UP A WAIT LOOP
2698  006664  012777  000200  172420          MOV     #200,@PSW               ;SET PROC. @ PRIORITY '4'
2699  006672  012714  000101                  MOV     #101,@FKCSR4            ;SET INTR. ENABLE & GO.
2700  006676  005200                          INC     R0
2701  006700  001376                          BNE     .-2
2702  006702  000402                          BR      .+6                     ;OK, NO INTERRUPT OCCURRED.
2703  006704  022626                  TAGB:   POP2SP                          ;RESET THE STACK
2704  005706  104400                          ERROR                           ;INTERRUPT OCCURRED W/ PROC. @ PRIORITY '4'
2705  006710  004737  016710                  JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
2706                                  ;*************************************************************
2707                                  ;TEST THAT 'NO' INTERRUPT OCCURRES W/ PROC. @ PRIORITY '5'
2708                                  ;*************************************************************
2709
2710  006714  104001  000014          FKT14:  SCOPE,14                        ;'FK' TEST 14
2711  006720  012777  006760  172552          MOV     #TAGC,@FKINT            ;SET UP THE INTERRUPT VECTOR
2712  006726  012777  000340  172546          MOV     #340,@FKLVL
2713  006734  012700  176000                  MOV     #-2000,R0               ;SET UP WAIT LOOP
2714  006740  012777  000240  172344          MOV     #240,@PSW               ;SET PROC. @ PRIORITY '5'
2715  006746  012714  000101                  MOV     #101,@FKCSR4            ;SET INTER. ENABLE & GO
2716  006752  005200                          INC     R0
2717  006754  001376                          BNE     .-2
2718  006756  000402                          BR      .+6                     ;OK, NO INTERRUPT OCCURRED
2719  006760  022626                  TAGC:   POP2SP                          ;RESET STACK POINTER
2720  006762  104400                          ERROR                           ;FK INTERRUPTED WITH PROC. @ PRIORITY '5'
2721  006764  004737  016710                  JSR     PC,CLRFKV               ;CLR 'FK' INTR. ADDR.
2722
2723                                  ;*************************************************************
2724                                  ;TEST THAT THE INTERVAL TIMER FLAG IS SET APPROXIMATELY EVERY '500' USEC.
2725                                  ;*************************************************************
2726
2727  006770  104001  000015          FKT15:  SCOPE,15                        ;'FK' TEST 15
2728  006774  005001                          CLR     R1
2729  006776  004737  016654                  JSR     PC,LDFKVT               ;SET UP THE INTR. ADDR.
2730  007002  007010                          TAGFA
2731  007004  005201                          INC     R1
2732  007006  001376                          BNE     .-2
2733
2734  007010  022626                  TAGFA:  POP2SP                          ;RESET STACK
2735  007012  004737  016710                  JSR     PC,CLRFKV               ;CLR 'FK' VECTOR ADDR.
```

```
2736  007016  022701  000400          CMP     #400,R1                 ;CHECK LOW LIMIT
2737  007022  101402                  BLOS    .+6                     ;BRANCH IF HIGHER
2738  007024  104400                  ERROR                           ;INTERVAL CLOCK FREQ. TO FAST
2739  007026  000404                  BR      FKT16                   ;EXIT ON ERROR
2740  007030  022701  000600          CMP     #600,R1                 ;CHECK HIGH LIMIT
2741  007034  103001                  BHIS    .+4                     ;BRANCH IF LOWER
2742  007036  104400                  ERROR                           ;INTERVAL CLOCK FREQ. TO SLOW
2743                          ;*****************************************************************
2744                          ;TEST THAT 'CLR ASCII' (BIT3) WILL CLEAR THE KEYBOARD FLAG & ERROR BIT.
2745                          ;*****************************************************************
2746
2747  007040  104001  000016  FKT16:  SCOPE,16                        ;'FK' TEST 16
2748  007044  052714  100200          BIS     #100200,@FKCSR4         ;SET THE KEYBOARD FLAG
2749  007050  105714                  TSTB    @FKCSR4                 ;TEST THAT DONE WAS SET
2750  007052  100401                  BMI     .+4
2751  007054  104400                  ERROR                           ;'DONE' DIDN'T SET
2752  007056  052714  000010          BIS     #10,@FKCSR4             ;SET 'CLR FLAG'
2753  007062  032714  100200          BIT     #100200,@FKCSR4         ;TEST IF FLAGS CLEARED
2754  007066  001401                  BEQ     .+4
2755  007070  104400                  ERROR                           ;SETTING BIT 3 DIDN'T CLR ASCII FLAGS
2756
2757                          ;*****************************************************************
2758                          ;TEST THAT THE KEYBOARD FLAG IS CLEARED WHEN THE KEYBOARD DATA IS READ
2759                          ;*****************************************************************
2760
2761  007072  104001  000017  FKT17:  SCOPE,17                        ;'FK' TEST 17
2762  007076  052714  000201          BIS     #201,@FKCSR4            ;SET THE KEYBOARD FLAG
2763  007102  005777  172370          TST     @FKDATA                 ;READ KEYBOARD DATA
2764  007106  105714                  TSTB    @FKCSR4                 ;RE-TEST IF FLAG CLEARED
2765  007110  100001                  BPL     .+4
2766  007112  104400                  ERROR                           ;READING KEYBOARD DATA DIDN'T CLR FLAG
2767  007114  012714  000012          MOV     #12,@FKCSR4             ;CLR FLAGS
2768
2769                          ;*****************************************************************
2770                          ;TEST THAT THE 'ERROR' FLAG IS CLEARED WHEN THE KEYBOARD DATA IS READ.
2771                          ;*****************************************************************
2772
2773  007120  104001  000020  FKT20:  SCOPE,20                        ;'FK' TEST 20
2774  007124  052714  100001          BIS     #100001,@FKCSR4         ;SET THE ERROR BIT
2775  007130  005714                  TST     @FKCSR4                 ;TEST THAT ERROR SET
2776  007132  100401                  BMI     .+4
2777  007134  104400                  ERROR                           ;ERROR WASN'T SET
2778  007136  005777  172334          TST     @FKDATA                 ;READ THE KEYBOARD DATA
2779  007142  005714                  TST     @FKCSR4                 ;RE-TEST THE ERROR FLAG
2780  007144  100001                  BPL     .+4
2781  007146  104400                  ERROR                           ;READING KEYBOARD DATA DIDN'T CLR 'ERROR'
2782  007150  005014                  CLR     @FKCSR4                 ;CLR 'GO'
2783  007152  052714  000012          BIS     #12,@FKCSR4             ;CLR FLAGS
2784  007156  104001                  SCOPE
2785  007160  000020                  20
2786  007162  005037  020376          CLR     ICOUNT                  ;RUN FOLLOWING TESTS '1' PASS ONLY
2787                          ;*****************************************************************
2788                          ;TEST THAT THE KEYBOARD FLAG CAN BE SET VIA THE KEYBOARD.
2789                          ;*****************************************************************
2790
2791  007166  104001  000021  FKT21:  SCOPE,21                        ;'FK' TEST 21
```

```
007172  032777  002000  172124            BIT     #2000,@SWR              ;IS MANUAL INHIBIT SW. SET?
007200  001076                            BNE     FKT24+4                 ;YES, TEST COMPLETE
007202  012737  007210  020402            MOV     #TAGFH,RETURN           ;RE-SET SCOPE ADDRESS
007210  052714  000001          TAGFH:    BIS     #1,@FKCSR4              ;SET 'GO'
007214  104000                            DISPLAY
007216  021513                            MES6                            ;REQUEST CHAR.
007220  005077  172264                    CLR     @VTCSR                  ;CLEAR MESSAGE
007224  012700  177773                    MOV     #-5,R0
007230  005001                  TAGH:     CLR     R1
007232  005201                            INC     R1
007234  001376                            BNE     .-2
007236  005200                            INC     R0
007240  001373                            BNE     TAGH
007242  105714                            TSTB    @FKCSR4                 ;TEST IF KEYBOARD FLAG SET
007244  100401                            BMI     .+4
007246  104400                            ERROR                           ;KEYBOARD FLAG DIDN'T SET
007250  012714  000012                    MOV     #12,@FKCSR4             ;CLR FLAGS

                                ;*****************************************************************
                                ;TEST THAT THE 'ASCII' FLAG ISN'T SET IF 'GO' IS CLEARED
                                ;*****************************************************************

007254  104001  000022          FKT22:    SCOPE,22                        ;'FK' TEST 22
007260  104000                            DISPLAY
007262  021541                            MES6A                           ;REQUEST CHAR.
007264  005077  172220                    CLR     @VTCSR                  ;CLEAR MESSAGE
007270  012700  177773                    MOV     #-5,R0
007274  005001                  TAGAH:    CLR     R1
007276  005201                            INC     R1
007300  001376                            BNE     .-2
007302  005200                            INC     R0
007304  001373                            BNE     TAGAH
007306  105714                            TSTB    @FKCSR4                 ;TEST IF KEYBOARD FLAG SET
007310  100001                            BPL     .+4
007312  104400                            ERROR                           ;ASCII FLAG SET W/ 'GO'CLEARED
007314  012714  000012                    MOV     #12,@FKCSR4             ;CLR FLAGS
                                ;*****************************************************************
                                ;TEST THAT THE 'ERROR' FLAG IS SET IF THE KEYBOARD FLAG ISN'T CLEARED
                                ;*****************************************************************

007320  104001  000023          FKT23:    SCOPE,23                        ;'FK' TEST 23
007324  012714  000201                    MOV     #201,@FKCSR4            ; SET 'GO'
007330  104000                            DISPLAY
007332  021567                            MES6B                           ;REQUEST CHAR.
007334  005077  172150                    CLR     @VTCSR                  ;CLEAR MESSAGE
007340  012700  177773                    MOV     #-5,R0
007344  005001                  TAGI:     CLR     R1
007346  005201                            INC     R1
007350  001376                            BNE     .-2
007352  005200                            INC     R0
007354  001373                            BNE     TAGI
007356  005714                            TST     @FKCSR4                 ;TEST IF 'ERROR SET'
007360  100401                            BMI     .+4
007362  104400                            ERROR                           ;ERROR DIDN'T SET ON 2ND CHAR.
007364  005014                            CLR     @FKCSR4                 ;CLR 'GO'
007366  052714  000012                    BIS     #12,@FKCSR4             ;CLR FLAGS
```

```
 2848
 2849                           ;*************************************************************************
 2850                           ;END OF 'FK' LOGIC TEST.
 2851                           ;*************************************************************************
 2852
 2853  007372  104001  000024  FKT24:  SCOPE,24                        ;'FK' TEST 24
 2854  007376  104000                  DISPLAY
 2855  007400  021615                  MES7                            ;TEXT 'LOGIC OK;
 2856  007402  104006                  ENDTST                          ;END OF TEST
 2857  007404  000402                  BR      .+6
 2858  007406  000137  003122          JMP     VTLOGIC                 ;RESTART THE 'VT' LOGIC TEST
 2859  007412  000137  006236          JMP     FKLOGIC                 ;RE-RUN 'FK' LOGIC TEST
 2860                           ;*************************************************************************
 2861                           ;FK11 FUNCTION KEYBOARD LIGHT & SWITCH TEST. (16)
 2862                           ;THIS IS AN OPERATOR INTERVENTION TEST WHICH MONITORS THE FUNCTION KEYS.
 2863                           ;WHEN A FUNCTION KEY OR KEYS ARE STRUCK ITS CORRESPONDING FUNCTION LIGHT
 2864                           ;IS LIT.  IF THE KEY IS STRUCK A SECOND TIME THE CORRESPONDING FUNCTION
 2865                           ;LIGHT IS TURNED OFF.
 2866                           ;*************************************************************************
 2867
 2868  007416  012737  007416  023402  FKFUN:  MOV     #FKFUN, RETJRN
 2869  007424  104000                  DISPLAY
 2870  007426  021112                  MES1                            ;TEXT 'FK FUNCTION KEYBOARD LIGHT TEST'
 2871  007430  005000                  CLR     R0                      ;CLR WORKINGS REG'S
 2872  007432  005001                  CLR     R1
 2873  007434  005002                  CLR     R2
 2874  007436  004737  C16654          JSR     PC,LDFKVT               ;LOAD THE INTR. ADDR.
 2875  007442  007450                  SRRVFK                          ;INTR SERVICE ROUTINE
 2876  007444  000001                  WAIT                            ;WAIT FOR INTERRUPT
 2877  007446  000776                  BR      .-2
 2878
 2879                           ;ROUTINE TO 'SERVICE' FUNCTION KEYBOARD INTERRUPTS
 2880
 2881  007450  005202          SRRVFK: INC     R2                      ;INTR. CNTR. (EVERY 1000. LITE DATA UPDATED)
 2882  007452  105777  172016          TSTB    @FKCSR
 2883  007456  100004                  BPL     1$
 2884  007460  022777  000003  172010  CMP     #3,@FKDATA              ;CONTROL C TYPED?
 2885  007466  001436                  BEQ     EXITF1                  ;IF SO-EXITF1
 2886  007470  022702  001000  1$:     CMP     #1000,R2                        ;SERVICED '512' FLAGS?
 2887  007474  001023                  BNE     EXITF0                  ;NO, EXIT
 2888  007476  005002                  CLR     R2                      ;CLR CNTR.
 2889  007500  017704  171766          MOV     @FKLDA,R4               ;READ SWITCH DATA 'A'
 2890  007504  001406                  BEQ     SRVF0B                  ;SW. DATA PRESENT?
 2891  007506  010037  024442          MOV     R0,SAVOLD               ;SAVE 'OLD' KEYBOARD DATA
 2892  007512  040400                  BIC     R4,R0                   ;CLR ALL CORRESPONDING BITS IN 'OLD' NO.
 2893  007514  043704  024442          BIC     SAVOLD,R4               ;CLR ALL CORRESPONDING BITS IN 'NEW' NO.
 2894  007520  050400                  BIS     R4,R0                   ;SET REMAINING BITS FOR 'NEW' VALUE
 2895  007522  017704  171742  SRVF0B: MOV     @FKLDB,R4               ;READ SWITCH DATA 'B'
 2896  007526  001406                  BEQ     EXITF0                  ;SW. DATA PRESENT?
 2897  007530  010137  024442          MOV     R1,SAVOLD
 2898  007534  040401                  BIC     R4,R1                   ;CLR ALL CORRESPONDING BITS IN 'OLD' WRD.
 2899  007536  043704  024442          BIC     SAVOLD,R4               ;CLR ALL CORRESPONDING BITS IN 'NEW WRD.
 2900  007542  050401                  BIS     R4,R1                   ;UPDATE 'OLD' WRD WITH ' NEW' WRD.
 2901  007544  010077  171722  EXITF0: MOV     R0,@FKLDA               ;TRANSMIT UPDATED LIGHT DATA 'A'
 2902  007550  010177  171714          MOV     R1,@FKLDB               ;TRANSMIT UPDATED LIGHT DATA 'B'
 2903  007554  052777  000012  171712  BIS     #12,@FKCSR              ;CLR FLAGS.
```

```
 2904  007562  000002                          RTI
 2905  007564  052777  000012  171702  EXITF1: BIS     #12,@FKCSR
 2906  007572  104006                          ENDTST
 2907  007574  000710                          BR      FKFUN
 2908                                  ;*************************************************************************
 2909                                  ;KEYBOARD CHARACTER TEST (17)
 2910                                  ;THIS TEST VERIFIES THAT ALL CHARACTERS CAN BE TRANSMITTED VIA THE FK11 KEYBOARD.
 2911                                  ;WHEN THE TEST IS STARTED, THE TEXT "READY FOR INPUT" IS DISPLAYED ON THE 'VT'.
 2912                                  ;THE OPERATOR THEN TYPES IN THREE PASSES OF THE ENTIRE KEYBOARD
 2913                                  ;(LOWER CASE, SHIFT AND CONTROL) STARTING AT THE TOP LEFT HAND SIDE OF THE
 2914                                  ;KEYBOARD FOR EACH PASS.  EACH CHARACTER TRANSMITTED FROM THE KEYBOARD IS VERIFIED
 2915                                  ;AGAINST A CHARACTER BUFFER AND DISPLAYED ON THE 'VT'.  IF THE TRANSMITTED
 2916                                  ;CHARACTER DOESN'T MATCH THE EXPECTED CHARACTER AN AUDIO 'BEEP' IS HEARD
 2917                                  ;AND THE CHARACTER IS DISPLAYED IN BOLD UNDERLINED.  ALL CHARACTERS RECEIVED
 2918                                  ;FROM THIS POINT ON WILL BE TRANSMITTED IN THIS MODE UNTIL EITHER THE CORRECT
 2919                                  ;CHARACTER CODE IS RECEIVED OR DATA 'SW15' IS SET TO A '1' ALLOWING THE PROGRAM
 2920                                  ;TO CONTINUE.  DATA 'SW14' MAY ALSO BE SET TO A '1' AT ANYTIME TO LOOP ON
 2921                                  ;THE CURRENT CHARACTER.
 2922                                  ;*************************************************************************
 2923
 2924  007576  012737  007576  020402  FKCHAR: MOV     #FKCHAR,RETURN
 2925  007604  104000                          DISPLAY
 2926  007606  021317                          MES4                    ;TEXT 'FK CHARACTER TEST'
 2927  007610  012701  020457                  MOV     #CHRTAB-1,R1    ;SET UP CHAR. TABLE POINTER
 2928  007614  005037  024150                  CLR     FIELDSW         ;CLR FIELD CONTROL SOFTWARE SW.
 2929  007620  005037  024444                  CLR     TEMP1           ;SET UP SOFTWARE SW.'S
 2930  007624  005037  024446                  CLR     TEMP2
 2931  007630  005037  024464                  CLR     KSTOR1
 2932  007634  004737  016654                  JSR     PC,LDFKVT       ;LOAD 'FK' VECTOR ADDR.
 2933  007640  007646                          CHKFKO
 2934  007642  000001                          WAIT                    ;WAIT FOR 'FK' INTERRUPTS
 2935  007644  000776                          BR      .-2
 2936
 2937                                  ;ENTERED HERE ON 'FK' INTERRUPTS
 2938
 2939  007646  105777  171622          CHKFKO: TSTB    @FKCSR          ;ASCII FLAG SET?
 2940  007652  100071                          BPL     EXTCHR          ;NO, INTERVAL TIMER FLAG
 2941  007654  117702  171616                  MOVB    @FKDATA,R2      ;READ & SAVE DATA
 2942  007660  032777  040000  171436          BIT     #SW14,@SWR      ;IS THE LOOP SW. SET?
 2943  007666  001007                          BNE     CHKFK1+2        ;YES, DON'T UPDATE POINTER
 2944  007670  005737  024150                  TST     FIELDSW         ;NO, IS THE ERROR FLAG SET?
 2945  007674  001403                          BEQ     CHKFK1          ;NO, UPDATE POINTER
 2946  007676  005777  171422                  TST     @SWR            ;YES, IS SCOPE SW. SET?
 2947  007702  100001                          BPL     CHKFK1+2        ;YES, LOOP ON CURRENT CHAR.
 2948  007704  105721                  CHKFK1: TSTB    (R1)+           ;NO, UPDATE POINTER
 2949  007706  120211                          CMPB    R2,(R1)         ;DOES RECV'D CHAR.=EXPT'D CHAR.
 2950  007710  001073                          BNE     CHRERR          ;NO CHAR. ERROR
 2951  007712  005737  024150                  TST     FIELDSW         ;IS FIELD SW. SET?
 2952  007716  001404                          BEQ     CHKFK2          ;NO
 2953  007720  005037  024150                  CLR     FIELDSW         ;YES, CLR IT
 2954  007724  112720  000360                  MOVB    #360,(R0)+      ;CLR FIELD CONTROL CHAR.
 2955  007730  022702  000033          CHKFK2: CMP     #33,R2
 2956  007734  001403                          BEQ     .+10
 2957  007736  005737  024464                  TST     KSTOR1          ;DOING CONTROL PASS?
 2958  007742  001402                          BEQ     .+6             ;NO, CONTINUE
 2959  007744  052702  000100                  BIS     #100,R2         ;YES, ADD BIT SO CHAR. IS VISIBLE
```

```
 2960  007750  110220                          MOVB    R2,(R0)+        ;MOV RECV'D CHAR. TO 'VT' BUFFER.
 2961  007752  005737  024444                  TST     TEMP1
 2962  007756  001012                          BNE     CHKFK4
 2963  007760  022701  020545                  CMP     #UPCASE-1,R1    ;STARTING 'UPPER CASE' PASS?
 2964  007764  001007                          BNE     CHKFK4          ;NO
 2965  007766  104000                          DISPLAY
 2966  007770  023103                          MES22                   ;TEXT "INPUT UPPER CASE"
 2967  007772  012737  007576  024134          MOV     #FKCHAR,AVECTR  ;RE-SET 'tA' RESTART ADDR.
 2968  010000  005237  024444                  INC     TEMP1           ;SET PRINT INHIBIT SW.
 2969  010004  005737  024446          CHKFK4: TST     TEMP2
 2970  010010  001012                          BNE     EXTCHR
 2971  010012  022701  020633                  CMP     #CONTRL-1,R1    ;STARTING CONTROL PASS?
 2972  010016  001007                          BNE     EXTCHR          ;NO
 2973  010020  104000                          DISPLAY
 2974  010022  023147                          MES23                   ;TEXT "INPUT CONROL CHAR.S"
 2975  010024  012737  007576  024134          MOV     #FKCHAR,AVECTR  ;RE-SET 'tA' RESTART ADDR.
 2976  010032  005237  024446                  INC     TEMP2           ;SET DISPLAY INHIBIT SW.
 2977  010036  022701  020647          EXTCHR: CMP     #CNTRIT,R1      ;STARTING CONTROL PASS
 2978  010042  001002                          BNE     .+6             ;NO, CONTINUE
 2979  010044  005237  024464                  INC     kSTOR1          ;YES, SET UP TO DISPLAY INVISIBLE CHAR.
 2980  010050  052777  000002  171416          BIS     #2,@FKCSR       ;CLR INTERVAL TIMER FLAG
 2981  010056  022701  020722                  CMP     #CHREND+1,R1    ;FINISHED?
 2982  010062  001401                          BEQ     .+4             ;YES, RE-START TEST
 2983  010064  000002                          RTI                     ;EXIT
 2984  010066  004737  016710                  JSR     PC,CLRFKV       ;CLR INTERRUPT VECTOR
 2985  010072  022626                          POP2SP                  ;RESET STACK
 2986  010074  104006                          ENDTST
 2987  010076  000637                          BR      FKCHAR          ;RESTART TEST
 2988  010100  052777  000200  171402  CHRERR: BIS     #200,@VTCSR     ;ISSUE AUDIO BEEP ON ERROR
 2989  010106  005737  024150                  TST     FIELDSW         ;IS FIELD SW SET?
 2990  010112  001306                          BNE     CHKFK2          ;YES
 2991  010114  005237  024150                  INC     FIELDSW         ;NO, SET 'FIELD' SW.
 2992  010120  112720  000374                  MOVB    #374,(R0)+      ;SELECT 'BOLD-UNDERLINE'
 2993  010124  000701                          BR      CHkFK2          ;SET FIELD CONTROL SOFTWARE SW.
 2994                          ;*****************************************************************************
 2995                          ;FUNCTION KEYBOARD REPEATIBILITY TEST (20)
 2996                          ;THIS TEST VERIFIES THAT A SELECTED CHARACTER OR CHARACTERS (MAXIMUM OF TWO)
 2997                          ;ARE RECEIVED CORRECTLY FROM THE KEYBOARD.  WHEN THE 'REPEATIBILITY' TEST IS
 2998                          ;SELECTED THE PROGRAM HALTS.  LOAD THE CODE FOR THE CHARACTER TO BE VERIFIED
 2999                          ;IN THE DATA SWITCHES AND PRESS CONTINUE.  THE PROGRAM WILL HALT AGAIN.
 3000                          ;LOAD THE CODE FOR THE 2ND CHARACTER TO BE VERIFIED.  IF ONLY
 3001                          ;ONE CHARACTER IS BEING CHECKED, PRESS CONTINUE THE 2ND TIME WITHOUT CHANG-
 3002                          ;ING THE SWITCH SETTING.  AS INPUTTED CHARACTERS ARE VERIFIED THEY ARE
 3003                          ;ALSO DISPLAYED ON THE 'VT'.  IF AN ERROR IS DETECTED, THE AUDIO 'BEEP' IS
 3004                          ;SOUNDED AND THE RECEIVED CHARACTER IS DISPLAYED AS 'BOLD-UNDERLINED'.
 3005                          ;*****************************************************************************
 3006
 3007  010126  005737  024160          FKREPT: TST     TTYSWH          ;TTY AVAILABLE?
 3008  010132  001404                          BEQ     FKRPTA          ;NO, HALT TO GET CHAR. CODE
 3009  010134  104012                          PRINT                   ;YES,
 3010  010136  022512                          MES15                   ;REQUEST CHAR. CODES
 3011  010140  104015                          ASEMBL                  ;DECODE CODES
 3012  010142  000406                          BR      FkRPTB
 3013  010144  000000                  FKRPTA: HALT                    ;GET 1ST CHAR. FROM THE SWITCHES
 3014  010146  117703  171152                  MOVB    @SWR,R3         ;SAVE 1ST CHAR.
 3015  010152  000000                          HALT                    ;GET 2ND CHAR. FROM THE SWITCHES
```

```
3016  010154  117704  171144              MOVB    @SWR,R4         ;SAVE '2ND' CHAR.
3017  010160  042703  000200      FKRPTB: BIC     #200,R3
3018  010164  042704  000200              BIC     #200,R4
3019  010170  012737  010160  020402      MOV     #FKRPTB,RETURN
3020  010176  104000                      DISPLAY
3021  010200  022437                      MES14                   ;TEXT 'FK REPEATIBILITY TEST'
3022  010202  012737  010126  024134      MOV     #FKREPT,AVECTR
3023  010210  005037  024150              CLR     FIELDSW         ;CLR FIELD CONTROL SOFTWARE SW.
3024  010214  005005                      CLR     R5              ;CLR WORKING REG.
3025  010216  004737  017062              JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
3026  010222  004737  016654              JSR     PC,LDFKVT       ;LOAD 'FK' INTR. ADDR.
3027  010226  010234                      FREPTO
3028  010230  000001                      WAIT                    ;WAIT FOR 'FK' INTR.
3029  010232  000776                      BR      .-2
3030                          ;ENTERED HERE ON 'FK' INTERRUPTS FOR THE REPEATIBILITY TEST
3031
3032  010234  105777  171234      FREPTO: TSTB    @FKCSR          ;ASCII FLAG SET?
3033  010240  100035                      BPL     EXREPT          ;NO, INTERVAL TIMER FLAG
3034  010242  117701  171230              MOVB    @FKDATA,R1      ;YES, READ & SAVE DATA
3035  010246  005705                      TST     R5              ;COMPARE 1ST OR 2ND CHAR.?
3036  010250  001003                      BNE     FREPT1          ;2ND
3037  010252  120103                      CMPB    R1,R3           ;RECV'D = EXPT'D CHAR.?
3038  010254  001416                      BEQ     REPTOK          ;YES
3039  010256  000402                      BR      REPTER          ;NO, SET UP ERROR
3040  010260  120104      FREPT1: CMPB    R1,R4           ;RECV'D = EXPT'D CHAR.?
3041  010262  001413                      BEQ     REPTOK          ;YES
3042  010264  052777  000200  171216  REPTER: BIS     #AUDIO, @VTCSR  ;ISSUE AUDIO BEEP
3043  010272  005737  024150              TST     FIELDSW         ;FIELD SW. SET?
3044  010276  001014                      BNE     FREPT2          ;YES
3045  010300  005237  024150              INC     FIELDSW         ;NO, SET IT
3046  010304  112720  000374              MOVB    #374,(R0)+      ;SEND BAD CHAR. AS 'BOLD-UNDERLINED'
3047  010310  000407                      BR      FREPT2
3048  010312  005737  024150      REPTOK: TST     FIELDSW         ;FIELD SW. SET
3049  010316  001404                      BEQ     FREPT2          ;YES
3050  010320  005037  024150              CLR     FIELDSW         ;NO, CLR IT
3051  010324  112720  000360              MOVB    #360,(R0)+      ;CLR FIELD CONTROL CHAR.
3052  010330  005105      FREPT2: COM     R5              ;SET UP SOFTWARE SW.
3053  010332  110120                      MOVB    R1,(R0)+        ;TRANSMIT RECV'D CHAR.
3054  010334  052777  000002  171132  EXREPT: BIS     #2,@FKCSR       ;CLR INTERVAL TIMER FLAG
3055  010342  000002                      RTI
3056                          ;*************************************************************************
3057                          ;VT20 SYSTEM EXERCISER TEST. (TEST 21)
3058                          ;THIS TEST IS DESIGNED TO EXERCISE THE VT20 IN A SYSTEM ENVIRONMENT. THE
3059                          ;TEST UTILIZES THE VT20 (DISPLAYS & KEYBOARDS) AND ALL ASSOCIATED DL11'S
3060                          ;IN CONJUNCTION WITH A HOST COMPUTER. TO RUN THIS TEST THE DL11 DATA
3061                          ;HANDLING ROUTINE MUST BE LOADED INTO THE HOST COMPUTER. THIS ROUTINE
3062                          ;RECEIVES AND RE-TRANSMITS DATA TO VT20 BEING TESTED. WHEN THE SYSTEM TEST
3063                          ;IS STARTED THE PROGRAM WAITS FOR CHARACTERS TO BE INPUTTED FROM THE
3064                          ;KEYBOARD.  AS THESE CHARACTERS ARE RECEIVED THEY ARE STORED IN A VT BUFFER
3065                          ;AND DISPLAYED. THE PROGRAM WILL ACCEPT UP TO 384 CHARACTERS (6 COMPLETE
3066                          ;LINES) FROM THE KEYBOARD. THESE 384 CHARACTERS, OR ANY PART THERE OF,
3067                          ;CAN BE AUTOMATICALLY GENERATED BY TYPING A '↑A'. THE OPERATOR MAY AT
3068                          ;THIS TIME OR ANYTIME PRIOR TO THIS, TYPE 'ALTMODE'. THIS WILL ENABLE
3069                          ;THE INPUTTED CHARACTERS TO BE TRANSMITTED TO THE HOST COMPUTER.
3070                          ;UPON RECEIVING THE COMPLETE DATA TRANSFER, THE HOST COMPUTER WILL
3071                          ;RE-TRANSMIT THE DATA BACK TO THE VT20. THE RECEIVED DATA IS THEN
```

```
3072                                    ;VERIFIED AGAINST THE DATA ORGINALLY TRANSMITTED AND THEN DISPLAYED.
3073                                    ;IF THE RECEIVED DATA DOESN'T MATCH THE TRANSMITTED DATA, THE RECEIVED
3074                                    ;CHACACTER IS DISPLAYED AS 'BOLD-UNDERLINED'.
3075                                    ;THE SYSTEM TEST USES AN ERROR REPORTING SCHEME OF DISPLAYING ALL
3076                                    ;ERRORS AS THEY ARE ENCOUNTERED AS UNIQUE CURSOR CHARACTERS. THESE
3077                                    ;CHARACTERS IDENTIFY THE PARTICULAR ERROR DETECTED. TYPING A '↑C' AT
3078                                    ;ANYTIME WILL ENABLE THE TEST TO BE RE-STARTED.
3079                                    ;TYPING A '↑T' AFTER THE DATA HAS BEEN INPUTTED WILL ENABLE
3080                                    ;FOR A CONTINUOUS TRANSFER OF DATA.  TYPING '↑T' A SECOND TIME
3081                                    ;WILL STOP THE DATA TRANSFER (ON COMPLETION OF CURRENT TRANSFER)
3082                                    ;AND ALLOW THE PROGRAM TO RESUME NORMAL OPERATION.
3083                                    ;*************************************************************************
3084
3085  010344  012737  010344  020402  SYSTST: MOV     #SYSTST,RETURN
3086  010352  000005                         RESET
3087  010354  012737  010344  024134         MOV     #SYSTST,AVECTR          ;SET UP THE RESTART ADDR.
3088  010362  012777  021656  171022         MOV     #MES9,@VTOSAR           ;DISPLAY TEST HEADER ON UNIT '0'
3089  010370  012777  021656  171030         MOV     #MES9,@VT1SAR           ;DISPLAY TEST HEADER ON UNIT '1'
3090  010376  022737  000004  024142         CMP     #4,     UNITNO  ;4 UNITS?
3091  010404  001013                         BNE     2$
3092  010406  012777  021656  171042         MOV     #MES9,  @VT3SAR ;DISPLAY TEST HEADER ON UNIT '3'
3093  010414  005277  171040                 INC     @VT3CSR
3094  010420  012777  021656  171014  1$:    MOV     #MES9,  @VT2SAR ;DISPLAY TEST HEADER ON UNIT '2'
3095  010426  005277  171012                 INC     @VT2CSR
3096  010432  000404                         BR      3$
3097
3098  010434  022737  000003  024142  2$:    CMP     #3,     UNITNO  ;3 UNITS?
3099  010442  001766                         BEQ     1$
3100  010444  005277  170744          3$:    INC     @VTOCSR
3101  010450  005277  170754                 INC     @VT1CSR
3102  010454  104020                         DELAYL
3103  010456  104020                         DELAYL
3104  010460  104005                         EOSBUF                          ;RE-LOAD DATA BUFFER W/ 'EOS'
3105  010462  012777  000340  170622  SYSIT1: MOV     #340,@PSW               ;SET PROC. PRIO. @7
3106  010470  012700  024174                 MOV     #RECSWO,R0              ;SET UP TO CLR SOFTWARE USER SW.'S
3107  010474  005020                         CLR     (R0)+
3108  010476  022700  024430                 CMP     #START,R0               ;DONE?
3109  010502  001374                         BNE     .-6                     ;NO
3110  010504  012777  011766  171026         MOV     #RECV0,@RINT0           ;LOAD DL11 REC. VECTOR ADDR.'S
3111  010512  012777  000200  171022         MOV     #200,@RLVL0             ;BR LEVEL '4'
3112  010520  012777  012000  171016         MOV     #RECV1,@RINT1
3113  010526  012777  000200  171012         MOV     #200,@RLVL1
3114  010534  012777  011706  171036         MOV     #TRAN0,@XINT0           ;LOAD DL11 TRANS. VECTOR ADDR.'S
3115  010542  012777  000200  171032         MOV     #200,@XLVL0
3116  010550  012777  011722  171026         MOV     #TRAN1,@XINT1
3117  010556  012777  000200  171022         MOV     #200,@XLVL1
3118  010564  012777  011626  170546         MOV     #SERFK0,@FK0INT         ;LOAD 'FK' VECTOR ADDRESSES
3119  010572  012777  000200  170542         MOV     #200,@FK0LVL
3120  010600  012777  011642  170546         MOV     #SERFK1,@FK1INT
3121  010606  012777  000200  170542         MOV     #200,@FK1LVL
3122  010614  012700  024500                 MOV     #VTBUFF,R0              ;LOAD 'VT' BUFFER POINTERS
3123  010620  010002                         MOV     R0,R2
3124  010622  004737  011610                 JSR     PC,DISPMS               ;DISPLAY HEADER MESSAGE
3125  010626  012702  026450                 MOV     #VTBUFF+1000.,R2
3126  010632  010077  170554                 MOV     R0,@VTOSAR              ;LOAD 'VT' STARTING ADDR. REG.'S
3127  010636  010277  170564                 MOV     R2,@VT1SAR
```

```
3128  010642  004737  011610                    JSR     PC,DISPMS
3129  010646  012700  024567                    MOV     #VTBUFF+55.,R0
3130  010652  010037  024410                    MOV     R0,BUF0R0               ;R0 BUFFER POINTER FOR UNIT '0'
3131  010656  010037  024420                    MOV     R0,BUF0R1               ;R1 BUFFER POINTER FOR UNIT '0'
3132  010662  012702  026537                    MOV     #VTBUFF+1055.,R2
3133  010666  010237  024412                    MOV     R2,BUF1R0               ;R0 BUFFER POINTER FOR UNIT '1'
3134  010672  010237  024422                    MOV     R2,BUF1R1               ;R1 BUFFER POINTER FOR UNIT '1'
3135  010676  012737  010726  000004            MOV     #DL11A,@#4      ;SET UP TIME-OUT FOR MISSING DL'S
3136  010704  012737  000340  000006            MOV     #340,@#6
3137  010712  012777  000100  170600            MOV     #100,@RCSR0             ;ENABLE DL11 REC. INTR'S
3138  010720  012777  000100  170632            MOV     #100,@XCSR0             ;ENABLE DL11 TRANS, INTR.'S
3139  010726  012737  010750  000004  DL11A:    MOV     #DL11B,@#4
3140  010734  012777  000100  170560            MOV     #100,@RCSR1
3141  010742  012777  000100  170612            MOV     #100,@XCSR1
3142  010750  022737  000004  024142  DL11B:    CMP     #4,UNITNO               ;4 UNITS?
3143  010756  001133                            BNE     TAG2X                   ;NO, TEST FOR 3
3144  010760  012777  012024  170566            MOV     #RECV3, @RINT3          ;LOAD DL11 REC VECTOR FOR UNIT '3'
3145  010766  012777  000200  170562            MOV     #200,@RLVL3             ;BR LEVEL '4'
3146  010774  012777  011752  170612            MOV     #TRAN3,@XINT3           ;LOAD DL11 XMIT VECTOR FOR UNIT '3'
3147  011002  012777  000200  170606            MOV     #200,@XLVL3             ;BE LEVEL '4'
3148  011010  012777  011672  170366            MOV     #SERFK3,@FK3INT         ;LOAD FK VECTOR FOR UNIT '3'
3149  011016  012777  000200  170362            MOV     #200,@FK3LVL            ;BR LEVEL '4'
3150  011024  012702  032370                    MOV     #VTBUFF+3000.,R2
3151  011030  010277  170422                    MOV     R2,@VT3SAR              ;LOAD 'VT'3 STARTING ADDR. REG
3152  011034  004737  011610                    JSR     PC,DISPMS
3153  011040  012702  032457                    MOV     #VTBUFF+3055.,R2
3154  011044  010237  024416                    MOV     R2,     BUF3R0          ;BUFFER POINTER FOR UNIT '3'
3155  011050  010237  024426                    MOV     R2,     BUF3R1
3156  011054  012737  011076  000004            MOV     #DL11C,@#4
3157  011062  012777  000100  170436            MOV     #100,@RCSR3             ;ENABLE DL11 FOR UNIT '3' TO INTERRUPT
3158  011070  012777  000100  170470            MOV     #100,   @XCSR3
3159  011076  012777  000101  170354  DL11C:    MOV     #101,@VT3CSR            ;START VT
3160  011104  012777  000101  170266            MOV     #101,@FK3CSR            ;ENABLE FK INTERRUPT
3161  011112  012777  012012  170430  TAG1X:    MOV     #RECV2,@RINT2           ;LOAD DL11 REC VECTOR FOR UNIT '2'
3162  011120  012777  000200  170424            MOV     #200,@RLVL2             ;BR LEVEL '4'
3163  011126  012777  011736  170454            MOV     #TRAN2,@XINT2           ;LOAD DL11 XMIT VECTOR FOR UNIT '2'
3164  011134  012777  000200  170450            MOV     #200,@XLVL2             ;BR LEVEL '4'
3165  011142  012777  011656  170220            MOV     #SERFK2,@FK2INT         ;LOAD FK VECTOR FOR UNIT '2'
3166  011150  012777  000200  170214            MOV     #200,@FK2LVL            ;BR LEVEL '4'
3167  011156  012702  030420                    MOV     #VTBUFF+2000.,R2
3168  011162  010277  170254                    MOV     R2,@VT2SAR              ;LOAD 'VT'2 STARTING ADDR. REG
3169  011166  004737  011610                    JSR     PC,DISPMS
3170  011172  012702  030507                    MOV     #VTBUFF+2055.,R2
3171  011176  010237  024414                    MOV     R2,BUF2R0               ;BUFFER POINTER FOR UNIT '2'
3172  011202  010237  024424                    MOV     R2,BUF2R1
3173  011206  012737  011230  000004            MOV     #DL11D,@#4
3174  011214  012777  000100  170302            MOV     #100,   @RCSR2  ;ENABLE DL11 FOR UNIT '2' TO INTERRUPT
3175  011222  012777  000100  170334            MOV     #100,@XCSR2
3176  011230  012777  000101  170206  DL11D:    MOV     #101,@VT2CSR            ;START VT
3177  011236  012777  000101  170120            MOV     #101,@FK2CSR            ;ENABLE FK INTERRUPTS
3178  011244  000404                            BR      TAG3X
3179
3180  011246  022737  000003  024142  TAG2X:    CMP     #3,UNITNO               ;3 UNITS?
3181  011254  001716                            BEQ     TAG1X                   ;YES, SET UP UNIT 2
3182  011256  012737  000006  000004  TAG3X:    MOV     #6,@#4                  ;RESET THE TIMEOUT VECTOR
3183  011264  012737  000004  000006            MOV     #4,@#6
```

```
3184  011272  012777  000101  170114          MOV     #101,@VTOCSR            ;START VT'S
3185  011300  012777  000101  170122          MOV     #101,@VT1CSR
3186  011306  012777  000101  170020          MOV     #101,@FK0CSR            ;ENABLE 'FK' INTERRUPTS
3187  011314  012777  000101  170026          MOV     #101,@FK1CSR
3188  011322  005077  167764                  CLR     @PSW                    ;SET PROC. PRIO. @0
3189
3190  011326  012700  024510          SERVCE: MOV     #VTBUFF+08.,R0
3191  011332  013702  024234                  MOV     CHRCT0,R2
3192  011336  104024                          BINDEC                          ;UPDATE THE CHARACTER COUNTER
3193  011340  012700  024526                  MOV     #VTBUFF+22.,R0
3194  011344  013702  024320                  MOV     RECTR0,R2
3195  011350  104024                          BINDEC                          ;UPDTAE THE RECV'D CHAR. COUNTER
3196  011352  012700  024544                  MOV     #VTBUFF+36.,R0
3197  011356  013702  024310                  MOV     TRANF0,R2
3198  011362  104024                          BINDEC                          ;UPDATE THE TRANSFER COUNTER
3199  011364  012700  024561                  MOV     #VTBUFF+49.,R0
3200  011370  013702  024340                  MOV     ERFLG0,R2
3201  011374  104024                          BINDEC                          ;UPDATE THE ERROR COUNTER
3202  011376  012700  026460                  MOV     #VTBUFF+1008.,R0
3203  011402  013702  024236                  MOV     CHRCT1,R2
3204  011406  104024                          BINDEC
3205  011410  012700  026476                  MOV     #VTBUFF+1022.,R0
3206  011414  013702  024322                  MOV     RECTR1,R2
3207  011420  104024                          BINDEC
3208  011422  012700  026514                  MOV     #VTBUFF+1036.,R0
3209  011426  013702  024312                  MOV     TRANF1,R2
3210  011432  104024                          BINDEC
3211  011434  012700  026531                  MOV     #VTBUFF+1049.,R0
3212  011440  013702  024342                  MOV     ERFLG1,R2
3213  011444  104024                          BINDEC
3214  011446  023727  024142  000003          CMP     UNITNO,#3               ;RUNNING '3' UNITS ?
3215  011454  002724                          BLT     SERVCE                  ;NO. RETURN TO SERVICE
3216  011456  012700  030430                  MOV     #VTBUFF+2008.,R0
3217  011462  013702  024240                  MOV     CHRCT2,R2
3218  011466  104024                          BINDEC                          ;UPDATE THE CHARACTER COUNTER
3219  011470  012700  030446                  MOV     #VTBUFF+2022.,R0
3220  011474  013702  024324                  MOV     RECTR2,R2
3221  011500  104024                          BINDEC                          ;UPDTAE THE RECV'D CHAR. COUNTER
3222  011502  012700  030464                  MOV     #VTBUFF+2036.,R0
3223  011506  013702  024314                  MOV     TRANF2,R2
3224  011512  104024                          BINDEC                          ;UPDATE THE TRANSFER COUNTER
3225  011514  012700  030501                  MOV     #VTBUFF+2049.,R0
3226  011520  013702  024344                  MOV     ERFLG2,R2
3227  011524  104024                          BINDEC                          ;UPDATE THE ERROR COUNTER
3228  011526  023727  024142  000004          CMP     UNITNO,#4               ;RUNNING '4' UNITS ?
3229  011534  002674                          BLT     SERVCE                  ;NO. RETURN TO SERVICE
3230  011536  012700  032400                  MOV     #VTBUFF+3008.,R0
3231  011542  013702  024242                  MOV     CHRCT3,R2
3232  011546  104024                          BINDEC
3233  011550  012700  032416                  MOV     #VTBUFF+3022.,R0
3234  011554  013702  024326                  MOV     RECTR3,R2
3235  011560  104024                          BINDEC
3236  011562  012700  032434                  MOV     #VTBUFF+3036.,R0
3237  011566  013702  024316                  MOV     TRANF3,R2
3238  011572  104024                          BINDEC
3239  011574  012700  032451                  MOV     #VTBUFF+3049.,R0
```

```
3240  011600  013702  024346          MOV     ERFLG3,R2
3241  011604  104024                  BINDEC
3242  011606  000647                  BR      SERVCE
3243
3244                                ;ENTERED HERE TO MOVE A MESSAGE INTO THE VT BUFFER
3245
3246  011610  012701  022164  DISPMS: MOV     #MES80,R1
3247  011614  111122                  MOVB    (R1),(R2)+
3248  011616  122721  000031          CMPB    #EOS,(R1)+
3249  011622  001374                  BNE     DISPMS+4
3250  011624  000207                  RTS     PC
3251                                ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '0'
3252
3253  011626  004737  012036  SERFK0: JSR     PC,SETBF0               ;SET UP BUFFER POINTERS
3254  011632  004737  012444          JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3255  011636  000137  012272          JMP     SAVBF0                  ;SAVE BUFFER POINTERS ON RETURN
3256
3257                                ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '1'
3258
3259  011642  004737  012102  SERFK1: JSR     PC,SETBF1               ;SET UP BUFFER POINTERS
3260  011646  004737  012444          JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3261  011652  000137  012316          JMP     SAVBF1                  ;SAVE BUFFER POINTERS ON RETURN
3262
3263                                ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '2'
3264
3265  011656  004737  012152  SERFK2: JSR     PC,SETBF2               ;SET UP BUFFER POINTERS
3266  011662  004737  012444          JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3267  011666  000137  012342          JMP     SAVBF2                  ;SAVE BUFFER POINTERS
3268
3269                                ;ENTERED HERE ON 'FK' INTERRUPTS FROM UNIT '3'
3270
3271  011672  004737  012222  SERFK3: JSR     PC,SETBF3               ;SET UP BUFFER POINTERS
3272  011676  004737  012444          JSR     PC,SERVFK               ;SERVICE 'FK' INTERRUPT
3273  011702  000137  012366          JMP     SAVBF3                  ;SAVE BUFFER POINTERS
3274                                ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '0'
3275
3276  011706  004737  012036  TRAN0:  JSR     PC,SETBF0               ;SET UP BUFFER POINTERS
3277  011712  004737  013130          JSR     PC,TRNSMT
3278  011716  000137  012272          JMP     SAVBF0
3279
3280                                ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '1'
3281
3282  011722  004737  012102  TRAN1:  JSR     PC,SETBF1               ;SET UP BUFFER POINTERS
3283  011726  004737  013130          JSR     PC,TRNSMT
3284  011732  000137  012316          JMP     SAVBF1
3285
3286                                ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '2'
3287
3288  011736  004737  012152  TRAN2:  JSR     PC,SETBF2               ;SET UP BUFFER POINTERS
3289  011742  004737  013130          JSR     PC,TRNSMT               ;SERVICE TRANSMITTER INTERRUPT
3290  011746  000137  012342          JMP     SAVBF2
3291
3292                                ;ENTERED HERE ON DL11 TRANSMITTER INTERRUPTS FROM UNIT '3'
3293
3294  011752  004737  012222  TRAN3:  JSR     PC,SETBF3               ;SET UP BUFFER POINTERS
3295  011756  004737  013130          JSR     PC,TRNSMT
```

```
  3296  011762  000137  012366                    JMP     SAVBF3
  3297
  3298                                    ;ENTERED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '0'
  3299
  3300  011766  004737  012036            RECV0:  JSR     PC,SETBF0
  3301  011772  004737  013362                    JSR     PC,RECVIT
  3302  011776  000535                             BR      SAVBF0
  3303                                    ;ENTERED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '1'
  3304
  3305  012000  004737  012102            RECV1:  JSR     PC,SETBF1
  3306  012004  004737  013362                    JSR     PC,RECVIT
  3307  012010  000542                             BR      SAVBF1
  3308
  3309                                    ;ENTERED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '2'
  3310
  3311  012012  004737  012152            RECV2:  JSR     PC,SETBF2               ;SET UP BUFFER POINTERS
  3312  012016  004737  013362                    JSR     PC,RECVIT               ;SERVICE RECEIVER INTERRUPT
  3313  012022  000547                             BR      SAVBF2
  3314
  3315                                    ;ENTERED HERE ON DL11 RECEIVER INTERRUPTS FROM UNIT '3'
  3316
  3317  012024  004737  012222            RECV3:  JSR     PC,SETBF3               ;SET UP BUFFER POINTERS
  3318  012030  004737  013362                    JSR     PC,RECVIT               ;SERVICE RECEIVER INTERRUPT
  3319  012034  000554                             BR      SAVBF3
  3320
  3321                                    ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '0'
  3322
  3323  012036  012637  024464            SETBF0: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
  3324  012042  010046                             MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
  3325  012044  010146                             MOV     R1,-(SP)
  3326  012046  010246                             MOV     R2,-(SP)
  3327  012050  010346                             MOV     R3,-(SP)
  3328  012052  010446                             MOV     R4,-(SP)
  3329  012054  013700  024410                     MOV     BUFOR0,R0
  3330  012060  013701  024420                     MOV     BUFOR1,R1
  3331  012064  005002                             CLR     R2                      ;R2 IS INDEX OFFSET FOR ADDRESSSING
  3332  012066  005003                             CLR     R3                      ;R3 IS INDEX OFFSET FOR DEVICE ADDRESSING
  3333  012070  012737  024567  024430            MOV     #VTBUFF+55.,START
  3334  012076  000177  012362                     JMP     @KSTOR1                 ;RETURN TO CALL
  3335
  3336                                    ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '1'
  3337
  3338  012102  012637  024464            SETBF1: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
  3339  012106  010046                             MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
  3340  012110  010146                             MOV     R1,-(SP)
  3341  012112  010246                             MOV     R2,-(SP)
  3342  012114  010346                             MOV     R3,-(SP)
  3343  012116  010446                             MOV     R4,-(SP)
  3344  012120  013700  024412                     MOV     BUF1R0,R0
  3345  012124  013701  024422                     MOV     BUF1R1,R1
  3346  012130  012702  000002                     MOV     #2,R2                   ;R2 IS INDEX OFFSET FOR ADDRESSSING
  3347  012134  012703  000014                     MOV     #12.,R3                 ;R3 IS INDEX OFFSET FOR DEVICE ADDRESSING
  3348  012140  012737  026537  024430            MOV     #VTBUFF+1055.,START
  3349  012146  000177  012312                     JMP     @KSTOR1                 ;RETURN TO CALL
  3350
  3351                                    ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '2'
```

```
3352
3353  012152  012637  024464          SETBF2: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
3354  012156  010046                          MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
3355  012160  010146                          MOV     R1,-(SP)
3356  012162  010246                          MOV     R2,-(SP)
3357  012164  010346                          MOV     R3,-(SP)
3358  012166  010446                          MOV     R4,-(SP)
3359  012170  013700  024414                  MOV     BUF2R0,R0
3360  012174  013701  024424                  MOV     BUF2R1,R1
3361  012200  012702  000004                  MOV     #4,R2                   ;R2 IS INDEX OFFSET FOR ADDRESSING
3362  012204  012703  000030                  MOV     #24.,R3                 ;R3 IS INDEX OFFSET FOR DEVICE ADDR.
3363  012210  012737  030507  024430          MOV     #VTBUFF+2055.,START
3364  012216  000177  012242                  JMP     @KSTOR1                 ;RETURN TO CALL
3365
3366                                  ;SUBROUTINE ENTERED TO SET UP BUFFER POINTERS FOR UNIT '3'
3367
3368  012222  012637  024464          SETBF3: MOV     (SP)+,KSTOR1            ;SAVE THE RETURN ADDRESS
3369  012226  010046                          MOV     R0,-(SP)                ;SAVE THE WORKING REG.'S
3370  012230  010146                          MOV     R1,-(SP)
3371  012232  010246                          MOV     R2,-(SP)
3372  012234  010346                          MOV     R3,-(SP)
3373  012236  010446                          MOV     R4,-(SP)
3374  012240  013700  024416                  MOV     BUF3R0,R0
3375  012244  013701  024426                  MOV     BUF3R1,R1
3376  012250  012702  000006                  MOV     #6,R2
3377  012254  012703  000044                  MOV     #36.,R3
3378  012260  012737  032457  024430          MOV     #VTBUFF+3055.,START
3379  012266  000177  012172                  JMP     @KSTOR1                 ;RETURN TO CALL
3380
3381                                  ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '0'
3382
3383  012272  010037  024410          SAVBF0: MOV     R0,BUF0R0
3384  012276  010137  024420                  MOV     R1,BUF0R1
3385  012302  012604                          MOV     (SP)+,R4                ;RESTORE THE WORKING REG.'S
3386  012304  012603                          MOV     (SP)+,R3
3387  012306  012602                          MOV     (SP)+,R2
3388  012310  012601                          MOV     (SP)+,R1
3389  012312  012600                          MOV     (SP)+,R0
3390  012314  000002                          RTI
3391
3392                                  ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '1'
3393
3394  012316  010037  024412          SAVBF1: MOV     R0,BUF1R0
3395  012322  010137  024422                  MOV     R1,BUF1R1
3396  012326  012604                          MOV     (SP)+,R4                ;RESTORE THE WORKING REG.'S
3397  012330  012603                          MOV     (SP)+,R3
3398  012332  012602                          MOV     (SP)+,R2
3399  012334  012601                          MOV     (SP)+,R1
3400  012336  012600                          MOV     (SP)+,R0
3401  012340  000002                          RTI
3402
3403                                  ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '2'
3404
3405  012342  010037  024414          SAVBF2: MOV     R0,BUF2R0
3406  012346  010137  024424                  MOV     R1,BUF2R1
3407  012352  012604                          MOV     (SP)+,R4                ;RESTORE THE WORKING REG.'S
```

```
3408  012354  012603                  MOV     (SP)+,R3
3409  012356  012602                  MOV     (SP)+,R2
3410  012360  012601                  MOV     (SP)+,R1
3411  012362  012600                  MOV     (SP)+,R0
3412  012364  000002                  RTI
3413
3414                          ;SUBROUTINE ENTERED TO SAVE BUFFER POINTERS FOR UNIT '3'
3415
3416  012366  010037  024416  SAVBF3: MOV     R0,BUF3R0
3417  012372  010137  024426          MOV     R1,BUF3R1
3418  012376  012604                  MOV     (SP)+,R4                ;RESTORE THE WORKING REG.'S
3419  012400  012603                  MOV     (SP)+,R3
3420  012402  012602                  MOV     (SP)+,R2
3421  012404  012601                  MOV     (SP)+,R1
3422  012406  012600                  MOV     (SP)+,R0
3423  012410  000002                  RTI
3424
3425                          ;SUBROUTINE ENTERED ON ENCOUNTERING SYSTEM ERRORS WHICH MOVES
3426                          ;THE CHARACTER IN THE PC INTO THE DISPLAY BUFFER.
3427
3428  012412  005262  024340  FLGERR: INC     ERFLG0(R2)              ;SET ERROR FLAG
3429  012416  112720  000362          MOVB    #BLINK,(R0)+            ;SET UP TO BLINK CHAR.
3430  012422  117620  000000          MOVB    @(SP),(R0)+             ;MOVE CHAR. INTO BUFFER
3431  012426  112720  000360          MOVB    #360,(R0)+              ;CLR PRESET FIELD
3432  012432  112710  000031          MOVB    #EOS,(R0)               ;TERMINATE W/ EOS.
3433  012436  062716  000002          ADD     #2,(SP)                 ;SET UP STACK TO EXIT
3434  012442  000207                  RTS     PC
3435
3436                          ;FK INTERRUPT SERVICE ROUTINE
3437
3438  012444  105773  001334  SERVFK: TSTB    @FKOCSR(R3)             ;ASCII FLAG SET?
3439  012450  100421                  BMI     SERV00                  ;YES, SERVICE KEYBOARD
3440  012452  005762  024300          TST     AUTSW0(R2)              ;IS THE AUTO CHAR. SW. SET?
3441  012456  001012                  BNE     AUTOLL                  ;YES, GENERATE A CHARACTER
3442  012460  005762  024254          TST     PATERN(R2)              ;GENERATING WORST CASE PATTERN?
3443  012464  001011                  BNE     JMPWOR                  ;YES, OUTPUT NEXT CHAR.
3444  012466  005762  024244          TST     HOLDSC(R2)              ;INTERVAL TIMER, IS ERROR HOLD SET?
3445  012472  001402                  BEQ     LINK0D                  ;NO, CLR INTERVAL TIME FLAG
3446  012474  004737  013760          JSR     PC,TESTSW               ;YES, TST SW15 TO CONTINUE
3447  012500  000137  013106  LINK0D: JMP     SERV0D                  ;SW15 STILL SET, EXIT
3448  012504  000137  013304  AUTOLL: JMP     AUTOUT
3449  012510  000137  013350  JMPWOR: JMP     WORST1
3450
3451  012514  005773  001334  SERV00: TST     @FKOCSR(R3)             ;ERROR FLAG SET?
3452  012520  100003                  BPL     SERV0A                  ;NO VALID CHAR.
3453  012522  004737  012412          JSR     PC,FLGERR               ;YES, ASCII ERRR FLAG SET
3454  012526  000113                  113                             ;DISPLAY A 'BLINKING K'
3455  012530  117304  001336  SERV0A: MOVB    @FKODAT(R3),R4          ;READ & SAVE CHAR.
3456  012534  122704  000003          CMPB    #3,R4                   ;CHAR='↑C'?
3457  012540  001051                  BNE     SERVAA                  ;NO, CONTINUE
3458  012542  005762  024400          TST     BUSY0(R2)               ;YES, DL'S BUSY?
3459  012546  001403                  BEQ     RELOAD                  ;NO, RE-START
3460  012550  005262  024330          INC     STOP0(R2)               ;YES, SET STOP SW.
3461  012554  000554                  BR      SERV0D                  ;EXIT
3462  012556  013700  024430  RELOAD: MOV     START,R0                ;RESET BUFFER POINTER
3463  012562  112710  000031          MOVB    #EOS,(R0)               ;SET UP FOR NEW INPUT
```

```
3464  012566  005062  024370                CLR     XFERO(R2)
3465  012572  005062  024360                CLR     RESTRO(R2)
3466  012576  005062  024330                CLR     STOPO(R2)
3467  012602  005062  024224                CLR     STCODO(R2)
3468  012606  005062  024400                CLR     BUSYO(R2)
3469  012612  005062  024266                CLR     BLKERR(R2)
3470  012616  005062  024254                CLR     PATERN(R2)
3471  012622  005062  024300                CLR     AUTSWO(R2)
3472  012626  005062  024310                CLR     TRANFO(R2)
3473  012632  005062  024320                CLR     RECTRO(R2)
3474  012636  005062  024340                CLR     ERFLGO(R2)
3475  012642  005062  024244                CLR     HOLDSO(R2)
3476  012646  005062  024234                CLR     CHRCTO(R2)              ;CLR CHAR. CNTR.
3477  012652  005062  024204                CLR     TRNSWO(R2)
3478  012656  005062  024174                CLR     RECSWO(R2)
3479  012662  000511                        BR      SERVOD                  ;EXIT
3480  012664  122704  000024        SERVAA: CMPB    #24,R4                  ;CHAR='↑T' ?
3481  012670  001017                        BNE     SERVOB                  ;NO, CONTINUE
3482  012672  005162  024370                COM     XFERO(R2)               ;YES, COMPLIMENT TRANSFER SW.
3483  012676  001503                        BEQ     SERVOD                  ;STOP AUTO TRANSFER.
3484  012700  005762  024400                TST     BUSYO(R2)               ;CURRENTLY TRANSFERRING DATA?
3485  012704  001100                        BNE     SERVOD                  ;YES, IGNORE SW.
3486  012706  005762  024360        SERVAB: TST     RESTRO(R2)              ;HAS A RESTART ADDR BEEN LOADED?
3487  012712  001403                        BEQ     .+10                    ;NO, TERMINATE INPUT BUFFER
3488  012714  016200  024360                MOV     RESTRO(R2),R0           ;YES, RESET BUFFER POINTER
3489  012720  000423                        BR      SERVBB                  ;SET UP TO START TRANSFER
3490  012722  112720  000014                MOVB    #EOT,(R0)+              ;TERMINATE MESSAGE
3491  012726  000420                        BR      SERVBB
3492                                ..
3493  012730  122704  000001        SERVOB: CMPB    #1,R4                   ;CHAR. = TO '↑A'
3494  012734  001556                        BEQ     AUTOST                  ;YES, AUTOMATICALLY GENERATE CHAR.'S
3495  012736  122704  000005                CMPB    #5,R4                   ;CHAR. = TO '↑E'
3496  012742  001705                        BEQ     RELOAD                  ;YES, RELOAD
3497  012744  122704  000027                CMPB    #27,R4                  ;CHAR. = TO '↑W' ?
3498  012750  001574                        BEQ     WORST                   ;YES, GENERATE WORST CASE PATTERN
3499  012752  005762  024400                TST     BUSYO(R2)               ;DL11 BUSY?
3500  012756  001053                        BNE     SERVOD                  ;YES, IGNORE CHAR.
3501  012760  122704  000033                CMPB    #33,R4                  ;REQUEST TRANSFER?
3502  012764  001024                        BNE     SERVOC                  ;NO
3503  012766  000747                        BR      SERVAB
3504  012770  112710  000031        SERVBB: MOVB    #EOT,(R0)
3505  012774  010062  024360                MOV     R0,RESTRO(R2)           ;SAVE LAST ADDR. FOR AUTO RESTART
3506  013000  012762  000001  024400        MOV     #1,BUSYO(R2)            ;SET BUSY SW.
3507  013006  012762  000001  024204        MOV     #1,TRNSWO(R2)           ;SET TRANSFER, SOFTWARE SW.
3508  013014  013701  024430                MOV     START,R1                ;SET UP BUFFER POINTER TO TRANSMIT DATA.
3509  013020  105741                        TSTB    -(R1)                   ;SET UP BUFFER POINTER
3510  013022  012762  000004  024214        MOV     #4,NULCTO(R2)
3511  013030  105072  001570        SERVBC: CLRB    @XBUFO(R2)              ;TRANSMIT THE START CODE
3512  013034  000424                        BR      SERVOD                  ;EXIT
3513  013036  122704  000177        SERVOC: CMPB    #177,R4                 ;CHAR. = 'RUBOUT'?
3514  013042  001010                        BNE     SERVCC                  ;NO, SAVE IT
3515  013044  005762  024234                TST     CHRCTO(R2)              ;BUFFER EMPTY?
3516  013050  001416                        BEQ     SERVOD                  ;YES, IGNORE IT
3517  013052  005362  024234                DEC     CHRCTO(R2)              ;NO, DECEMENT COUNTER
3518  013056  112740  000031                MOVB    #EOT,-(R0)              ;BACK UP BUFFER POINTER
3519  013062  000411                        BR      SERVOD                  ;EXIT
```

```
                                                    CMPCTO R2 ,#334.        ;BUFFER FULL?
                                                    BNE SER.CX              ;YES, IGNORE CHAR.
                                                    INC CHRCTO,R2           ;INCREMENT CHAR. CNTR.
                                                    MOVB R4,@RO)+           ;NO, SAVE CHAR.
                                                    MOVB #EOS, RO
                                                    BIC #2,@#DCSR,R3.       ;CLR INTERVAL TIMER FLAG
                                                    RTS PC                  ;EXIT
                                    SERVDX:         CLR PATERN(R2)          ;CLEAR WORST CASE PATTERN SW.
                                                    CLR AUTSWO(R2)          ;CLEAR AUTO SW.
                                                    BR SERVDD

                                    ;DL11 TRANSMITTER SERVICE ROUTINE

                                    TRNSMT:         TST INTRNO(R2)          ;LEGAL INTERRUPT?
                                                    BNE .+10                ;YES, SERVICE IT.
                                                    INC INTRNO(R2.          ;THIS IS THE INITIALIZE INTERRUPT.
                                                    RTS PC                  ;EXIT
                                                    TST TRNSWO(R2)          ;TRANSFER SW. SET?
                                                    BNE TRANOA              ;YES, LEGAL INTERRUPT
                                                    JSR PC,FLGERR           ;NO, ILLEGAL TRANSMITTER INTERRUPT
                                                    124                     ;DISPLAY A 'BLINKING T'.
                                                    RTS PC
                                    TRANOA:         MOV @SWR,R3PRT          ;READ THE SWITCHES
                                                    BIC #177774,R3PRT
                                                    CMP #SWDD+SWD1,R3PRT    ;LOOPING DATA?
                                                    BEQ TRANOC+4            ;YES, EXIT
                                                    DEC NULCTO(R2)
                                                    BGT SERVBC              ;XMITTED 4 NULL CHARS?
                                                    BLT TRANOB
                                                    MOVB #377,@XBUFO,R2)
                                                    CLR RECTRO(R2)          ;RECV'D CHARACTER COUNTER
                                                    RTS PC
                                    TRANOB:         CMPB #EOP,(R1)+         ;HAS LAST CHAR. BEEN TRANSMITTED?
                                                    BNE TRANOC              ;NO, TRANSMIT NEXT CHAR.
                                                    CLR TRNSWO(R2)          ;YES, CLR TRANSMIT SW.
                                                    CLR BLKERR(R2)          ;CLEAR BLOCK ERROR SW.
                                                    INC RECSWO(R2)          ;SET UP TO REC. DATA
                                                    MOV START,R1            ;RESET POINTER TO COMPARE DATA
                                                    RTS PC                  ;EXIT
                                    TRANOC:         MOVB (R1),@XBUFO(R2)    ;TRANSMIT NEXT CHAR.
                                                    RTS PC
                                    R3PRT:          .WORD 0

                                    ;ENTERED HERE TO INITALIZE THE AUTOMATIC CHARACTER GENERATOR

                                    AUTOST:         MOV #41,R4
                                                    MOV R4,AUTSWO(R2)
                                                    BR SERVCC

                                    ;ENTERED HERE TO GENERATE AND OUTPUT THE AUTOMATIC CHARACTERS

                                    AUTOUT:         INC AUTSWO(R2)          ;UPDATE THE CHARACTER
                                                    BIC #177600,AUTSWO(R2)
                                                    BIT #140,AUTSWO(R2)     ;IS CHARACTER LEGAL?
                                                    BNE .+10                ;YES
                                                    BIS #40,AUTSWO(R2)      ;NO, RESET IT
```

```
  3576  013334  016204  024300                MOV     AUTSWO(R2),R4
  3577  013340  000651                        BR      SERVCC
  3578
  3579                                  ;ENTERED HERE TO INITIALIZE AND OUTPUT A WORST CASE CHARACTER
  3580                                  ;PATTERN OF '125 & 252'.
  3581
  3582  013342  112762  000252  024254  WORST:  MOVB    #252,PATERN(R2)
  3583  013350  105162  024254          WORST1: COMB    PATERN(R2)              ;COMPLIMENT PATTERN
  3584  013354  116204  024254                  MOVB    PATERN(R2),R4
  3585  013360  000641                        BR      SERVCC
  3586
  3587                                  ;DL11 RECEIVER SERVICE ROUTINE
  3588
  3589  013362  017204  001530          RECVIT: MOV     @RBUF0(R2),R4           ;READ DATA BUFFER
  3590  013366  013705  024430                  MOV     START,R5
  3591  013372  062705  001604                  ADD     #90C.,R5                ;IS BUFFER LIMIT EXCEEDED?
  3592  013376  020500                        CMP     R5,R0                   ;IS BUFFER LIMIT EXCEEDED?
  3593  013400  003021                        BGT     RECVON                  ;NO, PROCESS CHARACTER
  3594  013402  017737  165716  014020        MOV     @SWR,R3PXT              ;IF SW0 AND SW1 ARE SET -
  3595  013410  042737  177774  014020        BIC     #177774,R3PXT           ;DO NOT EXIT- RESET THE BUFFER
  3596  013416  022737  000003  014020        CMP     #SW00+SW01,R3PXT        ;AND CONTINUE.
  3597  013424  001003                        BNE     1$
  3598  013426  013700  024430                MOV     START,R0
  3599  013432  000404                        BR      RECVON
  3600  013434  122704  000014          1$:     CMPB    #EOP,R4                 ;TRANSFER COMPLETE?
  3601  013440  001522                        BEQ     RECVOD                  ;YES, SET UP TO EXIT
  3602  013442  000207                        RTS     PC                      ;NO, IGNORE THE CHARACTER
  3603  013444  035762  024174          RECVON: TST     RECSWO(R2)              ;REC. SW. SET?
  3604  013450  001010                        BNE     RECVOA                  ;YES, LEGAL REC. INTERRUPT
  3605  013452  032777  000001  165644        BIT     #SW00,@SWR              ;IS SW '0' SET?
  3606  013460  001004                        BNE     RECVOA                  ;YES, HOST SENDING DATA
  3607  013462  004737  012412          STERR:  JSR     PC,FLGERR               ;NO, ILLEGAL REC. INTR.
  3608  013466  000122                        122                             ;DISPLAY A 'BLINKING R'.
  3609  013470  000207                        RTS     PC                      ;EXIT
  3610
  3611  013472  005704                  RECVOA: TST     R4                      ;REC. ERROR BIT SET?
  3612  013474  100021                        BPL     RECVOB                  ;NO, VALID CHAR.
  3613  013476  032704  040000                BIT     #40000,R4               ;OVERRUN ERROR?
  3614  013502  001404                        BEQ     .+12                    ;NO
  3615  013504  004737  012412                JSR     PC,FLGERR               ;YES, DATA OVERRUN ERROR
  3616  013510  000117                        117                             ;DISPLAY A 'BLINKING O'
  3617  013512  000412                        BR      RECVOB
  3618  013514  032704  020000                BIT     #20000,R4               ;FRAMING ERROR?
  3619  013520  001404                        BEQ     .+12                    ;NO
  3620  013522  004737  012412                JSR     PC,FLGERR               ;YES, DISPLAY A 'BLINKING F'
  3621  013526  000106                        106
  3622  013530  000403                        BR      RECVOB
  3623  013532  004737  012412                JSR     PC,FLGERR                       ;PARITY ERROR
  3624  013536  000120                        120                             ;DISPLAY A BLINKING 'P'
  3625  013540  017737  165560  014020  RECVOB: MOV     @SWR,R3PXT              ;READ THE SWITCHES
  3626  013546  042737  177774  014020        BIC     #177774,R3PXT
  3627  013554  022737  000003  014020        CMP     #SW00+SW01,R3PXT        ;LOOP DATA BACK REQUESTED
  3628  013562  001524                        BEQ     RECVAC                  ;YES
  3629  013564  105704                        TSTB    R4                      ;= NULL CHAR.?
  3630  013566  001513                        BEQ     RECVDE                  ;YES, IGNORE IT
  3631  013570  005762  024224                TST     STCODO(R2)              ;HAVE WE RECEIVED THE START CODE?
```

```
 3632  013574  001014                        BNE     RECVBB          ;YES, SAVE DATA
 3633  013576  122704  000377                CMPB    #377,R4         ;NO, IS THIS IT?
 3634  013602  001107                        BNE     RECVAB          ;NO, CHECK 'SW0'
 3635  013604  112720  000136                MOVB    #136,(R0)+      ;YES, DISPLAY ST. CODE AS '↑'
 3636  013610  112720  000012                MOVB    #12,(R0)+       ;ST NEW LINE
 3637  013614  112710  000031                MOVB    #EOS,(R0)       ;TERMINATE BUFFER.
 3638  013620  005262  024224                INC     STCOD0(R2)      ;SET SW.
 3639  013624  000474                        BR      RECVOE          ;EXIT
 3640
 3641  013626  122704  000014  RECVBB:      CMPB    #EOP,R4         ;TRANSFER DONE?
 3642  013632  001425                        BEQ     RECVDD          ;YES, SET UP TO EXIT
 3643  013634  005262  024320                INC     RECTR0(R2)      ;KEEP TRACK OF NO. OF RECV'D CHAR.S'S
 3644  013640  120421                        CMPB    R4,(R1)+        ;COMPARE REC. DATA TO TRANS. DATA
 3645  013642  001415                        BEQ     RECVOC          ;DATA OK
 3646  013644  005762  024266  RECVBC:      TST     BLKERR(R2)      ;HAS A BLOCK ERROR BEEN DETECTED?
 3647  013650  001002                        BNE     1$              ;YES, DON'T UPDATE ERROR COUNTER
 3648  013652  005262  024340                INC     ERFLG0(R2)      ;DATA ERROR SET SOFT SWITCH
 3649  013656  005262  024266  1$:          INC     BLKERR(R2)
 3650  013662  112720  000374                MOVB    #374,(R0)+      ;DISPLAY CHAR IN "BOLD-UNDERLINE"
 3651  013666  110420                        MOVB    R4,(R0)+
 3652  013670  112720  000360                MOVB    #360,(R0)+      ;CLR FIELD PRESET
 3653  013674  000401                        BR      .+4
 3654  013676  110420  RECVOC:      MOVB    R4,(R0)+        ;DISPLAY REC. CHAR.
 3655  013700  112710  000031                MOVB    #EOS,(R0)
 3656  013704  000207                        RTS     PC              ;EXIT
 3657  013706  005262  024310  RECVOD:      INC     TRANF0(R2)      ;TRANSFER COUNTER
 3658  013712  005062  024400                CLR     BUSY0(R2)
 3659  013716  005062  024174                CLR     RECSW0(R2)      ;CLR. REC. SOFTWARE SW.
 3660  013722  005062  024224                CLR     STCOD0(R2)      ;CLR ST. CODE SW.
 3661  013726  005762  024370                TST     XFER0(R2)       ;IS AUTO TRANSFER SW. SET?
 3662  013732  001431                        BEQ     RECVOE          ;NO, NORMAL TRANSFER EXIT
 3663  013734  005762  024330                TST     STOP0(R2)       ;YES, STOP SW. SET?
 3664  013740  001402                        BEQ     .+6             ;NO, START NEXT TRANSFER
 3665  013742  000137  012556                JMP     RELOAD          ;YES, RESTART
 3666  013746  016200  024360                MOV     RESTR0(R2),R0   ;NO, RESET BUFFER POINTER
 3667  013752  005762  024340                TST     ERFLG0(R2)      ;ERROR FLAG SET?
 3668  013756  001412                        BEQ     RECVDD          ;NO, START NEXT TRANSFER
 3669  013760  005777  165340  TESTSW:      TST     @SWR            ;YES, IS SW15 SET?
 3670  013764  100411                        BMI     RECVEE          ;YES, INHIBIT FURTHER TRANSFERS
 3671  013766  005762  024244                TST     HOLDS0(R2)      ;HAS SW15 BEEN SET AND NOW RESET?
 3672  013772  001404                        BEQ     RECVDD          ;NO, IT HASN'T BEEN SET
 3673  013774  005062  024340                CLR     ERFLG0(R2)      ;YES, RESET THE ERROR COUNTER TO '0'
 3674  014000  005062  024244                CLR     HOLDS0(R2)
 3675  014004  000137  012770  RECVDD:      JMP     SERVBB          ;SET UP NEXT TRANSFER
 3676  014010  012762  000001  024244  RECVEE:  MOV  #1,HOLDS0(R2)  ;SET TO HOLD XFER
 3677  014016  000207  RECVOE:      RTS     PC
 3678  014020  000000  R3PXT:       .WORD   0
 2679
 3680
 3681  014022  032777  000001  165274  RECVAB:  BIT     #SW00,@SWR      ;IS SW '0' SET?
 3682  014030  001322                        BNE     RECVOC          ;YES, JUST DISPLAY CHARACTER.
 3683  014032  000704                        BR      RECVBC          ;NO, POST AS ILLEGAL CHAR.
 3684
 3685                          ;ENTERED HERE IF BOTH 'SW00 & SW01' ARE SET
 3686
 3687  014034  005262  024320  RECVAC:  INC     RECTR0(R2)          ;UPDATE RECEIVE COUNT
```

```
3698  014040  005262  024310                  INC     TRANFO(R2)              ;UPDATE TRANSMIT COUNT
3699  014044  005262  024204                  INC     TRNSWO(R2)              ;SET THE TRANSMITTER SWITCH
3690  014050  110472  001570                  MOVB    R4,@XBUFO(R2)           ;TRANSMIT CHARACTER BACK
3691  014054  042704  177400                  BIC     #177400,R4              ;SET UP TO DISPLAY OCTAL REPRESENTATION
3692                                                                          ;OF THE CHARACTER
3693  014060  012737  000003  024474          MOV     #3,KSTOR5
3694  014066  012737  000376  015166          MOV     #376,MASK
3695  014074  012703  000006                  MOV     #6,R3
3696  014100  000401                          BR      .+4
3697  014102  006104                  MOVE1:  ROL     R4
3698  014104  006104                          ROL     R4
3699  014106  006104                          ROL     R4
3700  014110  005337  024474                  DEC     KSTOR5
3701  014114  002010                          BGE     MOVE2
3702  014116  110402                          MOVB    R4,R2
3703  014120  143702  015166                  BICB    MASK,R2
3704  014124  152702  000060                  BISB    #60,R2
3705  014130  110220                          MOVB    R2,(R0)+                ;PUT CHARACTER IN DISPLAY BUFFER
3706  014132  112710  000031                  MOVB    #EOS,(R0)               ;TERMINATE BUFFER
3707  014136  012737  000370  015166  MOVE2:  MOV     #370,MASK
3708  014144  005303                          DEC     R3                      ;DONE
3709  014146  001355                          BNE     MOVE1                   ;NO
3710  014150  112720  000040                  MOVB    #40,(R0)+               ;YES, PLACE 'SPACE' CODE IN BUFFER
3711  014154  112710  000031                  MOVB    #EOS,(R0)
3712  014160  000207                          RTS     PC                      ;RETURN
3713                          ;*****************************************************************************
3714                          ;ROUTINE TO LOOP THRU A SINGLE LOGIC SUBTEST.  ENTERED FROM THE 'MONITOR'
3715                          ;VIA SELECTING TEST '?'.  UPON ENTERING THIS SUBROUTINE ANOTHER 'HALT' IS EN-
3716                          ;COUNTERED.  LOAD THE ADDRESS OF THE SUBTEST TO BE EXECUTED IN THE DATA
3717                          ;SWITCHES AND PRESS CONTINUE.
3718                          ;NOTE THAT 'SW11' MUST BE '0' (DOWN) TO RUN THIS TEST.
3719                          ;*****************************************************************************
3720
3721  014162  005737  024160          TESTX:  TST     TTYSWH                  ;TTY AVAILABLE?
3722  014166  001406                          BEQ     TSTA                    ;NO, HALT FOR TEST ADDR.
3723  014170  104012                          PRINT
3724  014172  021641                          MES8                            ;TEXT 'TEST ADDR.? '
3725  014174  104015                          ASEMBL                          ;YES, GET ADDR. FROM TTY.
3726  014176  010337  024464                  MOV     R3,KSTOR1               ;SAVE ADDRESS
3727  014202  000404                          BR      TSTB                    ;EXECUTE TEST
3728  014204  000000                  TSTA:   HALT                            ;GET TEST ADDRESS
3729  014206  017737  165112  024464          MOV     @SWR,KSTOR1             ;SAVE ADDRESS
3730
3731  014214  023727  024464  006174  TSTB:   CMP     KSTOR1,#SETUPF          ;'FK' OR 'VT' LOGIC TEST?
3732  014222  101003                          BHI     .+10                    ;SET UP FOR 'FK'
3733  014224  004737  003026                  JSR     PC,SETUPV               ;SET UP FOR 'VT'
3734  014230  000402                          BR      .+6
3735  014232  004737  006174                  JSR     PC,SETUPF
3736  014236  062737  000002  024464          ADD     #2,KSTOR1               ;ADD '2' TO POINT TO INSTRUCTION AFTER SCOPE
3737  014244  017737  010214  024146          MOV     @KSTOR1,TSTNUM          ;LOAD TEST NO.
3738  014252  062737  000002  024464          ADD     #2,KSTOR1
3739  014260  005037  020400          XLOOP:  CLR     SCOPEF                  ;KEEP COUNT AT ZERO
3740  014264  012737  014260  020402          MOV     #XLOOP,RETURN           ;LOAD SCOPE LOOP RETURN POINTER
3741  014272  000177  010166                  JMP     @KSTOR1                 ;JUMP TO TEST
3742                          ;*****************************************************************************
3743                          ;SUBROUTINE TO ISSUE N SPACES
```

```
3744                                    ;N IS ONE PLUS VALUE CONTAINED IN SPACEX
3745                                    ;SPACEX IS CLEARED WITHIN THE SUBROUTINE, SO THAT A CALL ON
3746                                    ;SPACE WITHOUT LOADING SPACEX ISSUES ONLY ONE SPACE
3747                                    ;*************************************************************************
3748
3749  014276  105777  165016           XSPACE: TSTB    @TPS            ;WAIT FOR TTY READY
3750  014302  100375                           BPL     .-4
3751  014304  012777  000240  165010           MOV     #240,@TPB       ;OUTPUT A SPACE
3752  014312  005337  014326                   DEC     SPACEX          ;DECREMENT COUNT
3753  014316  003367                           BGT     XSPACE          ;LOOP IF NOT DONE
3754  014320  005037  014326                   CLR     SPACEX          ;RESET COUNT TO ZERO
3755  014324  000002                           RTI                     ;RETURN
3756  014326  000000                   SPACEX: 0
3757                                    ;*************************************************************************
3758                                    ;SUBROUTINE TO TEST FOR THE KEYBOARD FLAG BEING SET
3759                                    ;*************************************************************************
3760
3761  014330  005737  024160           TKSFLG: TST     TTYSWH          ;USING THE 'TTY' ?
3762  014334  001404                           BEQ     .+12            ;NO, EXIT
3763  014336  105777  164752                   TSTB    @TKS            ;FLAG SET?
3764  014342  100001                           BPL     .+4             ;NO, EXIT
3765  014344  104013                           TTYIN                   ;YES,INQUIRE
3766  014346  000002                           RTI
3767                                    ;*************************************************************************
3768                                    ;KEYBOARD SERVICE ROUTINE
3769                                    ;*************************************************************************
3770  014350  012704  014600           XTTYIN: MOV     #INBUF,R4       ;SETUP CHARACTER BUFFER
3771  014354  005037  024152                   CLR     CHRCNT          ;CLEAR CHARACTER COUNTER
3772  014360  005037  014602                   CLR     INBUF+2
3773  014364  105777  164724           INPUTA: TSTB    @TKS            ;CHARACTER READY?
3774  014370  100375                           BPL     INPUTA          ;NO, WAIT IT OUT
3775  014372  017701  164720                   MOV     @TKB,R1         ;SAVE CHARACTER
3776  014376  042701  000200                   BIC     #200,R1         ;STRIPE PARITY BIT
3777  014402  120127  000060                   CMPB    R1,#60          ;IS IT A SPECIAL CHARACTER
3778  014406  100420                           BMI     SPCHR1          ;YES, TEST IT
3779  014410  122701  000137                   CMPB    #137,R1
3780  014414  100415                           BMI     SPCHR1
3781  014416  010124                   INPUTB: MOV     R1,(R4)+        ;SAVE CHARACTER
3782  014420  005237  024152                   INC     CHRCNT          ;INCREMENT THE CHARACTER COUNT.
3783  014424  022737  000016  024152           CMP     #14.,CHRCNT
3784  014432  100450                           BMI     SPCHR5          ;TYPE '?' IF TOO MANY CHAR.
3785  014434  105777  164660           OUTPTA: TSTB    @TPS            ;ECHO CHARACTER
3786  014440  100375                           BPL     OUTPTA
3787  014442  110177  164654                   MOVB    R1,@TPB
3788  014446  000746                           BR      INPUTA          ;WAIT FOR NEXT CHARACTER
3789
3790                                    ;SUBROUTINE TO TEST FOR SPECIAL CHARACTERS :'↑C','+','CR','.',' OR 'RUBOUT'
3791  014450  122701  000003           SPCHR1: CMPB    #3,R1           ;CHAR.='↑C'
3792  014454  001002                           BNE     .+6             ;NO,NOT '↑C'
3793  014456  000137  002526                   JMP     MONITR          ;YES, EXIT TO MONITOR
3794  014462  122701  000001                   CMPB    #1,R1           ;CHAR. ='↑A'?
3795  014466  001005                           BNE     .+14            ;NOT '↑A'
3796  014470  022626                           POP2SP                  ;YES, RESTORE STACK
3797  014472  104012                           PRINT
3798  014474  023005                           CNTRLA                  ;TEXT '↑A'
3799  014476  000177  007432                   JMP     @AVECTR
```

```
3800  014502  122701  000177          CMPB    #177,R1         ;CHAR. = 'RUBOUT'
3801  014506  001011                  BNE     SPCHR3          ;IGNORE CHAR. & EXIT
3802  014510  005737  024152          TST     CHRCNT          ;IS RUBOUT LEGAL?
3803  014514  001723                  BEQ     INPUTA          ;NO, IGNORE IT
3804  014516  005337  024152          DEC     CHRCNT
3805  014522  012701  000134          MOV     #134,R1         ;TYPE '\' TO INDICATE RUBOUT
3806  014526  005744                  TST     -(R4)           ;POP OFF LAST CHARACTER
3807  014530  000741                  BR      OUTPTA          ;WAIT FOR NEXT CHARACTER
3808  014532  122701  000054  SPCHR3: CMPB    #54,R1          ;TEST FOR ','
3809  014536  001727                  BEQ     INPUTB          ;LEGAL CHAR., SAVE IT
3810  014540  122701  000015  SPCHR4: CMPB    #15,R1          ;=TO 'CARRIAGE RETURN' TO TERMINATE?
3811  014544  001003                  BNE     SPCHR5          ;NO, CONTINUE
3812  014546  104012                  PRINT                   ;YES, TYPE 'CR-LF'
3813  014550  023011                  CRLF
3814  014552  000002          EXTTY:  RTI                     ;EXIT
3815  014554  122701  000040  SPCHR5: CMPB    #40,R1          ;TEST FOR SPACE
3816  014560  001725                  BEQ     OUTPTA          ;ECHO BUT DON'T SAVE
3817  014562  105701                  TSTB    R1              ;TEST FOR A NULL CHAR.
3818  014564  001002                  BNE     .+6             ;NO, ECHO IT
3819  014566  000137  002526          JMP     MONITR          ;ILLEGAL CHAR. RETURN TO MONITR
3820  014572  104012                  PRINT                   ;OTHERWISE TYPE '?'
3821  014574  023013                  QMARK
3822  014576  000664                  BR XTTYIN               ;WAIT FOR NEW ENTRY
3823  014600  000000          INBUF:  0                       ;CHARACTER STORAGE BUFFER
3824          014632                  .=.+30
3825                          ;*****************************************************************
3826                          ;THIS ROUTINE CHECKS THE 'SWAP SW.' AND IF FOUND SET, SETS UP
3827                          ;THE OPPSITE UNIT TO BE TESTED.
3828                          ;*****************************************************************
3829
3830  014632  005037  024166  TSWAP:  CLR     DONESW
3831  014636  005737  024170          TST     SWAPEM          ;SWAP SW. SET?
3832  014642  001001                  BNE     .+4             ;YES, SWITCH UNITS
3833  014644  000207                  RTS     PC              ;NO, EXIT
3834  014646  005077  164636          CLR     @VTCSR          ;CLR GO BIT
3835  014652  005237  024140          INC     UNITFG          ;SELECT OPPOSITE UNIT
3836  014656  004737  016020          JSR     PC,STUNIT       ;SET IT UP
3837  014662  005737  024140          TST     UNITFG          ;BACK TO '0'
3838  014666  001414                  BEQ     SWPEXT          ;YES, CONTINUE TO NEXT TEST
3839  014670  005726                  POP1SP                  ;CLEAN UP STACK
3840  014672  104012                  PRINT
3841  014674  023735                  MES44                   ;TEXT 'RUNNING UNIT 'X'
3842  014676  112737  000260  014750  MOVB    #260,   MRUNN1
3843  014704  153737  024140  014750  BISB    UNITFG, MRUNN1
3844  014712  104012                  PRINT
3845  014714  014750                  MRUNN1
3846  014716  000002                  RTI                     ;RE-RUN TEST W/ UNIT 'X'
3847  014720  104012          SWPEXT: PRINT
3848  014722  023715                  MES43                   ;TEXT 'RUNNING UNIT '0'
3849  014724  005737  024244          TST     HOLDSO
3850  014730  001406                  BEQ     .+16
3851  014732  032777  010000  164364  BIT     #SW12,@SWR
3852  014740  001002                  BNE     .+6
3853  014742  005726                  POP1SP
3854  014744  000002                  RTI
3855  014746  000207                  RTS     PC
```

```
3856  014750  000     015     012  MRUNN1: .BYTE   0,15,12,100
3857  014753  100
3858
3859                               ;*************************************************************************
3860                               ;MESSAGE PRINT ROUTINE, ENTERED VIA EMT DISPATCH HANDLER.
3861                               ;ROUTINE PICKS UP CONTENTS OF THE 'PC' AND USES THIS AS
3862                               ;THE ADDRESS OF MESSAGE TO BE TYPED.
3863                               ;*************************************************************************
3864
3865  014754  004737  017062          TYPMES: JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
3866  014760  104002                          SAVREG                  ;SAVE REGISTERS ON STACK
3867  014762  017602  000000                  MOV     @(SP),R2        ;GET THE MESSAGE ADDRESS FROM START
3868  014766  062716  000002                  ADD     #2,(SP)         ;SET UP STACK TO EXIT
3869  014772  105777  164322          TYPERA: TSTB    @TPS
3870  014776  100375                          BPL     TYPERA          ;WAIT FOR TTY DONE
3871  015000  122712  000100                  CMPB    #100,(R2)       ;TEST FOR '@'
3872  015004  001003                          BNE     TYPER1          ;BRANCH IF NO EQUAL
3873  015006  104003                          GETREG                  ;RESTORE REGISTERS FROM STACK.
3874  015010  104022                          NULL                    ;TRANSMIT 'NULL' CHARACTER
3875  015012  000002                          RTI                     ;OTHERWISE EXIT
3876  015014  122712  000045          TYPER1: CMPB    #45,(R2)        ;TEST FOR '%'
3877  015020  001403                          BEQ     TYPECL          ;IF = TYPE 'CR-LF'
3878  015022  112277  164274          TYPER2: MOVB    (R2)+,@TPB      ;OUTPUT CHAR.
3879  015026  000761                          BR      TYPERA
3880  015030  012777  000015  164264  TYPECL: MOV     #15,@TPB        ;TYPE 'CR'
3881  015036  105777  164256                  TSTB    @TPS
3882  015042  100375                          BPL     .-4
3883  015044  012777  000012  164250          MOV     #12,@TPB
3884  015052  105722                          TSTB    (R2)+           ;INCREMENT BUFFER
3885  015054  000746                          BR      TYPERA
3886                               ;*************************************************************************
3887                               ;SUBROUTINE TO TYPEOUT A '6' DIGIT OCTAL NO. THE 'PC' CONTAINS
3888                               ;THE ADDRESS OF 'WORD' TO BE TYPED
3889                               ;*************************************************************************
3890
3891  015056  004737  017062          OCTPRT: JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
3892  015062  104002                          SAVREG                  ;SAVE REGISTERS ON STACK
3893  015064  017601  000000                  MOV     @(SP),R1        ;THE ADDRESS OF WORD TO BE TYPED
3894  015070  062716  000002                  ADD     #2,(SP)         ;SET UP STACK TO EXIT '
3895  015074  012737  000006  024472          MOV     #6,KSTOR4
3896  015102  012737  000376  015166          MOV     #376,MASK       ;MASK FOR FIRST BIT
3897  015110  000401                          BR      .+4
3898  015112  006111                  MOVEIT: ROL     (R1)
3899  015114  006111                          ROL     (R1)
3900  015116  006111                          ROL     (R1)
3901  015120  111102                          MOVB    (R1),R2
3902  015122  143702  015166                  BICB    MASK,R2
3903  015126  052702  000260                  BIS     #260,R2
3904  015132  132777  000200  164160          BITB    #200,@TPS
3905  015140  100374                          BPL     .-6
3906  015142  110277  164154                  MOVB    R2,@TPB         ;PRINT CHAR.
3907  015146  012737  000370  015166          MOV     #370,MASK       ;MASK FOR NEXT '5' DIGITS
3908  015154  005337  024472                  DEC     KSTOR4
3909  015160  001354                          BNE     MOVEIT
3910  015162  104003                          GETREG                  ;RESTORE REGISTERS FROM STACK.
3911  015164  000002                          RTI
```

```
                                MASK:   376
                                ;*******************************************************************
                                ;THIS SUBROUTINE IS USED TO CREATE THE 'VT' CHARACTER SET IN NUMERICAL ORDER
                                ;ALL LINES ARE SET UP TO BE TERMINATED WITH EITHER 'VISIBLE END OF LINE'
                                ;OR 'VISIBLE END OF SCREEN'. THIS SUBROUTINE WILL RETURN TO THE SUBROUTINE
                                ;'CALL+2' AFTER CREATING '1' VT CHARACTER CODE AND INSERTING IT INTO
                                ;THE 'VT' DISPLAY BUFFER. IF THIS WAS THE LAST CHARACTER OF THE LINE, THE
                                ;SUBROUTINE RETURNS TO THE 'CALL+4',AND IF IT WAS THE LAST CHARACTER OF THE
                                ;BUFFER IT RETURNS TO 'CALL+6'.
                                ;*******************************************************************
015170  005737  024156        INTCHR: TST     INTSWH                  ;INITIALIZE SUBROUTINE?
015174  001014                        BNE     LDCHAR                  ;NO,LOAD CHAR.
015176  012700  024500                MOV     #VTBUFF,R0              ;SET UP BUFFER ADDR.
015202  012701  000040                MOV     #40,R1                  ;CHAR. CODES START W/40 (SPACE).
015206  012702  177701                MOV     #-63.,R2                ;LOAD '63' CHAR.'S/LINE
015212  012703  177760                MOV     #-16.,R3                ;LOAD '16' LINES OF CHAR.'S
015216  012704  015520                MOV     #TERMTB,R4              ;LINE TERMINATOR TABLE (EOL,EOP)
015222  005237  024156                INC     INTSWH
015226  005202                LDCHAR: INC     R2                      ;INCREMENT CHAR. CNTR.
015230  001013                        BNE     LDCHR1                  ;NOT END OF LINE
015232  012702  177701        LDCHR0: MOV     #-63.,R2                 ;RESET CHAR. CNTR.
015236  112420                        MOVB    (R4)+,(R0)+              ;LOAD LINE TERMINATOR (EOL,EOP,EOS
015240  062716  000002                ADD     #2,(SP)
015244  005203                        INC     R3                      ;INCREMENT LINE NO. CNTR.
015246  001401                        BEQ     .+4                     ;EOS?
015250  000207                        RTS     PC                      ;NO, EOL-RETURN TO 'CALL+4'
015252  062716  000002                ADD     #2.(SP)                 ;YES, RETURN TO 'CALL+6'
015256  000207                        RTS     PC
015260  005737  001622        LDCHR1: TST     FCSET1
015264  001405                        BEQ     .+14
015266  112720  000040                MOVB    #40,(R0)+
015272  005037  001622                CLR     FCSET1
015276  000401                        BR      1$
015300  110120                        MOVB    R1,(R0)+                ;MOVE CHAR. TO BUFFER
015302  005201                1$:       INC     R1                      ;INCREMENT CHAR. VALUE
015304  022701  000335        LDCHEB: CMP     #335,R1                 ;CHAR. VALUE OVERFLO?
015310  001003                        BNE     .+10                     ;NO
015312  012701  000040                MOV     #40,R1                  ;YES, RESET CHAR. VALUE
015316  000745                        BR      LDCHR0                  ;RESET LINE COUNTER
015320  005037  001622                CLR     FCSET1
015324  005737  001620                TST     FCSET                   ;FORIEGN CHARACTER SET
015330  001043                        BNE     2$                      ;YES-IGNOR THIS SECTION
015332  020127  000177                CMP     R1,#177                 ;COUNT ABOVE 177?
015336  003440                        BLE     2$                      ;NO,IGNOR THIS SECTION
015340  120127  000203                CMPB    R1,#203                 ;IF NO FORIEGN CHARACTER SET
015344  001435                        BEQ     2$                      ;THEN THERE ARE SOME CODES
015346  120127  000207                CMPB    R1,#207                 ;THAT WE DON'T WANT TO PRINT
015352  001432                        BEQ     2$                      ;THIS IS THE INTELLIGENTS SECTION
015354  120127  000211                CMPB    R1,#211
015360  001427                        BEQ     2$
015362  120127  000216                CMPB    R1,#216
015366  001424                        BEQ     2$
015370  120127  000210                CMPB    R1,#210
015374  001421                        BEQ     2$
015376  120127  000237                CMPB    R1,#237
```

```
3968  015402  001416                  BEQ     2$
3969  015404  120127  000213          CMPB    R1,#213
3970  015410  001413                  BEQ     2$
3971  015412  120127  000215          CMPB    R1,#215
3972  015416  001410                  BEQ     2$
3973  015420  120127  000217          CMPB    R1,#217
3974  015424  001405                  BEQ     2$
3975  015426  120127  000255          CMPB    R1,#255
3976  015432  001402                  BEQ     2$
3977  015434  005137  001622          COM     FCSET1
3978  015440  120127  000212  2$:       CMPB    R1,#VISEOL          ;CHAR.=VISIBLE END OF LINE?
3979  015444  001407                  BEQ     LDCHR2              ;YES, SKIP IT
3980  015446  120127  000214          CMPB    R1,#VISEOP          ;CHAR.=VISIBLE END OF PARAGRAPH?
3981  015452  001404                  BEQ     LDCHR2              ;YES, SKIP IT
3982  015454  120127  000231          CMPB    R1,#VISEOS          ;CHAR.=VISIBLE END OF SCREEN?
3983  015460  001401                  BEQ     .+4                 ;NO,
3984  015462  000207                  RTS     PC
3985  015464  005201          LDCHR2: INC     R1                  ;YES, SKIP IT
3986  015466  005037  001622          CLR     FCSET1
3987  015472  000137  015304          JMP     LDCHEB              ;RETURN TO 'CALL+2'
3988
3989                          ;*************************************************************************
3990                          ;SUBROUTINE TO SET UP A WAIT LOOP TO WAIT FOR VERTICAL SYNC (17 MSEC.).
3991                          ;*************************************************************************
3992
3993  015476  004737  017062  XWAITS: JSR     PC,TTYENB           ;ENABLE TTY INTERRUPTS
3994  015502  012737  170000  024446          MOV     #-10000,TEMP2
3995  015510  005237  024446                  INC     TEMP2
3996  015514  001375                          BNE     .-4
3997  015516  000002                          RTI
3998
3999                          ;FOLLOWING IS A '16' BYTE BUFFER USED TO TERMINATE EACH LINE OF THE 'VT' BUFFER.
4000
4001  015520      212         TERMTB: .BYTE   VISEOL
4002  015521      214                 .BYTE   VISEOP
4003  015522      212                 .BYTE   VISEOL
4004  015523      214                 .BYTE   VISEOP
4005  015524      212                 .BYTE   VISEOL
4006  015525      214                 .BYTE   VISEOP
4007  015526      212                 .BYTE   VISEOL
4008  015527      214                 .BYTE   VISEOP
4009  015530      212                 .BYTE   VISEOL
4010  015531      214                 .BYTE   VISEOP
4011  015532      212                 .BYTE   VISEOL
4012  015533      214                 .BYTE   VISEOP
4013  015534      212                 .BYTE   VISEOL
4014  015535      214                 .BYTE   VISEOP
4015  015536      212                 .BYTE   VISEOL
4016  015537      231                 .BYTE   VISEOS
4017                                  .EVEN
4018
4019                          ;*************************************************************************
4020                          ;SUBROUTINE TO SET UP AN APPROXIMATE '3' SECOND DELAY. THE ROUTINE ALSO
4021                          ;CONTINUOUSLY MONITORS DATA 'SW14' AND IF FOUND SET CAUSES AN INDEFINITE DELAY.
4022                          ;*************************************************************************
4023
```

```
 4024  015540  012737  175000  024446  XDELAY: MOV     #175000,TEMP2           ;SET UP SHORT DELAY
 4025  015546  000402                          BR      .+6
 4026  015550  005037  024446          XDLAYL: CLR     TEMP2                   ;SET UP LONG DELAY
 4027  015554  004737  017062                  JSR     PC,TTYENB               ;ENABLE TTY INTERRUPTS
 4028  015560  012737  177777  024444          MOV     #-1,TEMP1
 4029  015566  005737  024432          XDLAY1: TST     DISPSW                  ;DISPLAYING MESSAGE?
 4030  015572  001010                          BNE     XDLAY3                  ;YES INHIBIT HOLD
 4031  015574  032777  040000  163522          BIT     #SW14,@SWR              ;TEST DATA SW14
 4032  015602  001374                          BNE     .-6                     ;IF SET, WAIT FOR CLEAR.
 4033  015604  032777  004000  163512          BIT     #SW11,@SWR              ;IS SW. '11' SET?
 4034  015612  001006                          BNE     XDLAY2                  ;YES, INIBIT DELAY
 4035  015614  005237  024446          XDLAY3: INC     TEMP2
 4036  015620  001362                          BNE     XDLAY1
 4037  015622  005237  024444                  INC     TEMP1
 4038  015626  001357                          BNE     XDLAY1
 4039  015630  000002                  XDLAY2: RTI
 4040                                  ;*************************************************************************
 4041                                  ;SUBROUTINE TO LOAD '64' CHAR.'S INTO 'VT' BUFFER.
 4042                                  ;*************************************************************************
 4043
 4044  015632  104002                  XLDLNE: SAVREG                          ;SAVE REG.'S
 4045  015634  012701  177700                  MOV     #-64.,R1                ;LOAD '64' CHAR. BYTES
 4046  015640  013702  024464                  MOV     KSTOR1,R2               ;FROM THIS ADDR. UP.
 4047  015644  112220                  XLDLNA: MOVB    (R2)+,(R0)+             ;'R0' IS POINTER TO A 'VT' BUFFER ADDR.
 4048  015646  005201                          INC     R1                      ;FINISHED?
 4049  015650  001375                          BNE     XLDLNA                  ;NO, LOAD NEXT CHAR.
 4050  015652  104003                          GETREG                          ;YES, RESTORE REG.'S
 4051  015654  000002                          RTI
 4052
 4053                                  ;*************************************************************************
 4054                                  ;SUBROUTINE TO LOAD THE 'VT' BUFFER W/ 'VISIBLE EOS' CHARACTER.
 4055                                  ;*************************************************************************
 4056
 4057  015656  104002                  XEOSBF: SAVREG                          ;SAVE REG.'S
 4058  015660  012700  024500                  MOV     #VTBUFF,R0              ;SET UP BUFFER ADDR. POINTER
 4059  015664  012701  004374                  MOV     #2300.,R1               ;LOAD '515' DATA WORDS
 4060  015670  112720  000031                  MOVB    #EOS,(R0)+
 4061  015674  005301                          DEC     R1
 4062  015676  001374                          BNE     .-6
 4063  015700  012700  024500                  MOV     #VTBUFF,R0              ;RE-SET POINTER
 4064  015704  005077  163574                  CLR     @VTCAR                  ;CLR CURSOR ADDR. REG.
 4065  015710  010077  163572                  MOV     R0,@VTSAR               ;LOAD STARTING ADDR.
 4066  015714  012777  000001  163566          MOV     #1,@VTCSR               ;ST. DISPLAY
 4067  015722  104003                          GETREG
 4068  015724  000002                          RTI
 4069
 4070                                  ;*************************************************************************
 4071                                  ;SUBROUTINE ENTERED AT THE END OF EVERY DISPLAY TEST TO CHECK STATUS
 4072                                  ;OF THE DATA SWITCHES
 4073                                  ;*************************************************************************
 4074
 4075  015726  005237  024166  XTHEND: INC     DONESW
 4076  015732  104020          XENDT:  DELAYL                                  ;ISSUE DELAY
 4077  015734  004737  017062                  JSR     PC,TTYENB               ;ENABLE TTY INTERRUPTS
 4078  015740  005737  024154                  TST     RUNSWH                  ;RUNING ALL TESTS?
 4079  015744  001015                          BNE     RUNEM                   ;YES, DO IT
```

```
4080  015746  032777  010000  163350          BIT     #SW12,@SWR              ;MANUAL ADVANCE SW. SET?
4081  015754  001011                          BNE     RUNEM                   ;YES,
4082  015756  013737  024166  024244          MOV     DONESW,HOLDSO           ;SAVE SW.
4083  015764  004737  014632                  JSR     PC,TSWAP                ;TEST FOR SWAP SW.
4084  015770  005737  024244                  TST     HOLDSO                  ;WAS SW. SET TO LOOP TEST?
4085  015774  001354                          BNE     XTHEND                  ;YES, LOOP IT.
4086  015776  000002                          RTI                             ;NORMAL EXIT
4087  016000  104020                  RUNEM:  DELAYL
4088  016002  004737  014632                  JSR     PC,TSWAP                ;TEST SWAP SW.
4089  016006  062716  000002          RUNEXT: ADD     #2,(SP)                 ;ADVANCE TO NEXT TEST
4090  016012  000002                          RTI
4091
4092                                  ;*************************************************************************
4093                                  ;POWER FAIL HANDLER
4094                                  ;*************************************************************; ***********
4095
4096  016014  010046                  PWRFAL: MOV     R0,-(SP)                ;SAVE REGISTERS ON STACK
4097  016016  010146                          MOV     R1,-(SP)
4098  016020  010246                          MOV     R2,-(SP)
4099  016022  010346                          MOV     R3,-(SP)
4100  016024  010446                          MOV     R4,-(SP)
4101  016026  010546                          MOV     R5,-(SP)
4102  016030  013746  000024                  MOV     24,-(SP)
4103  016034  010637  024444                  MOV     SP,TEMP1                ;SAVE STACK POINTER
4104  016040  012737  016050  000024          MOV     #PWRUP,@#24             ;POWER UP ROUTINE TO LOCATION 24
4105  016046  000000                          HALT                            ;HALT
4106  01 050  012777  000340  163234  PWRUP:  MOV     #340,@PSW               ;LOCK OUT INTERRUPTS
4107  016056  013706  024444                  MOV     TEMP1,SP                ;POWER DOWN ROUTINE TO LOCATION 24
4108  016062  012637  000024                  MOV     (SP)+,@#24
4109  016066  012605                          MOV     (SP)+,R5                ;RESTORE REGISTERS
4110  016070  012604                          MOV     (SP)+,R4
4111  016072  012603                          MOV     (SP)+,R3
4112  016074  012602                          MOV     (SP)+,R2
4113  016076  012601                          MOV     (SP)+,R1
4114  016100  012600                          MOV     (SP)+,R0
4115  016102  000005                          RESET
4116  016104  005001                          CLR     R1                      ;POWER UP DELAY
4117  016106  005201                          INC     R1
4118  016110  001376                          BNE     .-2
4119  016112  104000                          DISPLAY
4120  016114  023043                          MES21
4121  016116  000177  002260                  JMP     @RETURN
4122
4123                                  ;*************************************************************************
4124                                  ;SUBROUTINE TO SAVE 'R1-R5' ON STACK
4125                                  ;*************************************************************************
4126
4127  016122  012637  024454          XSAVRG: MOV     (SP)+,SAVEPC
4128  016126  012637  024456                  MOV     (SP)+,SAVPSW
4129  016132  012637  024460                  MOV     (SP)+,SAV2PC
4130  016136  012637  024462                  MOV     (SP)+,SAV2SW
4131  016142  010146                          MOV     R1,-(SP)
4132  016144  010246                          MOV     R2,-(SP)
4133  016146  010346                          MOV     R3,-(SP)
4134  016150  010446                          MOV     R4,-(SP)
4135  016152  010546                          MOV     R5,-(SP)
```

```
 4136  016154  013746  024462                MOV     SAV2SW,-(SP)
 4137  016160  013746  024460                MOV     SAV2PC,-(SP)
 4138  016164  013746  024456                MOV     SAVPSW,-(SP)
 4139  016170  013746  024454                MOV     SAVEPC,-(SP)
 4140  016174  000002                        RTI
 4141                                ;*************************************************************************
 4142                                ;SUBROUTINE TO RESTORE 'R1-R5' FROM THE STACK
 4143                                ;*************************************************************************
 4144
 4145  016176  012637  024454       XGETRG: MOV     (SP)+,SAVEPC
 4146  016202  012637  024456               MOV     (SP)+,SAVPSW
 4147  016206  012637  024460               MOV     (SP)+,SAV2PC
 4148  016212  012637  024462               MOV     (SP)+,SAV2SW
 4149  016216  012605                        MOV     (SP)+,R5
 4150  016220  012604                        MOV     (SP)+,R4
 4151  016222  012603                        MOV     (SP)+,R3
 4152  016224  012602                        MOV     (SP)+,R2
 4153  016226  012601                        MOV     (SP)+,R1
 4154  016230  013746  024462                MOV     SAV2SW,-(SP)
 4155  016234  013746  024460                MOV     SAV2PC,-(SP)
 4156  016240  013746  024456                MOV     SAVPSW,-(SP)
 4157  016244  013746  024454                MOV     SAVEPC,-(SP)
 4158  016250  000002                        RTI
 4159
 4160                                ;*************************************************************************
 4161                                ;SUBROUTINE TO PRE-LOAD THE 'VT' DATA BUFFER WITH DATA FROM THE 'PC'.
 4162                                ;*************************************************************************
 4163
 4164  016252  005077  163232       XPRELD: CLR     @VTCSR          ;STOP 'VT'
 4165  016256  012701  001000               MOV     #512.,R1        ;SET UP '512' WORDS
 4166  016262  011602                        MOV     (SP),R2         ;GET ADDR. OF DATA FROM STACK
 4167  016264  062716  000002               ADD     #2,(SP)         ;SET UP STACK TO EXIT
 4168  016270  012700  024500               MOV     #VTBUFF,R0      ;SET UP BUFFER ADDR. POINTER
 4169  016274  011220                        MOV     (R2),(R0)+      ;MOVE DATA TO BUFFER.
 4170  016276  005301                        DEC     R1              ;DONE?
 4171  016300  001375                        BNE     .-4             ;NO, MOVE NEXT WORD
 4172  016302  112710  014431               MOVB    #14431,(R0)     ;YES, TERMINATE BUFFER W/ 'EOS'
 4173  016306  012700  024500               MOV     #VTBUFF,R0      ;RE-SET BUFFER ADDR. POINTER
 4174  016312  010077  163170               MOV     R0,@VTSAR       ;LOAD 'STARTING ADDR. REG.
 4175  016316  000002                        RTI                     ;EXIT
 4176
 4177                                ;*************************************************************************
 4178                                ;SUBROUTINE TO SET UP THE 'FK' AND THE 'VT' DEVICE ADDRESSES
 4179                                ;*************************************************************************
 4180
 4181  016320  123737  024140  024142 STUNIT: CMPB    UNITFG,UNITNO           ;TESTED ALL UNITS?
 4182  016326  001414                        BEQ     SETUT0                  ;YES, RE-SELECT UNIT '0'
 4183  016330  022737  000001  024140        CMP     #1,UNITFG               ;RUNNING UNIT #1?
 4184  016336  001417                        BEQ     SETUT1                  ;YES
 4185  016340  022737  000002  024140        CMP     #2,UNITFG               ;RUNNING UNIT #2
 4186  016346  001420                        BEQ     SETUT2
 4187  016350  022737  000003  024140        CMP     #3,UNITFG               ;RUNNING UNIT #3
 4188  016356  001421                        BEQ     SETUT3
 4189  016360  005037  024140       SETUT0: CLR     UNITFG
 4190  016364  012700  001330               MOV     #FKOLDB,R0              ;NO, LOAD UNIT '0'
 4191  016370  012701  001410               MOV     #VTOCAR,R1
```

```
4192  016374  000416                        BR      LDUNIT
4193  016376  012700  001344        SETUT1: MOV     #FK1LDB,R0
4194  016402  012701  001424                MOV     #VT1CAR,R1
4195  016406  000411                        BR      LDUNIT
4196  016410  012700  001360        SETUT2: MOV     #FK2LDB,R0
4197  016414  012701  001440                MOV     #VT2CAR,R1
4198  016420  000404                        BR      LDUNIT
4199  016422  012700  001374        SETUT3: MOV     #FK3LDB,R0
4200  016426  012701  001454                MOV     #VT3CAR,R1
4201  016432  012702  001470        LDUNIT: MOV     #FKLDB,R2
4202  016436  012703  001504                MOV     #VTCAR,R3
4203  016442  012022                SETFKA: MOV     (R0)+,(R2)+
4204  016444  012123                        MOV     (R1)+,(R3)+
4205  016446  022702  001504                CMP     #FKLVL+2,R2
4206  016452  001373                        BNE     SETFKA
4207  016454  000207                        RTS     PC
4208                        ;*****************************************************************
4209                        ;SUBROUTINE ENTERED ON AN ILLEGAL TRAP. THE ROTUINE REPORTS WHERE IT
4210                        ;TRAPPED 'FROM' AND WHERE IT TRAPPED 'TO'. IF NO TTY IS AVAILABLE
4211                        ;A 'HALT' IS EXECUTED. HERE THE USER CAN EXAMINE THE 'FROMPC'
4212                        ;LOCATION AND THE 'TOPC' LOCATION TO TRACK DOWN THE ERROR.
4213                        ;*****************************************************************
4214
4215  016456  000005        ERTRAP: RESET                   ;CLEAR ALL FLAGS
4216  016460  104022                NULL
4217  016462  000240                NOP
4218  016464  011637  024162        MOV     (SP),TOPC       ;SAVE LOCATION WHERE IT TRAPPED 'TO'
4219  016470  022626                POP2SP
4220  016472  011637  024164        MOV     (SP),FROMPC     ;SAVE WHERE IT TRAPPED FROM
4221  016476  162737  000004  024162SUB     #4,TOPC
4222  016504  162737  000002  024164SUB     #2,FROMPC
4223  016512  005737  024160        TST     TTYSWH          ;IS TTY AVAILABLE?
4224  016516  001001                BNE     .+4             ;YES
4225  016520  000000                HALT                    ;NO, HALT
4226  016522  104012                PRINT
4227  016524  022133                MESSA                   ;TEXT 'ILLEGAL TRAP'
4228  016526  104014                PRTOCT
4229  016530  024162                TOPC                    ;TYPE 'PC TRAPPED TOO
4230  016532  104012                PRINT
4231  016534  022155                MES6C                   ;TEXT 'FROM'
4232  016536  104014                PRTOCT
4233  016540  024164                FROMPC                  ;WHERE IT TRAPPED FROM
4234  016542  000137  002526        JMP     MONITR          ;RETURN TO THE MONITOR
4235
4236                        ;*****************************************************************
4237                        ;SUBROUTINE TO PRINT THE DECIMAL VALUE IN R2
4238                        ;*****************************************************************
4239
4240  016546  005077  162540XBINDEC: CLR     @PSW            ;ENABLE INTERRUPTS
4241  016552  042702  160000        BIC     #160000,R2
4242  016556  012737  177774  016652MOV     #-4,R3PBT
4243  016564  012704  016642        MOV     #DECPNT,R4
4244  016570  012701  177777TYPT1:  MOV     #-1,R1
4245  016574  005201        TYPT2:  INC     R1
4246  016576  161402                SUB     (R4),R2
4247  016600  100375                BPL     TYPT2
```

```
016602  062402                          ADD     (R4)+,R2
016604  004737  016620                  JSR     PC,DECOUT
016610  005237  016652                  INC     R3PBT
016614  001365                          BNE     TYPT1
016616  000002                          RTI
016620  005701                  DECOUT: TST     R1
016622  001003                          BNE     DEC1
016624  112720  000060                  MOVB    #60.,(R0)+
016630  000207                          RTS     PC
016632  152701  000060          DEC1:   BISB    #60,R1
016636  110120                          MOVB    R1,(R0)+
016640  000207                          RTS     PC

016642  001750                  DECPNT: 1000.
016644  000144                          100.
016646  000012                          10.
016650  000001                          1.
016652  000000                  R3PBT:  .WORD   0

                                ;*******************************************************************************
                                ;SUBROUTINE TO SET UP 'FK' VECTOR ADDR. & INTR ENABLE
                                ;*******************************************************************************

016654  017677  000000  162616  LDFKVT: MOV     @(SP),@FKINT    ;LOAD INTR SERVICE ADDR
016662  062716  000002                  ADD     #2,(SP)
016666  012777  000200  162606          MOV     #200,@FKLVL
016674  052777  000101  162572          BIS     #101,@FKCSR
016702  005077  162404                  CLR     @PSW            ;ENABLE INTERRUPTS
016706  000207                          RTS     PC
                                ;*******************************************************************************
                                ;SUBROUTINE TO RESET 'FK' VECTOR ADDRESSES TO HALT ON INTERRUPTS
                                ;*******************************************************************************
016710  012777  000340  162374  CLRFKV: MOV     #340,@PSW
016716  012777  000012  162550          MOV     #12,@FKCSR
016724  013777  001502  162546          MOV     FKLVL,@FkINT
016732  005077  162544                  CLR     @FKLVL
016736  000207                          RTS     PC

                                ;*******************************************************************************
                                ;SUBROUTINE TO LOAD 'VT' VECTOR ADDR. & INTR ENABLE
                                ;*******************************************************************************

016740  017677  000000  162546  LDVTVT: MOV     @(SP),@VTINT    ;LOAD INTR SERVICE ADDR.
016746  062716  000002                  ADD     #2,(SP)
016752  012777  000240  162536          MOV     #240,@VTLVL
016760  052777  000101  162522          BIS     #101,@VTCSR     ;SET 'GO' & INTR 'ENABLE'
016766  005077  162320                  CLR     @PSW
016772  000207                          RTS     PC

                                ;*******************************************************************************
                                ;SUBROUTINE TO RESET 'VT' VECTOR ADDRESSES TO HALT ON INTERRUPTS
                                ;*******************************************************************************
017000  012777  000340  162310  CLRVTV: MOV     #340,@PSW
017002  042777  000101  162500          BIC     #101,@VTCSR
017010  013777  001516  162476          MOV     VTLVL,@VTINT
```

```
 4304  017016  005077  162474          CLR     @VTLVL
 4305  017022  000207                  RTS     PC
 4306
 4307
 4308                                  ;*************************************************************************
 4309                                  ;SUBROUTINE TO LOAD THE 'FIELD' CONTROL WITH THE '4' MODES AND SET UP
 4310                                  ;'VT' BUFFER.
 4311                                  ;*************************************************************************
 4312  017024  112720  000362  XSTLNE: MOVB    #BLINK,(R0)+            ;LOAD 'BLINK'
 4313  017030  104010                  LDLINE                          ;LOAD LINE OF 'BLINKING' CHAR'S
 4314  017032  112720  000364          MOVB    #BOLD,(R0)+
 4315  017036  104010                  LDLINE                          ;LOAD LINE OF 'BOLD' CHAR.'S
 4316  017040  112720  000361          MOVB    #BLANK,(R0)+
 4317  017044  104010                  LDLINE                          ;LOAD LINE OF 'BLANK' CHAR.'S
 4318  017046  112720  000370          MOVB    #UNLINE,(R0)+
 4319  017052  104010                  LDLINE                          ;LOAD LINE OF 'UNDERLINED' CHAR.'S
 4320  017054  112720  000360          MOVB    #CLRFLD,(R0)+
 4321  017060  000002                  RTI
 4322                                  ;*************************************************************************
 4323                                  ;SUBROUTINE TO TEST IF A TTY IS AVAILABLE, AND IF SO, ENABLE
 4324                                  ;KEYBOARD INTERRUPTS.
 4325                                  ;*************************************************************************
 4326
 4327  017062  005737  024160  TTYENB: TST     TTYSWH                  ;TTY AVAILABLE?
 4328  017066  001405                  BEQ     .+14                    ;NO, EXIT
 4329  017070  012777  000100  162216  MOV     #100,@TKS               ;YES, ENABLE TTY INTERRUPTS
 4330  017076  005077  162210          CLR     @PSW
 4331  017102  000207                  RTS     PC
 4332
 4333                                  ;*************************************************************************
 4334                                  ;SUBROUTINE TO WAIT FOR AND ASSEMBLE CHARACTERS INPUT
 4335                                  ;FROM THE KEYBOARD INTO OCTAL NUMBERS.
 4336                                  ;*************************************************************************
 4337
 4338  017104  104013          XASEMB: TTYIN                           ;GET CHAR.'S FROM KEYBOARD
 4339  017106  012701  014600          MOV     #INBUF,R1               ;SET UP CHAR. BUFFER POINTER
 4340  017112  004737  017200          JSR     PC,STRIPN               ;STRIPE NO.
 4341  017116  010403          XASEM1: MOV     R4,R3                   ;RETURNS HERE IF ONLY '1' NO.
 4342  017120  022703  000701          CMP     #701,R3
 4343  017124  001407                  BEQ     WHY
 4344  017126  022703  154765          CMP     #154765,R3
 4345  017132  001401                  BEQ     .+4
 4346  017134  000002                  RTI
 4347  017136  104012                  PRINT
 4348  017140  022710                  MES19
 4349  017142  000402                  BR      XASEXT
 4350  017144  104012          WHY:    PRINT
 4351  017146  022661                  MES18
 4352  017150  000137  002526  XASEXT: JMP     MONITR
 4353
 4354  017154  005721          WORD2:  TST     (R1)+                   ;ADVANCE POINTER PAST COMMA
 4355  017156  006204                  ASR     R4
 4356  017160  006204                  ASR     R4
 4357  017162  006204                  ASR     R4
 4358  017164  005337  024152          DEC     CHRCNT                  ;DEC. CHAR. CNTR.
 4359  017170  001752                  BEQ     XASEM1                  ;COMMA LAST CHAR.?
```

```
 4360  017172  010403                        MOV     R4,R3           ;NO, SAVE 1ST NO.
 4361  017174  062716  000002                ADD     #2,(SP)         ;SET UP STACK TO EXIT
 4362  017200  005004                STRIPN: CLR     R4
 4363  017202  005737  024152                TST     CHRCNT          ;WAS ANY DATA INPUTTED?
 4364  017206  001001                        BNE     .+4             ;YES, PROCESS IT
 4365  017210  000207                        RTS     PC              ;NO, RETURN
 4366  017212  022711  000054                CMP     #54,(R1)        ;CHAR. = COMMA?
 4367  017216  001756                        BEQ     WORD2           ;YES, SAVE 1ST NO.
 4368  017220  042711  177770                BIC     #177770,(R1)    ;NO, STRIPE NO. TO OCTAL
 4369  017224  062104                        ADD     (R1)+,R4
 4370  017226  005337  024152                DEC     CHRCNT          ;FINISHED?
 4371  017232  003001                        BGT     .+4             ;NO
 4372  017234  000207                        RTS     PC              ;YES, EXIT
 4373  017236  006304                        ASL     R4
 4374  017240  006304                        ASL     R4
 4375  017242  006304                        ASL     R4
 4376  017244  000756                        BR      STRIPN+2
 4377                                  ;*************************************************************************
 4378                                  ;VT MESSAGE ROUTINE, ENTERED VIA EMT DISPATCH HANDLER.
 4379                                  ;ROUTINE PICKS UP CONTENTS OF THE 'PC' AND USES THIS AS
 4380                                  ;THE ADDRESS OF MESSAGE TO BE DISPLAYED.
 4381                                  ;*************************************************************************
 4382
 4383  017246  004737  017062        VTMES:  JSR     PC,TTYENB       ;ENABLE TTY INTERRUPTS
 4384  017252  005077  162232                CLR     @VTCSR          ;CLR 'GO'
 4385  017256  104005                        EOSBUF                  ;PRE-LOAD BUFFER
 4386  017260  017637  000000  024472        MOV     @(SP),KSTOR4    ;GET THE MESSAGE ADDRESS FROM STACK
 4387  017266  011637  024134                MOV     (SP),AVECTR     ;GET ADDRESS OF CALL
 4388  017272  162737  000002  024134        SUB     #2,AVECTR       ;SAVE AS 'A' RE-START ADDR.
 4389  017300  062716  000002                ADD     #2,(SP)         ;SET UP STACK TO EXIT
 4390  017304  104002                        SAVREG                  ;SAVE REG'S
 4391  017306  013702  024472                MOV     KSTOR4,R2
 4392  017312  122712  000100        VTERA:  CMPB    #100,(R2)       ;TEST FOR '@'
 4393  017316  001007                        BNE     VTER1           ;BRANCH IF NO EQUAL
 4394  017320  104003                        GETREG                  ;RESTORE REG'S
 4395  017322  005237  024432                INC     DISPSW          ;SET DISPLAY SW.
 4396  017326  104020                        DELAYL
 4397  017330  005037  024432                CLR     DISPSW          ;CLR DISPLAY SW.
 4398  017334  000002                        RTI                     ;OTHERWISE EXIT
 4399  017336  122712  000045        VTER1:  CMPB    #45,(R2)        ;TEST FOR '%'
 4400  017342  001402                        BEQ     VTCRLF          ;IF = TYPE 'CR-LF'
 4401  017344  112220                VTER2:  MOVB    (R2)+,(R0)+     ;SAVE CHAR. IN BUFFER
 4402  017346  000761                        BR      VTERA
 4403  017350  112720  000012        VTCRLF: MOVB    #12,(R0)+       ;SAVE 'EOL' IN BUFFER
 4404  017354  105722                        TSTB    (R2)+           ;INCREMENT BUFFER
 4405  017356  000755                        BR      VTERA
 4406
 4407                                  ;*************************************************************************
 4408                                  ;SUBROUTINE TO TRANSMIT A 'NULL' CHAR. TO THE PRINTER.
 4409                                  ;*************************************************************************
 4410
 4411  017360  005037  014326        XNULL:  CLR     SPACEX
 4412  017364  105237  014326        1$:     INCB    SPACEX
 4413  017370  100375                        BPL     1$
 4414  017372  012737  000002  014326        MOV     #2,SPACEX
 4415  017400  004737  017432                JSR     PC,TSTTPS
```

```
 4416  017404  005077  161712          XNULL1: CLR     @TPB            ;TRANSMIT A NULL CHAR.
 4417  017410  004737  017432                  JSR     PC,TSTTPS
 4418  017414  005337  014326                  DEC     SPACEX
 4419  017420  001371                          BNE     XNULL1
 4420  017422  105237  014336          2$:     INCB    SPACEX
 4421  017426  100375                          BPL     2$
 4422  017430  000002                          RTI
 4423
 4424  017432  105777  161662          TSTTPS: TSTB    @TPS
 4425  017436  100375                          BPL     .-4
 4426  017440  000207                          RTS     PC
 4427                                  ;*************************************************************************
 4428                                  ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'FK' LOGIC TEST
 4429                                  ;THIS ROUTINE SETS UP REG'S 'R0-R5' WITH DATA
 4430                                  ;FROM THE 'FK' LOGIC TO BE EXAMINED ON A 'HALT' IF AN 'FK' LOGIC ERROR IS
 4431                                  ;DETECTED AND 'NO' TELEPRINTER IS AVAILABLE.
 4432                                  ;*************************************************************************
 4433
 4434  017442  005737  024160          FKERR:  TST     TTYSWH          ;TTY AVAILABLE?
 4435  017446  001065                          BNE     TYPFKR          ;YES, TYPE FK ERRORS
 4436  017450  052777  000200  162032          BIS     #AUDIO,@VTCSR   ;ISSUE 'AUDIO BEEP' ON ERROR
 4437  017456  005777  161642                  TST     @SWR            ;TEST IF 'SW15' IS SET
 4438  017462  100014                          BPL     CKHALT          ;NO, INHIBIT ERROR HALT.
 4439  017464  013700  024146                  MOV     TSTNUM,R0       ;SAVE 'TEST NO.' IN 'R0'
 4440  017470  013701  020400                  MOV     SCOPEF,R1       ;SAVE PASS COUNT
 4441  017474  005201                          INC     R1
 4442  017476  011602                          MOV     (SP),R2         ;SAVE 'ERROR PC' IN 'R2'
 4443  017500  162702  000002                  SUB     #2,R2
 4444  017504  017703  161764                  MOV     @FKCSR,R3       ;SAVE 'Fk CSR' IN 'R3'
 4445  017510  017704  161762                  MOV     @FKDATA,R4      ;SAVE 'FK DATA' IN 'R4'
 4446  017514  005777  161604          CKHALT: TST     @SWR            ;CHECK IF SW15 IS SET
 4447  017520  100001                          BPL     .+4             ;NO
 4448  017522  000000                          HALT                    ;YES, HALT
 4449  017524  000002                          RTI
 4450
 4451                                  ;*************************************************************************
 4452                                  ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'VT' LOGIC TEST
 4453                                  ;THIS ROUTINE SETS UP REG'S 'R0-R5' WITH DATA FROM THE 'VT' LOGIC TEST
 4454                                  ;TO BE EXAIMNED ON A 'HALT' IF A 'VT' ERROR IS DETECTED AN 'NO' TELEPRINTER
 4455                                  ;IS AVAILABLE.
 4456                                  ;*************************************************************************
 4457
 4458  017526  005737  024160          VTERR:  TST     TTYSWH          ;TTY AVAILABLE?
 4459  017532  001051                          BNE     TYPVTR          ;YES, TYPE 'VT' ERRORS
 4460  017534  005777  161564                  TST     @SWR            ;TEST IF SW15 IS SET
 4461  017540  100367                          BPL     CKHALT+4        ;NO, INHIBIT ERROR HALT
 4462  017542  104002                          SAVREG                  ;SAVE REGISTERS
 4463  017544  010003                          MOV     R0,R3           ;SAVE SHIFT COUNT
 4464  017546  010104                          MOV     R1,R4           ;SAVE EXPECTED DATA
 4465  017550  013700  024146                  MOV     TSTNUM,R0       ;SAVE FAILING TEST NO.
 4466  017554  013701  020400                  MOV     SCOPEF,R1       ;SAVE PASS COUNTER
 4467  017560  005201                          INC     R1
 4468  017562  011602                          MOV     (SP),R2         ;SAVE 'PC'
 4469  017564  162702  000002                  SUB     #2,R2
 4470  017570  005737  024436                  TST     SHFTSW          ;RUNNING SHIFT TEST?
 4471  017574  001005                          BNE     VTERR1          ;YES
```

```
 4472  017576  017703  161706                MOV     @VTCSR,R3       ;NO, SAVE CONTENTS OF 'CSR'
 4473  017602  017704  161704                MOV     @VTMAR,R4       ;SAVE CONTENTS OF THE 'MAR'
 4474  017606  000402                        BR      VTERR2
 4475  017610  017705  161676        VTERR1: MOV     @VTMAR,R5       ;SAVE CONTENTS OF 'MAR'
 4476  017614  000000                VTERR2: HALT                    ;EXAMINE REG.'S
 4477  017616  104003                        GETREG                  ;RESTORE REG.'S
 4478  017620  000002                        RTI                     ;EXIT
 4479
 4480                                ;*************************************************************************
 4481                                ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'FK' LOGIC TEST
 4482                                ;THIS ROUTINE IS USED TO PRINT OUT THE CONTENTS OF THE 'FK' REGISTERS
 4483                                ;WHEN A LOGIC ERROR IS DETECTED
 4484                                ;*************************************************************************
 4485
 4486  017622  012737  022374  020162 TYPFKR: MOV     #MES13,ERRMES   ;SET UP TO PRINT LOGIC ERROR HEADER
 4487  017630  012737  000003  024444         MOV     #3,TEMP1        ;SET UP TO PRINT 'FK' REG'S
 4488  017636  012701  001474                MOV     #FKCSR,R1
 4489  017642  011637  024470                MOV     (SP),KSTOR3     ;PC OF FAILING ROUTINE
 4490  017646  004737  020114                JSR     PC,PRTERR       ;PRINT ERROR DATA
 4491  017652  000137  017514                JMP     CKHALT          ;EXIT
 4492
 4493                                ;*************************************************************************
 4494                                ;ENTERED WITH SYSTEM TRAP 'TRAP' CALL (ERROR) FROM THE 'VT' LOGIC TEST
 4495                                ;THIS ROUTINE IS USED TO PRINT OUT THE CONTENTS OF THE 'VT' REGISTERS
 4496                                ;WHEN A LOGIC ERROR IS DETECTED
 4497                                ;*************************************************************************
 4498
 4499  017656  104002                TYPVTR: SAVREG                  ;SAVE REGISTERS ON STACK
 4500  017660  005737  024436                TST     SHFTSW          ;DOING SHIFT TEST?
 4501  017664  001422                        BEQ     VTNORM          ;NO, REPORT AS NORMAL ERROR
 4502  017666  005737  024452                TST     MESPT1          ;TYPED HEADER?
 4503  017672  001004                        BNE     .+12            ;YES
 4504  017674  005237  024452                INC     MESPT1
 4505  017700  005037  024450                CLR     MESPRT
 4506  017704  012737  022575  020162        MOV     #MES17,ERRMES   ;YES, SET UP SHIFT HEADER
 4507  017712  010037  024476                MOV     R0,KSTOR6       ;SAVE SHIFT COUNT
 4508  017716  010137  024474                MOV     R1,KSTOR5       ;SAVE EXPT'D DATA
 4509  017722  012737  000001  024444        MOV     #1,TEMP1        ;PRINT '1' DATA WORD
 4510  017730  000413                        BR      VTDUMP
 4511  017732  042777  000100  161550 VTNORM: BIC     #100,@VTCSR     ;CLR 'VT' INTR ENABLE
 4512  017740  012737  022532  020162        MOV     #MES16,ERRMES   ;SET UP TO PRINT LOGIC ERROR HEADER
 4513  017746  012737  000002  024444        MOV     #2,TEMP1        ;SET UP TO PRINT 'VT' REG'S
 4514  017754  012701  001510                MOV     #VTCSR,R1
 4515  017760  011637  024470        VTDUMP: MOV     (SP),KSTOR3     ;PC OF FAILING ROUTINE
 4516  017764  004737  020114                JSR     PC,PRTERR       ;PRINT ERROR DATA
 4517  017770  032777  020000  161326        BIT     #SW13,@SWR      ;IS PRINT INHIBIT SW. SET?
 4518  017776  001043                        BNE     EXTVTR          ;YES, EXIT
 4519  020000  005737  024436                TST     SHFTSW          ;DOING SHIFT TEST?
 4520  020004  001414                        BEQ     GETMAR          ;NO, GET AND TYPE 'MAR'
 4521  020006  013737  024476  024470        MOV     KSTOR6,KSTOR3   ;YES, SET UP TO TYPE SHIFT COUNT.
 4522  020014  004737  020232                JSR     PC,XPRTA        ;TYPE SHIFT COUNT
 4523  020020  104012                        PRINT                   ;PRINT IF DATA IS SHIFTED IN OR OUT
 4524  020022  000000        SHFMES: 0
 4525  020024  013737  024474  024470        MOV     KSTOR5,KSTOR3   ;SET UP TO TYPE EXPT'D DATA
 4526  020032  004737  020232                JSR     PC,XPRTA        ;TYPE EXPT'D DATA
 4527  020036  032777  000007  161260 GETMAR: BIT     #7,@SWR         ;ARE THE 'MAR' SELECT BITS SET?
```

```
4528  020044  001412                        BEQ     TYPMAR          ;NO, PRINT MAR AS IS
4529  020046  042777  000017  161434        BIC     #17,@VTCSR      ;YES, RESELECT THE 'MAR'
4530  020054  017702  161244                MOV     @SWR,R2         ;GET 'MAR' ADDR. FROM SW'S
4531  020060  042702  177770                BIC     #177770,R2
4532  020064  006302                        ASL     R2
4533  020066  050277  161416                BIS     R2,@VTCSR       ;SET UP SELECTED 'MAR' REG.
4534  020072  017737  161414  024470  TYPMAR: MOV     @VTMAR,KSTOR3
4535  020100  004737  020232                JSR     PC,XPRTA
4536  020104  104022                        NULL                    ;TRANSMIT 'NULL' CHARACTER
4537  020106  104003                EXTVTR: GETREG                  ;RESTORE REGISTERS FROM STACK.
4538  020110  000137  017514                JMP     CKHALT          ;EXIT
4539
4540                                        ;*************************************************************
4541                                        ;SUBROUTINE TO PRINT OUT ERROR INFORMATION ON 'FK' & 'VT' LOGIC
4542                                        ;ERRORS.
4543                                        ;*************************************************************
4544
4545  020114  005737  020456        PRTERR: TST     XORFLG
4546  020120  001402                        BEQ     .+6
4547  020122  005237  020264                INC     XORHLT
4548  020126  032777  020000  161170        BIT     #20000,@SWR     ;TEST SW-13 FOR INHIBIT PRINT OUT
4549  020134  001044                        BNE     EXTPRT          ;INHIBIT,CHECK FOR HALT
4550  020136  162737  000002  024470        SUB     #2,KSTOR3
4551  020144  042777  000100  161322        BIC     #100,@FKCSR     ;CLR 'FK' INTR. ENABLE
4552  020152  005737  024450                TST     MESPRT          ;HEADER BEEN TYPED?
4553  020156  001004                        BNE     .+12            ;YES,
4554  020160  104012                        PRINT
4555  020162  000000                ERRMES: 0                       ;PRINT LOGIC ERROR HEADER
4556  020164  005237  024450                INC     MESPRT          ;SET PRINT INHIBIT SW.
4557  020170  104012                        PRINT
4558  020172  023011                        CRLF
4559  020174  104014                        PRTOCT
4560  020176  024146                        TSTNUM                  ;PRINT FAILING TEST NO.
4561  020200  104016                        SPACE
4562  020202  013737  020400  024446        MOV     SCOPEF,TEMP2
4563  020210  005237  024446                INC     TEMP2
4564  020214  104014                        PRTOCT
4565  020216  024446                        TEMP2                   ;PRINT THE FAILING PASS NO.
4566  020220  104016                        SPACE
4567  020222  000403                        BR      XPRTA
4568  020224  012102                XPRT:   MOV     (R1)+,R2
4569  020226  011237  024470                MOV     (R2),KSTOR3
4570  020232  104014                XPRTA:  PRTOCT
4571  020234  024470                        KSTOR3
4572  020236  104016                        SPACE
4573  020240  005337  024444                DEC     TEMP1
4574  020244  003367                        BGT     XPRT
4575  020246  005737  020264        EXTPRT: TST     XORHLT
4576  020252  001403                        BEQ     1$
4577  020254  104012                        PRINT
4578  020256  023756                        MXORH
4579  020260  000000                        HALT
4580
4581  020262  000207                1$:     RTS     PC
4582  020264  000000                XORHLT: 000000
4583                                        ;*************************************************************
```

```
                                        ;SCOPE AND/OR ITERATION LOOP FOR EACH LOGIC TEST
                                        ;*********************************************************************
020266  104017                  SCOPEC: TSTTKS                  ;CHECK FOR KEYBOARD FLAG
020270  032777  040000  161026          BIT     #40000,@SWR     ;TEST SW-14 FOR SCOPE
020276  001015                          BNE     SCOPEB          ;YES, SCOPE
020300  005737  020456                  TST     XORFLG          ;SEE IF RUNNING WITH XOR
020304  001037                          BNE     XORT
020306  032777  004000  161010  SCOP1:  BIT     #4000,@SWR      ;NO-TEST SW-11 FOR ITERATION
020314  001013                          BNE     SCOPEG          ;INHIBIT ITERATION
020316  023737  020400  020376          CMP     SCOPEF,ICOUNT   ;COMPARE CURRENT COUNT TO MAX NUMBER
020324  100007                          BPL     SCOPEG          ;EXIT-DONE
020326  005237  020400                  INC     SCOPEF          ;INCREMENT COUNT
020332  022606                  SCOPEB: CMP     (6)+,SP         ;REPOSITION STACK
020334  012677  160752                  MOV     (6)+,@PSW       ;RESTORE PREVIOUS PROCESSOR STATUS
020340  000177  000036                  JMP     @RETURN         ;REPEAT TEST
020344  005037  020400          SCOPEG: CLR     SCOPEF          ;CLEAR COUNT
020350  011601                          MOV     @SP,R1          ;SAVE TEST NO.
020352  011137  024146                  MOV     (R1),TSTNUM
020356  062716  000002                  ADD     #2,(SP)
020362  011637  020402                  MOV     @SP,RETURN      ;SAVE SCOPE RETURN POINTER
020366  052777  000200  161114          BIS     #AUDIO,@VTCSR   ;ISSUE 'AUDIO BEEP' ON END OF TEST
020374  000002                          RTI                     ;RETURN INLINE-NEXT TEST
020376  000200                  ICOUNT: 200                     ;ITERATION COUNT
020400  000000                  SCOPEF: 0                       ;COUNT LOCATION FOR ITERATION LOOP
020402  006246                  RETURN: FKT1
                                        ;*****BEGINNING OF XOR TESTING
020404  013746  000004          XORT:   MOV     @#4,    -(6)    ;SAVE CONTENTS OF LOCATION 4
020410  012737  020430  000004          MOV     #XORERR,@#4     ;SET UP FOR XOR TIMEOUT
020416  005737  177060                  TST     @#177060        ;IF XOR TIMES OUT IT DETECTED AN ERROR
020422  012637  000004                  MOV     (6)+,@#4        ;NO TIMEOUT-RESTORE LOCATION 4
020426  000727                          BR      SCOP1           ;RETURN TO TEST
020430  022626                  XORERR: CMP     (6)+,(6)+       ;CLEAR STACK
020432  012637  000004                  MOV     (6)+,@#4        ;RESTORE LOCATION 4
020436  032777  000001  160660          BIT     #1,     @SWR    ;RING BELL ON ERROR?
020444  001402                          BEQ     1$              ;IF NO, SKIP AHEAD
020446  104012                          PRINT                   ;SEND BELL
020450  023754                          MBELL
020452  000177  177724          1$:     JMP     @RETURN
020456  000000                  XORFLG: 000000
                                        ;*********************************************************************
                                        ;KEYBOARD CHARACTER TABLE
                                        ;*********************************************************************
                                                                ;START, TOP UN-SHIFTED
020460      133                 CHRTAB: .BYTE   '[
020461      061                         .BYTE   '1
020462      062                         .BYTE   '2
020463      063                         .BYTE   '3
020464      064                         .BYTE   '4
020465      065                         .BYTE   '5
020466      066                         .BYTE   '6
020467      067                         .BYTE   '7
020470      070                         .BYTE   '8
020471      071                         .BYTE   '9
```

```
 4640  020472  060                    .BYTE   '0
 4641  020473  055                    .BYTE   '-
 4642  020474  137                    .BYTE   '←
 4643  020475  135                    .BYTE   ']
 4644
 4645                                ;2ND ROW LOWER CASE
 4646
 4647  020476  011                    .BYTE   11              ;TAB
 4648  020477  033                    .BYTE   33              ;ALTMODE
 4649  020500  161                    .BYTE   161             ;Q
 4650  020501  167                    .BYTE   167             ;W
 4651  020502  145                    .BYTE   145             ;E
 4652  020503  162                    .BYTE   162             ;R
 4653  020504  164                    .BYTE   164             ;T
 4654  020505  171                    .BYTE   171             ;Y
 4655  020506  165                    .BYTE   165             ;U
 4656  020507  151                    .BYTE   151             ;I
 4657  020510  157                    .BYTE   157             ;O
 4658  020511  160                    .BYTE   160             ;P
 4659  020512  100                    .BYTE   '@
 4660  020513  012                    .BYTE   12              ;'LF'
 4661  020514  015                    .BYTE   15              ;'CR'
 4662
 4663                                ;3RD ROW LOWER CASE
 4664
 4665  020515  136                    .BYTE   '↑
 4666  020516  141                    .BYTE   141             ;A
 4667  020517  163                    .BYTE   163             ;S
 4668  020520  144                    .BYTE   144             ;D
 4669  020521  146                    .BYTE   146             ;F
 4670  020522  147                    .BYTE   147             ;G
 4671  020523  150                    .BYTE   150             ;H
 4672  020524  152                    .BYTE   152             ;J
 4673  020525  153                    .BYTE   153             ;K
 4674  020526  154                    .BYTE   154             ;L
 4675  020527  073                    .BYTE   ';
 4676  020530  072                    .BYTE   ':
 4677  020531  177                    .BYTE   177             ;RUBOUT
 4678
 4679                                ;4TH ROW LOWER CASE
 4680
 4681  020532  172                    .BYTE   172             ;Z
 4682  020533  170                    .BYTE   170             ;X
 4683  020534  143                    .BYTE   143             ;C
 4684  020535  166                    .BYTE   166             ;V
 4685  020536  142                    .BYTE   142             ;B
 4686  020537  156                    .BYTE   156             ;N
 4687  020540  155                    .BYTE   155             ;M
 4688  020541  054                    .BYTE   ',
 4689  020542  056                    .BYTE   '.
 4690  020543  057                    .BYTE   '/
 4691  020544  134                    .BYTE   '\
 4692  020545  040                    .BYTE   40                      ;TERMINATE W/ SPACE
 4693
 4694                                ;START OF 'SHIFTED' PASS
 4695
```

```
 4696  020546  173                UPCASE: .BYTE   173             ;LEFT BRACE
 4697  020547  041                        .BYTE   '!
 4698  020550  042                        .BYTE   '"
 4699  020551  043                        .BYTE   '#
 4700  020552  044                        .BYTE   '$
 4701  020553  045                        .BYTE   '%
 4702  020554  046                        .BYTE   '&
 4703  020555  047                        .BYTE   ''
 4704  020556  050                        .BYTE   '(
 4705  020557  051                        .BYTE   ')
 4706  020560  060                        .BYTE   '0
 4707  020561  075                        .BYTE   '=
 4708  020562  137                        .BYTE   '←
 4709  020563  175                        .BYTE   175             ;RIGHT BRACE
 4710
 4711                             ;2ND ROW 'SHIFTED'
 4712
 4713  020564  011                        .BYTE   11
 4714  020565  033                        .BYTE   33              ;ALTMODE
 4715  020566  121                        .BYTE   'Q
 4716  020567  127                        .BYTE   'W
 4717  020570  105                        .BYTE   'E
 4718  020571  122                        .BYTE   'R
 4719  020572  124                        .BYTE   'T
 4720  020573  131                        .BYTE   'Y
 4721  020574  125                        .BYTE   'U
 4722  020575  111                        .BYTE   'I
 4723  020576  117                        .BYTE   'O
 4724  020577  120                        .BYTE   'P
 4725  020600  140                        .BYTE   140
 4726  020601  012                        .BYTE   12              ;'LF
 4727  020602  015                        .BYTE   15              ;'CR'
 4728
 4729                             ;3RD ROW 'SHIFTED'
 4730
 4731  020603  176                        .BYTE   176             ;'↑'
 4732  020604  101                        .BYTE   'A
 4733  020605  123                        .BYTE   'S
 4734  020606  104                        .BYTE   'D
 4735  020607  106                        .BYTE   'F
 4736  020610  107                        .BYTE   'G
 4737  020611  110                        .BYTE   'H
 4738  020612  112                        .BYTE   'J
 4739  020613  113                        .BYTE   'K
 4740  020614  114                        .BYTE   'L
 4741  020615  053                        .BYTE   '+
 4742  020616  052                        .BYTE   '*
 4743  020617  177                        .BYTE   177             ;RUBOUT
 4744
 4745                             ;4TH ROW 'SHIFTED'
 4746
 4747  020620  132                        .BYTE   'Z
 4748  020621  130                        .BYTE   'X
 4749  020622  103                        .BYTE   'C
 4750  020623  126                        .BYTE   'V
 4751  020624  102                        .BYTE   'B
```

```
 4752  020625  116                    .BYTE   'N
 4753  020626  115                    .BYTE   'M
 4754  020627  074                    .BYTE   '<
 4755  020630  076                    .BYTE   '>
 4756  020631  077                    .BYTE   '?
 4757  020632  174                    .BYTE   174
 4758  020633  040                    .BYTE   40                      ;TERMINATE W/ SPACE
 4759
 4760                                 ;START OF 'CONTROL' PASS
 4761
 4762  020634  033                    CONTRL: .BYTE   33              ;'↑['
 4763  020635  061                            .BYTE   '1
 4764  020636  062                            .BYTE   '2
 4765  020637  063                            .BYTE   '3
 4766  020640  064                            .BYTE   '4
 4767  020641  065                            .BYTE   '5
 4768  020642  066                            .BYTE   '6
 4769  020643  067                            .BYTE   '7
 4770  020644  070                            .BYTE   '8
 4771  020645  071                            .BYTE   '9
 4772  020646  060                            .BYTE   '0
 4773  020647  015                    CNTRIT: .BYTE   15
 4774  020650  037                            .BYTE   37              ;'↑←'
 4775  020651  035                            .BYTE   35              ;'↑'RIGHT BRACKET
 4776
 4777                                 ;2ND ROW 'CONTROL'
 4778
 4779  020652  011                            .BYTE   11
 4780  020653  033                            .BYTE   33              ;ALTMODE
 4781  020654  021                            .BYTE   21              ;'↑Q'
 4782  020655  027                            .BYTE   27              ;'↑W'
 4783  020656  005                            .BYTE   05              ;'↑E'
 4784  020657  022                            .BYTE   22              ;'↑R'
 4785  020660  024                            .BYTE   24              ;'↑T'
 4786  020661  031                            .BYTE   31              ;'↑Y'
 4787  020662  025                            .BYTE   25              ;'↑U'
 4788  020663  011                            .BYTE   11              ;'↑I'
 4789  020664  017                            .BYTE   17              ;'↑O'
 4790  020665  020                            .BYTE   20              ;'↑P'
 4791  020666  000                            .BYTE   00              ;'↑@'
 4792  020667  012                            .BYTE   12              ;'LF'
 4793  020670  015                            .BYTE   15              ;'CR'
 4794
 4795                                 ;3RD ROW 'CONTROL'
 4796
 4797  020671  036                            .BYTE   36              ;'↑↑'
 4798  020672  001                            .BYTE   01              ;'↑A'
 4799  020673  023                            .BYTE   23              ;'↑S'
 4800  020674  004                            .BYTE   04              ;'↑D'
 4801  020675  006                            .BYTE   06              ;'↑F'
 4802  020676  007                            .BYTE   07              ;'↑G'
 4803  020677  010                            .BYTE   10              ;'↑H'
 4804  020700  012                            .BYTE   12              ;'↑J'
 4805  020701  013                            .BYTE   13              ;'↑K'
 4806  020702  014                            .BYTE   14              ;'↑L'
 4807  020703  073                            .BYTE   73              ;';'
```

```
 4808  020704  072                    .BYTE   72              ;':'
 4809  020705  177                    .BYTE   177             ;'↑'RUBOUT
 4810
 4811                                 ;4TH ROW 'CONTROL'
 4812
 4813  020706  032                    .BYTE   32              ;'↑Z'
 4814  020707  030                    .BYTE   30              ;'↑X'
 4815  020710  003                    .BYTE   03              ;'↑C'
 4816  020711  026                    .BYTE   26              ;'↑V'
 4817  020712  002                    .BYTE   02              ;'↑B'
 4818  020713  016                    .BYTE   16              ;'↑N'
 4819  020714  015                    .BYTE   15              ;'↑M'
 4820  020715  054                    .BYTE   ','
 4821  020716  056                    .BYTE   '.
 4822  020717  057                    .BYTE   '/
 4823  020720  034                    .BYTE   34
 4824  020721  040                    .BYTE   40              ;SPACE,  'DONE'
 4825                                 .EVEN
 4826                                 ;*************************************************************************
 4827                                 ;'VT' SHIFT DATA TABLE USED BY THE 'VT' LOGIC SHIFT TEST
 4828                                 ;THIS BUFFER IS SHIFTED IN 2 PASSES (64 TIMES IN ALL) TO TEST THE
 4829                                 ;SHIFT REGISTER INPUT & OUTPUT LOGIC. AFTER EACH NPR TRANSFER IS MADE
 4830                                 ;THE DATA FROM THIS TABLE IS COMPARED TO THE CONTENTS OF THE MAINTENANCE
 4831                                 ;ADDRESS REGISTER. THE SHIFTS ARE COMMENTED IN OCTAL AND REPORTED IN
 4832                                 ;OCTAL. THE LOW BYTE OF EACH SHIFT IS A CONTROL FIELD CHARACTER AND
 4833                                 ;THE HIGH BYTE IS VALID DATA CHARACTER.
 4834                                 ;*************************************************************************
 4835
 4836  020722  360                    .BYTE   360
 4837  020723  040                    .BYTE   40              ;SHIFT '1' OR '41'
 4838  020724  361                    .BYTE   361
 4839  020725  041                    .BYTE   41              :SHIFT '2' OR '42'
 4840  020726  362                    .BYTE   362
 4841  020727  042                    .BYTE   42              ;SHIFT '3' OR '43'
 4842  020730  364                    .BYTE   364
 4843  020731  043                    .BYTE   43              ;SHIFT '4' OR '44'
 4844  020732  370                    .BYTE   370
 4845  020733  044                    .BYTE   44              ;SHIFT '5' OR '45'
 4846  020734  363                    .BYTE   363
 4847  020735  045                    .BYTE   45              ;SHIFT '6' OR '46'
 4848  020736  365                    .BYTE   365
 4849  020737  046                    .BYTE   46              :SHIFT '7' OR '47'
 4850  020740  367                    .BYTE   367
 4851  020741  047                    .BYTE   47              ;SHIFT '10' OR '50'
 4852  020742  371                    .BYTE   371
 4853  020743  050                    .BYTE   50              ;SHIFT '11' OR '51'
 4854  020744  373                    .BYTE   373
 4855  020745  061                    .BYTE   61              :SHIFT '12' OF '52'
 4856  020746  375                    .BYTE   375
 4857  020747  072                    .BYTE   72              ;SHIFT '13' OR '53'
 4858  020750  377                    .BYTE   377
 4859  020751  107                    .BYTE   107             ;SHIFT '14' OR '54'
 4860  020752  366                    .BYTE   366
 4861  020753  106                    .BYTE   106             ;SHIFT '15' OR '55'
 4862  020754  372                    .BYTE   372
 4863  020755  170                    .BYTE   170             ;SHIFT '16' OR '56'
```

```
 4864  020756  374                        .BYTE   374
 4865  020757  123                        .BYTE   123             ;SHIFT '17' OR '57'
 4866  020760  376                        .BYTE   376
 4867  020761  137                        .BYTE   137             ;SHIFT '20' OR '60'
 4868  020762  360                        .BYTE   360
 4869  020763  177                        .BYTE   177             ;SHIFT '21' OR '61'
 4870  020764  360                        .BYTE   360
 4871  020765  200                        .BYTE   200             ;SHIFT '22' OR '62'
 4872  020766  376                        .BYTE   376
 4873  020767  237                        .BYTE   237             ;SHIFT '23' OR '63'
 4874  020770  374                        .BYTE   374
 4875  020771  277                        .BYTE   277             ;SHIFT '24' OR '64'
 4876  020772  372                        .BYTE   372
 4877  020773  253                        .BYTE   253             ;SHIFT '25' OR '65'
 4878  020774  366                        .BYTE   366
 4879  020775  235                        .BYTE   235             ;SHIFT '26' OR '66'
 4880  020776  360                        .BYTE   360
 4881  020777  040                        .BYTE   40              ;SHIFT '27' OR '67'
 4882  021000  361                        .BYTE   361
 4883  021001  041                        .BYTE   41              ;SHIFT '30' OR '70'
 4884  021002  363                        .BYTE   363
 4885  021003  043                        .BYTE   43              ;SHIFT '31' OR '71'
 4886  021004  362                        .BYTE   362
 4887  021005  045                        .BYTE   45              ;SHIFT '32' OR '72'
 4888  021006  364                        .BYTE   364
 4889  021007  047                        .BYTE   47              :SHIFT '33' OR '73'
 4890  021010  366                        .BYTE   366
 4891  021011  050                        .BYTE   50              ;SHIFT '34' OR '74'
 4892  021012  365                        .BYTE   365
 4893  021013  070                        .BYTE   70              ;SHIFT '35' OR '75'
 4894  021014  367                        .BYTE   367
 4895  021015  071                        .BYTE   71              ;SHIFT '36' OR '76'
 4896  021016  370                        .BYTE   370
 4897  021017  072                        .BYTE   72              ;SHIFT '37' OR '77'
 4898  021020  360               SHFEND:  .BYTE   360
 4899  021021  031                        .BYTE   31              ;SHIFT '40' OR '100'
 4900
 4901                            ;*****************************************************************
 4902                            ;MESSAGES
 4903                            ;*****************************************************************
 4904  021022  000                        .BYTE
 4905  021023  045  053045  031124  TITLE:  .ASCII  '%%VT20 DIAGNOSTIC TEST 4/25/75, MAINDEC-11-DBVTA-C-PB @'
 4906  021030  020060  044504  043501
 4907  021036  047516  052123  041511
 4908  021044  052040  051505  020124
 4909  021052  027464  032462  033457
 4910  021060  026065  046440  044501
 4911  021066  042116  041505  030455
 4912  021074  026461  041104  052126
 4913  021102  026501  026503  041120
 4914  021110  040040
 4915
 4916  021112  043042  020113  053523  MES1:   .ASCII  '"FK SWITCH & LIGHT TEST", READY FOR INPUT%@'
 4917  021120  052111  044103  023040
 4918  021126  046040  043511  052110
 4919  021134  052040  051505  021124
```

```
 4920  021142  020054  042522  042101
 4921  021150  020131  047506  020122
 4922  021156  047111  052520  022524
 4923  021164     100
 4924  021165     045  051101  020105  MES1A:  .ASCII ;%ARE DEFAULT ADDRESSES & VECTORS OK (Y OR N) ? @;
 4925  021172  042504  040506  046125
 4926  021200  020124  042101  051104
 4927  021206  051505  042523  020123
 4928  021214  020046  042526  052103
 4929  021222  051117  020123  045517
 4930  021230  024040  020131  051117
 4931  021236  047040  020051  020077
 4932  021244     100
 4933
 4934  021245     045  043042  020113  MES2:   .ASCII  '%"FK LOGIC TEST"%%@'
 4935  021252  047514  044507  020103
 4936  021260  042524  052123  022442
 4937  021266  040045
 4938  021270  052045  051505  020124  MES3:   .ASCII  ;%TEST NO. (0-22) ....%@;
 4939  021276  047516  020056  030050
 4940  021304  031055  024462  027040
 4941  021312  027056  022456     100
 4942
 4943  021317     042  045506  041440  MES4:   .ASCII  ;"FK CHARACTER TEST",INPUT 'LOWER CASE' CHARACTERS.%@;
 4944  021324  040510  040522  052103
 4945  021332  051105  052040  051505
 4946  021340  021124  044454  050116
 4947  021346  052125  023440  047514
 4948  021354  042527  020122  040503
 4949  021362  042523  020047  044103
 4950  021370  051101  041501  042524
 4951  021376  051522  022456     100
 4952
 4953  021403     045  052524  042502  MES5:   .ASCII  ;%TUBE '0', '1' ,'2','3', OR '4'? @;
 4954  021410  023440  023460  020054
 4955  021416  030447  020047  023454
 4956  021424  023462  023454  023463
 4957  021432  020054  051117  023440
 4958  021440  023464  020077     100
 4959
 4960  021445     045  047506  044522  MFCP:   .ASCII  /%FORIEGN CHARACTER SET ('Y' OR 'N' ?)@/
 4961  021452  043505  020116  044103
 4962  021460  051101  041501  042524
 4963  021466  020122  042523  020124
 4964  021474  023450  023531  047440
 4965  021502  020122  047047  020047
 4966  021510  024477     100
 4967  021513     045  047111  052520  MES6:   .ASCII  ;%INPUT CHAR., TEST 1.@;
 4968  021520  020124  044103  051101
 4969  021526  026056  052040  051505
 4970  021534  020124  027061     100
 4971  021541     045  047111  052520  MES6A:  .ASCII  ;%INPUT CHAR., TEST 2.@;
 4972  021546  020124  044103  051101
 4973  021554  026056  052040  051505
 4974  021562  020124  027062     100
 4975  021567     045  047111  052520  MES6B:  .ASCII  ;%INPUT CHAR., TEST 3.@;
```

```
 4976  021574  020124  044103  051101
 4977  021602  026056  052040  051505
 4978  021610  020124  027063     100
 4979
 4980  021615     045  052042  051505  MES7:   .ASCII  :%"TEST COMPLETE"  @;
 4981  021622  020124  047503  050115
 4982  021630  042514  042524  020042
 4983  021636  020040     100
 4984
 4985
 4986  021641     124  051505  020124  MES8:   .ASCII  ;TEST ADDR.^ @;
 4987  021646  042101  051104  037456
 4988  021654  043040
 4989
 4990  021656  053042  031124  020060  MES9:   .ASCII  :"VT20 SYSTEM EXERCISER TEST". ;
 4991  021664  054523  052123  046505
 4992  021672  042440  042530  041522
 4993  021700  051511  051105  052040
 4994  021706  051505  021124  020056
 4995  021714     012                          .BYTE   12
 4996  021715     012                          .BYTE   12
 4997  021716  047516  042524  020072  .ASCII  ;NOTE: THE HOST DIAGNOSTIC, EITHER FOR THE PDP-8 OR 11 MUST BE;
 4998  021724  044124  020105  047510
 4999  021732  052123  042040  040511
 5000  021740  047107  051517  044524
 5001  021746  026103  042440  052111
 5002  021754  042510  020122  047506
 5003  021762  020122  044124  020105
 5004  021770  042120  026520  020070
 5005  021776  051117  030440  020061
 5006  022004  052515  052123  041040
 5007  022012     105
 5008  022013     012                          .BYTE   12
 5009  022014  047514  042101  042105          .ASCII  ;LOADED INTO THE HOST COMPUTER BEFORE RUNNING THIS TEST.;
 5010  022022  044440  052116  020117
 5011  022030  044124  020105  047510
 5012  022036  052123  041440  046517
 5013  022044  052520  042524  020122
 5014  022052  042502  047506  042522
 5015  022060  051040  047125  044516
 5016  022066  043516  052040  044510
 5017  022074  020123  042524  052123
 5018  022102     056
 5019  022103     031                          .BYTE   31
 5020
 5021  022104  053042  020124  042117  MES10:  .ASCII  ;"VT CDD ADDRESS TEST".@;
 5022  022112  020104  042101  051104
 5023  022120  051505  020123  042524
 5024  022126  052123  027042     100
 5025
 5026
 5027  022133     045  046111  042514  MESSA:  .ASCII  ;%ILLEGAL TRAP TO @;
 5028  022140  040507  020114  051124
 5029  022146  050101  052040  020117
 5030  022154     100
 5031
```

```
 5032  022155     040  051106  046517  MES6C:  .ASCII  ; FROM @;
 5033  022162  040040
 5034  022164  044103  051101  027123  MES9C:  .ASCII  :CHARS.: 0000  RECV'D: 0000  XFERS.: 0000  ;
 5035  022172  020072  030060  030060
 5036  022200  020040  042522  053103
 5037  022206  042047  020072  030060
 5038  022214  030060  020040  043130
 5039  022222  051105  027123  020072
 5040  022230  030060  030060  020040
 5041  022236  051105  051522  035056          .ASCII  :ERRS.: 0000;
 5042  022244  030040  030060     060
 5043  022251     012                          .BYTE   12
 5044  022252     012                          .BYTE   12
 5045  022253     031                          .BYTE   31
 5046
 5047                                          .EVEN
 5048  022254     000                          .BYTE
 5049  022255     124  044510  020123  MES11:  .ASCII  :THIS MESSAGE IS BEING LOADED FROM AN ODD STARTING ;
 5050  022262  042515  051523  043501
 5051  022270  020105  051511  041040
 5052  022276  044505  043516  046040
 5053  022304  040517  042504  020104
 5054  022312  051106  046517  040440
 5055  022320  020116  042117  020104
 5056  022326  052123  051101  044524
 5057  022334  043516     040
 5058  022337     101  042104  042522          .ASCII  :ADDRESS. HOW  DOES IT LOOK??;
 5059  022344  051523  020056  047510
 5060  022352  020127  042040  042517
 5061  022360  020123  052111  046040
 5062  022366  047517  037513     077
 5063  022373     031                          .BYTE   31
 5064  022374  052045  052123  047040  MES13:  .ASCII  ;%TST NO  PASS    MA    CSR    DATA@;
 5065  022402  020117  050040  051501
 5066  022410  020123  020040  046440
 5067  022416  020101  020040  041440
 5068  022424  051123  020040  020040
 5069  022432  040504  040524     '00
 5070  022437     042  045506  051040  MES14:  .ASCII  ;"Fk REPEATIBILITY TEST", READY FOR INPUT.%@;
 5071  022444  050105  040505  044524
 5072  022452  044502  044514  054524
 5073  022460  052040  051505  021124
 5074  022466  020054  042522  042101
 5075  022474  020131  047506  020122
 5076  022502  047111  052520  027124
 5077  022510  040045
 5078  022512  041445  040510  027122  MES15:  .ASCII  ;%CHAR.-CODE(S) @;
 5079  022520  041440  042117  024105
 5080  022526  024523  040040
 5081
 5082  022532  022445  051524  020124  MES16:  .ASCII  ;%%TST NO  PASS    MA    CSR    MAR@;
 5083  022540  047516  020040  040520
 5084  022546  051523  020040  020040
 5085  022554  040515  020040  020040
 5086  022562  051503  020122  020040
 5087  022570  046440  051101     100
```

```
5088
5089  022575     045  052045  052123  MES17:  .ASCII  ;'%TST NO  PASS    MA   SHIFT  IN/OUT  EXPT'D RECV'D@;
5090  022602  047040  020117  050040
5091  022610  051501  020123  020040
5092  022616  046440  020101  020040
5093  022624  044123  043111  020124
5094  022632  044440  027516  052517
5095  022640  020124  042440  050130
5096  022646  023524  020104  042522
5097  022654  053103  042047     100
5098  022661     045  042502  040503  MES18:  .ASCII  ;%BECAUSE I LOVE YOU!!%@;
5099  022666  051525  020105  020111
5100  022674  047514  042526  054440
5101  022702  052517  020441  040045
5102  022710  044445  020106  047516  MES19:  .ASCII  ;%IF NOT WITH THE ONE YOU LOVE,;
5103  022716  020124  044527  044124
5104  022724  052040  042510  047440
5105  022732  042516  054440  052517
5106  022740  046040  053117  026105
5107  022746  046040  053117  020105          .ASCII  ; LOVE THE ONE YOUR WITH.%@;
5108  022754  044124  020105  047117
5109  022762  020105  047531  051125
5110  022770  053440  052111  027110
5111  022776  040045
5112
5113  023000  041536  027045     100  CNTRLC: .ASCII  '↑C%@'
5114  023005     136  022501     100  CNTRLA: .ASCII  '↑A%@'
5115
5116  023011     045     100          CRLF:   .ASCII  '%@'
5117
5118
5119  023013     077  020040     100  QMARK:  .ASCII  '?  @'
5120
5121
5122  023017     045  053042  031124  MES20:  .ASCII  ;%"VT20 LOGIC TEST"%@;
5123  023024  020060  047514  044507
5124  023032  020103  042524  052123
5125  023040  022442     100
5126
5127  023043     045  051045  041505  MES21:  .ASCII  '%%RECOVERED FROM POWER FAILURE @'
5128  023050  053117  051105  042105
5129  023056  043040  047522  020115
5130  023064  047520  042527  020122
5131  023072  040506  046111  051125
5132  023100  020105     100
5133
5134  023103     111  050116  052125  MES22:  .ASCII  ;INPUT THE 'UPPER CASE' CHARACTERS.%@;
5135  023110  052040  042510  023440
5136  023116  050125  042520  020122
5137  023124  040503  042523  020047
5138  023132  044103  051101  041501
5139  023140  042524  051522  022456
5140  023146     100
5141
5142  023147     111  050116  052125  MES23:  .ASCII  ;INPUT THE 'CONTROL' CHARACTERS.%@;
5143  023154  052040  042510  023440
```

```
 5144  023162  047503  052116  047522
 5145  023170  023514  041440  040510
 5146  023176  040522  052103  051105
 5147  023204  027123  040045
 5148  023210  053042  020124  044103  MES24:  .ASCII                          ;"VT CHARACTER DISPLAY TEST"@;
 5149  023216  051101  041501  042524
 5150  023224  020122  044504  050123
 5151  023232  040514  020131  042524
 5152  023240  052123  040042
 5153  023244  053042  020124  044506  MES25:  .ASCII                          ;"VT FIELD CONTROL TEST"@;
 5154  023252  046105  020104  047503
 5155  023260  052116  047522  020114
 5156  023256  042524  052123  040042
 5157  023274  053042  020124  051503  MES26:  .ASCII                          ;"VT CSR PRESET TEST"@;
 5158  023302  020122  051120  051505
 5159  023310  052105  052040  051505
 5160  023316  021124     100
 5161  023321     042  052126  046040  MES27:  .ASCII                          ;"VT LINE FEED TEST"@;
 5162  023326  047111  020105  042506
 5163  023334  042105  052040  051505
 5164  023342  021124     100
 5165  023345     042  052126  042440  MES30:  .ASCII                          ;"VT END OF SCREEN TEST"@;
 5166  023352  042116  047440  020106
 5167  023360  041523  042522  047105
 5168  023366  052040  051505  021124
 5169  023374     100
 5170  023375     042  052126  041040  MES31:  .ASCII                          ;"VT BLANK CONTROL TEST"@;
 5171  023402  040514  045516  041440
 5172  023410  047117  051124  046117
 5173  023416  052040  051505  021124
 5174  023424     100
 5175  023425     042  052126  040440  MES33:  .ASCII                          ;"VT ALIGNMENT TEST"@;
 5176  023432  044514  047107  042515
 5177  023440  052116  052040  051505
 5178  023446  021124     100
 5179  023451     042  052126  043040  MES34:  .ASCII                          ;"VT FOCUS TEST"@;
 5180  023456  041517  051525  052040
 5181  023464  051505  021124     100
 5182  023471     042  052126  053440  MES35:  .ASCII                          ;"VT WORST CASE TEST"@;
 5183  023476  051117  052123  041440
 5184  023504  051501  020105  042524
 5185  023512  052123  040042
 5186  023516  053042  020124  052503  MES36:  .ASCII                          ;"VT CURSOR MOVEMENT TEST"@;
 5187  023524  051522  051117  046440
 5188  023532  053117  046505  047105
 5189  023540  020124  042524  052123
 5190  023546  040042
 5191
 5192  023550  020040  047111  020040  MES37:  .ASCII          ;  IN    @;
 5193  023556  020040     100
 5194
 5195  023561     040  052517  020124  MES38:  .ASCII          ; OUT    @;
 5196  023566  020040  040040
 5197
 5198  023572  041042  046117  020104  MES39:  .ASCII  ;"BOLD PRESET TEST"@;
 5199  023600  051120  051505  052105
```

```
 5200  023606  052040  051505  021124
 5201  023614     100
 5202
 5203  023615     042  046102  047111  MES40:  .ASCII  ;"BLINK PRESET TEST"@;
 5204  023622  020113  051120  051505
 5205  023630  052105  052040  051505
 5206  023636  021124     100
 5207
 5209  023641     042  046102  047101  MES41:  .ASCII  ;"BLANK PRESET TEST"@;
 5209  023646  020113  051120  051505
 5210  023654  052105  052040  051505
 5211  023662  021124     100
 5212
 5213  023665     042  047125  042504  MES42:  .ASCII  ;"UNDERLINE PRESET TEST"@;
 5214  023672  046122  047111  020105
 5215  023700  051120  051505  052105
 5216  023706  052040  051505  021124
 5217  023714     100
 5218  023715     045  052522  047116  MES43:  .ASCII  ;%RUNNING TUBE 0@;
 5219  023722  047111  020107  052524
 5220  023730  042502  030040     100
 5221
 5222  023735     045  052522  047116  MES44:  .ASCII  ;%RUNNING TUBE @;
 5223  023742  047111  020107  052524
 5224  023750  042502  040040
 5225
 5226  023754  040007                  MBELL:  .ASCII  <7>/@/
 5227  023756  050045  047522  051107  MXORH:  .ASCII  /%PROGRAM RUN IS ABORTED@/
 5228  023764  046501  051040  047125
 5229  023772  044440  020123  041101
 5230  024000  051117  042524  040104
 5231  024006  047045  046525  042502  MED1:   .ASCII  /%NUMBER OF TUBES? @/
 5232  024014  020122  043117  052040
 5233  024022  041125  051505  020077
 5234  024030     100
 5235          024032                  .EVEN
 5236  024032  053045  020124     100  MUTP:   .ASCII  /%VT @/
 5237  024037     045  045506     040  MFKP:   .ASCII  /%FK /
 5238  024043     040  042526  052103  MAUFK:  .ASCII  / VECTOR AND ADDRESS FOR TUBE @/
 5239  024050  051117  040440  042116
 5240  024056  040440  042104  042522
 5241  024064  051523  043040  051117
 5242  024072  052040  041125  020105
 5243  024100     100
 5244  024101     045  042040  030514  MDLVA:  .ASCII  /% DL11 VECTOR FOR TUBE @/
 5245  024106  020061  042526  052103
 5246  024114  051117  043040  051117
 5247  024122  052040  041125  020105
 5248  024130     100
 5249          024132          .EVEN
 5250                          ;ADDRESS AND CONSTANTS TABLE
 5251
 5252  024132  000000          PIMP:   0
 5253  024134  001726          AVECTR: INITB                   ;CNTR 'A' VECTOR ADDRESS
 5254  024136  001000          STACK:  1000                    ;'SP' INITIALIZATION ADDRESS
 5255  024140  000000          UNITFG: 0                       ;SOFTWARE SW., SET IF BOTH UNITS AVAIL.
```

```
5256  024142  000000  JNITNO: 0                       ;CONTAINS NO. OF UNITS AVAILABLE
5257  024144  000000  MEMSIZ: 0                       ;CONTAINS HIGHEST 'K' OF MEM. ADDR.
5258  024146  000000  TSTNUM: 0                       ;CONTAINS CURRENT LOGIC TEST NO.
5259  024150  000000  FIELDSW:0                       ;FIELD CONTROL SOFTWARE SW.
5260  024152  000000  CHRCNT: 0                       ;'VT' BUFFER CHARACTER COUNTER
5261  024154  000000  RUNSWH: 0                       ;'RUN ALL' SOFTWARE SW.
5262  024156  000000  INTSWH: 0                       ;SOFTWARE SW.
5263  024160  000000  TTYSWH: 0                       ;SOFTWARE SW., SET IF TTY AVAIBLE
5264  024162  000000  TOPC:   0
5265  024164  000000  FROMPC: 0
5266  024166  000000  DONESW: 0
5267  024170  000000  SWAPEM: 0
5268  024172  000000  MTRSWH: 0                       ;SOFTWARE SW.
5269  024174  000000  RECSW0: 0                       ;SET IF RECEIVING OR EXPECTING DATA
5270  024176  000000  RECSW1: 0
5271  024200  000000  RECSW2: 0
5272  024202  000000  RECSW3: 0
5273  024204  000000  TRNSW0: 0                       ;SET IF TRANSMITTING DATA
5274  024206  000000  TRNSW1: 0
5275  024210  000000  TRNSW2: 0
5276  024212  000000  TRNSW3: 0
5277  024214  000000  NULCT0: 0
5278  024216  000000  NULCT1: 0
5279  024220  000000  NULCT2: 0
5280  024222  000000  NULCT3: 0
5281  024224  000000  STCOD0: 0                       ;SET IF START CODE (377) WAS RECV'D
5282  024226  000000  STCOD1: 0
5283  024230  000000  STCOD2: 0
5284  024232  000000  STCOD3: 0
5285  024234  000000  CHRCT0: 0                       ;CONTAINS NO. OF CHAR.'S ON SCREEN
5286  024236  000000  CHRCT1: 0
5287  024240  000000  CHRCT2: 0
5288  024242  000000  CHRCT3: 0
5289  024244  000000  HOLDS0: 0                       ;SET ON ERROR DETECTION, CLR'D VIA SW15
5290  024246  000000  HOLDS1: 0
5291  024250  000000  HOLDS2: 0
5292  024252  000000  HOLDS3: 0
5293  024254  000000  PATERN: 0
5294          024266          .=.+10
5295  024266  000000  BLKERR: 0
5296          024300          .=.+10
5297  024300  000000  AUTSW0: 0
5298  024302  000000  AUTSW1: 0
5299  024304  000000  AUTSW2: 0
5300  024306  000000  AUTSW3: 0
5301  024310  000000  TRANF0: 0
5302  024312  000000  TRANF1: 0
5303  024314  000000  TRANF2: 0
5304  024316  000000  TRANF3: 0
5305  024320  000000  RECTR0: 0
5306  024322  000000  RECTR1: 0
5307  024324  000000  RECTR2: 0
5308  024326  000000  RECTR3: 0
5309  024330  000000  STOP0:  0                       ;SET TO STOP CONTINOUS TRANSFER
5310  024332  000000  STOP1:  0
5311  024334  000000  STOP2:  0
```

```
 5312  024336  000000          STOP3:  0
 5313  024340  000000          ERFLG0: 0
 5314  024342  000000          ERFLG1: 0
 5315  024344  000000          ERFLG2: 0
 5316  024346  000000          ERFLG3: 0
 5317  024350  000000          INTRN0: 0                ;SET ON DL11 TRANS. INITIALIZATION
 5318  024352  000000          INTRN1: 0
 5319  024354  000000          INTRN2: 0
 5320  024356  000000          INTRN3: 0
 5321  024360  000000          RESTR0: 0                ;CONTAINS 1ST FREE LOCATION AFTER
 5322  024362  000000          RESTR1: 0                ;THE USER DATA
 5323  024364  000000          RESTR2: 0
 5324  024366  000000          RESTR3: 0
 5325  024370  000000          XFER0:  0
 5326  024372  000000          XFER1:  0
 5327  024374  000000          XFER2:  0
 5328  024376  000000          XFER3:  0
 5329  024400  000000          BUSY0:  0                ;SET IF DOING CONTINOUS TRANSFERS
 5330  024402  000000          BUSY1:  0
 5331  024404  000000          BUSY2:  0
 5332  024406  000000          BUSY3:  0
 5333  024410  000000          BUF0R0: 0
 5334  024412  000000          BUF1R0: 0
 5335  024414  000000          BUF2R0: 0
 5336  024416  000000          BUF3R0: 0
 5337  024420  000000          BUF0R1: 0
 5338  024422  000000          BUF1R1: 0
 5339  024424  000000          BUF2R1: 0
 5340  024426  000000          BUF3R1: 0
 5341  024430  000000          START:  0
 5342  024432  000000          DISPSW: 0
 5343  024434  000000          SELECT: 0
 5344  024436  000000          SHFTSW: 0
 5345  024440  000000          SHFPRT: 0
 5346  024442  000000          SAVOLD: 0                ;TEMPORARY STORAGE
 5347  024444  000000          TEMP1:  0                ;TEMPORARY STORAGE
 5348  024446  000000          TEMP2:  0                ;TEMPORARY STORAGE
 5349  024450  000000          MESPRT: 0                ;SOFTWARE PRINT INHIBIT SW.
 5350  024452  000000          MESPT1: 0
 5351  024454  000000          SAVEPC: 0                ;TEMPORARY STORAGE OF 'PC'
 5352  024456  000000          SAVPSW: 0                ;TEMPORARY STORAGE OF 'PSW'
 5353  024460  000000          SAV2PC: 0
 5354  024462  000000          SAV2SW: 0
 5355  024464  000000          KSTOR1: 0                ;PERMANENT STORAGE
 5356  024466  000000          KSTOR2: 0                ;PERMANENT STORAGE
 5357  024470  000000          KSTOR3: 0                ;PERMANENT STORAGE
 5358  024472  000000          KSTOR4: 0                ;PERMANENT STORAGE
 5359  024474  000000          KSTOR5: 0
 5360  024476  000000          KSTOR6: 0
 5361
 5362                          ;HERE STARTS A '1200' WORD 'VT' BUFFER USED FOR ALL
 5363                          ;VT DISPLAY TESTS.  DATA TO BE DISPLAYED IS ASSEMBLED
 5364                          ;IN EACH INDIVIDUAL TEST AND STORED HERE.
 5365
 5366  024500      031         VTBUFF: .BYTE   EOS
 5367  024501      031                 .BYTE   EOS
```

```
 5368
 5369
 5370          000001                          .END
```

```
ASEMBL= 104015          1271#   1562    1567    1570    1582    1599    1616    1629    1672    3011    3725
AUDIO = 000200          1460#   3042    4436    4605
AUTOLL  012504          3441    3448#
AUTOST  013272          3494    3565#
AUTOUT  013304          3448    3571#
AUTSW0  024300          3440    3471*   3528*   3566*   3571*   3572*   3573    3575*   3576    5297#
AUTSW1  024302          5298#
AUTSW2  024304          5299#
AUTSW3  024306          5300#
AVECTR  024134          1658*   2967*   2975*   3022*   3087*   3799    4387*   4388*   5253#
BINDEC= 104024          1278#   3192    3195    3198    3201    3204    3207    3210    3213    3218    3221    3224    3227
                        3232    3235    3238    3241
BLANK = 000361          1458#   2439    2444    4316
B_INK = 000362          1459#   3429    4312
BLKERR  024266          3469*   3555*   3646    3649*   5295#
BOLD  = 000364          1457#   4314
BUF0R0  024410          3130*   3329    3383*   5333#
BUF0R1  024420          3131*   3330    3384*   5337#
BUF1R0  024412          3133*   3344    3394*   5334#
BUF1R1  024422          3134*   3345    3395*   5338#
BUF2R0  024414          3171*   3359    3405*   5335#
BUF2R1  024424          3172*   3360    3406*   5339#
BUF3R0  024416          3154*   3374    3416*   5336#
BUF3R1  024426          3155*   3375    3417*   5340#
BUSY0   024400          3458    3468*   3484    3499    3506*   3658*   5329#
BUSY1   024402          5330#
BUSY2   024404          5331#
BUSY3   024406          5332#
CAR   = 000010          1478#   1840    1853    1866    1879    1890
CHKFK0  007646          2933    2939#
CHKFK1  007704          2943    2945    2947    2948#
CHKFK2  007730          2952    2955#   2990    2993
CHKFK4  010004          2962    2964    2969#
CHRCNT  024152          3771*   3782*   3783    3802    3804*   4358*   4363    4370*   5260#
CHRCT0  024234          3191    3476*   3515    3517*   3520    3522*   5285#
CHRCT1  024236          3203    5286#
CHRCT2  024240          3217    5287#
CHRCT3  024242          3231    5288#
CHREND  020721          2981    4824#
CHRERR  010100          2950    2988#
CHRTAB  020460          2927    4630#
CKHALT  017514          4438    4446#   4461    4491    4538
CLRFKV  016710          1659    2672    2689    2705    2721    2735    2984    4280#
CLRFLD= 000360          1464#   4320
CLRVTV  016774          1660    2149    2171    2195    4301#
CNTRIT  020647          2977    4773#
CNTRLA  023005          3798    5114#
CNTRLC  023000          1671    5113#
CONTRL  020634          2971    4762#
CORSIZ  001702          1500    1504#
CRLF    023011          3813    4558    5116#
CSR   = 000000          1474#
DAR   = 000002          1475#   1974    1994    2013
DECOUT  016620          4249    4253#
DECPNT  016642          4243    4261#
DEC1    016632          4254    4257#
```

```
DELAY = 104004          1262# 2281  2375  2392  2417  2442  2445  2517
DELAYL= 104020          1274# 2352  3102  3103  4076  4087  4396
DISPLA= 104000          1258# 2276  2298  2325  2328  2332  2336  2340  2363  2408  2434  2458  2474
                        2489  2507  2534  2556  2796  2815  2834  2854  2869  2925  2965  2973  3020
                        4119
DISPMS  011610          3124  3128  3152  3169  3246# 3249
DISPSW  024432          4029  4395* 4397* 5342#
DL11A   010726          3135  3139#
DL11B   010750          3139  3142#
DL11C   011076          3156  3159#
DL11D   011230          3173  3176#
DONESW  024166          1664* 3830* 4075* 4082  5266#
EMTOK   001220          1291  1293#
EMTSRV  001200          1244  1287#
EMTTAB  001240          1295  1301#
ENDCUR  006132          2519  2524#
ENDEOS  005654          2419  2423#
ENDTST= 104006          1264# 2204  2285  2344  2395  2423  2448  2524  2856  2906  2986
EOL   = 000012          1461#
EOP   = 000014          1462# 3490  3552  3600  3641
EOS   = 000031          1463# 3248  3432  3463  3504  3518  3524  3637  3655  3706  3711  4060  5366
                        5367
EOSBUF= 104005          1263# 2278  2300  2327  2365  3104  4385
ERFLG0  024340          3200  3428* 3474* 3648* 3667  3673* 5313#
ERFLG1  024342          3212  5314#
ERFLG2  024344          3226  5315#
ERFLG3  024346          3240  5316#
ERRMES  020162          4486* 4506* 4512* 4555#
ERROR = 104400          1257# 1780  1791  1802  1814  1826  1832  1843  1856  1870  1882  1893  1909
                        1922  1934  1949  1965  1977  1997  2016  2040  2043  2057  2073  2089  2103
                        2117  2132  2147  2152  2168  2193  2260  2569  2581  2593  2604  2616  2630
                        2645  2657  2669  2688  2704  2720  2738  2742  2751  2755  2766  2777  2781
                        2807  2826  2845
ERTRAP  016456          1239  4215#
EXITF0  007544          2887  2896  2901#
EXITF1  007564          2885  2905#
EXITSF  005160          2245  2264#
EXREPT  010334          3033  3054#
EXTCHR  010036          2940  2970  2972  2977#
EXTPRT  020246          4549  4575#
EXTTY   014552          3814#
EXTVTR  020106          4518  4537#
FCSET   001620          1469# 1512* 1564* 3953
FCSET1  001622          1470# 3941  3944* 3952* 3977* 3986*
FIELDS  024150          2928* 2944  2951  2953* 2989  2991* 3023* 3043  3045* 3049  3050* 5259#
FKCHAR  007576          1727  2924# 2967  2975  2987
FKCSR   001474          1406# 2548  2882  2903* 2905* 2939  2980* 3032  3054* 4274* 4281* 4444  4488
                        4551*
FKCSR4=%000004          1221# 2548* 2567  2576* 2577  2582* 2589* 2590* 2591  2594* 2600* 2602  2614
                        2617* 2625* 2628  2631* 2638* 2639  2641* 2642* 2643  2651* 2652  2655  2683*
                        2699* 2715* 2748* 2749  2752* 2753  2762* 2764  2767* 2774* 2775  2779  2782*
                        2783* 2795* 2805  2808* 2824  2827* 2833* 2843  2846* 2847*
FKDATA  001476          1407# 2763  2778  2884  2941  3034  4445
FKERR   017442          1246  2551  4434#
FKFUN   007416          1725  2868# 2907
FKINT   001500          1408# 1524* 1535* 1583* 1600* 1687  1688* 1689  1691* 2679* 2695* 2711* 4271*
```

```
                        4282*
FKLDA   001472          1405*   2889    2901*
FKLDB   001470          1404*   2895    2902*   4201
FKLOGI  006236          1723    2541    2556*   2859
FKLVL   001502          1409*   2680*   2696*   2712*   4205    4273*   4282    4283*
FKREPT  010126          1729    3007*   3022
FKRPTA  010144          3008    3013*
FKRPTB  010160          3012    3017*   3019
FKT1    006246          2552    2564*   4609
FKT10   006504          2650*
FKT11   006532          2663*
FKT12   006556          2678*
FKT13   006640          2694*
FKT14   006714          2710*
FKT15   006770          2727*
FKT16   007040          2739    2747*
FKT17   007072          2761*
FKT2    006262          2568    2575*
FKT20   007120          2773*
FKT21   007166          2791*
FKT22   007254          2814*
FKT23   007320          2832*
FKT24   007372          2793    2853*
FKT3    006314          2588*
FKT4    006342          2599*
FKT5    006362          2610*
FKT6    006412          2623*
FKT7    006446          2637*
FKOCSR  001334          1336*   3186*   3438    3451    3525*
FKODAT  001336          1337*   3455
FKOINT  001340          1338*   3118*
FKOLDA  001332          1335*
FKOLDB  001330          1334*   1522    1575    4190
FKOLVL  001342          1339*   3119*
FK1CSR  001350          1345*   3187*
FK1DAT  001352          1346*
FK1INT  001354          1347*   3120*
FK1LDA  001346          1344*
FK1LDB  001344          1343*   4193
FK1LVL  001356          1348*   3121*
FK2CSR  001364          1354*   3177*
FK2DAT  001366          1355*
FK2INT  001370          1356*   3165*
FK2LDA  001362          1353*
FK2LDB  001360          1352*   4196
FK2LVL  001372          1357*   3166*
FK3CSR  001400          1363*   3160*
FK3DAT  001402          1364*
FK3INT  001404          1365*   3148*
FK3LDA  001376          1362*
FK3LDB  001374          1361*   4199
FK3LVL  001406          1366*   3149*
FLGERR  012412          3428*   3453    3539    3607    3615    3620    3623
FREPT0  010234          3027    3032*
FREPT1  010260          3036    3040*
FREPT2  010330          3044    3047    3049    3052*
```

```
FROMPC  024164          4220*  4222*  4233   5265*
GETMAR  020036          4520   4527*
GETREG= 104003          1261*  3873   3910   4050   4067   4394   4477   4537
HOLDS0  024244          1663*  3444   3475*  3671   3674*  3676*  3849   4082*  4084   5289*
HOLDS1  024246          5290*
HOLDS2  024250          5291*
HOLDS3  024252          5292*
ICOUNT  020376          1752*  2084*  2140*  2550*  2786*  4594   4607*
INBUF   014600          3770   3772*  3823*  4339
INITAC  002232          1575*
INITB   001726          1508*  1658   5253
INITBA  002136          1253   1553*  1626
INITC   002150          1509   1556*
INITCB  002436          1519   1572   1574   1625*
INITD   002456          1555   1631*
INITSF  004752          2087   2099   2115   2129   2214*  2239
INITTY  001624          1251   1490*
INPUTA  014364          3773*  3774   3788   3803
INPUTB  014416          3781*  3809
INTCHR  015170          2282   2309   2349   3923*
INTRN0  024350          3533   3535*  5317*
INTRN1  024352          5318*
INTRN2  024354          5319*
INTRN3  024356          5320*
INTSWH  024156          2279*  2302*  2348*  3923   3930*  5262*
JMPWOR  012510          3443   3449*
KSTOR1  024464          2301*  2313*  2931*  2957   2979*  3323*  3334   3338*  3349   3353*  3364   3368*  3379
                        3726*  3729*  3731   3736*  3737   3738*  3741   4046   5355*
KSTOR2  024466          5356*
KSTOR3  024470          2303*  2314*  4489*  4515*  4521*  4525*  4534*  4550*  4569*  4571   5357*
KSTOR4  024472          3895*  3908*  4386*  4391   5358*
KSTOR5  024474          3693*  3700*  4508*  4525   5359*
KSTOR6  024476          4507*  4521   5360*
LDCHAR  015226          3924   3931*
LDCHEB  015304          3948*  3987
LDCHRO  015232          3933*  3951
LDCHR1  015260          3932   3941*
LDCHR2  015464          3979   3981   3985*
LDFKVT  016654          2665   2729   2874   2932   3026   4271*
LDLINE= 104010          1266*  4313   4315   4317   4319
LDUNIT  016432          4192   4195   4198   4201*
LDVTVT  016740          2143   2161   2182   4290*
LINKOD  012500          3445   3447*
LOOP12  003510          1889*  1897
LOOP16  003704          1944*  1951
LOOP24  004102          2011*  2019
MASK    015166          3694*  3703   3707*  3896*  3902   3907*  3912*
MAUFK   024043          1596   5238*
MBELL   023754          4621   5226*
MDLVA   024101          1613   5244*
MED1    024006          1566   5231*
MEMSIZ  024144          1505*  5257*
MESPRT  024450          1750*  2549*  4505*  4552   4556*  5349*
MESPT1  024452          1751*  4502   4504*  5350*
MES1    021112          2870   4916*
MES1A   021165          1569   4924*
```

```
MES10   022104          2535    5021#
MES11   022255          2537    5049#
MES13   022374          4486    5064#
MES14   022437          3021    5070#
MES15   022512          3010    5078#
MES16   022532          4512    5082#
MES17   022575          4506    5089#
MES18   022661          4351    5098#
MES19   022710          4348    5102#
MES2    021245          2557    4934#
MES20   023017          1765    5122#
MES21   023043          4120    5127#
MES22   023103          2966    5134#
MES23   023147          2974    5142#
MES24   023210          2277    5148#
MES25   023244          2299    5153#
MES26   023274          2326    5157#
MES27   023321          2364    5161#
MES3    021270          1641    4938#
MES30   023345          2409    5165#
MES31   023375          2435    5170#
MES33   023425          2459    5175#
MES34   023451          2475    5179#
MES35   023471          2490    5182#
MES36   023516          2508    5186#
MES37   023550          2085    2113    5192#
MES38   023561          2097    2126    5195#
MES39   023572          2329    5198#
MES4    021317          2926    4943#
MES40   023615          2333    5203#
MES41   023641          2337    5208#
MES42   023665          2341    5213#
MES43   023715          3848    5218#
MES44   023735          3841    5222#
MES5    021403          1628    4953#
MES5A   022133          4227    5027#
MES6    021513          2797    4967#
MES6A   021541          2816    4971#
MES6B   021567          2835    4975#
MES6C   022155          4231    5032#
MES7    021615          2203    2855    4980#
MES8    021641          3724    4986#
MES80   022164          3246    5034#
MES9    021656          3088    3089    3092    3094    4990#
MFCP    021445          1561    4960#
MFKP    024037          1579    5237#
MONITR  002526          1639    1649#   1680    3793    3819    4234    4352
MOVBLK  005716          2443#   2447
MOVEIT  015112          3898#   3909
MOVEI   014102          3697#   3709
MOVE2   014136          3701    3707#
MRUNN1  014750          1577*   1581    1586*   1591*   1598    1603*   1610*   1615    1621*   3842*   3843*   3845    3856#
MTRSWH  024172          1556    1661*   5268#
MUTP    024032          1594    5236#
MXORH   023756          4578    5227#
NOP   = 000240          1228#
```

```
NOTTY   001634          1252    1492#
NULCTO  024214          3510*   3546*   5277#
NULCT1  024216          5278#
NULCT2  024220          5279#
NULCT3  024222          5280#
NULL    = 104022        1276#   1650    1766    3874    4216    4536
OCTPRT  015056          1313    3891#
OUTPTA  014434          3785#   3786    3807    3816
PATERN  024254          3442    3470*   3527*   3582*   3583*   3584    5293#
PC      =000007         1217#   1528*   1539*   1585*   1602*   1637*   1657*   1659*   1660*   1692*   1760*   1767*   2087*
                        2088*   2099*   2102*   2115*   2116*   2128*   2131*   2143*   2149*   2161*   2171*   2182*   2195*
                        2220*   2239*   2259*   2265*   2292*   2304*   2309*   2316*   2331*   2335*   2339*   2343*   2349*
                        2353*   2554*   2558*   2665*   2672*   2689*   2705*   2721*   2729*   2735*   2874*   2932*   2984*
                        3025*   3026*   3124*   3128*   3152*   3169*   3250*   3253*   3254*   3259*   3260*   3265*   3266*
                        3271*   3272*   3276*   3277*   3282*   3293*   3288*   3289*   3294*   3295*   3300*   3301*   3305*
                        3306*   3311*   3312*   3317*   3318*   3434*   3446*   3453*   3526*   3536*   3539*   3541*   3551*
                        3558*   3560*   3602*   3607*   3609*   3615*   3620*   3623*   3656*   3677*   3712*   3733*   3735*
                        3833*   3836*   3855*   3865*   3891*   3938*   3940*   3984*   3993*   4027*   4077*   4083*   4088*
                        4207*   4249*   4256*   4259*   4276*   4284*   4295*   4305*   4331*   4340*   4365*   4372*   4383*
                        4415*   4417*   4426*   4490*   4516*   4522*   4526*   4535*   4581*
PIMP    024132          1631*   1653    5252#
POP1SP= 005726          1224#   3839    3853
POP2SP= 022626          1227#   2671    2687    2703    2719    2734    2985    3796    4219
PRELOA= 104011          1267#   2410    2436    2460    2476    2491    3509
PRINT   = 104012        1268#   1558    1560    1565    1569    1578    1590    1593    1595    1597    1612    1614    1627
                        1640    1670    1678    1764    2202    3009    3723    3797    3812    3820    3840    3844    3847
                        4226    4230    4347    4350    4523    4554    4557    4577    4620
PRTERR  020114          4490    4516    4545#
PRTOCT= 104014          1270#   4228    4232    4559    4564    4570
PSW     001312          1325#   1493*   1755*   2163*   2682*   2698*   2714*   3105*   3188*   4106*   4240*   4275*   4280*
                        4294*   4301*   4330*   4598*
PUSH1S= 005746          1225#
PUSH2S= 024646          1226#
PWRFAL  016014          1241    4096#
PWRUP   016050          4104    4106#
QMARK   023013          1679    3821    5119#
RBUF0   001530          1424#   3589
RBUF1   001532          1425#
RBUF2   001534          1426#
RBUF3   001536          1427#
RCSR0   001520          1420#   3137*
RCSR1   001522          1421#   3140*
RCSR2   001524          1422#   3174*
RCSR3   001526          1423#   3157*
RECSW0  024174          3106    3478*   3556*   3603    3659*   5269#
RECSW1  024176          5270#
RECSW2  024200          5271#
RECSW3  024202          5272#
RECTR0  024320          3194    3473*   3550*   3643*   3687*   5305#
RECTR1  024322          3206    5306#
RECTR2  024324          3220    5307#
RECTR3  024326          3234    5308#
RECVAB  014022          3634    3681#
RECVAC  014034          3628    3687#
RECVBB  013626          3632    3641#
RECVBC  013644          3646#   3683
```

```
RECVDD  014004          3668    3672    3675*
RECVEE  014010          3670    3676*
RECVIT  013362          3301    3306    3312    3318    3589*
RECVON  013444          3593    3599    3603*
RECVO   011766          3110    3300*
RECVOA  013472          3604    3606    3611*
RECVOB  013540          3612    3617    3622    3625*
RECVOC  013676          3645    3654*   3682
RECVOD  013706          3601    3642    3657*
RECVOE  014016          3630    3639    3662    3677*
RECV1   012000          3112    3305*
RECV2   012012          3161    3311*
RECV3   012024          3144    3317*
RE_LOAD 012556          3459    3462*   3496    3665
REPORT  005126          2247    2255*
REPTER  010264          3039    3042*
REPTOK  010312          3038    3041    3048*
RESCNT  005472          2367*   2374
RESTR0  024360          3465*   3486    3488    3505*   3666    5321*
RESTR1  024362          5322*
RESTR2  024364          5323*
RESTR3  024366          5324*
RETURN  020402          1754*   2275*   2297*   2324*   2362*   2407*   2433*   2457*   2473*   2488*   2506*   2533*   2553*
                        2794*   2868*   2924*   3019*   3085*   3740*   4121    4599    4604*   4609*   4622
RINT0   001540          1428*   1543    1609    3110*
RINT1   001544          1430*   3112*
RINT2   001550          1432*   3161*
RINT3   001554          1434*   3144*
RLVL0   001542          1429*   3111*
RLVL1   001546          1431*   3113*
RLVL2   001552          1433*   3162*
RLVL3   001556          1435*   3145*
RUNEM   016000          4079    4081    4087*
RUNEXT  016006          4089*
RUNSW   024154          1662*   1742*   4078    5261*
R0      =%000000        1210*   1521*   1529*   1533*   1540*   1542*   1551*   1576*   1587*   1590*   1604*   1607*   1622*
                        2101*   2130*   2164*   2165*   2179*   2187    2188    2218*   2227*   2235    2240*   2242    2255
                        2261    2313    2367*   2369*   2384*   2385    2387*   2388    2389*   2414*   2415*   2420*   2421*
                        2439*   2443*   2444*   2513*   2514    2515*   2520*   2521    2522*   2577*   2578*   2579    2611*
                        2612*   2624*   2626*   2664*   2667*   2681*   2684*   2697*   2700*   2713*   2716*   2799*   2903*
                        2818*   2822*   2837*   2841*   2871*   2891    2892*   2894*   2901    2954*   2960*   2992*   3046*
                        3051*   3053*   3106*   3107*   3108    3122*   3123    3126    3129*   3130    3131    3190*   3193*
                        3196*   3199*   3202*   3205*   3208*   3211*   3216*   3219*   3222*   3225*   3230*   3233*   3236*
                        3239*   3324    3329*   3339    3344*   3354    3359*   3369    3374*   3383    3389*   3394    3400*
                        3405    3411*   3416    3422*   3429*   3430*   3431*   3432*   3462*   3463*   3488*   3490*   3504*
                        3505    3518*   3523*   3524*   3592    3598*   3635*   3636*   3637*   3650*   3651*   3652*   3654*
                        3655*   3666*   3705*   3706*   3710*   3711*   3925*   3934*   3943*   3946*   4047*   4058*   4060*
                        4063*   4065    4096    4114*   4168*   4169*   4172*   4173*   4174    4190*   4193*   4196*   4199*
                        4203    4255*   4258*   4312*   4314*   4316*   4318*   4320*   4401*   4403*   4439*   4463    4465*
                        4507
R1      =%000001        1211*   1501*   1502    1504*   1505    1681    1682*   1685*   1690*   1777*   1778*   1789*   1799*
                        1798*   1799*   1800    1828*   1829*   1830    2178*   2190*   2232*   2233*   2234*   2246*   2248
                        2250*   2252*   2253    2366*   2372*   2373    2413*   2418*   2440*   2446*   2512*   2519*   2729*
                        2731*   2736    2740    2800*   2801*   2919*   2820*   2938*   2839*   2872*   2997    2898*   2900*
                        2902    2927*   2948    2949    2963    2971    2977    2981    3034*   3037    3040    3053    3246*
                        3247    3248    3325    3330*   3340    3345*   3355    3360*   3370    3375*   3394    3389*   3395
```

```
                        3399* 3406  3410* 3417  3421* 3508* 3509  3552  3557* 3559  3644  3775* 3776*
                        3777  3779  3781  3787  3791  3794  3800  3805* 3803  3810  3815  3817  3893*
                        3898* 3899* 3900* 3901  3926* 3946  3947* 3948  3950* 3955  3957  3959  3961
                        3963  3965  3967  3969  3971  3973  3975  3978  3980  3982  3985* 4045* 4048*
                        4059* 4061* 4097  4113* 4116* 4117* 4131  4153* 4165* 4170* 4191* 4194* 4197*
                        4200* 4204  4244* 4245* 4253  4257* 4258  4339* 4354  4356  4368* 4369  4440*
                        4441* 4464  4466* 4467* 4488* 4508  4514* 4568  4601* 4602
R2      =%000002        1212# 1543* 1608* 2100  2129  2184* 2185* 2217* 2232  2873* 2881* 2886  2888*
                        2941* 2949  2955  2959* 2960  3123* 3125* 3127  3132* 3133  3134  3150* 3151
                        3153* 3154  3155  3167* 3168  3170* 3171  3172  3191* 3194* 3197* 3200* 3203*
                        3206* 3209* 3212* 3217* 3220* 3223* 3226* 3231* 3234* 3237* 3240* 3247* 3326
                        3331* 3341  3346* 3356  3361* 3371  3376* 3387* 3398* 3409* 3420* 3428* 3440
                        3442  3444  3458  3460* 3464* 3465* 3466* 3467* 3468* 3469* 3470* 3471* 3472*
                        3473* 3474* 3475* 3476* 3477* 3478* 3482* 3484  3486  3488  3499  3505* 3506*
                        3507* 3510* 3511* 3515  3517* 3520  3522* 3527* 3528* 3533  3535* 3537  3546*
                        3549* 3550* 3554* 3555* 3556* 3559* 3566* 3571* 3572* 3573  3575* 3576  3582*
                        3583* 3594  3589  3603  3631  3638* 3643* 3646  3648* 3649* 3657* 3658* 3659*
                        3660* 3661  3663  3666  3667  3671  3673* 3674* 3676* 3687* 3688* 3689* 3690*
                        3702* 3703* 3704* 3705  3867* 3871  3876  3878  3884  3901* 3902* 3903* 3906
                        3927* 3931* 3933* 4046* 4047  4098  4112* 4132  4152* 4166* 4169  4201* 4203*
                        4205  4241* 4246* 4248* 4391* 4392  4399  4401  4404  4442* 4443* 4468* 4469*
                        4530* 4531* 4532* 4533  4568* 4569
R3      =%000003        1213# 1546* 1563* 1564  1568  1571  1573  1583  1600  1630  1668* 1673* 1674*
                        1675  1677  1888* 1889  1891  1896* 1943* 1945  1947  1950* 2010* 2011  2014
                        2018* 2215* 2229  2230  2368* 2370* 3014* 3017* 3037  3327  3332* 3342  3347*
                        3357  3362* 3372  3377* 3386* 3397* 3408* 3419* 3438  3451  3455  3525* 3695*
                        3708* 3726  3928* 3936* 4099  4111* 4133  4151* 4202* 4204* 4341* 4342  4344
                        4360* 4444* 4463* 4472*
R3PBT   016652          4242* 4250* 4265#
R3PRT   013270          3542* 3543* 3544  3561#
R3PXT   014020          3594* 3595* 3596  3625* 3626* 3627  3678#
R4      =%000004        1214# 1527* 1538* 1683  1684* 2383* 2390* 2889* 2892  2893* 2894  2895* 2898
                        2899* 2900  3016* 3018* 3040  3328  3343  3358  3373  3385* 3396* 3407* 3412*
                        3455* 3456  3480  3493  3495  3497  3501  3513  3523  3565* 3566  3576* 3584*
                        3589* 3600  3611  3613  3618  3629  3633  3641  3644  3651  3654  3690  3691*
                        3697* 3698* 3699* 3702  3770* 3781* 3806  3929* 3934  4100  4110* 4134  4150*
                        4243* 4246  4248  4341  4355* 4356* 4357* 4360  4362* 4369* 4373* 4374* 4375*
                        4445* 4464* 4473*
R5      =%000005        1215# 1522* 1532* 1544* 1575* 1589* 1609* 2382* 2393* 3024* 3035  3052* 3590*
                        3591* 3592  4101  4109* 4135  4149* 4475*
SAR     = 000012        1479# 1904  1917  1929  1944  1958  1962
SAVBF0  012272          3255  3278  3302  3383#
SAVBF1  012316          3261  3284  3307  3394#
SAVBF2  012342          3267  3290  3313  3405#
SAVBF3  012366          3273  3296  3319  3416#
SAVEPC  024454          4127* 4139  4145* 4157  5351#
SAVOLD  024442          2891* 2893  2897* 2899  5346#
SAVPSW  024456          4128* 4138  4146* 4156  5352#
SAVREG= 104002          1260# 3866  3892  4044  4057  4390  4462  4499
SAV2PC  024460          4129* 4137  4147* 4155  5353#
SAV2SW  024462          4130* 4136  4148* 4154  5354#
SCOPE = 104001          1259# 1786  1796  1809  1821  1838  1851  1864  1876  1887  1902  1915  1927
                        1941  1956  1971  1984  1991  2003  2009  2025  2033  2050  2067  2082  2111
                        2139  2159  2177  2201  2575  2588  2599  2610  2623  2637  2650  2663  2679
                        2694  2710  2727  2747  2761  2773  2784  2791  2814  2832  2853
SCOPEB  020332          4589  4597#
```

```
SCOPEC  020266          1302    4587#
SCOPEF  020400          1759*   3739*   4440    4466    4562    4594    4596*   4600*   4608#
SCOPEG  020344          4593    4595    4600#
SCOPI   020306          4592#   4615
SELECT  024434          2086*   2098*   2114*   2127*   2228    5343#
SERFK0  011626          3118    3253#
SERFK1  011642          3120    3259#
SERFK2  011656          3165    3265#
SERFK3  011672          3148    3271#
SERVAA  012664          3457    3480#
SERVAB  012706          3486#   3503
SERVBB  012770          3489    3491    3504#   3675
SERVBC  013030          3511#   3547
SERVCC  013064          3514    3520#   3567    3577    3585
SERVCE  011326          3190#   3215    3229    3242
SERVDX  013116          3521    3527#
SERVFK  012444          3254    3260    3266    3272    3438#
SERVOA  012530          3452    3455#
SERVOB  012730          3481    3493#
SERVOC  013036          3502    3513#
SERVOD  013106          3447    3461    3479    3483    3485    3500    3512    3516    3519    3525#   3529
SERVOO  012514          3439    3451#
SETBF0  012036          3253    3276    3300    3323#
SETBF1  012102          3259    3282    3305    3338#
SETBF2  012152          3265    3288    3311    3353#
SETBF3  012222          3271    3294    3317    3368#
SETED   002700          1528    1539    1585    1602    1681#
SETFKA  016442          4203#   4206
SETLNE= 104007          1265#   2312
SETUPF  006174          2548#   2558    3731    3735
SETUPV  003026          1748#   1767    3733
SETUT0  016360          4182    4189#
SETUT1  016376          4184    4193#
SETUT2  016410          4186    4196#
SETUT3  016422          4188    4199#
SHFBAD  005074          2238    2246#
SHFCUR  006106          2517#   2523
SHFEND  021020          2068    4898#
SHFEOL  005536          2385#   2391
SHFLST  005032          2235#
SHFMES  020022          2085*   2097*   2113*   2126*   4524#
SHFPRT  024440          1758*   5345#
SHFTBF  020722          2034    2036    2051    2053    2179    2180    2214    2215    2217    4836#
SHFTSW  024436          1757*   2141*   2219*   4470    4500    4519    5344#
SHF32   005046          2239#   2262
SHF64   005060          2236    2242#
SHIFT   005004          2088    2102    2116    2131    2227#   2241    2254    2263
SP      =%000006        1216#   1287*   1288*   1289*   1290    1293*   1294*   1295*   1296*   1297    1494*   1651*   2148*
                        2169*   2194*   2264*   3323    3324*   3325*   3326*   3327*   3328*   3338    3339*   3340*   3341*
                        3342*   3343*   3353    3354*   3355*   3356*   3357*   3358*   3368    3369*   3370*   3371*   3372*
                        3373*   3385    3386    3387    3388    3389    3396    3397    3398    3399    3400    3407    3408
                        3409    3410    3411    3418    3419    3420    3421    3422    3430    3433*   3867    3868*   3893
                        3894*   3935*   3939*   4089*   4096*   4097*   4098*   4099*   4100*   4101*   4102*   4103    4107*
                        4108    4109    4110    4111    4112    4113    4114    4127    4128    4129    4130    4131*   4132*
                        4133*   4134*   4135*   4136*   4137*   4138*   4139*   4145    4146    4147    4148    4149    4150
                        4151    4152    4153    4154*   4155*   4156*   4157*   4166    4167*   4218    4220    4271    4272*
```

```
                        4290    4291*   4361*   4386    4387    4389*   4442    4469    4489    4515    4597    4601    4603*
                        4604
SPACE = 104016          1272*   4561    4566    4572
SPACEX  014326          3752*   3754*   3756*   4411*   4412*   4414*   4418*   4420*
SPCHR1  014450          3778    3780    3791*
SPCHR3  014532          3801    3808*
SPCHR4  014540          3810*
SPCHR5  014554          3784    3811    3815*
SRHEOL  005526          2383*   2394
SRHEOS  005634          2417*   2422
SRIDE = 000004          1476*   2037    2054    2070    2086
SRIDO = 000014          1480*   2114
SRODE = 000006          1477*   2098
SRODO = 000016          1481*   2127
SRRVFK  007450          2875    2881*
SRVFOB  007522          2890    2895*
STACK   024136          1494    1651    2148    2169    2194    5254*
START   024430          3108    3333*   3348*   3363*   3379*   3462    3508    3557    3590    3598    5341*
STCOD0  024224          3467*   3631    3638*   3660*   5281*
STCOD1  024226          5282*
STCOD2  024230          5283*
STCOD3  024232          5284*
STERR   013462          3607*
STOP0   024330          3460*   3466*   3663    5309*
STOP1   024332          5310*
STOP2   024334          5311*
STOP3   024336          5312*
STRIPN  017200          4340    4362*   4375
STUNIT  016320          1637    1657    3836    4181*
SWAPEM  024170          1632*   1636*   1652*   1656*   3831    5267*
SWPEXT  014720          3838    3847*
SWR     001324          1330*   1512    1515    1517    1524    1527    1535    1538    1546    1554    1668    2257    2792
                        2942    2946    3014    3016    3542    3594    3605    3625    3669    3681    3729    3851    4031
                        4033    4080    4437    4446    4460    4517    4527    4530    4548    4588    4592    4618
SWR0    001326          1331*
SW00  = 000001          1206*   3544    3596    3605    3627    3681
SW01  = 000002          1205*   3544    3596    3627
SW02  = 000004          1204*
SW03  = 000010          1203*
SW04  = 000020          1202*
SW05  = 000040          1201*
SW06  = 000100          1200*
SW07  = 000200          1199*
SW08  = 000400          1198*
SW09  = 001000          1197*
SW10  = 002000          1196*
SW11  = 004000          1195*   4033
SW12  = 010000          1194*   2257    3851    4080
SW13  = 020000          1193*   4517
SW14  = 040000          1192*   2942    4031
SW15  = 100000          1191*
SYSIT1  010462          3105*
SYSTST  010344          1731    3085*   3087
TAGA    006560          2666    2671*
TAGAB   006630          2679    2687*
TAGAH   007274          2819*   2823
```

```
TAGB    006704          2695    2703#
TAGC    006760          2711    2719#
TAGFA   007010          2730    2734#
TAGFH   007210          2794    2795#
TAGH    007230          2800#   2804
TAGI    007344          2838#   2842
TAGVA   004544          2144    2148#
TAGVB   004410          2090    2104#
TAGVC   004502          2118    2133#
TAGVD   004624          2162    2168#
TAGVE   004672          2194#   2191
TAGVF   004716          2183    2193#
TAGIX   011112          3161#   3181
TAG2X   011246          3143    3180#
TAG3X   011256          3178    3182#
TEMP1   024444          2929*   2961    2968*   4028*   4037*   4103*   4107    4487*   4509*   4513*   4573*   5347#
TEMP2   024446          2930*   2969    2976*   3994*   3995*   4024*   4026*   4035*   4562*   4563*   4565    5348#
TERMTB  015520          3929    4001#
TESTSW  013760          3446    3669#
TESTX   014162          1733    3721#
TEST00  002750          1677    1697#
TEST01  002752          1699#
TEST02  002754          1701#
TEST03  002756          1703#
TEST04  002760          1705#
TEST05  002762          1707#
TEST06  002764          1709#
TEST07  002766          1711#
TEST10  002770          1713#
TEST11  002772          1715#
TEST12  002774          1717#
TEST13  002776          1719#
TEST14  003000          1721#
TEST15  003002          1723#
TEST16  003004          1725#
TEST17  003006          1727#
TEST20  003010          1729#
TEST21  003012          1731#
TEST22  003014          1733#
THEND = 104023          1277#   2305    2464    2480    2495    2539
TITLE   021023          1559    4905#
TKB     001316          1327#   3775
TKS     001314          1326#   3763    3773    4329*
TKSFLG  014330          1316    3761#
TOPC    024162          4218*   4221*   4229    5264#
TPB     001322          1329#   3751*   3787*   3878*   3880*   3883*   3906*   4416*
TPS     001320          1328#   3749    3785    3869    3881    3904    4424
TRANF0  024310          3197    3472*   3657*   3688*   5301#
TRANF1  024312          3209    5302#
TRANF2  024314          3223    5303#
TRANF3  024316          3237    5304#
TRAN0   011706          3114    3276#
TRAN0A  013162          3538    3542#
TRAN0B  013232          3548    3552#
TRAN0C  013262          3545    3553    3559#
TRAN1   011722          3116    3282#
```

```
TRAN2   011736          3163    3288#
TRAN3   011752          3146    3294#
TRNSMT  013130          3277    3283    3289    3295    3533#
TRNSW0  024204          3477*   3507*   3537    3554*   3689*   5273#
TRNSW1  024206          5274#
TRNSW2  024210          5275#
TRNSW3  024212          5276#
TSTA    014204          3722    3728#
TSTB    014214          3727    3731#
TSTNUM  024146          1756*   2553*   3737*   4439    4465    4560    4602*   5258#
TSTSHF  005102          2243    2248#
TSTTKS= 104017          1273#   4587
TSTTPS  017432          4415    4417    4424#
TSWAP   014632          3830#   4083    4088
TTYENB  017062          3025    3865    3891    3993    4027    4077    4327#   4383
TTYIN = 104013          1269#   3765    4338
TTYMTR  002644          1666    1670#
TTYMT1  002652          1669    1673#
TTYSWH  024160          1490*   1492*   1508    1625    1638    1665    1762    3007    3721    3761    4223    4327    4434
                        4458    5263#
TYPECL  015030          3877    3880#
TYPERA  014772          3869#   3870    3879    3885
TYPER1  015014          3872    3876#
TYPER2  015022          3878#
TYPFKR  017622          4435    4486#
TYPMAR  020072          4528    4534#
TYPMES  014754          1311    3865#
TYPT1   016570          4244#   4251
TYPT2   016574          4245#   4247
TYPVTR  017656          4459    4499#
UNITFG  024140          1554*   1630*   1631    1633    1635*   1655*   3835*   3837    3843    4181    4183    4185    4187
                        4189*   5255#
UNITNO  024142          1515*   1516*   1521    1533    1542    1568*   1576    1590    1607    3090    3098    3142    3180
                        3214    3228    4181    5256#
UNLINE= 000370          1456#   4318
UPCASE  020546          2963    4696#
VISEOL= 000212          1465#   2367    2385    2389    3978    4001    4003    4005    4007    4009    4011    4013    4015
VISEOP= 000214          1466#   3980    4002    4004    4006    4008    4010    4012    4014
VISEOS= 000231          1467#   2415    2421    2515    2522    3982    4016
VTALIN  005740          1711    2457#   2466
VTBLAN  005660          1709    2433#   2449
VTBUFF  024500          1903    1916    1928    1942    1957    2301    2384    2414    2513    3122    3125    3129    3132
                        3150    3153    3167    3170    3190    3193    3196    3199    3202    3205    3208    3211    3215
                        3219    3222    3225    3230    3233    3236    3239    3333    3348    3363    3378    3925    4058
                        4063    4168    4173    5366#
VTCAR   001504          1411#   1839*   1845*   1852*   1858*   1865*   1867*   1877*   1889*   1895*   2036*   2053*   2514*
                        2521*   4064*   4202
VTCHAR  005166          1699    2206    2275#   2286
VTCRLF  017350          4400    4403#
VTCSR   001510          1413#   1748    1840*   1844*   1853*   1857*   2058    2330*   2334*   2338*   2342*   2416*   2441*
                        2463*   2479*   2494*   2516*   2536*   2538*   2798*   2817*   2836*   2988*   3042*   3834*   4066*
                        4293*   4302*   4384*   4436*   4472    4511*   4514    4529*   4533*   4605*
VTCSR4=%000004          1222#   1748*   1775*   1777    1797*   1798    1803*   1810*   1812    1815*   1822*   1827*   1866*
                        1879*   1890*   1894*   1904*   1917*   1929*   1944*   1958*   1962*   1974*   1978*   1994*   1998*
                        2013*   2017*   2037*   2038    2041    2044*   2054*   2062*   2070*   2071    2074*   2150    2228*
VTCURS  006042          1717    2506#   2525
```

```
VTDUMP  017760          4510    4515#
VTEOL   005454          1705    2346    2362#   2396
VTEOS   005574          1707    2407#   2424
VTERA   017312          4392#   4402    4405
VTERR   017526          1753    4458#
VTERR1  017610          4471    4475#
VTERR2  017614          4474    4476#
VTER1   017336          4393    4399#
VTER2   017344          4401#
VTFOCU  005766          1713    2473#   2481
VTINT   001514          1415#   4290*   4303*
VTLOGI  003122          1697    1743    1762#   2958
VTLVL   001516          1416#   4292*   4303    4304*
VTMAR   001512          1414#   1749    4473    4475    4534
VTMAR5=%000005          1223#   1749*   1788    1824    1828    1841    1854    1868    1880    1891    1907    1920    1932
                        1947    1963    1975    1995    2014    2055    2237    2244    2253
VTMD1A  005274          2304    2309#   2315
VTMD2A  005342          2330#
VTMD2B  005434          2331    2335    2339    2343    2348#
VTMES   017246          1301    4383#
VTMOD1  005226          1701    2297#   2306
VTMOD2  005322          1703    2307    2324#   2345
VTNORM  017732          4501    4511#
VTODD   006136          1719    2533#   2540
VTREST  003136          1767#   2207
VTRUN   003016          1721    1742#
VTSAR   001506          1412#   1903*   1905*   1916*   1918*   1928*   1930*   1942*   1945*   1957*   1959*   1972*   1992*
                        2011*   2034*   2051*   2068*   2142*   2160*   2180*   2197*   2214*   2230*   2537*   4065*   4174*
VTWORS  006014          1715    2488#   2496
VTOCAR  001410          1370#   1532    1589    4191
VTOCSR  001414          1372#   3100*   3184*
VTOINT  001420          1374#
VTOLVL  001422          1375#
VTOMAR  001416          1373#
VTOSAR  001412          1371#   3088*   3126*
VTI     003142          1754    1773#
VTICAR  001424          1379#   4194
VTICSR  001430          1381#   3101*   3185*
VTIINT  001434          1383#
VTILVL  001436          1384#
VTIMAR  001432          1382#
VTISAR  001426          1380#   3089*   3127*
VTIO    003422          1864#
VTI1    003452          1876#
VTI2    003500          1887#
VTI3    003540          1902#
VTI4    003576          1915#
VTI5    003634          1927#
VTI6    003666          1941#
VTI7    003730          1956#
VT2     003166          1786#
VT2CAR  001440          1388#   4197
VT2CSR  001444          1390#   3095*   3176*
VT2INT  001450          1392#
VT2LVL  001452          1393#
VT2MAR  001446          1391#
```

```
VT2SAR  001442          1389*   3094*   3168*
VT30    003772          1971*
VT21    004024          1984*
VT22    004032          1991*
VT23    004064          2003*
VT24    004072          2009*
VT25    004130          2025*
VT26    004136          2033*
VT27    004206          2050*
VT3     003206          1796*
VT3CAR  001454          1397*   4200
VT3CSR  001460          1399*   3093*   3159*
VT3INT  001464          1401*
VT3LVL  001466          1402*
VT3MAR  001462          1400*
VT3SAR  001456          1398*   3092*   3151*
VT30    004260          2067*
VT31    004312          2082*
VT32    004412          2111*
VT33    004504          2139*
VT34    004564          2159*
VT35    004640          2177*
VT36    004730          2201*
VT4     003236          1809*
VT5     003262          1821*
VT6     003322          1838*
VT7     003362          1851*
WAITSY= 104021          1275*   1811    1823    1906    1919    1931    1946    1961    1973    1993    2012    2035    2052
                        2069    2145    2146    2181    2189    2216    2231
WHY     017144          4343    4350*
WORD2   017154          4354*   4367
WORST   013342          3498    3582*
WORST1  013350          3449    3583*
X       = 000025        1229*   2575    2576*   2588    2589*   2599    2600*   2610    2611*   2623    2624*   2637    2638*
                        2650    2651*   2663    2664*   2678    2679*   2694    2695*   2710    2711*   2727    2728*   2747
                        2748*   2761    2762*   2773    2774*   2791    2792*   2814    2815*   2832    2833*   2853    2854*
XASEMB  017104          1314    4338*
XASEM1  017116          4341*   4359
XASEXT  017150          4349    4352*
XBINDE  016546          1321    4240*
XBUF0   001570          1441*   3511*   3549*   3559*   3690*
XBUF1   001572          1442*
XBUF2   001574          1443*
XBUF3   001576          1444*
XCSR0   001560          1437*   3138*
XCSR1   001562          1438*   3141*
XCSR2   001564          1439*   3175*
XCSR3   001566          1440*   3158*
XDELAY  015540          1305    4024*
XDLAYL  015550          1317    4026*
XDLAY1  015566          4029*   4036    4038
XDLAY2  015630          4034    4039*
XDLAY3  015614          4030    4035*
XENDT   015732          1307    4076*
XEOSBF  015656          1306    4057*
XFER0   024370          3464*   3482*   3661    5325*
```

```
XFER1   024372          5326#
XFER2   024374          5327#
XFER3   024376          5328#
XGETRG  016176          1304    4145#
XINT0   001600          1445#   1544    1609    3114*
XINT1   001604          1447#   3116*
XINT2   001610          1449#   3163*
XINT3   001614          1451#   3146*
XLDLNA  015644          4047#   4049
XLDLNE  015632          1309    4044#
XLOOP   014260          3739#   3740
XLVL0   001602          1446#   3115*
XLVL1   001606          1448#   3117*
XLVL2   001612          1450#   3164*
XLVL3   001616          1452#   3147*
XNULL   017360          1319    4411#
XNULL1  017404          4416#   4419
XORERR  020430          4612    4616#
XORFLG  020456          4545    4590    4623#
XORHLT  020264          4547*   4575    4582#
XORT    020404          4591    4611#
XPRELD  016252          1310    4164#
XPRT    020224          4568#   4574
XPRTA   020232          4522    4526    4535    4567    4570#
XSAVRG  016122          1303    4127#
XSPACE  014276          1315    3749#   3753
XSTLNE  017024          1308    4312#
XTHEND  015726          1320    4075#   4085
XTTYIN  014350          1249    1312    3770#   3822
XWAITS  015476          1318    3993#
Y       = 000037        1230#   1786    1787#   1796    1797#   1809    1810#   1821    1822#   1838    1839#   1851    1852#
                        1864    1865#   1876    1877#   1887    1888#   1902    1903#   1915    1916#   1927    1928#   1941
                        1942#   1956    1957#   1971    1972#   1984    1985#   1991    1992#   2003    2004#   2009    2010#
                        2025    2026#   2033    2034#   2050    2051#   2067    2068#   2082    2083#   2111    2112#   2139
                        2140#   2159    2160#   2177    2178#   2201    2202#
.       = 024502        1233#   1238#   1240#   1243#   1248#   1250#   1286#   1491    1503    1557    1634    1676    1763
                        1779    1790    1801    1813    1825    1831    1842    1855    1869    1881    1892    1908    1921
                        1933    1948    1964    1976    1996    2015    2039    2042    2056    2059    2072    2151    2166
                        2167    2186    2192    2205    2249    2251    2256    2258    2280    2284    2310    2351    2371
                        2386    2580    2592    2603    2613    2615    2627    2629    2640    2644    2653    2656    2668
                        2670    2685    2686    2701    2702    2717    2718    2732    2737    2741    2750    2754    2765
                        2776    2780    2802    2806    2821    2825    2840    2844    2957    2877    2935    2956    2958
                        2978    2982    3029    3109    3487    3534    3574    3614    3619    3653    3664    3696    3732
                        3734    3750    3762    3764    3792    3795    3818    3824#   3832    3850    3852    3882    3897
                        3905    3937    3942    3949    3983    3996    4025    4032    4062    4118    4171    4224    4329
                        4345    4364    4371    4425    4447    4503    4546    4553    5235#   5249#   5294#   5296#
```

FKTX	1255#	2575	2588	2599	2610	2623	2637	2650	2663	2679	2694	2710	2727	2747	2761
	2773	2791	2814	2832	2853										
TA	1255#	1786	1796	1809	1821	1838	1851	1864	1876	1887	1902	1915	1927	1941	1956
	1971	1984	1991	2003	2009	2025	2033	2050	2067	2082	2111	2139	2159	2177	2201
TS	1254#	2575	2588	2599	2610	2623	2637	2650	2663	2678	2694	2710	2727	2747	2761
	2773	2791	2814	2832	2853										
VTX	1255#	1786	1796	1809	1821	1838	1851	1864	1876	1887	1902	1915	1927	1941	1956
	1971	1984	1991	2003	2009	2025	2033	2050	2067	2082	2111	2139	2159	2177	2201

```
        3673  3674  3739  3754  3771  3772  3830  3834  3944  3952  3986  4026  4064  4116  4164
        4189  4240  4275  4283  4294  4304  4330  4362  4384  4397  4411  4416  4505  4600
CLRB    2369  3511
CMP     1573  1633  1653  1675  1800  1812  1824  1841  1854  1891  1907  1920  1947  1975  1995
        2014  2235  2237  2242  2244  2253  2255  2261  2373  2579  2736  2740  2884  2886  2955
        2963  2971  2977  2981  3090  3098  3108  3142  3180  3214  3228  3520  3544  3592  3596
        3627  3731  3783  3948  3955  4183  4185  4187  4205  4342  4344  4366  4594  4597  4616
CMPB    2385  2949  3037  3040  3248  3456  3480  3493  3495  3497  3501  3513  3552  3600  3633
        3641  3644  3777  3779  3791  3794  3800  3808  3810  3815  3871  3876  3957  3959  3961
        3963  3965  3967  3969  3971  3973  3975  3978  3980  3982  4181  4392  4399
COM     3052  3482  3977
COMB    3583
DEC     1529  1540  1551  1587  1604  1622  1685  2190  2370  2390  2393  2418  2446  2518  3517
        3546  3700  3708  3752  3804  3908  4061  4170  4358  4370  4418  4573
EMT     1258  1259  1260  1261  1262  1263  1264  1265  1266  1267  1268  1269  1270  1271  1272
        1273  1274  1275  1276  1277  1278
HALT    1292  1510  1513  1523  1526  1534  1537  1545  1553  1667  3013  3015  3728  4105  4225
        4448  4476  4579
INC     1636  1656  1742  2101  2130  2165  2185  2227  2314  2372  2463  2479  2494  2538  2612
        2626  2667  2684  2700  2716  2731  2801  2803  2820  2822  2839  2841  2881  2968  2976
        2979  2991  3045  3093  3095  3100  3101  3428  3460  3522  3535  3556  3571  3638  3643
        3648  3649  3657  3687  3688  3689  3782  3835  3930  3931  3936  3947  3985  3995  4035
        4037  4048  4075  4117  4245  4250  4395  4441  4467  4504  4547  4556  4563  4596
INCB    1586  1603  1621  4412  4420
JMP     1251  1252  1253  1297  1519  1677  1743  2207  2858  2859  3255  3261  3267  3273  3278
        3284  3290  3296  3334  3349  3364  3379  3447  3448  3449  3665  3675  3741  3793  3799
        3819  3987  4121  4234  4352  4491  4538  4599  4622
JSR     1528  1539  1585  1602  1637  1657  1659  1660  1767  2087  2088  2099  2102  2115  2116
        2128  2131  2143  2149  2161  2171  2182  2195  2239  2282  2304  2309  2331  2335  2339
        2343  2349  2558  2665  2672  2689  2705  2721  2729  2735  2874  2932  2984  3025  3026
        3124  3128  3152  3169  3253  3254  3259  3260  3265  3266  3271  3272  3276  3277  3282
        3283  3288  3289  3294  3295  3300  3301  3305  3306  3311  3312  3317  3318  3446  3453
        3539  3607  3615  3620  3623  3733  3735  3836  3865  3891  3993  4027  4077  4083  4088
        4249  4340  4383  4415  4417  4490  4516  4522  4526  4535
MOV     1287  1289  1296  1490  1493  1494  1500  1501  1505  1506  1507  1512  1515  1521  1522
        1524  1527  1532  1533  1535  1538  1542  1543  1544  1546  1547  1548  1549  1550  1554
        1564  1568  1575  1576  1583  1589  1590  1600  1607  1608  1609  1617  1618  1619  1620
        1630  1631  1651  1658  1661  1668  1681  1682  1683  1687  1689  1690  1748  1749  1752
        1753  1754  1755  1756  1775  1777  1788  1797  1798  1810  1822  1828  1839  1840  1852
        1853  1865  1866  1877  1879  1888  1889  1890  1903  1904  1905  1916  1917  1918  1928
        1929  1942  1943  1944  1946  1957  1958  1959  1962  1972  1974  1992  1994  2010  2011
        2013  2034  2036  2037  2051  2053  2054  2068  2070  2084  2085  2086  2097  2098  2113
        2114  2126  2127  2140  2142  2148  2160  2163  2164  2169  2178  2179  2180  2187  2194
        2214  2215  2217  2219  2228  2230  2232  2240  2275  2297  2301  2303  2313  2324  2330
        2334  2338  2342  2362  2368  2382  2383  2384  2407  2413  2414  2416  2420  2433  2440
        2441  2457  2473  2488  2506  2512  2513  2514  2515  2516  2520  2521  2522  2533  2537
        2548  2550  2551  2552  2553  2577  2582  2594  2611  2617  2624  2631  2664  2679  2680
        2681  2683  2695  2696  2697  2698  2699  2711  2712  2713  2714  2715  2767  2794  2799
        2808  2818  2827  2833  2837  2868  2889  2891  2895  2897  2901  2902  2924  2927  2967
        2975  3019  3022  3085  3087  3088  3089  3092  3094  3105  3106  3110  3111  3112  3113
        3114  3115  3116  3117  3118  3119  3120  3121  3122  3123  3125  3126  3127  3129  3130
        3131  3132  3133  3134  3135  3136  3137  3138  3139  3140  3141  3144  3145  3146  3147
        3148  3149  3150  3151  3153  3154  3155  3156  3157  3158  3159  3160  3161  3162  3163
        3164  3165  3166  3167  3168  3170  3171  3172  3173  3174  3175  3176  3177  3182  3183
        3184  3185  3186  3187  3190  3191  3193  3194  3196  3197  3199  3200  3202  3203  3205
        3206  3208  3209  3211  3212  3216  3217  3219  3220  3222  3223  3225  3226  3230  3231
```

```
            3233  3234  3236  3237  3239  3240  3246  3323  3324  3325  3326  3327  3328  3329  3330
            3333  3338  3339  3340  3341  3342  3343  3344  3345  3346  3347  3348  3353  3354  3355
            3356  3357  3358  3359  3360  3361  3362  3363  3368  3369  3370  3371  3372  3373  3374
            3375  3376  3377  3378  3383  3384  3385  3386  3387  3388  3389  3394  3395  3396  3397
            3398  3399  3400  3405  3406  3407  3408  3409  3410  3411  3416  3417  3418  3419  3420
            3421  3422  3462  3488  3505  3506  3507  3508  3510  3542  3557  3565  3566  3576  3589
            3590  3594  3598  3625  3666  3676  3693  3694  3695  3707  3726  3729  3737  3740  3751
            3760  3765  3781  3805  3867  3880  3883  3893  3895  3896  3907  3925  3926  3927  3929
            3930  3933  3950  3994  4024  4028  4045  4046  4058  4059  4063  4065  4066  4082  4096
            4097  4098  4099  4100  4101  4102  4103  4104  4106  4107  4108  4109  4110  4111  4112
            4113  4114  4127  4128  4129  4130  4131  4132  4133  4134  4135  4136  4137  4138  4139
            4145  4146  4147  4148  4149  4150  4151  4152  4153  4154  4155  4156  4157  4165  4166
            4168  4169  4173  4174  4190  4191  4193  4194  4196  4197  4199  4200  4201  4202  4203
            4204  4218  4220  4242  4243  4244  4271  4273  4280  4281  4282  4290  4292  4301  4303
            4329  4339  4341  4360  4396  4397  4391  4414  4439  4440  4442  4444  4445  4463  4464
            4465  4466  4468  4472  4473  4475  4486  4487  4488  4489  4506  4507  4508  4509  4512
            4513  4514  4515  4521  4525  4530  4534  4562  4568  4569  4598  4601  4602  4604  4611
            4612  4614  4617
MC.B        1577  1591  1610  2367  2387  2389  2415  2421  2439  2443  2444  2941  2954  2960  2992
            3014  3016  3034  3046  3051  3053  3247  3429  3430  3431  3432  3455  3463  3490  3504
            3518  3523  3524  3548  3559  3582  3584  3635  3636  3637  3650  3651  3652  3654  3655
            3680  3702  3705  3706  3710  3711  3787  3842  3878  3901  3906  3934  3943  3946  4047
            4060  4172  4255  4258  4312  4314  4316  4318  4320  4401  4403
NOP         1496  1498  1497  1498  1499  1511  1514  1525  1536  1773  1774  2060  2104  2133  2293
            2311  2360  2564  2565  4217
RESET       1644  1776  1787  1878  1960  1985  2004  2026  2083  2112  2153  2170  2566  2601  2654
            3086  4115  4215
RTI         1896  1953  2018  3697  3698  3699  3898  3899  3900
RTS         2904  2983  3055  3390  3401  3412  3423  3755  3766  3814  3846  3854  3875  3911  3997
            4039  4051  4068  4086  4090  4140  4158  4175  4252  4321  4346  4398  4422  4449  4479
            4606
RTS         1692  1760  2220  2259  2265  2316  2353  2554  3250  3434  3526  3536  3541  3551  3558
            3560  3602  3609  3656  3677  3712  3833  3855  3938  3940  3984  4207  4256  4259  4276
            4284  4295  4305  4331  4365  4372  4426  4581
SLB         1289  1504  4221  4222  4246  4388  4443  4469  4550
SWAB        2234
TRAP        1257
TST         1290  1502  1508  1517  1556  1571  1625  1638  1665  1762  1830  1868  1980  1932  1963
            2100  2129  2150  2188  2229  2567  2602  2655  2763  2775  2778  2779  2843  2944  2946
            2951  2957  2961  2969  2989  3007  3035  3043  3048  3440  3442  3444  3451  3458  3484
            3486  3499  3515  3533  3537  3603  3611  3631  3646  3661  3663  3667  3669  3671  3721
            3761  3802  3806  3831  3837  3849  3923  3941  3953  4029  4078  4084  4223  4253  4327
            4354  4363  4434  4437  4446  4458  4460  4470  4500  4502  4519  4545  4552  4575  4590
            4613
TSTB        2388  2749  2764  2805  2824  2882  2939  2948  3032  3438  3509  3629  3749  3763  3773
            3785  3817  3869  3881  3884  4404  4424
WAIT        2876  2934  3028
.ABS.       1181
.ASCII      4905  4916  4924  4934  4938  4943  4953  4960  4967  4971  4975  4980  4986  4990  4997
            5009  5021  5027  5032  5034  5041  5049  5058  5064  5070  5078  5082  5089  5098  5102
            5107  5113  5114  5116  5119  5122  5127  5134  5142  5148  5153  5157  5161  5165  5170
            5175  5179  5182  5186  5192  5195  5198  5203  5208  5213  5218  5222  5226  5227  5231
            5236  5237  5238  5244
.BYTE       2411  2412  2437  2438  2461  2462  2477  2478  2492  2493  2510  2511  3856  4001  4002
            4003  4004  4005  4006  4007  4008  4009  4010  4011  4013  4013  4014  4015  4016  4630
            4631  4632  4633  4634  4635  4636  4637  4638  4639  4640  4641  4642  4643  4647  4649
```

```
            4649    4650    4651    4652    4653    4654    4655    4656    4657    4658    4659    4660    4661    4665    4666
            4667    4668    4669    4670    4671    4672    4673    4674    4675    4676    4677    4681    4682    4683    4684
            4685    4686    4687    4698    4689    4690    4691    4692    4696    4697    4698    4699    4700    4701    4702
            4703    4704    4705    4706    4707    4708    4709    4713    4714    4715    4716    4717    4718    4719    4720
            4721    4722    4723    4724    4725    4726    4727    4731    4732    4733    4734    4735    4736    4737    4738
            4739    4740    4741    4742    4743    4747    4748    4749    4750    4751    4752    4753    4754    4755    4756
            4757    4758    4762    4763    4764    4765    4766    4767    4768    4769    4770    4771    4772    4773    4774
            4775    4779    4780    4781    4782    4783    4784    4785    4786    4787    4788    4789    4790    4791    4792
            4793    4797    4798    4799    4800    4801    4802    4803    4804    4805    4906    4807    4808    4809    4813
            4814    4815    4816    4817    4818    4819    4820    4921    4822    4823    4824    4836    4837    4838    4839
            4840    4841    4842    4843    4844    4845    4846    4847    4848    4849    4850    4851    4852    4953    4954
            4855    4856    4857    4859    4859    4960    4861    4862    4863    4864    4865    4966    4867    4868    4869
            4870    4871    4872    4873    4874    4975    4876    4877    4878    4979    4880    4881    4882    4883    4894
            4885    4886    4887    4898    4889    4890    4891    4892    4893    4894    4895    4996    4897    4898    4899
            4904    4995    4996    5008    5019    5043    5044    5045    5048    5063    5366    5367
.ENABL      1181
.END        5370
.EVEN       4017    4825    5047    5235    5249
.LIST       1181    1254    1787    1797    1810    1822    1839    1852    1865    1877    1888    1903    1916    1929    1942
            1957    1972    1985    1992    2004    2010    2026    2034    2051    2068    2083    2112    2140    2160    2178
            2202    2576    2589    2600    2611    2624    2638    2651    2664    2679    2695    2711    2728    2748    2762
            2774    2792    2815    2833    2854
.MACRO      1254    1255
.NLIST      1181    1254    1787    1797    1810    1822    1839    1852    1865    1877    1888    1903    1916    1929    1942
            1957    1972    1985    1992    2004    2010    2026    2034    2051    2068    2083    2112    2140    2160    2178
            2202    2576    2589    2600    2611    2624    2638    2651    2664    2679    2695    2711    2728    2748    2762
            2774    2792    2815    2833    2854
.REM           1
.REPT       1234
.TITLE      1192
.WORD       3561    3678    4265


 ERRORS DETECTED:  0
 DEFAULT GLOBALS GENERATED:  0

 *DBVTAC,DBVTAC.SEQ/SOL/CRF/DS:ERFZ/EN:ABS=DSKM:DBVTAC.P11
 RUN-TIME:  16 34 7 SECONDS
 RUN-TIME RATIO: 169/59=2.8
 CORE USED:  14K  (28 PAGES)
```